24th COBEM - 2017

24th ABCM International Congress of Mechanical Engineering
December 3-8, 2017, Curitiba, PR, Brazil

COBEM-2017-0474
OFFLINE VIEWPOINT PLANNING FOR UNDERWATER INSPECTION USING OCTOMAP

Bruno Locks Floriani
Lucas Weihmann
Henrique Simas
Federal Univesrity of Santa Catarina
florianibl@gmail.com
lucas.weihmann@ufsc.com
henrique.simas@ufsc.com

Narcís Palomeras
Pere Ridão
Universitat de Girona
narcispr@gmail.com
pere.ridao.rodriguez@gmail.com

Abstract. *Viewpoint path planning is an essential research area to develop underwater 3D maps for exploring underwater environments. This paper presents a new algorithm to create offline paths for inspecting underwater 3D complex structures and high relief environments. An* a priori *environment representation as an a octree is assumed. The algorithm optimizes the planning in two different steps: first the number of viewpoints and later the path between them.*

Keywords: *AUVs, Inspection, Complex Structures, Offline, Octree*

1. INTRODUCTION

Almost $3/4$ of the Earth's surface is covered by water, which is essential for the existence and maintenance of the life in our planet. A long the years many city's were build near rivers, lakes, oceans. Nowadays the oceans are source of food, renewable energy, over it, the oceans are the main trade route and under the oceans are huge oil and gas reserves. Those means that have many industrial facilities such oil fields, dams, bridges, offshore wind turbines build in the oceans and a long the years those structures become older and need to be inspected, even to plan the maintenance (de Souza, 2006; Siciliano and Khatib, 2008; Zanoni and de Barros, 2015; Jacobi, 2015).

Underwater inspection used to be performed by specialized divers and/or Remotely Operated Vehicles (ROV). ROVs are a class of the Underwater Unmanned Vehicles (UUVs) characterized by a physical connection with the base, through this connection energy is sent to the vehicle and signal is exchanged. Based on this characteristics the ROV requires a pilot and a crew onshore or in a mother-ship operating during the entire mission.

However divers need special training and is often a dangerous activity. In the some direction, ROV requires a team to operate and has limiting movements because the physical connection.

Considering this limitations the development of Autonomous Underwater Vehicle (AUV) has received more effort. The AUVs are another class of the UUVs which don't have connection with the base, to it the vehicle have to have it's own source of energy and be capable to plan tasks and take decisions face to unexpected events. For this they can perform those tasks and highly reduce the costs, allowing regular inspections (Siciliano and Khatib, 2008; Zanoni and de Barros, 2015).

Nevertheless AUVs are been used for 2D survey, usually just performing simple 2D trajectories which makes impossible to safely inspect areas with a high 3D relief or complex structures.

Moreover to reconstruct or inspect complex objects, environments with high 3D relief, one single view, picture could be not enough to produce robust results requiring more views. The problem of finding the number of required views and which are those views is called view planning problem (Roberts and Marshall, 1998).

There are two main approaches to solve the view planning problem: (a) next best view (NBV) also called non-model based method, where the algorithm/system do not have any *a priori* knowledge of the environment then the vehicle have to determine the next best view based on the analyses of the current and previews views, and (b) model-based methods in which based on *a priori* model with some fidelity with the real environment select the best viewpoints to inspect a structure/environment Scott *et al.* (2003). The model-based methods permit optimizing the number of views, the path length and even guarantee the full coverage.

In this paper a model-based view planning approach is presented. Girona 500 AUV Ribas *et al.* (2012) equipped with a multibeam sonar mounted over a pan and tilt unit has been used in the tests. Our objective is to plan a sequence of viewpoints to inspect a 3D structure in the underwater environment giving a simple *a priori* model, taking into account the sensor limitations and an operational safe distance.

This paper presents the viewpoints, the path as well as evaluation of the algorithm.

An short overview about related works is presented in Section 2. In Section 3.is presented the proposed approach description, followed by the evaluation of the proposed method in Section 4.1 Simulations are presented in Section 5and, finally the conclusions.

