Deep-Sea Seafloor Shape Reconstruction from Side-Scan Sonar Data for AUV Navigation

Philipp Woock
Fraunhofer Institute of
Optronics, System Technologies and
Image Exploitation IOSB
Fraunhoferstr. 1
76131 Karlsruhe, Germany
Email: philipp.woock@iosb.fraunhofer.de

***Abstract*—Dead-reckoning navigation in the deep sea is subject to errors due to accumulation of sensor inaccuracies. As no global referencing method exists for the deep sea like, e.g., GNSS (global navigation satellite system) for land or airborne vehicles other referencing solutions need to be employed. SLAM (Simultaneous Localization And Mapping) is a technique that exploits significant environmental features to reduce the positioning error of a vehicle and to simultaneously build a map of the mission environment. It is crucial for SLAM methods to recognize places that have been visited before. In many cases this is done by extracting salient features from the environment. Obtaining those landmarks from side-scan sonar data is a challenging task as the side-scan sonar data does not consist of spatial information but rather represents an echo amplitude over time.**

In Coiras et al. ([1], [2]) it is shown how a seafloor shape can be estimated from side-scan sonar data by inversion and regularization. In this paper their method is extended to allow arbitrary vehicle motion.

The results of the work will be presented in examples of synthetic and real data.

I. Introduction

In the TIETeK project[1] an AUV (Autonomous Underwater Vehicle) is developed which is equipped with frequency-switchable side-scan sonars. The goal is to perform SLAM on the side-scan data acquired by the vehicle which means to create a map of the seafloor and localize the vehicle within this map.

To perform SLAM it is necessary to extract features (often also called landmarks) from the environment. To achieve that in a robust way it is advisable to acquire more sophisticated landmarks as they are not easily mistaken for another one. Since the sonar shadows in a 2D sonar image may lead to wrong feature associations, it is useful to utilize a three-dimensional description of the environment to reliably re-identify an already known spot (see [3]).

Deducing three-dimensional shape information from the seafloor surface out of side-scan sonar data poses a so-called ill-posed inverse problem which means that (possibly infinitely) many explanations for a given sonar echo exist.

Coiras et al. developed methods to estimate seafloor geometry from side-scan images [1], [2]. For the reconstruction in [1] the scan lines from a side-scan sonar equipped AUV are first ground range corrected and then stacked on top of each other to create a 2D image. The obtained 2D sonar image is then used to perform an estimation of the seafloor shape. This is done with an expectation-maximization-based method while additionally employing a multi-resolution approach. This method works well for a vehicle driving on a straight path without height changes at a constant speed.

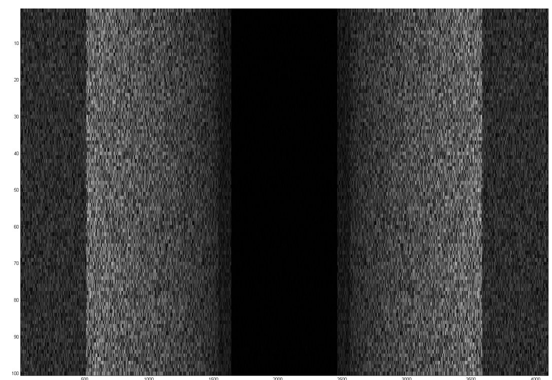

Fig. 1. Synthetic input sonar image. The image has multiplicative noise applied to mimic true sonar image errors.

A synthetic example side scan sonar image showing an ascending slope per line in across-track direction ending with a very steep descent (hence forming a broad groove) is shown in Figure 1 (top).

However, simply stacking scan lines of a vehicle that performs a different motion than going steadily straight ahead introduces distortions in the created 2D image (see, e.g., [4] [5]). These errors are propagated to the estimated seafloor shape and impede a subsequent robust feature extraction.

We extend this approach by taking into account the ego-motion of the vehicle. This is done by means of a registration of the scan lines in 3D space. In order to achieve this we adopt the multi-resolution inversion method described in [1] but modify it to do the estimation bearing in mind the three-dimensional positions and poses of the vehicle at the times the sonar lines were recorded.

