EasyChair Preprint

№ 1493

Autonomous 3D Semantic Mapping of Coral Reefs

Md Modasshir, Sharmin Rahman and Ioannis Rekleitis

September 12, 2019

Chapter 1
Autonomous 3D Semantic Mapping of Coral Reefs

Md Modasshir, Sharmin Rahman, and Ioannis Rekleitis

Abstract This paper presents the first-ever approach for autonomous 3D semantic mapping of coral reefs. The position of corals in 3D coordinates and the type of the coral are presented in such a3D semantic map, The intended application of this work is coral reef health monitoring, as the current assessment is based entirely on direct or indirect human observation. The proposed system joins a convolutional neural network (CNN) with a direct visual odometry approach and a correlation filter based tracker, Kernelized Correlation Filter (KCF), to identify the different coral species detected. In addition to the coral classification, the 3D position of each coral is identified producing a semantic map of the observed reef. Each coral is identified once and tracked to prevent a recount. The number of different coral species encountered in two separate traversed areas is reported. Furthermore, the shape and size of a coral can be extracted from the 3D reconstruction enabling the extraction of volumetric data for subsequent studies. Experimental results from the coral reefs of Barbados verify the robustness and accuracy of the proposed approach.

1.1 Introduction

Coral reefs play an integral part of the marine ecosystem and are home to numerous aquatic species [28, 2]. However, coral reefs are on a steep decline in health and population due to global warming and ocean pollution. Scientists predict that by 2100, global temperature will increase by $2-4.5°C$. Because of such alarming situ-

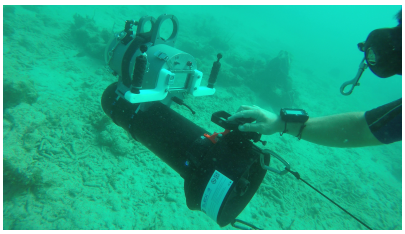


Fig. 1.1 Data collection over coral reefs, Barbados; Sensor Suite mounted on DPV.

Md Modasshir, Sharmin Rahman, and Ioannis Rekleitis
Computer Science and Engineering Department, University of South Carolina, USA. e-mail: [modasshm,srahman]@email.sc.edu,yiannisr@cse.sc.edu

ation, marine biologists are vigilantly monitoring the coral reef with the help of scuba divers, and by deploying autonomous or human-operated vehicles; see Fig. 1.1 where a diver collects data over a transect. If divers examine the reefs, they usually follow a pre-specified transact and stop over a coral reef to record the health of corals as shown in Fig. 1.2. In the case of data collection by vehicles or automated methods, the experts first annotate the collected images for coral species and then, analyze the recorded data to determine coral health. This monitoring process can be made more efficient by collecting visual data using autonomous or underwater vehicles and then only sending humans to analyze certain species if required. This efficiency can be achieved by utilizing a 3D semantic map of coral reefs which will provide useful information for both annotation and further study of the coral species. A 3D semantic map of a coral reef includes the 3D position of corals and their corresponding labels. Since the coral positions are known in such a map, scientists can navigate the same environment staying closer to certain species for detailed analysis. While there are works to automate the annotation process [3, 19, 22], there are few efforts to incorporate coral species label information into the 3D maps.

Fig. 1.2 Expert observing and recording health of corals [1].

Building a 3D semantic map of a coral reef will require two components: semantic information and a 3D map. The semantic information can be obtained using a coral detector. In recent times, deep learning has achieved tremendous success in detection [29, 18, 10, 27, 17]. A recent work by Modasshir *et al.* [23] proposed a CNN based automated annotation and population counting system for coral reefs while moving in transacts. This approach used the RetinaNet [17] as a backbone classification network. In this approach [23], an image is passed through the RetinaNet for detection, thereby producing a bounding box annotation of corals. The detected corals are then tracked by a Kernelized Correlation Filter (KCF) [13] for a fixed number of successive frames, and then the detection is performed again to detect newer corals in the images. The authors utilized intersection-over-union (IOU) among newly identified and old tracked corals to determine whether corals were observed before to prevent a recount of those corals. Our work is inspired by [23], and RetinaNet is used for detection to create semantic labelling of corals in the images.