2. Related Work

A comprehensive literature review with the most relevant works in model-based viewpoint planning can be found in (Scott *et al.*, 2003; Tarabanis *et al.*, 1995; Roy *et al.*, 2004; Almadhoun *et al.*, 2016).

Majority of the algorithms in model-based view planning works in a two step "generate-test"approach, which became popular in the late 1990s. More recently a two-step optimization scheme have been employed. First step is solve the Art Gallery Problem (AGP), which means found the minimum set of viewpoints to cover, inspect the object, region, structure. Then, in the second step compute the shortest path to connect the viewpoints, in other words, solve the traveling salesman problem (TSP) (Bircher *et al.*, 2015; Golden *et al.*, 1980).

In 1995, Tarbox and Gottschlich (1995) presented an approach, called IVIS (Integrated Volumetric Inspection System), based on the "view sphere" to generate the viewpoints focused in feature tolerance inspection. The "view sphere" approach reduce the viewpoint space (the space where a viewpoint can be pose) to the surface of a sphere centered in the object enclosing the target object. This assumption simplifies the problem from 6D to 2D. IVIS uses a volumetric representation, many others systems uses a boundary-based representation. For the authors the use of volumetric representation make easily to incorporate the tolerance and sensing operations. This work introduced many important concepts as the viewability and measurability matrix. A surface point is assumed measurability if it is seen by the camera and the light source, then the measurability matrix captures the visibility analysis over all surface points and all admissible viewpoints (Tarbox and Gottschlich, 1995; Scott *et al.*, 2003).

Although for large areas or objects it is not possible to apply the "view sphere" approach, mainly because the radius to enclose whole the region to be inspected would be larger than the sensor range making it impossible to inspect. Nevertheless a variation of this technique using coarsely sampled space carving technique can be applied. Supported in this idea Schmid *et al.* (2012) introduced the DSM (Digital Surface Model), first the DSM is smoothed in two steps, then equally spaced normals of the smoothed DSM (2.5D) are calculated and with the multi-stereo range the viewpoint is defined. Using the camera angle range and an heuristic criteria some viewpoints are selected. Later the TSP is solved using a Farthest-Insertion-Heuristic Rosenkrantz *et al.* (1974) algorithm.

Ellefsen *et al.* (2016) proposed one approach that select viewpoints at the same time as planning the path for this and a genetic algorithm is applied. An important contribution of this work is consider that 100% coverage is not always possible or desirable.

A view planning for shape reconstruction using a triangular mesh representation of the environment is proposed by Bircher *et al.* (2015).

Ososinski and Labrosse (2014) presented an approach to estimate the visibility of a complex 3D environment from viewpoints posed in a 2D plan. They use octomap to represent of the environment and set viewpoints in the nodes of grid pose in a plane. For each viewpoint the visibility of each cell is estimate and then the visibility of all the cells seen from the viewpoint is summed and divided for the number of faces of the full environment. To estimate the local visibility of a cell is necessary to evaluate the six faces that compose that cell into a very time consuming process.

A based-model view planning was proposed for Floriani *et al.* (2017), this approach represent the environment using octomap and makes an intensive use of the octree structures and raycasting techniques allowing a fast solution that coverage more than 95.00% using a few thousands of points.

3. Proposed Approach

We are proposing a model-based view planning algorithm to inspect 3D complex structures while trying to optimize the number of viewpoints that guarantee the largest visibility and finding the lowest-cost path that connects all the viewpoints avoiding to collide with the *a priori* known model. As a result, we obtain a set of viewpoints, the path to be executed, and an estimation of the coverage.

Our method can be organized in four main steps: (a) environment representation, (b) viewpoint selection, (c) viewpoint sequence, and (d) final path computation.

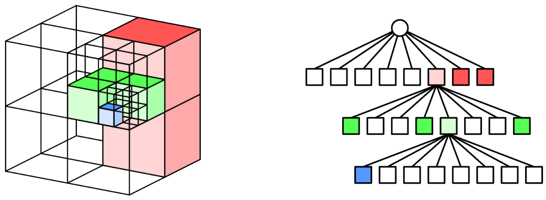

Figure 1. An octree model, where the blue cell is occupied. In the left, the volume representation and in the right the corresponding tree. Figure extracted from Hormung Hornung *et al.* (2013).

