IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 19 January 2024, accepted 7 March 2024, date of publication 9 April 2024, date of current version 17 April 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3386640

RESEARCH ARTICLE

On Automatic Person-in-Water Detection for Marine Search and Rescue Operations

JUSSI TAIPALMAA[1], JENNI RAITOHARJU[2], (Senior Member, IEEE), JORGE PEÑA QUERALTA[3,4], (Member, IEEE), TOMI WESTERLUND[3], (Senior Member, IEEE), AND MONCEF GABBOUJ[1], (Fellow, IEEE)

[1]Department of Computing Sciences, Tampere University, 33014 Tampere, Finland
[2]Faculty of Information Technology, University of Jyväskylä, 40014 Jyväskylä, Finland
[3]Turku Intelligent Embedded and Robotic Systems Group, University of Turku, 20014 Turku, Finland
[4]SCAI Laboratory, SPZ, Swiss Federal School of Technology in Zürich (ETH Zürich), 8092 Zürich, Switzerland

Corresponding author: Jussi Taipalmaa (jussi.taipalmaa@tuni.fi)

This work was supported by the Academy of Finland's AutoSOS Project under Grant 328755.

ABSTRACT In marine search and rescue missions, the objective is to find a missing person in the water. Time is a critical factor in the identification of the missing person, as any delay in locating them can have life-threatening consequences. Autonomous unmanned aerial vehicles (UAVs) possess the potential to help in the search task by providing a bird's-eye view helping to cover larger areas faster. Therefore, it is very important that UAVs can efficiently and accurately detect persons in the water. This work studies automatic person detection in the water from a UAV. We performed experiments on both lakes and sea near Turku, Finland, and captured videos of people in the water from various altitudes and different viewing angles. Our person-in-water detection tests focus on important factors that have not received sufficient attention in prior studies: evaluation metrics and detection thresholds, the impact and use of different bounding box sizes, multi-frame detection and performance in unseen environments. We provide analysis of the suitability of different approaches for the person detection task and we also publish our training and testing data that includes over 72000 frames. To the best of our knowledge, this is the largest publicly available person-in-water detection dataset.

INDEX TERMS Search and rescue (SAR), person-in-water, unmanned aerial vehicle (UAV), object detection, deep learning (DL), dataset.

I. INTRODUCTION

Marine search and rescue (SAR) missions consist of finding the emergency site on sea or lake, locating the target, and performing the rescue operation. SAR operations in the marine environment can be challenging due to the vast and often unpredictable nature of the sea or lake area. The operations are also highly time-critical because a person overboard can stay above the surface for only a limited time. The actual time depends on different factors, such as weather or fitness, current condition or equipment of the target person, but ultimately hypothermia begins when the human body core temperature drops below $35°C$ [1]. Depending on the water temperature and clothing (e.g., drysuit vs. ordinary clothes), the onset of hypothermia may differ from hours to only minutes [2].

Recent studies have shown that SAR operations can significantly benefit from supporting autonomous or tele-operated robots and multi-robot systems [3], [4], [5]. For marine SAR operations, unmanned aerial vehicles (UAVs)

The associate editor coordinating the review of this manuscript and approving it for publication was Yangmin Li.

can offer an efficient solution by covering the search area and helping to detect the target to be rescued. In the future, an efficient solution could be that a UAV or a swarm of UAVs autonomously perform area coverage above the target area and use computer vision to find the target; then, the rescue personnel can perform the actual rescue mission.

During the last decade, the number of applications using computer vision algorithms and also the performance of the algorithms have advanced significantly. With the rapid development of deep learning networks for object detection tasks, the performance of object detectors has greatly improved [6]. This suggests that deep learning-based object detection can be used in SAR missions. While some object detection methods can already surpass human performance in accuracy [7], some solutions can be rather slow and computationally heavy and the performance measured in run-time can differ a lot. In SAR operations, it is vital that the selected algorithms can run as close to real-time as possible on portable devices while still working with a high level of accuracy. The faster the algorithm can operate, the faster the UAV can search the area, which in turn can lead to a faster rescue of the persons in distress. A high level of performance of the algorithm assures that no important information is missed.

In this paper, we study different aspects of deep learning-based object detection in SAR missions. We focus on important factors that have not received sufficient attention in prior studies: (i) evaluation metrics and detection thresholds, (ii) the impact and use of different bounding box sizes, (iii) multi-frame detection, and (iv) performance in unseen environments. Based on our experiments we provide recommendations for further studies on the topic. For this purpose, we have collected, annotated, and processed a dataset of over 72000 frames which contains images of a person swimming or floating in a water body, taken from a UAV flying above the area. To the best of our knowledge, this is currently the largest dataset on this topic and we will make it publicly available.

The rest of the paper is organized as follows. Section II briefly reviews related research. Section III describes the data acquisition process and the collected dataset. In Section IV, we introduce the study setup and the design factors considered in this study. Experimental results are presented in Section V. Finally, Section VI concludes the paper.

II. RELATED WORK

In recent years, a lot of research has been conducted towards enabling autonomous drones to assist in SAR operations as shown by recent reviews on related topics [3], [8], [9], [10], [11], [12]. In this paper, we focus on the perception needed in marine search and rescue operations, specifically when searching for a person in the water.

The person-in-water detection in SAR missions can be performed either using object detection or semantic segmentation. A survey of available deep learning semantic segmentation techniques [13], provides an extensive view of different methods to perform the segmentation. For autonomous driving, semantic segmentation has been studied fairly well [14], but for marine environments, the studies have been less common and have been focusing on a view from a surface vessel. In [15], three commonly used state-of-the-art deep learning semantic segmentation methods (U-Net [16], PSP-Net [17], and DeepLabv2 [18]) were benchmarked for obstacle detection on a marine environment. The leaderboard for a publicly available dataset, Modd2 [19], which contains images of swimmers and rowers among other obstacles, lists semantic segmentation methods capable of performing in marine environments [17], [18], [19], [20], [21], [22], [23], [24]. In [25], a dataset for surface vessel water segmentation was introduced, and in [26], the same data was used to train a segmentation model for a UAV view with swimmers and other obstacles in the water. While excellent results can be obtained when the algorithm is applied in conditions and environments that resemble the training images, it was observed that the performance decreases notably in different conditions. This further highlights the need for diverse training images and domain adaption techniques that help to adjust to unseen conditions [27].

