Real-Time Background-Agnostic Fish Localization in Underwater Videos towards Autonomous Species Monitoring

Anuj Shrivatsav Srikanth[a,1], Saicharan Thirandas[a,1], Dhanush Adithya Balamuguran[a,1], Anurag Daga[a,1],
Robert Vincent[b,2], Taskin Padir[a,3], and Dipanjan Saha[a,4]

Abstract— **This paper investigates a novel machine learning framework for autonomous, real-time fish localization in underwater videos with diverse backgrounds. The framework consists of three different algorithms from the family of deep learning and computer vision. Each of them is a good solution to one or more specific needs; however, each algorithm has its own limitations. Combining these methods using ensemble learning is a way to accomplish background-agnostic fish localization in real-time. A specific combination called weighted voting learns an optimal set of weights, such that the highest weight goes to the algorithm with the highest prediction accuracy. Results presented for two underwater datasets with significantly varying background and illumination demonstrate that weighted voting can produce consistent localization irrespective of the environment.**

I. INTRODUCTION

Accurate assessment of fish population and spawning abundance is crucial for aquaculture and long-term ecosystem studies. This assessment provides fishery management staff and marine scientists with valuable data necessary for sustainability, ecosystem monitoring, understanding patterns of fish abundance and behavior in underwater habitats including near aquaculture farm habitats. A motivating example is River Herring (RH) in the northeastern United States [1], [2]. While the adult Herrings grow in seawater, they migrate to freshwater for spawning every spring. In turn, the juvenile Herrings migrate to seawater every summer and fall. This yearly migration pattern makes RH an essential component of both freshwater and marine ecosystems, as well as coastal fisheries. The need for continuous monitoring was established more than ten years ago, when a decline in population to less than 3% of the historical peaks led to the closure of many RH fisheries.

RH populations (alewife, *Alosa pseudoharengus*; and blueback, *Alosa aestivalis*) are often counted visually, followed by statistical corrections [3], [4]. Visual counting is labor-intensive and impractical for thousands of hours of monitoring video. Moreover, volunteers often count sporadically only during daylight hours and potentially miss a large portion of the fishery. In addition, multiple species might share a habitat, and the end-user(s) might need the count of only a subset of all the species, swimming in a given direction (e.g., upstream). Real-time, automated species identification and counting is a complex problem. An underwater environment has multiple sources of uncertainty [5], [6], such as water turbidity, lighting conditions (e.g., day vs. night), fish overlap during periods of high passage rates, potential for double counting (i.e., tracking multiple occurrences of the same fish entering the view frame), varying size and orientation of fish, and fish partially hidden from view. A previous work [7] attempted to develop an automated solution by leveraging the change in the electrical conductivity of water during fish passage. Although the presence of fish could be detected, it was difficult to establish a quantitative relationship between the change in water conductivity and the count of fish.

A fully automated underwater video monitoring framework may be constructed in three stages. The first stage is the localization of every fish in a video frame. The second stage is the species identification of each localized fish. The third and the final stage is counting the species of interest and ensuring consistency of counts across video frames. The first stage of fish localization is pivotal to the next two. Recently, autonomous perception has been studied in many works in computer vision and deep learning [8], [9], [10]. While deep learning has great potential to account for uncertainties, very few works applied the same towards autonomous localization of fish in underwater videos. Underwater data is noisier than terrestrial, and the necessary enhancement is difficult to achieve in real-time. A deep learning-based localization algorithm itself requires high processing time. Moreover, in contrast to air or ground, underwater background might vary significantly from one environment to another. This variability makes it challenging for a localization algorithm to quickly adapt to an unseen background. In addition, very few annotated underwater fish video datasets are publicly available. Manual labeling is expensive, especially so for high passage rates.

This work seeks to overcome these key challenges and achieve real-time, background-agnostic fish localization in underwater videos. The objective is to develop a framework which - (a) produces reasonably accurate fish masks to be processed further for species-specific counting, and (b) adapts to diverse backgrounds with minimal manual labeling. This is achieved by combining three different localization algorithms based on computer vision and deep learning. One of them works in real-time and *subtracts*

[a] Institute for Experiential Robotics, Northeastern University, Boston, MA 02115, USA

[b] Massachusetts Institute of Technology, Department of Mechanical Engineering, Sea Grant, Cambridge, MA 02139, USA

[1] {srikanth.anu, thirandas.sa, balamurugan.d, daga.an} @northeastern.edu

[2] rvincent@mit.edu

[3] t.padir@northeastern.edu

[4] d.saha@northeastern.edu

the static background from all moving objects. The other two algorithms work in near real-time and helps narrow the set of moving objects down to fish. One of them uses optical flow to detect objects with significant motion in a specific direction. The other is trained on fish semantics using thousands of labeled fish images, and it *segments* the fish contours out from the background. The diversity in background as well as fish movement pattern might lead to one of the localization algorithms outperforming the other two. In order to achieve consistent results across backgrounds, two different combinations of the individual localization algorithms are investigated and compared. One combination weighs all of the three algorithms equally. The other requires some manual labeling as the ground truth, and it assigns the highest weight to the algorithm producing results closest to the ground truth.