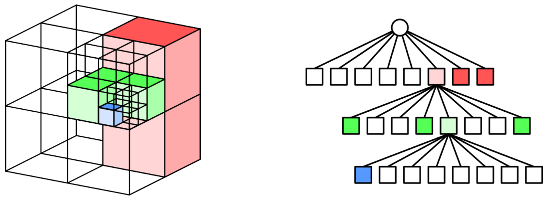

Figure 2. Tree Structure environment represented using an Octomap.

3.1 Environment representation

Our environment is represented using an octree. An octree is a set of nodes, each one representing a cubic volume (voxel or cell) which is recursively subdivided until reach a defined size or a number of pre-defined subdivisions, as shown in Fig. 1. The Octomap implementation Hornung *et al.* (2013) of the octree was choose because has a compact size, a fast computation and is well documented. Figure 2 shows a test environment that represents a subsea tree structure using an octomap (or octree).

3.2 Viewpoint selection

In the proposed solution, first we have to generate samples (candidate viewpoints) and then select the best ones. Initially a working area must be set, keeping inside all the region to be inspected, this task is performed by the user. In this working area the candidate viewpoints are randomly generated. The number of candidates viewpoints is set by the user.

Analyzing the probabilistic completeness of the coverage sampling problem based on randomly viewpoints generation is possible to guarantee the complete coverage of the underwater structure under a minimum number of candidate viewpoints (Englot, 2012).

During the candidate viewpoint generation phase, the points are evaluate and only those that accomplish this criteria are considered as valid candidates: i) the viewpoint must be inside a free cell, ii) occupied cells must be visible from the candidate viewpoint under sensors range criteria, and iii) the viewpoint is at a safest distance from any occupied cell. For each candidate viewpoint not only its 3D position is stored but also a submap, which is an octomap that contains all the cells visible from that position.

A viewpoint selection pseudo code is presented in Algorithm 1, where X is the number of random candidate viewpoints (Floriani *et al.*, 2017).

Algorithm 1 - Viewpoint

Input: Octomap
Output: Selected Viewpoints

Initialization :
N - Vector of Viewpoints
for $N = 0$ to X **do**
    vp = random
    if $(vp_occupancy < 0.5)$ **then**
        Raycasting from vp
        $ds = sqrt(vp^2 - raycast^2)$
        if $ds > min_{range}$ && $ds < max_{range}$ **then**
            vp is acceptable viewpoint
            $N \leftarrow N + vp$
return N

Once all the candidates have been generated the best viewpoints, which means the viewpoints that add more information about the model, are selected. We present and evaluate three different approaches to select the views.

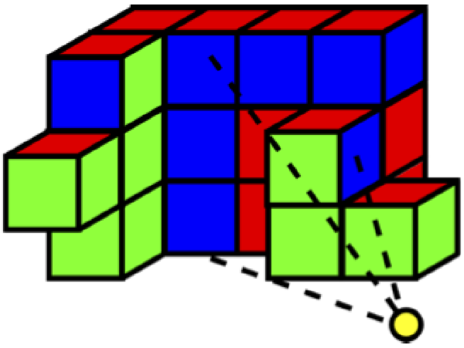

Figure 3. Face visibility from the viewpoint (yellow). Green are fully visible, Blue are partially visible and red are not visible. Figure extracted from Ososinski and Labrosse (2014)

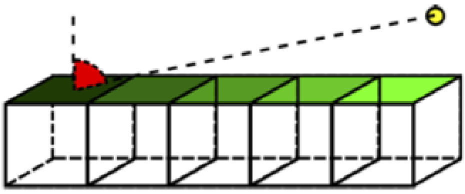

Figure 4. Incidence angle and distance affect the visibility Ososinski and Labrosse (2014)

3.2.1 Selection based on the number of visible nodes

The viewpoint with more visible nodes is selected, then the cells visible from this viewpoint are removed from the submap of all other viewpoints and the process is repeated until there are no more candidates or the number of cells in the submap for any remaining candidate viewpoint is below a threshold defined by the user.

