Underwater SLAM: Challenges, state of the art, algorithms and a new biologically-inspired approach

Conference Paper · August 2014

DOI: 10.1109/BIOROB.2014.6913908

5 authors, including:

Felipe Guth
University College Dublin

Luan Silveira
Universidade Federal do Rio Grande (FURG)

Silvia Silva da Costa Botelho
Universidade Federal do Rio Grande (FURG)

Paulo Drews-Jr
Universidade Federal do Rio Grande (FURG)

Underwater SLAM: Challenges, state of the art, algorithms and a new biologically-inspired approach

Felipe Guth, Luan Silveira, Silvia Botelho, Paulo Drews Jr, Pedro Ballester

Abstract— **The unstructured scenario, the extraction of significant features, the imprecision of sensors along with the impossibility of using GPS signals are some of the challenges encountered in underwater environments. Given this adverse context, the Simultaneous Localization and Mapping techniques (SLAM) attempt to localize the robot in an efficient way in an unknown underwater environment while, at the same time, generate a representative model of the environment. In this paper, we focus on key topics related to SLAM applications in underwater environments. Moreover, a review of major studies in the literature and proposed solutions for addressing the problem are presented. Given the limitations of probabilistic approaches, a new alternative based on a bio-inspired model is highlighted.**

I. Introduction

Nowadays one of the main sources of mineral and biological resources are underwater environments. Many activities such as installation and monitoring of oil pipelines are performed by underwater robots. The limited range of light, presence of marine currents, turbidity and uniformity of the environment are some of the problems faced in robotics applications for localization and mapping. Although there are methods of triangulation for underwater environments, these usually require great effort, high cost and considerable logistics resources.

SLAM problems arise when the robot does not have previous knowledge about the map of the environment, nor does it know its own pose. Instead, all it is given are measurements $z_{1:t}$ and controls $u_{1:t}$. The term "Simultaneous Localization and Mapping" describes the resulting problem: In SLAM, the robot acquires a map of its environment while simultaneously localizing itself relative to this map [1].

At present, we have robust methods for mapping environments that are static, structured, and of limited size. These methods deal with issues related to computational complexity, data association and representation of the environment [2],[3]. Mapping unstructured, dynamic, or large-scale environments remain an open research problem [4].

In this context, we note the importance of SLAM in underwater applications with the objective to allow operation of robots in unknown environments. This paper presents the state of the art and main concepts related to underwater SLAM. Also, topics such as challenges, sensors and algorithms are revised. A report of major works of the literature is presented along with experimental results and a comprehensive analysis of a case study using a new bio-inspired approach for underwater SLAM.

The authors are with Centro de Ciências Computacionais , Universidade Federal do Rio Grande - FURG, Rio Grande - RS, Brazil. Email: felipeguth@furg.br

II. Challenges in Underwater SLAM

The implementation of SLAM in real underwater environments can be considered an unsolved problem in robotics. The underwater nature imposes a number of challenges not faced by terrestrial or indoor applications.

a) Sensors.: Taking into account the particularities of the underwater environment, the sensing of the environment is a key aspect to be mentioned. Generally, the data coming from the sensors have limited accuracy, particularly for environments with low-light, strong ocean currents and turbid waters. The estimated noise ends up causing significant impact on the tasks of localization and mapping, often causing the non-convergence of the system. The modeling of sensor noise is an important and difficult task, especially for large-scale environments since the noise may be different in certain situations, requiring a new calibration to obtain a better estimate of the system. The sensors usually have a limited depth for operation, making applications costly for high depths. The Table I shows a compilation of the main sensors used in underwater robots as well as the type of information provided by each one.

b) Feature extraction.: One of the most important factors for SLAM is the landmarks. The recognition of landmarks is essential to keep the estimated location of the robot, reducing the uncertainty of the system and recognizing places before traveled. Due the unstructured characteristics of the environment, in most cases there are no distinct objects or features. The turbidity disturbs the operation of sensors, particularly those based on light such as cameras and lasers. The solar lighting reaches only a few meters of the water layer. In the very clear water of the open ocean, less than 0.5% of the surface light reaches a depth of $100m$. The field of view of the robot is very small, thus the use of artificial lighting is required at higher depths. As a result of having greater range, even in adverse conditions, sonars are the main sensors used for feature extraction of underwater resources.

