Semantic Segmentation of Water Surfaces With U-Net

Kristjan Šoln
Faculty of Electrical Engineering,
University of Ljubljana
ks4835@student.uni-lj.si

Abstract

Computer vision in aquatic environment is important for applications such as security and automated navigation of unmanned surface vehicles. The complexity of such an unconstrained environment poses a real challenge for detection and segmentation algorithms. Various semantic segmentation networks exist, some designed explicitly for aquatic semantic segmentation, while others address different domains or serve general purposes. In this paper, we evaluate U-Net, a well known semantic segmentation network, in a fluvial segmentation task. We perform experiments on a set of 2,091 images captured from a boat navigating a busy waterway. The images are automatically annotated using WaSR, a state of the art water segmentation algorithm. We report the performance between 97.66 and 99.42 percent in terms of IoU on our dataset.

1. Introduction

Obstacle detection in aquatic environments is vital for a range of applications such as vessel security and automated surveillance. With advancements in robotics, unmanned surface vehicles (USVs) have emerged, integrating computer vision systems to identify and navigate around obstacles, performing autonomous decision-making. One of the most affordable approaches for such detection is the usage of cameras, combined with computer vision algorithms [4, 5, 8].

Classical methods for obstacle detection in aquatic environments are often handcrafted and include object classification [21], salience-based methods [7, 30, 32], and stereo-based [2, 25, 33] detection methods. These approaches often struggle with issues related to generalization, dataset size requirements, and robustness in the presence of motion, reflections, waves, and other visual artifacts. Deep learning-based approaches typically outperform traditional methods in object detection [34]. However, semantic segmentation is often preffered in such tasks as it provides a more detailed description of the entire scene. Several general purpose segmentation models have been evaluated in aquatic segmentation scenarios [5, 8]. Novel methods tend to use segmentation networks, specifically intended for aquatic environments [4, 15, 31].

Segmentation in aquatic environments for the purpose of obstacle detection and navigation is still a developing area of research. Most approaches focus on maritime environments and harbor zones, and segmentation networks focusing specifically on fluvial environment are particularly rare [18]. This is partly due to lack of appropriate public datasets [5, 18].

In this paper we use U-Net [28], a well known semantic segmentation network originally intended for biomedical image segmentation, and evaluate its performance in a fluvial environment. We train it on 2,091 images, captured from a boat navigating a busy waterway, and annotated using current state of the art (SOTA) algorithm for water segmentation [4].

The rest of the paper is structured as follows: section 2 discusses related work in the area of aquatic obstacle detection and segmentation. In section 3, we discuss the selected method and present an analysis of the model structure. Section 4 presents our experimental setup, used datasets and obtained results. The paper concludes with a summary in section 5, discussing the effectiveness of selected method and outlining potential future work.

2. Related Work

Visual detection of obstacles in maritime surveillance is an active area of research. In one of the early approaches, Marques *et al.* [22] detect vessels within aerial images by sets of color features. Another approach by Loomans *et al.* [21] detects maritime vessels using Histograms of Oriented Gradients and is specifically trained to detect ship cabins. Several eariler methods perform maritime obstacle detection based on saliency. Approach by Cane *et al.* [7] generates a saliency map and matches obstacles between neighbouring frames using a Kalman filter. Approach by Sobral *et al.* [30] uses Robust Principal Component Analysis to generate a saliency map. In [32], Tran *et al.* use Spectral

Redisual and Maximum Symetric Surround methods combined with background substraction. These methods still only detect objects and cannot distinguish between different surfaces in the image. In [17], Kristan *et al.* used graphical models to detect USVs, relying on random fields networks to calculate the saliency map. Mentioned approaches are often handcrafted, and while they can offer good obstacle detection, they still suffer in the presence of reflections, ambiguous objects and diverse scenery.

There also exist several maritime obstacle detection approaches using stereo camera systems [2, 25, 29, 33]. These methods often suffer in situations where little texture is present on the surface of the water or objects. In [25], Muhovic *et al.* introduced depth fingerprinting, which improves performance in such situations. Because this method assumes water to be the largest surface in the image, it can perform poorly in the presence of large objects such as piers and large ships. In [3, 6], Bovcon *et al.* used readings from an on-board inertial measurement unit (IMU) to improve performance of the stereo camera system.

