This WACV 2021 paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Visual tracking of deepwater animals using machine learning-controlled robotic underwater vehicles

Kakani Katija, Paul L D Roberts, Joost Daniels, Alexandra Lapides, Kevin Barnard,
Mike Risi, Ben Y Ranaan
Monterey Bay Aquarium Research Institute
{kakani, proberts, joost, alapides, kbarnard, mrisi, byranaan}@mbari.org

Benjamin G Woodward, Jonathan Takahashi
CVision AI
{benjamin.woodward, jonathan.takahashi}@cvisionai.com

Abstract

The ocean is a vast three-dimensional space that is poorly explored and understood, and harbors unobserved life and processes that are vital to ecosystem function. To fully interrogate the space, novel algorithms and robotic platforms are required to scale up observations. Locating animals of interest and extended visual observations in the water column are particularly challenging objectives. Towards that end, we present a novel Machine Learning-integrated Tracking (or ML-Tracking) algorithm for underwater vehicle control that builds on the class of algorithms known as tracking-by-detection. By coupling a multi-object detector (trained on in situ underwater image data), a 3D stereo tracker, and a supervisor module to oversee the mission, we show how ML-Tracking can create robust tracks needed for long duration observations, as well as enable fully automated acquisition of objects for targeted sampling. Using a remotely operated vehicle as a proxy for an autonomous underwater vehicle, we demonstrate continuous input from the ML-Tracking algorithm to the vehicle controller during a record, 5+ hr continuous observation of a midwater gelatinous animal known as a siphonophore. These efforts clearly demonstrate the potential that tracking-by-detection algorithms can have on exploration in unexplored environments and discovery of undiscovered life in our ocean.

1. Introduction

In order to explore a landscape as vast as the ocean, researchers have turned to robotics as the enabling technology for discovery [38]. While efforts to observe the ocean have gone on for many generations, these combined efforts have barely scratched the surface, with some estimates claiming that 95% of the ocean remains unexplored and 91% of marine life remains unknown to science [29]. The ocean's midwaters, the region of the ocean that connects the lighted surface waters to the deep seafloor, is the largest habitable ecosystem on Earth [10], and we know little about the inhabitants in this region. The midwater environment is a fully three-dimensional space without any functional boundaries, human-made features, and light, and artificially augmenting the scene can disturb the behavior of animals researchers wish to study [34]. In order to observe animal behavior, careful consideration of illumination conditions (both light intensity and wavelength [44]), platform noise, and hydrodynamic disturbance are required to minimize disruptions [45]. To address this need, observational platforms are required to non-invasively execute targeted sampling and maintain a persistent presence to track animals over a period of time.

Underwater tracking of animals has a long history, primarily using modalities like acoustics due to their long-range sensing capabilities [3, 5, 20, 22]. Despite the widespread use of acoustics, imaging is still an attractive modality due to its low cost and capability of providing high-resolution spatiotemporal data needed for identifying individual animals and quantifying behavior. Terrestrial applications of animal tracking rely on imaging for both long- and short-range applications [1, 18], however long-range underwater imaging is intrinsically challenging due to the optical properties of seawater, and the noisy visual field due to marine snow, particles, and small animals and plants in the water column. Despite these challenges, underwater imaging continues to be effective for short-range applications, including vision-based underwater vehicle tracking.

Vision-based underwater vehicle tracking, or visual servoing, has been around for several decades, and with developments in modern computer vision and machine learning, this field has seen renewed interest [21, 45, 16]. Efforts to automate visual tracking of realistic underwater objects to generate real-time target range and vehicle control were demonstrated by [28] in the early 1990s in controlled environments. Later, efforts to evaluate vision algorithms for in situ tracking and detection of behavioral mode changes of animals using supervised machine learning in the form of support vector machines were conducted on pre-collected underwater imagery [34, 32]. These algorithms, called *JellyTrack*, were eventually demonstrated in the field using the remotely operated vehicle (ROV) *Ventana*, culminating in tracking a single jellyfish for 89 minutes. Since then, *JellyTrack* has undergone a complete overhaul using OpenCV libraries, and has subsequently been integrated onto multiple underwater vehicles, including a new class of underwater vehicle with similar tracking performance called the *Mesobot* [45]. While these efforts have been fruitful, additional methods like tracking-by-detection and machine learning classifiers, have shown promise for robust tracking, which is necessary for longer duration, 24+ hr-long deployments that are required to study critical behaviors of midwater animals.

Tracking-by-detection represents a class of algorithms where a detector is applied to image data, and detections are subsequently tracked to obtain positions of objects in 2D (or 3D with stereo imaging) space [1]. Tracking-by-detection has had widespread applicability in the automotive industry [40], aerial vehicles [8], and construction [18], and these algorithms are beginning to emerge in underwater applications [39, 16]. Using in situ and synthetic imagery of an underwater robot, researchers have demonstrated how reduced conventional neural network architectures can be applied to track other similar-looking robots during convoying [39, 16]. Since the detector was trained on images of a known object with invariant size, the 3D position could be reconstructed from a single-camera view deployed on the trailing vehicle. While these in situ demonstrations were successful in tracking-by-detection of underwater vehicles, the sizes of midwater animals are variable and cannot be known a priori. To determine the 3D position of a midwater animal of unknown size relative to a tracking vehicle, a stereo imaging system is required. In addition, while underwater vehicles look dissimilar to other natural objects underwater and a single-shot detector could inform tracking robustly, many midwater animals have similar features; this requires simultaneous detection of multiple classes to distinguish between objects of interest and objects to be ignored.

Here we present a Machine Learning-integrated Tracking (or ML-Tracking) algorithm that incorporates multi-class detectors and stereo imaging to track midwater animals for long durations (Figure 1). A detector was trained on in situ, underwater color imagery from the Monterey Bay Aquarium Research Institute's (MBARI) Video Annotation and Reference System (VARS) and monochrome imagery collected during multiple midwater dives using the stereo camera system described in [45]. ML-Tracking algorithms were demonstrated in midwater using ROV *MiniROV* in the Monterey Bay National Marine Sanctuary. While delays due to COVID-19 have prevented field deployments of the most recent iteration of ML-Tracking described here, we will demonstrate its performance on previously collected in situ data. Finally, we propose additional enhancements that could improve robustness of tracking and enable fully autonomous acquisition of tracking targets, which will lead to targeted sampling and persistent observations of phenomena in the ocean.