A drawback of semantic segmentation is that it requires pixel-level analysis of the images and may waste precious time for analysing areas of lower interest, such as ground locations. Object detection, on the other hand, only requires region-level image analysis and can lead to faster detection of the person in water. A survey of deep learning-based object detection [6] has been published quite recently. These tasks usually still require high computing power and memory for real-time applications. Therefore, cloud computing or small-sized object detection methods have been used for UAV applications. In [28], cloud computing was used for object detection while performing low-level object detection and navigation on a UAV. Cloud computing assists the system with high computing power and memory, but communication with a cloud server can bring unpredictable delays. An alternative to cloud computing is to rely on specific object detection models, specifically designed for limited computational power and memory.

In [29], frame rates of 5-18 fps were obtained on different lightweight embedded processing platforms when running a lightweight object detection model for vehicle detection. In [30], an approach for learning efficient deep object detectors for real-time UAV applications through channel pruning of convolutional layers was proposed in [30]. The approach was tested on different generic object classes. In [31], another lightweight object detection model was tested on a dataset including people and objects in water. In [32], the authors proposed an adaptive submodularity and deep learning-based spatial search method for detecting humans with a UAV. In [33], the authors performed human detection for SAR operations using images from different in-land environments.

Recently, also person-in-water detection has received more attention. SeaDroneSee dataset [34] includes 54,000 frames with humans in open water captured with drones from various altitudes and viewing angles ranging from 5 to 260 meters and 0 to 90 degrees while providing the respective meta information for altitude, viewing angle and other metadata. A Maritime Computer Vision Challenge [35] included tasks for UAV-based maritime object detection and tracking. An approach for obtaining fast region-of-interest proposals in a video stream on an embedded GPU for maritime human detection tasks was proposed in [36].

III. DATA DESCRIPTION

We collected our dataset at four different water bodies in the Turku area, Southwest Finland, in summer and early autumn weather. Specifically, the data was collected between early June and mid-September 2020. The dataset consists of videos from different environments, sea and lakes, recorded with a drone flying over the area at various heights. All dataset images/frames contain a similar structure, an image of a water area from above, with at least one person swimming in the water. The camera angle is set to face straight downwards, i.e., the camera pitch is always -90°. The only changing factor in the setup is the flight altitude of the drone. All people appearing in videos are part of the study group and their informed consent was obtained.

The data collection was carried out with a small drone equipped with an RGB camera with a rolling shutter. The camera has a fixed focal length of 24 mm (35 mm format equivalent) with a field of view of 83° and an aperture f/2.8. The still images are taken with a resolution of 9 MP (4000 × 2250 pixels), while the videos in the datasets are recorded in FullHD resolution (1920 × 1080 pixels) at a rate of 60 fps.

From the videos, we extract each frame. The different locations, their environment type, the number of recorded videos, and the overall length of the videos are shown in Table 1. The dataset contains videos of different types of water bodies with various turbidity levels and swimmers with different swimwear. The weather varies from sunny to overcast, which affects the illumination,e.g., the color of the water, and different wind speeds affect to the texture of the water. The maximum height recorded is 120 m owing to the limitations imposed by EU-level drone regulations.

TABLE 1. Dataset collection sites, environment types, number of video clips, and combined length of the videos collected at each site.

Collection site	Environment type	Videos (#)	Length (time)
Maaria	Lake	7	09:47
Masku	Lake	4	08:51
Mustfinn	Sea (coastal)	19	04:27
Littoinen	Lake (shallow)	1	02:13

The different subsets, listed in Table 1, contain the captured frames and locations of the bounding boxes of the swimmer. The annotations were made by using the Labelbox-annotation tool [37]. The tool allows the interpolation of the bounding boxes in frames that are between hand-annotated frames.

The videos for training and testing were selected from Maaria, Masku and Mustfinn subsets. The Littoinen subset was left outside, so it could be used later for testing as a completely unseen and different environment. The training and testing splits were determined in a way that we use the frames of a single video clip either for training or testing. This way we can ensure that consecutive frames are not used for both training and testing, and we can also reduce the possibility of over-fitting. The splits are described in Table 2 and Table 3.

TABLE 2. List of subsets, number of videos and images used for training.

Dataset	Videos (#)	Images (#)
Maaria	6	22566
Masku	3	14703
Mustfinn	12	11021
Total	21	48290

TABLE 3. List of subsets, number of videos and images used for testing.

Dataset	Videos (#)	Images (#)
Maaria	1	12486
Masku	1	7234
Mustfinn	6	4107
Total	8	23827

This gives us an overall training/testing split of 67/33. All the images contain one object that we are trying to detect, a swimmer in the water. There are also other objects such as rock, vegetation, beach equipment and birds in the water, but we are not trying to detect them and they have not been annotated. This setup makes this a one-class detection problem.

We released the dataset:

SAR-HumanDetection-FinlandProper, along with the partitions used for training and testing. To the best of our knowledge, our dataset is the largest publicly available dataset for person-in-water detection in terms of the number of frames. The dataset can be found in the following link: https://doi.org/10.23729/9b3fcb5d-9655-4c62-9762-7442040f7579

IV. STUDY DESIGN

Our purpose is to study different design factors in deep learning-based UAV person-in-water detection in SAR missions. The considered factors include evaluation metrics and detection thresholds (Section IV-B), the impact and use of different bounding box sizes (Section IV-C1), multi-frame detection (Section IV-D), and performance in unseen environments (Section IV-E).

In this study, we do not compare different network models or try to find a perfect hyperparameter setup, but focus

on the above-mentioned factors that are important for the task at hand, but typically receive less attention. Therefore, we selected a commonly-used baseline model and setup that is used in all the experiments as described in Section IV-A.

A. BASELINE MODEL AND SETUP

As our baseline model we selected the widely-used YOLOv4-model [38] with MobileNetV3Small backbone. We used a general YOLOv4/v3/v2 object detection pipeline, implemented with keras and tensorflow.[1] YOLOv3 -loss function was used, and the Adam algorithm was used for optimization. All the experiments were performed using NVIDIA GeForce RTX 2080 GPU with 8GB own memory.

The drone equipment we used allowed us to gather high-quality (1080 × 1920 pixels) videos. YOLO models require the input size to be a multiple of 32. Therefore, we first downsampled our original data from 1080 × 1920 pixels resolution initially to 1024 × 1920 pixels. We made experiments with this input size and input size downsampled by a factor of 2, but these models turned out to be computationally too heavy for our use case and equipment. As the problem is highly time-critical and the model is planned to be used in an edge device, we decided to compromise between the output frame rate and input size. Therefore, we decided to use an input size that corresponds to downsampling by a factor of 4, i.e., 256 × 480 pixels.