3.2.2 Selection based on the local visibility

Another selection criteria is the global visibility proposed by Ososinski and Labrosse (2014) where he considers how well a cell and each of it's face is seen by the viewpoint. To obtain the visibility of a cell from the viewpoint first is verified how well a face can be seen (fully visible, partially visible or not visible), as shown in Fig. 3. It is measured how many faces of a cell are visible, the distance and the angle between the viewpoint and each face of the cell, Fig. 4 shows the influence of the incidence angle and distance in the visibility quality, where the lighter green shows a better visibility and as the green became more dark the quality of the visibility became lower.

$$D(V_p, f) = \frac{d(V_p, f) - R_{min}}{R_{max}} \tag{1}$$

, where D is the normalized distance between the viewpoint and the cell face, R_{min} is the minimum sensor range and R_{max} is the maximum sensor range.

$$A(V_p, f) = \frac{\theta(V_p, f) - \frac{\pi}{2}}{\frac{\pi}{2}} \tag{2}$$

, where A is the normalized angle of incidence between the viewpoint and the cell face.

$$V(V_p, f) = 0.5 + 0.3xD(V_p, f) + 0.2xA(V_p, f) \tag{3}$$

, where $V(V_p, f)$ is the visibility of a face from the viewpoint (V_p). If the face is fully visible we use 0.5, if the face is partially visible (< 4 corners visible) we use 0.3 on the Eq. 3.

The global visibility (Eq. 4) for each viewpoint (V_p) is given by the normalized sum of the visibility of the faces (Eq. 3).

$$\wedge_{V_p \in S} G_v(S, F) = \sum_{f \in F} \frac{max(V(V_p, f))}{|F|} \tag{4}$$

In this approach is selected the viewpoint that has a larger value of global visibility then all visible cells are removed from the submap of all other viewpoints and the process is repeated until there are no more candidates or the number of cells in the submaps for any remaining candidate viewpoint is below a threshold defined by the user.

3.3 Viewpoint sequence

Determine the optimum inspection sequence, which means the viewpoint sequence to be followed with a less cost, is done by a Travel Salesman Problem (TSP) algorithm, which is a NP-complete, well-studied problem with many fast solver algorithms Rosenkrantz *et al.* (1974); Golden *et al.* (1980); Reinelt (1994).

For the TSP, we are using the classic nearest neighbour approach which is an $O(n^2)$ algorithm Rosenkrantz *et al.* (1974). This solver is enough for our necessities because we are dealing with a small number of viewpoints (i.e., below 100). To solve the TSP problem, first, the distance between the viewpoints are compute and stored on a cost matrix. Two alternatives can be used here: use the euclidean distance between each pair of viewpoints or to compute a free path between them using the RRT* (running time n^2, where n is the number of viewpoints) which is a asymptotic optimal variant of the Rapidly-exploring Random Tree (RRT) (LaValle and Kuffner Jr, 2001). RRT is a sampling-base algorithm who builds a tree of collision-free vehicle configurations to find a collision-free path. Despite the more precise cost matrix using RRT* the running time is not negligible. For this reason the euclidean distance are been used to generate the cost matrix.

3.4 Final Path computation

The final path is build using the RRT* to connect the sequence of viewpoints produced by the nearest neighbor approach (Karaman and Frazzoli, 2011). The *a priori* model/environment has been used to define the obstacles in the environment. The RRT* has already been used in other view planning methods as Bircher *et al.* (2015) and underwater applications Hernández *et al.* (2016). We have used the open motion planning library (OMPL) Sucan *et al.* (2012) which has many state-of-the-art planning algorithms implemented.

The proposed algorithm output is a list of viewpoints $x, y, z, min\_yaw, max\_yaw$ and the paths that connects them, where min_yaw and max_yaw define the arc, with respect to the north, in which the vehicle can see the object to inspect once in the viewpoint x, y, z.

4. View planning algorithm evaluation

To evaluate the proposed approach various tests with different scenarios were done. We evaluate our approach with the different methods to view selection in a simple environment composed of cubic blocks. Later the tests were performed in more realistic and challenging scenarios.

