Machine Learning Approaches for Automated Detection of *Nephrops norvegicus* Burrows in Underwater Surveys

Oscar Papini[1], Enrico Cecapolli[2], Filippo Domenichetti[2], Michela Martinelli[2], Gabriele Pieri[1], Marco Reggiannini[1] and Lorenzo Zacchetti[2,3]

[1] Institute of Information Science and Technologies (ISTI), National Research Council of Italy, Via G. Moruzzi 1, 56124 Pisa, Italy
{oscar.papini,gabriele.pieri,marco.reggiannini}@isti.cnr.it

[2] Institute for Marine Biological Resources and Biotechnology (IRBIM), National Research Council of Italy, Largo Fiera della Pesca 2, 60125 Ancona, Italy
{enrico.cecapolli,lorenzo.zacchetti}@irbim.cnr.it
{filippo.domenichetti,michela.martinelli}@cnr.it

[3] Department of Biological, Geological, and Environmental Sciences (BiGeA), Alma Mater Studiorum – University of Bologna, Piazza di Porta S. Donato 1, 40126 Bologna, Italy

Abstract

This paper presents an analysis of computer vision methods designed to automate the detection, recognition, and classification of *Nephrops norvegicus* burrows in underwater videos. The proposed approach seeks to evaluate the accuracy, minimise human error, and standardise the existing manual video analysis process. By leveraging machine learning techniques, the system described in this paper autonomously processes video streams and identifies *N. norvegicus* burrow openings on the seabed. Additionally, this study investigates data augmentation algorithms to expand an annotated dataset and evaluates the performances of the first results under different configurations.

Keywords: Deep learning · Video annotation · Dataset augmentation · Fisheries · *Nephrops norvegicus*

1 Introduction

Nephrops norvegicus (Norway lobster) is a benthic burrowing decapod crustacean that typically lives at depths between 50 m and 800 m on the continental shelf of the northeastern Atlantic Ocean and the Mediterranean Sea [3]. This species greatly contributes to the total fish landings in Europe, with an average of 3700 tonnes per year just in the Mediterranean Sea, even though landings have seen a significant decrease since 2010 [7].

Underwater Television (UWTV) surveys are carried out to evaluate *N. norvegicus* stocks [2, 6]. These consists of the examination of underwater (UW) video footage of the sea floor to count the number of *N. norvegicus* burrows, distinguishing them from other species' burrows and other objects. This gives an estimate of the *N. norvegicus* population under the assumption of "1 burrow = 1 animal" [18], which is considered reliable among experts despite being currently under discussion [1]. Still, the careful visual analysis of hours of UW footage to count burrows is a highly demanding and time-consuming task, often susceptible to human errors. Several workshops have been organized by the International Council for the Exploration of the Sea (ICES) with the purpose of defining standard procedures for video reading [5, 8–10], and a Working Group on *Nephrops* Surveys (WGNEPS) has been established within ICES to coordinate the activities regarding UWTV surveys [12]. For example, a training program has been developed for readers to learn to recognize *N. norvegicus* burrows using reference material and videos.

In this context, an automatic system that detects (and potentially counts) *N. norvegicus* burrows would be an invaluable tool to ease the workload of marine biologists and potentially reduce the impact of human errors and subjectivity. Within the deep underwater environment, such a system should first be able to identify the specific entrances dug by the *N. norvegicus* individuals. This task also implies the rejection of several candidate targets, such as rocks, small sand clouds and shadows, that share similar visual properties with the *N. norvegicus* openings. In a second instance, the tool must be able to recognize a burrow, i.e. a system of two or more entrances connected by underground tunnels.

This paper focuses on the first of the two mentioned steps: it investigates how effective machine learning (ML) techniques—and in particular deep learning algorithms—are in recognizing entrances of *N. norvegicus* burrows in UWTV footage. Some recent works already show promising results [4, 16]. The use of deep neural networks has risen considerably in the last few years in the field of computer vision, with the most diverse applications. In case of UW images and videos, there are a few intrinsic issues to be taken into consideration: indeed the underwater environment alters the optical features, often producing low-quality images that are not suitable to be fed to a ML algorithm; for example the strong light absorption taking place underwater filters out a large portion of the captured light, returning images with prevailing green and blue components; additionally, turbidity strongly affects the overall contrast; finally, the typical depths at which UWTV footage is recorded during surveys require necessarily the use of artificial illumination systems, a further unbalancing factor for brightness and contrast. Given such conditions, the effort spent to create training datasets to be used in ML algorithms becomes of utter importance; on the other hand, the manual annotation of frames to obtain a high number of samples, as demanded by most algorithms, suffers from the same flaws as the burrows counting task: it takes a lot of time, and it is prone to errors.

