

YOLOv10s based human victim detection in cluttered urban environments with a crawler type rescue robot

R. Farkhetdinov¹, B. Abbyasov¹, A. Eryomin¹, A. Dobrokvashina¹, M. Svinin², E. Magid^{1,3}

¹ Institute of Information Technology and Intelligent Systems, Kazan Federal University, Kremlevskaya St. 35, Kazan, 420008, Russian Federation;

² Graduate School of Information Science and Engineering, Ritsumeikan University, 2-150 Iwakura-cho, Ibaraki, 5678570, Osaka, Japan;

³ School of Electronic Engineering, Tikhonov Moscow Institute of Electronics and Mathematics, HSE University, 34 Tallinskaya St., Moscow, 123592, Russian Federation

Abstract

Rescue robots are widely utilized in search and rescue operations to enhance operations' efficiency. To reduce operators' load a robot could perform some functions automatically, including victims' detection. This paper introduces a robot operating system based victim detection framework for Servosila Engineer crawler rescue robot with four cameras. The victim detection algorithm employs video stream frames from a single camera and a trained YOLOv10s neural network that detects human body parts within a picture of a cluttered urban environment. To train the YOLOv10s model, a human body dataset of 15068 images was created by combining an existing dataset with a new dataset collected with the robot's camera. The model was trained to detect a person and his/her body parts: a head, a hand, and a foot. The study analyzed an impact of a distance between the robot and a human victim in cluttered environments on detection accuracy. The algorithm showed acceptable performance in validation experiments with three human participants under artificial lighting conditions when the robot's camera was positioned within 50 to 200 cm distance from a cluttered area. Within this distance, an Average Precision (AP) of 0.75, 0.91, and 0.73 was achieved for the head, hand, and foot classes respectively; the AP rapidly degraded with distance. The experiments showed that hand class objects were detected more reliably compared to other objects across all three intervals. Unlike prior approaches that employed high-end hardware or multiple cameras, our system achieved a reasonable accuracy using a single camera and low-power onboard computing.

Keywords: computer vision, deep learning, human detection, mobile robot, search and rescue.

Citation: Farkhetdinov R, Abbyasov B, Eryomin A, Dobrokvashina A, Svinin M, Magid E. YOLOv10s based human victim detection in cluttered urban environments with a crawler type rescue robot. *Computer Optics* 2026; 50 (2): 1710. DOI: 10.18287/COJ1710.

Introduction

Natural and human-caused disasters in dense populated areas lead to catastrophic consequences, including loss of human lives and severe infrastructure damages. The disasters, e.g., earthquakes or landslides, cause building collapses and devastation, while leaving multiple victims trapped under constructions' debris and creating hazardous environments for rescue teams [1]. Addressing these challenges requires advanced solutions that apply robotics and automation for rescue operation challenges [2].

Rescue robotics is a promising field dedicated to developing robotic technologies for search and rescue (SAR) missions. One of key application domains is urban SAR (USAR) [3], which focuses on locating and assisting victims within urban disaster zones [4]. Such areas often consist of extensive debris from collapsed structures that bury people within and do not allow escaping without external assistance. Deploying specialized robots in USAR environments minimizes risks to human rescuers and accelerates rescue operations [5]. Crawler robots are widely utilized in SAR operations to minimize risks for human rescuers and to enhance operations' efficiency. Typically, two human operators control a single rescue robot: the first operator performs local motion planning and teleoperated navigation, while the second explores an environment through robot's cameras in order to produce semantic mapping, detection of victims and potential dangers (e.g., open fire or chemical leaks), structures' conditions evaluation and other high-level analysis tasks. To reduce operators' cognitive load and human factor caused errors it would be beneficial to maximize a number of functions that could be performed automatically by the robot, e.g., an automatic detection of potential victims, which appeared within a camera's field of view.

Rescue robots rely on their perception systems to detect, identify, and locate victims in disaster environments. Typical rescue robots are equipped with various cameras that capture sensory data from surroundings [6] and process them using recognition and detection algorithms to identify signs of human presence, including motion and body heat. BING [7], RIHOG [8], and *skin objectness windows* based approaches [9] detect human presence by identifying potential body regions within an image. A main challenge comes from a complex, articulated structure of a human body, which makes detection even more difficult when the body is covered in dust or trapped under debris [8]. Object detection methods

based on a Convolutional Neural Network (CNN) can help overcome these challenges by leveraging their ability to learn complex patterns and features from image data [10].

This paper presents an approach for camera-based human victims' detection. The solution involves a post-trained CNN model for detecting victims beneath artificially constructed rubble. The YOLOv10s [11 – 12] was trained on in-lab custom SAR images' dataset. Finally, a Russian Servosila Engineer robot [13] was employed in a laboratory environment for human detection experiments.

The rest of the paper is structured as follows. An overview of a related work is introduced in Section 1. Section 2 describes a system setup. Section 3 presents datasets, a selection of a neural network model and a detection pipeline. Section 4 outlines an experimental setup and results of experiments. Section 5 discusses a rationale for performing victim detection with a single camera and our USAR dataset limitations.

