Robotic Detection of Marine Litter Using Deep Visual Detection Models

Michael Fulton[1], Jungseok Hong[2], Md Jahidul Islam[3], Junaed Sattar[4]

Abstract— **Trash deposits in aquatic environments have a destructive effect on marine ecosystems and pose a long-term economic and environmental threat. Autonomous underwater vehicles (AUVs) could very well contribute to the solution of this problem by finding and eventually removing trash. This paper evaluates a number of deep-learning algorithms performing the task of visually detecting trash in realistic underwater environments, with the eventual goal of exploration, mapping, and extraction of such debris by using AUVs. A large and publicly-available dataset of actual debris in open-water locations is annotated for training a number of convolutional neural network architectures for object detection. The trained networks are then evaluated on a set of images from other portions of that dataset, providing insight into approaches for developing the detection capabilities of an AUV for underwater trash removal. In addition, the evaluation is performed on three different platforms of varying processing power, which serves to assess these algorithms' fitness for real-time applications.**

(a)   (b)

Fig. 1: Examples of plastic and other refuse material in various marine environments. (a) A plastic bottle lying on the sea floor off the coast of Barbados; image collected by the authors, January 2018. (b) A beverage can on the sea bed off the coast of Japan, taken from the J-EDI dataset [3].

I. INTRODUCTION

Marine debris poses a growing threat to the health of our planet. Beginning as discarded fishing gear, improperly recycled packaging, or simply discarded plastic such as grocery bags and soda bottles [1], marine debris makes its way into the ocean by a number of means and remains there [2]. There is virtually no place on earth that is unpolluted by marine debris, which kills and injures aquatic life, chokes ecosystems, and contaminates the water. Recycling and other efforts to keep debris out of the ocean have had a limited impact on the rate of pollution, but the vast amount of trash already in the ocean must somehow be removed. Despite the importance of this problem, there are few large-scale efforts attempting to combat it, due in part to the manpower required. We propose that a key element of an effective strategy for removing debris from marine environments is the use of autonomous underwater vehicles (AUVs) for trash detection and removal.

In this paper, we examine the problem of detecting debris, particularly plastic debris, in an underwater environment, the first of a set of capabilities needed for such AUVs. We consider a number of deep learning-based visual object detection algorithms, build a dataset to train and evaluate them on, and compare their performance in this task by several metrics. Our core question is simply this: is deep-learning based visual detection of underwater trash plausible in real-time, and how do current methods perform in this area?

The authors are with the Department of Computer Science & Engineering, University of Minnesota, 200 Union St SE, Minneapolis, MN, 55455, USA {[1]fulto081, [2]jungseok, [3]islam034, [4]junaed} at umn.edu.

Detection of marine debris through purely visual means is a difficult problem, to say the least. As with many visual object detection problems, small changes in the environment can cause massive changes in the appearance of an object, and nowhere is that truer than with underwater environments. Not only do light changes affect shallower water, but the changing turbidity of the water may make the objects difficult or entirely impossible to detect [4], [5]. Additionally, marine debris is rarely in pristine condition and degrades over time, so our detectors must be capable of recognizing different types of trash in any condition. Lastly, this problem is difficult to solve simply because of the enormous variety of objects that are considered marine debris. As a starting point for the vast problem of detecting all marine litter, we focus on detecting plastic, one of the most prevalent and damaging types of littler in the ocean today. Even in the limited group of plastic trash items, the variety is staggering. Figure 1 shows two examples of actual marine refuse which have completely different appearances. The plastic bottle in Figure 1a is only one example of the many thousands of different styles of plastic bottles that may be found at the bottom of the ocean, to say nothing of plastic bags, containers, and other objects. Plastic is also particularly destructive to the environment, with plastic grocery bags and packaging frequently taking the lives of marine animals which attempt to eat them.

To be useful for the goal of removing those plastics and other trash, object detection algorithms must be able to run in near real-time on robotic platforms. To gauge their readiness for such a deployment, we test all of our networks and models on three different devices, approximating the

capabilities of an offline data processing machine, a high-powered robotic platform, and a low-powered platform.

In this paper, we make the following contributions.