For the tests we considered that our vehicle must keep a safety distance of 2 meters from *a priori* known model, the acquisition sensor has been defined with a minimum range of 2 meters, a maximum range of 5 meters and a field of view of 1200 degrees, with a ray every 2 degrees, mounted over a pan and tilt unit in which the pan angle is fixed but the tilt can rotate from -90 to 45 degrees. Additionally Girona 500 AUV can rotate over itself.

4.1 Evaluation of the Viewpoint Selection

The viewpoint selection were evaluated in a simple environment composed of cubic blocks. Table 1 shows the results obtained using each selection viewpoint approach, tree different results are shown, the number of visible nodes, time consuming and the global visibility. As presented in Eq. 4 the global visibility consider the total number of faces, but some of them are not possible to see, or because is pointed to inside the cube or because has a neighbour.

Table 1. Cubic Blocks - Tests

	Num. of Cand. (3)	Used Views	Visible Cells	Time Consuming (s)	Global Visibility
Based on Vis Nodes (1)	10	7	1066	13.99	33.71%
Based on Loc Vis(2)	10	6	914	176.60	31.53%
Based on Vis Nodes (1)	20	9	1373	28.16	34.98 %
Based on Loc Vis(2)	20	9	1066	542.43	31.93 %

(1) Based on the number of visible nodes
(2) Based on the local visibility
(3) Number of Candidate Viewpoints

Figure 5 shows the local visibility using 10 candidate viewpoints for each selection approach. Is worth to note that for each selected viewpoint the value of local visibility using the second approach is higher, but using the same stop criteria this approach select a smaller number of viewpoints resulting in a lower global visibility, as shown in Tab. 1.

Another important parameter is the number of fully visible faces and the quality of this data (evaluate based on the incidence angle and distance from the viewpoint to the face) using the submap of the viewpoint, since the global visibility

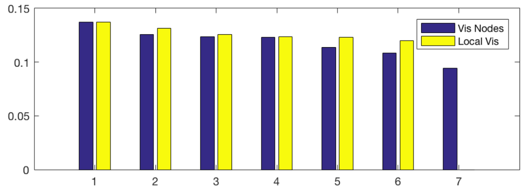

Figure 5. Local visibility per selected view.

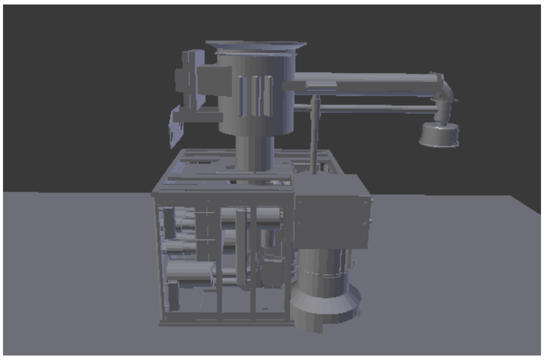

Figure 6. Tree structure.

take into account the full octomap. Table 2 shows the percentage of fully visible faces of the region seen by the viewpoint and the visibility quality.

Table 2. Fully Visible and Quality

	Based on Vis Nodes (1)	Based on Loc Vis(2)
Vp1	63.98% of quality 0.36	63.98% of quality 0.36
Vp2	46.61% of quality 0.25	100% of quality 0.17
Vp3	68.43% of quality 0.42	74.40% of quality 0.26
Vp4	98.98% of quality 0.37	100% of quality 0.20
Vp5	82.63% of quality 0.22	64.38% of quality 0.20
Vp6	100% of quality 0.19	99.46% of quality 0.35
Vp7	83.33% of quality 0.25	-

(1) Selection based on the number of visible nodes

(2) Selection based on the local visibility

Analyzing Tab. 1, Tab. 2 and Fig. 5 is possible to see that use the local visibility to select the viewpoints produce better viewpoints in terms of visibility as was expected, although a smaller volume is inspected.

4.2 Tree Structure

The first realistic scenario tested was a subsea tree structure commonly used in the oil and gas industry posed in the sea-bottom (see the tree structure in Fig. 6. The test were performed with 500 candidates viewpoints, because for this scenario more than 500 candidate viewpoints guarantee a coverage of more than 95.00% of the this structure (Floriani *et al.*, 2017).