With this reduced input size, it is still possible to detect objects in the water. But since a portion of the videos are captured from a relatively high altitude, up to 120 m, the loss of detail during downsampling is quite significant as can be noticed in Figure 1. This makes detection very difficult from high altitudes and other methods for solving this problem need to be considered as will be further discussed in Section IV-C.

FIGURE 1. Images from low and high altitude, before (upper row) and after (lower row) downsampling. The low resolution images have been scaled back to the same size for easier comparison, and all the images have been cropped for better representation.

The training was done in the batches of 8 images and the training/validation split was 90/10. The Model was pre-trained with Imagenet [39] and transfer learning was then used to train the model with our train dataset described in Table 2. The output layers were first trained for 10 epochs, after which the backbone was unfrozen and the model continued to train for a maximum of 240 more epochs. The learning rate of 0.001 was used. For most experiments, testing was performed using the test set described in Table 3. Only when experimenting with unseen environments, the Littoinen subset was used instead.

[1] github.com/david8862/keras-YOLOv3-model-set

B. EVALUATION CRITERIA AND DETECTION THRESHOLDS

Object detection algorithms output bounding boxes with different confidence scores. A detection is considered *true positive* (TP) when the confidence is high enough and the detected bounding box coordinates are close enough to the ground-truth bounding box coordinates. To this end, thresholds for the confidence level and for *intersection over union* (IoU) between the detected and ground-truth bounding boxes are typically used. If a detected bounding box does not correspond to any ground-truth bounding box, it is a *false positive* (FP) detection. A ground truth bounding box that is not detected at all is a *false negative* (FN). Different object detection evaluation metrics can be computed based on the numbers of TPs, FPs, and FNs.

For given IoU and confidence thresholds, precision (P) and recall (R) are defined as

$$P = \frac{TP}{TP + FP}, \qquad R = \frac{TP}{TP + FN}. \tag{1}$$

Too strict confidence thresholds lead to a situation where only a few bounding boxes are detected. If the detections are correct, this can yield perfect precision, while recall is low due to many FNs. Lower thresholds lead to many detected bounding boxes. The number of FNs becomes low, but also the number of FPs typically increases, which yields high recall values but low precision. Object detection algorithms need to balance between optimizing precision and recall by selecting suitable thresholds.

Object detection algorithms are most commonly evaluated using different variants of Average Precision (AP) metric [40], [41]. The most common variant selects an IoU threshold (typically 0.5) and then computes precision and recall values for detections arranged using their confidence values. Average Precision (AP) with the all-point interpolation [41] is computed as

$$\begin{aligned} mAP &= \sum_{n} (R_{n+1} - R_n) P_{interp}(R_{n+1}) \\ P_{interp}(R_{n+1}) &= \max_{\hat{R}:\hat{R} \geq R_{n+1}} P(\hat{R}), \end{aligned} \tag{2}$$

where $P_{interp}(R_{n+1})$ denotes the maximum precision for which the corresponding recall value is greater than R_{n+1}, and thus the metric is computed for a single IoU threshold but all possible confidence thresholds. Another commonly used variant, AP@50:5:95, computes the average over 10 different IoU metrics ranging from 0.5 to 0.95. Computing the results over multiple thresholds measures the ability of

the algorithms to yield good results for a range of user requirements instead of studying which approach is optimal for specific requirements.

In SAR missions, the most critical errors occur if a person in the water is completely missed. Therefore, the number of FNs should be as close to zero as possible. False detections (FPs) can be relatively easily checked by the rescue personnel assuming that the number of FPs is not overwhelmingly large. Therefore, compared to many other object detection tasks, the importance of minimizing the number of FNs is emphasized, while FPs are less critical.

Here, it should be noted that the roles of IoU and confidence thresholds are significantly different. The IoU threshold does not impact the detected bounding boxes, only whether they are considered TPs or FPs. If a detected bounding box partially overlaps with a ground truth bounding box is considered FP, and the corresponding object is considered undetected (FN). Thus, a smaller IoU threshold increases the number of TPs and typically also decreases the number of FNs. Nevertheless, the actual detection output (bounding boxes) do not change. The main question is which labels (TP/FP) for the detected bounding boxes lead to the most meaningful performance evaluation for the task at hand.

In person-in-water detection, bounding boxes are typically small compared to the overall image. Furthermore, minor inaccuracy in the bounding box coordinates is not a problem as the people are expected to float around anyway and the main goal is to help the rescue personnel to find the person in distress. Therefore, we argue that smaller IoU thresholds can lead to more meaningful evaluations for person-in-water detection.

The impact of the confidence threshold can be much more dramatic. The threshold defines, which bounding boxes are given as the detection output. Tightening the confidence threshold can change a TP detection to an FN (or erase an FP detection). As mentioned above, in person-in-water detection tasks the FN can be deadly, while FPs can be handled to some extent. Therefore, we believe that focusing on low confidence thresholds is reasonable.

While existing approaches in object detection for SAR use AP or AP@50:5:95 metrics [34], [35] for performance evaluation, we argue that this may not be the most meaningful choice. Instead of averaging over different confidence and IoU thresholds, we are more interested in thresholds minimizing the number of FNs. In this article, we will study the impact of these thresholds (along with other factors) in person-in-water detection.

As an additional metric in our studies, we use F_1-score which considers both precision and recall for single IoU and confidence thresholds as

$$F_1 = 2 \cdot \frac{P \cdot R}{P + R}. \tag{3}$$

C. OBJECT SIZE

A critical factor in the design of a SAR mission is the flying altitude (distance to the target), which directly affects the perceived size of the person in the water. By flying too high, the risk of not detecting the person in the water is increased, while lower altitudes mean that more critical time is needed to cover the search area. Naturally, the optimal altitude is affected by the camera equipment, resolution, environment, weather, clothing and other possible equipment of the person in the water. As described above, the models' input size may require downscaling, which further affects the detection probabilities. No generic rules can be given, but study the importance of the perceived target size in our use case in different ways as described in Sections IV-C1-IV-C3.

1) BOUNDING BOX SIZE

Our data does not contain information on the distance to the object to be detected. However, the target bounding box sizes serve as a rough proxy. We compare detection rates for bounding box sizes 1-25, 24-50, 51-100, and 100+ pixels. Furthermore, we study the impact of using larger (zoomed-in) bounding box sizes in training and testing as described below.