Even though these methods perform reasonably well, deep learning approaches provide better performance in detection and segmentation tasks. For example, Lee *et al.* [19] and Yang *et al.* [34] used this approach to perform detection and classification of maritime vessels. While detection using deep learning approaches outperforms conventional detection methods, it is difficult to generalize them to obstacles that were not included in the training dataset.

Semantic segmentation is used to perform classification for every pixel in the image. This fine-grained inference can provide a better description of the image compared to bounding boxes provided by detection algorithms [13]. Some of the pre-deep learning methods for such segmentation include [24] thresholding, k-means clustering, region growing and random fields methods. Deep learning semantic segmentation approaches can be separated in several categories [13, 24], including Fully Convolutional Networks (FCN), Encoder-Decoder based models, Reccurent Neural Networks (RNN), Attention based models, Dilated Convolutional models and Generative Adversarial (GAN) models. Long *et al.* [20] introduced a general purpose FCN model that modified successful classification models in order to output a spatial map, later upsampled using deconvolutions. This produces a dense classification of individual pixels. This work served as a basis and inspiration for many other segmentation methods [13] and provided significant improvement over traditional methods. Yu *et al.* [35] introduced a Dilated Convolutional model that use a multi-scale context aggregation module. Paszke *et al.* introduced ENet [26], a general purpose segmentation model that can operate in real-time. Ronneberger *et al.* introduced U-Net [28], an Encoder-Decoder segmentation network, intended for biomedical image segmentation. The network improves in localization of the pixels by introducing skip connections, which take the features from layers in the encoder part and concatenate them with the decoder features. Later, Badrinarayanan *et al.* introduced SegNet [1], a similar model which reuses pooling indices to guide precise upsampling during decoding, allowing more accurate segmentation. Mehta *et al.* introduced ESPNet [23], which is based on a spatial pyramid module. Zhao *et al.* introduced PSPNet [36], a general purpose pyramid scene parsing network which uses a ResNet [14] backbone. Similarly, Chen *et al.* introduced DeepLabv1 [9] and DeepLabv2 [10], both Dilated Convolutional Models and are considered to be among the most popular segmentation algorithms [24]. Later, the authors proposed DeepLabv3 [11] and DeepLabv3+ [12], the latter being an Encoder-Decoder model which uses the DeepLabv3 model for the encoder part.

Ferryman *et al.* was among the first to evaluate deep learning segmentation methods in the context of maritime obstacle detection and surveillance [8]. The study compared SegNet [1], ENet [26] and ESPNet [23] on several maritime surveillance datasets. SegNet and ENet achieved higher detection accuracy while ESPNet performed better during classification. Another study by Bovcon *et al.* [5] evaluated three different segmentation models in the context of maritime detection on unmanned surface vehicles. They compared U-Net [28], PSPNet [36] and DeepLabv2 [10] on their public MaSTr1325 dataset [5]. They discovered that DeepLabv2 outperformed the other two models and provides a good trade-off between reliable obstacle detection and low false alarm rate.

Kim *et al.* proposed Skip-ENet [15], a segmentation model intended specifically for maritime environment. The model is based on ENet [26], with whitening to improve performance in foggy conditions, adjusted reception size in the dilated convolutions and additional skip connections. Steccanella *et al.* proposed a modified U-Net structure [31] that uses depthwise convolution in order to improve performance. Recently, Bovcon *et al.* introduced WaSR [4], a DeepLabv2 [10] based segmentation model, designed specifically for maritime environment and usage with USV. The model uses IMU readings to improve water-edge estimation. Authors report state-of-the-art performance in obstacle detection and segmentation tasks.

As mentioned, several earlier aquatic segmentation methods use existing, general purpose models and train them on appropriate maritime datasets. Similarly, in this work we evaluate the usage of U-Net [28] in a fluvial environment.

3. Methodology

Our approach analyzes the performance of U-Net architecture [28] in semantic segmentation of a river-like environment. U-Net is composed of an encoder and a decoder part, capturing context and expanding it to create a segmentation mask of the input image. The network improves localization by implementing skip connections between high resolution features of the encoder and the expanded output features. This allows for a more precise assembly of the output mask. The architecture of the network is presented in figure 1.