1. Related work

The state-of-the-art techniques for human body detection in SAR focus on image processing. Cruz Ulloa et al. [14] conducted a comparative analysis of RGB, thermal, and multispectral vision systems for victim detection in SAR robotics. Their study emphasized advantages and limitations of each sensor type, highlighting effectiveness of an infrared spectrum for thermal cameras and the Near Infrared and Red Edge bands for multispectral cameras. The evaluation (performed using a quadrupedal robot) achieved detection precisions exceeding 92% and 86% for thermal and multispectral cameras, respectively. Zafar et al. [15] used YOLOv8 model to accurately detect human victims in rescue conditions using a single image. Their study presented an integrated approach that combines deep learning-based victim detection with gesture-based control for UGVs in SAR operations. They introduced the Meerkat Optimization Algorithm-Stacked Convolutional Neural Network-Bi-LSTM-GRU (MOA-SConv-Bi-LSTM-GRU) model, achieving high accuracy in hand gesture recognition, which enhanced UGV maneuverability in complex environments.

Huang et al. [16] proposed multistep search of human victims. YOLOv3 model pre-limited a location of a human body, which was then refined by thermal images, human voices and activities. Their FPGA-based system enabled efficient edge AI processing, reducing latency and power consumption compared to traditional cloud-based approaches. The proposed integration of UAVs and UGVs enhances accuracy and scalability of SAR operations by combining aerial and ground-based survivor detection. Louie and Nejat [17] considered a 3D scene extracted from a depth camera to find human geometric and skin region features. The Support Vector Machine (SVM) was used for classification of different body parts. Their methodology enhanced victim detection by identifying partially occluded body parts, which is crucial in cluttered USAR environments. The integration of 2D and 3D sensory data allowed for more accurate victim localization, reducing operator workload and improving rescue mission efficiency.

De Cubber et al. [18] developed the ICARUS system, an integrated set of unmanned aerial, ground, and sea vehicles designed to support SAR operations. These autonomous platforms were equipped with victim detection sensors and communicated via an ad hoc cognitive radio network to enhance their coordination. Cruz Ulloa et al. [19] explored an application of deep learning models trained on both real and synthetic datasets for victim detection in post-disaster environments. Their study evaluated effectiveness of models trained on real-world images, virtual reconstructions, and a combination of both. The results indicated that CNN models trained with synthetic data could be successfully generalize to real-world scenarios.

Morales et al. [20] introduced the UMA-SAR dataset – a multimodal dataset collected during real-world SAR training exercises. The dataset includes RGB and thermal infrared camera data, 3D LiDAR scans, inertial measurement unit (IMU) readings, and GPS coordinates, making it a valuable resource for SAR research. A key contribution of this work was a provision of data sequences covering structured and unstructured environments, enabling research in multispectral image fusion, object detection, and mapping. Manns et al. [21] explored advancements in software development for USAR robots, with a focus on enhancing autonomy and reliability. A key contribution was an integration of RGB, thermal, and depth imaging for victim localization, minimizing false positives through multimodal sensor fusion. Additionally, they introduced an innovative 3D ray-casting technique to improve accuracy of victim mapping in post-disaster environments.

Rafael et al. [22] introduced TXRob, a low-cost teleoperated robot designed for exploration in post-disaster environments. The robot was equipped with artificial vision, gas recognition sensors, and a track-based mobility system. TXRob incorporated a motion detection algorithm that identifies changes between consecutive frames and alerts an operator when a person is detected. Bahadori et al. [23] proposed a stereo vision-based approach for human body detection in mobile robotic surveillance. Their method classified detected objects as human or non-human by leveraging stereo vision and localization data. To reduce computational complexity, the approach filtered out known environmental features using a pre-built map, allowing the system to focus on novel objects. An integration of a stereo vision with a LADAR-based mapping enhanced detection accuracy.

Castillo and Chang [24] introduced a fast template matching approach that utilized silhouette and visible skin information for victim detection. The method employed twelve templates representing individuals lying horizontally on the ground. Experimental evaluations in long hallways with minimal office debris demonstrated system's sensitivity to both a person's shape and the surrounding environment. Due to its reliance on edge detection and template similarity

measures, the approach can only handle occlusions up to knees, limiting its effectiveness in more cluttered scenarios. Kleiner and Kummerle [25] proposed a method that integrated images from color and infrared cameras to detect victims based on color, motion, facial features, and heat signatures. A Markov Random Field model was employed to capture the dependencies between these features. The approach was successfully evaluated in laboratory settings designed to approximate a standard NIST arena [26].

2. Setup

Servosila Engineer is a Russian crawler-type mobile robot designed for SAR, reconnaissance, and hazardous environment operations (Fig. 1) [27 – 28]. In this study, the robot was utilized for validation of an object detection software module. The robot with Intel Core i7-3517UE (1.70 GHz) onboard CPU and 32 GB Solid-State Drive (SSD) is equipped with four cameras: a central camera with optical zoom, stereoscopic left and right cameras, and a rear camera. Each camera of the stereo pair provides a video stream with a frame resolution of 744×480 at 30 frames per second, using an 8-bit Bayer GRBG pixel format. The stereo pair consists of two front-facing wide-angle cameras operating in the visible spectrum, each with a horizontal field of view of 90 degrees. The cameras are located in the robot's head, which is positioned above an end effector of a 4 degrees of freedom manipulator.

The Servosila Engineer robot uses Robot Operating System (ROS) Noetic version as a middleware for data exchange between robot components [29]. ROS facilitates communication between the robot's sensors and actuators, enabling seamless data integration and real-time processing. It provides a modular framework that allows for easy customization and expansion of the robot's capabilities. Video streams from the cameras are incorporated into the ROS communication system and made available to perception modules for subsequent visual analysis. These streams can be accessed from a computer connected to the same network as the robot.