The rest of the paper is organized as follows. Section II explores the literature and selects the localization algorithms. The implementation of these algorithms is in Section III. Subsequently, Section IV describes the datasets used in this study. Section V discusses the results. Section VI concludes the work.

II. LITERATURE REVIEW & SELECTION OF ALGORITHMS

Historically, many works on localization relied on background subtraction (BGS) as the fundamental technique [11], [12], [13]. BGS creates a model of the background environment and then subtracts this model from the observed frames to identify dynamic elements. The primary goal is to extract meaningful information about the moving entities such as people, vehicles, or other objects of interest, regardless of the background. In order to evaluate BGS algorithms, a number of benchmark datasets were reported in the literature [14]. Among these datasets, the `changedetection.net` (CDnet) [15] features realistic and challenging video sequences that accurately represent uncertainties. The uncertainties include variations in the illumination, dynamic backgrounds, and object occlusions.

An extensive review of the state-of-the-art BGS algorithms can be found in the work of Sobral and Vacavant [16]. In particular, they evaluated and compared a total of 29 algorithms on the Background Models Challenge (BMC) dataset. The comparison metrics were related to the robustness as well as the processor and memory requirements for each method. The top five methods were identified based on this comparison. However, this paper did not evaluate any of the BGS algorithms on the CDnet dataset and instead mentioned the same as future work. In contrast, the work of St-Charles *et. al.* [17] did use the CDnet dataset for evaluation, and their algorithm outperformed all the previously used methods when tested on the CDnet dataset. This algorithm, called the Self-Balanced SENsitivity SEgmenter (SuBSENSE), dynamically adapts its internal parameters as it continuously monitors the model fidelity and local segmentation noise levels. It utilizes spatiotemporal binary features and color information to detect changes in video sequences with high accuracy and efficiency.

Notably, the performance of many BGS algorithms including SuBSENSE often degrade in underwater environments, especially when there is significant motion of other objects in the background. In addition, these algorithms do not account for fish semantics. As a result, they might separate only part of the foreground, more so when the lighting interferes with the fish. These major drawbacks encouraged researchers to look for alternatives, including combining BGS with other techniques. Liu *et. al.* [18] combined BGS with three-frame difference using the logical AND operation and subjected the result to morphology processing for noise removal. This approach reliably detected moving objects in underwater videos with complex scenes and poor lighting. On the other hand, Wu *et. al.* [19] proposed a clustering-based multi-state background representation model to accurately subtract waving background objects and enhance the accuracy of BGS. Although these approaches produced good results for specific datasets, how they can adapt to diverse underwater environments is yet to be investigated. Moreover, none of them considered the semantics of the object of interest.

Braham *et. al.* [20] combined BGS with object semantics and developed a new algorithm called the Semantic Background Subtraction (SBS). This algorithm reduced the error rate by 50% compared to the traditional BGS algorithms. Despite the large increase in accuracy, the aspect of semantic segmentation led to non-real-time computation. A significant improvement over SBS was put forth by Cioppa *et. al.* [21]. Their algorithm, called the Real-Time Semantic Background Subtraction (RT-SBS) combined a real-time BGS algorithm with high-quality semantic information which can be provided at a slower pace, independently for each pixel. This enabled the combination to work in real-time and still perform similar to SBS. However, to this date RT-SBS has shown exemplary results on terrestrial datasets only, and it is yet to be tested in underwater scenarios. A possible reason for this could be the shortage of labeled underwater datasets, which hinders large-scale training and benchmark evaluation of semantic segmentation models. The work of Islam *et. al.* [22] developed an annotated, extensive underwater dataset called the Segmentation of Underwater Imagery (SUIM). SUIM contains over $1,500$ underwater images with pixel annotations for eight object categories: fish (vertebrates), reefs (invertebrates), aquatic plants, wrecks/ruins, human divers, robots, and sea-floor. In addition to this valuable dataset potentially useful for many underwater applications, they also proposed SUIM-Net, a fully convolutional deep neural network for semantic segmentation, trained on SUIM.

While combining BGS with semantic segmentation has its own merits, the latter is achieved by a deep neural network, sensitive to the training data. As a consequence, it may not adapt well to a new environment. A typical underwater environment has a dynamic background with various moving objects, and fish often exhibit agile and swift movements unlike the other objects. Such movement patterns of fish create substantial variations in pixel intensities over consecutive frames. Optical flow [23], [24] is a useful approach to capture these variations. It is also capable of localizing fish moving in a specific direction by setting the pixel intensity gradient positive or negative. Furthermore, marine biologists are often interested in understanding fish behavior, and it is possible to combine optical flow with other approaches for the same purpose [25], [26]. Depending on how the gradient is computed, optical flow algorithms can be classified as global or local. While the global methods are generally more accurate, local methods have better runtime performance and are more suitable to track swift fish movements. Recently, Senst *et. al.* [27] proposed a Robust Local Optical Flow (RLOF) algorithm which accomplished good tracking even in scenarios violating the assumptions of local optical flow. Such scenarios include motion boundaries, changing illuminations, and appearing pixels. In addition, a standard implementation of RLOF is available in the OpenCV library of Python.