3.1. Encoder and Decoder Architecture

Originally inspired by [20], U-Net first encodes the input image via a contracting path, similar to a conventional convolutional network. This allows compression of features from the original image into a latent-space representation. The encoder stage consists of repeated sections of two 3×3 convolutions, each convolution followed by a ReLU (rectified linear unit) layer. Sections are separated by 2×2 max pooling operations. This way, every section doubles the amount of feature channels.

The semantic information, captured by the encoder, is then fed into the decoder, which upsamples the latent features into an output mask. The features are expanded using a series of standard convolutions, transposed convolutions (also known as up-convolutions [28]) and feature concatenations. Convolution operations are always followed by a ReLU layer. The final layer consists of a 1×1 convolution, which converts the features into a single-channel representation of the image. Finally, these values are normalized through sigmoid transformation.

Our particular implementation differs from the original [28] in one aspect: unlike the original U-Net design, which incorporates cropping of copied feature maps, our approach employs padding equal to 1 during standard convolutions. This choice eliminates the need for additional cropping of the copied features.

3.2. Skip connections

It is important for the network to localize well. This is achieved using skip connections, denoted by the *copy* operation in figure 1. High resolution features from the encoding stage are combined with the features in the decoding stage. This results in the networks ability to precisely localize the information and generate a more accurate output mask.

4. Experiments

This section provides details on the experimental protocol, performance measures and used datasets. We present the results of our evaluation and a corresponding discussion.

4.1. Databases

Segmentation models, such as U-Net, need to be trained on datasets that are annotated in a pixel-wise fashion. Our dataset comprises of 2,091 images with a resolution of $1{,}104 \times 621$ pixels, captured from a boat navigating a river. Obstacles, such as other vessels, piers, and structures in proximity of the water are also present in the dataset. Ground truth for these images is generated using WaSR [4], and contain regions of water, sky and obstacles, which include shoreline, houses and other vessels. A sample from the dataset is presented in figure 2.

The dataset is split into training, validation, and test subsets, comprising 70, 10, and 20 percent of the dataset, respectively. The images are assigned their subset using two different approaches: random assignment and manual assignment. As the images are just successive frames, taken from video footage, two neighboring frames do not differ significantly. Manual selection is chosen to avoid having very similar images in both training and test datasets. This way, the split contains groups of multiple neighbouring frames, representing scenes of longer duration. All three created subsets of images were examined to contain challenging elements such as other vessels, structures in proximity to the river, and piers. With this manual split, the subsets are less similar to each other, which allows for a fairer evaluation of the model.

4.2. Performance measures

As suggested by [20], we use the following measures, often used in semantic segmentation evaluation [5, 13, 15, 20, 24]:

- *Pixel Accuracy* is the ratio of correctly classified pixels divided by the sum of all pixels. In two-class segmentation context, it can be defined as shown in equation 1.

$$PA = \frac{TP + TN}{TP + FP + TN + FN} \tag{1}$$

- *Intersection over Union* (*IoU*), also known as the *Jaccard Index*, is the intersection between the prediction A and the ground truth B, divided by the union of both areas. In two-class segmentation context, it can be defined as shown in equation 2.

$$IoU = J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{2}$$

4.3. Implementation details

The provided images are preprocessed in order to augment the dataset. As proposed in [5], images are geometrically augmented by random vertical mirroring and central rotations in the range of ± 5 degrees. All images are normalized, subjected to random brightness and contrast change by a factor of up to 0.2, and rescaled to 528×960 pixels according to hardware capabilities.