This works extends the results presented in [17], describing the first outcomes of the training of a state-of-the-art deep neural network, developed for the object detection on a set of UW images, to get some models able to recognize the entrances of *N. norvegicus* burrows. First, two training datasets are described in detail: one of them consists of a set of video frames manually annotated by experts, the other has been automatically generated by “extending” these annotations to the other frames of the video. Then, the employed neural network architecture and the related training configurations are illustrated and examined. The paper continues with a discussion about the predictive performance of the trained models and concludes with some final remarks and perspectives on future work.

2 Experimental Setup

2.1 Dataset

An annotated dataset of *N. norvegicus* burrows images has been prepared starting from UW video recorded during a series of campaigns carried out jointly by the Institute for Marine Biological Resources and Biotechnology of CNR (Italy) and the Institute of Oceanography and Fisheries (Croatia) in the Pomo/Jabuka Pits area in the central Adriatic Sea [11, 14].

The recording equipment consists of a UW high-resolution colour camera (Kongsberg Maritime oe14-372) mounted on a sledge towed through a cable by the research vessel *G. Dallaporta* at a constant speed of approximately 1 kn, thus collecting images of the seafloor with a fixed field of view of 80 cm. The captured footage was saved through an HD/SD digital video recorder (Datavideo HDR-60) with a resolution of 768×576 pixels and a frame rate of 25 FPS. The effective height was reduced to 432 pixels due to the presence of an automatic overlaid timestamp in the upper part of the frame. For the purposes of this work, a segment of about 4 minutes and 12 seconds of length was selected and deinterlaced using the Neural Network Edge Directed Interpolation (NNEDI) filter, and a set of 505 still frames, one every half second, was extracted and given to the annotators.

The annotation process has been performed using Microsoft VoTT [15]. Two object labels were chosen: “Entrance”, to denote the single openings of a *N. norvegicus* burrow on the sea floor, and “Burrow”, to group together the entrances relative to the same system (see Figure 1). In all the experiments presented in this work only objects belonging to the class “Entrance” are considered. In the end, only 156 frames contained at least one object, and the total number of “Entrance” boxes is 315. This dataset will be called D_0.

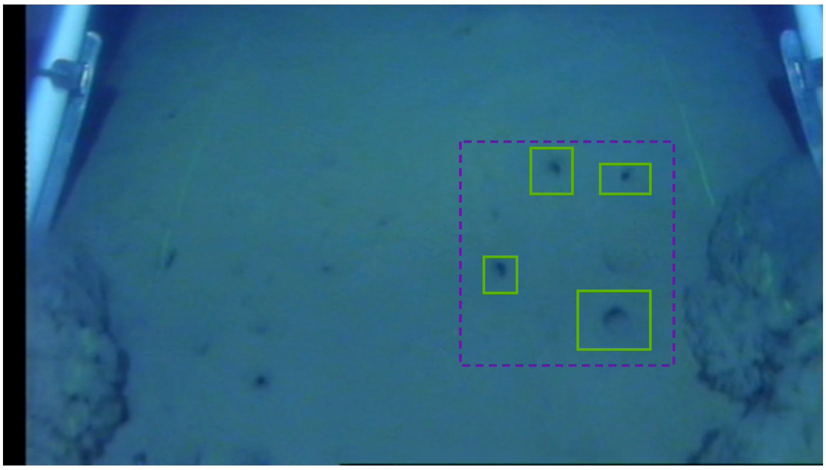

Figure 1: Example of an annotated image in the dataset. The green boxes are “Entrance” objects; the dashed purple box is the “Burrow” object that groups them.

In order to have an increased number of samples to be fed to the ML algorithms, an automatic “boosting” process has been carried out. The premise is that, given that the footage is acquired by a camera mounted on a sledge at a fixed angle and (ideally) moving linearly at constant speed, a target on the seabed persists for a certain number of frames. Thus, the annotation of an entrance in a frame $F \in D_0$ defines a template that can be sought through correlation-based techniques in the temporally neighbouring frames. By applying this method, a new “extended” dataset, called D^+, has been obtained; it contains 2406 images for a total of 4844 “Entrance” objects (see Figure 2).