Multiple sources of perturbations in sonar measurements can be listed such as hydrostatic effects of currents or waves, seismic activity, inhomogeneous distribution of pressure, vessel traffic, marine animals and the propellers of the own vehicle. Besides the noise of sonar readings, it is often necessary to deal with the presence of ghosts, reflections and poor resolution of acoustic imaging for extracting features. The dynamics of underwater natural resources is another factor that makes recognition of previously visited locations a hard task.

c) Absolute Location.: The underwater environment does not allow the use of GPS. To overcome this limitation, alternative solutions such as the use of triangulation systems LBL , USBL and SBL are proposed. These systems require great effort for installation, beyond a special demand of logistics, having high cost. Besides, these solutions limit the area of operation of the robot.

d) Computational Complexity.: The computational complexity of SLAM applications are closely related to the size of the environment to be explored and the methods used for feature extraction, tracking, data association and filtering. Once the robot moves in the environment, most features are recognized and monitored. The map elements increase as well as the uncertainty calculations of the position of the robot, the landmarks of the environment and their correlations. Large-scale environments, today are still considered as a limiting factor for SLAM applications.

III. Probabilistic Approach

There are various techniques to treat the issues associated with SLAM, such as recognizing features previously extracted (i.e., data association or loop closure detection), and re-skewing recent parts of the map to make sure that different instances of the same feature, detected in different times, is represented in only one instance in the system. Statistical techniques used in SLAM include Kalman filters, particle filters and scan matching of range data. Bio-inspired approaches are also beginning to be explored.

In a probabilistic manner, the problem of Simultaneous Localization and Mapping requires that the probability distribution

$$P(x_k, m|Z_{0:k}, U_{0:k}, x_0) \tag{1}$$

is computed for all times k. This probability distribution describes the connection of the posterior density of the locations of the reference points and the state of the vehicle (in time k) given the observations $z_{0:k}$, the input controls $u_{0:k}$ and the initial state of the vehicle x_0. In general, a recursive solution of the problem is desirable. Starting with an estimate for distribution $P(x_{k-1}, m|Z_{0:k-1}, U_{0:k-1})$ at time $k-1$, the posterior probability following a control u_k and an observation z_k is computed using Bayes theorem. This computation requires that a state transition model and an observation model are defined, which describe the effect of the input control and observation respectively.

The main algorithms based on probabilistic Bayesian approach are Extended Kalman Filters and Particle Filters approaches, GraphSLAM, Extended Information Filters. Detailed information can be found on [1], [4], [5] and [6].

IV. Related Works

For the development of this section, a research of 70 SLAM underwater works was conducted in the bases IEEE Explorer, ACM and Google Scholar. Due the pages limit restriction, we present a review of the state of the art established with 20 influential recent works.

Table I summarizes the SLAM works in underwater environments. It can be noted a predominance of the use of the EKF algorithm for performing SLAM. This method was the first successful proposal for conducting SLAM, and even today, is still considered one of the best solutions to solve the problem, due its potential to represent the state of uncertainty and convergence. The quadratic complexity of updating is one of its problems. For the contour of this issue, newer applications make use of submaps to limit the size of the elements of the state and covariance.

Considering the nature of the underwater environment, the most widely used sensor in robotics applications is sonar since sound waves have far-reaching of the aquatic environment, unlike lasers and cameras. Thus, we note that most of the works rely on features extracted from sonar data to build the map. Often, such data comes from objects that are man-made since these typically provide better features than natural structures.

Another method for extracting features for underwater SLAM is the use of bathymetric data acquired with echo sounder sonars from the seabed. In addition to bathymetric data and acoustic images derived from echo sounder and imaging sonars, some applications use data from cameras to extract features, especially in environments with good lighting and presence of natural structures like corals. Another highlight is the use of SLAM applications combined with tasks for mosaic building of the seafloor.

The last three fields of Table I represent a subjective analysis of questions of Performance (Pfc), Accuracy (Acc) and Simplicity of implementation (Smp). On the issue of performance, a work qualified with '++' allows real-time processing. When it comes to the accuracy, a work review '++' is one that generated similar results compared to the ground truth. Simplicity refers to how easy are to implement the methods and techniques proposed by a paper. This qualifier takes into account factors such as the number of parameters, clarity and explanation of the algorithms for further replication.