In summary, the variability in underwater background, fish movement pattern, passage rate as well as the shortage of labeled datasets make it challenging to achieve adequate localization with a single algorithm. It is worthwhile to use an inexpensive, real-time BGS algorithm (e.g., SuBSENSE) as a baseline, and fine-tune its results from time to time using the more sophisticated semantic segmentation (e.g., SUIM-Net) and optical flow (e.g., RLOF) algorithms. Moreover, the background diversity may be addressed by choosing an optimal combination of these individual algorithms, where the optimal weights differ from one background to another. The details of combining the algorithms will be discussed in Section III.

III. IMPLEMENTATION OF ALGORITHMS

Each of the selected algorithms, viz. SuBSENSE, SUIM-Net, and RLOF, has their own implications in the context of fish localization in underwater videos. SuBSENSE is the fastest of the three. It runs in real-time and isolates all moving objects from the static surroundings. However, the set of moving objects might include fish swimming upstream as well as downstream, water waves, floating leaves, seagrass with waving motions, and so on. SUIM-Net being a deep neural network is the slowest of the three. It cannot run in real-time; however, it is aware of fish semantics, and it can be fine-tuned if needed. RLOF cannot run in real-time either, but it is faster than SUIM-Net. In addition, RLOF does not account for fish semantics, but it can localize the fish moving in a specific direction by analyzing movement patterns.

The current work uses two steps to combine the algorithms. The first step is preprocessing, intended to fine-tune SUIM-Net for improved accuracy. The second step is to merge the individual predictions using Ensemble Learning [28].

A. Fine-Tuning the SUIM-Net

The SUIM-Net architecture [22] is a fully convolutional, residual learning, encoder-decoder model with optional skip connections. The encoder network extracts 256 feature maps from input RGB images. The feature maps are utilized by three sequential decoder layers to generate per-channel binary pixel labels for each object category. The optional skip layers result in real-time inference while achieving competitive segmentation performance.

The preprocessing step consists of making a few modifications to the existing SUIM-Net architecture to improve its performance. The first modification is to keep only one decoder layer instead of three. This reduces the network to a binary segmentation model, and predicts for every pixel whether it is the desired object (e.g., fish) or not. This modification leverages SUIM-Net's existing capacity to capture intricate patterns and nuances specific to fish, increases its precision, and makes it easier to generalize to diverse fish datasets. The second modification is to use the pre-trained weights for the first two encoder layers and train only the deeper layers. This signifies honing in only on the higher level features of the desired object (i.e., fish), and hence better performance. The third modification is to determine an optimal set of hyperparameters using Bayesian optimization [29] and the validation score as a loss function. Upon these modifications, the resulting network was trained with images from one of the datasets. All of these led to a significant performance improvement of the SUIM-Net.

B. Ensemble Learning to Combine Predictions

SuBSENSE, SUIM-Net, and RLOF make their individual predictions of whether a pixel in a video frame is a foreground (i.e., fish) or a background pixel. These predictions may be merged using one of two Ensemble Learning techniques, viz. soft voting (SV) or weighted voting (WV) [28]. Both of the voting methods return a weighted average of the individual predictions. However, SV weighs all predictions equally, whereas WV determines a set of optimal weights w_1, w_2, w_3 by minimizing the following mean square error loss:

$$L = \sum_x \sum_y (w_1 \cdot I_1(x,y) + w_2 \cdot I_2(x,y) + w_3 \cdot I_3(x,y) - I_G(x,y))^2 \tag{1}$$

where $I_1(x,y), I_2(x,y), I_3(x,y)$ are the intensities of the $(x,y)^{th}$ pixel of the individual output images, and $I_G(x,y)$ is the intensity of the $(x,y)^{th}$ pixel of the ground truth image. In other words, the error is simply the difference between the prediction and the ground truth, and thus WV requires some annotated video frames to serve as the ground truth. Intuitively, the optimal weights are such that the highest weight corresponds to the algorithm with the most accurate prediction. A change in the background corresponds to a different set of optimal weights. In general $w_1 \neq w_2 \neq w_3$ for WV. SV is a special case where $w_1 = w_2 = w_3$.

The overall localization framework thus consists of SuBSENSE (an algorithm under BGS), SUIM-Net (a deep neural network performing semantic segmentation), and

RLOF (an algorithm under optical flow), combined using SV or WV (algorithms under Ensemble Learning). A pictorial representation of the framework is shown in Fig. 1.