2) ZOOMING-BASED DATA AUGMENTATION IN TRAINING PHASE

Even though a lot of information is lost because of the need for downsampling from 1080 × 1920 pixels to 256 × 480 pixels, it is still possible to obtain more accurate images of the swimmers for training the model. To this end, we do the following: First, the original image is converted into the size of 1024×1920 pixels. After the conversion, we use a 4×4 grid of 256 × 480 pixels to obtain images that have the same size as the input of the model (Figure 2). From this grid, we select the ones containing an object for training. This allows us to use the portion of the original high-quality images with the model's input size (later called zoomed images) in addition to the downsampled images for training the model. Without using any additional data augmentation or pre-processing, such as flipping images or adding noise to images, we will study how including larger bounding box sizes affects the final performance.

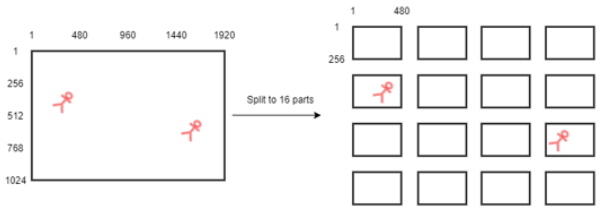

FIGURE 2. Image split into smaller regions during the training phase.

3) GRID ZOOMING IN TEST PHASE

In the test phase, the images can also be separated into smaller images using a 4 × 4 grid (Figure 3), similarly as in the data augmentation phase. The original image is split into 16 sub-images and all the sub-images are fed into the network and

then combined back to the original size. This allows us to use higher-quality images in testing, but the disadvantage is that the network needs to process 16 times the amount of images compared to performing the detection with downsampled images.

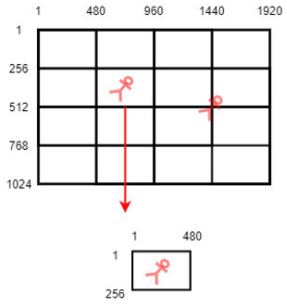

FIGURE 3. Illustration of the original image, and a smaller image obtained with grid-zooming.

In the post-processing phase, the smaller 256 × 480 pixel images need to be stitched back together into the original size (1024 × 1920 pixels). Since the detection can be at the edge of two or more smaller images, there can be multiple detections of a single object. This could be avoided by using more small images, some of which are overlapping, but since the problem is time-critical, adding more input regions would further slow down the overall process. Instead, we merge the multiple detections of a single object using Algorithm 1.

Algorithm 1 Algorithm for Merging Detections Made With Grid-Zooming Method: Two Bounding Boxes are Considered Overlapping if they are Within the *dist* of Each Other. Overlapping Bounding Boxes are Merged. Algorithm Continues Until there are No Overlapping Bounding Boxes

for all found bounding boxes:
Data: $bbox(x_{min}, y_{min}, x_{max}, y_{max})$
$B_1 \leftarrow bbox_1$;
$B_2 \leftarrow bbox_2$;
$dist \leftarrow 10$ pixels;
if *overlap*$(B_1, B_2, dist)$ **then**
  $B_{new}.x_{min} \leftarrow min(B_1.x_{min}, B_2.x_{min})$;
  $B_{new}.y_{min} \leftarrow min(B_1.y_{min}, B_2.y_{min})$;
  $B_{new}.x_{max} \leftarrow max(B_1.x_{max}, B_2.x_{max})$;
  $B_{new}.y_{max} \leftarrow max(B_1.y_{max}, B_2.y_{max})$;
  save B_{new}
else
  save B_1;
  save B_2;
end

D. MULTI-FRAME DETECTION

In SAR tasks, the goal is to find the missing person in the water. It does not matter if the person is detected in every frame where he/she appears or only in sufficiently many frames to alert the rescue personnel. Furthermore, as the drones are taking videos, it is reasonable to assume that despite the movement, any person in water would appear in several frames.

We will study a multi-frame detection scheme, where windows of N frames are used. We consider that we have a TP detection if a person-in-water appearing in the window is detected at least in 1/2 of the frames. We hope that this approach can remove FPs appearing in single frames, while not causing additional FNs. Using the multi-frame scenario we will also study if the frame rate can be lowered without significantly harming the results to make the computations faster.

E. UNSEEN ENVIRONMENT

While all the test images used in our experiments are unseen and selected from different video clips than the training data, the test set used in most experiments still comes from the same water bodies. We will study how the results are affected when the test frames are from a completely unseen environment.

V. EXPERIMENTAL RESULTS

A. EVALUATION METRICS AND DETECTION THRESHOLDS -RESULTS

We first studied the impacts of IoU and confidence thresholds on person-in-water detection. We trained our baseline approach described in Section IV-A and evaluated the model with different IoU-thresholds (0.10, 0.25, 0.50, 0.75) using the confidence threshold 0.1. The AP results are computed using the all-point interpolation over all confidence thresholds. Thus, they are not directly based on the given numbers of TPs, FPs and FNs. The results are given in Table 4. A couple of examples of detection results are shown in Figure 4. Similarly, we evaluated the model with different confidence thresholds (0.1, 0.3, 0.5, 0.7, 0.9) using the IoU threshold 0.1. The results are given in Table 5. Here, we do not give AP values as they are computed over all confidences, not separately for different confidence thresholds.

As expected, in Table 4, the numbers of TPs, FPs, and FNs are all best for the lowest threshold and, thus, also all the other evaluation metrics show the best results for this threshold. The IoU examples in Figure 4 show that it is meaningful to consider the detected bounding boxes as TPs because despite the low IoU they can lead rescue personnel to find the person-in-water.

The confidence threshold has a larger impact on the number of FNs as shown in Table 5. The difference between confidence thresholds 0.1 and 0.9 is over 3-fold. This can be a critical life-saving difference in a SAR task. Furthermore, it can be seen that the precision for the lowest confidence threshold remains relatively high at 0.8880 showing that the number of FPs does not explode too high.