Despite the inherent advances, there are still challenges to be solved by underwater SLAM applications. Extracting good features of the environment is a highly complex problem, especially when it comes to underwater environments. Most of them are unstructured or have few particular aspects. The dynamicity of the underwater environment is another factor that makes recognition of places and features detected in a previous time a hard task. Online applications are fundamental for robots AUVs in underwater missions. Given the high level of demand for computational resources required for SLAM performance, the development of these solutions remain a difficult task, especially for large-scale environments.

TABLE I

MAIN APPLICATIONS OF UNDERWATER SLAM

Year	Author	Scenario	Main Sensors	Algorithm	Landmark	Descriptor	Ground Truth	Pfc	Acc	Smp
2000	[7]	Swiming Pool, Sea	Imaging Sonar	EKF	Artificial Targets	Point Features	Yes, No	-	-	++
2001	[8]	Tank, Sea	Imaging Sonar, INS, DVL	EKF	Seabed	Point Features	Yes	-	+	-
2001	[9]	Tank, Pier	Imaging Sonar	EKF	Cylinders, legs of a pier	Point Features	No	-	-	+
2003	[10]	Sea	LBL, Imaging Sonar, DVL, Compass	LS Optimization	Transponders locations	Point Features	Yes	-	+	-
2004	[11]	Sea	Side-scan sonar	EKF	Objects on seabed	Point Features	No	-	-	-
2004	[12]	Great barrier reef	Gyroscope, Pressure sensor, Camera, Imaging Sonar	EKF	Reef	Point Features	No	-	-	+
2004	[13]	Stellwagen Bank National Marine Sanctuary	Camera, Tilt Sensor, Compass, AHRS, DVL, Pressure sensor	EKF	Rocks on seafloor	SIFT and Harris Points	No	+	-	+
2005	[14]	Subsea hydrothermal mount	FOG, Tilt sensors, Pressure sensor, DVL, Echo Sounder Sonar	EKF	Bathymetric patches	-	Yes	+	+	+
2006	[15]	Barge	Cameras, IMU	SLAM through Entropy Minimization	Hull of barge, Iron beams of barge structure	3D Point Clouds	No	-	+	-
2006	[16]	Italian coast sea	DVL, INS, Compass, LBL	EKF	Beacons locations	Point Features	Yes	-	+	-
2006	[17]	Tagiri Vent Area (sea)	RDI Navigator, Pressure sensor, AHRS, FOG, Profile Sonar, Camera	PF	Acoustic Reflector, Buble Plume	Point Features	No	++	-	-
2007	[18]	Tank, Cenote	DVL, INS, Depth sensor, 56 narrow beam sonar transducers	PF	Walls	Evidence grid	Yes, No	++	-	- -
2008	[19]	Ningaloo Marine Park	DVL, Compass, Tilt sensor, Pressure sensor, Stereovision rig.	EIF	Sponge beds	SURF points	No	+	-	+
2008	[20]	Marina	DVL, MRU, Imaging Sonar	EKF	Walls	Line Features	Yes	++	++	-
2008	[21]	Barge	IMU, Depth sensor, Compass, DVL, Imaging Sonar	ESEIF	Box targets, cylindrical targets, small brick-shapped objects.	Point Features	Yes	-	+	-
2010	[22]	Marina	DVL, MRU, Imaging Sonar	EKF	Walls	Point Features	Yes	++	++	+-
2011	[23]	Sea	Camera	PF	Tumbling target	SIFT points	No	++	-	-
2011	[24]	Sea	Sonar, Depth Sensor, DVL, Compass	PF	Featureless	-	Yes	++	++ -	
2012	[25]	Sea	Camera	EKF	Seabed	SURF points	No	++	+	+-
2013	[26]	USS Aircraft Carrier	IMU, DVL, Pressure sensor, Profiling Sonar	Pose Graph SLAM Using Submap Alignment	Hull, propeller	Point Cloud	No	++	-	-

The Extended Kalman Filter can represent only a single estimate belief, and if it is wrong, the robot can get lost in the environment. The Particle Filter can represent various estimates for iteration, since each particle represents an estimate of the state. In terms of computational complexity the EKF have complexity $O(M^2)$ where M is the size of the state vector, the Particle Filter algorithm on FastSLAM has complexity $O(MK)$, where M is the number of landmarks and K the number of particles.