- We evaluate the accuracy and performance of four state-of-the-art object detection algorithms for the problem of trash detection underwater.
- We evaluate the use of different training techniques on one particular network, YOLOv2.
- We produce a unique dataset of plastic marine debris for training deep visual detection models for this task.
- We plan to provide this dataset to the public for continued use in research and education, pending completion of some organization, data augmentation, and additional labeling.

II. Related Work

Underwater autonomous robotics has been growing rapidly in significance and number of applications. Some research has focused on developing multi-robot systems for exploration of natural marine environments, both on the surface and underwater. Learning-based detection of underwater biological events for marine surveillance has also been explored [6]. Additionally, underwater robots have been used in multi-robot systems for environmental surveillance (*e.g.*, [7]), environmental mapping (*e.g.*, [8]), marine robotics for navigation and localization (*e.g.*, [9], [10], [11], [12], [13]), and more. There have been a number of efforts to identify the location of trash and marine litter and analyze their dispersal patterns in the open ocean, particularly in the presence of eddies and other strong ocean currents. Mace looked into at-sea detection of marine debris [14] where he discussed possible applications of strategies and technologies. He points out that one of the major challenges is the debris being small, partially submerged, and buried in the sea bed. His focus is thus narrowed down to the floating debris on water parcel boundaries and eddy lines. Howell et al. [15] make a similar assessment, with their work investigating marine debris, particularly derelict fishing gear, in the North Pacific Ocean. Researchers have looked at the removal of debris from the ocean surface after the tsunami off the Japanese coast in 2011 [16]. The work by Ge et al. [17] uses LIDAR to find and map trash on beaches. Autonomous trash detection and pickup for terrestrial environments have also been investigated; for example, by Kulkarni et al. [18]. While this particular work is applied for indoor trash and uses ultrasonic sensors, a vision-based system can also be imagined.

Recent work by Valdenegro-Toro [19] has looked into using forward-looking sonar (FLS) imagery to detect underwater debris by training a deep convolutional neural network (CNN) and has been demonstrated to work with approximately 80 *percent* accuracy. However, this work uses an in-house dataset constructed by placing objects that are commonly found with marine litter into a water tank and taking FLS captures in the tank. The evaluation was also performed on water tank data. This work demonstrates the applicability of CNNs and other deep models to the detection

of small marine debris, but it is unknown whether it will be applicable to natural marine environments.

Some debris detection processes use remotely operated vehicles underwater (ROVs) to record sightings of debris and manually operate a gripper arm to remove items of interest. The FRED (Floating Robot for Eliminating Debris) vehicles have been proposed by Clear Blue Sea [20], a non-profit organization for environmental protection and cleanup; however, the FRED platform is not an UAV. Another non-profit organization, the Rozalia project, has used underwater ROVs equipped with multibeam and side-scan sonars to locate trash in ocean environments [21].

To enable visual or sensory detection of underwater trash using a deep-learned appearance model, a large annotated dataset of underwater debris is needed. For capturing the varied appearances across wide geographical regions, this dataset needs to include data collected from a large spread of diverse underwater environments. Fortunately, some of these datasets exist, although most are not annotated for deep learning purposes. The Monterey Bay Aquarium Research Institute (MBARI) has collected a dataset over 22 years to survey trash littered across the sea bed off the western seaboard of the United States of America [22], specifically plastic and metal inside and around the undersea Monterey Canyon, which serves to trap and transport the debris in the deep oceans. Another such example is the work of the Global Oceanographic Data Center, part of the Japan Agency for Marine Earth Science and Technology (JAMSTEC). JAMSTEC has made a dataset of deep-sea debris available online as part of the larger J-EDI (JAMSTEC E-Library of Deep-sea Images) dataset [3]. This dataset has images dating back to 1982 and provides type-specific debris data in the form of short video clips. The work presented in this paper has benefited from annotating this data to create deep learning-based models.

III. Data Source and Training Set Construction

This work evaluates four deep learning architectures for underwater trash detection. To evaluate these four networks, we construct a first of its kind dataset for training and evaluation, define our data model, and annotate images for training.

A. Dataset Construction

The dataset for this work was sourced from the J-EDI dataset of marine debris which is described in detail in the previous section. The videos that comprise that dataset vary greatly in quality, depth, objects in scenes, and the cameras used. They contain images of many different types of marine debris, captured from real-world environments, giving us a variety of objects in different states of decay, occlusion, and overgrowth. Additionally, the clarity of the water and quality of the light vary significantly from video to video. This allows us to create a dataset for training which closely conforms to real-world conditions, unlike previous contributions, which mostly rely on internally generated datasets.