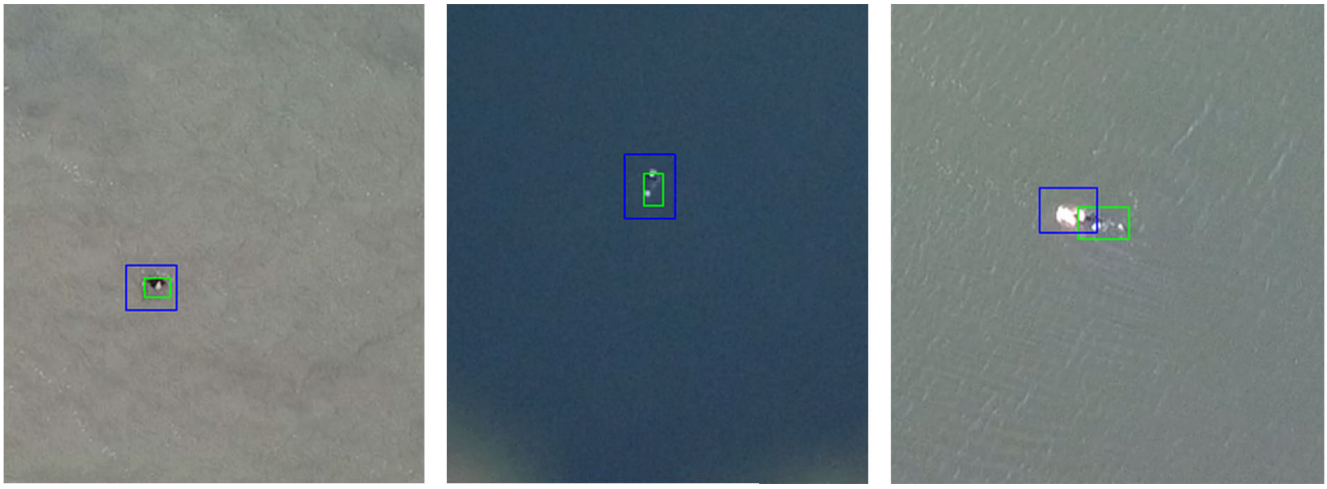

FIGURE 4. Three images from different environments. The green bounding box is the ground truth and the blue bounding box is the detection. All the images have been cropped for better representation. In all images $0.1 \leq IoU < 0.25$.

TABLE 4. Impact of IoU threshold.

Confidence threshold = 0.10, train = non-augmented, test = original								
IoU th.	**Objects**	**TP**	**FP**	**FN**	**Precision**	**Recall**	**F1-score**	**AP***
0.10	23827	**21966**	**2771**	**1861**	**0.8880**	**0.9219**	**0.9046**	**0.9170**
0.25	23827	21683	3054	2144	0.8765	0.9100	0.8930	0.9048
0.50	23827	20996	3741	2831	0.8488	0.8812	0.8647	0.8730
0.75	23827	20559	4178	3268	0.8311	0.8628	0.8467	0.8477

*The AP in computed using the all-point interpolation over all confidence levels.

TABLE 5. Impact of confidence threshold.

IoU threshold = 0.10, train = non-augmented, test = original							
Confidence th.	**Objects**	**TP**	**FP**	**FN**	**Precision**	**Recall**	**F1-score**
0.1	23827	**21966**	2771	**1861**	0.8880	**0.9219**	0.9046
0.3	23827	21567	1505	2260	0.9348	0.9051	0.9197
0.5	23827	21078	889	2749	0.9595	0.8846	**0.9206**
0.7	23827	20028	453	3799	0.9779	0.8406	0.9040
0.9	23827	17896	**70**	5931	**0.9962**	0.7511	0.8564

As a conclusion, these experiments validate our expectations from Section IV-B. Evaluating the person-in-water detection performance in SAR tasks using low IoU thresholds is justified. Due to the high variation of FNs for different confidence thresholds, it is important to focus on the low-confidence results. Using the commonly used AP metric overall confidence can potentially lead to choosing a model that performs better on high confidence, but fails to optimize the most important result with the lowest thresholds. Due to these observations, we will use the IoU and confidence threshold of 0.1 in the following experiments and report only evaluation metrics that are computed for these thresholds leaving out the AP considerations.

B. SIZE OF THE OBJECT

Next, we study the impact of the bounding box sizes in different ways. We first study the different combinations of zooming-based data augmentation in the training phase (Section IV-C2) and the grid-zooming-based evaluation in the test phase (Section IV-C3). The results are given in Table 6. The *non-augmented* training set and the *original* test set denote unmodified data that has been only scaled to the input size of 256 × 480 pixels. The *augmented* training set denotes training data that has been augmented by using the 4 × 4 grid-zoomed images in training in addition to the original downsampled image. The *grid-zoom* test data denotes that the images have gone through the grid-zoom procedure during the test phase.

As we can see from Table 6, using augmented training data gives better results than the original training data for both test set types. The data augmentation does not affect the model's inference speed and is therefore clearly a recommendable approach for person-in-water detection tasks. The described approach will be used in the remaining experiments of this paper. It should be noted that there is potential for further improvement using more varied downscaling factors,

TABLE 6. Impact of zooming-based data augmentation and grid-zooming in test phase.

IoU threshold = 0.10, confidence threshold = 0.10								
Training set	**Test set**	**Objects**	**TP**	**FP**	**FN**	**Precision**	**Recall**	**F1-score**
non-augmented	original	23827	21966	2771	1861	0.8880	0.9219	0.9046
non-augmented	grid-zoom	23827	22154	13109	1673	0.6283	0.9298	0.7498
augmented	original	23827	22437	**2053**	1390	**0.9162**	0.9417	0.9287
augmented	grid-zoom	23827	**23606**	2624	**221**	0.9000	**0.9907**	**0.9432**

TABLE 7. Impact of object size (bounding box area).

IoU threshold = 0.10, confidence threshold = 0.10, train = augmented, test = original							
Bounding box area (pixels)	**Objects**	**TP**	**FP**	**FN**	**Precision**	**Recall**	**F1-score**
1-25	4463	3919	496	544	0.8877	0.8781	0.8829
26-50	8463	7664	1163	799	0.8682	0.9056	0.8865
51-100	5429	5382	385	47	0.9332	0.9913	0.9614
100+	5472	5472	9	0	0.9984	1.0000	0.9992
IoU threshold = 0.10, confidence threshold = 0.10, train = augmented, test = grid-zoom							
Bounding box area (pixels)	**Objects**	**TP**	**FP**	**FN**	**Precision**	**Recall**	**F1-score**
1-25	4463	4305	522	158	0.8919	0.9646	0.9268
26-50	8463	8424	1163	39	0.8787	0.9954	0.9334
51-100	5429	5413	394	16	0.9322	0.9971	0.9635
100+	5472	5464	545	8	0.9093	0.9985	0.9518

TABLE 8. Using multiple consecutive frames in detection.