3. Methodology

3.1. Datasets

Neural network models are typically trained, validated, and tested on large publicly available datasets, such as PASCAL VOC [30] and MS COCO [31]. However, many specific tasks do not require a large number of object detection classes. Additionally, dataset collection can be time-consuming and costly, often making it the most expensive aspect of a neural network training.



Fig. 1. Servosila Engineer crawler-type mobile robot

Using the Roboflow tool [32], we performed dataset annotation, preprocessing to a uniform resolution of 640×640 , and augmentation. Through augmentation techniques (such as Gaussian blur application, brightness adjustment, rotations, shears, crops, and flips) the datasets were significantly expanded. The augmentation was applied to the training set only and did not affect the validation and test sets.

Our solution uses *foot*, *head*, *hand*, and *person* (an entire body) classes for object detection. For human detection algorithm, we used the following datasets:

- The SAR image dataset LIRS-USAR-set.v1 from the Laboratory of Intelligent Robotic Systems (LIRS) of Kazan Federal University (KFU) consists of 2395 images and 13313 annotations. The dataset includes images of simulated casualties in SAR scenarios, with people in various poses including people lying under simulated rubble. Students of KFU captured these images using various monocular cameras and annotated them manually using the Makesense tool. The original images had different resolutions prior to processing with the Roboflow tool: 4032×3024 pixel (34 images, 1.4%), 1280×1280 (369 images, 15.4%), 1280×960 (1965 images, 82.1%), and 1224×1224 (27 images, 1.1%). After augmentation, the dataset was expanded to 7185 images and 39939 annotations. This dataset was used as a training set;
- The custom dataset collected from the left camera of the Servosila Engineer robot's stereo pair (further referred as *CD.LCam*), consisting of 2043 images and 12118 annotations. The specifications of the camera are described in Section 2. All dataset images were captured directly from the robot's camera without any post-processing (before passing them to the Roboflow tool). This dataset was divided into three parts. A training set contained 1460 images and 8988 annotations, which was expanded to 7300 images and 44940 annotations after augmentation. A validation set contained 365 images and 1754 annotations, while a test set consisted of 218 images and 1376 annotations.

The training data of the augmented datasets were combined into a single dataset consisting of 15068 images and 88009 annotations. It is worth mentioning that CD.LCam has a greater value as images captured by the robot's camera are of relatively low quality and are more similar to the expected input of a real world situation. Therefore, the validation set and the test set consisted entirely of the CD.LCam images.

Despite a significant difference in image quality of the LIRS-USAR-set.v1 and CD.LCam datasets, the earlier was included into the training set due to expected benefits of model generalization. LIRS-USAR-set.v1 provides a substantial number of annotated images depicting human subjects in diverse poses and simulated USAR environments, which complements the more realistic yet lower-quality CD.LCam images. Combining these datasets helped the model to better capture structural and semantic features of a human body under a wide range of visual and environmental conditions, thereby improving its performance in real-world SAR scenarios.

The dataset images were captured at varying distances from objects, under different lighting conditions, and from different angles. Examples of the training data are shown in Fig. 2 and Fig. 3. The target objects in the images were highlighted with bounding boxes of specific colors: red (*foot*), purple (*head*), orange (*hand*), and yellow (*person*). The simulated rescue scenes in Fig. 2 involved wooden boards, buckets, a shovel, a cart, and cardboard boxes. Each scene contains a single *person* object, and its bounding box within the dataset always encapsulates all other parts of a body. Additionally, the top-left image contains one *head* and one *hand* objects; the bottom-left – one *hand* and two *foot* objects; the top-right and the bottom-right – one *head*, two *hand*, and two *foot* objects.

The video frames in Fig. 3 were obtained by the robot camera and contain tables, chairs, and armchairs. The top-left image contains one *head*, one *hand*, two *foot*, and two *person* objects; the top-right – two *hand* and one *person* objects; the bottom-left – two *head*, two *hand*, two *foot*, and two *person* objects; the bottom-right – one *head*, one *hand*, one *foot*, and one *person* objects. The robot's crawler appears in camera frames due to a head-mounted sensor configuration; in a default configuration, the intermediate joints fold in such a way that the front camera partially captures the robot's crawler platform.

Tab. 1 shows a number of labeled objects of each class in each set. A single image may contain multiple labeled objects.

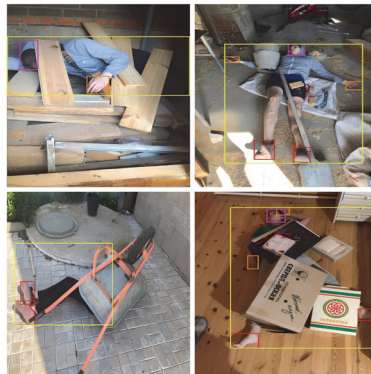


Fig. 2. Example of LIRS-USAR-set.v1 images used for the training stage

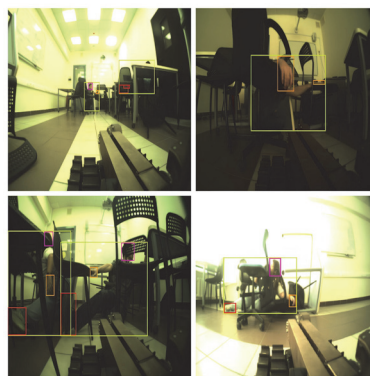


Fig. 3. Example of the CD.LCam images from the Servosila Engineer robot used for the training stage