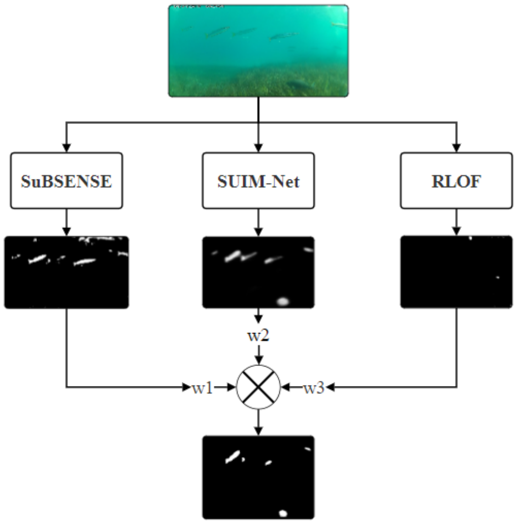


Fig. 1. Individual algorithms and their combination using weighted voting; soft voting is a special case where the weights w_1, w_2, w_3 are equal

IV. DATASETS

The localization framework shown in Fig. 1 is tested on two distinct datasets with significantly different backgrounds.

A. The Seagrass Dataset

Ditria *et. al.* [30] developed a dataset named "Annotated videos of luderick from estuaries in southeast Queensland, Australia". This dataset includes footage from remote underwater video recordings of luderick and Australian bream in seagrass habitats of estuaries in southeast Queensland. This will be referred to as the "Seagrass" dataset for the remainder of the paper. The raw data were collected using submerged action cameras in the Tweed River estuary and Tallebudgera Creek between February and July 2019. Six cameras were deployed each sampling day, capturing a variety of seagrass patches. The dataset comprises 4,281 video frames and 9,429 annotations, with backgrounds varying in complexity due to different camera angles, depths, lighting conditions, and fish positions. Because of readily available annotations, all of the 4,281 frames were used for either one of training, validation, or testing. Sample video frames from the Seagrass dataset are shown in Fig. 2.

B. The IRWA Dataset

The Ipswich River Watershed Association (IRWA) RH monitoring camera is located at the top of the Ipswich Mills Dam near downtown Ipswich, Massachusetts, USA. A Seaviewer 950 dropcam is housed inside of an aluminum and wood "camera box" which also functions to direct the fish in a single path past the camera. The Box, measuring approximately 1 m high, by 1.5 m wide, and 1 m long is connected to the fish ladder at the top of the dam to record RH passing upstream of the dam. A picture of the camera setup is shown in Fig. 3(a)-3(b). The camera uses iSpy motion trigger software and is equipped with infrared lights for night recording. The system is controlled by a laptop computer housed inside of a waterproof pelican case. Power is provided by an adjacent building. Video is recorded onto a 5 TB external hard drive, which is retrieved weekly for transfer, storage, and analysis of video in the lab. The camera records 24/7 during the RH spawning run period, approximately April 1st through June 1st annually. A total of 1,055 video recordings were collected between the years 2015 and 2018, leading to a significantly large dataset for machine learning applications.

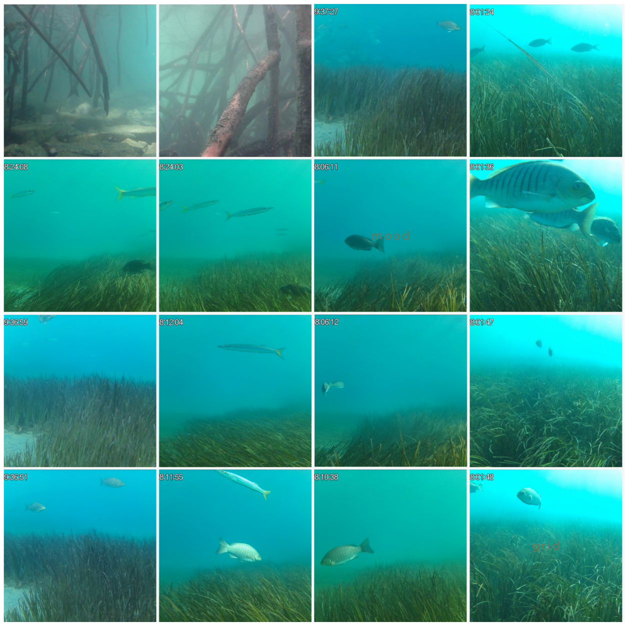
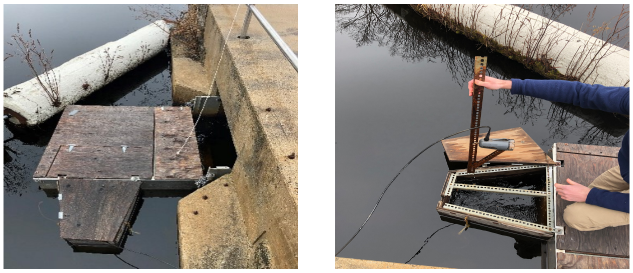


Fig. 2. Sample video frames collected at seagrass habitats in Southeast Queensland, Australia, showing luderick and Australian bream fish passage



(a) camera box

(b) camera installation

Fig. 3. Installation of the Seaviewer 950 Camera at the Ipswich Mills Dam near downtown Ipswich, Massachusetts, USA.