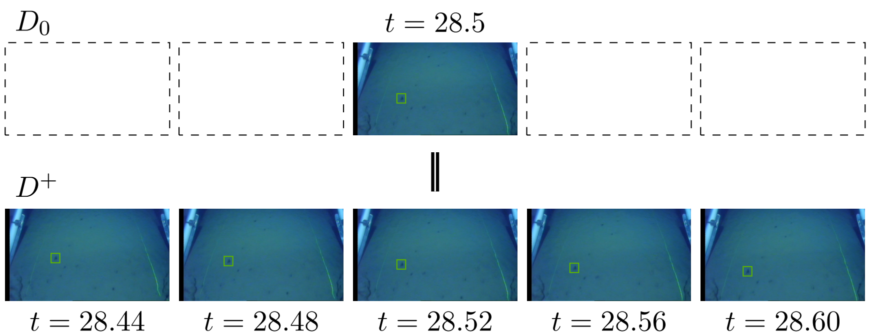

Figure 2: Concept of boosting algorithm. A burrow entrance in a frame of D_0 is used as a template to find the same entrance in neighbouring frames in D^+.

An image in the datasets is uniquely identified by its timestamp, i.e. the number of seconds passed since the beginning of the video. Given the frame rates in play, it was established that images of D_0 have a timestamp with a single digit after the decimal point, that can be either 0 or 5; images of D^+ instead have two digits, which represent an integer divisible by 4. Some examples: 21.5 is a timestamp of an image in D_0; 77.28 is a timestamp of an image in D^+; the frame after 1 minute of the video has timestamp 60.0 as an image in D_0 and timestamp 60.00 as an image in D^+; 34.8 and 154.62 are not valid timestamps. Since 50 is not divisible by 4, there is a slight mismatch between timestamps of D_0 and D^+, where the "half-second" images of D_0 correspond to images in D^+ whose timestamps end with 52. The last images in D_0 and D^+ have timestamps 251.5 and 251.52 respectively.

2.2 Network Training

The datasets described above have been used to train several instances of the YOLOv8 network by Ultralytics [13], a state-of-the-art deep neural network designed for object detection and classification. More precisely, all the experiments have been conducted on a computational cluster equipped with a NVIDIA A100 40 GB PCIe GPU using the "small" variant YOLOv8s.

Both datasets D_0 and D^+ have been split into training and validation sets. In particular, each dataset has been divided into five parts, each containing about 20% of the boxes (using D^+ as a reference), so that a 5-fold validation scheme can be implemented by training five different models using one of those parts as a validation set and the union of the others as a training set, and then averaging the performances. Due to the extension mechanism described in Section 2.1, which is based on the integration of additional frames from the original video that are temporally close to the ones already present in D_0, a random split could significantly bias the results in favour of D^+. Indeed, in this case, the images in the validation set are likely to be very similar to those in the training set. To avoid this, the random split has been replaced by a more controlled approach: each part is populated with frames extracted from strictly separated time intervals. In practice, the whole video has been cut into five segments as indicated in Table 1.

As is usual for object detection tasks, the *average precision* (AP) has been chosen to evaluate the performances of the model during training, with an Intersection over Union (IoU) threshold of 50% (AP@0.5). AP of the validation set is computed automatically after each epoch by the training algorithm.

The following settings are common to all the experiments:

- to comply with the network specifications and considering the available computational resources, the images have been resized to 416×416 pixels;

Table 1: The five parts in which the datasets have been split.

			D_0		D^+	
Fold	Time start	Time end	images	boxes	images	boxes
1	000.00	038.52	38	84	498	968
2	038.56	079.80	26	65	413	969
3	079.84	129.44	30	59	450	970
4	129.48	191.64	31	51	531	972
5	191.68	251.52	31	56	514	965

- the training duration is set to 200 epochs;
- the "patience" value, i.e. the number of epochs after which the training is stopped automatically if no significant improvement is measured, has been disabled;
- all other parameters maintain their default values unless explicitly specified.

Four opposite pairs of configurations have been selected, for a total of 16 possible experimental setups:

- use the Original dataset D_0 (**O**) or the Extended one D^+ (**E**);
- use the RGB images from the video (**C**, from "Colour") or convert them in Grayscale (**G**);
- initialize the network with random weights (**S**, from "Scratch") or use a network pre-Trained on the COCO dataset (**T**);
- disable the data augmentation techniques described in Table 2 (**N**, for "No augmentation"), which are some of the techniques performed by default by the training algorithm of YOLOv8s to improve detection by artificially increasing the variability of the environmental conditions in which the images of a dataset are taken, or enable them (**A**, for "Augmentation").