The next step in creating a 3D semantic map is building the 3D map by calculating 3D positions of observed features in subsequent images. In recent years, many vision-based real-time state estimation algorithms have been developed for indoor and outdoor environments providing robust solutions covering large-scale areas using monocular or stereo cameras, to name a few Simultaneous Localization and Mapping (SLAM) systems – ORB-SLAM [24], SVO [9], DSO [6], LSD-SLAM [7], and SVIn [26]. However, underwater environments suffer from low visibility, poor contrast, light and color attenuation, haze and scattering. Most of the state estimation algorithms often fail or show poor performances in such challenging scenarios. Joshi *et al.* [14] presented a comprehensive study and performance analysis of state-

of-the-art open-source visual odometry algorithms. Among the vision-based SLAM systems, Direct Sparse Odometry (DSO) acquired the most accurate trajectory along with excellent 3D reconstruction in coral datasets collected by GoPro in Barbados. The DSO method [6] provides full photometric calibration which accounts for lens attenuation, gamma correction, and known exposure times. Hence, in this paper, we augment the architecture of DSO by incorporating coral semantic labels according to the CNN-based detection network. The semantic information acquired from the CNN-based detection network is color-coded into the reconstruction by DSO. Once the 3D semantic map is built, the shape and size of an individual coral can be retrieved by a least-square shape fitting method suitable for different types of corals.

In this work, we propose an automated method for creating 3D semantic map only from visual information and estimating the coral population in the observed area. Our primary contributions are three folds:

- Integrating CNN prediction into a point cloud generated by DSO to build a 3D semantic map.
- Calculating the volume of individual corals from features belonging to a coral.
- Estimating the coral population using detection and tracking.

We test our proposed method on two transacts in the Caribbean reef and show that our method reconstructs the 3D semantic map accurately and also counts the corals with accuracy on par with [23]. We also use an ellipsoid fitting method on a starlet coral to estimate its volume.

The next section discusses the works related to object detection and visual odometry. Section 1.3 illustrates the proposed methodology to combine CNN detection with DSO to build a semantic map. Experimental results from two transacts collected by GoPro cameras in Barbados are discussed in section 1.4. We summarize the paper and discuss future research directions in section 1.5.

1.2 Related Work

1.2.1 Object Detection

There are several works in coral classification in the literature. Most of the traditional approaches focused on using pixel-based information and textural appearances [21, 25, 20, 31]. Early work by Beijbom *et al.* [4] proposed to use color descriptors and texture at multiple scales. The authors proposed a Maximum Response filter bank for color and texture feature extraction. Each color channel was passed through the filter separately, and the filter responses were stacked. The stacked response was further passed through a Support Vector Machine with Radial Basis Function kernel to train the model. Other than the traditional approaches, there are a few works using deep learning for coral classification. Mahmood *et al.* [19] proposed a method combining learned features and hand-crafted features from

multi-scale patches. The authors extracted the learned features from the last convolutional layer of VGGNet [30]. These learned features and the handcrafted features were passed through a multi-layer perceptron classifier. In previous work [22] we proposed a densely connected CNN for coral classification. A point annotated dataset [4] was used and patches were extracted centered on point annotation. These patches are feed to the classifier where the patches were cropped three times at different sizes keeping the same center. State-of-the-art performance was achieved for coral classification. This paper extends our work in Modasshir *et al.* [23] as the dataset used contains severe class imbalance.

1.2.2 Visual Odometry

Vision-based SLAM systems can be categorized into direct and indirect (feature-based) method. Indirect methods (e.g., ORB-SLAM[24],OKVIS[15], SVIn[26]) include a pre-processing step accounting for the detection and tracking of features (e.g., corners) in consecutive images and optimizing the *geometric error*. Direct methods (e.g., DSO [6], LSD-SLAM [7]), on the other hand, consider pixel intensity and optimize the *photometric error* based on the direct image alignment without the need of any pre-processing step. Intermediate method, i.e., *semi-direct* also exists (e.g., SVO [9]) which combines both direct and indirect methods. While direct methods provide a dense or semi-dense representation of the environment, they often fail due to the *brightness consistency* assumption in low contrast environment with numerous lighting variations. As feature based methods use photometric invariant features, they show a better estimation of the pose but as a result of using only a sparse set of *keypoints* the 3D reconstruction of the surroundings is very sparse. Direct methods suffers heavily from large rotational change and slow initialization, however, the reconstruction by direct methods is less sparse compared to that of feature-based methods. DSO [6] has shown promising performance both in terms of tracking and 3D reconstruction. Instead of considering a dense formulation, DSO selects a sparse set of gradient-rich pixels in the image which retains excellent reconstruction along with accurate pose estimation. In this work, we have utilized DSO as it has provided the best reconstruction with the most accurate trajectories. As the field of Visual SLAM evolves, the proposed methodology can be extended to the latest products.