A total of 10 videos from this dataset were selected to implement the localization framework developed in the current work. The selection of the videos ensured sufficient variability in lighting, fish swimming patterns, and fish count. None of the videos came with annotations, so a total of about 1,200 video frames were extracted for the generation of ground truth information via manual labeling. The manual labeling involved drawing precise polygons around each observed fish for each frame using the Roboflow software. Sample video frames from the IRWA dataset are shown in Fig. 4. Figs. 2 and 4 clearly show the distinction between the background and illumination in the two datasets used to test the localization framework.

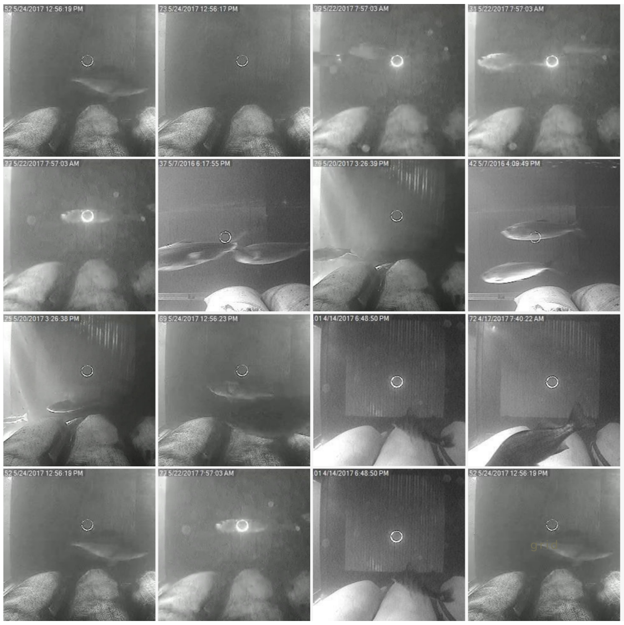


Fig. 4. Sample video frames collected at the Ipswich Mills Dam in Massachusetts, USA, showing River Herring fish passage

V. Results

The end goal is to have the proposed framework localize fish in real-time for diverse underwater backgrounds characterized by the Seagrass and the IRWA datasets. To this end, it is worthwhile to investigate how well each algorithm can accomplish localization individually, and how much improvement occurs when the predictions are combined via either SV or WV. The validation accuracy of the individual algorithms as well as the combinations are measured using the F-score and the mean Intersection over Union (mIoU). The F-score signifies a balance between precision and recall, where the mIoU evaluates the overlap between the predicted and ground truth regions. Together, these metrics contribute to a comprehensive evaluation framework.

Section III mentioned that part of the preprocessing is to fine-tune the SUIM-Net. This fine-tuning includes modifying its architecture and training it with fish images. For the current work, the modified SUIM-Net, still a deep neural network, was trained *one-time* with 2,200 already annotated fish images from the Seagrass dataset. It was not re-trained with manually labeled IRWA frames, and this saved hours of additional training time. The manually labeled IRWA frames only trained the single-layer neural network for weighted voting, and this took only a few minutes.

Fig. 5 shows that the mean square error loss in eq. (1) converges to zero as the model learns the optimal weights. This is true for both Seagrass and IRWA datasets. Table I compares the performance of the individual localization methods and their combinations. Fig. 6 shows the weights for each method determined by WV. Figs. 7 and 8 show the outputs of each algorithm for one video frame each from Seagrass and IRWA respectively. Since the SUIM-Net is fine-tuned with the Seagrass images, it achieves much higher validation accuracy compared to SuBSENSE and RLOF. Figures 7(b) - 7(d) further demonstrate SUIM-Net's superior performance. Unlike SUIM-Net, SuBSENSE fails to detect the fish hidden in the background (seagrass). Moreover, RLOF's performance is sub-par because of no substantial fish movement in this video. Fig. 6 and Table I together confirm that WV assigns higher weight to SUIM-Net and achieves better accuracy than SV. This is also reflected in Figures 7(e) - 7(f), which show that WV can indeed localize the fish hidden in seagrass, but SV cannot.

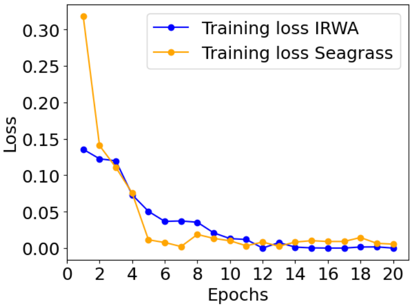


Fig. 5. Training progress of weighted voting: loss vs. epoch for the Seagrass and IRWA datasets

TABLE I

Validation Accuracy: F-score and Mean Intersection over Union (mIoU) for the Seagrass and IRWA Datasets

Method	Seagrass F-score	Seagrass mIoU	IRWA F-score	IRWA mIoU
SuBSENSE	28.17	30.88	**38.42**	**24.53**
SUIM-Net	**81.25**	**60.94**	31.65	25.71
RLOF	26.57	14.54	19.08	16.37
SV	75.63	58.48	43.35	35.46
WV	**80.71**	**60.29**	**61.05**	**41.66**