To measure the coverage of the proposed algorithm, the algorithm have been executed using the nodes of a fine grid as candidates viewpoint to know which will be, approximately, the maximum coverage that can be obtained in this scenario with these conditions. It is worth noting that for a relatively small environment, like the one proposed here, this test can be done despite it generates 295.245 valid candidate viewpoints. However, for a larger environment the number of generated candidate viewpoints is intractable. On the other hand, this give us a ground truth to compare the proposed random exploration at least for a particular scenario. The number of visible cells obtained in this test is considered the maximum number of cells (i.e., 100% coverage) that can be seen in this scenario.

Figure 7 shows the tree structure, the viewpoints and the path to connect then to inspect the environment.

Table 3. Tree Structure - Tests

	Num. of Cand.	Visible Cells	Time Consuming (s)	Global Visibility
Based on Vis Nodes (1)	500	999	100.65	32.17 %
Based on Loc Vis(2)	500	427	3196.49	25.20 %

(1) Based on the number of visible nodes

(2) Based on the local visibility

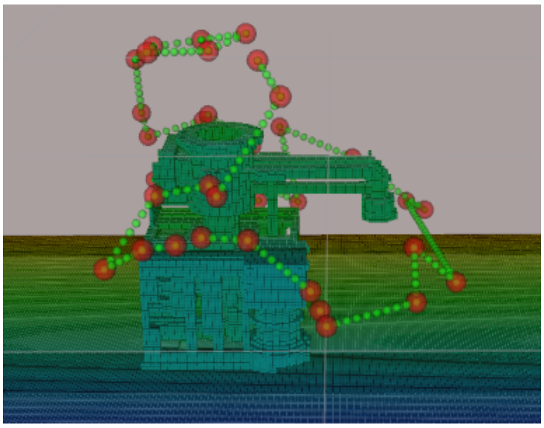

Figure 7. Path planned, viewpoints and structure visible cells.

5. Conclusions

This paper presented an off line viewpoint planning algorithm for underwater inspection of 3D complex structures and environments with a strong 3D relief, moreover different selection viewpoints approaches have been tested. The algorithm is able to find the minimum number of viewpoints and their positions to inspect and a path that connects them reducing the navigation cost.

The view planning algorithm has been tested in different scenarios. Obtaining viewpoints and trajectories using a few thousands of candidates views in short time producing high coverage results (i.e., greater than 95% for an appropriate number of candidates according to the area to inspect).

The use of local visibility to select the viewpoints improve the quality of the seen cells, although this process consumes much more time, since calculate the local visibility is a very time consuming. And keeping the same stop criteria (number of visible cells) using the local visibility approach result in a lower global visibility.

As future works, we plan to optimize the local visibility approach to consume less time and CPU memory allowing the use of this method for large environments with more candidate viewpoints.

6. ACKNOWLEDGEMENTS

The authors would like to thank Spanish project DPI2015-73978-JIN and CAPES: Edital Ciências do Mar 2 n° 43/2013.

7. REFERENCES

Almadhoun, R., Taha, T., Seneviratne, L., Dias, J. and Cai, G., 2016. "A survey on inspecting structures using robotic systems". *International Journal of Advanced Robotic Systems*, Vol. 13, No. 6, p. 1729881416663664.

Bircher, A., Alexis, K., Burri, M., Oettershagen, P., Omari, S., Mantel, T. and Siegwart, R., 2015. "Structural inspection path planning via iterative viewpoint resampling with application to aerial robotics". In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, pp. 6423–6430.

de Souza, L.A.P., 2006. *Revisão crítica da aplicabilidade dos métodos geofísicos na investigação de áreas submersas rasas*. Ph.D. thesis, Universidade de São Paulo.

Ellefsen, K.O., Lepikson, H.A. and Albiez, J.C., 2016. "Planning inspection paths through evolutionary multi-objective optimization". In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference*. ACM, pp. 893–900.

Englot, B.J., 2012. *Sampling-based coverage path planning for complex 3D structures*. Ph.D. thesis, Massachusetts Institute of Technology.

Floriani, B.L., Palomeras, N., Weihmann, L., Simas, H. and Ridão, P., 2017. "Model-based underwater inspection via viewpoint planning using octomap". In *Oceans, 2017*. IEEE, pp. 1–8.

