IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 23, 2021, accepted February 27, 2021, date of publication March 4, 2021, date of current version March 11, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3063681

Automatic Person Detection in Search and Rescue Operations Using Deep CNN Detectors

SASA SAMBOLEK[1] **AND MARINA IVASIC-KOS**[2,3]
[1]High School Tina Ujevica, 44320 Kutina, Croatia
[2]Department of Informatics, University of Rijeka, 51000 Rijeka, Croatia
[3]Center for Artificial Intelligence and Cybersecurity, University of Rijeka, 51000 Rijeka, Croatia

Corresponding author: Marina Ivasic-Kos (marinai@uniri.hr)

This work was supported in part by the Croatian Science Foundation through projects "Automatic recognition of actions and activities in multimedia content from the sports domain" (RAASS) under Project IP-2016-06-8345, in part by the "A Knowledge-based Approach to Crowd Analysis in Video Surveillance" (KACAVIS) under Project IP-2018-01-7619, and in part by the University of Rijeka under Project 18-222.

ABSTRACT Due to a growing number of people who carry out various adrenaline activities or adventure tourism and stay in the mountains and other inaccessible places, there is an increasing need to organize a search and rescue operation (SAR) to provide assistance and health care to the injured. The goal of SAR operation is to search the largest area of the territory in the shortest time possible and find a lost or injured person. Today, drones (UAVs or drones) are increasingly involved in search operations, as they can capture a large, controlled area in a short amount of time. However, a detailed examination of a large amount of recorded material remains a problem. Even for an expert, it is not easy to find searched people who are relatively small considering the area where they are, often sheltered by vegetation or merged with the ground and in unusual positions due to falls, injuries, or exhaustion. Therefore, the automatic detection of persons and objects in images/videos taken by drones in these operations is very significant. In this paper, the reliability of existing state-of-the-art detectors such as Faster R-CNN, YOLOv4, RetinaNet, and Cascade R-CNN on a VisDrone benchmark and custom-made dataset SARD build to simulate rescue scenes was investigated. After training the models on selected datasets, detection results were compared. Because of the high speed and accuracy and the small number of false detections, the YOLOv4 detector was chosen for further examination. YOLOv4 model results related to different network sizes, different detection accuracies, and transfer learning settings were analyzed. The model robustness to weather conditions and motion blur were also investigated. The paper proposes a model that can be used in SAR operations because of the excellent results in detecting people in search and rescue scenarios.

INDEX TERMS Convolutional neural networks, object detector, person detection, search and rescue operations, UAV, YOLO.

I. INTRODUCTION

Many people are included in sport tourism to actively spend leisure time such as skiing, hiking, or nautical, which motivate them to stay in nature. Adrenaline or adventure tourism such as hiking, free climbing, mountain biking, paragliding, and rafting is gaining popularity, therefore the need to protect human life in hard-to-reach areas such as mountains, forests, canyons, caves, bodies of water and, karst phenomena is growing.

Due to a growing number of people living and carrying out various activities in the mountains and other inaccessible places, and because of the very nature of these activities and the physical and mental lack of preparedness for such activities, there is an increasing number of injuries, fractures and various accidents such as slipping, burying, etc. Risks that increase the insecurity of hikers, climbers, and other adrenaline athletes are, in addition to the occurrence of injury or illness, their skills and experience in coping with possible emergencies. Emergencies can arise, for example, due to incorrect assessment of the distance of the destination, incorrect assessment of the difficulty of the road, due to changes

The associate editor coordinating the review of this manuscript and approving it for publication was Mostafa Rahimi Azghadi.

in weather conditions, inadequate clothing or equipment, non-compliance with information and warnings, or insufficient preparation and overestimation of one's capabilities or knowledge. Reports of missing persons due to disorientation, illness, or suicidal intentions are also common.

To aid and health care to the injured in these circumstances, it is necessary to organize a search and rescue operation. The search action refers to a situation when the position and condition of the missing person are unknown, so the goal of the action is to locate the position of the missing person in nature. The rescue operation refers to a situation in which it is known that it is necessary to intervene and organize a person's rescue. If the accident's location is unknown in advance, this action includes search elements, too [1], [2].

The organization of assistance and health care in inaccessible areas is very complex, whatever the reason for the intervention. It is necessary to conduct demanding searches of large and complex terrains, especially when searching for a missing person. Besides, time is also an important factor in the search. As time goes on, the probability of a missing person's survival decreases and the searching area grows exponentially [3].

Search and rescue operations (SAR) require great human potential and material resources because they usually involve a large number of members of the mountain rescue service, search dogs, police and air forces, and more recently, crewless aerial vehicles (drones). Drones are now used for various purposes [4]–[10] and have become a standard in all SAR services globally. Except for searches in urban and non-urban areas, drones are used for searches on water (sea, rivers, floods [11]) or from avalanches. Their compactness, mobility, relatively low cost, and high-resolution real-time video recording are important when making quick decisions during actions and performing tasks that are potentially dangerous to humans, e.g., cliff search. The use of drones has increased the probability of finding a person, and due to "scanning" a larger area in one flight, the search time is shortened.

During search and rescue operations, the operator must analyze real-time images on a small screen while operating the aircraft. As the searching person is relatively small compared to the environment, they often take up only a few pixels on the screen. It is challenging to maintain long-term concentration and attention, even for people trained for it, to search for people in a large mountainous area or an area covered with vegetation. Persons searched for are often sheltered by vegetation, hidden behind a stone, or fused to the ground, further complicating the search even during favorable weather conditions. During rain, fog, and snow, the challenge of searching for a person is even more significant. Also, the searching person is very often in unusual places, most often due to loss of orientation, fall or dementia, in atypical postures and body positions due to injury, such as lying with unnaturally placed limbs or kneeling and sitting on the ground due to exhaustion or sudden disease or covered with stones due to slipping or landslides and the like and are very difficult to spot even in these selected parts of the image (Figure 1).

In SAR operations, operators could be greatly assisted by automatic person detection methods that would mark the persons in the images in real-time, i.e., their position and movement direction.

In recent years, deep convolutional neural networks such as Faster-RCNN [12], Cascade R-CNN [13], RetinaNet [14], SSD [15], YOLOv3 [16] have become successful in detecting people in images of mainly urban scenes and achieve even greater accuracy than humans. To achieve such good performances, deep network models had to be trained on large data sets such as MS COCO [17], Pascal VOC [18], ImageNet [19]. Then, to achieve good detection results or significant improvements in specific domains such as thermal images of the monitored area, some sports scenes, etc., not included in large data sets, it is necessary to additionally train deep networks on the image set from the selected domain [20]–[23].

In SAR operations, the key object is the person, however, recorded from a bird's eye view, and such recordings are not contained in the large data sets on which these state-of-the-art detectors are trained. To achieve the highest possible accuracy of the detection model, the data set on which the model is trained must have similar conditions to those that appear when testing the model, so it is necessary to train the model with a bird's eye view data. Recently, datasets that include images taken by a drone such as Visdrone [24], Okutama-action [25], UAVDT [26] have emerged. Those images are collected for various purposes [24]– [30], such as detecting objects in images and videos, tracking one or more persons, detecting an action, predicting a person's movement, or recognizing events in images. On the other hand, each dataset is tailored to a specific purpose and generally does not include scenes and rescue operations cases. The most similar scenarios shot by a drone to those in search and rescue are those involving people in a park while walking or running, standing in a square, walking down a street, or lying on a beach. Nevertheless, in these cases, persons' poses differ significantly from those who are injured, exhausted, or lost. For this reason, our dataset called SARD was created.

In this work, the SARD dataset was used for transfer learning of the selected state of the art person detectors: Faster R-CNN, YOLOv4, RetinaNet, and Cascade R-CNN and for fine-tuning for person detection in search and rescue scenes. We compared the model results on the SARD dataset. The YOLOv4 model was selected for further research because of achieving the highest accuracy and detection speed. To improve the detection results of the YOLOv4 model, we have analyzed the influence of different network resolutions, detection accuracy, and transfer learning settings on detection performance. The robustness of the YOLOv4 model to weather conditions and motion blur was also tested. Finally, after comprehensive testing and analysis of the results, we propose a model for person detection in search and rescue scenarios that can be of great help in SAR operations.

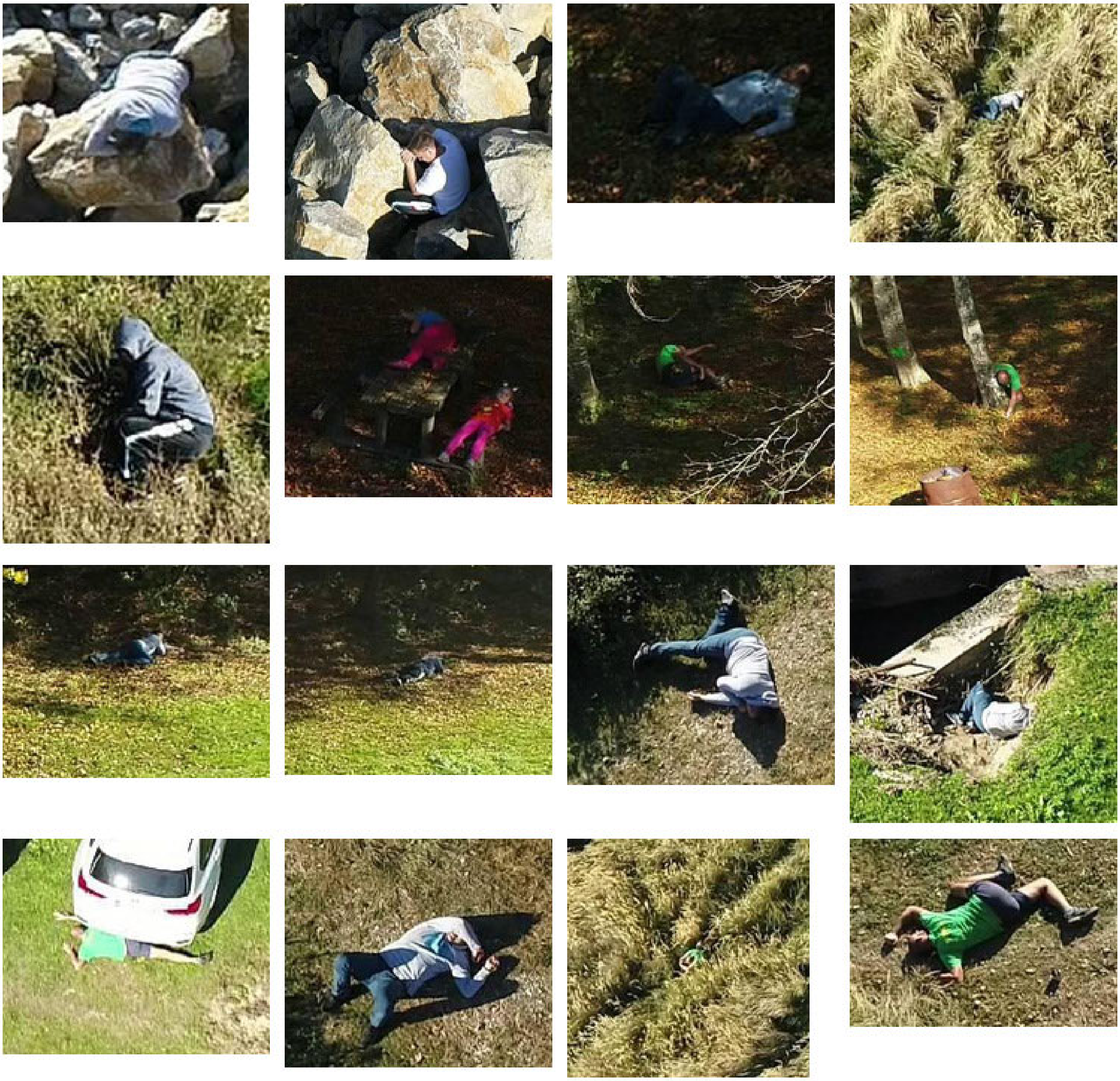

FIGURE 1. Some of the unusual places and atypical positions of the people being searched for, cut from images taken by a drone.