Given the drawbacks of probabilistic algorithms, regarding computational complexity for large-scale maps, we present in the next section a new bio-inspired technique to solve SLAM problems, focused on underwater environments. The bio-inspired approach has the ability to represent multiple estimates simultaneously on the CANN (continuous attractor neural network) dynamics, which compete each other in the network to represent different beliefs. Although being able to maintain lifelong maps, the algorithm has complexity $O(N^2+M)$ where N is the number of neurons of the CANN network and M the number of landmarks in the map.

V. BIOLOGICALLY-INSPIRED APPROACH

Recently, researches are adopting discoveries of animals neuronal behavior to develop new algorithms and methods applied in computational solutions in fields such as robotics. In the SLAM context the pioneering work of [27], [28] proposed an algorithm, dubbed RatSLAM, that models the navigation mechanisms of rats brains. This method focus on modelling of Grid Cells found in mammalian brain, that are related to navigation purposes. A characteristic of this methods is to build topological maps of the environment, as it is enforced for neuronal experiments with rats [29] and bats [30]. This algorithm has been adapted to underwater environments [31], with promising results.

The system is divided into three modules, Local View Cells, Pose Cells and Experience Map, as shown in Fig. 1.

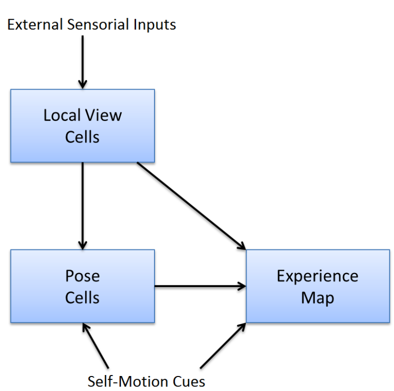

Fig. 1. System architecture.

A. Pose Cell

The Pose Cell is the core of the algorithm. It is composed of a CANN (Continuous Attractor Neural Network) designed to simulate the behaviour of brain mechanisms of navigation. In our approach, we extended the RatSLAM Pose Cell, adding another layer in the Neural Network, to make the estimation of position in four Degrees of Freedom. From now on, it's composed of a 4D CANN to estimate the pose xyz and yaw orientation.

B. Local View Cells

The accumulated error in the path-integration process is reset by learning associations between pose cell activity and local view cell activity, while simultaneously recalling prior associations. The local view is represented as a vector V , with each element of the vector representing the activity of a local view cell. During a loop-closure event, the familiar visual scene activates local view cells with learnt connections to the pose cells representing the pose where the visual scene was first encountered. Because of the attractor dynamics of the pose cells, a single visual scene is not enough to force an immediate change of pose; several consecutive and consistent views are required to update the pose.

The Data Association was solved using the Bag of Features(BoF) [32] algorithm and each phase of the process will be explained below.

1) Extracting and Recognizing Places: Concerning the tasks of learning and recognizing places visited by the robot in the environment, it was developed an algorithm using Bag of Visual Features[32]. This algorithm plays the role of dictionary of visual features, extracted from the camera images of the environment.

The first step is the dictionary creation, which corresponds to a set of every characteristics points in the image. The keypoints extraction is performed by the Speeded Up Robust Features Algorithm (SURF) [33], a robust local feature detector.

A clustering algorithm is required to build a discrete vocabulary from millions(or billions) of local features sampled from the training data. [34]. This work uses K-Means algorithm to group similar features [35]. Each cluster region forms a word in the dictionary. This dictionary is the final result of the training phase, when an "a-priori" database of environment features. The Fig. 2 depicts the general process to represent and recognize places of the environment through the use of visual features.

In the recognizing phase, firstly the SURF algorithm is applied to extract the keypoints. After that, we create a histogram of features, representing the number of features of each group in the image. Then, the histogram is normalized and compared to every template (histogram) already stored. In the case that one template match with another already created, is injected energy in the neuron associated with this Local View Template of the CANN of Pose Cells, leading to loop closures if the energy is enough to shift the activity packet of the network. Otherwise, when a histogram does not match any created template, a new template is created and assigned to a particular neuron of the CANN of Pose Cells.