[1] http://www.tietek.de

II. Existing method

Coiras' inversion method is an iterative multi-resolution method. It operates by stacking the ground-range corrected side-scan sonar lines on top of each other to yield a 2D image. Then, a surface geometry, a reflectivity pattern and a sonar beam pattern are estimated as parameters which might have produced this particular sonar echo. Using these parameters, a simple forward model creates an estimated sonar image. The estimation is iteratively refined according to the difference between estimated and true sonar image.

Assumptions to keep the model computationally tractable in their approach are the use of a knife-edge shaped sonar beam that has a width in along-track direction of zero. Further, they use the Lambertian scattering model to calculate the surface brightness and additionally assume sound speed isovelocity.

In order to avoid solutions of the estimation that fit mathematically but do not represent real-world conditions, so-called regularization steps are included: It is assumed that the beam pattern is identical for all sonar lines as they have been recorded by the same sensor which should not change its send/receive characteristics too quickly over time. However, the radial decay and time-varying gain and other effects are also grouped into the beam profile variable which means that it should not be completely fixed but allowed to undergo small variations. Another assumption is that in shadowed areas the seafloor reflectivity is similar to the neighboring ensonified areas. The reasoning for that decision is as follows: An area which appears dark in the sonar echo has not necessarily low reflectivity but could just as well lie in a sonar shadow. That's why there is a classification into shadowed and non-shadowed areas as an additional input to the estimation. The reflectivity in the shadowed areas is then interpolated from the neighboring areas. All in all, regularization means to choose a plausible explanation that is in accordance with a priori knowledge or suffices certain constraints to the parameters like the ones mentioned above or additional ones like, e.g., smoothness constraints for the geometry and/or for the reflectivity pattern.

The multi-resolution part of the method means that the estimation is applied first to a subsampled version of the sonar image, therefore recovering lower spatial frequencies first. After convergence of estimation and measurement, the next resolution stage is initialized with an upsampled version of the estimation parameters in order to provide a better starting point for the optimization.

As the side-scan sonar lines are merely stacked to a 2D image, the method is unable to cope with a vehicle that is not going straight ahead. Motion of real AUVs consists not only of going straight-ahead but usually there are turns (yawing), as well as pitch and roll motion. Just stacking the sonar lines on top of each other in such cases introduces distortions and other artifacts which make subsequent feature extraction unnecessarily difficult.

III. Extension of the method

From the vehicular ego-motion estimation a 3D position and orientation of the vehicle is known for each sonar scan line. Therefore, the line relief reconstruction obtained can be correctly embedded in a 3D environment model. As the modified estimation technique is based on 3D coordinates, correct neighborhood relations will be used to perform the estimation task and distortions are avoided.

After having obtained an undistorted 3D surface representation of the environment, it is possible to establish features on this data which allow robust recognition of previously visited places.

In [1] the brightness of a surface patch is estimated through a Lambertian scattering model and is therefore dependent on the inclination angle of the surface patch to the sonar sensor. The height gradients of the relief represent that information and in [1] the height values of the neighboring pixels $\frac{\partial Z}{\partial x}$ and $\frac{\partial Z}{\partial y}$ are used to calculate them as finite differences.

But with arbitrary vehicle motion, a sonar sample in the neighborhood of the stacked line in the image grid does not necessarily belong to the nearest neighboring surface patch in space. Depending on the vehicle position at the time of recording the signal, another sample from a different line could be much nearer. See Figure 2. Therefore, it is essential to consider the true locations of the measurements in space.

The estimation is done for each line and the samples are inserted into a 3D point cloud according to the vehicle motion inertial sensor measurements. The gradients are then obtained through nearest neighbor search on the point cloud[2].

In order to minimize side-effects, the gradients in across-track/range direction (x axis in [1]) are still calculated between the sonar line samples. Obtaining the gradients in along-track direction (y axis in [1]) is done by estimating the local normal vector from the nearest neighbor sonar samples from the other lines. The sonar line itself is not part of the query for the along-track gradients due to the fact that the spatially nearest neighbors would most likely be the across-track sonar samples on the line itself as the samples within a line are in most cases spatially much closer than samples from other lines.