The main contributions of the paper are:

a) a novel dataset (SARD) of drone imagery in search and rescue operation, with statistics of the occurrence of a small, medium, and large object, annotated and prepared for supervised machine learning,

b) comparison of the performance of selected CNN detectors (Cascade R-CNN, Faster R-CNN, RetinaNet, YOLOv4) for use in SAR operations,

c) analyses of the influence of different network resolutions, detection accuracies, and confidence values on YOLOv4 person detection performance, and analysis of different transfer learning strategies considering the impact on detection results,

e) proposal of ROpti metrics for evaluating detector performances for SAR operations taking into account that there are as many positive detections as possible and as few false detections as possible,

f) proposal of YOLOv4 model to be used for person detection in SAR actions taking care to achieve the highest possible accuracy, with a few false detections as possible, with a network configuration that allows a person's online location and a configuration for off-line analysis, robust to various weather conditions.

The rest of the paper is organized as follows: Section 2 provides an overview of the research related to the commonly used methods for person detection in search and rescue operations assisted by drones and drone datasets. In Section 3, the SARD dataset was described, which was built and prepared for training models for person detection in SAR operations, as well as CNN architectures used for person detection. Section 4 describes in detail the experiments and analyzes the obtained results. The paper ends with the conclusion and direction for future research.

II. RELATED WORK

Today most object detectors consist of two parts, the backbone of the detector as a CNN network trained to extract features and a head that predicts the class and boundary box of the detected objects. Networks such as VGG [31], ResNet [32], ResNeXt [33] or MobileNet [34]–[36] pre-trained on the ImageNet [19] or OpenImages [37] dataset, are most commonly used as backbones. The head of a detector can be divided into two types: one-stage and two-stage detectors. YOLO [16], [38]–[40], SSD [15] and RetinaNet [14] are examples of the one-stage detector. The most representative two-stage detectors are R-CNN detectors [41], including Fast R-CNN [42], Faster R-CNN [12] and, R-FCN [43]. Two-stage detectors are usually more accurate in terms of localization and classification accuracy. On the other hand, they are slower in processing than one-stage detectors. Many detectors add extra layers between the backbone and head (neck), e.g., Feature Pyramid Network (FPN) [44] typically used to collect multiple feature maps, each with a different resolution, which is useful for recognizing objects at different scales.

A. DEEP CNN DETECTORS IN SEARCH AND RESCUE OPERATIONS AND DRONE IMAGERY

According to [45], search and rescue operations can be divided into four areas: search in military operations, search on water, in urban and non-urban areas. The use of drones in search and rescue operations has been discussed in [46]–[49]. The domain of our interest is the non-urban area and water.

In [49], image segmentation and contrast enhancement were applied, followed by an SSD detector to detect persons in drone images. They also used a 3D game editor to generate synthetic datasets depicting search and rescue actions.

The Inception model with the Support Vector Machine (SVM) classifier was used in [50] to detect people trapped in an avalanche by searching with drones. In [51], the focus is on detecting people at sea recorded by crewless aerial vehicles equipped with a multispectral camera, and a modified MobileNet architecture is used for detection.

The authors in [52] developed a system for detecting people and recognizing actions on the Okutama-action dataset with GPS location calculation. A model upgraded to MobileNetv2 and named POINet was used to detect objects. Another example of a GPS signal using in search and rescue operations is given in [53]. It is assumed that the injured person has a mobile device switched on, so the injured person's position is determined by combining the GSM signal's strength and the drone's GPS position.

A platform for detecting a person in the water with the Tiny YOLO V3 architecture was presented in [54]. The model is trained on the MS COCO dataset and dataset recorded by a drone equipped with a GoPro camera in HD resolution. A real-time algorithm for detecting and tracking ocean surface objects has been proposed in [55].

A strategy for using semi-supervised and supervised machine learning approaches to classify drone imagery and object detection, along with a proposed hardware and software architecture for the UAV platform, is given in [56].

An algorithm for planning a search path for crewless aerial vehicles (UAVs) and using crewless ground vehicles (UVGs) to verify the identity of the object detected by the UAV is given in [57]. In [58], the authors compare several CNN architectures for the binary classification task to classify drone images as with or without persons. According to [59], it was the first work to apply multiple visual tracking of objects on drone photographs for search and rescue purposes. Person detection is based on color and depth data and a human shape filter that uses human joint locations derived from the Convolutional Pose Machine [60]. The purpose of using the filter is to investigate the human body's shape on the proposed detections to avoid false detections.

B. DRONE IMAGE DATASETS FOR CNN TRAINING

Recently, an increasing number of datasets have been made using a drone and prepared to train deep neural networks. These datasets include footage containing scenes of urban areas such as squares, streets, playgrounds, parking lots, etc. (Figure 2).

VisDrone [24] data set contains 263 video clips and additional 10,209 images related to detection tasks and tracking one or more objects. Videos/images were taken on different drone platforms (DJI Mavic, DJI Phantom Series 3, 3A, 3SE, 3P, 4, 4A, 4P) in 14 different cities in China. The set covers different weather and light conditions of maximum video (3840 x 2160 px) and image (2000 × 1500 px) resolution.

Okutama-action [25] contains videos that tag people and those people's actions such as walking, running, sitting, or lying down. Also, interaction with other objects is annotated, such as reading, drinking, carrying, pushing, and interactions between people such as hugging and handling. A CARPK dataset [27] is appropriate for counting the objects since it contains 89,777 marked cars recorded by a drone. Campus [28] is the largest set of data recorded from a bird's eye view, including pedestrians, cyclists, cars, buses, etc.

The UAV123 [29] dataset, in addition to drone images, also contains synthetic video recordings made by a drone simulator on the Unreal 4 Game Engine. UAVDT [26] contains drone-recorded videos of an urban area such as streets, squares, intersections taken in various weather conditions (day, night, fog). ERA [30] dataset has 24 event classes that can occur on aerial video footage such as fire, flood, traffic jam, concert, etc.

None of the above datasets contain recordings specific to search and rescue operations, so although there are object

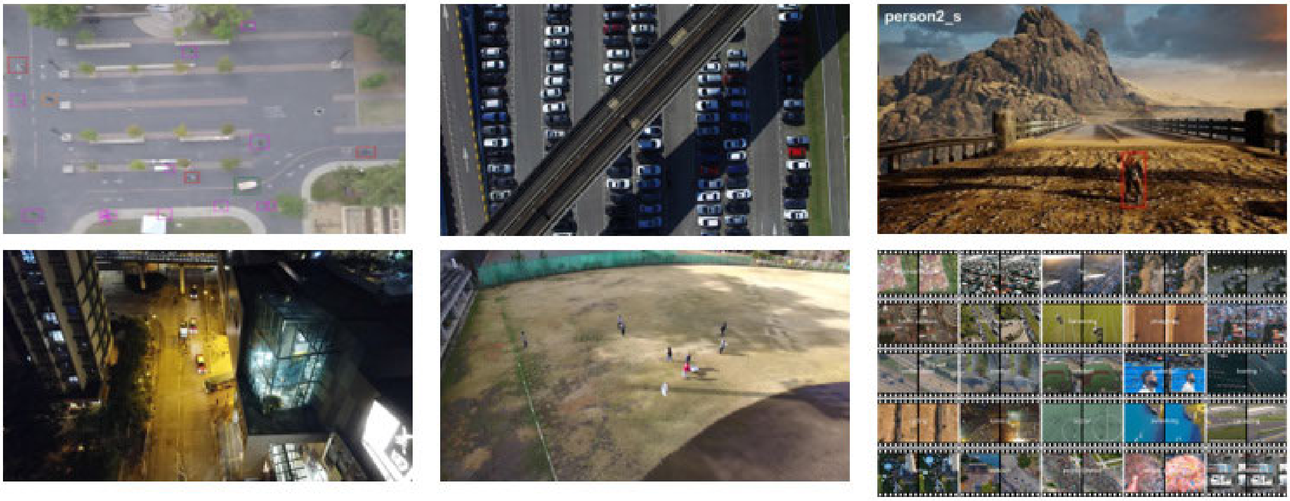

FIGURE 2. Examples of images from existing datasets. Top-left Campus [28], top-middle CARPK [17], top-right UAV123 [29], bottom-left VisDrone [24], bottom-middle Okutama-action [25], bottom-right ERA [30].

detectors who achieve excellent results in detecting people on urban scenes, the question is how successful they would be in SAR operations in rural/ mountainous areas? How to test the performance of detectors in the SAR domain if there is no appropriate test set? What performance can be achieved after training the model on examples of SAR scenes and with which model and learning parameters?

III. EXPERIMENT WORKFLOW

A. PROBLEM FORMULATION

The experiment automatically detects persons using object detectors in images taken by a drone in non-urban areas during search and rescue operations.

Guided by the experience from previous work [61], [62], we have analyzed state-of-the-art object detectors such as Faster-RCNN [12], YOLOv4 [40], RetinaNet [14], and Cascade R-CNN [13]. The aim was to select the one that achieves the best results in terms of accuracy and inference speed and best fits our task.

All considered detectors were pretrained on the MS COCO dataset, and the feature maps learned on that dataset are expected to be useful for detecting persons for our task, too. However, to improve the detection results in SAR applications, the models should be re-trained on an appropriate dataset that contains scenes typical for search and rescue operations.

We searched the available databases of drone images and found out that appropriate publicly available datasets for this purpose did not exist. The existing [24]–[30] do not fully coincide with the intended goal of detecting (injured/exhausted) persons in the non-urban area. However, we decided to use the VisDrone dataset for transfer learning since it contains images of people in the urban scenes that are the closest scenario to our task. Also, we decided to build a dataset of images with scenes that simulate the poses of injured/exhausted people in the non-urban area taken by drones. Also, to simulate different weather conditions and increase the generality of the model, we will use the available algorithms and generate new images to increase the data set.

We re-train the models on the built dataset, and the model that achieves the best results was selected for further testing and adjustments to improve the detection result further.

B. DATASET CREATION

SARD database was built to detect casualties and persons in search and rescue scenarios in drone images and videos. The actors in the footage have simulate exhausted and injured persons and "classic" types of movement of people in nature, such as running, walking, standing, sitting, or lying down. Since diverse terrain and backgrounds determine possible events and scenarios in captured images and videos, the shots include persons on macadam roads, quarries, low and high grass, forest shade, and similar.

1) COLLECTION AND PREPROCESSING OF SARD DATASET

During the daylight, the shooting was carried out in the fall, with a high-performance camera of the DJI Phantom 4A drone with a 3-axis solo gimbal stand. Videos were recorded at an FHD resolution of 1920 × 1080 pixels at a frequency of 50 frames per second. The drone flew at different altitudes from 5 m to 50 m and different camera angles (ranging from 45° to 90°). All videos were shot in the area of Moslavacka gora, in Croatia, outside the urban area. Positions of persons in the images range from standard (standing position, sitting, lying, walking, running) to positions typical of exhausted or injured persons reconstructed by actors at their discretion, Figure 3. The actors were nine people of different ages and genders, aged 7 to 55 years, to include differences in movement and postures associated with age and different body constitutions. Also, actors are in various locations, from clearly visible (to the eye) to locations in the woods, tall grass, shade, and similar, which further complicates detection.

FIGURE 3. Actors, different ages and genders who participated in the recording of the SARD dataset.