1.3 Approach

The proposed method works with videos or sequences of images. Therefore, the corals detected can be tracked over subsequent images which reduce the frequency of running the detection algorithm. This reduction makes the system faster and thus capable of running online. There are three steps of the proposed method: detection,

tracking and semantic mapping. For a sequence of images, f_1 to f_{n-1}, we acquire the location of corals in the image, f_0 from the detection system. These locations are provided as bounding box, termed Regions of Interest (ROI), and later are utilized to initialize the KCF tracker in order to track the observed corals in the next f_1 to f_{n-1} frames before the detection runs again on f_n. The newly detected corals in f_n are then matched against already tracked locations for overlap. If there is significant overlap among new ROIs and tracked ROIs, then merely the locations of tracked ROIs are updated to the newly detected ROIs. Otherwise, the system initializes new instances of KCF tracker with the new ROIs. In parallel to tracking, the ROIs for each frame are also passed to DSO which uses the ROIs to save the semantic information of the 3D points by color-coding them while creating the 3D map.

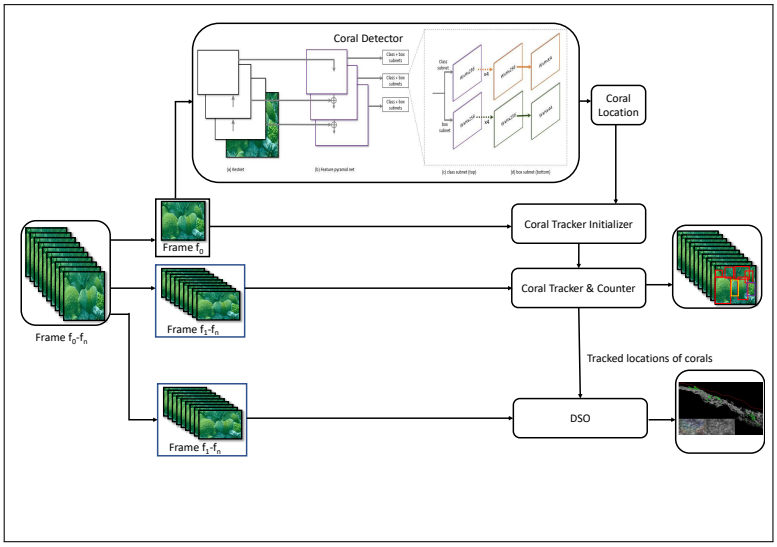


Fig. 1.3 Overview of the proposed approach.

1.3.1 Detection

RetinaNet: RetinaNet [17] is a one-stage-detector that includes a backbone network and two subnetworks. The backbone network is used to extract generic learned features from images. The Feature Pyramid Network (FPN) [16] is chosen as the backbone network. The FPN takes a single resolution image as input and then creates multi-scale features to augment a standard CNN with a top-down pathway and lateral connections. This use of multi-scale features enables the network to detect objects of different sizes. The FPN network can use various networks as a base network such as VGGNet[30], ResNets [12]. In our system, we choose ResNet with 50

layer variation as the base network of the FPN. The two subnetworks of the RetinaNet are utilized to classify and regress the bounding box locations. The final layer of the RetinaNet network is redesigned to accommodate eight classes of corals.

Focal Loss: The classification subnet of the RetinaNet is optimized using focal loss [17]. The focal loss is specifically suited for our classification problem as the focal loss is designed to handle a high-class imbalance in the training dataset. For a target class, k with an estimated probability $s_k \in [0,1]$, we define the focal loss as

$$loss(s_k) = -\alpha(1-s_k)^{\gamma} log(s_k)$$

where $\gamma \geq 0$ is a focusing parameter which can be tuned and $\alpha \in [0,1]$ is a weighting factor. Because of these parameters, γ and α, a lower loss is assigned to easily classifiable examples, thereby, an overall update of hyper-parameters in the network can focus on hard instances in the dataset. We choose the inverse of samples per class as the weighting factor of α. The γ controls how smoothly the loss for easily classifiable examples is down-scaled.

Smooth L1 Loss: The bounding box regression or ROIs subnet is optimized using smooth L1 loss. For predicted ROIs p and ground truth ROIs g, the smooth L1 loss is

$$Loss_{roi} = \sum_{i \in x,y,w,h} smooth_{L_1}(p_i - g_i)$$

where x and y are the top-left coordinates of the ROI, and w and h are width and height of the ROI. The function $smooth_{L_1}$ is defined piece-wise as

$$smooth_{L_1}(q) = \begin{cases} 0.5\,q^2, \text{if } |q| < 1. \\ |q| - 0.5, \text{otherwise.} \end{cases} \tag{1.1}$$

Training: The pretrained weights on ImageNet dataset[5] were used to initialize the base network, ResNet of the FPN network. We initialize all other layers to zero-mean Gaussian distribution with a standard deviation of 0.01. The hyper-parameters were optimized by stochastic gradient descent (SGD). The training epochs are 150 with 0.001 initial learning rate and $1e^{-5}$ decay.