Binary cross-entropy loss is used to measure the error between output of the model and provided ground truth,

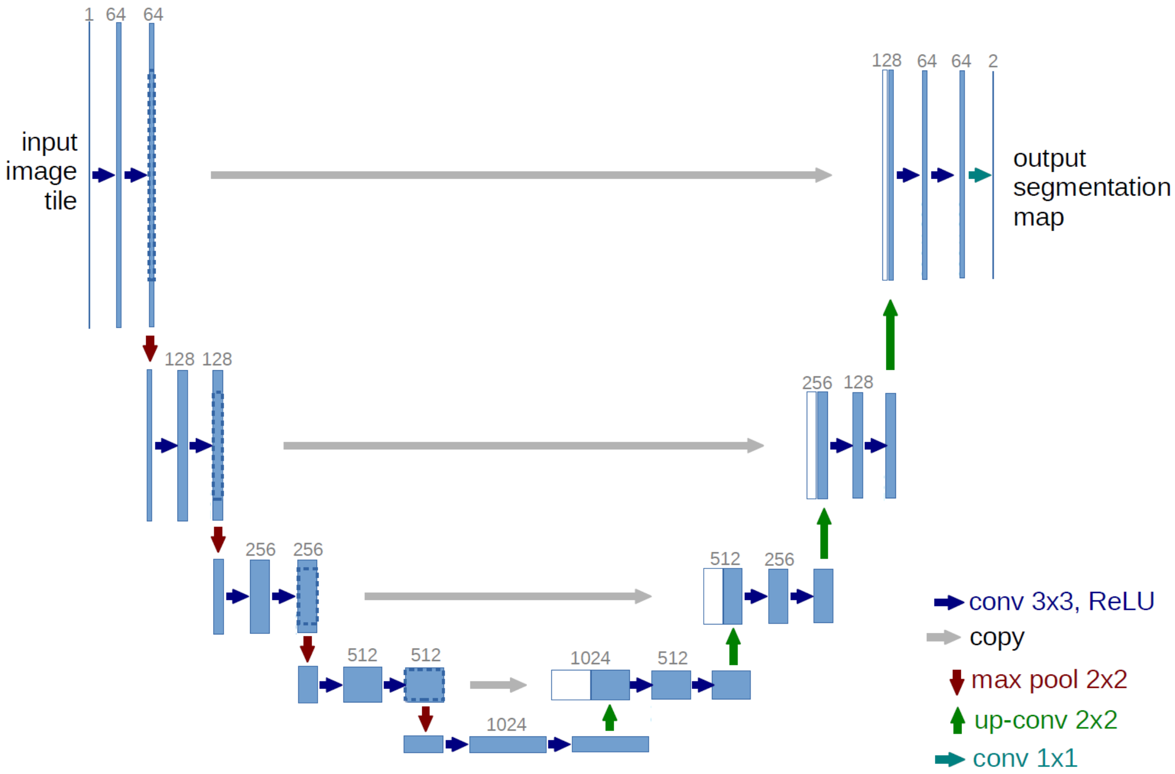

Figure 1. **U-Net architecture.** Each blue box represents a multi-channel feature map, with the number of channels denoted on the top of the box. White boxes represent copied feature maps, while arrows represent operations as denoted on the right. Convolution operations use 3 × 3 kernels with padding equal to 2 and stride equal to 1. All up-convolution operations use a 2 × 2 kernel, no padding and a stride equal to 2. Figure extracted from [28].

along with the Adam optimizer [16] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ to train the network. The initial learning rate is set to 10^{-4}. A scheduler with a 1% threshold is employed, reducing the rate to 10^{-5} when the loss value plateaus. Following this, an early stop mechanism can stop the training once the same conditions are met again. The patience for both mechanisms is set to 5 epochs.

The model is trained on both dataset split variants, as described in 4.1, for a total of 40 epochs, using batch size of 1, adjusting to the capabilities of selected hardware. The model is implemented in PyTorch [27] and run on a desktop computer with Intel Core i5-13600K and Nvidia GeForce GTX 1050 Ti.

4.4. Results

Evaluation results are displayed in table 1. The model performs better when the assignment of images to subsets is random. Notably, in this scenario, the model trained for the maximum specified epochs, whereas in the case of manual subset assignment, the early stop mechanism halted training at epoch 29. We hypothesize that worse performance in the manual assignment case could be caused by increased diversity between individual subsets images. Here, subsets consists of longer, uninterrupted sequences of images, leading to less similarity among subsets. This could imply a fairer evaluation of the model. On the contrary, it could also imply that the training dataset is less diverse, potentially leading to less successful training. The early stop mechanism failed to trigger in the case of random assignment. This could indicate that it was unable to detect when the model stopped extracting useful information, possibly due to lack of diversity between subsets.