From the recordings with a total length of about 35 minutes, 1,981 single frames with people on them were singled out. In the selected images, the persons were manually tagged by a horizontal bounding box typically used for object annotation in remote sensing images and natural scene images [63] so that annotated images could be used to train a supervised model. Tags are stored as XML files in PASCAL VOC format and the YOLO format.

2) GENERATION OF CORR DATASET

An extension of the SARD set called Corr was created to increase the robustness of the SARD data. Corr dataset includes images that further simulate various weather conditions that may occur in actual search and rescue situations such as fog, snow, and ice. Also, blur images are included in the Corr set to simulate camera movement and aerial shooting in motion.

The Corr train set was generated from images of the SARD train set, and likewise, the Corr test set was generated from the images of the SARD test set using the same methods [64]. To achieve an even distribution of data with different weather conditions in the set, we generated the images sequentially by adding the effect of snow, fog, frost, and blur in turn. Each of the effects was added at four levels of concentration to simulate the range of possible weather conditions and motion effects that may occur in actual SAR missions, e.g., light snow and heavy snow, snowstorms, rain, and showers, and the like. For the maximum level of concentration of an effect, we chose the level at which objects, which are relatively small in most images, could still be visually recognized. To test the detection results for specific weather conditions, we created four subsets for testing Corr-snow, Corr-fog, Corr-frost, Corr-fogging, each containing 714 images. The image tags remained the same as in the SARD dataset, so no additional tagging was required. An example of generated images of the Corr dataset is given in Figure 7.

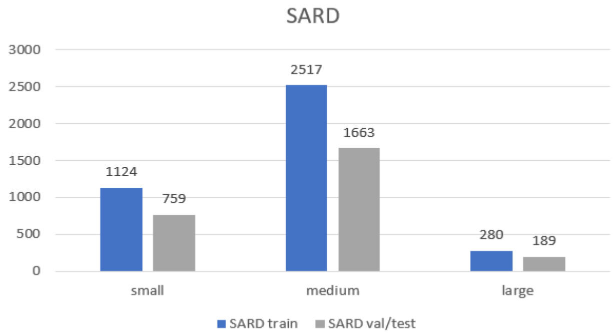

FIGURE 4. Marked persons according to the size of the bounding box area for the SARD dataset.

3) STATISTICS OF DATASETS USED FOR TRANSFER LEARNING

The SARD set images were divided 60:40 into a train set and a test set so that they were evenly distributed according to the scenes (background, lighting, person pose, camera angle). The training set contains 1189 images, on which 3921 persons are marked, while the test set contains 792 images, on which 2611 persons are marked. The bounding boxes' dimensions in the SARD set range from 7px for the smallest width and 8px for the smallest height, while the maximum width is 353px and the maximum height is 337px. The area of the smallest object bounding box is 7 × 12px while the largest is 322 × 231px, and the average bounding box size is 47px x 58 px. The SARD set contains 1883 small person objects (objects whose boundary box area is less than 32^2), 4180 medium person objects ($32^2 <$ boundary box area $<96^2$), and 469 large objects (boundary box area $> 96^2$). The frequency of occurrence of persons in the SARD dataset concerning the size of the object bounding box is graphically shown in Figure 4.

The Corr train set's size corresponds to the SARD train set in terms of the number of images and the number and size of objects. There are 1,903 images in the set, which show 6,265 persons, of which 1,775 are small objects, 4,026 are medium-sized objects, and 464 are large objects. The Corr test set is slightly smaller than the SARD test set because the images on which the persons were not visible after adding blur, rain, snow were deleted. These are mostly images in which people were in the shadows, took up very few pixels, or were occluded. The number of persons in the Corr dataset is shown in Figure 5.

Another 2,129 images from the VisDrone image set, which includes a person or pedestrian tag, were selected for model training to generalize the learning data set. For selected images, person or pedestrian tags are merged into one class: person. This set is referred to as VisDrone2000. The VisDrone2000 drone image dataset was divided into a training set consisting of 1,598 images with 29,797 tagged persons and a test set containing 531 images with 13 969 persons.

The set contains 36,951 small person objects, 6,719 medium-sized objects, and only 96 large person objects. A

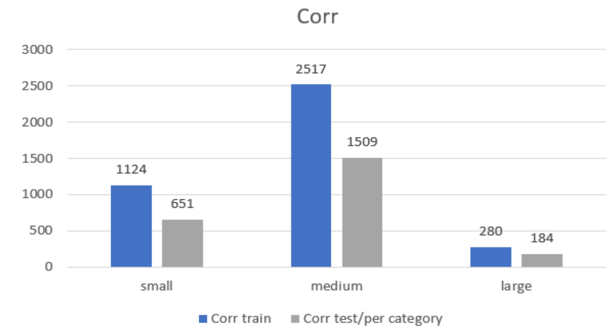

FIGURE 5. Marked persons according to the size of the bounding box area for the Corr dataset.

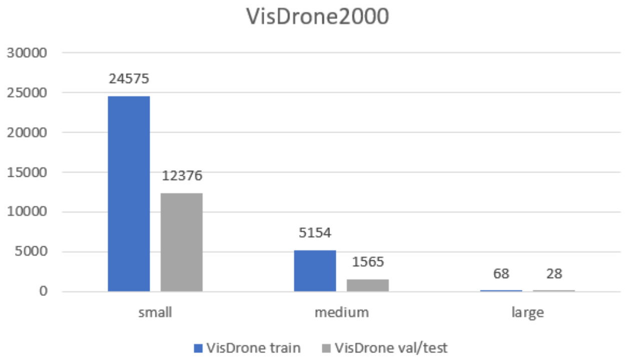

FIGURE 6. Marked persons according to the size of the bounding box area for the VisDrone2000 dataset.

VisDrone2000 data statistic shows that the VisDrone dataset recordings were made at higher altitudes than the SARD dataset (Figure 6.).

Combinations of used sets and learning methods are described in Section 4.

C. SELECTED OBJECT DETECTORS

We have tested the state-of-the-art object detectors on a custom-made SARD dataset and selected drone images from the VisDrone benchmark dataset to select the best-suited detector for our task of detecting persons in search and rescue scenes.

In the experiment, we have compared the performance of the CNN-based detectors: Faster R-CNN, YOLOv4, RetinaNet, and Cascade R-CNN. All selected detectors were previously trained on the MS COCO [17] dataset. All detector models are further trained on bird's eye view images from a part of the VisDrone and a SARD custom dataset to improve their performances.

Below is a brief description of the architecture of examined detectors.

1) FASTER R-CNN

The Faster R-CNN detector from the R-CNN series [12], [41], [42], detectors is a two-phase region-based detector. These detectors' basic idea is to select the regions of interest from the image in the first phase. In the second phase, the classification and correction of the coordinates of the object will be performed.

In our case, ResNet50 [32], a pre-trained deep neural network, is used as a backbone, which receives an image at the input and provides feature maps at the output that predicts regions of interest using the Region Proposal Network (RPN). RPN for feature maps of any dimensions, as an output gives a list of RoI's with a certain probability that the object is in the default RoI. The tested Faster R-CNN detector uses FPN to collect multiple feature maps of different resolutions. In this experiment, the implementation of a faster_rcnn_r50_fpn_1x detector from a MMDetection codebase [65] was used.

2) YOLOV4

The YOLO architecture seeks to merge localization and classification problems into one deep convolutional neural network. It divides the image into a grid of dimensions S x S in which each cell provides frames for the object. The probability, which is calculated for each frame, tells us how sure the model is when there is an object inside the frame and how sure it is of the boundaries' accuracy.

For the latest version of the YOLO detector, the authors explored typical algorithms used in deep learning models and further designed and improved some modules.

This model uses CSPDarkNet53 as the backbone [66]. DarkNet53 is a deep residual network with 53 layers, while in the case of YOLOv4, CSPNet (Cross Stage Partial Network) is added to the basic DarkNet53. The authors added Spatial Pyramid Pooling (SPP) [67] as a neck to increase the receiving (receptive) field without causing a decrease in velocity. Instead of the Feature Pyramid Network (FPN) used in the YOLOv3 version, the authors chose the Path Aggregation Network (PAN) [68] while using the original YOLOv3 [16] network for the head.

In addition to the new architecture, the authors also use training optimization to achieve greater accuracy without additional hardware costs, which the authors call "Bag of Freebies." Bag of Freebies includes CutMix, Mosaic, CIoU-loss, DropBlock regularization, etc. On the other hand, the authors propose a "Bag of Specials," a set of modules such as Mish activation, SAM-block, Cross-stage partial connections (CSP), etc., that only slightly increase the hardware cost with a significant increase in detection accuracy.

We used the Darknet framework to train and evaluate the YOLOv4 model.

3) RETINANET

RetinaNet is a single-phase detector composed of a backbone and two sub-networks specific to the task. The ResNet-FPN network, as the RetinaNet detector's backbone, is responsible for calculating the input image's feature map. The first sub-network performs the classification while the second regresses the boundary frames.

A sub-classification network predicts the probability of an object's presence in each spatial position for each class.

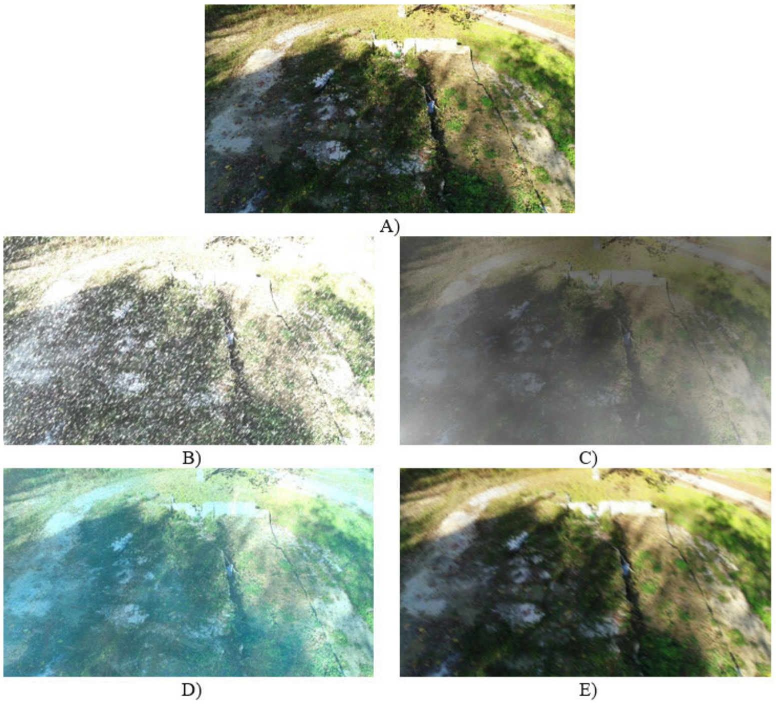

FIGURE 7. SARD Corr set with the added effect of bad weather and camera shift on the image, examples of generated images: A) original image, B) snow, C) fog, D) ice, E) motion blur.

This subnetwork is a small FCN associated with each FPN level; this sub-grid parameter is shared at all pyramids levels. Unlike RPN [12], the RetinaNet sub-network for object classification is deeper, uses only 3 × 3 convolutions, and does not share parameters with the frame regression network.

In parallel with the sub-network of object classification, they attach another small FCN to each level of the pyramid for regression of the boundary frame. In experiments, the implementation of the retinanet_r50_fpn_1x detector in the MMDetection codebase [69] was used.

4) CASCADE R-CNN

Cascade R-CNN is a multi-phase extension of the Faster R-CNN architecture that aims to increase detection quality by constantly increasing IoU values. The focus is on the detection subnet, adopting an RPN to detect suggestions. However, Cascade R-CNN is not limited to this proposed mechanism since other choices are possible. The goal is to simultaneously increase the quality of hypotheses and improve detection results by combining cascade boundary frame regression and cascade detection. The implementation of a cascade_rcnn_r50_fpn_1x detector in a MMDetection codebase [70] was used.