2. Robotic platform for at-sea trials

Field trials of earlier versions of ML-Tracking were conducted using the ROV *MiniROV* in the Monterey Bay National Marine Sanctuary near Midwater Station 1 (latitude: 36° 41.8792 N, longitude: 122° 2.9929 W) with bottom depths exceeding 400 m. While a remotely operated vehicle is normally manually operated, by integrating and testing ML-Tracking algorithms on *MiniROV*, this enabled fully autonomous operations of the vehicle and, if needed, human intervention. Multiple dives with ROV *MiniROV* were made in the spring and summer of 2019 and 2020, and in the autumn of 2019. ROV *MiniROV* (Figure 2) is a 1500 m-rated flyaway vehicle that is equipped with a main camera (Insite Pacific Incorporated Mini Zeus II), a stereo imaging system (Allied Vision G-319B monochrome cameras and Marine Imaging Technologies underwater housings with domed-glass optical ports), a pair of red lights (Deep Sea Power and Light MultiRay LED Sealite 2025 at 650–670 nm), and additional vehicle sensors. Red illumination was used throughout our tracking trials to minimize disruptions and changes in animal behavior (e.g., avoidance, attraction). Additional details on the stereo tracking hardware can be found in [46]. The vehicle reference frame is centered on the "left" stereo camera optical port, with the positive z-direction (or range) oriented forward of the vehicle. On board the ship (or topside), a Tensorbook (Lambda Labs) laptop with an Nvidia RTX 2070 GPU was used to ingest stereo video data, run models and 3D tracking software, and issue control commands to the vehicle.

3. Machine learning-integrated tracking algorithm

The ML-Tracking algorithm described here involves a multi-class *RetinaNet* [19] detection model, 3D stereo tracker subroutines, and a supervisor module that sends commands to the vehicle controller (Figure 3). Detection of

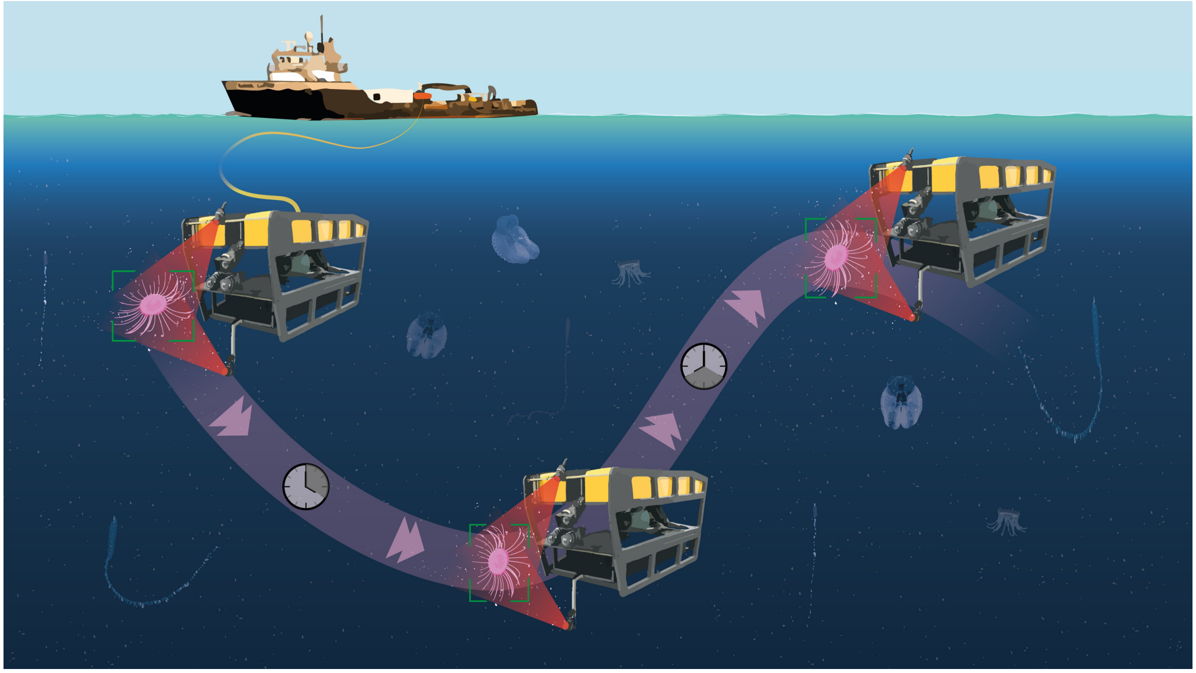

Figure 1. Midwater exploration and discovery of inhabitants, like the jellyfish *Solmissus* sp., require the use of deep-diving underwater vehicles with the ability to autonomously track targets over long durations.

the potential target classes is conducted simultaneously on imagery from both the left and right stereo cameras. The 2D positions, or bounding boxes, of the detected classes are then used to determine the 3D position of the detected objects using a number of criteria described below (Figure 4). The 3D positions of the detected objects are then shared with the supervisor module, which evaluates whether the detected concepts match the target class of interest and modifies the vehicle behavior based on the phase of the autonomous tracking mission (Figure 5) and the distance between the target and the vehicle (or range). The vehicle controller has been adapted from [35, 45, 46] for ROV *MiniROV*. The software extensively uses Lightweight Communications and Marshaling (LCM, [12]) to communicate between various modules. The ML models were containerized and deployed using the nvidia-docker codebase [31].

3.1. Multi-class detector and in situ training data

Training data for the multi-class detector came from two separate sources: (1) color imagery from previous, expertly annotated ROV dives found in the VARS database and (2) monochrome imagery collected from ROV *MiniROV* dives using the stereo camera system described above. The color imagery used for training corresponded to representative midwater animals commonly observed in the upper water column of Monterey Bay (Supplementary Figure 1), which included a subset from *FathomNet*, an underwater image training set [2]. To augment this data, monochrome imagery of animals corresponding to the same classes were also used. Objects in the training data were annotated and localized by experts using a number of software tools (e.g., GridView [36], RectLabel [15], and Tator [6]). In addition to classes that identify animals to the genus or family taxonomic level (e.g., *Aegina*, *Atolla*, *Bathochordaeus*, *Bathocyroe*, *Beroe*, *Calycophorae*, *Cydippida*, *Lobata*, *Mitrocoma*, *Physonectae*, *Poeobius*, *Prayidae Solmissus*, *Thalassocalyce*, *Tomopteridae*), parts or associated structures of animals (e.g., *Bathochordaeus house*, *outer filter*, siphonophore *nectosome*) were also localized to evaluate the efficacy of detection on these complex objects. This process of in situ data curation resulted in 28485 localized images for 17 different classes, with 205–6927 images per class in the training set.