1.3.2 Tracking

The KCF [13] tracker utilizes multiple channels of color images to improve a correlation filter. Let the Gaussian shaped response be $r = [r_1, r_2, ..., r_j]^T \in \mathbb{R}$ and the input vector be $c_d \in \mathbb{R}^{j \times 1}$.

The filter weights w of the KCF are updated by optimizing:

$$\dot{w} = arg \min_{w} \sum_{j=1}^{J} (r_j - \sum_{d=1}^{D} R_{j,d}^T W_d)_2^2 + \lambda \|W\|_2^2 \tag{1.2}$$

where $R_{j,d}$ is the j-step circular shift of the input vector R_d, r_j is the j-th element of r, $W = [W_1^T, W_2^T, ..., W_D^T]^T$ where $W_d \in \mathbb{R}^{J*1}$ refers to the filter of the d-th channel [32].

1.3.3 Counting

For a number of frames, f_1 to f_{n-1}, the CNN-based detector localizes corals by providing bounding boxes in the frame f_1. The KCF tracker instances are initialized to these bounding boxes and then tracked in the frames from f_2 to f_{n-1}. The detector performs bounding box prediction again on the f_n frame. The predicted bounding boxes are then matched against KCF tracked bounding boxes using Intersection-over-Union (IOU) [8] to prevent a recount. To be considered as a new coral object, the bounding box prediction, R_p and the tracked bounding box R_t must have overlap ratio M_o less than 0.5. The overlap ratio, M_o, is defined as

$$M_o = \frac{area(R_p \cap R_t)}{area(R_p \cup R_t)} \tag{1.3}$$

where $R_p \cap R_t$ indicates the intersection between predicted and tracked bounding boxes and $R_p \cup R_t$ indicates their union. If the overlap ratio, M_o, is below 0.5, the detected coral counts as a new coral object and a new instance of KCF tracks the new coral. Otherwise, KCF tracked bounding box coordinates are updated to that of the CNN detector produced bounding box.

1.3.4 Semantic Mapping

DSO: DSO provides a *direct* and *sparse* formulation for a monocular visual odometry system by combining the benefits of the direct approach and the flexibility of sparse approaches, i.e., efficient, joint optimization of all model parameters including the inverse depth in a reference frame, camera motion, and camera intrinsics. By sampling from pixels across all image regions with a high-intensity gradient (e.g., edges) and omitting the smoothness prior used in other direct methods, DSO is capable of real-time tracking. DSO accounts for the full photometric calibration which leads to accurate and robust state estimation.

Like the more recent Visual Odometry (VO) or Visual Inertial Odometry (VIO) systems, DSO follows windowed optimization and marginalization. A window of a fixed number of active keyframes is maintained. DSO provides a fully direct probabilistic model by continuous optimization of the photometric error over a local window of the current frames. The geometry of a 3D point is presented only by one parameter – the inverse depth in the reference frame. The photometric error of a point is represented as the weighted SSD over a small neighborhood of pixels, thus providing a good trade-off between computational requirement and robustness to the motion blur. Keyframe tracking is based on the two-frame direct image alignment. New keyframes are created if the field-of-view changes or the translation causes occlusions and disocclusions. Keyframe marginalization takes place when the active set of variables becomes too large; at first, all the points belonging to that keyframe as well as points that have not been observed in the last two keyframes are marginal-

ized. Initialization in DSO is critical and slow. Being monocular, DSO requires low translational change and cannot handle sizeable rotational change.

Integrating CNN and DSO: Once DSO initializes and starts building the map, the detection and tracking algorithms begin their operations on a parallel thread. For each frame, DSO is only modified to utilize semantic information provided as bounding boxes by the detection or the tracking pipeline to color-code the reprojected feature-points in the point cloud.