The obtained surface normal is then projected onto a plane orthogonal to the across-track direction of the line. That way, the unwanted vector components of the normal vector which point in across-track direction are removed. This is done because they are already considered in the across-track gradient calculated directly on the line.

A. Compensation for yaw motion

Incorporating yaw motion of the vehicle is done in a straightforward way: The only thing done is a rotation of the estimated line relief in space according to the yaw angle, everything else remains the same. This normally leads to uneven sample coverage: dense in the inner side of a turn and sparse in the outer part. It is clear that the gradients built from "nearest" neighbors that are in reality very far away can only give a very rough estimate of the seafloor geometry. In the original method [1] the same problem exists as the vehicle speed is not taken into account and the sonar lines may also sample the floor sparsely at higher speeds.

[2] http://www.pointclouds.org

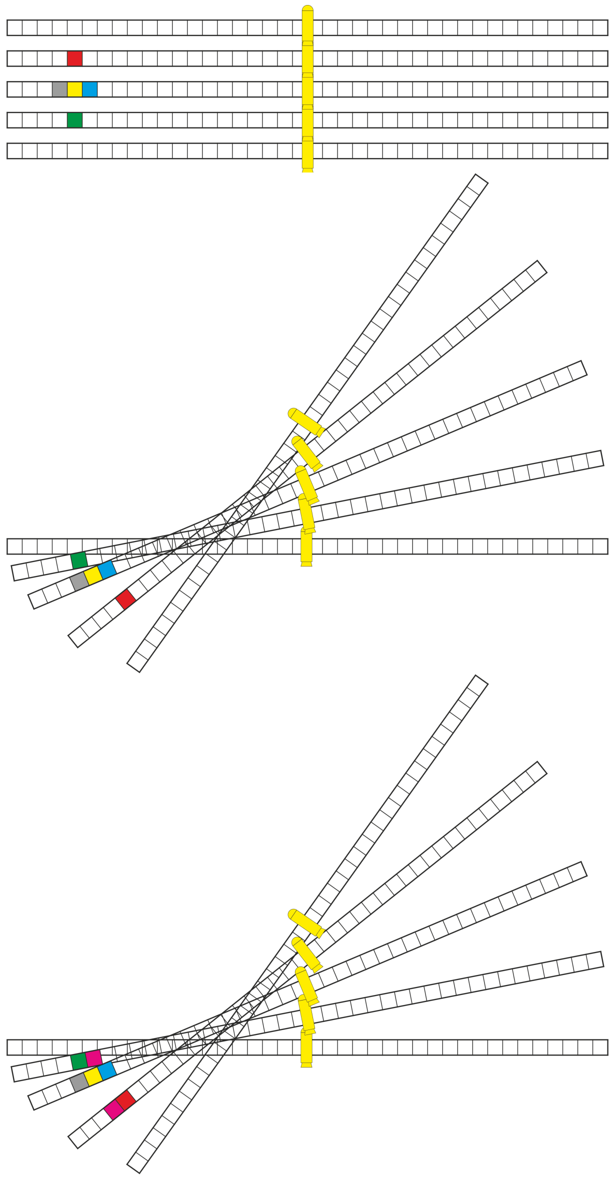

Fig. 2. Case of going straight ahead where the neighbors in the grid are the nearest neighbors in space (top). Sonarlines aligned in space for a curved vehicle trajectory (middle). Note that the upper neighbor (red) in the grid has become the lower neighbor and vice versa. Samples that are spatially nearest neighbors are colored in magenta (bottom). The nearest neighbors do not necessarily lie on neighboring lines in the original grid.

B. Compensation for pitch motion

Pitching of the vehicle has two effects to be considered: First, the detected FBR in the side-scan sonar image f is no more directly related to the true vehicle altitude above ground h (Figure 3) and therefore the usage of the first bottom return (FBR) as an altitude sensor has to take this into account for vehicle ego-motion estimation. Second, there is a displacement d of the ensonified area that causes the line to end up in a different neighborhood in space.