Using the *RetinaNet* architecture with a *ResNet50* [11] backbone pre-trained on *ImageNet* [7], we trained and fine-tuned parameters using the aforementioned in situ training data. Since all available pre-trained *ResNet* models were trained on color (three-channel) backbones, we evaluated methods that enable the applicability of these models for transfer learning on monochrome (single-channel) imagery. While significant effort has been done on "colorizing" single-channel imagery, by either using standard computer vision or deep learning solutions [47, 4], due to the nature of the tracking targets in midwater (e.g., semi-transparent body surfaces, reflectance properties under differing lighting conditions, variability in poses, etc.), these

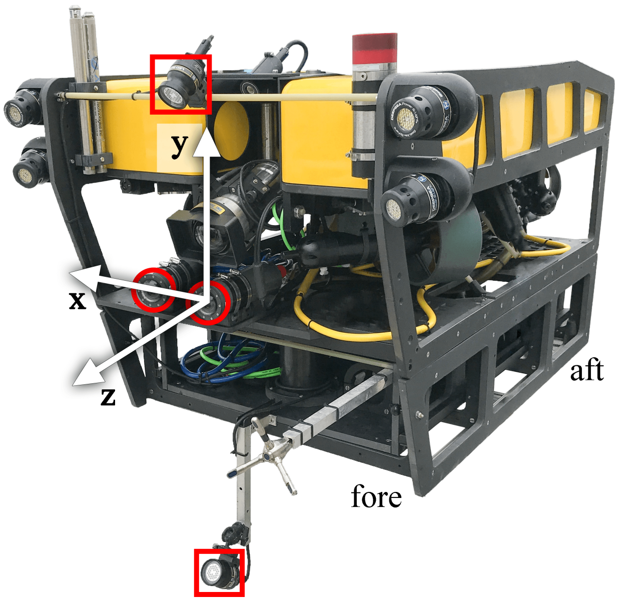

Figure 2. Machine Learning-Integrated Tracking (ML-Tracking) algorithms were deployed using ROV *MiniROV*. The imaging and illumination hardware used for the tracking demonstration are indicated by the red circles and squares, respectively. The origin of the vehicle reference frame is located at the viewport of the stereo camera seen on the right (referred to as the "left" camera), with positive and negative z-direction corresponding to the fore and aft vehicle directions, respectively.

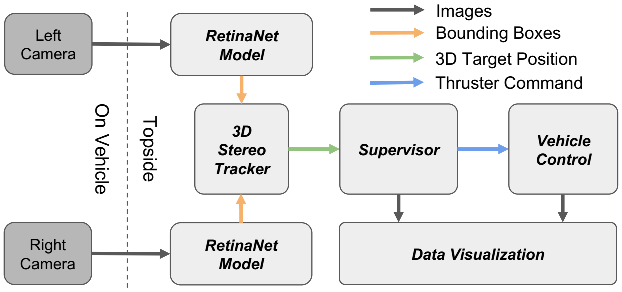

Figure 3. Overview of the ML-Tracking algorithm. Colored arrows indicate the specific data type being passed between modules, where black, orange, green, and blue corresponds to images, bounding boxes, 3D target positions, and thruster commands, respectively.

techniques had significant drawbacks. By comparing a number of these methods, we found that a rather simple solution – replicating the single channel to make a three-channel image – was sufficient for enabling fine-tuning of pre-trained networks using the mixed-channel images.

3.2. 3D stereo tracker

A high level depiction of the 3D stereo tracker can be seen in Figure 4, and can be broken into vehicle state and image measurement, stereo matching, and track association and management processes. All stereo computations include optical calibration data for both cameras and stereo calibration parameters (translation vector and rotation matrix between cameras). The ML-Tracking algorithm is therefore highly dependent on an accurate optical calibration.

3.2.1 Vehicle state and image measurement

The core of the 3D stereo tracker module is an unscented Kalman filter (UKF, [43]) that accepts a measurement in image space and generates a state estimate in 3D vehicle coordinates. The measurement vector consists of 8 elements (horizontal and vertical positions of top-left and bottom-right points of corresponding left and right bounding boxes) and the state consists of 5 elements (3D center of target position in vehicle coordinates, width and height of target). Using different coordinate systems for the state (vehicle) and measurement (image) allows for noise in the bounding box position to be propagated into the 3D state estimate, and provides a more useful track representation since it includes distance to target and target size. Measurement covariance is defined to be proportional to bounding box size and is set such that the standard deviation in pixel position is 10% of the bounding box size. The state transition function does not include a motion model since animal movement is typically nonlinear. Instead, motion is accounted for by state transition covariance, which is set dynamically using the estimated size of the object such that the standard deviation in apparent size or position may vary by 15% of the object size from frame to frame.

3.2.2 Finding stereo bounding box pairs

Advances in finding stereo bounding box pairs (or stereo matching) have been achieved due to the application of object detectors based on convolutional neural networks. Leveraging object-centric model output, stereo matching algorithms can integrate correspondence matching into the network to outperform pixel-level matching [33], compute object-specific disparity [42], or estimate volumetric bounding boxes from monocular imagery [30]. As these efforts show, 3D estimation at the object level can improve accuracy and reduce computational complexity compared to methods that operate at the unsegmented pixel level. Here we need to estimate only the centroid of the object of interest whose location is represented by a bounding box. To do this, we find matching stereo pairs by using "stereo intersection over union" (IOU_s), and computation of IOU_s is done by comparing the bounding box projection from the first camera view in the second camera view (dark blue rectangle in Figure 4) with the bounding box in the second camera view (light blue rectangle in Figure 4). The higher the

IOU_s value, the higher the likelihood that the two bounding boxes correspond to the same target. We compute stereo IOU between all possible pairs of bounding boxes and define a cost matrix where $\text{Cost}_s = (1 - IOU_s)$. Assignments between pairs are then computed using the Hungarian algorithm [17], and each assigned pair becomes a measurement that may be used to either update an existing track or start a new one.