Our training data was drawn from videos labeled as containing debris, between the years of 2000 and 2017. From that portion of data, we further selected all videos which appeared to contain some kind of plastic. This was done in part to reduce the problem to a manageable size for our purposes, but also because plastic is an important type of marine debris [23]. At this point, every video was sampled at a rate of three frames *per* second to produce images which could be annotated to prepare them for use in learning models. This sampling produced over $240,000$ frames, which were searched manually for the good examples of plastic marine debris and then annotated. The annotating process was completed by a number of volunteers, who used the freely available LabelImg [24] tool. The final training dataset used in this work was composed of $5,720$ images, with dimensions of 480x320. More images have been labeled since, and a fully labeled dataset will be publicly released in the near future.

B. Data Model

Our trash detection data model has three classes, defined as follows:

- **Plastic:** Marine debris, all plastic materials.
- **ROV:** All man-made objects(*i.e.*, ROV, permanent sensors, etc), intentionally placed in the environment.
- **Bio:** All natural biological material, including fish, plants, and biological detritus.

The model is designed to find all plastic debris, separating debris from biological entities and intentionally placed man-made objects. These two distinctions are important, as accidental removal of biological entities such as plants or animals could damage the very ecosystems we hope to protect, and the destruction of intentionally placed parts such as a sensor array could set back cleanup efforts significantly.

Other versions of this data model were tested: one including a label for timestamp text on the screen (rejected due to lower accuracy) and models with multiple classes for different plastic objects (rejected for lack of data on some classes and lower detection performance overall). Both versions achieved lower accuracy than this model on all networks. Examples of labeled classes in images from this data model can be seen in Figure 2.

C. Evaluation Data

To evaluate the models, test images which contained examples of every class in our model were selected for a test set. However, these images were drawn from videos of objects which had not been used to train the network, so there is no overlap between the training and test set. Different types of plastic objects were selected, such as grocery bags, plastic bottles, etc. At least three different videos and a minimum of 20 images per class were collected for each of these types of objects. In the end, 820 such images were collected and annotated for testing. These images contained examples for each class, in a variety of environments, which were intentionally selected to be challenging to the debris

Fig. 2: An example of a labeled training image showing two classes: the 'plastic' class in blue, and the 'ROV' class in yellow.

detectors, so as to provide a realistic evaluation of how these detectors would perform in field conditions.

IV. NETWORK ARCHITECTURES

The four network architectures selected for this project were chosen from the most popular and successful object detection networks in use today. Each one has its own benefits and drawbacks, with varying levels of accuracy and runtime speeds.

A. YOLOv2

You Only Look Once (YOLO) v2 [25] is the improved version of YOLO, an earlier network by the same authors. Although YOLO is much faster than Fast R-CNN, Faster R-CNN, and other models, it has notable localization errors and low recall in comparison with other state-of-the-art object detection algorithms. With several techniques such as batch normalization, a higher resolution for input images, and a change in the way bounding boxes are proposed to use anchor boxes, YOLOv2 improves upon the accuracy of its predecessor. In order to process faster, YOLOv2 uses a custom network called Darknet-19 rather than VGG-16, which is used by many object detection algorithms. Darknet-19 requires only about one-sixth of the floating point operations of VGG-16 for a single pass of an input image.

B. Tiny-YOLO

Tiny YOLO [26] was introduced along with YOLO. Tiny-YOLO is based on the neural network that has a smaller number of convolutional layers and filters inside the layers compared to the network used by YOLO while sharing the same parameters with YOLO for training and testing. In this way, tiny-YOLO runs at even faster speeds than YOLO while having similar mean Average Precision (mAP) values. Tiny-YOLO is especially useful to implement object detection for applications where computational power is limited (*e.g.*, for energy consumption reasons) such as field robotics.

C. Faster RCNN with Inception v2

Faster RCNN [27] is an improvement on R-CNN [28] that introduces a Region Proposal Network (RPN) to make the whole object detection network end-to-end trainable. The RPN uses the last convolutional feature map to produce region proposals, which is then fed to the fully connected layers for the final detection. The original implementation uses VGG-16 [29] for feature extraction; we use the Inception v2 [30] as feature extractor instead because it is known to produce better object detection performance in standard datasets [31].