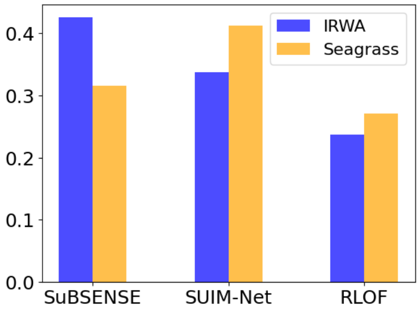
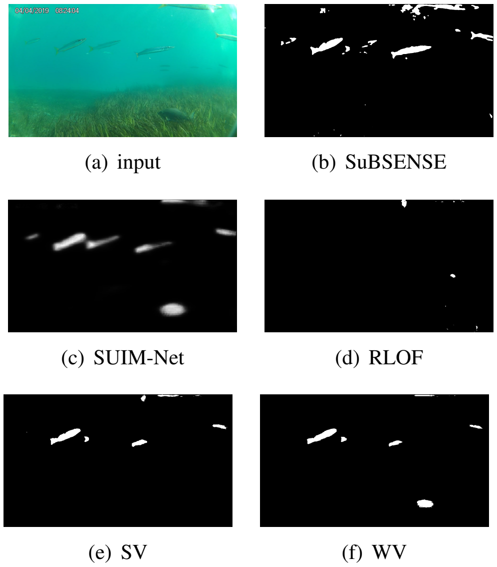


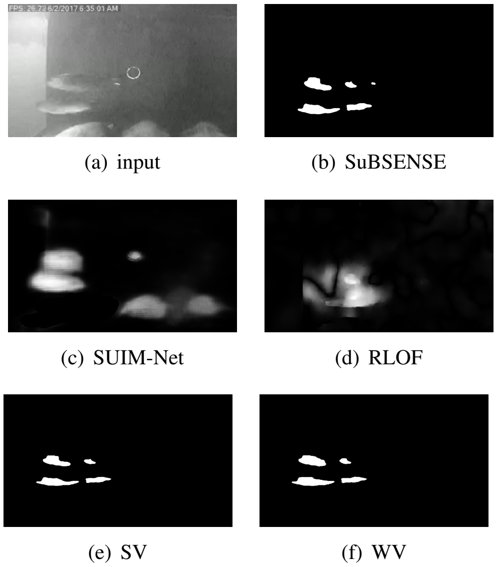
Fig. 6. Computed optimal weights assigned to SuBSENSE, SUIM-Net, and RLOF for both Seagrass and IRWA datasets towards weighted voting


(a) input (b) SuBSENSE

(c) SUIM-Net (d) RLOF

(e) SV (f) WV

Fig. 7. Localized fish in a video frame of the Seagrass dataset


(a) input (b) SuBSENSE

(c) SUIM-Net (d) RLOF

(e) SV (f) WV

Fig. 8. Localized fish in a video frame of the IRWA dataset

For the IRWA dataset, Table I shows that SuBSENSE has better validation accuracy than SUIM-Net and RLOF. It can be seen in Fig. 8(c) that SUIM-Net underperforms in an unseen background setting. The presence of sandbags in the background and the lack of color information make it challenging for SUIM-Net to identify the fish. In particular, it misidentifies the sandbags as semantics belonging to fish. In contrast, Figure 8(b) shows that SuBSENSE can eliminate the static sandbags and localize the moving fish. Fig. 8(d) shows that RLOF can eliminate the sandbags but fails to identify the fish. Even though the fish had swift movements in this video, the noise level was high, and the illumination was not so good as Seagrass. This might have resulted in the poor performance of the RLOF, which can be confirmed by the accuracy metrics in Table I. Figs. 8(e) - 8(f) show reasonably good performance of SV as well as WV. Both are able to localize the two fish in the video frame, although the output of WV is slightly closer to the actual shape of the fish. Fig. 6 shows that for this dataset WV assigns the highest weight to SuBSENSE which predicts the closest to the ground truth, and lesser weights to the other two methods.

The above observations for two significantly different backgrounds establish WV as a strong candidate for background-agnostic, real-time fish localization. While the performance of individual algorithms varied from one background to another, the combination using WV was able to achieve adequate localization. This was accomplished by choosing optimal weights such that the method with a higher validation accuracy received a higher weight. WV outperformed SV, albeit at the cost of some manual labeling. Furthermore, for both datasets the baseline algorithm SuBSENSE ran at 20 frames per second, while RLOF and SUIM-Net supported SuBSENSE every 2-3 frames. The combination still worked in real-time and maintained a balance between accuracy and speed, demonstrating its practicality for real-world fish localization applications.

VI. Conclusion

This work developed a real-time, background-agnostic underwater fish localization framework by combining three algorithms using ensemble learning. The baseline algorithm performed background subtraction in real-time, and its output was fine-tuned with semantic segmentation and optical flow from time to time. The combination worked in real-time and accomplished localization with good accuracy. Each individual algorithm came with its own unique benefits, and results showed that the performance of the same algorithm may not remain uniform across underwater environments. However, weighted voting led to a robust localization framework which produced consistent results irrespective of the background. The consistency was maintained by prioritizing the algorithm with the highest prediction accuracy in a given environment. Results generated by the framework are useful for fish species identification and counting, which in turn are the prerequisites for autonomous ecosystem monitoring and understanding of long-term fish behavior.