In the following, each setup is identified with a string of four letters using the ones in bold in the list above. For example, in the setup OCTA the network is initialized with the pre-trained weights and trained on the RGB images of D_0, enabling the standard data augmentation techniques.

3 Results and Discussion

Figure 3 shows the plots of the AP computed during the network training for each experiment. Recall that the AP is a rational number between 0 and 1, and that the higher the AP is, the better performances the model has.

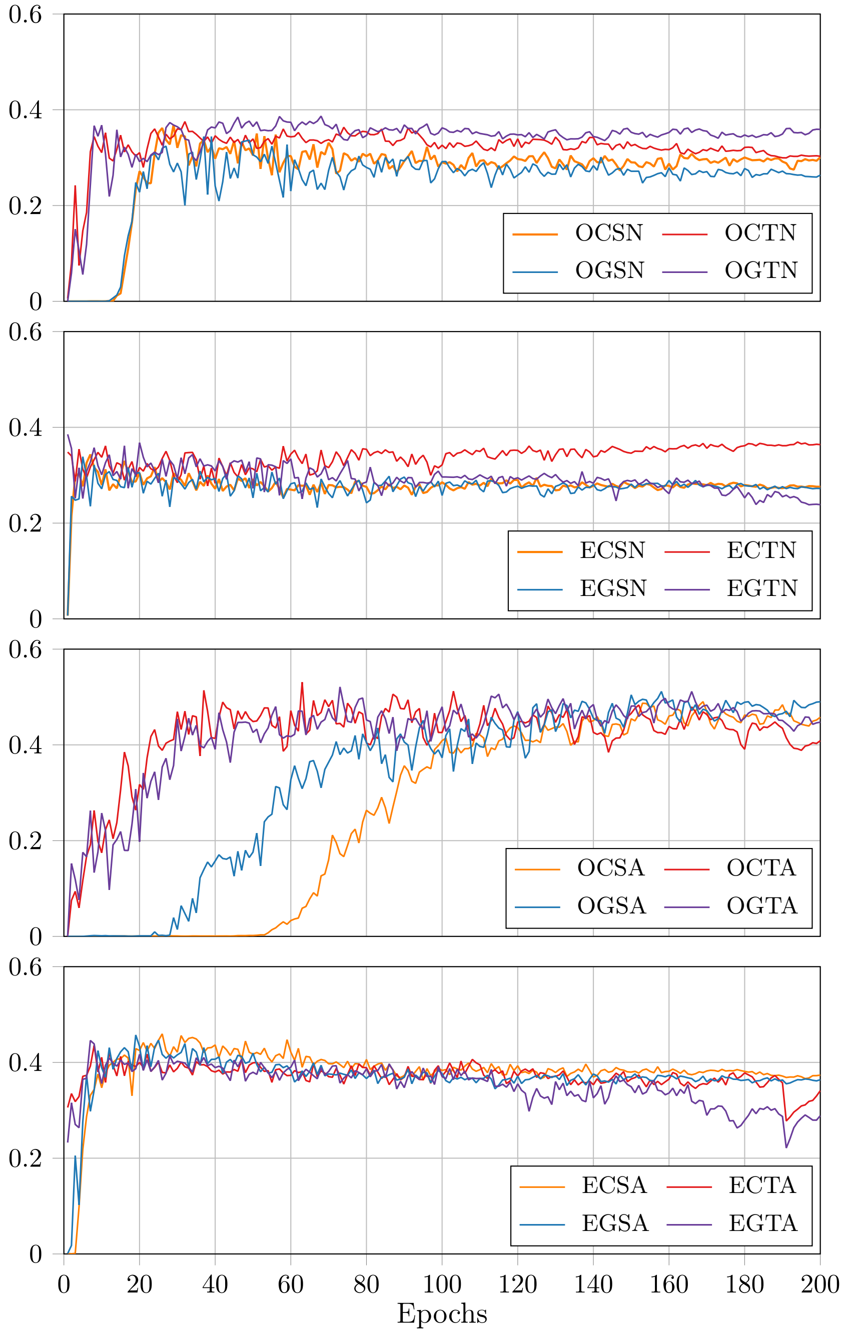

Figure 3: Plots of the AP@0.5 computed on the validation set during the training phase.

Table 2: Some data augmentation techniques applied by default (table adapted from https://docs.ultralytics.com/modes/train/).