D. Single Shot MultiBox Detector (SSD) with MobileNet v2

Single Shot MultiBox Detector (SSD) [32] is another object detection model that performs object localization and classification in a single forward pass of the network. The basic architectural difference of SSD with YOLO is that it introduces additional convolutional layers to the end of a base network which results in improved performance. In our implementation, we use MobileNet v2 [33] as the base network.

V. TRAINING

All four networks were trained according to the methods suggested by the community around the networks [31], [34], using on a Linux machine, using four NVIDIA^{TM} GTX 1080s. In general, this involved fine-tuning the networks with some pretrained weights which had been trained on the COCO dataset [35], a large-scale detection dataset. There were 5,720 images in our training set for each network, resized to 416x416 in the case of YOLOv2 and Tiny-YOLO. For RCNN, SSD, and Tiny-YOLO, we simply fine-tuned each network for a few thousand iterations.

However, in the case of YOLO, the network was not only fine-tuned, but also trained using a transfer learning. Transfer learning [36] is a technique used to improve learning in a new task by transferring knowledge from related tasks. The goal of transfer learning is to expedite the learning process by utilizing prior knowledge from one domain to new related domains. There are several benefits of applying transfer learning to this problem. First, the chance of the network properly converging for a new task may be better. Instead of training from a base, it uses the transferred knowledge to set initial training parameters. This may prevent a model from becoming trapped in local minima. Second, learning time to find parameters of a network may be reduced compared to learning from a base. Lastly, the final inference may have better performance than learning without transferred knowledge. In this study, we froze all but the last several layers so that only the last few layers had their weights updated during the training process. In this way, we attempt to capitalize on the fact that earlier layers' pretrained weights already encode for basic image features, and simply train the last few layers of YOLOv2. We report the results of these transfer-learned versions separately from the RCNN, SSD, and Tiny-YOLO results, to explore the possibility of using transfer learning to overcome deficiencies in a dataset.

VI. EVALUATION

A. Metrics

The four different networks used for the marine debris detector were evaluated using the two standard performance metrics below:

- mAP (Mean Average Precision) is the average of precision at different recall values. Precision is the ratio $precision = \frac{TP}{TP+FP}$, and recall is the ratio $recall = \frac{TP}{TP+FN}$, where TP, FP, and FN stand for True Positive, False Positive, and False Negative.
- IoU (Intersection Over Union) is a measure of how well predicted bounding boxes fit the location of an object, defined as $IoU = \frac{area\ of\ intesection}{area\ of\ union}$, where the intersection and union referred to are the intersection and union of the true and predicted bounding boxes.

These two metrics concisely describe the accuracy and quality of object detections.

B. Hardware

Additionally, runtime performance metrics are provided on a GPU (NVIDIA^{TM} 1080), embedded GPU (NVIDIA^{TM} Jetson TX2), and CPU(an Intel^{TM} i3-6100U) in terms of the number of frames which can be processed per second (FPS). The hardware used for evaluation is only relevant to the runtime. Performance metrics, such as mAP, IoU, and recall, can be calculated identically on any device. The devices were selected to provide an indication of how these models could be expected to perform in an offline manner (GPU), in real-time on a computationally powerful robotic platform (embedded GPU), and in a lower-power robotic platform (CPU).

VII. RESULTS

The results below were obtained on the test dataset described in subsection III-C, which was designed to be challenging to the models and to provide a true-to-life representation of how a marine debris detector would have to operate. The videos from which the test data is sourced are mostly from 10 years or more before the time of videos in the training set; they differ from each other and the training set in location, depth, and the camera used to capture the scene. Because of these factors, the test set provides a good indicator of how these detectors would perform across different platforms over the course of many years and through many environmental changes. This does degrade the performance of the detector from what could have been achieved if data with similar appearance was chosen, but it is a better evaluation of what the detector's performance would be in the field. Along with evaluating the data on our test set, we also processed three additional videos, with each network, which can be seen in Figure 3 as well as in the video submitted with this paper. The third video, which is shown in Figures 3i - 3l, is a viral video of ocean trash recorded in Bali [37]. Being from an entirely different visual environment than our training and test data, it challenges all of our detectors greatly, but some promising detections can be seen over the course of the video.