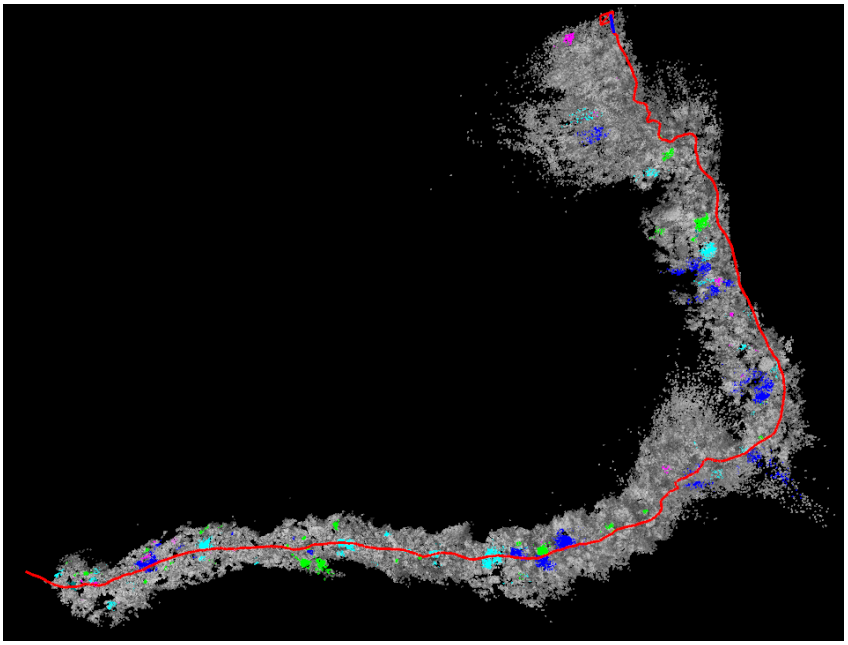


Fig. 1.4 3D semantic map of the 3 min trajectory. Features from different corals are displayed in different colors according to table 1.1.

1.3.5 Coral Volume Estimation

There are different shape fitting methods. We consider the ellipsoid fitting method as ellipsoids are particularly suitable for the most common corals in the collected data: starlet and brain. We follow the algebraic ellipsoid fitting method. Generally, nine parameters are required to define an ellipsoid. The required parameters are three coordinates of the center, three semi-axes a, b, c and three rotational angles. Given data points, P, unknown ellipsoid parameters, X and a design matrix A, we have

$$A_{n,u}.\delta X_u = l_n \tag{1.4}$$

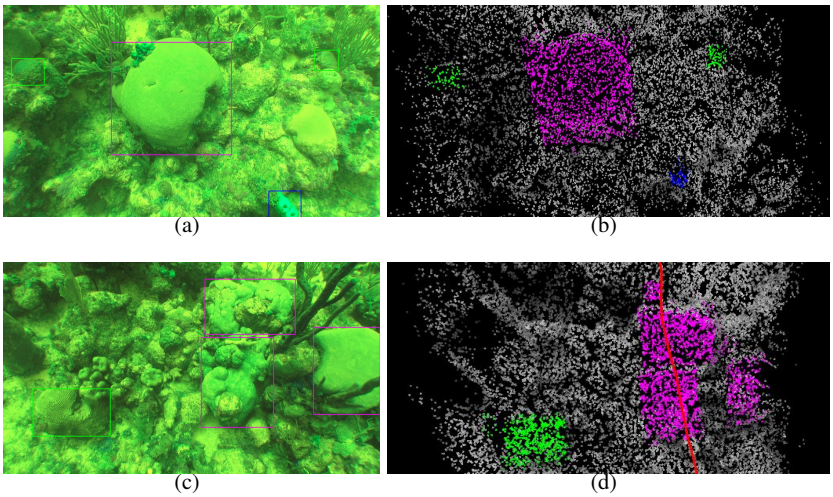


Fig. 1.5 Detail of the 3D semantic map of Fig. 1.4. (a),(c) are the raw images (b),(d) are the detected features in 3D color coded. In (a) the main object is a starlet coral (marked in magenta), and in (c) there are some sponges and a few brain corals (marked in blue and green accordingly) .

where n is the number of data points, u is the number of unknown parameters. For this equation to have a solution, we need $u \leq n$. We choose the $l_2 - norm$ method to solve the system of linear equations. Once we have the semi-axes a, b, c, the volume V is

$$V = \frac{4}{3}\pi abc \tag{1.5}$$

1.4 Experimental Results and Discussion

To validate our approach, we have selected two different trajectories: 3 min 27 sec (henceforth, referred to as the 3 min trajectory) trajectory with a length of 40.29 meters and a 10 min trajectory with a length of 315.39 meters. Both trajectories were collected by a GoPro camera over live coral reefs. The 3 min trajectory data were collected by a scuba diver while the 10 min trajectory data were collected by utilizing an underwater scooter. The CNN detector was trained to detect the following corals of seven types: Brain, Mustard, Star, Starlet, Maze, Sponge, Finger and Fire coral. However, in both trajectories, Finger and Fire coral are absent. Therefore, we do not report count for these two types of corals.