IoU threshold = 0.10, confidence threshold = 0.10, train = augmented, test = original							
Time window (frames)	**Objects**	**TP**	**FP**	**FN**	**Precision**	**Recall**	**F1-score**
1	22800	21410	2017	1390	0.9139	0.9390	0.9263
10	2280	2151	151	129	0.9344	0.9434	0.9389
20	1140	1076	72	64	0.9373	0.9439	0.9406
50	456	432	28	24	0.9391	0.9474	0.9432
100	228	218	12	10	0.9478	0.9561	0.9520
150	152	146	8	6	0.9481	0.9605	0.9542
300	76	73	5	3	0.9359	0.9605	0.9481
IoU threshold = 0.10, confidence threshold = 0.10, train = augmented, test = grid-zoom							
Time window (frames)	**Objects**	**TP**	**FP**	**FN**	**Precision**	**Recall**	**F1-score**
1	22800	22579	2604	221	0.8966	0.9903	0.9411
10	2280	2262	236	18	0.9055	0.9921	0.9468
20	1140	1132	111	8	0.9107	0.9930	0.9501
50	456	453	43	3	0.9133	0.9934	0.9517
100	228	226	20	2	0.9187	0.9912	0.9536
150	152	152	13	0	0.9212	1.0000	0.9590
300	76	76	8	0	0.9048	1.0000	0.9500

TABLE 9. Impact of dropping frame rate for 300 frame window.

IoU th. = 0.10, confidence th. = 0.10, train = augmented, test = grid-zoomed							
Frame rate	**Objects**	**TP**	**FP**	**FN**	**Precision**	**Recall**	**F1-score**
100 %	76	76	8	0	0.9048	1.0000	0.9500
50 %	76	76	8	0	0.9048	1.0000	0.9500
33 %	76	76	8	0	0.9048	1.0000	0.9500
25 %	76	76	7	0	0.9157	1.0000	0.9560
10 %	76	76	7	0	0.9157	1.0000	0.9560

but we leave further studies on this topic for future work.

Using grid-zooming does not give desirable results when using non-augmented training data. This can be explained by the fact that such an arrangement requires the model to detect objects that are clearly larger than the training objects. Grid-zooming with the augmented training data gives the best results among all options. In particular, the number of FNs is significantly lower than for the other options. Nevertheless, it should remembered that using the grid-zooming option makes the inference 16 times slower, which may not be doable on the edge device. Therefore, studying other ways to improve the inference speed is important.

Next, we study the direct impact of the target bounding box size, which can be interpreted as a proxy for the distance to the target. The higher the altitude, the smaller the target bounding box. The results categorized by the bounding box size are given in Table 7. Unsurprisingly, the results suggest that detection is more difficult from the higher altitudes. The model that does not use grid-zooming performs better at the

TABLE 10. Unseen environment: Littoinen subset.

IoU th. = 0.10, conf. th. = 0.10 train = augmented, test = original							
Time window (frames)	**Objects**	**TP**	**FP**	**FN**	**Precision**	**Recall**	**F1-score**
1	3900	2634	346	1266	0.8839	0.6754	0.7657
10	390	264	20	126	0.9296	0.6769	0.7834
20	195	131	8	64	0.9424	0.6718	0.7844
50	78	53	3	25	0.9464	0.6795	0.7910
100	39	26	1	13	0.9630	0.6667	0.7879
150	26	19	0	7	1.0000	0.7308	0.8444

IoU th. = 0.10, conf. th. = 0.10 train = augmented, test = grid-zoom							
Time window (frames)	**Objects**	**TP**	**FP**	**FN**	**Precision**	**Recall**	**F1-score**
1	3900	3673	3734	227	0.4959	0.9418	0.6497
10	390	376	251	14	0.5997	0.9641	0.7394
20	195	189	127	6	0.5981	0.9692	0.7397
50	78	77	53	1	0.5923	0.9872	0.7404
100	39	39	27	0	0.5909	1.0000	0.7429
150	26	26	18	0	0.5909	1.0000	0.7429

lowest altitude, but when going to higher altitudes the model that uses grid-zooming increases the number of TP detections and greatly reduces the number of FP detections. These results emphasize that the flying altitude must be carefully optimized in SAR tasks to balance between the area coverage speed and the risk of missing a person in the water [42].

C. MULTI-FRAME DETECTION

In Table 8, detections were considered as detections, if they occurred in at least 50% of the frames. We used non-overlapping windows. When the video lengths are not divisible by the window lengths, the excess frames were removed from the end of the videos for all window lengths (i.e., the same frames are included in all the test sets for different window lengths). The ground-truth number of objects is lower for the longer windows as in the multi-frame detection, a single person-in-water appearing in multiple frames is considered as a single object.

From these results, we can notice that observing multiple consecutive frames together increases both the precision and recall of both models tested. It can also be noticed, that with the grid-zoom model, the number of false negative detections decreases as the time window increases, and eventually vanishes. This can be considered an excellent outcome, considering the scope of this research.

The results also indicate that dropping the frame rate in the multi-frame detection scheme may be a viable option. Taking the 300 frames (5 seconds) windows, we consider only every 2,3,4, or 10 frames and compare the results with the original ones in Table 9. We observe that dropping the frame rate even to 10% does not lead to any missed detections. This is a promising outcome considering that time is a critical factor in SAR missions.

D. UNSEEN ENVIRONMENTS

Here we test our model on the Littoinen subset, which was recorded on a different day, different location, and different environment as the training set collected at Maaria, Masku and Mustfinn sites. We use the same time window approach described in the previous section. The results are given in Table 10.

It can be noticed that the performance drops quite drastically. The model that does not use grid-zooming, maintains quite high precision, while the model which uses grid-zooming maintains quite high recall. However, the grid-zoom model detects a high number of FPs, while the other model results in more FNs. It should be also taken into account that the unseen environment is still in the same country and same part of the country. If we moved to a different climate, the results would probably be much worse. The conclusion is not surprising: to guarantee optimal performance in critical SAR tasks, the modes should be trained with datasets including data collected in the intended use environments.

VI. CONCLUSION

In this paper, we studied deep learning-based person-in-water detection in SAR missions using a non-modified YOLOv4 model as our baseline. We collected a large person-in-water dataset using a UAV on Finnish lakes and sea areas. The initial findings from the automatic person-in-water detection system are encouraging for its potential use in SAR missions.