3.2.3 Track association and management

The midwater tracking task requires multi-object tracking to be successful. To address this need, each track maintains a separate UKF. In order to associate bounding box pairs with tracks, we convert each measurement into state-space by fitting a 3D box position (x, y, z, width, height) to the four 3D points that correspond to a stereo box pair. We then assume multivariate Gaussian distributed noise in state-space and compute the square of the Mahalanobis Distance [24] of the measurement using the track's current state mean and covariance. The square of the Mahalanobis Distance is used to populate a cost matrix between measurements and tracks (or $\text{Cost}_{m,t}$), where

$$\text{Cost}_{m,t} = (\mathbf{x}_m - \boldsymbol{\mu}_t)^T \Sigma^{-1} (\mathbf{x}_m - \boldsymbol{\mu}_t), \qquad (1)$$

and assignments are made using the Hungarian algorithm [17].

Finally, if a measurement is assigned to a track, its original measurement vector is used to update the track's UKF. A simple heuristic called "track score" is used to aid in track management. For each frame a track receives an update, we add one to the track score. For each frame a track does not receive an update, we subtract one from the track score and the track is in "coast" mode. If a track's score falls below a threshold of -100 it is deleted. New tracks are created when a valid box pair has been found but the box pair cannot be associated to any existing tracks.

3.3. Supervisor for vehicle control

The vehicle supervisor acts as a mission executive, overseeing and evaluating the quality of the target classes being tracked, and modifying the vehicle behavior based on this input (Figure 5). While the vehicle is in "search" mode, the vehicle auto depth and heading are activated, and the thruster power is at 20% in the forward direction. If object detections are absent, the vehicle continues to search. If object detections occur, the vehicle transitions to the "acquire" mode, where the vehicle approaches the target with PID control and the vehicle begins to slow down. If the detected object does not match the target class, the vehicle returns to the "search" mode; if there are inconsistent classifications, the vehicle holds its position until either (1) the target class has been identified and the vehicle mode transitions to "confirm" or (2) times out and returns to the "search" mode. Once the detected object has been confirmed to be the target class, the vehicle continues to hold position on the target with PID control. At this juncture, the vehicle can either await external verification (e.g., supervised autonomy) of the object, or transition to the "track" mode where the vehicle continues to hold its relative position to the target constant. This mode will continue until either the 3D position data for the target class is lost or the vehicle mission has been completed. The "confirm" mode also allows for external communications that can enable supervised autonomy operations in the future. In the instance that the 3D position is lost, the vehicle will transition into the "reacquire" mode, where auto depth and heading are reengaged and thruster power is at 5% in the forward direction. The supervisor loop will then continue until the vehicle mission has been completed.

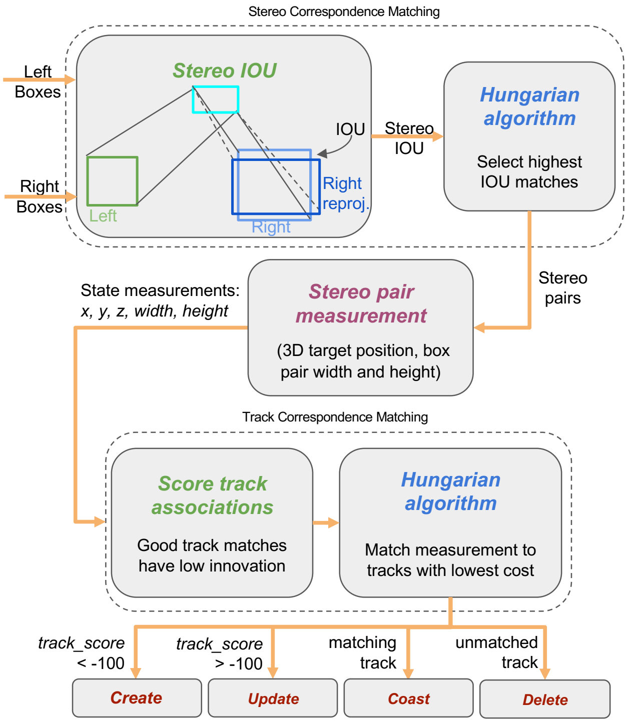
Stereo Correspondence Matching
Left Boxes
Right Boxes
Stereo IOU
IOU
Left
Right
Right reproj.
Stereo IOU
Hungarian algorithm
Select highest IOU matches
Stereo pairs
State measurements: x, y, z, width, height
Stereo pair measurement
(3D target position, box pair width and height)
Track Correspondence Matching
Score track associations
Good track matches have low innovation
Hungarian algorithm
Match measurement to tracks with lowest cost
track_score < -100
track_score > -100
matching track
unmatched track
Create
Update
Coast
Delete

Figure 4. Schematic representation of the "3D Stereo Tracker" module in the ML-Tracking algorithm.

4. Results

Using the color and monochrome image training data, the multi-class detector was trained and its performance is summarized in the confusion matrix shown in Figure 6. The detector performed well on most of the classes, however it performed moderately well on *Calycophora* nectosome, *Mitrocoma*, and *Cydippida*. We suspect that this per-