Subset assignment type	PA	IoU	Epochs
Random	99.74	99.42	40
Manual	98.90	97.66	29

Table 1. **Comparison of model performance with regard to type of dataset split.** The model performs worse when images are manually assigned to either train, validation or test subsets. The early stop mechanism triggered only during manual assignment. This could indicate failure to detect overfitting in the random assignment case, possibly due to lack of diversity between subsets.

An example of a successful segmentation is presented in

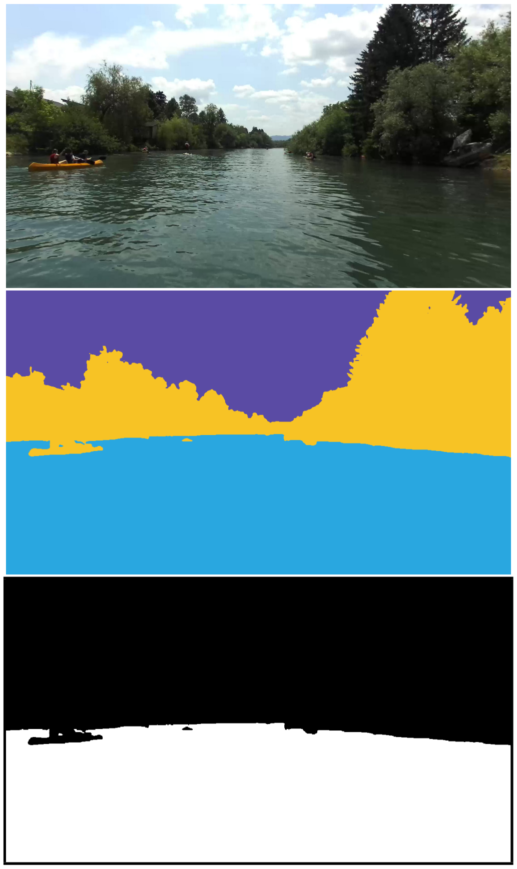

Figure 2. **Sample from the dataset used to train and evaluate the model.** The first image is the original image, captured by the camera. The second image represents the ground truth, automatically generated using WaSR [4]. Obstacle and sky annotations are then merged into a single class, as shown in the third image. This way, the final ground truth only contains information on whether a pixel represents water or not.

figure 3. The model performs well in the presence of boat-like objects, but struggles with large, bridge-type objects. Even though the dataset with manual assignment includes bridge-type objects in all three subsets, such objects are not common in the dataset, which negatively affects the ability of the model to correctly segment such scenes. Bridges seem particularly problematic when directly above the vessel. Additionally, ground truth in such scenes is often poorly annotated by WaSR, which further degrades performance. This is demonstrated in figure 4. The performance of U-Net in such situations is better when using the dataset with random assignment. This might indicate a richer train dataset, improving the performance of the model. Overall, we conclude that the model displays promising performance in the segmentation task of a fluvial environment.

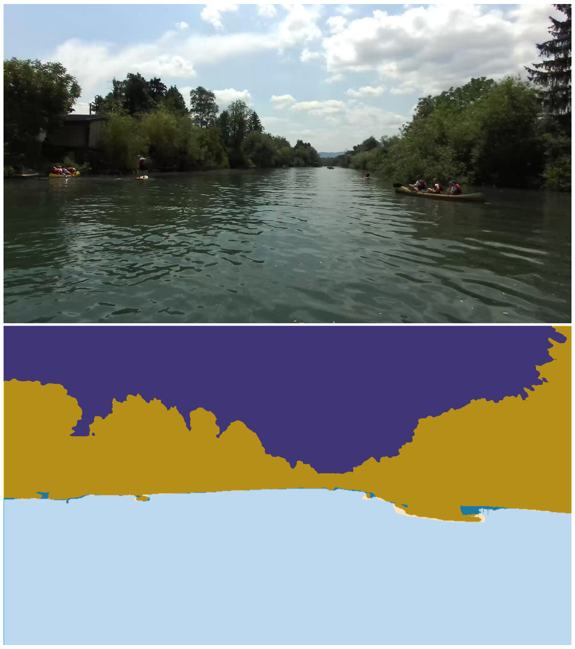

Figure 3. **Example of a successful segmentation by U-Net.** The top figure depicts the original scene captured by the camera, while the bottom representation combines both the ground truth produced by WaSR [4] and the output of the U-Net model. In the representation, purple, yellow, and blue denote the WaSR output, corresponding to the sky, obstacles, and water, respectively. The U-Net output is represented by superimposed brighter and darker areas, with the former indicating the presence of water. In this case, U-Net is trained on the manually assigned dataset. This particular image achieved $IoU = 99.05$ and $PA = 99.57$.