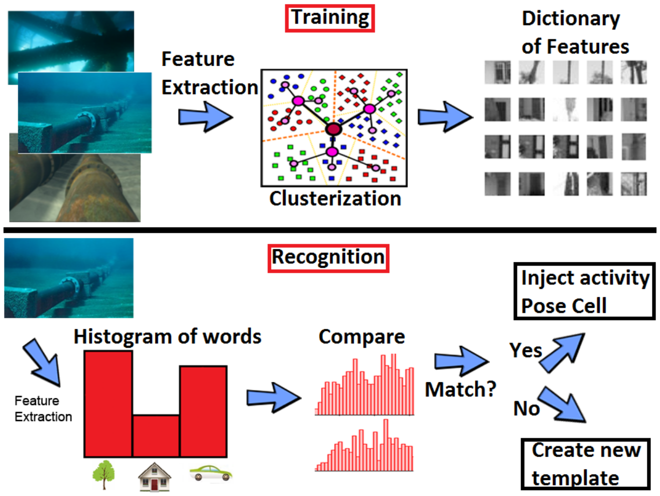

Fig. 2. In the training phase the keypoints extracted from the images with the SURF algorithm are clustered in groups using the K-means algorithm. These groups form the dictionary of the bag of words. In the recognition phase, keypoints are extracted from the image, a histogram of words is generated according to the dictionary. If the histogram, that represents a local view, match with another one stored in the local view module, it means that the robot is in a place seen before and this place in the Pose Cell Network receive energy, leading to loop closure. If there is no match, a new template is created in the Pose Cell Network associated with the current local view template.

C. Experience Map

The Experience Map module has the biggest change, because now it is necessary to create a map in three dimensions. For this, the processes of create, update experiences and close loops were modified.

An experience remains a tuple, as shown in Eq. (2).

$$e_i = \{P_i, V_i, pos_i\} \tag{2}$$

where the change in the representation of the position pos_i, represented by a 3D pose, as shown in Eq. (3). This pose has an origin and orientation, the last one represented by a quaternion.

$$pos_i = [Origin; Orientation] \tag{3}$$

where

$$Origin = [x, y, z] \tag{4}$$

$$Orientation = [x, y, z, w] \tag{5}$$

In the same way, the link between two experience has a n, making a transformation between two poses, as follows:

$$l_{ij} = \{T_{ij}\}, \tag{6}$$

this transformation T_{ij} has the same components of the pose shown in the (3).

The position of a new experience will be calculated from the position of the current experience multiplied by transformation experiments conducted between:

$$pos_j = pos_i * T_{ij} \tag{7}$$

D. Case Study

In this case study, the system receives data from DVL and compass for motion estimation and a visual camera to perceive the environment. The image features are detected using Bag of Features [36] of SURF descriptors. A continuous attractor neural network is responsible to filter the motion and perception information, estimating the robot's position on the environment. The Fig. 3 shows the map generated in a simulated underwater environment (blue), compared with ground truth (green) and dead reckoning (red). The method adjust the robot location error significantly when compared to the dead reckoning trajectory as shown in Fig. 4. The residual error is about $2m$ and comes from the motion error of DVL sensor in the vertical axis. As the robot did not travel during a long period along this axis, the system was not able to detect a loop. Further information about the algorithm is available in [31].

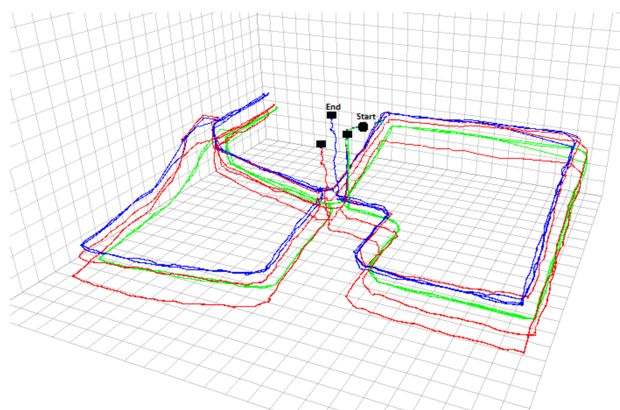

Fig. 3. Result of the Bioinspired Algorithm: Blue path shows the map created by the SLAM method. Green line represents the Ground truth positions. The dead reckoning path is shown in red.