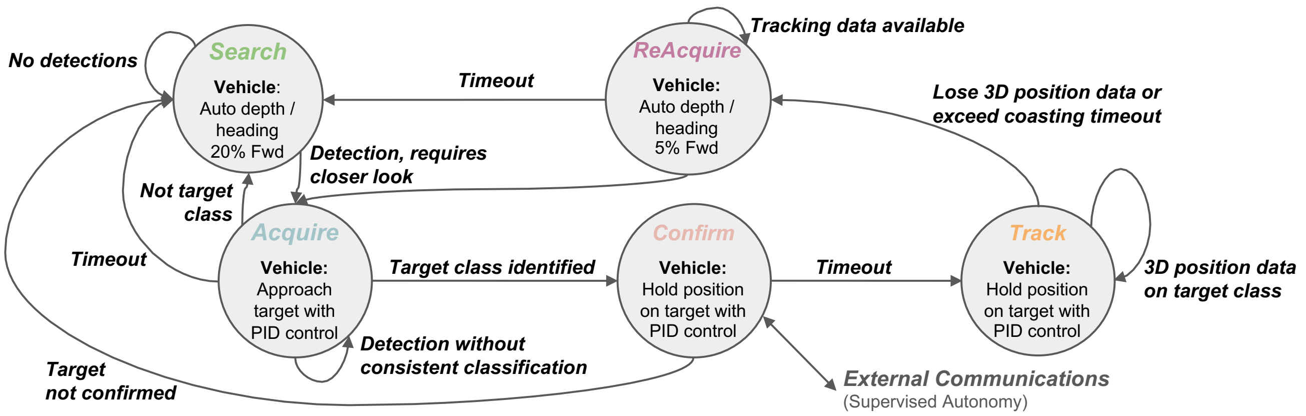
Search
Vehicle: Auto depth / heading 20% Fwd
No detections
ReAcquire
Vehicle: Auto depth / heading 5% Fwd
Tracking data available
Timeout
Lose 3D position data or exceed coasting timeout
Detection, requires closer look
Not target class
Acquire
Vehicle: Approach target with PID control
Timeout
Target class identified
Confirm
Vehicle: Hold position on target with PID control
Timeout
Track
Vehicle: Hold position on target with PID control
3D position data on target class
Detection without consistent classification
Target not confirmed
External Communications
(Supervised Autonomy)

Figure 5. Schematic representation of the ”supervisor” module in the ML-Tracking algorithm.

formance is linked to difficulties associated with detecting highly transparent objects, and these classes can be notoriously difficult for a human operator to see during ideal vehicle operations and lighting conditions. For the class *Calycophorae* nectosome, it is most confused with *Physonectae* nectosome, which is another type of siphonophore.

Over the course of our ML-Tracking at-sea testing, we have amassed nearly 50 hours of recorded footage from the stereo cameras and ROV science camera. In that period of time, we determined that tracking scenarios can be generalized by five different functional categories (or use cases) that include, from least to most challenging: (1) a steadily swimming, single-object, (2) a dynamically swimming, single-object, (3) a nested class, multi-object, (4) a multi-object, multi-class occlusion, and (5) a multi-object, single-class occlusion. While steadily or dynamically swimming or moving targets are straightforward concepts, nested classes are common representations of mid-water animals that are associated with different structures or have complex morphology. For example, in the case of a giant larvacean *Bathochordaeus* that lives within a mucus house within an outer filter [14], being able to distinguish between the animal and its mucus house can lead to differing but equally valuable scientific lines of inquiry [37, 13]. In the case of a gelatinous colonial organism called a siphonophore – with physonectae, prayidae, and calycophorae taxonomic subgroups – their bodies are comprised of features (e.g., nectosome, siphosome; [23]) that change pose and readily occlude other features, with some being easier to detect than others.

The performance of the ML-Tracking algorithm for examples of all 5 use cases is shown in Table 1, framegrabs at various time intervals in Figure 7, and corresponding videos in Supplementary Videos 1–5. In addition to the generalized use cases, performance of ML-Tracking for a single-object – a siphonophore *Lychnagalma* sp., representing the longest duration observation we collected during our at-sea deployments – is also shown in Table 1, with corresponding framegrabs presented in Supplementary Figure 2.

Independent of use case, for clip durations ranging from 14 to 18987 seconds (or 5.27 hrs), the percent amount of time that the vehicle controller received 3D position information from the ML-Tracking algorithm (binned at 1 s intervals) remained above 99%. For use cases 1–4, the standard deviation of range (distance between the vehicle and target) and altitude (the height between the center of the vehicle's field of view and the target) never exceeded 5 cm; the standard deviation of the bearing (angle between the center of the vehicle's field of view and the target) never exceeded 6 deg for all scenarios shown. The higher values of range standard deviation for case 5 and the long duration observation is due to tracking being transferred to another target during that time interval. This behavior is clearly shown in Figure 7 for case 5, when tracking (indicated by the orange bounding box) is transferred from the larger jellyfish to the smaller one in the final image. Supplementary Figure 3 shows vehicle behavior during the long duration siphonophore tracking event.

5. Discussion and future considerations

Performance improvements of vision-based underwater vehicle tracking or visual servoing, can be achieved in a number of ways that could include advancements in platforms and hardware, imaging, and algorithms. While development of more agile imaging platforms can be costly, modifications can be made to the imaging and illumination system to not only augment image volume size [27] but increase the responsiveness of the imaging system to rapid target movements by mounting the imaging system to a pan-and-tilt [28]. Additionally, there are cases whereby imaging as the sole sensing modality fail, especially in long-range applications, and more robust tracking could be achieved

Tracking Type	Clip Length (s)	Control (%)	Range Std. Dev. (cm)	Bearing Std. Dev. (deg)	Altitude Std. Dev. (cm)
Single-Object, Steady Swimming, *Case 1*	63	100	1.2	0.7	0.5
Single-Object, Dynamic Swimming, *Case 2*	21	100	1.4	3.4	4.6
Multi-Object, Nested Classes, *Case 3*	61	100	2.8	1.6	1.4
Multi-Object, Multi-Class Occlusion, *Case 4*	23	100	2.1	0.8	0.6
Multi-Object, Single-Class Occlusion, *Case 5*	14	100	13.6	5.3	4.1
Single-Object, Long Duration	18987	100	10.4	3.2	8.3

Table 1. Performance summary of the ML-Tracking algorithm on field-collected data using ROV *MiniROV*.

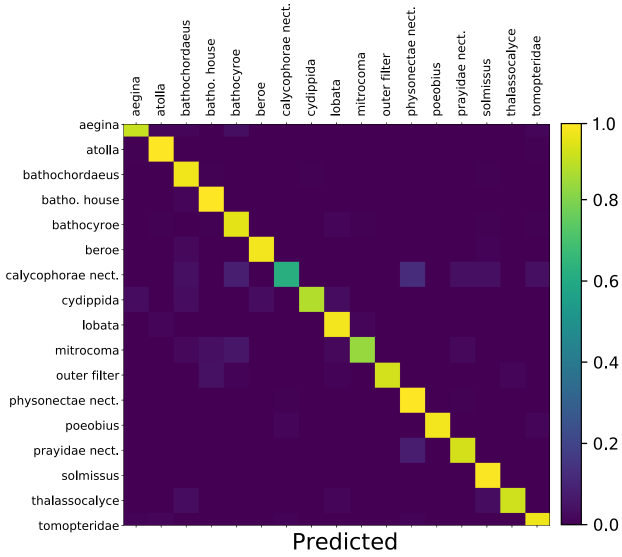

Figure 6. Confusion matrix of the multi-class detector used in the ML-Tracking algorithm.

by integrating and transitioning to acoustic sensing in those scenarios [26].