D. EVALUATION METRICS

Detector performance (bounding box of detected objects, the class assignment, and a reliability value) was assessed on unseen images using standard evaluation measures such as precision, recall, and mean average precision (mAP). In our case, only the class person is considered, so the mAP is equal to the average precision (AP).

In the case of SAR operations finding a person as soon as possible is key to a successful SAR operation, so it is essential to detect missing people if they exist on the scene. Equally important is to have a few false detections as possible so that human resources are not wasted. Precision measures how accurate the detection results are, i.e., the percentage of true positive detections to the total number of detections. In contrast, recall measures how many true positive detections there are concerning the number of all possible detections [62].

$$Precision = \frac{TP}{(TP + FP)} \tag{1}$$

$$Recall = \frac{TP}{(TP + FN)} \tag{2}$$

TP = True Positive, TN = True Negative, FP = False Positive, FN = False Negative.

The detection is considered positive if the intersectional ratio of the detected bounding box and the corresponding ground truth bounding box and their union is 50% or higher. This measure is referred to as intersection-over-union (IoU). An example of positive and negative person detection considering IoU>= 0.5 is shown in Figure 8.

To precisely evaluate and characterize the performance of the detector, taking into account not only the accuracy of detection but also the size of objects in the image, six average

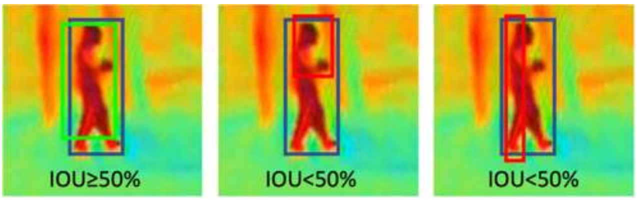

FIGURE 8. Visual representation of positive (left) and negative (center and right) representation of intersection over union (IoU) criteria equal to or greater than 50% [71].

TABLE 1. Comparative preliminary detection results on SARD and VisDrone datasets (%).

Test set	Model	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
VisDrone	Cascade R-CNN	8	18	**6**	4	25	48
	Faster R-CNN	8	19	**6**	5	26	43
	RetinaNet	7	15	4	3	22	43
	YOLOv4	**13**	**30**	1	**11**	**36**	**80**
SARD	*Cascade R-CNN*	19	35	18	9	21	43
	Faster R-CNN	20	39	18	10	22	42
	RetinaNet	17	34	14	5	19	**47**
	YOLOv4	**23**	**40**	**25**	**13**	**26**	41

precision measures in MS COCO format were considered using the original script[1]:

- AP overall 10 IoU thresholds (0.5: 0.05: 0.95),
- AP50 at IoU = 0.50,
- AP75 at IoU = 0.75.

Average precision across different object scales is evaluated as:

- AP_S for small objects with an area of less than 32^2 px,
- AP_M for medium objects with an area between 32^2 and 96^2px,
- AP_L for large objects with an area of more than 96^2px.

IV. EXPERIMENTS

A. PRELIMINARY DETECTION RESULTS

Preliminary detection results of models of selected state-of-the-art CNN-based object detectors pre-trained on MS COCO dataset on our custom-made SARD test set andVisDrone2000 are given in Table 1. The best results are marked in bold. YOLOv4 achieved significantly better overall results on both test sets considering precision accuracy and object scales.

B. DETECTION PERFORMANCE AFTER TRAINING ON DOMAIN IMAGES

To achieve better person detection in the search and rescue scenes, we have also trained the original detectors on the Visdrone data set and on the SARD data set and compared the models' performances.

The MMDetection codebase was used to train the Cascade R-CNN, Faster R-CNN, and RetinaNet models, and the darknet framework model was used to train the YOLOv4 model. The learning rate (lr) was set to 0.005 as the training was performed on a single GPU computer. All other settings are the same as the original model settings. YOLOv4 models are trained on Google Colab with batch = 64 and subdivision = 32, with the network resolution set to 512 × 512. All models are tested on a laptop with one 1660Ti GPU.

TABLE 2. Comparative results of models trained and tested on VisDrone2000 dataset (%).

Model	AP	IMP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Cascade R-CNN (VisDrone2000)	17	8.80	38	13	12	39	38
Faster R-CNN (VisDrone2000)	13	4.70	34	8	9	33	28
RetinaNet (VisDrone2000)	8	1.80	26	3	6	20	18
YOLOv4 (VisDrone2000)	**23**	**10.2**	**55**	**15**	**21**	**40**	**34**

After training the model on the selected dataset, each model is referred to in the text as a ***model*(*dataset*)** to make it easier to compare the models' performances. For example, *Cascade R-CNN (VisDrone2000)* means a Cascade R-CNN detector trained on the VisDrone2000 dataset.

1) TRANSFER LEARNING WITH THE VISDRONE DATASET

The Cascade R-CNN(VisDrone2000), Faster R-CNN (VisDrone2000) and RetinaNet(VisDrone2000) models were trained in 6 epochs with batch_size set to 1, while the YOLOv4(VisDrone2000) model was trained with max_batches = 6000 and batch = 64.

The detection results on the VisDrone2000 test set for AP are shown in Table 2. The Imp column shows the progress of the model relative to the pre-trained model tested on the same data set.

YOLOv4 (VisDrone2000) achieves an average score of 23% AP which is the best result compared to other tested detectors. Yolo proved to be equally the best in all AP measures related to object size and detection accuracy. By far, the best results of 55.1% AP YOLOv4 achieved on IoU = 0.50.

Cascade R-CNN(VisDrone2000) achieves the second-best results but still significantly worse results than YOLOv4(VisDrone2000). Similar conclusions were reached in [72], [73].

2) TRANSFER LEARNING WITH THE SARD DATASET

When training the models on the SARD set, the same model learning parameters were used as at the Visdrone2000 set. The detection results on the SARD test set are given in Table 3. The best results were obtained with YOLOv4 (SARD), while the results of Cascade R-CNN (SARD) and Faster R-CNN (SARD) detectors are very similar but significantly worse than YOLOv4. All detectors achieve the best results for the case of AP50, with the best results of over 96% achieved by YOLOv4 (SARD). If higher detector precision is required, AP_{75}, all detectors perform significantly worse,

[1] https://github.com/cocodataset/cocoapi

TABLE 3. **comparative results of models trained and tested on the SARD dataset (%).**

Model	AP	IMP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Cascade R-CNN (SARD)	49	30.0	88	51	31	54	63
Faster R-CNN (SARD)	50	29.9	91	51	30	56	65
RetinaNet (SARD)	34	17.2	73	25	13	41	53
YOLOv4 (SARD)	**61**	**37.9**	**96**	**71**	**45**	**66**	**73**

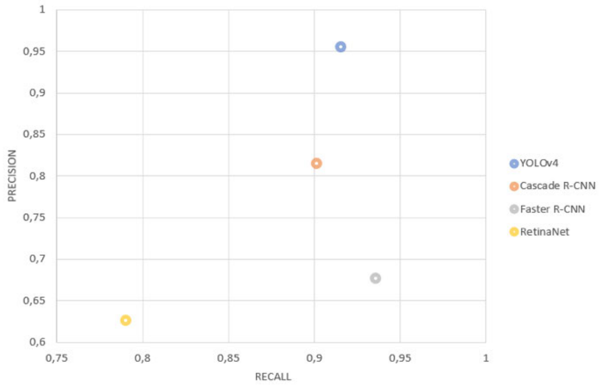

FIGURE 9. **The precision vs. recall ratio for models YOLOv4 (SARD), Cascade R-CNN (SARD), Faster R-CNN (SARD), and RetinaNet (SARD).**

with the highest mean accuracy of 71% being achieved again by YOLOv4 (SARD). All detectors' results are significantly higher on the SARD set than on the VisDrone set and significantly better than the original model when no additional training on domain images was performed.

When comparing the detection results concerning the objects' size, it is clear from Table 3 that all detectors achieve significantly better results for large objects than for medium and small objects. The best average accuracy of 73% is achieved by YOLOv4 (SARD) large objects, followed by 66% for medium objects. Faster R-CNN (SARD) and Cascade R-CNN (SARD) perform similarly but score 10% lower in the case of large and medium objects. For small object detection (AP_S), YOLOv4 (SARD) proved to be the best with an accuracy of 45%, while Faster R-CNN (SARD) and YOLOv4 (SARD) achieved comparative results, for about 15% lower.

Figure 9. shows the precision and recall ratio for all tested models. The best ratio of precision and recall, with 96% of precision for recall greater than 91% was achieved by YOLOv4 (SARD), which means that it was the most precise in the detection and has detected the most significant number of objects that exist in the image (ground truth). The best recall was achieved by Faster R-CNN (SARD) but with a precision of 67% and much more false positive detections than YOLOv4 (SARD). RetinaNet (SARD) had the lowest precision and the lowest recall.

In search and rescue operations, the goal is to detect all persons present on the scene. Still, on the other hand, the detector's precision is also important so that resources are not wasted on false detections. For this reason, based on the achieved results of average precision and the ratio of precision and recall, the YOLOv4 detector was selected for further research.

Examples of person detection results with models trained on the SARD dataset are shown in Figure 10. The columns in Figure 10 represent the detection results, respectively, in column A) Cascade R-CNN (SARD) model, in column B) Faster R-CNN (SARD) model, C) RetinaNet (SARD), D) YOLOv4 (SARD), and in E) ground truth. All possible detection outcomes appeared in Figure 10.: a positive detection where a person is detected, and IoU of bounding box and person's ground truth is more or equal than 50%, then a negative detection where a person is not detected, or IoU of the bounding box and person's ground truth is less than 50% and a false-positive detection where a part of the image that does not contain a person was marked as a person.

The first row in Figure 10 shows a quarry case with one person on a pile of rocks while two people are on a dusty road. All detectors successfully detect a person on the road, while only Cascade R-CNN (SARD) and YOLOv4 (SARD) also detect a person sitting on rocks. Faster R-CNN (SARD) has one false detection and multiple detections of a person on the road.

The second row shows an example of three people with an overlap (occlusion) on low grass. All detectors successfully detected the standing person on the top right. Faster R-CNN (SARD) gives multiple detections of overlapping persons. At the same time, Cascade R-CNN (SARD) and RetinaNet (SARD) have occlusion problems and did not detect a person kneeling behind a moving person. YOLOv4 (SARD) successfully detects all persons.

The third scene with eight people was shot from a greater height than the first two examples. Cascade R-CNN (SARD) detects seven individuals with one false detection, Faster R-CNN (SARD) has five accurate detections as well as RetinaNet (SARD), which also has three false detections. YOLOv4 (SARD) precisely detects all persons in the image.

In the last case, taken from an even greater height and distance from the object, nine people are in the tall grass and macadam road. The Cascade R-CNN (SARD and the Faster R-CNN (SARD) accurately detect seven persons while the RetinaNet (SARD) detects only five of them. YOLOv4 (SARD) successfully detects all subjects in the image.