Network	mAP	Avg. IoU	plastic AP	bio AP	rov AP
YOLOv2	47.9	54.7	**82.3**	9.5	52.1
Tiny-YOLO	31.6	49.8	**70.3**	4.2	20.5
Faster R-CNN	81.0	60.6	**83.3**	73.2	71.3
SSD	67.4	53.0	**69.8**	6.2	55.9

TABLE I: Detection metrics in mAP, IoU, and AP.

Network	1080	TX2	CPU
YOLOv2	74	6.2	0.11
Tiny-YOLO	**205**	**20.5**	0.52
Faster R-CNN	18.75	5.66	0.97
SSD	25.2	11.25	**3.19**

TABLE II: Performance metrics in frames per second.

YOLO Training	mAP	Avg. IoU	plastic AP	bio AP	rov AP
Fine Tuned	47.9	54.7	**82.3**	9.5	52.1
Last 4 Layers	33.9	45.4	**71.3**	13.6	17.0
Last 3 Layers	39.5	34.1	**74.6**	19.9	23.9

TABLE III: Detection metrics for different training methods for YOLO.

A. Quantitative Results

The results shown in Table III are encouraging. The networks tested have fairly good average precision for the plastic class (similar to the accuracy of the FLS imagery based approach in [19]). They also display some known traits of the network architectures used. Overall, YOLOv2 and Tiny-YOLO have lower mAP when compared to Faster R-CNN and SSD. Conversely, Faster R-CNN and SSD have higher processing times, as seen in Table III. These traits are well known, but it is important to note that their performance remain consistent in this application. This trade-off between mAP and FPS does not affect IoU, however. All four network architectures have similar IoU values, meaning that none are the clear victor in terms of how accurate their bounding boxes are.

In terms of which method would be the ideal for underwater trash detection, Faster R-CNN is the obvious choice purely from a standpoint of accuracy, but falters in terms of inference time. YOLOV2 strikes a good balance of accuracy and speed, while SSD provides the best inference times on CPU. If performance is the primary consideration, however, Tiny-YOLO outpaces all other algorithms significantly on the TX2, the most realistic hardware for a modern AUV.

We can also see the results of including classes with too few examples, particularly in the case of the bio class, which has many fewer examples in the training set, despite being at least as varied as the plastic trash in the underwater environment. The fact that classes with few training images relative to their complexity will perform well is not surprising in any way, but highlights some of the difficulties inherent in creating a dataset for trash detection: some objects may only be encountered a handful of times but should still count as trash, or be marked as biological and therefore not be removed.

It is worth noting that in the case of the bio class, the versions of YOLO trained using transfer learning are significantly more accurate. We believe this to be related to the types of objects that the pre-trained weights were trained to detect and the preponderance of plastic objects in the dataset. Simply put, by not updating those earlier layers, we avoid skewing the basic image features towards plastic objects, reducing plastic detection accuracy slightly which increasing that of the bio class.

B. Qualitative Results

The most substantial result from this evaluation is the answer to the question we set out to investigate: can we use deep-learning based visual object detection methods to detect marine debris in real-time? The answer, borne out by our detection accuracy and performance results is, yes, plausibly. There are some difficulties in constructing a representative dataset, but overall our detection results lead us to believe that detectors similar to these could be used to find trash underwater with high enough accuracy to be useful. Moreover, the observed performance, especially on the NVIDIA™ Jetson TX2, is very encouraging for the prospect of a real-time application. The Jetson is small enough to conceivably be used in an AUV, and although it increases the electrical and thermal load of the system, the authors believe the increase to be manageable, with the Jeston only requiring 15 watts [38] and producing minimal heat. This, combined with its performance in tasks such as these, makes the Jetson ideal for mobile robotics, even for our specific use case. With a Jetson installed, any AUV should be able to achieve real-time trash detections with reasonable accuracy. The authors' Aqua AUV [39] will be fitted with a TX2 in the near future for this very purpose.

VIII. FUTURE WORK

In the future, we hope to expand upon this work by testing similar algorithms on a dataset collected from our own observations of marine debris in real-world environments. Additionally, we plan to explore other approaches to this task. The problem of detecting trash in underwater environments causes some inherent difficulties in deep-learning-based solutions, as it is difficult, verging on impossible, to collect a dataset with sufficient coverage. The use of transfer learning, progressive neural networks, data augmentation, synthetic training images, or other methods which help to overcome data limitations are areas we intend to explore. In a similar vein, we hope to explore generative methods for data synthesis to help increase the amount of available trash data for training.