3.2. Object detection model

The You Only Look Once (YOLO) framework is a widely used deep learning-based approach for real-time object detection. Unlike traditional two-stage object detectors, YOLO performs object localization and classification in a single forward pass, which significantly increases inference speed and maintains competitive accuracy [33]. The relatively recent generation, YOLOv10, offers an enhanced feature extraction and improved trade-offs between accuracy and speed [11 – 12]. In our approach, three versions of YOLOv10 were selected for evaluation: YOLOv10n (nano),

YOLOv10s (small), and YOLOv10m (medium). These models provide different balances between computational efficiency and detection accuracy. Additionally, they are among the fastest versions of this generation, making them well-suited for real-time object detection on low-power devices.

Tab. 1. Number of labeled objects of each class in each set

Class	Train	Validation	Test
Foot	26967	636	502
Head	15993	189	227
Hand	23679	564	316
Person	18240	365	331

Fig. 4 illustrates training results of the YOLOv10n (top row), YOLOv10s (middle row), and YOLOv10m (bottom row) models over 20 epochs, depicting loss functions for the validation set of the dataset, which is described in detail in Subsection 3.1. Each row presents three validation loss components: a box loss (blue), representing an error in predicted bounding box coordinates; a classification loss (orange), measuring an error in an object class prediction; and a distribution focal loss (green), capturing quality of a bounding box localization through a refined distribution-based regression. Since the loss functions did not exhibit a significant decrease approaching 20 epochs, this number was determined to be sufficient for training. The YOLO CNN family implementations provided by the Ultralytics library were trained using the default input parameter profile. A weight file with the best training results from each model was exported to the OpenVINO format [34 – 35] for further comparison. The export process involved first converting the model to the ONNX format, followed by converting it from ONNX to the OpenVINO format using the OpenVINO toolkit. The models were evaluated based on two criteria: object detection accuracy and speed.

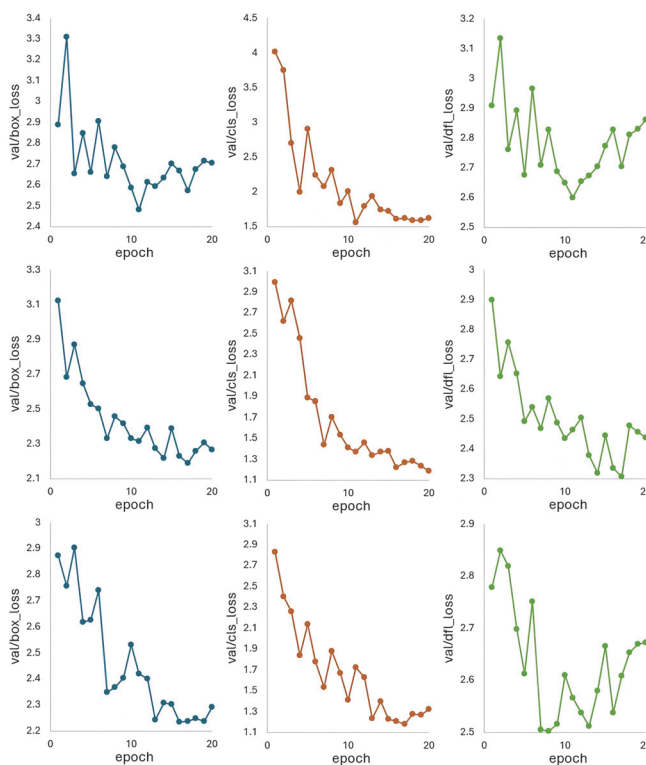


Fig. 4. The training results of the models over 20 epochs: (top) YOLOv10n; (middle) YOLOv10s; (bottom) YOLOv10m

Detection accuracy was measured using Average Precision (AP) and mean Average Precision (mAP), which are most widely adopted metrics in object detection tasks [33, 36]. mAP was used with an Intersection-over-Union (IoU) threshold of 0.5; the IoU threshold is a standard benchmark for object detection performance evaluation and is typically set to 0.5 in practice [36]. The detection accuracy of the models was evaluated using the test set of the same dataset. To measure detection speed each model was independently deployed on the Servosila Engineer robot and run for multiple iterations. During execution time the robot remained stationary and camera's measurements were within acceptable limits with no outliers observed. Each iteration consisted of three stages: preprocessing, inference, and postprocessing. All these stages involved a single image. During the preprocessing stage, an image was resized to 640×640 and interpolated using a resampling technique based on pixel area relation [37]. During the inference stage, the neural network predicted

an object class and a location within the image. In the postprocessing stage, all detected objects in the image were filtered using a confidence threshold of 0.5. The filtered objects were processed using the Non-Maximum Suppression (NMS) algorithm [38] with an IoU threshold of 0.5. After completing the iterations, the average time per iteration (latency) was calculated in milliseconds.

The models' comparison results are presented in Tab. 2. YOLOv10s demonstrated a good balance between detection accuracy (0.657) and speed (730 ms). This model was further used in experiments described in Section 4.