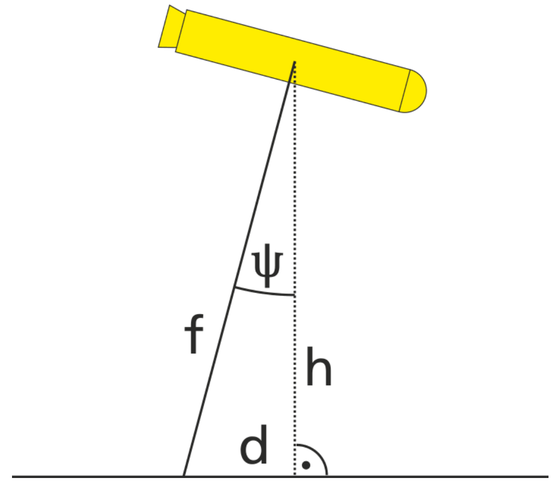

Fig. 3. Scheme drawing of AUV pitch motion

$$h = f \cdot \cos\psi \tag{1}$$

$$d = f \cdot \sin\psi \tag{2}$$

As the originally proposed estimation method reconstructs seafloor relief height values which point towards the vehicle the same is done here: The whole reconstructed line relief is tilted according to the pitch angle. It makes no difference to the surface estimation whether the vehicle is tilted to the seafloor or if the seafloor is sloping towards the vehicle. These cases can only be distinguished if inertial measurements of the vehicular motion are available. The technique of registering the relief estimation into 3D space inherently handles the fact well that the sensor is ”looking forward” respectively ”looking back” when the pitch angle changes.

C. Compensation for roll motion

When an AUV performs roll motion the effect to the side-scan echo is twofold: First, if the roll angle ϱ is greater than the inclination angle of the inner part of the beam γ, there will be simultaneous echoes from the seafloor parts located to both sides of the nadir normal (see Figure 4). This is a case of layover, the sonar echo are superposed additively. Those echoes are hard to interpret and hence the affected samples are discarded. Only the rest of the sonar line where there is presumably only a single echo is used[3]. Second, the inclination angle to the ground of the side-scan beam on the other side becomes larger than γ and therefore the ground range correction has to be modified to take that effect into account.

In the simplified case where $\varphi = 90°$ the ground range correction is done through Pythagoras' rule and yields $r_g = \sqrt{r_s^2 - b^2}$. However, taking roll motion into account, the first return stems from a non-perpendicular direction. Therefore the correction is done using the red triangle in Figure 4 (top) and after applying the cosine theorem one obtains Equation (4).

[3]Of course layover may happen in other parts of the echo too but given the geometric configuration this part is highly likely to contain layover.

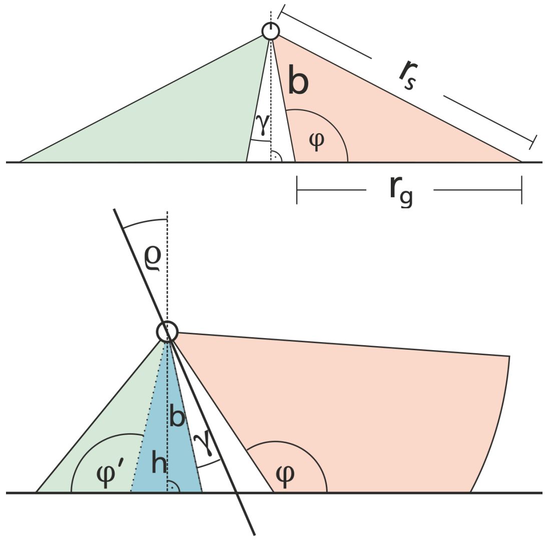

Fig. 4. Scheme drawing of AUV roll motion and extended ground range correction

$$r_s^2 = b^2 + r_g^2 - 2br_g \cos\varphi \tag{3}$$

$$\Rightarrow \quad r_g = b\cos\varphi + \sqrt{r_s^2 - b^2 \sin^2\varphi} \tag{4}$$

The angles φ and φ' used for ground range correction are given as:

$$\varphi = 90^\circ + \varrho + \gamma$$
$$\varphi' = 90^\circ + \varrho - \gamma$$

To find the sonar sample b in the sonar data line that corresponds to the end of the layover region one calculates

$$b = \frac{h}{\cos(\varrho - \gamma)}$$

The part from sample b to the end of the data line (maximum distance) is then used to create the image in ground range coordinates.