The color codes used to represent different corals in the 3D semantic map is shown in table 1.1. Fig. 1.4 shows the 3D semantic map of the 3 min trajectory. The point cloud clearly shows the coral features belonging to an individual coral are

close together in 3D space and forms the shape of the coral. Fig. 1.5 shows two pairs of observed image and their corresponding zoomed-in semantic map. As can be seen from Fig. 1.5, the reconstructed 3D point cloud retains the shape and size of the observed corals well. It is worth noting that the 3D point cloud is constructed using many successive frames, and therefore, does not correspond precisely to the location of the bounding box predictions in the raw images in Fig. 1.5.

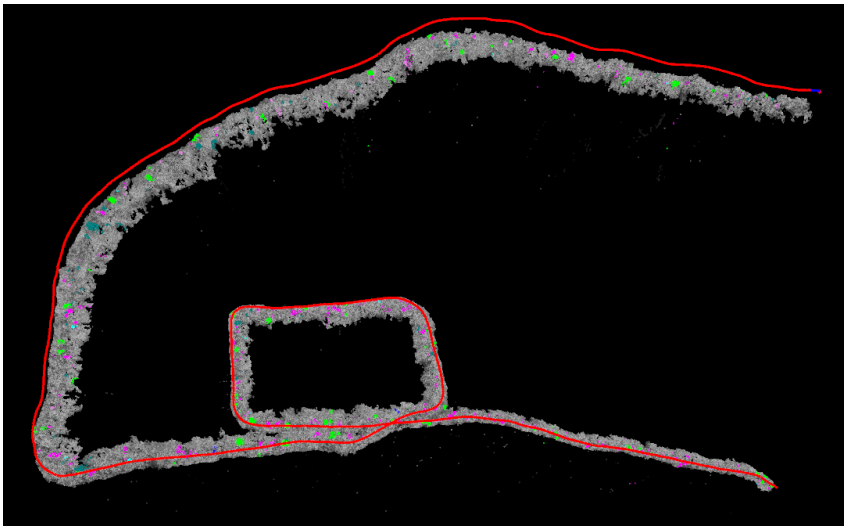


Fig. 1.6 3D semantic map of the 10 minute trajectory. Features from different corals are displayed in different colors according to table 1.1.

Fig. 1.6 shows the 3D semantic mapping of the 10 min trajectory. Similar to the 3 minutes trajectory mapping, the coral features are located nearby in 3D space and preserve the shape of the corals in the 3D mapping. However, the reconstruction could have been improved if loop closure was used along with the DSO method. Due to the absence of loop closure, we found the coral features being duplicated and positioned in different 3D space when the camera revisits the same place. It is worth noting that we tried Direct Sparse Odometry with Loop Closure [11] and the loop closure did not work. Most of the SLAM systems fail to produce accurate trajectories on both of our transacts. In the future, we plan to use SVIn[26], which fuses sonar and vision sensors, to acquire precise ground truth trajectory for better reconstruction.

Fig. 1.7 shows two pairs of close-up semantic reconstruction along with corresponding raw images and the predicted bounding boxes. Fig. 7(c) shows a critical case for the rebuilding of the semantic map. When the bounding boxes of different corals overlap, the color-coding process cannot retain spatial information appropriately, therefore, also loses the shape and size data of the coral. This challenge comes

	Brain	Mustard	Star	Starlet	Maze	Sponge
Color	Green	Purple	Teal	Magenta	Aqua	Blue

Table 1.1 Color codes used in 3D semantic mapping for different types of corals.

inherently with the use of bounding box prediction and can only be overcome by CNN-based semantic segmentation.

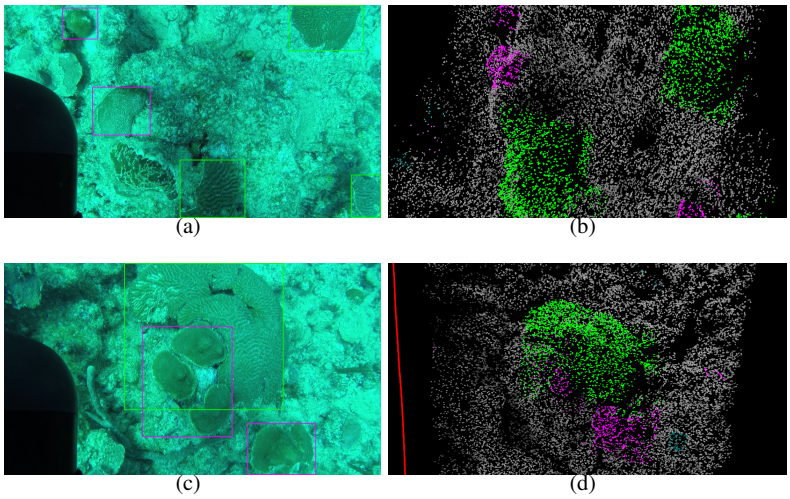


Fig. 1.7 Detail of the 3D semantic map of Fig. 1.6. (a),(c) are the raw images (b),(d) are the detected features in 3D color coded. In (a) two starlet coral and three brain corals are detected (marked in magenta and green respectively), and in (c) there are one brain coral and two starlet coral detected (marked in green and magenta in order) .