From the qualitative analysis of the selected examples, it is clearly shown that YOLOv4 (SARD) was the most successful in detecting persons in SAR scenarios. However, there are also examples where the YOLOv4 (SARD) model was not successful, Figure 12. The most common examples of false detection are the cases when two people are standing very close to each other or overlap (Figure 12, first row) and when the detector detects darker parts of vegetation (Figure 12, second row) or shadows (Figure 12, third row) as a person. It is almost typical for a person to merge with the background in search and rescue operations practically. In that case, it is challenging to detect a person even for a trained person, so it

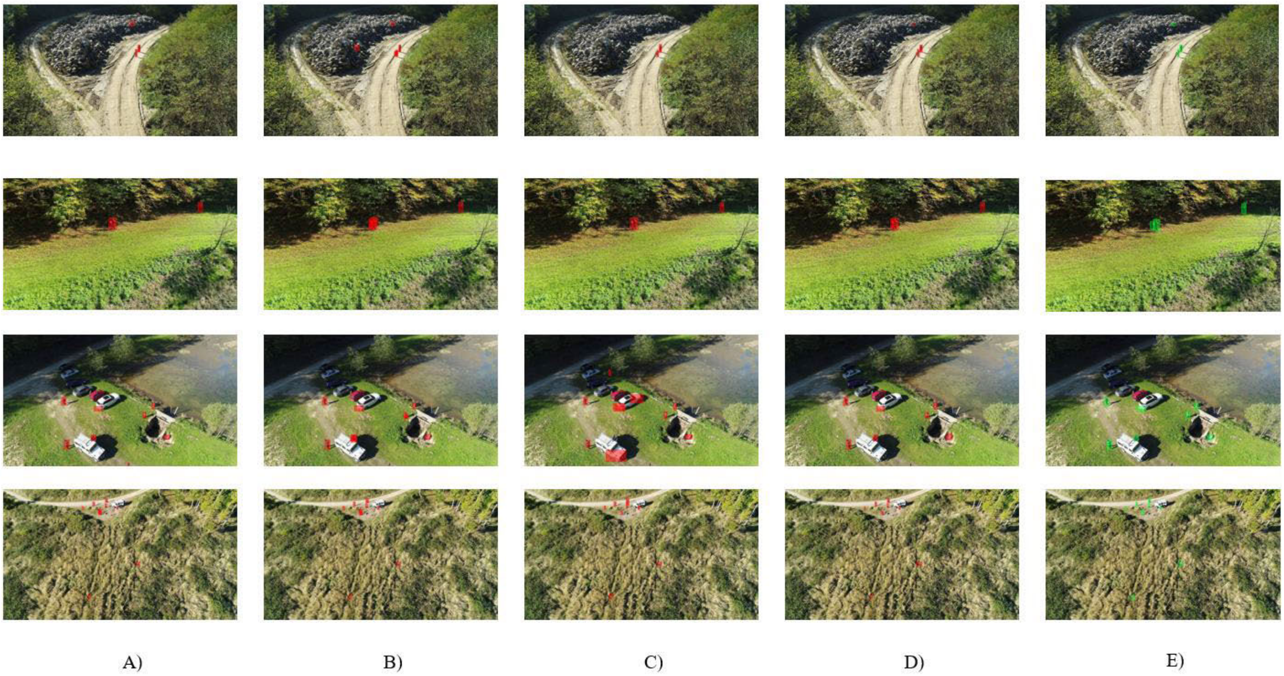

FIGURE 10. Examples of person detection results of different models retrained on the SARD dataset: A column: Cascade R-CNN (SARD), B column: Faster R-CNN (SARD), C column: RetinaNet (SARD), D column: YOLOv4 (SARD), E column: ground truth.

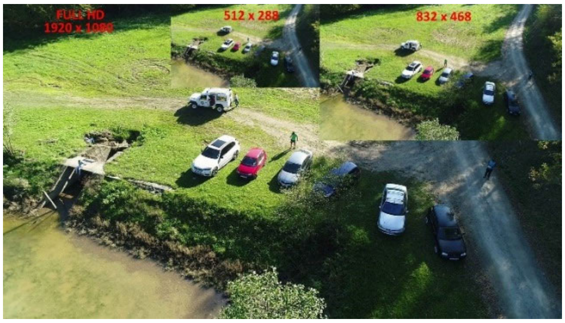

FIGURE 11. Comparison of different images resolution.

is not unexpected that the detectors have the most missed detections in that case (Figure 11, third row).

We try to adjust the model parameters and learning conditions to achieve even better detection results with the YOLOv4 detector in the experiment's continuation.

C. DETECTION RESULTS REGARDING THE NETWORK RESOLUTION

The YOLO architecture resizes the input image, preserving the aspect ratio to the resolution defined in the.cfg weights file, defined by the width and height parameters. These parameters are called network resolution. Transformation of input image resolution in Yolo architecture is given by:

$$\begin{aligned} Img_{train\_width} &= Net_{width}, \\ Img_{train\_height} &= \frac{Net_{width}}{Img_{width}} Img_{heigth} \end{aligned} \tag{3}$$

For example, if the input resolution of an image is 1920 × 1080 and the network resolution is defined as width, $Net_{width} = 512$ height, $Net_{height} = 512$, YOLO will change the resolution of the input image to the set width, Net_{width}, preserving the original ratio between image width, Img_{width} and height, Img_{height} e.g. 1920 × 1080 will be transformed to 512 × 288. Comparison of different images resolution is shown in Figure 11.

When done in both train and test sets of the model, this subsampling of image resolution does not violate the general rule of model training since the model was trained on similar object sizes as those that appear in the test set.

To improve the detection performance, especially the detection of small objects, one alternative was to use the higher resolution of input images and train the network at higher resolutions, e.g.:

$$Net_{\overline{widt}} = Net_{width} + k, k = 32n, \quad n \in \mathbb{N} \tag{4}$$

Values Net_{width} and Net_{height} that are multiples of 32 can be used, such as 608 × 608 or 832 × 832, because the YOLO network down-samples the input image by 32.

In our case, the input images size is (Img_{train_width}) 1920 × 1080, and the YOLOv4 (SARD) model was trained on (Net_{train_width}) 512 × 512 network resolution. Our computer was too weak to train the network at higher resolutions than that, so the alternative was to increase the network resolution during testing (Net_{test_width}) [74]. The idea was always to use input images of the same resolution of 1920 × 1080 when training and to test the model on higher resolution images

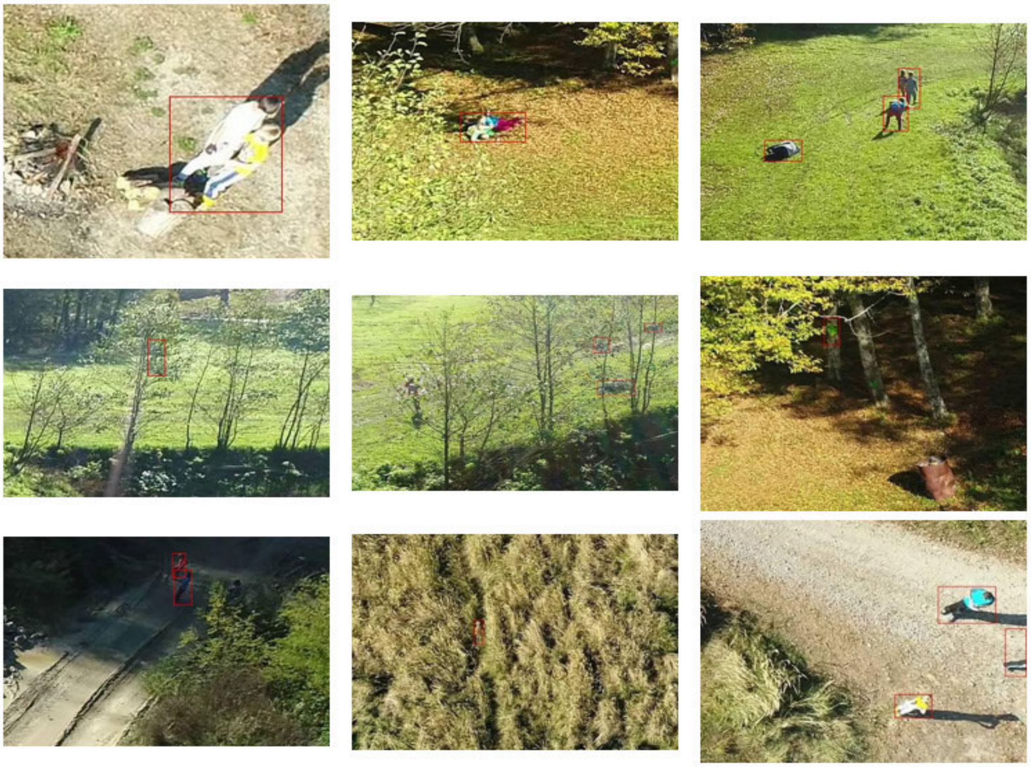

FIGURE 12. Miss detections of the YOLOv4 (SARD) model (cropped images to make it easier to notice the persons in the image): two people are standing very close to each other or overlapping (first row); darker parts of vegetation detected as a person (second row); shadows detected as persons (third row).

without compromising the sizes and ratio among the objects learned during training:

$$\frac{Net_{test\_width}}{Img_{test\_width}} = \frac{Net_{train\_width}}{Img_{train\_width}};$$
$$\frac{Img_{test\_width}}{Img_{train\_width}} = \frac{Net_{test\_width}}{Net_{train\_width}} \tag{5}$$

To preserve the ratio (5) for higher image resolution during testing, it was necessary to increase the network resolution. To examine the effect of changing the network resolution during testing (Net_{test_width}) on object detection performance, we have tested different network resolutions below and above the resolution at which the model was trained: 320 × 320, 416 × 416, 512 × 512, 608 × 608, 832 × 832, 1024 × 1024. The network resolutions 320 × 320 and 416 × 416 are below the resolution at which the YOLOv4 (SARD) model was trained, while the resolutions 608 × 608, 832 × 832, 1024 × 1024 are above. The detection results are given in Table 4.

The best accuracy results are achieved for a network resolution of 832 × 832, Table 4, except in the case of large objects (AP_L). A comparison of the results shows that better detection results can be obtained by increasing the network resolution when testing. Better results are achieved at resolutions 608 × 608 and 1024 × 1024 than at a resolution of 512 × 512 at which the model was trained. However, results also show that there is a limit after which the results no longer improve, such as in the case of network resolution of 1024 × 1024, when the results started to decrease.

TABLE 4. YOLOv4 (SARD) detection performance depending on the network resolution (%).

Network resolution, Net_{test}	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	fps
320x320	37	77	31	11	44	68	**10.3**
416x416	51	88	54	26	59	75	9.73
512x512	57	93	63	34	63	**77**	7.37
608x608	60	95	67	39	63	**77**	6.61
832x832	**61**	**96**	**71**	45	**66**	73	3.73
1024x1024	60	95	64	**46**	65	66	2.50

In the case of testing at the lower resolutions than the network resolution on which the model was trained, in general, worse results are obtained except in the case of the large object where just slightly worse results are achieved. It can be noted that the inference speed is about 10 fps for the lowest network resolution, which is 2.5x faster than at a resolution of 832 × 832, at which the most accurate results are obtained.

The best average precision of 77% is obtained with 512 × 512 pixels and 608 × 608 pixels for large objects. For medium objects, the best average precision is 66% with a network resolution of 832 × 832, and for small objects, the best average precision of 46% is got with a network resolution of 1024 × 1024 pixels.

When approving the resolution of the most suitable model for SAR operations, we were guided by the fact that detection can be performed on-site during flight operations and off-line on the recorded materials since the drone's flight time is limited to the battery life.

In real-time detection on a video received while the drone was flying over the area being searched, the detection speed is important, as well as the model's accuracy. There is also a need to transfer as little data as possible from the drone to the tablet control console. For this mode of use, the most suitable would be a network resolution of 416 × 416 at which the model has 10 fps with an accuracy of only 2% less than the same model at a network resolution of 832 × 832 for larger objects that are likely to be directly detectable in the field, and about 10% less for other cases.

Off-line detection is performed on the recorded materials using a computer with a higher power CPU + GPU. The required detection speed is not crucial in that case, especially if we compare it with about 25 seconds needed for a human video analyst to detect a victim on drone images [50]. In that case, the best model is the one that achieves greater accuracy, and that would be with a resolution of 832 × 832 or 608 × 608 since the differences in performance are negligible.