5. Conclusion

In this study, we assessed the performance of U-Net, a widely recognized semantic segmentation network, in the context of a fluvial environment segmentation task. Our experiments involved a dataset comprising 2,091 images captured from a boat navigating a busy waterway. The results indicate promising performance, with an average IoU of 97.66% in the more challenging experiment and an even higher IoU in the alternative case.

Potential areas for future work include: improving the quality of ground truth annotations, (as demonstrated in figure 4), expanding and diversifying the dataset, perhaps by incorporating separate footage for testing, and evaluating additional, newer segmentation models.

References

[1] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

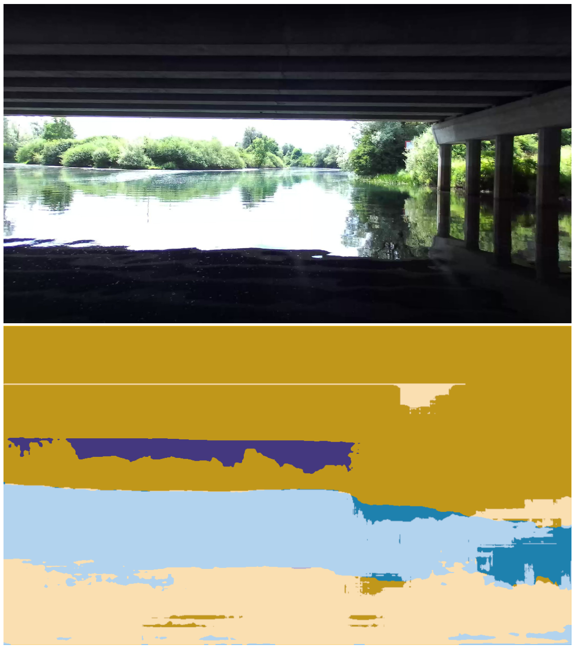

Figure 4. **Example of a poor ground truth annotation as provided by WaSR.** See figure 3. Here, WaSR detected the poorly lit water under the bridge as an obstacle, which generated an incorrect ground truth annotation. This model was trained on the manually assigned subsets. This particular image achieved $IoU = 45.20$ and $PA = 74.37$.

[2] Borja Bovcon and Matej Kristan. Obstacle detection for usvs by joint stereo-view semantic segmentation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5807–5812. IEEE, 2018.

[3] Borja Bovcon and Matej Kristan. Obstacle detection for usvs by joint stereo-view semantic segmentation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5807–5812. IEEE, 2018.

[4] Borja Bovcon and Matej Kristan. Wasr—a water segmentation and refinement maritime obstacle detection network. *IEEE Transactions on Cybernetics*, 52(12):12661–12674, 2021.

[5] Borja Bovcon, Jon Muhovič, Janez Perš, and Matej Kristan. The mastr1325 dataset for training deep usv obstacle detection models. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3431–3438. IEEE, 2019.

[6] Borja Bovcon, Janez Perš, Matej Kristan, et al. Stereo obstacle detection for unmanned surface vehicles by imu-assisted semantic segmentation. *Robotics and Autonomous Systems*, 104:1–13, 2018.

[7] Tom Cane and James Ferryman. Saliency-based detection for maritime object tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 18–25, 2016.

[8] Tom Cane and James Ferryman. Evaluating deep semantic segmentation networks for object detection in maritime surveillance. In *2018 15th IEEE international conference on advanced video and signal based surveillance (AVSS)*, pages 1–6. IEEE, 2018.

[9] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014.

[10] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.

[11] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[12] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.