Acknowledgment

This research is supported by the Massachusetts Institute of Technology Sea Grant under subaward number s5969. The authors are grateful to John Sheppard, Meghna Marjadi, and Cameron Mackenzie for supporting this work.

References

[1] J. Rosset, A. H. Roy, B. I. Gahagan, A. R. Whiteley, M. P. Armstrong, J. J. Sheppard, and A. Jordan, *"Temporal Patterns of Migration and Spawning of River Herring in Coastal Massachusetts"*, Transactions of the American Fisheries Society, vol. 146, pp. 1101 – 1114, 2017, doi: 10.1080/00028487.2017.1341851.

[2] H. D. Legett, A. Jordaan, A. H. Roy, J. J. Sheppard, M. Somos-Valenzuela, and M. D. Staudinger, *"Daily Patterns of River Herring (Alosa spp.) Spawning Migrations: Environmental Drivers and Variation among Coastal Streams in Massachusetts"*, Transactions of the American Fisheries Society, vol. 150, iss. 4, pp. 501 – 513, 2021, doi: 10.1002/tafs.10301.

[3] G. A. Nelson, *"A Guide to Statistical Sampling for the Estimation of River Herring Run Size Using Visual Counts"*, Massachusetts Division of Marine Fisheries Technical Report TR-25, 2006 (34 pages).

[4] K. H. Bieluch, T. Willis, J. Smith, and K. A. Wilson, *"The Complexities of Counting Fish: Engaging Citizen Scientists in Fish Monitoring"*, Maine Policy Review, vol. 26, iss. 2, pp. 9 – 18, https://digitalcommons.library.umaine.edu/mpr/vol26/iss2/4.

[5] D. Li, Z. Miao, F. Peng, L. Wang, Y. Hao, Z. Wang, T. Chen, H. Li, and Y. Zheng, *"Automatic counting methods in aquaculture: A review"*, Journal of the World Aquaculture Society, vol. 52, pp. 269 – 283, 2020, doi: 10.1111/jwas.12745.

[6] X. Yang, S. Zhang, J. Liu, Q. Gao, S. Dong, and C. Zhou, *"Deep learning for smart fish farming: applications, opportunities and challenges"*, Reviews in Aquaculture, vol. 13, iss. 1, pp. 66 – 90, 2020, doi: 10.1111/raq.12464.

[7] J. J. Sheppard and M. S. Bednarski, *"Utility of Single-Channel Electronic Resistivity Counters for Monitoring River Herring Populations"*, North American Journal of Fisheries Management, vol. 35, no. 6, pp. 1144-1151, 2015, doi: 10.1080/02755947.2015.1084407.

[8] H. -H. Jebamikyous and R. Kashef, *"Autonomous Vehicles Perception (AVP) Using Deep Learning: Modeling, Assessment, and Challenges"*, IEEE Access, vol. 10, pp. 10523-10535, 2022, doi: 10.1109/ACCESS.2022.3144407.

[9] L. Tai, S. Li, and M. Liu, *"Autonomous exploration of mobile robots through deep neural networks"*, International Journal of Advanced Robotic Systems, Special Issue on Robotic Applications Based on Deep Learning, pp. 1 – 9, July – August 2017, doi: 10.1177/1729881417703571.

[10] J. Janai, F. Güney, A. Behl, and A. Geiger, *"Computer Vision for Autonomous Vehicles: Problems, Datasets and State of the Art"*, Foundations and Trends in Computer Graphics and Vision: vol. 12, no. 1–3, pp 1-308, 2020. doi: http://dx.doi.org/10.1561/0600000079

[11] M. Piccardi, *"Background subtraction techniques: a review"*, pp. 3099-3104, Proceedings of the 2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583), The Hague, Netherlands, 10 – 13 October 2004, doi: 10.1109/ICSMC.2004.1400815.

[12] Y. Benezeth, P. M. Jodoin, B. Emile, H. Laurent, and C. Rosenberger, *"Review and Evaluation of Commonly-Implemented Background Subtraction Algorithms"*, pp. 1-4, Proceedings 2008 19th International Conference on Pattern Recognition, Tampa, FL, USA, 8 – 11 December 2008, doi: 10.1109/ICPR.2008.4760998.

[13] T. Bouwmans and B. G. Garcia, *"Background Subtraction in Real Applications: Challenges, Current Models and Future Directions"*, Computer Science Review, vol. 35, no. 1, pp. ?? -??, 2019. DOI:10.1016/j.cosrev.2019.100204.

[14] R. Kalsotra and S. Arora, *"A Comprehensive Survey of Video Datasets for Background Subtraction"*, IEEE Access, vol. 7, pp. 59143-59171, 2019, doi: 10.1109/ACCESS.2019.2914961.