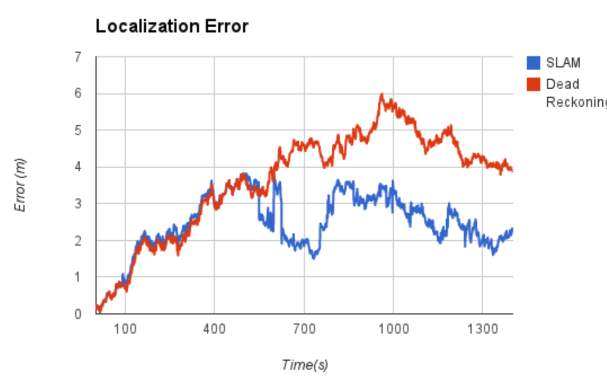

Fig. 4. Localization error comparison: The errors are calculated using the Euclidean distance with respect to Ground Truth. The blue line decay (t = 550 seconds) represents the moment when the first loop closure was detected, decreasing the robot position estimation error.

This approach is well suitable for life-long maps, where the environment has some degree of ambiguity and a considerable number of landmarks. The algorithm complexity is linear in the number of nodes in the topological map.

VI. CONCLUSION

This paper presented the main challenges faced by SLAM applicationsand related the key techniques and main sensors to deal with SLAM problems. A review of 20 major studies in the literature on the topic of underwater SLAM was taken along a table with detailed information. Finally, a new point of view on the SLAM problem using bio-inspired techniques was presented. This approach is based on the localization performed by neural structures found in mammals such as rats and bats, and it is expected to be similar to those used by dolphins. The inherent potential of this approach, for large-scale environments, is an important fact for motivation of further research, since this issue is considered a major factor limiting the application of currently probabilistic techniques.

ACKNOWLEDGMENT

The authors gratefully acknowledge the contribution of CAPES, CNPQ and Oil Brazilian Agency (ANP).

REFERENCES

[1] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.

[2] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part i," *IEEE Robotics and Automation Magazine*, vol. 13, no. 2, pp. 99–110, 2006.

[3] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): Part II The State of The Art," *IEEE Robotics and Automation Magazine*, vol. 13, no. 3, pp. 108–117, 2006.

[4] S. Thrun, "Simultaneous localization and mapping," *Robotics and Cognitive Approaches to Spatial Mapping*, pp. 13–41, 2008.

[5] S. Thrun and M. Montemerlo, "The graphslam algorithm with applications to large-scale mapping of urban structures," *IJRR*, vol. 25, no. 5, pp. 403–430, 2006.

[6] S. Thrun, "Simultaneous mapping and localization with sparse extended information filters: Theory and initial results," 2002.

[7] S. Williams, P. Newman, G. Dissanayake, and H. Durrant-Whyte, "Autonomous underwater simultaneous localisation and map building," in *ICRA*, vol. 2, 2000, pp. 1793–1798.

[8] J. L. Leonard, R. N. Carpenter, and H. J. S. Feder, "Stochastic mapping using forward look sonar," *Robotica*, vol. 19, no. 5, pp. 467–480, 2001.

[9] T. I., Y. R. Petillot, D. M. Lane, and C. Salson, "Feature extraction and data association for auv concurrent mapping and localisation." in *ICRA*. IEEE, 2001, pp. 2785–2790.

[10] P. Newman and J. Leonard, "Pure range-only sub-sea slam," in *ICRA*, 2003, pp. 1921–1926.

[11] S. Tena I., Y. Petillot, and D. Lane, "Concurrent mapping and localization using sidescan sonar," *IEEE Journal of Oceanic Engineering*, vol. 29, no. 2, pp. 442–456, 2004.

[12] S. Williams and I. Mahon, "Simultaneous localisation and mapping on the great barrier reef," in *ICRA*, vol. 2, 2004, pp. 1771–1776.

[13] R. Eustice, O. Pizarro, and H. Singh, "Visually augmented navigation in an unstructured environment using a delayed state history," in *ICRA*, vol. 1, 2004, pp. 25–32 Vol.1.

[14] C. N. Roman and H. Singh, "Improved vehicle based multibeam bathymetry using sub-maps and slam." in *IROS*, 2005, pp. 3662–3669.