Name	Default value	Effect
hsv_h	0.015	Adjusts the hue of the image by a fraction of the colour wheel, introducing colour variability.
hsv_s	0.7	Alters the saturation of the image by a fraction, affecting the intensity of colours. Useful for simulating different environmental conditions.
hsv_v	0.4	Modifies the value (brightness) of the image by a fraction, helping the model to perform well under various lighting conditions.
translate	0.1	Translates the image horizontally and vertically by a fraction of the image size, aiding in learning to detect partially visible objects.
scale	0.5	Scales the image by a gain factor, simulating objects at different distances from the camera.
fliprl	0.5	Flips the image left to right with the specified probability, useful for learning symmetrical objects and increasing dataset diversity.
mosaic	1.0	Combines four training images into one, simulating different scene compositions and object interactions.
erasing	0.4	Randomly erases a portion of the image during classification training, encouraging the model to focus on less obvious features for recognition.

As it is clear from the plots, none of the experimental settings gives rise to a model that performs exceptionally well on the validation set. There are several potential reasons of this behaviour.

In general, the detection task is made more difficult by the nature of the images themselves, as they are extracted from a video recorded in an underwater environment with several drawbacks, such as bad lighting and turbidity; the images quality is also hindered by the technical aspects of the video (low resolution, deinterlacing). Furthermore, as stated in the Introduction, detection of *N. norvegicus* entrances is intrinsically difficult even for trained human eyes—it is worth remembering that YOLO is a general-purpose object detector that behaves very well on "everyday" tasks, but it probably needs some tweaks to reach the same performances on highly specialized tasks such as the one described in this paper.

The performances of the experiments involving D_0 could be explained by the size of the dataset, which contains too few relevant images for the network to learn the discriminating features of *N. norvegicus* entrances. By comparing the experiments that are trained on D_0 (first and third plots of Figure 3) and the ones trained on D^+ (second and fourth plots), one can notice that there is no improvement in the AP, despite the increase by an order of magnitude in the number of training images. This is probably due to the fact that the new images of D^+ are actually too similar to the ones already present in D_0, so adding them does not in fact provide substantially "new" information that the model can learn—rather, it increases the risk of overfitting.

From these first results, it appears that the configurations providing the best AP, albeit slightly, are those using the dataset D_0 and the standard augmentation techniques (third plot of Figure 3). On the other hand, the choices regarding the colour space of the input images and the network weights initialization don't seem to affect the outcomes. In any case, further investigations on this behaviour will be carried out in the future.

4 Conclusion

This work describes the first steps to tackle the problem of detecting *N. norvegicus* burrows in UW video footage using ML techniques. Considering the extraordinary results achieved by deep neural networks in the object detection domain in recent years, it is natural to ask whether they can also be successfully applied in this very specific field. A positive answer will prove extremely useful in helping marine biologists who currently spend a large part of their work time in a tedious and error-prone activity such as manually reviewing hours of video footage to count *N. norvegicus* burrows in it.

The application of an already available general-purpose network, without further tweaking, to a peculiar setting such as images extracted from UW video footage did not obtain particularly good performances. This is not completely unexpected; nevertheless, these results establish a good starting point and give us the motivation to continue pursuing this path of research.

Future work on the subject will encompass further attempts with different settings and more ground truth datasets, including the additional development of scripts that can generate large datasets starting from smaller, manually annotated ones. Also, a comparative analysis of the performances of other deep neural network architectures will be carried out, with the objective to eventually reach the best setup for an automatic system able to detect *N. norvegicus* burrow entrances.

Acknowledgments

The authors express gratitude to Andrea Belardinelli, Pierluigi Penna and Paolo Scarpini of IRBIM-CNR for their invaluable support in the design and maintenance of the instrumentation, the data recording and its digitalization.

Funding

This work is part of a project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 101000825 – NAUTILOS Project (https://www.nautilos-h2020.eu).

Disclosure of Interests

The authors of this work declare that they have no conflicts of interest.

References

[1] Jacopo Aguzzi et al. "Burrow emergence rhythms of *Nephrops norvegicus* by UWTV and surveying biases". In: *Scientific Reports* 11, 5797 (2021). DOI: 10.1038/s41598-021-85240-3.

[2] Jacopo Aguzzi et al. "Advancing fishery-independent stock assessments for the Norway lobster (*Nephrops norvegicus*) with new monitoring technologies". In: *Frontiers in Marine Science* 9 (2022). DOI: 10.3389/fmars.2022.969071.

[3] Mike C. Bell, Frank Redant and Ian Tuck. "*Nephrops* Species". In: *Lobsters: Biology, Management, Aquaculture and Fisheries*. Ed. by Bruce F. Phillips. Blackwell Publishing Ltd, 2006. Chap. 13, pp. 412–461. DOI: 10.1002/9780470995969.ch13.