As we demonstrate here with ML-Tracking, improvements to visual servoing by incorporating multi-object detectors based on real-world image data and stereo tracking can significantly enhance underwater tracking performance. Additional improvements to the tracking algorithm could include incorporating target motion for entity matching [18], motion prediction to enable fine tuning of tracking parameters [32], and methods utilized by [1] to account for cases of short- or long-duration occlusions of targets; all of these approaches could increase the robustness of tracker performance. Based on the generalized use cases we highlight here (see Table 1, Figure 7, and Supplementary Videos 1–5), occlusion caused by objects of the same class is responsible for catastrophic failure of ML-Tracking, and will be addressed in future versions of the algorithm.

As tracking-by-detection algorithms mature, their implementation will hopefully lead to longer-duration observations while also enabling fully automated acquisition of targets that models are able to identify as known or unknown. While full autonomy – as defined by Level 5 in [41] for self-driving cars – will be more readily attainable for terrestrial rather than underwater applications, providing options to remotely oversee mission operations to provide as-needed human-in-the-loop interventions or target validations can be strategic. Autonomous surface vessels have demonstrated the capability to track subsea assets, including transiting autonomous underwater vehicles, providing a mobile hotspot that can enable capability to connect remote operators [48]. Exploiting the External Communications (or Supervised Autonomy) interface with the supervisor module (Figure 5) can be used to enable a human operator to periodically review subsampled images of tracking targets and confirm or deny the object of interest. These human labels can also be added to future training sets to evaluate the efficacy of future ML-Tracking algorithms, necessarily speeding up the development timeline for underwater vehicles to explore and discover life in the ocean's midwaters.

While we have focused our efforts in midwater for scientific and practical purposes (e.g., less cluttered image fields, lack of substrate, minimization of collision risk), these ML-Tracking algorithms are certainly applicable to seafloor environments. Being able to survey, search for, and observe objects of interest on the benthos could enable novel observations of coral spawning, cephalopod maternal care, and organismal associations with various substrate including rare and valuable minerals for deep sea mining. By incorporating promising algorithms that use imagery to make close-range navigation decisions to avoid collisions [25], automated acquisition and tracking of visual targets near the seafloor can be accomplished. Finally, by combining our multi-object detectors as a baseline for known objects in the water column with a probabilistic framework to enable observations of "interesting" features [9], this could ultimately enable exploration in unexplored underwater habitats and lead to descriptions of undescribed life in our ocean.