[15] Y. Wang, P.-M. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, P. Ishwar, *"CDnet 2014: An Expanded Change Detection Benchmark Dataset"*, 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops, Columbus, OH, USA, 2014, pp. 393-400, doi: 10.1109/CVPRW.2014.126.

[16] A. Sobral and A. Vacavant, *"A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos"*, Computer Vision and Image Understanding, vol. 122, pp. 4 – 21, May 2014, doi: https://doi.org/10.1016/j.cviu.2013.12.005.

[17] P. -L. St-Charles, G. -A. Bilodeau, and R. Bergevin, *"SuBSENSE: A Universal Change Detection Method With Local Adaptive Sensitivity"*, IEEE Transactions on Image Processing, vol. 24, no. 1, pp. 359-373, Jan. 2015, doi: 10.1109/TIP.2014.2378053.

[18] H. Liu, J. Dai, R. Wang, H. Zheng and B. Zheng, *"Combining background subtraction and three-frame difference to detect moving object from underwater video"*, pp. 1-5, Proceedings of the 2016 IEEE OCEANS Conference, Shanghai, 10 – 13 April 2016, doi: 10.1109/OCEANSAP.2016.7485613.

[19] J. Wu, G. Huang, H. Zheng, G. -L. Huang, Y. Hu and J. He, *"Repeatable Pattern Mining for Accurate Subtraction of Backgrounds with Waving Objects in Underwater Videos"*, pp. 1 – 11, Proceedings of the 2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA), Shenzhen, China, 13 – 16 October 2022, doi: 10.1109/DSAA54385.2022.10032438.

[20] M. Braham, S. Piérard, and M. Van Droogenbroeck, *"Semantic background subtraction"*, pp. 4552 – 4556, Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17 – 20 September 2017, doi: 10.1109/ICIP.2017.8297144.

[21] A. Cioppa, M. V. Droogenbroeck, and M. Braham, *"Real-Time Semantic Background Subtraction"*, pp. 3214-3218, Proceedings of the 2020 IEEE International Conference on Image Processing (ICIP), Abu Dhabi, United Arab Emirates, 25 – 28 October 2020, doi: 10.1109/ICIP40778.2020.9190838.

[22] M. J. Islam, C. Edge, Y. Xiao, P. Luo, M. Mehtaz, C. Morse, S. S. Enan, and J. Sattar, *"Semantic Segmentation of Underwater Imagery: Dataset and Benchmark"*, pp. 1769 – 1776, Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, October 25 – 29, 2020, doi: 10.1109/IROS45743.2020.9340821.

[23] D. J. Fleet and Y. Weiss, *"Optical Flow Estimation"*, Ch. 15, pp. 239 – 258 of book "Mathematical Models in Computer Vision: The Handbook", editors: N. Paragios, Y. chen, and O. Faugeras, Springer, New York, NY, 2005, doi: https://doi.org/10.1007/0-387-28831-7.

[24] A. Agarwal, S. Gupta, and D. K. Singh, *"Review of optical flow technique for moving object detection"*, Proceedings of the 2nd International Conference on Contemporary Computing and Informatics (IC3I), Greater Noida, India, 2016, pp. 409-413, doi: 10.1109/IC3I.2016.7917999.

[25] Y. Tanaka, S. Yamabe, K. Fukae, T. Imai, K. Arai and T. Kobayashi, *"Quantification of Fish Behavior Using Optical Flow"*, Proceedings of the IEEE 11th Global Conference on Consumer Electronics (GCCE), Osaka, Japan, 2022, pp. 125-126, doi: 10.1109/GCCE56475.2022.10014411.

[26] M. Ravanbakhsh, M. R. Shortis, F. Shafait, A. Mian, E. S. Harvey, and J. W. Seager, *"Automated Fish Detection in Underwater Images Using Shape-Based Level Sets"*, Photogrammetric Record, vol. 30, no. 149, pp. 46–62, 2015, doi: 10.1111/phor.12091.

[27] T. Senst, V. Eiselein, and T. Sikora, *"Robust Local Optical Flow for Feature Tracking"*, IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 9, pp. 1377-1387, Sept. 2012, doi: 10.1109/TCSVT.2012.2202070.

[28] R. Polikar, *"Ensemble Learning"*, Ch. 1, pp. 1 – 34 of book "Ensemble Machine Learning", editors: C. Zhang and Y. Ma, Springer, New York, NY, 2012, https://doi.org/10.1007/978-1-4419-9326-7_1.

[29] J. Snoek, H. Larochelle, and R. P. Adams, *"Practical Bayesian optimization of machine learning algorithms"*, pp. 2951 – 2959, Proceedings of the 25th International Conference on Neural Information Processing Systems (NIPS), vol. 2, December 2012.

[30] E. M. Ditria, R. M. Connolly, E. L. Jinks, and S. Lopez-Marcano, *"Annotated Video Footage for Automated Identification and Counting of Fish in Unconstrained Seagrass Habitats"*, Frontiers in Marine Science, vol. 8, article 629485, (5 pages), 2021, doi: 10.3389/fmars.2021.629485.