In the case where the roll angle is smaller than the beam inclination angle $\varrho < \gamma$, no layover effects occur due to roll motion and φ and φ' are given as

$$\varphi = 90^\circ + \varrho + \gamma$$
$$\varphi' = 90^\circ - \varrho + \gamma$$

D. Compensation for heave motion

Assuming that the first bottom return has been detected properly, there is no further action necessary to compensate for heave motion. When flying at low altitude it may happen that all echoes have arrived earlier than the maximum range of the sonar. This yields a more or less silent signal after the last bottom return but does not change the way the signal is treated. The only difference is a potential change in spatial resolution as the time resolution of the sonar does not change over time. That way the full angular coverage area of the sonar beam is packed into fewer time samples than at a greater altitude.

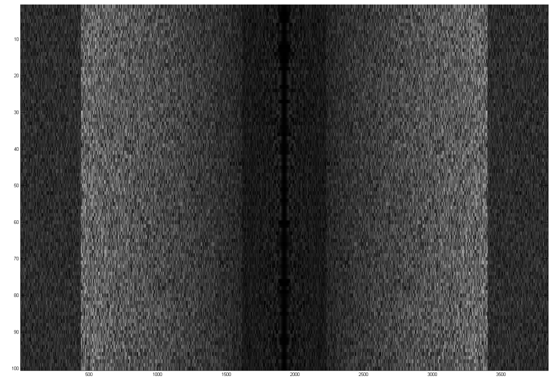

Fig. 5. The synthetic image after ground range correction. The image is also cut off a bit at the end to obtain a resolution that suits the multi-resolution approach.

E. Changes in the regularization steps

The detection of shadowed areas is done only per-line as methods from 2D image processing can not be easily transferred to arbitrary 3D geometry. However, it is possible to set a shadow attribute for affected points in the point cloud and to try to define contiguous shadowed areas for the whole estimation window later on. This would of course not help the estimation process itself but could be helpful for the subsequent processing in such a way that features in shadowed areas would not be used (or at least be weighted lower) in the matching stage.

The beam pattern normalization was not changed compared to [1], it is still assumed that the pattern stays mainly constant (or changes only very slowly) over time.

IV. Results

A. Synthetic results

The artificial example image 1 was processed with an artificial straight-ahead vehicle trajectory (as it was the case in [1]) with 1 m/s. For the subsequent processing the image has been converted to ground range coordinates (see Figure 5) with the same first bottom return (and therefore the same vehicle altitude) for each line. The results of the method applied to that synthetic image is shown in Figure 6. It is visible that the method can deal with sonar shadows in such a way that the reconstructed surface is tangential to the beam in dark areas. Why a reconstruction of shadowed areas cannot be better than this is explained in [3]. Compared to the results [1] the reconstruction is noisier which may be due to the regularization of the reflectivity which is only per-line and no longer done in the image plane.

B. Results with real data

The method was also applied to real data which was obtained using a small boat on a river. The sonar sensor was a Humminbird Side Imaging sonar with custom built sample conversion. The vehicle orientation data was recorded with an Xsens inertial measurement unit (IMU) that was mounted on the same pole as the sonar sensor. The position data was obtained by means of a GPS sensor (to avoid the errors that