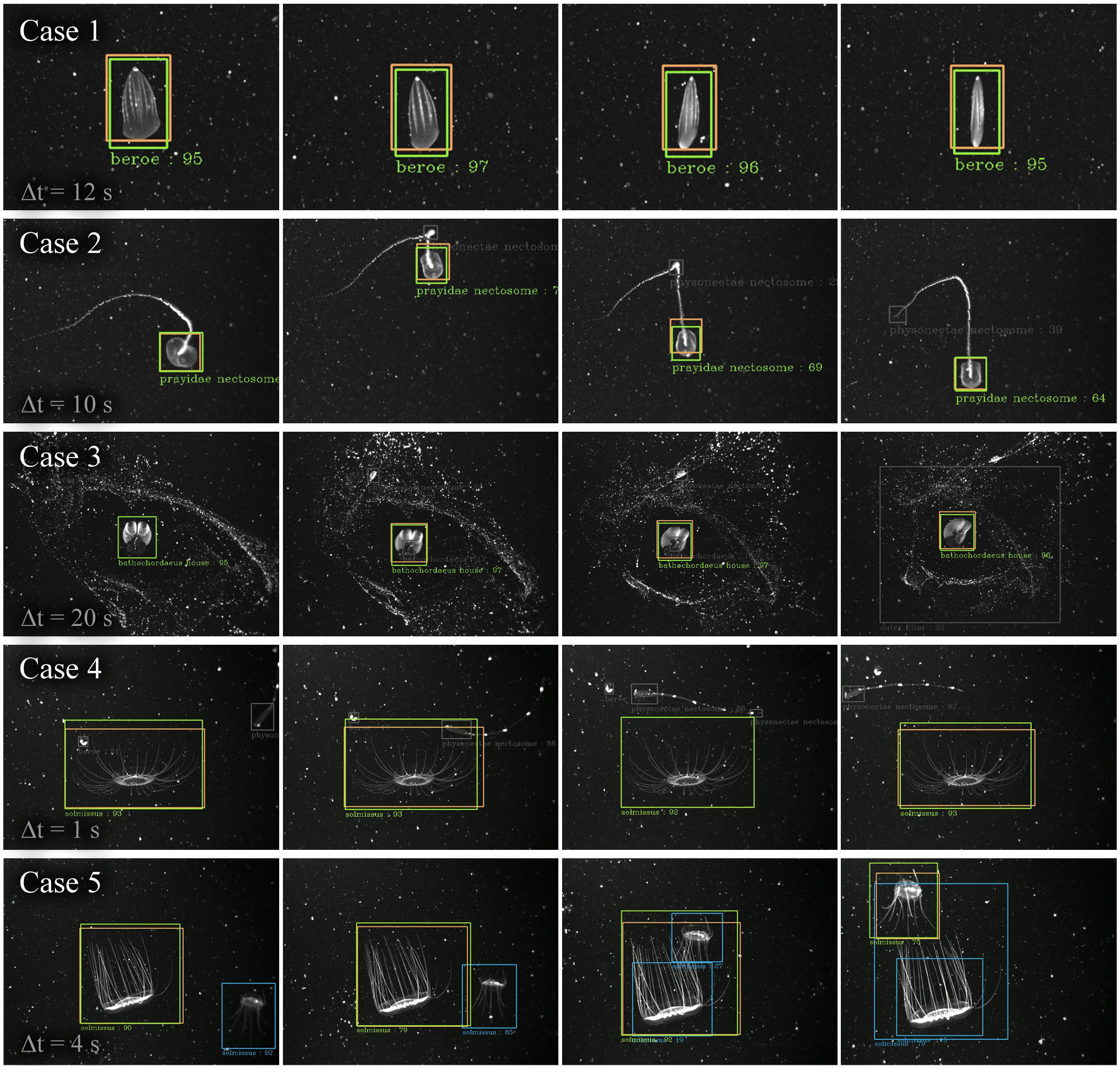

Figure 7. Framegrabs demonstrating the performance of the ML-Tracking algorithm across all generalized use cases. First row: single-object (a ctenophore, *Beroe*), steady swimming (case 1), left camera, 33% crop; second row: single-object (a siphonophore, prayidae, nectosome), dynamic swimming (case 2), left camera, 50% crop; third row: multi-object (a giant larvacean, *Bathochordaeus*, with its mucus house and outer filter), nested classes (case 3), left camera, 75% crop; fourth row: multi-object (a jellyfish, *Solmissus*, and siphonophore, physonectae, nectosome), multi-class occlusion (case 4), right camera, 75% crop; fifth row: multi-object (two jellyfish, *Solmissus*), single-class occlusion (case 5), right camera, 85% crop. Green bounding boxes correspond to the detected location of the target class; orange bounding boxes correspond to the object being tracked; blue bounding boxes indicate an object that the detector has identified as a potential target; gray bounding boxes indicate objects of a non-target class.

6. Acknowledgments

The authors would like to thank Brett Hobson, Dale Graves, Alana Sherman, Frank Flores, and the crew of RV *Rachel Carson* for their engineering and operational contributions to this effort, and Steve Rock (Stanford Univ.) for his invaluable input and guidance. Additional thanks go to members of the Video Lab (especially Lonny Lundsten and Nancy Jacobsen Stout), Brian Schlining, Oceane Boulais, and Katy Croff Bell for their contributions to FathomNet, the image training set that was used for machine learning detector development. This effort was supported by NSF-GEO-OTIC grant #1812535 (to KK) and the David and Lucile Packard Foundation.

References

[1] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[2] O. Boulais, B. Woodward, B. Schlining, L. Lundsten, K. Barnard, K. Croff Bell, and K. Katija. Fathomnet: An underwater image training database for ocean exploration and discovery. *arXiv preprint arXiv:2007.00114*, 2020.

[3] V. Chandrasekhar, W. K. G. Seah, Y. S. Choo, and H. V. Ee. Localization in underwater sensor networks. In *Proceedings of the 1st ACM international workshop on underwater networks*, pages 33–40, 2006.

[4] Z. Cheng, Q. Yang, and B. Sheng. Deep colorization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 415–423, December 2015.

[5] P. Corke, C. Detweiler, M. Dunbabin, M. Hamilton, D. Rus, and I. Vasilescu. Experiments with underwater robot localization and tracking. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 4556–4561, 2007.

[6] CVision AI, Inc. Tator. `https://github.com/cvisionai/tator`, 2019.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[8] D. Du, Y. Qi, H. Yu, Y. Yang, K. Duan, G. Li, W. Zhang, Q. Huang, and Q. Tian. The unmanned aerial vehicle benchmark: Object detection and tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 370–386, 2018.

[9] Y. Girdhar, P. Giguère, and G. Dudek. Autonomous adaptive exploration using realtime online spatiotemporal topic modeling. *International Journal of Robotics Research*, 33(4):645–657, 2014.

[10] S. H. D. Haddock, L. M. Christianson, W. R. Francis, S. Martini, C. W. Dunn, P. R. Pugh, C. E. Mills, K. J. Osborn, B. A. Seibel, C. A. Choy, C. E. Schnitzler, G. I. Matsumoto, M. Messié, D. T. Schultz, J. R. Winnikoff, M. L. Powers, R. Gasca, W. E. Browne, S. Johnsen, K. L. Schlining, S. von Thun, B. E. Erwin, J. F. Ryan, and E. V. Thuesen. Insights into the biodiversity, behavior, and bioluminescence of deep-sea organisms using molecular and maritime technology. *Oceanography*, 30(4):38–47, 2017.

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[12] A. S. Huang, E. Olson, and D. C. Moore. LCM: Lightweight Communications and Marshalling. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4057–4062. IEEE, 2010.

[13] K. Katija, R. E. Sherlock, A. D. Sherman, and B. H. Robison. New technology reveals the role of giant larvaceans in oceanic carbon cycling. *Science Advances*, 3(5):e1602374, 2017.

[14] K. Katija, G. Troni, J. Daniels, K. Lance, R. E. Sherlock, A. D. Sherman, and B. H. Robison. Revealing enigmatic mucus structures in the deep sea using *DeepPIV*. *Nature*, pages 1–5, 2020.

[15] R. Kawamura. Rectlabel. `https://rectlabel.com/`.

[16] K. Koreitem, J. Li, I. Karp, T. Manderson, F. Shkurti, and G. Dudek. Synthetically trained 3D visual tracker of underwater vehicles. In *OCEANS 2018 MTS/IEEE Charleston*, 2019.

[17] H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.

[18] Y. J. Lee and M. W. Park. 3D tracking of multiple onsite workers based on stereo vision. *Automation in Construction*, 98(August 2018):146–159, 2019.

[19] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):318–327, 2020.

[20] Y. H. Lin, S. M. Wang, L. C. Huang, and M. C. Fang. Applying the stereo-vision detection technique to the development of underwater inspection task with PSO-based dynamic routing algorithm for autonomous underwater vehicles. *Ocean Engineering*, 139(February):127–139, 2017.

[21] D. Lindsay, H. Yoshida, Takayuki Uemura, H. Yamamoto, S. Ishibashi, J. Nishikawa, J. Reimer, R. J. Beaman, R. Fitzpatrick, K. Fujikura, and T. Maruyama. The untethered remotely operated vehicle PICASSO-1 and its deployment from chartered dive vessels for deep sea surveys off Okinawa, Japan, and Osprey Reef, Coral Sea, Australia. *Marine Technology Society Journal*, 46(4):20–32, 2012.