Of course, our primary future direction is the general goal of this work: to move towards an AUV capable of clearing marine environments of trash, with little to no human help. This will involve the development of navigation, exploration, and manipulation strategies to better find all of the trash in the water and remove it for good. We are very interested in the possibility of multi-robot collaboration between multi-modal robots (for instance aerial and aquatic) in solving this problem.
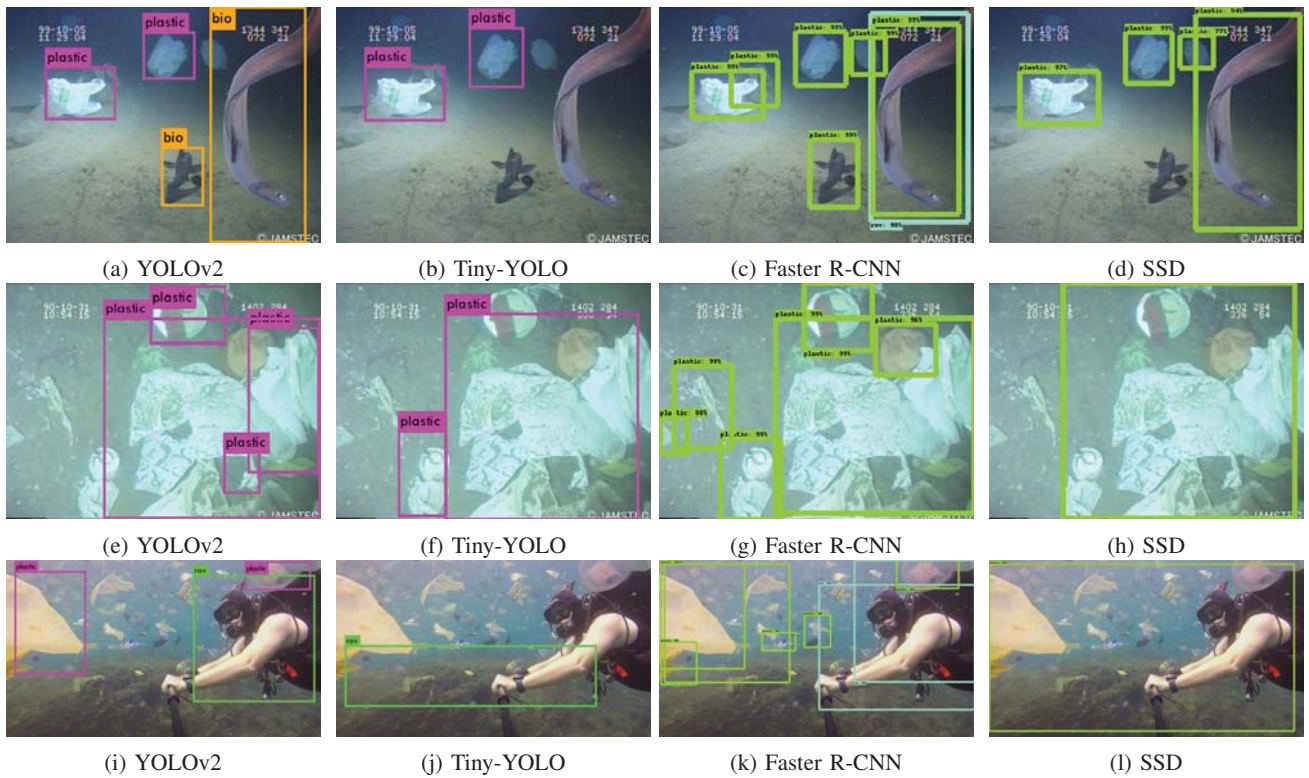
Fig. 3: Example detection results. In images 3a-3d it can be seen that while all networks detect some plastic objects, only YOLOv2 (3a) correctly identifies the fish in the scene as bio and Faster-RCNN (3c) detects more individual plastic items than any other network. In images 3e-3h, the same evidence of Faster-RCNN's (3g) ability to detect more individual objects can be seen. Images 3i-3l are from [37], and display how the detectors work on data drawn from a visually different source. YOLOv2 (3i) and Faster-RCNN (3k) both successfully find plastic and ROV objects, but no detectors perform sufficiently.