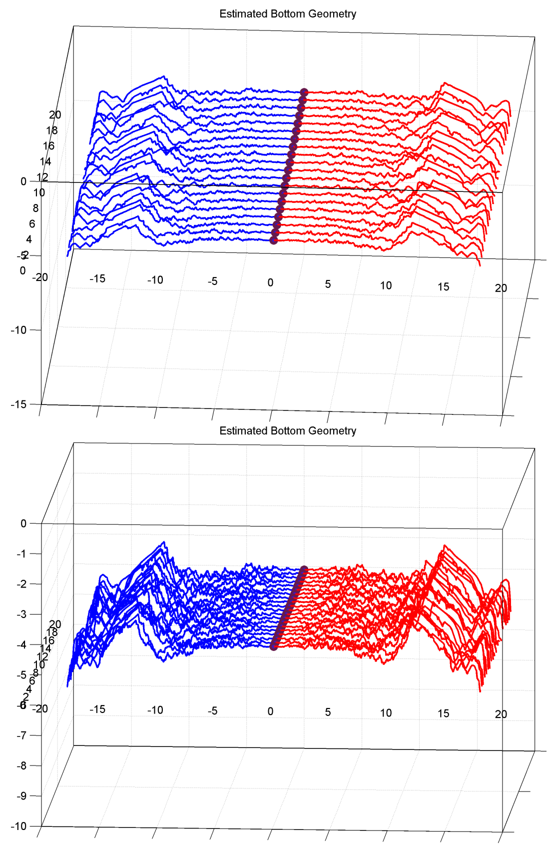

Fig. 6. Reconstructed synthetic surface geometry from different views (axes are given in meters). The different sonar channels are shown in red (right) and blue (left). After the bright ridge the model assumes the surface to be tangent to the sonar beam. The vehicle trajectory is shown as projection to the ground with the recording times as dot (dark red).

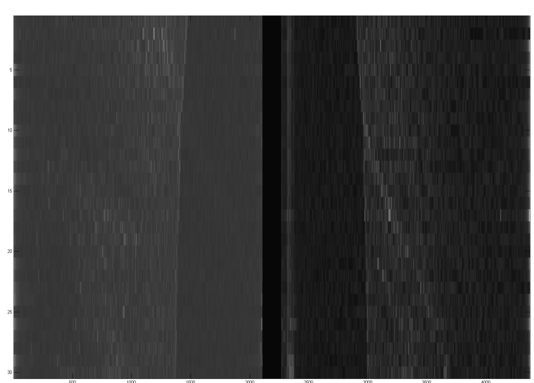

Fig. 7. Ground range corrected data of the river seabed.

occur by double integration of acceleration measurements). Unfortunately, the river seabed was quite poor in features (see Figure 7). A part with strong vehicle motion was selected for the figures. The river is between 5 and 10 meters deep. The resulting swath width was between 10 and 20 meters per side.

The proposed method uses the sonar data according to the true neighborhoods in space according to the vehicle position at the time of record.

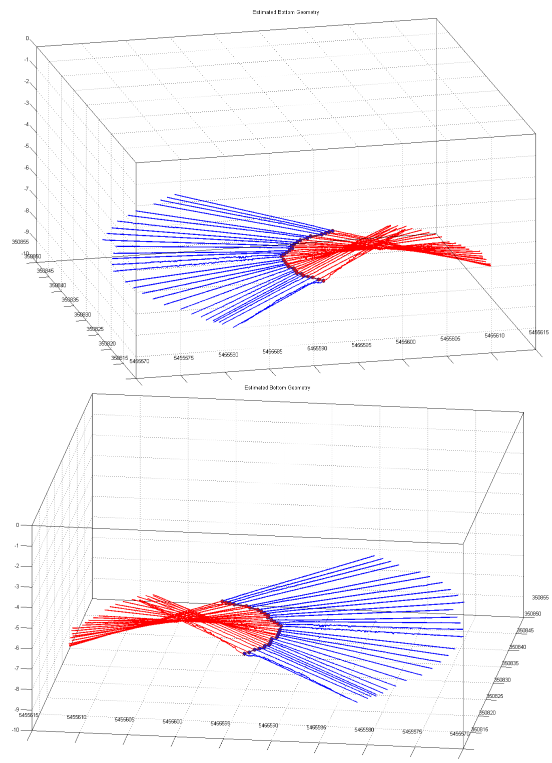

Fig. 8. Reconstructed riverbed surface from different views (axes are gives in meters). The overlapping of sonar lines on this curved trajectory can be seen. The vehicle trajectory is shown as projection to the ground.