Our focus was on different design factors that have not received much attention in prior studies. We considered different evaluation metrics and detection thresholds. We discussed how false negatives can be much more disastrous than FPs in SAR missions because they can lead to missing the person to be rescued. We showed that using low values of both IoU and detection thresholds is meaningful in the task at hand. As a result, we also recommend avoiding the AP performance metric commonly used in object detection and instead using performance metrics computed for single IoU and confidence thresholds. The main focus should be minimizing the number of FNs while checking that the number of FPs does not explode.

We then studied the impact of target bounding box sizes in different ways. As expected, the results are better for larger bounding box sizes, which highlights the importance of finding a suitable flying altitude and frame resolution. We showed that the results can be improved by including zoomed-in images in the training set. Similarly, using zoomed images in the test phase can improve the results but with additional computational cost.

In SAR tasks, the goal is to find the missing person in the water. It does not matter if the person is detected in every frame where he/she appears. Therefore, we introduced a multi-frame detection scheme where we focus on detecting a person-in-water in different-length windows. We show that this approach can lead to reducing false positive detections while not leading to additional missed targets. We also show that the frame rate in the multi-frame detection scheme can be significantly lowered without harming performance.

Finally, we tested our model in an unseen but similar environment. The results were worse than in the known environments, which highlights the importance of collecting training datasets in as many different environments and weather conditions as possible and training the models with data that includes images collected in the intended application environments.

Our original wish was to use the downsampled images for obtaining fast low-level confidence detection of an object, and then use active zooming with high-quality images to get higher confidence in what we detected. However, it turned out that with the downsampled images, the model does not detect anything when the drone gets to higher altitudes and therefore we could not use the active zooming method. Because of this, we performed the grid zooming. While this is not optimal time-wise, it proved that zooming in can improve the results. In the future, we will further study approaches for faster computation of high-resolution results. The multi-frame detection approach along with lowering the frame rates looks like a promising direction.

REFERENCES

[1] D. J. Brown, H. Brugger, J. Boyd, and P. Paal, "Accidental hypothermia," *New England J. Med.*, vol. 367, no. 20, pp. 1930–1938, 2012.

[2] C. I. Proulx, M. B. Ducharme, and G. P. Kenny, "Effect of water temperature on cooling efficiency during hyperthermia in humans," *J. Appl. Physiol.*, vol. 94, no. 4, pp. 1317–1323, Apr. 2003.

[3] J. P. Queralta, J. Taipalmaa, B. Can Pullinen, V. K. Sarker, T. Nguyen Gia, H. Tenhunen, M. Gabbouj, J. Raitoharju, and T. Westerlund, "Collaborative multi-robot search and rescue: Planning, coordination, perception, and active vision," *IEEE Access*, vol. 8, pp. 191617–191643, 2020.

[4] T. Nguyen, R. Katila, and T. N. Gia, "An advanced Internet-of-Drones system with blockchain for improving quality of service of search and rescue: A feasibility study," *Future Gener. Comput. Syst.*, vol. 140, pp. 36–52, Mar. 2023.

[5] M. M. Z. Shaheen, H. H. Amer, and N. A. Ali, "Robust air-to-air channel model for swarms of drones in search and rescue missions," *IEEE Access*, vol. 11, pp. 68890–68896, 2023.

[6] L. Jiao, F. Zhang, F. Liu, S. Yang, L. Li, Z. Feng, and R. Qu, "A survey of deep learning-based object detection," *IEEE Access*, vol. 7, pp. 128837–128868, 2019.

[7] S. Basu, M. Gupta, P. Rana, P. Gupta, and C. Arora, "Surpassing the human accuracy: Detecting gallbladder cancer from USG images with curriculum learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 20854–20864.

[8] Y. Karaca, M. Cicek, O. Tatli, A. Sahin, S. Pasli, M. F. Beser, and S. Turedi, "The potential use of unmanned aircraft systems (drones) in mountain search and rescue operations," *Amer. J. Emergency Med.*, vol. 36, no. 4, pp. 583–588, Apr. 2018.

[9] E. Kakaletsis, C. Symeonidis, M. Tzelepi, I. Mademlis, A. Tefas, N. Nikolaidis, and I. Pitas, "Computer vision for autonomous UAV flight safety: An overview and a vision-based safe landing pipeline example," *ACM Comput. Surveys*, vol. 54, no. 9, pp. 1–37, Dec. 2022.

[10] L. Lopez-Fuentes, J. van de Weijer, M. González-Hidalgo, H. Skinnemoen, and A. D. Bagdanov, "Review on computer vision techniques in emergency situations," *Multimedia Tools Appl.*, vol. 77, no. 13, pp. 17069–17107, Jul. 2018.

[11] L. Qingqing, J. Taipalmaa, J. P. Queralta, T. N. Gia, M. Gabbouj, H. Tenhunen, J. Raitoharju, and T. Westerlund, "Towards active vision with UAVs in marine search and rescue: Analyzing human detection at variable altitudes," in *Proc. IEEE Int. Symp. Saf., Secur., Rescue Robot. (SSRR)*, Nov. 2020, pp. 65–70.

[12] D. S. Drew, "Multi-agent systems for search and rescue applications," *Current Robot. Rep.*, vol. 2, no. 2, pp. 189–200, Mar. 2021.

[13] F. Lateef and Y. Ruichek, "Survey on semantic segmentation using deep learning techniques," *Neurocomputing*, vol. 338, pp. 321–348, Apr. 2019.

[14] M. Siam, M. Gamal, M. Abdel-Razek, S. Yogamani, M. Jagersand, and H. Zhang, "A comparative study of real-time semantic segmentation for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 700–70010.

[15] B. Bovcon and M. Kristan, "Benchmarking semantic segmentation methods for obstacle detection on a marine environment," in *Proc. Comput. Vis. Winter Workshop*, 2019, pp. 1–10.

[16] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Med. Image Comput. Comput.-Assist. Intervent. (MICCAI)*, 2015, pp. 234–241.

[17] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6230–6239.

[18] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[19] B. Bovcon, J. Muhovič, J. Perš, and M. Kristan, "Stereo obstacle detection for unmanned surface vehicles by IMU-assisted semantic segmentation," *Robot. Auto. Syst.*, vol. 104, pp. 1–13, Jun. 2018.

[20] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder–decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[21] B. Bovcon, R. Mandeljc, J. Pers, and M. Kristan, "Improving vision-based obstacle detection on USV using inertial sensor," in *Proc. Symp. Image Signal Process. Anal.*, 2017, pp. 1–6.