D. DETECTION RESULTS AS A FUNCTION OF TP-FP

We mentioned earlier that in search and rescue operations, the crucial is the accuracy of detection and the speed of finding the missing person. Therefore, it is important to build a model with a few false detections (FP) as possible because they consume human resources and take valuable time.

For this reason, we introduced additional metrics that we called ROpti, computed as the ratio of the difference between true (TP) and false positive (FP) detections and possible detections (TP+FN) in the dataset:

$$ROpti = \frac{(TP - FP)}{(TP + FN)} \tag{6}$$

For perfect precision (no false positive), ROpti is equal to recall, and with perfect recall (no false negative), ROpti is equal to 1, and this is a perfect score. As the number of FPs grows, ROpti decreases. In case TP is equal to FP, then ROpti is equal to zero, ROpti becomes negative, while TP is less than FP.

The detection results considering ROpti measure, e.g., true and false-positive detections out of a total of 2611 objects for different network resolutions with default thresh of 0.25, are given in Table 5.

Considering the ROpti measure, the resolution 832 × 832 surpass all other tested network resolutions as it has only 88 FP and the highest ROpti value of 0.928.

Therefore, we propose a model for detection of persons in SAR actions shown in Figure 13, with 416 × 416 network resolution for on-board detections on videos received from the drone to the control console-tablet (or using RTMP server to live stream from a drone to laptop) and 832 × 832 for further off-line analyses.

TABLE 5. YOLOv4 (SARD) detection results in terms of a true positive, false negative, and ROpti for different network resolutions.

Model	TP	FP	ROpti
320x320	2088	295	0.687
416x416	2346	184	0.828
512x512	2438	147	0.877
608x608	2485	133	0.901
832x832	2512	**88**	**0.928**
1024x1024	2491	102	0.915

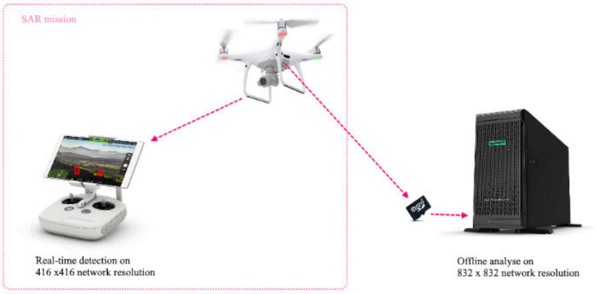
SAR mission
Real-time detection on 416 x416 network resolution
Offline analyse on 832 x 832 network resolution

FIGURE 13. Proposed model for person detection in SAR mission.

E. DETECTION RESULTS AS A FUNCTION OF CONFIDENCE (THRESH) VALUE

In the case of searching for a particular object, any detection that recognizes the object and its location can be taken as a positive detection, regardless of the percentage of the IoU between the ground truth bounding box and the detected bounding box, i.e., precision in terms of the bounding box which is the smallest closure of the object is not so important, so in our case, an IoU of 10% is also acceptable. Decreasing the IOU value and confidence value of the model affects the accuracy of the detection, and this, in turn, affects the model usability for automatic detection of persons in SAR missions. On the other hand, the goal is to achieve as few false-positive detections as possible, i.e., achieve the highest possible ROpti value, so the limit to which it is still effective to decrease the confidence or threshold value needs to be determined.

By default, YOLO detects objects with a confidence (threshold) of 0.25 or more. This value directly affects the number of marked objects in the set, so we examined how the thresh value changes affect the ROpti value.

Figure 14. shows detection results for thresh in the range from 0.10 to 0.90 with a step of 0.10 in two network resolutions 832 × 832 and 416 × 416. The best results were achieved when the network resolution was 832 × 832, and the thresh was 40%, so this is the configuration we would recommend for the model for person detection in SAR scenes.

With a network resolution of 416 × 416, results are 1 to 8% worse than with 832 × 832, but with 2.5 times shorten detection times, so this network setting with thresh = 0.10 can be recommended as a reasonable solution in on-board online

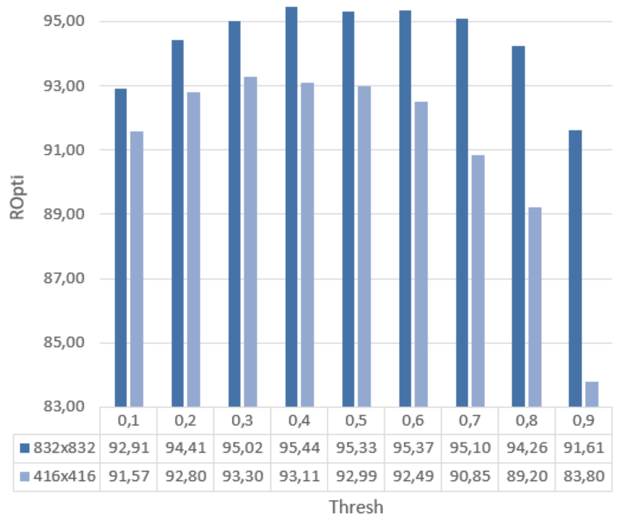

FIGURE 14. The ROpti value for the YOLOv4(SARD) model with network resolutions 832 × 832 and 416 × 416 considering different confidence values (thresh).

detection when speed and a small amount of data are important. To improve the ROpti results and reduce the number of FP detections in real-time, the drone pilot can "remove" false-positive detections by lowering the drone to a lower altitude when necessary to capture larger objects.

F. DETECTION DEPENDENCE OF RECORDING HEIGHT

The altitude at which the drone is located plays a major role in detecting people in aerial photographs. The higher the altitudes at which the drone flies, the smaller the captured material and fewer pixels are used to represent them. However, at higher altitudes, the drone can capture a larger terrain area. In the case of SAR operations, it makes no sense to increase the flight altitude above the level at which persons can be detected. Obviously, it is easier to detect a person represented in the image with a larger number of pixels, so it will be more suitable for detecting people when the drone is flying at a lower altitude. But this extends the time required to cover the target search area. Therefore, the goal is to determine the highest possible altitude at which the drone should fly so that people on the scene can still be detected automatically by a detector.

Flight altitude recommendations depend on the number of pixels in the camera and the lenses used, and the area being monitored. With DJI Phantom 4 Advance, we record images at a resolution of 5472 × 3078 px with a camera angle of 90°, Field of View (FOV) by specification is 84°.

In the experiment, we took images of two persons (women and a boy) at different heights (15 m, 30 m, 45 m, 60 m, and 75 m). Figure 15. shows detection results, and it can be seen that all detections are accurate at the height of 30 m. Therefore, considering that there are different specifications of drone cameras, we suggest that the drone flies at a height from which it can capture images in which people occupy an area of 100 × 100 px.

TABLE 6. Comparative results of YOLOv4(sard) and YOLOv4(sard+corr) on corr dataset and its parts concerning different weather conditions (%).

Model	TEST	AP	IMP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
YOLOv4(SARD)	SARD	**61**		**96**	**71**	**45**	**66**	**73**
	Corr	35.5		65.7	34.7	20.8	39.4	53.3
	Corr-Snow	32.5		59.0	32.9	18.0	35.8	56.9
	Corr-Fog	30.2		55.0	30.4	22.5	32.2	40.4
	Corr-Frost	35.9		62.9	37.1	22.5	39.3	50.8
	Corr-M. blur	31.6		67.8	24.7	14.7	35.1	58.1
YOLOv4(SARD+Corr)	SARD	**59.4**	**-1.6**	**94.7**	**67.4**	**42.2**	**64.7**	**72.8**
	Corr	51.9	16.4	89.5	53.0	32.7	57.1	69.0
	Corr-Snow	50.3	17.8	88.5	51.5	33.4	54.7	65.1
	Corr-Fog	54.7	24.5	91.6	60.2	38.2	59.5	65.3
	Corr-Frost	53.1	18.2	90.5	57.8	36.7	57.5	66.5
	Corr-M. blur	43.9	12.3	84.9	41.0	24.4	49.4	61.6

G. ROBUSTNESS TO WEATHER CONDITIONS AND MOTION BLUR

To test the YOLOv4 (SARD) model's performance with a network resolution of 832 × 832 in conditions that can occur in search and rescue operations, we have tested the model's performance on the Corr test set. The Corr test set includes images with various weather conditions such as snow, fog, frost (Corr-Snow, Corr-Fog, Corr-Frost), and motion blur that may occur during recording, e.g., due to moving and camera shake (Corr-M. Blur).

The examination results are given in Table 6 in terms of average precision (AP), respecting IoU precision and the object size. The results show several important facts.

A significant decrease in detection performance occurred in the case of testing on images with bad weather conditions and blur images that did not exist in the training set. e.g., the decrease in AP_{50} was from 96% on SARD set to 66% on the Corr dataset that contains the same images but with bad weather conditions. The drop in performance is not the same for all bad weather conditions, e.g., AP_{50} is 59% for snow, 55% for fog, 63% for frost, and 68% for motion blur.

To improve the YOLOv4 (SARD) model results in bad weather, we additionally trained the model on the Corr train set, referred to as the YOLOv4 (SARD+Corr) model. The YOLOv4 (SARD+Corr) model achieves similar or slightly worse results than the YOLOv4 (SARD) model on the SARD test set and significantly better results on the Corr test set. Detection results are presented in Table 6.

Examples of detection results of the YOLOv4 (SARD+Corr) model for all different weather categories and motion blur are shown in Figure 16.

H. TRANSFER LEARNING STRATEGY

To improve model training, we wanted to investigate further how different transfer learning strategies regarding different combinations of datasets affect the detection result. We exam-

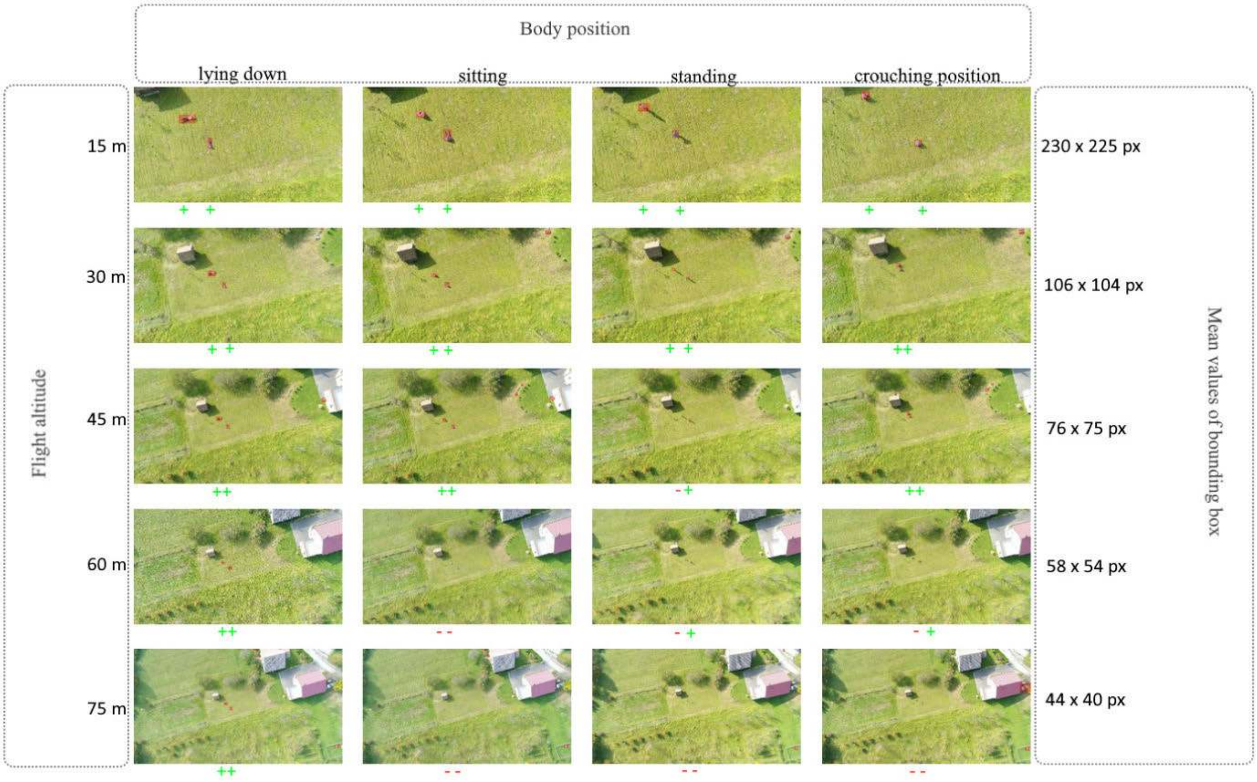

FIGURE 15. Detection results on different drone heights (15 m, 30 m, 45 m, 60 m, and 75 m) show that below or equal to 30 m of height all detections are accurate.