[4] Antoni Burguera Burguera, Francisco Bonin-Font, Damianos Chatzievangelou, Maria Vigo Fernandez and Jacopo Aguzzi. "Deep learning for detection and counting of *Nephrops norvegicus* from underwater videos". In: *ICES Journal of Marine Science* 81.7 (2024), pp. 1307–1324. DOI: 10.1093/icesjms/fsae089.

[5] Neil Campbell, Helen Dobby and Nick Bailey. "Investigating and mitigating uncertainties in the assessment of Scottish *Nephrops norvegicus* populations using simulated underwater television data". In: *ICES Journal of Marine Science* 66.4 (2009), pp. 646–655. DOI: 10.1093/icesjms/fsp046.

[6] Helen Dobby et al. "ICES Survey Protocols – Manual for *Nephrops* Underwater TV Surveys, coordinated under ICES Working Group on *Nephrops* Surveys (WGNEPS)". In: *ICES Techniques in Marine Environmental Science* (May 2021). DOI: 10.17895/ices.pub.8014.

[7] FAO Fisheries Division, Statistics and Information Branch. *FishStatJ: Software for Fishery and Aquaculture Statistical Time Series*. Accessed on September 18, 2024. URL: https://www.fao.org/fishery/en/statistics/software/fishstatj.

[8] ICES. *Report of the Workshop and training course on* Nephrops *burrow identification (WKNEPHBID)*. ICES CM 2008/LRC:03. 25–29 February 2008, Belfast, Northern Ireland, UK, 2008. 44 pp. DOI: 10.17895/ices.pub.9828.

[9] ICES. *Report of the Workshop on* Nephrops *Burrow Counting (WKNEPS)*. ICES CM 2016/SSGIEOM:34. 9–11 November 2016, Reykjavík, Iceland, 2017. 62 pp. DOI: 10.17895/ices.pub.8651.

[10] ICES. *Report of the Workshop on* Nephrops *Burrow Counting (WKNEPS)*. ICES CM 2018/EOSG:25. 2–5 October 2018, Aberdeen, UK, 2019. 44 pp. DOI: 10.17895/ices.pub.8180.

[11] ICES. *Working Group on* Nephrops *Surveys (WGNEPS; outputs from 2021 meeting)*. ICES Scientific Reports 4:29. 2022. 183 pp. DOI: 10.17895/ices.pub.19438472.

[12] ICES. *Working Group on* Nephrops *Surveys (WGNEPS; outputs from 2022 meeting)*. ICES Scientific Reports 5:26. 2023. 125 pp. DOI: 10.17895/ices.pub.22211161.v1.

[13] Glenn Jocher, Ayush Chaurasia and Jing Qiu. *Ultralytics YOLO*. Version 8.2.60. 2023. URL: https://github.com/ultralytics/ultralytics.

[14] Michela Martinelli et al. "Towed underwater television towards the quantification of Norway lobster, squat lobsters and sea pens in the Adriatic Sea". In: *Acta Adriatica* 54.1 (2013), pp. 3–12.

[15] Microsoft CSE. *Visual Object Tagging Tool (VoTT)*. Version 2.2.0. 2020. URL: https://github.com/microsoft/VoTT.

[16] Atif Naseer. "Advanced Detections of Norway Lobster (Nephrops Norvegicus) Burrows using Deep Learning Techniques". In: *International Journal of Advanced Computer Science and Applications* 14.3 (2023), pp. 493–499. DOI: 10.14569/IJACSA.2023.0140357.

[17] Oscar Papini et al. "Advancing Automated Detection of *Nephrops norvegicus* Burrows in Underwater Television Surveys Through Machine Learning". In: *Pattern Recognition and Image Analysis* 34.4 (2024). To be published.

[18] Francisco Sardà and Jacopo Aguzzi. "A review of burrow counting as an alternative to other typical methods of assessment of Norway lobster populations". In: *Reviews in Fish Biology and Fisheries* 22 (2012), pp. 409–422. DOI: 10.1007/s11160-011-9242-6.

This preprint has not undergone peer review or any post-submission improvements or corrections. The Version of Record of this contribution is published in *Pattern Recognition. ICPR 2024 International Workshops and Challenges. Kolkata, India, December 1, 2024. Proceedings, Part III*, LNCS 15616, and is available online at https://doi.org/10.1007/978-3-031-87663-9_24