[15] J. M. Sáez, A. Hogue, F. E. Ruiz, and M. R. M. Jenkin, "Underwater 3d slam through entropy minimization." in *ICRA*, 2006, pp. 3562–3567.

[16] E. Olson, J. Leonard, and S. Teller, "Robust range-only beacon localization," *IEEE Journal of Oceanic Engineering*, vol. 31, no. 4, pp. 949–958, 2006.

[17] T. Maki, H. Kondo, T. Ura, and T. Sakamaki, "Photo mosaicing of tagiri shallow vent area by the auv "tri-dog 1" using a slam based navigation scheme," in *OCEANS*, 2006, pp. 1–6.

[18] N. Fairfield, D. Jonak, G. Kantor, and D. Wettergreen, "Field results of the control, navigation, and mapping systems of a hovering auv," in *UUST*, 2007, pp. 20–22.

[19] I. Mahon, S. Williams, O. Pizarro, and M. Johnson-Roberson, "Efficient view-based slam using visual loop closures," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1002–1014, 2008.

[20] D. Ribas, P. Ridao, J. Tardos, and J. Neira, "Underwater slam in man-made structured environments," *Journal of Field Robotics*, vol. 25, no. 11-12, pp. 898–921, 2008.

[21] M. Walter, F. Hover, and J. Leonard, "Slam for ship hull inspection using exactly sparse extended information filters," in *ICRA*, 2008, pp. 1463–1470.

[22] A. Mallios, P. Ridao, D. Ribas, F. Maurelli, and Y. R. Petillot, "Ekf-slam for auv navigation under probabilistic sonar scan-matching." in *IROS*, 2010, pp. 4404–4411.

[23] S. Augenstein and S. Rock, "Improved frame-to-frame pose tracking during vision-only slam/sfm with a tumbling target," in *ICRA*, 2011, pp. 3131–3138.

[24] S. Barkby, S. B. Williams, O. Pizarro, and M. V. Jakuba, "A featureless approach to efficient bathymetric slam using distributed particle mapping." *Journal of Fields Robotics*, vol. 28, no. 1, pp. 19–39.

[25] F. Ferreira, G. Veruggio, M. Caccia, and G. Bruzzone, "Real-time optical slam-based mosaicking for unmanned underwater vehicles," *Intelligent Service Robotics*, vol. 5, no. 1, pp. 55–71, 2012.

[26] M. VanMiddlesworth, M. Kaess, F. Hover, and J. Leonard, "Mapping 3D underwater environments with smoothed submaps," in *Conference on Field and Service Robotics*, 2013.

[27] M. Milford and G. Wyeth, "Mapping a suburb with a single camera using a biologically inspired slam system," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1038–1053, 2008.

[28] M. Milford, D. McKinnon, M. Warren, G. Wyeth, and B. Upcroft, "Feature-based visual odometry and featureless place recognition for slam in 2.5d environments," in *The Australasian Conference on Robotics and Automation 2011*. Australian Robotics & Automation Association, 2011, pp. 1–8.

[29] E. I. Moser, E. Kropff, and M.-B. Moser, "Place cells, grid cells, and the brain's spatial representation system," *Annual Review of Neuroscience*, vol. 31, no. 1, pp. 69–89, 2008.

[30] M. M. Yartsev and N. Ulanovsky, "Representation of three-dimensional space in the hippocampus of flying bats," *Science*, vol. 340, no. 6130, pp. 367–372, 2013.

[31] L. Silveira, F. Guth, D. Fischer, P. Drews-Jr., and S. Botelho, "3D robotic mapping: A biologic approach," in *IEEE ICAR*, 2013.

[32] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *In Workshop on Statistical Learning in Computer Vision, ECCV*, 2004, pp. 1–22.

[33] H. Bay, T. Tuytelaars, and L. V. Gool, "Surf: Speeded up robust features," in *In ECCV*, 2006, pp. 404–417.

[34] S. O'Hara and B. A. Draper, "Introduction to the bag of features paradigm for image classification and retrieval," *arXiv preprint arXiv:1101.3354*, 2011.

[35] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035.

[36] S. O'Hara and B. A. Draper, "Introduction to the bag of features paradigm for image classification and retrieval," *CoRR*, vol. abs/1101.3354, 2011.