TABLE 7. Detection results for YOLOv4 model trained on different sets and tested on sard test set and mixture of sard and visdrone2000 test sets (%).

TRAIN	TEST	AP	IMP	AP$_{50}$	AP$_{75}$	AP$_{S}$	AP$_{M}$	AP$_{L}$	AR$_{S}$	AR$_{M}$	AR$_{L}$	ROpti
SARD	SARD	61.3	37.90	95.7	71.1	45.0	66.4	72.6	52.3	72.1	**77.8**	92.8
VisDrone2000	SARD	18.9	-4.5	33.2	20.5	13.2	21.6	17.4	14.8	23.6	19.5	30.0
Corr	SARD	54.9	31.5	90.5	61.9	35.1	61.3	66.8	42.8	67.2	73.1	85.9
S+V	SARD	22.8	-0.6	41.7	23.7	16.4	25.5	23.0	19.1	28.3	25.8	35.4
V+S	SARD	61.3	37.9	95.8	70.6	46.2	66.3	71.5	53.1	71.8	76.6	**93.5**
V+C+S	SARD	**62.0**	**38.6**	**95.9**	**71.9**	**46.9**	**66.9**	**72.1**	**53.9**	**73.2**	77.1	92.6
SC	SARD	59.4	36.0	94.7	67.4	42.2	64.7	72.8	49.5	69.9	76.7	90.9
SV	SARD	55.4	32.0	92.5	60.8	38.4	60.6	67.1	46.0	66.0	71.7	86.2
SVC	SARD	56.4	33.0	93.6	63.1	39.9	61.5	67.3	47.5	66.8	71.7	88.2
S+V	SV	23.7	8.3	52.9	17.4	21.0	32.9	24.8	28.6	38.3	28.1	33.1
V+S	SV	18.7	3.3	35.7	17.5	9.9	48.2	**96.9**	12.9	53.6	74.6	26.3
SV	SV	**29.7**	**14.3**	**61.7**	**24.6**	**22.3**	**52.4**	65.8	**30.0**	**58.3**	70.3	**40.3**

ined the possibility of learning the models successively, on one training set and then on the other, taking into account the order of sets used for training or in one step but using the images taken from both training sets.

The goal was to get the best possible results of the YOLOv4 model at the SARD test set. Firstly, to train the model, SARD, VisDrone, and Corr sets were used separately, and then combinations of them. The results achieved by training the models at different training sets using one by another in a different order, or mixed, are shown in Table 7. In addition to the accuracy values, the improvement (Imp) of the model concerning the initial weights (original model) and ROpti value are also shown.

The S+V means that the model is first trained on the SARD train set and then on the VisDrone train set, V+S that it is first trained on VisDrone, then on the SARD train set, and for V+C+S, the model is trained on VisDrone2000, and then on Corr and finally on SARD train set.

The SV refers to a mixture of SARD and VisDrone2000 train sets when images are used randomly from

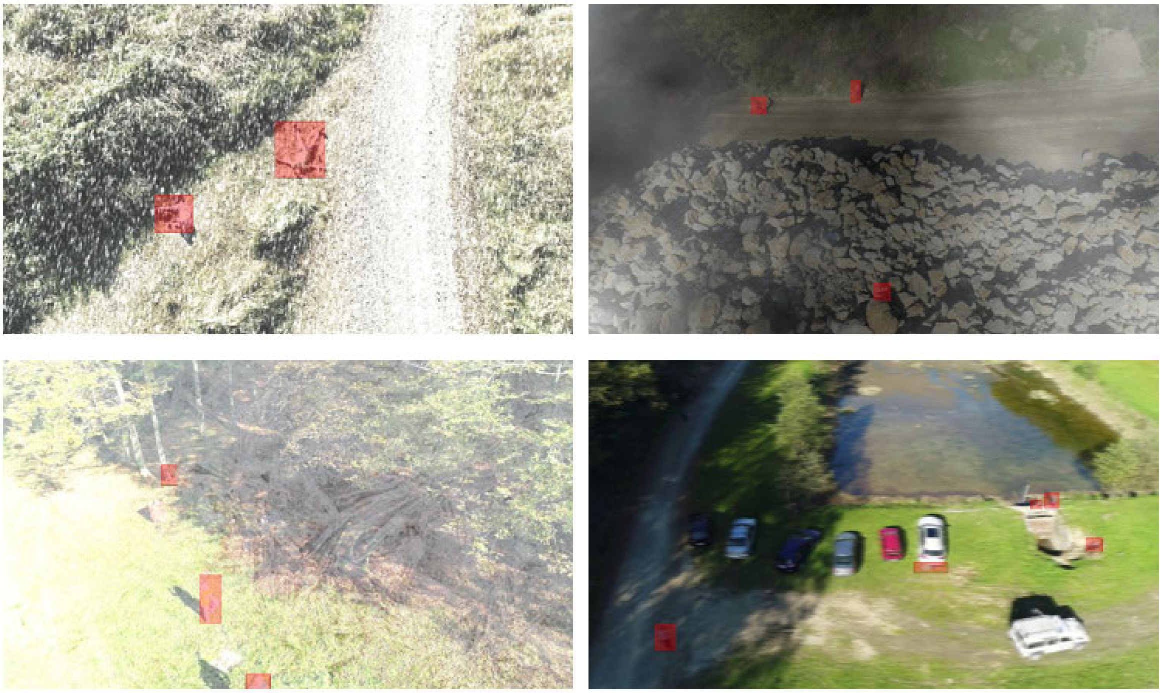

FIGURE 16. Example of detection of YOLOv4 (SARD+Corr) model. Up-left snow, up-right fog, down left ice, down right motion blur.

both of them for training, while for testing purposes, the SV test refers to a combination of SARD and VisDrone test images. Similarly, the SC is a mixture of SARD and Corr set, and SVC is a mixture of SARD, VisDrone2000, and Corr test set.

It can be observed that the improvement of the results is achieved in the case when the last trained set is the closest to the tested set (S+V vs. V+S). Also, training models with more data from multiple sets ultimately contribute to a better result, especially if the sets are compatible, i.e., contain similar images. In this experiment, the best results (AP 62%, AP_S 46.9%) were achieved when learning the model on sets in the order V+C+S, but this is an improvement of only 1% than in the case when the model was trained only on the SARD set of images, which is certainly not a significant improvement.

The V+S model achieves the same results on the SARD test set as the model trained only on the SARD set for all cases except for smaller objects. The V+S model gets better results since a larger number of smaller objects fromVisDrone2000 train set were included in the V+S training set.

Training the model on data from a mixture of sets (SV, SC, SVC) had given worse results than when the model was trained only on the SARD set or on a series of sets ending with SARD so that the weights of the model are last adjusted to the set being tested.

The same conclusion applies when the model is tested on images from multiple sets, e.g., the SV set. The best results are achieved when the model is trained on a particular combination of these sets.

V. CONCLUSION

The ability to detect people on drone images using computer vision methods automatically is a significant help in SAR operations. In this paper, we explored the state-of-the-art person detectors in drone images and proposed a model for detecting persons in SAR actions.

We have re-trained and tested CNN-based object detectors, Cascade R-CNN, Faster R-CNN, RetinaNet, and YOLOv4 on selected drone images in the VisDrone set and our custom-made set of SAR-s scenes.

YOLOv4 has achieved the best detection performances on the SARD dataset in terms of average precision (AP) considering IoU precision and the object size as well as the least false detection (FP), so it was further used in the experiment, referred to as YOLOv4 (SARD). When the model was trained on 512 × 512 image resolution, the best AP of 60% was achieved for a network resolution of 832 × 832.

In SAR operations, the model must have a few false detections (FPs) as possible that resources are not wasted unnecessarily, so we introduced an additional metric called ROpti, calculated as the ratio of the difference between true and false positive detections and possible detections in a dataset.

In searching for a missing person, the most important thing is that the detector locates that person, and it is less important how accurate the detection is. We experimentally selected parameters as a trade-off between accuracy and recall so that the model can be helpful in SAR actions. The results showed that the YOLOv4 (SARD) model in a network resolution of 832 × 832, IoU = 0.1, achieved the best results for thresh of 0.4, namely AP of 97.15% (TP: 2538, FP: 46).

The model's robustness was tested on images with artificially generated bad weather conditions and image blur, and the results show a severe decrease in AP in more than 30%. After the model was also trained on the part of the images with bad weather effects, the model achieves significantly better results (AP 50.3% for snow, 54.7% fog, 53.1% ice, 43.8% motion blur).

In future work, the plan is to use a thermal camera to increase detection performance and develop a model for recognizing human activity (running, walking, standing, sitting, lying down) and tracking people in SAR scenes.

REFERENCES

[1] M. Šuperina and K. Pogačić, "Ucestalost Hrvatske gorske službe spašavanja u traganju za nestalim osobama," *Policija I Sigurnost*, vol. 16, nos. 3–4, pp. 235–256, 2007.

[2] K. Butorac, M. Šuperina, and L. Mikšaj-Todorović, "Developing police search strategies for elderly missing persons in Croatia," *Varstvoslovje*, vol. 17, no. 1, pp. 24–45, 2015.

[3] R. J. Koester, *Lost Person Behavior: A Search and Rescue*. Charlottesville, VA, USA: DBS Productions LLC, 2008.

[4] A. S. Laliberte and A. Rango, "Texture and scale in object-based analysis of subdecimeter resolution unmanned aerial vehicle (UAV) imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 3, pp. 761–770, Mar. 2009.

[5] G. Pajares, "Overview and current status of remote sensing applications based on unmanned aerial vehicles (UAVs)," *Photogrammetric Eng. Remote Sens.*, vol. 81, no. 4, pp. 281–330, Apr. 2015.

[6] A. Bhardwaj, L. Sam, Akanksha, F. J. Martín-Torres, and R. Kumar, "UAVs as remote sensing platform in glaciology: Present applications and future prospects," *Remote Sens. Environ.*, vol. 175, pp. 196–204, Mar. 2016.

[7] S. Harwin and A. Lucieer, "Assessing the accuracy of georeferenced point clouds produced via multi-view stereopsis from unmanned aerial vehicle (UAV) imagery," *Remote Sens.*, vol. 4, no. 6, pp. 1573–1599, May 2012.

[8] S. Yahyanejad and B. Rinner, "A fast and mobile system for registration of low-altitude visual and thermal aerial images using multiple small-scale UAVs," *ISPRS J. Photogramm. Remote Sens.*, vol. 104, pp. 189–202, Jun. 2015.

[9] N. Tijtgat, W. Van Ranst, B. Volckaert, T. Goedeme, and F. De Turck, "Embedded real-time object detection for a UAV warning system," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 2110–2118.

[10] X. X. Zhu, D. Tuia, L. Mou, G.-S. Xia, L. Zhang, F. Xu, and F. Fraundorfer, "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 8–36, Dec. 2017.