V. Sliding window approach

In order to use the method as an online method, it is necessary to perform a surface shape estimation in a continuous way. These goals are best achieved through a windowed approach where submaps are created. Using a sliding window approach the estimation takes place for a certain fixed number of lines (see Figure 9). After new lines are received, older lines are dropped and the estimation is performed again to create a new submap. It is not necessary to repeat the estimation for every new line that is recorded but can be designed in such a way that there is always a certain percentage of overlap. It could also be designed adaptively, considering the derivation from a straight-ahead trajectory. Because only in cases of strong vehicle motion the line neighborhoods would change significantly enough so that a different estimation would be the result.

The first and last line in each estimation window are particularly susceptible to wrong normal estimation as the estimation of surface normals by means of nearest neighbors is inherently more unstable at the boundaries which can be seen in the reconstruction shown in Figure 8. Hence, it is advisable to discard the reconstruction of the first and last line for each window. The effect may also arise for other lines in the sliding window, depending on their position in space but for not too curvy AUV motion it should already be helpful to discard the first and the last line as they are the most likely ones to end up at a boundary position in space.

Additionally, the creation of submaps facilitates the incorpo-

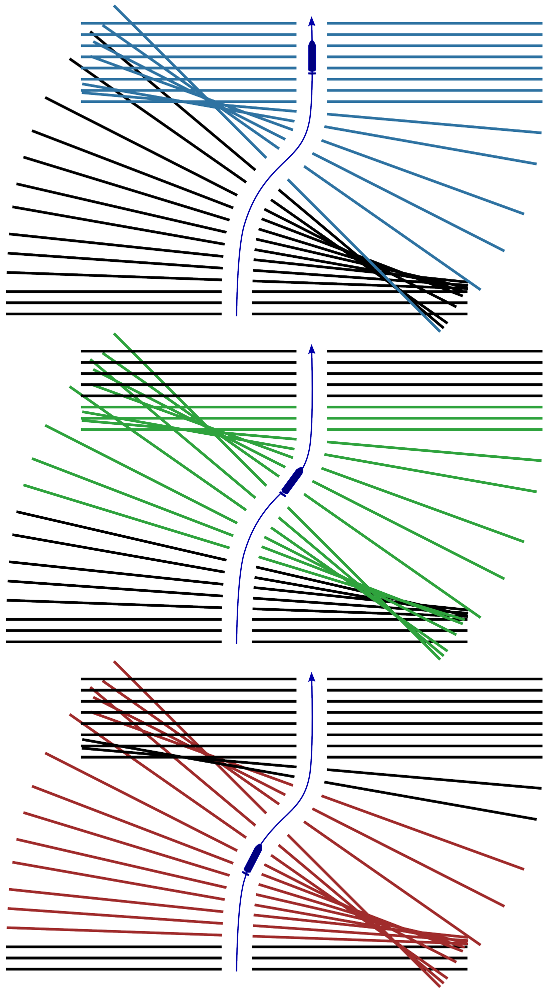

Fig. 9. Sliding window approach

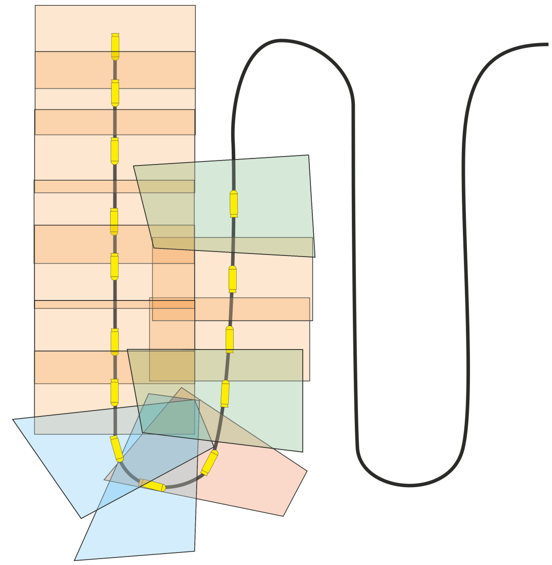

Fig. 10. Obtained submaps as input for SLAM

ration of existing SLAM solutions, e.g., Fairfield's SegSLAM (see [6]) and Bosse's methods (see [7], [8] use submap approaches to map the environment. All submaps as a whole form a map of the whole environment with consideration of the vehicle trajectory (see Figure 10).