Tab. 2. Comparison results of the models

Model	mAP:0.5	Latency (ms)
YOLOv10n	0.601	265
YOLOv10s	0.657	730
YOLOv10m	0.687	1955

3.3. Detection pipeline

The implemented human body detection is based on the YOLOv10s deep neural network model, which was trained using the Transfer Learning method [39] on the dataset described in Subsection 3.1. An absence of a Graphics Processing Unit and the limited SSD's storage capacity of the robot required to export the trained YOLOv10s model to the OpenVINO format, which needs a minimal disk space and does not demand high computational resources [40 – 41]. Due to these limitations, alternative formats such as PyTorch, TorchScript, and TensorRT could not be utilized.

A custom-developed package *hum_det* provided a ROS node for using the YOLOv10s model with ROS-based robots. The ROS node *yolov10s_node* was deployed directly on the Servosila Engineer robot to perform object detection. This node subscribed to the */stereo/left/image_raw* ROS topic, which corresponds to a video stream from the left camera of the robot's stereoscopic pair. The neural network model processed the acquired video frames and provided an output stream containing frames with detected objects. Next, the processed video stream was published to the */yolov10s_node/stereo/left/det_image* ROS topic. Fig. 5 shows a generic human detection pipeline.

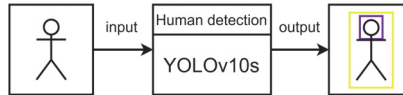


Fig. 5. A pipeline of the object detection algorithm: image input, human detection processing, and target highlighting

4. Experiments

4.1. Experimental setup

A single USAR scenario was used for all experiments. A cluttered indoor environment was set up in a university classroom using chairs, tables, and other furniture. The environment used artificial lighting (during a dark time of a day) with no influence from natural light. The lamps were evenly distributed across the room and positioned 300 cm above the floor. The average illumination level was 500 lux; the measurements were taken with BH1750FVI sensor (by Rohm) at a height that corresponded to the height of the robot's camera.

The Servosila Engineer robot performed human detection with its left camera. The camera was positioned at a height of 30 cm above the floor (by setting up corresponding joints' angles of the manipulator), and all measurements were taken relatively to this camera. The robot maintained the same posture throughout the experiments. Three reference positions for the robot's camera were selected relative to a *start line* of the experimental environment, based on the following distance intervals: [50, 200], [200, 350], and [350, 500] cm. Fig. 6 illustrates the room layout and the distance intervals' scheme. This interval-based division was used for experimental evaluation of the system. Three participants were located in a predefined *CLUTTERED ENVIRONMENT* area (Fig. 6).

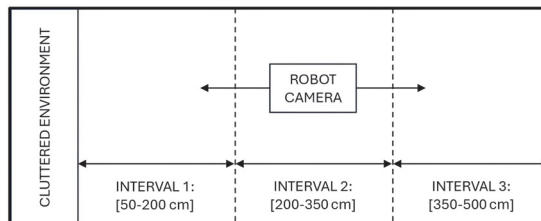


Fig. 6. Layout of the room and available positions for the robot's camera: the camera could be positioned within any of these three intervals

The three participants were selected for simultaneous detection of multiple individuals; the participants adopted various poses and wore different types of clothing. While the experimental environment and the participants were different from those in the training data, the environment and the participants' poses remained unchanged throughout all three experiments.

The developed human victims' detection algorithm was executed with a confidence threshold of 0.5 and NMS with an IoU threshold of 0.5 for all detectable object classes throughout all experiments. Each frame of the video stream contained the three participants. An object was considered undetected if its confidence score was below the threshold or if its bounding box overlapped significantly with that of another object with a higher confidence. Otherwise, a colored bounding box was displayed for the detected object.

4.2. Evaluation

A video stream of 50 images was recorded at each location with the robot's camera positioned randomly within the three specified distance intervals; thus, a total dataset size across the three experiments contained 150 images. The images were annotated to compute detection accuracy using the AP metric. Tab. 3 presents a number of labeled objects of each class in the dataset for each experiment (a distinct distance interval). While there were 150 appearances of *person* class objects for each interval in total (50 frames with three different participants each), the variation in the number of other classes' objects across the experiments was caused by changing a distance between the robot and the human participants.

Tab. 3. Number of labeled objects of each class for each experiment

Class	Interval 1	Interval 2	Interval 3
Foot	150	131	100
Head	93	100	91
Hand	226	249	241
Person	150	150	150

Fig. 7 shows output images generated by the detection algorithm for the same scene being viewed by the robot camera from different distances and positions within Interval 1 (the top row), Interval 2 (the middle row), and Interval 3 (the bottom row). For each bounding box, its class's probability is labeled above the box. In the top-left image, the algorithm detected one *head*, four *hand*, two *foot*, and one *person* objects; in the top-right — two *head*, three *hand*, three *foot*, and two *person* objects. In the middle-left image, the algorithm detected two *head*, two *hand*, one *foot*, and two *person* objects; in the middle-right — one *head*, two *hand*, one *foot*, and two *person* objects. In the bottom-left image, the algorithm detected two *hand* (the second box and its label are partially occluded by the "0.55" box) and one *person* objects; in the bottom-right — one *hand* and one *person* objects. Note that the detection capabilities of the system drastically degrade with the distance increase.