[11] G. Milani, M. Volpi, D. Tonolla, M. Doering, C. Robinson, M. Kneubühler, and M. Schaepman, "Robust quantification of riverine land cover dynamics by high-resolution remote sensing," *Remote Sens. Environ.*, vol. 217, pp. 491–505, Nov. 2018.

[12] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[13] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high-quality object detection," in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, Jun. 2018, pp. 6154–6162.

[14] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[15] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single shot multi-box detector," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 21–37.

[16] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: http://arxiv.org/abs/1804.02767

[17] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. ECCV*, 2014, pp. 740–755.

[18] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes challenge: A retrospective," *Int. J. Comput. Vis.*, vol. 111, no. 1, pp. 98–136, Jan. 2015.

[19] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, Dec. 2015, Art. no. 211252.

[20] M. Pobar and M. Ivasic-Kos, "Mask R-CNN and optical flow based method for detection and marking of handball actions," in *Proc. 11th Int. Congr. Image Signal Process., Biomed. Eng. Informat. (CISP-BMEI)*, Oct. 2018, pp. 1–6.

[21] M. Ivasic-Kos and M. Pobar, "Building a labeled dataset for recognition of handball actions using mask R-CNN and STIPS," in *Proc. 7th Eur. Workshop Vis. Inf. Process. (EUVIP)*, Nov. 2018, pp. 1–6.

[22] M. Ivašić-Kos, M. Krišto, and M. Pobar, "Human detection in thermal imaging using YOLO," in *Proc. 5th Int. Conf. Comput. Technol. Appl. (ICCTA)*, New York, NY, USA, Apr. 2019, pp. 20–24.

[23] M. Ivasic-Kos, M. Kristo, and M. Pobar, "Person detection in thermal videos using YOLO," in *Proc. SAI Intell. Syst. Conf.* Cham, Switzerland: Springer, 2019.

[24] P. Zhu, L. Wen, D. Du, X. Bian, Q. Hu, and H. Ling, "Vision meets drones: Past, present, and future," 2020, *arXiv:2001.06303*. [Online]. Available: https://arxiv.org/abs/2001.06303

[25] M. Barekatain, M. Martí, H.-F. Shih, S. Murray, K. Nakayama, Y. Matsuo, and H. Prendinger, "Okutama-action: An aerial view video dataset for concurrent human action detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 28–35.

[26] D. Du, Y. Qi, H. Yu, Y. Yang, K. Duan, G. Li, and Q. Tian, "The unmanned aerial vehicle benchmark: Object detection and tracking," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 370–386.

[27] M.-R. Hsieh, Y.-L. Lin, and W. H. Hsu, "Drone-based object counting by spatially regularized regional proposal network," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 4145–4153.

[28] A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese, "Learning social etiquette: Human trajectory understanding in crowded scenes," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 549–565.

[29] M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for UAV tracking," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 445–461.

[30] L. Mou, Y. Hua, P. Jin, and X. X. Zhu, "ERA: A dataset and deep learning benchmark for event recognition in aerial videos," 2020, *arXiv:2001.11394*. [Online]. Available: http://arxiv.org/abs/2001.11394

[31] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[33] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1492–1500.

[34] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: http://arxiv.org/abs/1704.04861

[35] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 4510–4520.

[36] A. Howard, M. Sandler, B. Chen, W. Wang, L.-C. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, R. Pang, H. Adam, and Q. Le, "Searching for MobileNetV3," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1314–1324.

[37] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, A. Kolesnikov, T. Duerig, and V. Ferrari, "The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale," 2018, *arXiv:1811.00982*. [Online]. Available: http://arxiv.org/abs/1811.00982

[38] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[39] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7263–7271.

[40] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*. [Online]. Available: http://arxiv.org/abs/2004.10934

[41] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 580–587.

[42] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[43] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 379–387.

[44] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.

[45] S. N. A. M. Ghazali, H. A. Anuar, S. N. A. S. Zakaria, and Z. Yusoff, "Determining position of target subjects in maritime search and rescue (MSAR) operations using rotary wing unmanned aerial vehicles (UAVs)," in *Proc. Int. Conf. Inf. Commun. Technol. (ICICTM)*, 2016, pp. 1–4.

[46] P. Doherty and P. Rudol, "A UAV search and rescue scenario with human body detection and geolocalization," in *Proc. Australas. Joint Conf. Artif. Intell.* Berlin, Germany: Springer, 2007, pp. 1–13.

[47] M. A. Goodrich, B. S. Morse, C. Engh, J. L. Cooper, and J. A. Adams, "Towards using unmanned aerial vehicles (UAVs) in wilderness search and rescue: Lessons from field trials," *Interact. Stud.*, vol. 10, no. 3, pp. 453–478, Dec. 2009.

[48] S. Waharte and N. Trigoni, "Supporting search and rescue operations with UAVs," in *Proc. Int. Conf. Emerg. Secur. Technol.*, Sep. 2010, pp. 142–147.

[49] C. A. Baker, S. Ramchurn, W. T. Teacy, and N. R. Jennings, "Planning search and rescue missions for UAV teams," in *Proc. 22nd Eur. Conf. Artif. Intell.*, 2016, pp. 1777–1778.

[50] K. Yun, L. Nguyen, T. Nguyen, D. Kim, S. Eldin, A. Huyen, and E. Chow, "Small target detection for search and rescue operations using distributed deep learning and synthetic data generation," *Proc. SPIE*, vol. 10995, May 2019, Art. no. 1099507.

[51] A. Gallego, A. Pertusa, P. Gil, and R. B. Fisher, "Detection of bodies in maritime rescue operations using unmanned aerial vehicles with multispectral cameras," *J. Field Robot.*, vol. 36, no. 4, pp. 782–796, Jun. 2019.

[52] R. Geraldes, A. Gonçalves, T. Lai, M. Villerabel, W. Deng, A. Salta, K. Nakayama, Y. Matsuo, and H. Prendinger, "UAV-based situational awareness system using deep learning," *IEEE Access*, vol. 7, pp. 122583–122594, 2019.

[53] S. O. Murphy, C. Sreenan, and K. N. Brown, "Autonomous unmanned aerial vehicle for search and rescue using software defined radio," in *Proc. IEEE 89th Veh. Technol. Conf. (VTC-Spring)*, Apr. 2019, pp. 1–6.

[54] E. Lygouras, N. Santavas, A. Taitzoglou, K. Tarchanidis, A. Mitropoulos, and A. Gasteratos, "Unsupervised human detection with an embedded vision system on a fully autonomous UAV for search and rescue operations," *Sensors*, vol. 19, no. 16, p. 3542, Aug. 2019.

[55] F. S. Leira, T. A. Johansen, and T. I. Fossen, "Automatic detection, classification and tracking of objects in the ocean surface from UAVs using a thermal camera," in *Proc. IEEE Aerosp. Conf.*, Mar. 2015, pp. 1–10.

[56] J. Sun, B. Li, Y. Jiang, and C.-Y. Wen, "A camera-based target detection and positioning UAV system for search and rescue (SAR) purposes," *Sensors*, vol. 16, no. 11, p. 1778, Oct. 2016.

[57] Z. Kashino, G. Nejat, and B. Benhabib, "Aerial wilderness search and rescue with ground support," *J. Intell. Robotic Syst.*, vol. 99, pp. 147–163, 2020.

[58] T. Marasović and V. Papić, "Person classification from aerial imagery using local convolutional neural network features," *Int. J. Remote Sens.*, vol. 40, no. 24, pp. 9084–9102, 2019.

[59] A. Al-Kaff, M. Gómez-Silva, F. Moreno, A. de la Escalera, and J. Armingol, "An appearance-based tracking algorithm for aerial search and rescue purposes," *Sensors*, vol. 19, no. 3, p. 652, Feb. 2019.

[60] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 4724–4732.

[61] S. Sambolek and M. Ivasic-Kos, "Detection of toy soldiers taken from a bird's perspective using convolutional neural networks," in *ICT Innovations*, 1st ed. Oct. 2019, pp. 13–26.

[62] M. Ivasic-Kos, I. Ipsic, and S. Ribaric, "A knowledge-based multi-layered image annotation system," *Expert Syst. Appl.*, vol. 42, no. 24, pp. 9539–9553, Dec. 2015.

[63] K. Li, G. Wan, G. Cheng, L. Meng, and J. Han, "Object detection in optical remote sensing images: A survey and a new benchmark," *ISPRS J. Photogramm. Remote Sens.*, vol. 159, pp. 296–307, Jan. 2020.

[64] C. Michaelis, B. Mitzkus, R. Geirhos, E. Rusak, O. Bringmann, A. S. Ecker, M. Bethge, and W. Brendel, "Benchmarking robustness in object detection: Autonomous driving when winter is coming," 2019, *arXiv:1907.07484*. [Online]. Available: http://arxiv.org/abs/1907.07484

[65] *Faster R-CNN Mmdetection Models*. Accessed: Jan. 20, 2021. [Online]. Available: https://github.com/open-mmlab/mmdetection/tree/master/configs/faster_rcnn

[66] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "CSPNet: A new backbone that can enhance learning capability of CNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 390–391.

[67] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

[68] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 8759–8768.

[69] *RetinaNet Mmdetection Models*. Accessed: Jan. 20, 2021. [Online]. Available: https://github.com/open-mmlab/mmdetection/tree/master/configs/retinanet

[70] *Cascade R-CNN Mmdetection Models*. Accessed: Jan. 20, 2021. [Online]. Available: https://github.com/open-mmlab/mmdetection/tree/master/configs/cascade_rcnn

[71] M. Kristo, M. Ivasic-Kos, and M. Pobar, "Thermal object detection in difficult weather conditions using YOLO," *IEEE Access*, vol. 8, pp. 125459–125476, 2020.

[72] D. Du *et al.*, "VisDrone-DET2019: The vision meets drone object detection in image challenge results," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Seoul, South Korea, 2019, pp. 213–226, doi: 10.1109/ICCVW.2019.00030.

[73] S. Sambolek and M. Ivašić-Kos, "Detecting objects in drone imagery: A brief overview of recent progress," in *Proc. Mipro*, 2020, pp. 1052–1057.

[74] *Yolo-v4 and Yolo-v3/v2 for Windows and Linux*. Accessed: Jan. 20, 2021. [Online]. Available: https://github.com/AlexeyAB/darknet

[75] M. Pobar and M. Ivasic-Kos, "Active player detection in handball scenes based on activity measures," *Sensors*, vol. 20, no. 5, p. 1475, Mar. 2020.

SASA SAMBOLEK graduated in physics and computer science from the Department of Physics, Faculty of Science, University of Zagreb, in 2009. He is currently pursuing the Ph.D. degree in informatics with the Department of Informatics, University of Rijeka. He is currently working with High School Tina Ujevica. His research interests include artificial intelligence and computer vision. His doctoral dissertation aims to develop algorithms that could be used to detect persons on aerial photographs during search and rescue operations.

MARINA IVASIC-KOS received the Ph.D. degree in computer science from the Faculty of Electrical Engineering and Computing, University of Zagreb. She is currently an Associate Professor and the Head of the Department of Informatics, University of Rijeka, and the Head of the Laboratory for Computer Vision, Virtual and Augmented Reality, Centre for Artificial Intelligence, University of Rijeka. She is also the Leader of a National Research Project dealing with automatic recognition of actions in sports and a Researcher at a project dealing with crowd analysis in surveillance. She was involved in numerous business and research projects. Her research interests include AI, computer vision, knowledge representation, and soft computing. She is a Technical Committee Member and a Reviewer for numerous scientific conferences and high-cited journals like IEEE TRANSACTIONS ON FUZZY SYSTEMS (TFS), IEEE ACCESS, *Signal Processing*, and PR and ESWA (Elsevier).