IX. CONCLUSION

In this paper, we applied and evaluated four deep-learning-based object detectors to the problem of finding marine debris, particularly plastic, in an attempt to answer the question "Can visual learning object detectors be used to detect trash underwater in real-time?" We built a dataset from publicly available data and devised a data model encompassing plastic trash, biological matter and intentional man-made objects. Following this, we trained YOLOv2, Tiny-YOLO, Faster R-CNN, and SSD using standard fine-turning procedures, then evaluated them on a test dataset composed of challenging objects that had not been seen during training. We also trained YOLOv2 using transfer learning techniques to train only the last few layers, once with the last four un.rozen and once with the last three unfrozenf We performed evaluations on three different devices, acquiring information about the mAP and IoU for each model and network, as well as the runtime for all configurations. Based on these results, we feel that trash detection using visual deep learning models is plausible in real-time. In the future, these techniques should be used as the first part of an AUV designed to remove trash from aquatic environments which will be a critical tool in our fight against the pollution of our oceans.

AUTHOR CONTRIBUTION

M.F and J.H. conceived of the presented idea, developed the analysis plan, selected the networks, and built the dataset. M.F. developed automated dataset processing and managed training and inference for YOLO and Tiny-YOLO. J.H. managed transfer learning training and analysis, and developed the validation dataset. M.J.I. managed training and inference for Faster-RCNN and SSD. J.S. advised, encouraged, and directed all research related to this project. All authors discussed the project and contributed to this final manuscript.

REFERENCES

[1] United States Environmental Protection Agency, *Sources of Aquatic Trash*, https://www.epa.gov/trash-free-waters/sources-aquatic-trash. Accessed 02-10-2018.

[2] National Oceanic and Atmospheric Administration, *What is marine debris?*, https://oceanservice.noaa.gov/facts/marinedebris.html. Accessed 02-10-2018.

[3] Japan Agency for Marine Earth Science and Technology, *Deep-sea Debris Database*, http://www.godac.jamstec.go.jp/catalog/dsdebris/e/index.html. Accessed 02-10-2018.

[4] H. Lu, Y. Li, X. Xu, L. He, Y. Li, D. Dansereau, and S. Serikawa, "Underwater image descattering and quality assessment," in *2016 IEEE International Conference on Image Processing (ICIP)*, Sept 2016, pp. 1998–2002.

[5] C. Fabbri, M. J. Islam, and J. Sattar, "Enhancing underwater imagery using generative adversarial networks," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 7159–7165.

[6] M. Bernstein, R. Graham, D. Cline, J. M. Dolan, and K. Rajan, "Learning-based event response for marine robotics," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov 2013, pp. 3362–3367.

[7] Y. Girdhar, A. Xu, B. B. Dey, M. Meghjani, F. Shkurti, I. Rekleitis, and G. Dudek, "MARE: Marine Autonomous Robotic Explorer," in *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '11)*, San Francisco, USA, September 2011, pp. 5048–5053.

[8] A. Kim and R. M. Eustice, "Active Visual SLAM for Robotic Area Coverage: Theory and Experiment," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 457–475, 2015.

[9] X. Yuan, J.-F. Martínez-Ortega, J. A. S. Fernández, and M. Eckert, "AEKF-SLAM: A New Algorithm for Robotic Underwater Navigation," *Sensors*, vol. 17, no. 5, 2017. [Online]. Available: http://www.mdpi.com/1424-8220/17/5/1174

[10] J. J. Leonard and A. Bahr, "Autonomous underwater vehicle navigation," in *Springer Handbook of Ocean Engineering*. Springer, 2016, pp. 341–358.

[11] L. Stutters, H. Liu, C. Tiltman, and D. J. Brown, "Navigation technologies for autonomous underwater vehicles," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 4, pp. 581–589, 2008.

[12] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV navigation and localization: A review," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, pp. 131–149, 2014.

[13] F. Hidalgo and T. Bräunl, "Review of underwater SLAM techniques," in *Automation, Robotics and Applications (ICARA), 2015 6th International Conference on*. IEEE, 2015, pp. 306–311.