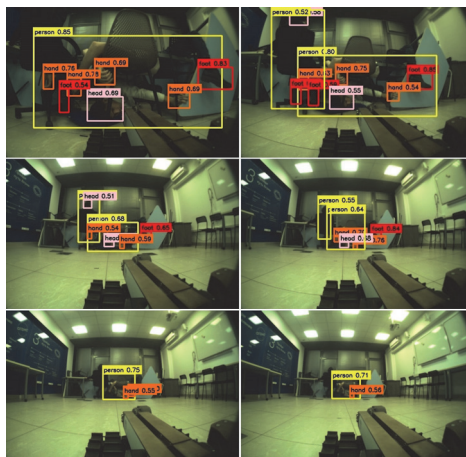


Fig. 7. Human detection within intervals: (top) [50,200] cm; (middle) [200,350] cm; (bottom) [350,500] cm

Fig. 8 illustrates detection reliability of objects depending on the distance between the camera and the *start line*. The AP metric used with an IoU threshold of 0.5 demonstrates higher detection efficiency in the first interval: all classes in the first interval, except for the *person* class (0.36), lead in the detection accuracy across all three intervals. In the second interval, the detection accuracy of *foot* class objects drops significantly (0.3) compared to other objects. Since all *foot* class objects had dark colors and blended in the dark environmental background, distinguishing their boundaries was challenging for the algorithm. The third interval was the most challenging for the neural network model: *hand* and *person* class objects were detected with low accuracy (0.48 and 0.32, respectively), while *foot* and *head* class objects were not detected at all (both zeros). Across all three intervals, *hand* class objects were detected most accurately; it is worth mentioning that the *hand* class is typically one of the most difficult to detect in object detection tasks for majority of neural networks.

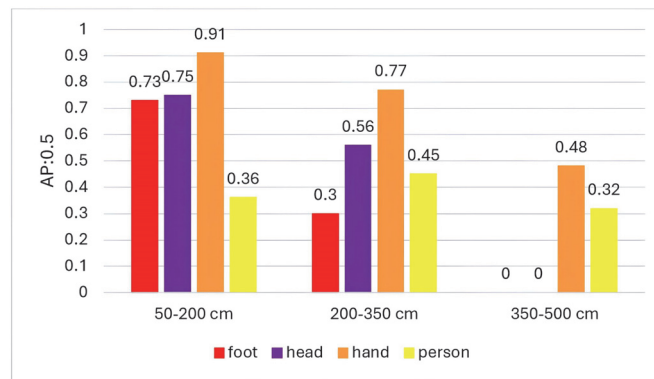


Fig. 8. Object detection results based on the distance between the camera and the cluttered environment

The *person* class demonstrated a distinct (from all other classes) behavior. In highly cluttered environments where heads, hands, and feet of different individuals overlapped significantly, the neural network had difficulty in accurately associating them with correct participants. To process the entire scene context, the neural network merged all these objects into a single large *person* class object. During all experiments, the neural network merged two victims lying parallel to each other into a single *person* class object. Accuracy evaluation results showed that the predicted bounding box for the *person* class object had a small IoU value with either of the two ground-truth bounding boxes of the parallelly lying individuals. The closer the camera was to the victims, the smaller this IoU value became, which caused lower accuracy for the *person* class in the first interval (0.36) since the detections were not always assigned to the *person* class objects. With the distance growth, the IoU value between predicted and ground-truth bounding boxes increased and thus a probability of certain object class detection increased as well. As a result, the *person* class showed higher accuracy in the second interval (0.45) compared to the others. A person that sat in a chair was not detected properly, thus causing lower accuracy in the third interval (0.32).

5. Discussion

While four cameras of the Servosila Engineer robot would cover a large percentage of environment, detecting humans in all video streams simultaneously is impractical due to computational limitations of the robot's onboard hardware. Therefore, the object detection algorithm used a video stream from a single camera to ensure a proper performance during real-time rescue missions. Similar challenges were addressed in previous studies through optimization techniques to balance performance and computational demands [42 – 43].

Unfortunately, our USAR datasets included only Caucasian race individuals, which was caused by available for experiments students of the Intelligent Robotics Department at Kazan Federal University. This limitation may lead to reduced detection accuracy when identifying people with different skin tones [44 – 45]. To improve the model's generalizability, future work should incorporate a more diverse dataset that includes individuals of various racial backgrounds.

Another drawback of the model was a limited amount of training data obtained by the robot's camera and low diversity of the data. The limited variety of shooting conditions and the small number of people per image (ranging from 1 to 2 individuals) in the training set negatively affected the trained model. As a result, the model demonstrated a lower ability to generalize and accurately detect people in conditions different from those in the training set. In the future, the dataset should be expanded by collecting new images under more diverse conditions and with a higher number of people per image.

The obtained detection accuracy at short distances of the first interval of [50, 200] cm is enough in order to serve as an intelligent assistant for a human operator, yet it is not suitable for running the robot in an autonomous mode. To improve the detection accuracy, our ongoing work concentrates on fusing together data about the four labels through a sequence of frames and analyzing spatial relationships between the detected objects with different labels.

The Servosila Engineer robot features an integrated flashlight (located in the robot head, next to the right camera of the stereo pair) to allow operation in poorly lit conditions. One of promising future research directions is an evaluation of the proposed human detection algorithm in low-light environments, which could combine flashlight artificial lighting and image processing techniques.

While YOLOv5 serves lightweight applications (e.g., basic surveillance), a newer YOLOv10s meets rescue robotics requirements, including real-time processing and reliable object detection in dynamic environments. Future studies could assess trade-offs between different YOLO models to optimize a selection for specific rescue tasks.