Golden, B., Bodin, L., Doyle, T. and Stewart Jr, W., 1980. "Approximate traveling salesman algorithms". *Operations research*, Vol. 28, No. 3-part-ii, pp. 694–711.

Hernández, J.D., Istenič, K., Gracias, N., Palomeras, N., Campos, R., Vidal, E., García, R. and Carreras, M., 2016. "Autonomous underwater navigation and optical mapping in unknown natural environments". *Sensors*, Vol. 16, No. 8, p. 1174.

Hornung, A., Wurm, K.M., Bennewitz, M., Stachniss, C. and Burgard, W., 2013. "OctoMap: An efficient probabilistic 3D mapping framework based on octrees". *Autonomous Robots*. doi:10.1007/s10514-012-9321-0. URL `http://octomap.github.com`. Software available at `http://octomap.github.com`.

Jacobi, M., 2015. "Autonomous inspection of underwater structures". *Robotics and Autonomous Systems*, Vol. 67, pp. 80–86.

Karaman, S. and Frazzoli, E., 2011. "Sampling-based algorithms for optimal motion planning". *The international journal of robotics research*, Vol. 30, No. 7, pp. 846–894.

LaValle, S.M. and Kuffner Jr, J.J., 2001. "Randomized kinodynamic planning". *The International Journal of Robotics Research*, Vol. 20, No. 5, pp. 378–400.

Ososinski, M. and Labrosse, F., 2014. "Multi-viewpoint visibility coverage estimation for 3d environment perception volumetric representation as a gateway to high resolution data". In *Computer Vision Theory and Applications (VISAPP), 2014 International Conference on*. IEEE, Vol. 2, pp. 462–469.

Reinelt, G., 1994. *The traveling salesman: computational solutions for TSP applications*. Springer-Verlag.

Ribas, D., Palomeras, N., Ridao, P., Carreras, M. and Mallios, A., 2012. "Girona 500 auv: From survey to intervention". *IEEE/ASME Transactions on Mechatronics*, Vol. 17, No. 1, pp. 46–53.

Roberts, D. and Marshall, A.D., 1998. "Viewpoint selection for complete surface coverage of three dimensional objects." In *BMVC*. pp. 1–11.

Rosenkrantz, D.J., Stearns, R.E. and Lewis, P.M., 1974. "Approximate algorithms for the traveling salesperson problem". In *Switching and Automata Theory, 1974., IEEE Conference Record of 15th Annual Symposium on*. IEEE, pp. 33–42.

Roy, S.D., Chaudhury, S. and Banerjee, S., 2004. "Active recognition through next view planning: a survey". *Pattern Recognition*, Vol. 37, No. 3, pp. 429–446.

Schmid, K., Hirschmüller, H., Dömel, A., Grixa, I., Suppa, M. and Hirzinger, G., 2012. "View planning for multi-view stereo 3d reconstruction using an autonomous multicopter". *Journal of Intelligent & Robotic Systems*, Vol. 65, No. 1, pp. 309–323.

Scott, W.R., Roth, G. and Rivest, J.F., 2003. "View planning for automated three-dimensional object reconstruction and inspection". *ACM Computing Surveys (CSUR)*, Vol. 35, No. 1, pp. 64–96.

Siciliano, B. and Khatib, O., 2008. *Springer handbook of robotics*. Springer.

Sucan, I.A., Moll, M. and Kavraki, L.E., 2012. "The open motion planning library". *IEEE Robotics & Automation Magazine*, Vol. 19, No. 4, pp. 72–82.

Tarabanis, K.A., Allen, P.K. and Tsai, R.Y., 1995. "A survey of sensor planning in computer vision". *IEEE transactions on Robotics and Automation*, Vol. 11, No. 1, pp. 86–104.

Tarbox, G.H. and Gottschlich, S.N., 1995. "Planning for complete sensor coverage in inspection". *Computer Vision and Image Understanding*, Vol. 61, No. 1, pp. 84–111.

Zanoni, F.D. and de Barros, E.A., 2015. "A real-time navigation system for autonomous underwater vehicle". *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, Vol. 37, No. 4, pp. 1111–1127.

8. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.