[14] T. H. Mace, "At-sea detection of marine debris: Overview of technologies, processes, issues, and options," *Marine Pollution Bulletin*, vol. 65, no. 1, pp. 23–27, Jan 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0025326X11004668

[15] E. A. Howell, S. J. Bograd, C. Morishige, M. P. Seki, and J. J. Polovina, "On North Pacific circulation and associated marine debris concentration," *Marine Pollution Bulletin*, vol. 65, no. 1, pp. 16–22, Jan. 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0025326X11002372

[16] National Oceanic and Atmospheric Administration, "Detecting Japan Tsunami Marine Debris at Sea: A Synthesis of Efforts and Lessons-Learned | OR&R's Marine Debris Program," Jan 2015. [Online]. Available: https://marinedebris.noaa.gov/japan-tsunami-marine-debris/detecting-japan-tsunami-marine-debris-sea-synthesis-efforts-and-lessons

[17] Z. Ge, H. Shi, X. Mei, Z. Dai, and D. Li, "Semi-automatic recognition of marine debris on beaches," *Scientific Reports*, vol. 6, p. 25759, May 2016. [Online]. Available: http://dx.doi.org/10.1038/srep25759

[18] S. Kulkarni and S. Junghare, "Robot based indoor autonomous trash detection algorithm using ultrasonic sensors," in *2013 International Conference on Control, Automation, Robotics and Embedded Systems (CARE)*, Dec 2013, pp. 1–5.

[19] M. Valdenegro-Toro, "Submerged marine debris detection with autonomous underwater vehicles," in *2016 International Conference on Robotics and Automation for Humanitarian Applications (RAHA)*, Dec 2016, pp. 1–7.

[20] Non-Profit Organization, *Clear Blue Sea*, http://clearbluesea.org. Accessed 02-10-2018.

[21] R. Z. Miller, *Underwater Robots Clean Up Marine Debris on the Seafloor*, The Rozalia Project, https://www.sea-technology.com/features/2012/1212/underwater_robots.php. Accessed 02-10-2018.

[22] K. Schlining, S. v. Thun, L. Kuhnz, B. Schlining, L. Lundsten, N. J. Stout, L. Chaney, and J. Connor, "Debris in the deep: Using a 22-year video annotation database to survey marine litter in Monterey Canyon, central California, USA," *Deep Sea Research Part I: Oceanographic Research Papers*, vol. 79, pp. 96 – 105, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0967063713001039

[23] J. G. Derraik, "The pollution of the marine environment by plastic debris: a review," *Marine pollution bulletin*, vol. 44, no. 9, pp. 842–852, 2002.

[24] "Github - tzutalin/labelimg: Labelimg is a graphical image annotation tool and label object bounding boxes in images," https://github.com/tzutalin/labelImg, (Accessed on 02/25/2019).

[25] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," *arXiv preprint arXiv:1612.08242*, 2016.

[26] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 779–788.

[27] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems (NIPS)*, 2015.

[28] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '14. Washington, DC, USA: IEEE Computer Society, 2014, pp. 580–587. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2014.81

[29] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[30] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2818–2826.

[31] Tensorflow, "Tensorflow object detection zoo," https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md, 2017, accessed: 2-20-2018.

[32] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[33] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation," *arXiv preprint arXiv:1801.04381*, 2018.

[34] AlexeyAB, "Yolo: How to train (to detect your custom objects)," https://github.com/AlexeyAB/darknet#how-to-train-to-detect-your-custom-objects), 2018, accessed: 9-14-2018.

[35] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[36] L. Torrey and J. Shavlik, "Transfer learning," 2009.

[37] R. Horner, "The ocean currents brought us in a lovely gift today... - YouTube." [Online]. Available: https://www.youtube.com/watch?v=AWgfOND2y68

[38] D. Franklin, *NVIDIA Jetson TX2 Delivers Twice the Intelligence to the Edge*, https://devblogs.nvidia.com/jetson-tx2-delivers-twice-intelligence-edge/. Accessed 03-01-2018.

[39] J. Sattar, G. Dudek, O. Chiu, I. Rekleitis, A. Mills, P. Giguère, N. Plamondon, C. Prahacs, Y. Girdhar, M. Nahon, and J.-P. Lobos, "Enabling autonomous capabilities in underwater robotics," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Nice, France, September 2008, pp. 3628–3634.