1.4.1 Counting

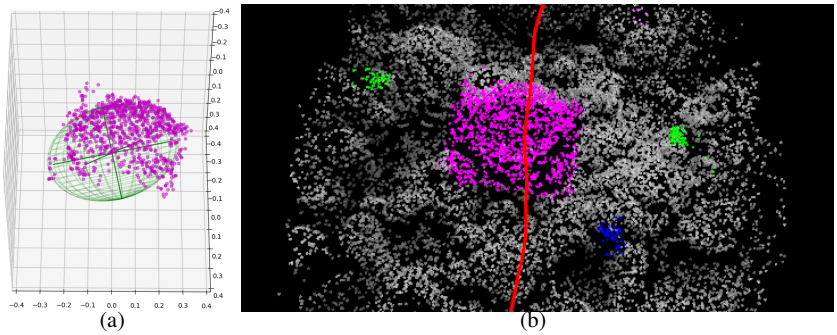
Quantitative results are reported in table 1.2 for both trajectories for four types of corals: Brain, Mustard, Star, Starlet, Maze, and one non-coral type: Sponge. We report the count of each kind of coral objects by our prediction and tracking system as well as human-annotated count. The counting system performs favorably well when compared to the work of Modasshir *et al.* [23]. Empirically, the detection performed relatively well on the 10 min trajectory compared to the 3 min trajectory, because the 3 min trajectory images have greenish appearance due to severe red-channel suppression which is a known phenomenon in underwater photos.

	Brain	Mustard	Star	Starlet	Maze	Sponge
T1:3min	31/37	0/2	5/7	37/43	11/15	17/17
T2:10min	90/97	39/47	68/75	161/176	26/28	5/6

Table 1.2 Coral counting for different trajectories. CNN-prediction/Human-Annotated

1.4.2 Volumetric Coral Evaluation

Different types of corals require different shape fitting methods. Starlet and Brain corals are the most common types in our trajectories. Both coral types are of ellipsoid shape. We show ellipsoid fitting on a starlet coral in Fig. 1.8. The features corresponding only to the starlet coral in the point cloud (shown in magenta) are used for the ellipsoid fitting. The calculated radii of the ellipsoid are $0.24m, 0.26m, 0.31m$ which closely matches our empirical observation of the starlet coral. The volume of the starlet coral using equation 1.5 is $0.081m^3$. In future work, we plan to integrate different shape fitting methods suitable for different types of corals.



(a) (b)

Fig. 1.8 Ellipsoid fit on the features from starlet coral. (a) shows the result of ellipse fitting and (b) displays corresponding features (in magenta) in the point cloud

1.5 Conclusion

In this paper, we presented a system for building a 3D semantic map of a coral reef in a transect using only visual information as well as estimating the coral population in the transact. In our proposed method, we showed how to incorporate semantic information retrieved from CNN based detector into a SLAM system to create an enriched 3D semantic map. The proposed approach will enable marine scientists

to monitor and assess the health of coral reefs of a much larger area swiftly. By retrieving the shape of corals from transects over time in specific areas, our method will also allow evaluating erosion of corals.

While visual information helped create a 3D semantic map, more sensor fusion will make the 3D semantic map more accurate by helping in the localization of 3D points. Future work will investigate how to integrate 3D points belonging to a coral object into tracking the coral, thereby, replacing the KCF tracker and making the system even faster.

1.6 Acknowledgement

This work was made possible through the generous support of National Science Foundation grants (NSF 1513203).