Conclusions

This paper presented a robot operating system based victim detection framework for Servosila Engineer rescue robot. The solution was based on a deep learning approach and used a post-trained CNN model for detecting victims beneath rubble. The human body detection module employed the YOLOv10s trained on the SAR image dataset consisting

of data from the existing LIRS-USAR-set.v1 dataset and the custom dataset collected using the Servosila Engineer robot's camera. The resulting dataset contained 15068 SAR images and can be used for the tasks of detecting casualties and alive humans in SAR scenarios.

After the model training, three validation experiments with different distances between the robot and the cluttered area were run. Each experiment simultaneously employed three participants that simulated human victims of an earthquake and produced 50 images for human detection. The experimental results demonstrated that human detection algorithm performance in cluttered environments depends on a distance between the robot and the cluttered area. In contrast to prior research relying on either high-end computing infrastructure or multiple synchronized cameras, our system achieved acceptable detection accuracy using a single camera and low-power onboard hardware, making it more practical for real-time deployment in disaster scenarios. The algorithm showed reasonable performance under artificial lighting conditions when the robot's camera was positioned within a distance of 50 to 200 cm from a *start line* of a cluttered area. Within this distance, an AP of 0.75, 0.91, and 0.73 was achieved for the *head*, *hand*, and *foot* classes respectively; as the distance between the robot and the victim increased the AP rapidly degraded to zero. The experiments showed that *hand* class objects were detected more reliably compared to other objects across all three intervals. These findings highlight both practical relevance and a methodological contribution of our approach, which advances the field by demonstrating a scalable and generalizable solution under realistic operational constraints.

Acknowledgements

This paper has been supported by the Kazan Federal University Strategic Academic Leadership Program ("PRIORITY-2030").

References

- [1] Mavroulis S, Mavrouli M, Lekkas E, Tsakris A. Managing earthquake debris: Environmental issues, health impacts, and risk reduction measures. *Environments* 2023; 10(11): 192. DOI: 10.3390/environments10110192.
- [2] AlAli ZT, Alabady SA. A survey of disaster management and SAR operations using sensors and supporting techniques. *Int J Disaster Risk Reduct* 2022; 82: 103295. DOI: 10.1016/j.ijdrr.2022.103295.
- [3] Magid E, Pashkin A, Simakov N, Abbyasov B, Suthakorn J, Svinin M, Matsuno F. Artificial intelligence based framework for robotic search and rescue operations conducted jointly by international teams. In Book: Ronzhin A, Shishlakov V, eds. *Proceedings of 14th International Conference on Electromechanics and Robotics "Zavalishin's Readings": ER(ZR) 2019*, Kursk, Russia, 17-20 April 2019. Singapore: Springer Nature Singapore Pte Ltd; 2020: 15-26. DOI: 10.1007/978-981-13-9267-2_2.
- [4] Messina E, Jacoff A. Performance standards for urban search and rescue robots. *Proc SPIE* 2006; 6230: 62301V. DOI: 10.1117/12.663320.
- [5] Blackburn MR, Everett HR, Laird RT. After action report to the joint program office: Center for the robotic assisted search and rescue (CRASAR) related efforts at the world trade center. Technical Document 2002; 3141: 8.
- [6] Jiao J, Wei H, Hu T, Hu X, Zhu Y, He Z, Wu J, Yu J, Xie X, Huang H, Geng R, Wang L, Liu M. Fusionportable: A multi-sensor campus-scene dataset for evaluation of localization and mapping accuracy on diverse platforms. *IEEE/RSJ Int Conf on Intelligent Robots and Systems (IROS) 2022*: 3851-3856. DOI: 10.1109/IROS47612.2022.9982119.
- [7] Cheng M-M, Zhang Z, Lin W-Y, Torr P. BING: Binarized normed gradients for objectness estimation at 300fps. *IEEE Conf on Computer Vision and Pattern Recognition (CVPR) 2014*: 3286-3293. DOI: 10.1109/CVPR.2014.414.
- [8] Myrzin V, Tsoy T, Bai Y, Svinin M, Magid E. Visual data processing framework for a skin-based human detection. In Book: Ronzhin A, Rigoll G, Meshcheryakov R, eds. *Interactive collaborative robotics. 6th International Conference, ICR 2021*. Cham, Switzerland: Springer Nature Switzerland AG; 2021: 138-149. DOI: 10.1007/978-3-030-87725-5_12.
- [9] Dadwhal YS, Kumar S, Sardana HK. Data-driven skin detection in cluttered search and rescue environments. *IEEE Sens J* 2019; 20(7): 3697-3708. DOI: 10.1109/JSEN.2019.2959787.
- [10] Zagitov A, Chebotareva E, Toshev A, Magid E. Comparative analysis of neural network models performance on low-power devices for a real-time object detection task. *Computer Optics* 2024; 48(2): 242-252. DOI: 10.18287/2412-6179-CO-1343.
- [11] Wang A, Chen H, Liu L, Chen K, Lin Z, Han J, Ding G. YOLOv10: Real-time end-to-end object detection. *arXiv Preprint*. 2024. Source: <<https://arxiv.org/abs/2405.14458>>. DOI: 10.48550/arXiv.2405.14458.
- [12] Alif MAR, Hussain M. YOLOv1 to YOLOv10: A comprehensive review of YOLO variants and their application in the agricultural domain. *arXiv Preprint*. 2024. Source: <<https://arxiv.org/abs/2406.10139>>. DOI: 10.48550/arXiv.2406.10139.
- [13] Abdulganeev R, Lavrenov R, Dobrokvashina A, Bai Y, Magid E. Autonomous door opening with a rescue robot. *10th Int Conf on Automation, Robotics and Applications (ICARA) 2024*: 7-11. DOI: 10.1109/ICARA60736.2024.10552969.
- [14] Cruz Ulloa C, Orbea D, del Cerro J, Barrientos A. Thermal, multispectral, and RGB vision systems analysis for victim detection in SAR robotics. *Appl Sci* 2024; 14(2): 766. DOI: 10.3390/app14020766.
- [15] Zafar MH, Moosavi SKR, Sanfilippo F. Enhancing unmanned ground vehicle performance in SAR operations: integrated gesture-control and deep learning framework for optimised victim detection. *Front Robot AI* 2024; 11: 1356345. DOI: 10.3389/frobt.2024.1356345.
- [16] Huang C-H, Chen Y-C, Hsu C-Y, Yang J-Y, Chang C-H. FPGA-based UAV and UGV for search and rescue applications: A case study. *Comput Electr Eng* 2024; 119(A): 109491. DOI: 10.1016/j.compeleceng.2024.109491.
- [17] Louie W-YG, Nejat G. A victim identification methodology for rescue robots operating in cluttered USAR environments. *Adv Robot* 2013; 27(5): 373-384. DOI: 10.1080/01691864.2013.763743.
- [18] De Cubber G, Doroftei D, Baudoin Y, Serrano D, Chintamani K, Sabino R, Ourevitch S. ICARUS: Providing unmanned search and rescue tools. *6th IARP Workshop on Risky Interventions and Environmental Surveillance (RISE) 2012*.