It should be also be noted that the algorithm is basically a so-called anytime algorithm since estimation can be interrupted at any time and the resulting surface relief is still meaningful. Of course, the estimation becomes increasingly better the more iterations are performed.

VI. Outlook

A. Extensions

At the moment the reconstruction does not include the distortion of a scan line due to the vehicle (and hence the sonar sensor) motion between sending and receiving signals, shown in [5], but it is possible to incorporate this effect in the proposed framework.

The sliding window could be extended to using all the lines that are geometrically within a rectangle even if the lines were recorded a longer time before. This would surely increase the consistency of the estimation. However, before the estimation could start in this case the complex selection problem which lines contribute significantly to the area of the estimation window would have to be solved first. As the number of affected lines changes for each estimation, computational time guarantees cannot be given. Furthermore, the computation time would increase with the mission duration especially on re-visiting of places where many previously recorded lines would be reused.

B. Submap matching

There are several possibilites how to match the submaps, e.g., full matching using ICP (iterative closest point) variants, feature extraction and matching, or a combination of both. Given the computational constraints we opt for a purely feature-based matching. The features should be rotation-invariant as the submaps will most certainly be created from different viewpoints and to compensate for small geometric distortions originating from reconstruction inaccuracies. A survey of 3D features that may be used for matching is given in [9].

An interesting approach to use as SLAM framework could also be a hybrid SLAM solution as proposed in [10] where a FastSLAM-based (see [11]) front-end is used together with an Extended Kalman Filter based backend to mitigate the mutual weaknesses of each technique.

VII. Conclusion

The proposed extension of the seafloor shape estimation method by Coiras [1] lays the foundation to use side-scan sonar data for SLAM-based seafloor mapping with AUVs that perform arbitrary motion. The effects of motion and their influence on the inversion method have been described and examples of the resulting surface submaps have been given.

References

[1] E. Coiras, Y. Petillot, and D. M. Lane, “Multiresolution 3-D Reconstruction From Side-Scan Sonar Images,” *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 382–390, Feb. 2007, heriot-Watt University Edinburgh.

[2] E. Coiras and J. Groen, “3D target shape from SAS images based on a deformable mesh,” in *Proceedings of the 3rd International Conference on Underwater Acoustic Measurements (UAA), Nafplion, Greece*, 2009.

[3] P. Woock and C. Frey, “Deep-sea AUV navigation using side-scan sonar images and SLAM,” in *Proceedings of the IEEE Oceans Conference 2010*, may 2010.

[4] C. Mazel, *Side Scan Sonar Training Manual*. Salem, Klein Associates, 1985.

[5] J. Bell, “A model for the simulation of sidescan sonar,” Ph.D. dissertation, Heriot-Watt University, Department of Computing and Electrical Engineering, September 1995.

[6] N. Fairfield, “Localization, mapping, and planning in 3d environments,” Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, January 2009.

[7] M. Bosse and R. Zlot, “Map matching and data association for large-scale two-dimensional laser scan-based slam,” *I. J. Robotic Res.*, vol. 27, no. 6, pp. 667–691, 2008.

[8] ——, “Place recognition using regional point descriptors for 3D mapping,” in *International Conference on Field and Service Robotics*, 2009, cambridge.

[9] A. Runte, “Robuste Merkmale auf 3D-Daten,” Term Paper, Vision and Fusion Laboratory (IES), Karlsruhe Institute of Technology, July 2010.

[10] A. Brooks and T. Bailey, “HybridSLAM: Combining FastSLAM and EKF-SLAM for reliable mapping,” in *WAFR 2008: The Eighth International Workshop on the Algorithmic Foundations of Robotics*, 12 2008.

[11] M. Montemerlo and S. Thrun, *FastSLAM: A Scalable Method for the Simultaneous Localization and Mapping Problem in Robotics*, ser. (Springer Tracts in Advanced Robotics), B. Siciliano, O. Khatib, and F. Groen, Eds. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007.