[13] Alberto Garcia-Garcia, Sergio Orts-Escolano, Sergiu Oprea, Victor Villena-Martinez, and Jose Garcia-Rodriguez. A review on deep learning techniques applied to semantic segmentation. *arXiv preprint arXiv:1704.06857*, 2017.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[15] Hanguen Kim, Jungmo Koo, Donghoon Kim, Byeolteo Park, Yonggil Jo, Hyun Myung, and Donghwa Lee. Vision-based real-time obstacle segmentation algorithm for autonomous surface vehicle. *IEEE Access*, 7:179420–179428, 2019.

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[17] Matej Kristan, Vildana Sulić Kenk, Stanislav Kovačič, and Janez Perš. Fast image-based obstacle detection from unmanned surface vehicles. *IEEE transactions on cybernetics*, 46(3):641–654, 2015.

[18] Reeve Lambert, Jalil Chavez-Galaviz, Jianwen Li, and Nina Mahmoudian. Rosebud: A deep fluvial segmentation dataset for monocular vision-based river navigation and obstacle avoidance. *Sensors*, 22(13):4681, 2022.

[19] Sung-Jun Lee, Myung-Il Roh, Hye-Won Lee, Ji-Sang Ha, and Il-Guk Woo. Image-based ship detection and classification for unmanned surface vehicle using real-time object detection neural networks. In *ISOPE International Ocean and Polar Engineering Conference*, pages ISOPE–I. ISOPE, 2018.

[20] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[21] Marijn JH Loomans, Peter HN de With, and Rob GJ Wijnhoven. Robust automatic ship tracking in harbours using active cameras. In *2013 IEEE International Conference on Image Processing*, pages 4117–4121. IEEE, 2013.

[22] Jorge S Marques, Alexandre Bernardino, Gonçalo Cruz, and Maria Bento. An algorithm for the detection of vessels in aerial images. In *2014 11th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 295–300. IEEE, 2014.

[23] Sachin Mehta, Mohammad Rastegari, Anat Caspi, Linda Shapiro, and Hannaneh Hajishirzi. Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation. In *Proceedings of the european conference on computer vision (ECCV)*, pages 552–568, 2018.

[24] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3523–3542, 2021.

[25] Jon Muhovic, Rok Mandeljc, Janez Perš, and Borja Bovcon. Depth fingerprinting for obstacle tracking using 3d point cloud. In *Proc. 23rd Comput. Vis. Winter Workshop*, pages 71–78, 2018.

[26] Adam Paszke, Abhishek Chaurasia, Sangpil Kim, and Eugenio Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv preprint arXiv:1606.02147*, 2016.

[27] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.

[29] Armando J Sinisterra, Manhar R Dhanak, and Karl von Ellenrieder. Stereo vision-based target tracking system for an usv. In *2014 Oceans-St. John's*, pages 1–7. IEEE, 2014.

[30] Andrews Sobral, Thierry Bouwmans, and El-hadi ZahZah. Double-constrained rpca based on saliency maps for foreground detection in automated maritime surveillance. In *2015 12th IEEE international conference on advanced video and signal based surveillance (AVSS)*, pages 1–6. IEEE, 2015.

[31] Lorenzo Steccanella, Domenico Daniele Bloisi, Alberto Castellini, and Alessandro Farinelli. Waterline and obstacle detection in images from low-cost autonomous boats for environmental monitoring. *Robotics and Autonomous Systems*, 124:103346, 2020.

[32] Thanh-Hai Tran and Thi-Lan Le. Vision based boat detection for maritime surveillance. In *2016 International Conference on Electronics, Information, and Communications (ICEIC)*, pages 1–4. IEEE, 2016.

[33] Han Wang, Zhuo Wei, Sisong Wang, Chek Seng Ow, Kah Tong Ho, and Benjamin Feng. A vision-based obstacle detection system for unmanned surface vehicle. In *2011 IEEE 5th International Conference on Robotics, Automation and Mechatronics (RAM)*, pages 364–369. IEEE, 2011.

[34] Jie Yang, Yinghao Li, Qingnian Zhang, and Yongmei Ren. Surface vehicle detection and tracking with deep learning and appearance feature. In *2019 5th international conference on control, automation and robotics (ICCAR)*, pages 276–280. IEEE, 2019.

[35] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

[36] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.