[22] B. Bovcon and M. Kristan, "Obstacle detection for USVs by joint stereo-view semantic segmentation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 5807–5812.

[23] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder–decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 801–818.

[24] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "BiseNet: Bilateral segmentation network for real-time semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 325–341.

[25] J. Taipalmaa, N. Passalis, H. Zhang, M. Gabbouj, and J. Raitoharju, "High-resolution water segmentation for autonomous unmanned surface vehicles: A novel dataset and evaluation," in *Proc. IEEE 29th Int. Workshop Mach. Learn. Signal Process. (MLSP)*, Oct. 2019, pp. 1–6.

[26] J. Taipalmaa, N. Passalis, and J. Raitoharju, "Different color spaces in deep learning-based water segmentation for autonomous marine operations," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2020, pp. 3169–3173.

[27] Y. Zhang, P. David, and B. Gong, "Curriculum domain adaptation for semantic segmentation of urban scenes," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2039–2049.

[28] J. Lee, J. Wang, D. Crandall, S. Šabanovic, and G. Fox, "Real-time, cloud-based object detection for unmanned aerial vehicles," in *Proc. 1st IEEE Int. Conf. Robotic Comput. (IRC)*, Apr. 2017, pp. 36–43.
[29] C. Kyrkou, G. Plastiras, T. Theocharides, S. I. Venieris, and C.-S. Bouganis, "DroNet: Efficient convolutional neural network detector for real-time UAV applications," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2018, pp. 967–972.
[30] P. Zhang, Y. Zhong, and X. Li, "SlimYOLOv3: Narrower, faster and better for real-time UAV applications," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 37–45.
[31] X. Zhang, C. Hao, H. Lu, J. Li, Y. Li, Y. Fan, K. Rupnow, J. Xiong, T. Huang, H. Shi, W.-M. Hwu, and D. Chen, "SkyNet: A champion model for DAC-SDC on low power object detection," 2019, *arXiv:1906.10327*.
[32] Y.-C. Tsai, B.-X. Lu, and K.-S. Tseng, "Spatial search via adaptive submodularity and deep learning," in *Proc. IEEE Int. Symp. Saf., Secur., Rescue Robot. (SSRR)*, Sep. 2019, pp. 112–113.
[33] S. Caputo, G. Castellano, F. Greco, C. Mencar, N. Petti, and G. Vessio, "Human detection in drone images using YOLO for search-and-rescue operations," in *Proc. Adv. Artif. Intell.*, 2022, pp. 326–337.
[34] L. A. Varga, B. Kiefer, M. Messmer, and A. Zell, "SeaDronesSee: A maritime benchmark for detecting humans in open water," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2022, pp. 3686–3696.
[35] B. Kiefer, M. Kristan, J. Perš, L. Žust, F. Poiesi, F. Andrade, A. Bernardino, M. Dawkins, J. Raitoharju, and Y. Quan, "1st workshop on maritime computer vision (MaCVi) 2023: Challenge results," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. Workshops (WACVW)*, Jan. 2023, pp. 265–302.
[36] B. Kiefer and A. Zell, "Fast region of interest proposals on maritime UAVs," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 3317–3324.
[37] (2022). *Labelbox*. [Online]. Available: https://labelbox.com
[38] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.
[39] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[40] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.
[41] R. Padilla, S. L. Netto, and E. A. B. da Silva, "A survey on performance metrics for object-detection algorithms," in *Proc. Int. Conf. Syst., Signals Image Process. (IWSSIP)*, Jul. 2020, pp. 237–242.
[42] P. N. Thuan, J. P. Queralta, and T. Westerlund, "Simulation analysis of exploration strategies and UAV planning for search and rescue," *New Developments and Environmental Applications of Drones*. Berlin, Germany: Springer, 2023, pp. 75–84.

JUSSI TAIPALMAA received the B.Sc. degree in signal processing and multimedia and the M.Sc. degree in pervasive systems from Tampere University of Technology, in 2013 and 2016, respectively. He is currently pursuing the Ph.D. degree with the Signal Analysis and Machine Intelligence Group, Tampere University. Since 2017, he has been a Researcher with the Signal Analysis and Machine Intelligence Group, Tampere University. His research interests include machine learning, computer vision, and biomedical signals along with software applications in autonomous and learning systems.

JENNI RAITOHARJU (Senior Member, IEEE) received the Ph.D. degree in information technology from Tampere University of Technology, in 2017. She is currently an Assistant Professor of signal processing with the University of Jyväskylä, Finland, and a part-time Senior Research Scientist with the Finnish Environment Institute. Her research interests include machine learning and pattern recognition methods along with their application in environmental monitoring and autonomous systems.

JORGE PEÑA QUERALTA (Member, IEEE) received the B.Sc. degree in mathematics and physics engineering from UPC BarcelonaTech, Spain, in 2016, the M.Sc. (Tech.) degree in ICT from the University of Turku, the M.Eng. degree in electronics and communication engineering from Fudan University, China, in 2018, and the Ph.D. (Tech.) degree from the University of Turku, in 2022. He is currently a Postdoctoral Researcher with the Sensory-Motor Systems (SMS) Laboratory, Swiss Federal School of Technology in Zürich (ETH Zürich), and the SCAI Laboratory, SPZ. His research interests include multi-robot systems, machine learning in robotics, and reinforcement learning.

TOMI WESTERLUND (Senior Member, IEEE) is currently a Professor of autonomous systems and robotics with the University of Turku, Finland, and a Research Professor with Wuxi Institute, Fudan University, Wuxi, China. He also leads the Turku Intelligent Embedded and Robotic Systems Research Group (tiers.utu.fi), University of Turku. His current research interests include the Industrial IoT, smart cities, autonomous vehicles (aerial, ground, and surface), and (co-)robots. In all these application areas, the core research interests are in multi-robot systems, collaborative sensing, interoperability, fog and edge computing, and edge AI.

MONCEF GABBOUJ (Fellow, IEEE) was an Academy of Finland Professor. He is currently a Professor with the Department of Computing Sciences, Tampere University, Finland. His research interests include big data analytics, multimedia analysis, artificial intelligence, machine learning, pattern recognition, nonlinear signal processing, video processing, and coding. He is a member of the Academia Europaea, the Finnish Academy of Science and Letters, and the Finnish Academy of Engineering Sciences, and a fellow of Asia-Pacific Artificial Intelligence Association.