References

1. : Coral reef monitoring — reef recharge. http://reefrecharge.com/coral-reef-monitoring/ (Accessed on 04/18/2019).
2. : Corals of the world - variation in species. http://coral.aims.gov.au/info/taxonomy-variation.jsp (Accessed on 11/30/2017).
3. Beijbom, O., Edmunds, P.J., Kline, D., Mitchell, B.G., Kriegman, D., et al.: Automated annotation of coral reef survey images. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). (2012) 1170–1177
4. Beijbom, O., Edmunds, P.J., Kline, D.I., Mitchell, B.G., Kriegman, D.: Automated annotation of coral reef survey images. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). (2012) 1170–1177
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR09. (2009)
6. Engel, J., Koltun, V., Cremers, D.: Direct sparse odometry. **40**(3) (2018) 611–625
7. Engel, J., Schps, T., Cremers, D.: LSD-SLAM: Large-Scale Direct Monocular SLAM. Volume 8690. Springer Int. Publishing (2014) 834–849
8. Everingham, M., Eslami, S.A., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes challenge: A retrospective. International journal of computer vision **111**(1) (2015) 98–136
9. Forster, C., Zhang, Z., Gassner, M., Werlberger, M., Scaramuzza, D.: SVO: Semidirect Visual Odometry for Monocular and Multicamera Systems. **33**(2) (2017)
10. Fu, C.Y., Liu, W., Ranga, A., Tyagi, A., Berg, A.C.: Dssd: Deconvolutional single shot detector. arXiv preprint arXiv:1701.06659 (2017)
11. Gao, X., Wang, R., Demmel, N., Cremers, D.: Ldso: Direct sparse odometry with loop closure. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE (2018) 2198–2204
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2016) 770–778
13. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: High-speed tracking with kernelized correlation filters. IEEE Transactions on Pattern Analysis and Machine Intelligence **37**(3) (2015) 583–596

14. Joshi, B., Rahman, S., Cain, B., Johnson, J., Kalitazkis, M., Xanthidis, M., Karapetyan, N., Hernandez, A., Quattrini Li, A., Vitzilaios, N., Rekleitis, I.: Experimental comparison of open source vision-inertial-based state estimation algorithms. (2019) (under review).
15. Leutenegger, S., Lynen, S., Bosse, M., Siegwart, R., Furgale, P.: Keyframe-based visual-inertial odometry using nonlinear optimization. **34**(3) (2015) 314–334
16. Lin, T.Y., Dollár, P., Girshick, R.B., He, K., Hariharan, B., Belongie, S.J.: Feature pyramid networks for object detection. In: CVPR. Volume 1. (2017) 4
17. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. (2017) 2980–2988
18. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: European conference on computer vision, Springer (2016) 21–37
19. Mahmood, A., et al.: Coral classification with hybrid feature representations. In: IEEE Int. Conf. on Image Processing (ICIP). (2016) 519–523
20. Marcos, M.S.A., David, L., Peñaflor, E., Ticzon, V., Soriano, M.: Automated benthic counting of living and non-living components in ngedarrak reef, palau via subsurface underwater video. Environmental monitoring and assessment **145**(1-3) (2008) 177–184
21. Mehta, A., Ribeiro, E., Gilner, J., van Woesik, R.: Coral reef texture classification using support vector machines. In: Int. Conf. on Computer Vision Theory and Applications (VISAPP). (2007) 302–310
22. Modasshir, M., Li, A.Q., Rekleitis, I.: Mdnet: Multi-patch dense network for coral classification. In: MTS/IEEE Oceans Charleston, Charleston, SC, USA (Oct. 2018) (accepted)
23. Modasshir, M., Rahman, S., Youngquist, O., Rekleitis, I.: Coral Identification and Counting with an Autonomous Underwater Vehicle. In: IEEE International Conference on Robotics and Biomimetics (ROBIO), Kuala Lumpur, Malaysia (Dec. 2018) 524–529
24. Mur-Artal, R., Montiel, J., Tardos, J.: ORB-SLAM: A Versatile and Accurate Monocular SLAM System. Volume 31. (2015) 1147–1163
25. Pizarro, O., Rigby, P., Johnson-Roberson, M., Williams, S.B., Colquhoun, J.: Towards image-based marine habitat classification. In: OCEANS 2008, IEEE (2008) 1–7
26. Rahman, S., Li, A.Q., Rekleitis, I.: Sonar Visual Inertial SLAM of Underwater Structures. In: IEEE International Conference on Robotics and Automation, Brisbane, Australia (May 2018) 5190–5196
27. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2016) 779–788
28. Rogers, C., Garrison, G., Grober, R., Hillis, Z., F.ie, M.: Coral reef monitoring manual for the caribbean and western atlantic. Virgin Islands National Park, 110 p. Ilus. (1994)
29. Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y.: Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229 (2013)
30. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR **abs/1409.1556** (2014)
31. Stokes, M.D., Deane, G.B.: Automated processing of coral reef benthic images. Limnology and Oceanography: Methods **7**(2) (2009) 157–168
32. Sun, C., Wang, D., Lu, H., Yang, M.H.: Correlation tracking via joint discrimination and reliability learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 489–497