[22] J. Luo, Y. Han, and L. Fan. Underwater acoustic target tracking: A review. *Sensors (Switzerland)*, 18(1):1–38, 2018.

[23] G. O. Mackie, P. R. Pugh, and J. E. Purcell. Siphonophore biology. In *Advances in Marine biology*, volume 24, pages 97–262. Elsevier, 1988.

[24] P. C. Mahalanobis. Reprint of: P. C. Mahalanobis (1936) "On the Generalised Distance in Statistics". *Sankhya A*, 80(1):1–7, 2018.

[25] T. Manderson, J. C. G. Higuera, R. Cheng, and G. Dudek. Vision-based autonomous underwater swimming in dense coral for combined collision avoidance and target selection. In *IEEE International Conference on Intelligent Robots and Systems*, pages 1885–1891, 2018.

[26] F. Mandić, I. Rendulić, N. Mišković, and D. Na. Underwater object tracking using sonar and USBL measurements. *Journal of Sensors*, 2016:8070286:1–8070286:10, 2016.

[27] P. Mariani, I. Quincoces, K. H. Haugholt, Y. Chardard, A. W. Visser, C. Yates, G. Piccinno, G. Reali, P. Risholm, and J. T. Thielemann. Range-gated imaging system for underwater monitoring in ocean environment. *Sustainability (Switzerland)*, 11(1), 2018.

[28] R. L. Marks, S. M. Rock, and M. J. Lee. Automatic object tracking for an unmanned underwater vehicle using real-time image filtering and correlation. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, volume 3, pages 337–342, 1993.

[29] C. Mora, D. P. Tittensor, S. Adl, A. G. B. Simpson, and B. Worm. How many species are there on earth and in the ocean? *PLoS Biology*, 9(8):e1001127, 2011.

[30] A. Mousavian, D. Anguelov, J. Flynn, and J. Kosecka. 3D bounding box estimation using deep learning and geometry. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5632–5640. IEEE, 2017.

[31] NVIDIA. Nvidia-docker. `https://github.com/NVIDIA/nvidia-docker`.

[32] A. M. Plotnik and S. M. Rock. Improving performance of a jelly-tracking underwater vehicle using recognition of animal motion modes. *Proceedings of the Unmanned Untethered Submersible Technology Conference (UUST)*, 2003.

[33] Z. Qin, J. Wang, and Y. Lu. Triangulation learning network: From monocular to stereo 3D object detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7607–7615. IEEE, 2019.

[34] J. Rife and S. M. Rock. Segmentation methods for visual tracking of deep-ocean jellyfish using a conventional camera. *IEEE Journal of Oceanic Engineering*, 28(4):595–608, 2003.

[35] J. H. Rife and S. M. Rock. Design and validation of a robotic control law for observation of deep-ocean jellyfish. *IEEE Transactions on Robotics*, 22(2):282–291, 2006.

[36] P. L. D. Roberts. GridView. `https://bitbucket.org/mbari/gridview/`, 2020.

[37] B. H. Robison, K. R. Reisenbichler, and R. E. Sherlock. Giant larvacean houses: Rapid carbon transport to the deep sea floor. *Science*, 308(5728):1609–1611, 2005.

[38] B. H. Robison, K. R. Reisenbichler, and R. E. Sherlock. The coevolution of midwater research and ROV technology at MBARI. *Oceanography*, 30(4):26–37, 2017.

[39] F. Shkurti, W. D. Chang, P. Henderson, M. J. Islam, J. C. G. Higuera, J. Li, T. Manderson, A. Xu, G. Dudek, and J. Sattar. Underwater multi-robot convoying using visual tracking by detection. In *IEEE International Conference on Intelligent Robots and Systems*, volume 2017, pages 4189–4196, 2017.

[40] S. Sivaraman and M. M. Trivedi. Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis. *IEEE transactions on intelligent transportation systems*, 14(4):1773–1795, 2013.

[41] Society for Automotive Engineering. Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles (j3016). Technical report, Society for Automotive Engineering, 2016.

[42] J. Sun, L. Chen, Y. Xie, S. Zhang, Q. Jiang, X. Zhou, and H. Bao. Disp R-CNN: Stereo 3D object detection via shape prior guided instance disparity estimation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10545–10554. IEEE, jun 2020.

[43] E. A. Wan and R. Van Der Merwe. The unscented Kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, volume 31, pages 153–158. IEEE, 2000.

[44] E. A. Widder, B. H. Robison, K. R. Reisenbichler, and S. H. D. Haddock. Using red light for in situ observations of deep-sea fishes. *Deep Sea Research Part I: Oceanographic Research Papers*, 52(11):2077–2085, 2005.

[45] D. R. Yoerger, M. Curran, J. Fujii, C. R. German, D. Gomez-Ibanez, A. F. Govindarajan, J. C. Howland, J. K. Llopiz, P. H. Wiebe, B. W. Hobson, K. Katija, M. Risi, B. H. Robison, C. J. Wilkinson, S. M. Rock, and J. A. Breier. Mesobot: An autonomous underwater vehicle for tracking and sampling midwater targets. *IEEE AUV*, 2018.

[46] D. R. Yoerger, A. F. Govindarajan, J. C. Howland, J. K. Llopiz, P. H. Wiebe, M. Curran, J. Fujii, D. Gomez-Ibanez, K. Katija, B. H. Robison, B. W. Hobson, M. Risi, and S. M. Rock. *Mesobot*: A hybrid underwater robot for multidisciplinary investigation of the Ocean Twilight Zone. *Science: Robotics (forthcoming)*, 2020.

[47] R. Zhang, P. Isola, and A. A. Efros. Colorful image colorization. In *ECCV*, 2016.

[48] Y. Zhang, B. Kieft, B. W. Hobson, J. P. Ryan, B. Barone, C. M. Preston, B. Roman, B. Raanan, R. Marin III, T. C. O'Reilly, C. A. Rueda, D. Pargett, K. M. Yamahara, S. Poulos, A. Romano, G. Foreman, H. Ramm, S. T. Wilson, E. F. DeLong, D. M. Karl, J. M. Birch, J. G. Bellingham, and C. A. Scholin. Autonomous tracking and sampling of the deep chlorophyll maximum layer in an open-ocean eddy by a long-range autonomous underwater vehicle. *IEEE Journal of Oceanic Engineering*, 45(4):1308–1321, 2020.