- [19] Cruz Ulloa C, Garcia M, del Cerro J, Barrientos A. Deep learning for victims detection from virtual and real search and rescue environments. In Book: Tardioli D, Matellán V, Heredia G, Silva MF, Marques L, eds. ROBOT2022: Fifth Iberian Robotics Conference. Advances in Robotics, Volume 2. Cham: Springer International Publishing; 2022: 3-13. DOI: 10.1007/978-3-031-21062-4_1.
- [20] Morales J, Vázquez-Martín R, Mandow A, Morilla-Cabello D, García-Cerezo A. The UMA-SAR dataset: Multimodal data collection from a ground vehicle during outdoor disaster response training exercises. *Int J Robotics Res* 2021; 40(6-7): 835-847. DOI: 10.1177/02783649211004959.
- [21] Kohlbrecher S, Kunz F, Koert D, Rose C, Manns P, Daun K, Schubert J, Stumpf A, von Stryk O. Towards highly reliable autonomy for urban search and rescue robots. In Book: Bianchi RAC, Akin HL, Ramamoorthy S, Sugiura K, eds. RoboCup 2014: Robot World Cup XVIII. Cham: Springer International Publishing Switzerland; 2015: 118-129. DOI: 10.1007/978-3-319-18615-3_10.
- [22] Rafael VM, Jose CS, Abel AH, Andres MA, Joseph GM, Jarelh GB, Jesus TS. Development of a low-cost teleoperated explorer robot (TXRob). *Int J Adv Comput Sci Appl* 2022; 13(7): 897-903. DOI: 10.14569/IJACSA.2022.01307104.
- [23] Bahadori S, Iocchi L, Nardi D, Settembre GP. Stereo vision based human body detection from a localized mobile robot. *IEEE Conf on Advanced Video and Signal Based Surveillance* 2005: 499-504. DOI: 10.1109/AVSS.2005.1577319.
- [24] Castillo C, Chang C. A method to detect victims in search and rescue operations using template matching. *IEEE Int Safety, Security and Rescue Robotics, Workshop* 2005: 201-206. DOI: 10.1109/SSRR.2005.1501256.
- [25] Kleiner A, Kummerle R. Genetic MRF model optimization for real-time victim detection in search and rescue. *2007 IEEE/RSJ Int Conf on Intelligent Robots and Systems* 2007: 3025-3030. DOI: 10.1109/IROS.2007.4399006.
- [26] Jacoff AS, Messina ER, Evans J. Experiences in deploying test arenas for mobile autonomous robots. *Proc 2001 Performance Metrics for Intelligent Systems* 2001: 1-8.
- [27] Gabdrahmanov R, Tsoy T, Bai Y, Svinin MM, Magid E. Gear wheels based simulation of crawlers for mobile robot Servosila Engineer. *19th Int Conf on Informatics in Control, Automation and Robotics (ICINCO) 2022*: 565-572. DOI: 10.5220/0011355200003271.
- [28] Mavrin I, Lavrenov R, Svinin M, Sorokin S, Magid E. Remote control library and GUI development for Russian crawler robot Servosila Engineer. *MATEC Web of Conferences* 2018; 161: 03016. DOI: 10.1051/mateconf/201816103016.
- [29] St-Onge D, Herath D. The robot operating system (ROS1 &2): Programming paradigms and deployment. In Book: Herath D, St-Onge D, eds. *Foundations of robotics: A multidisciplinary approach with Python and ROS*. Singapore: Springer Nature Singapore Pte Ltd; 2022: 105-126. DOI: 10.1007/978-981-19-1983-1_5.
- [30] Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A. The pascal visual object classes (VOC) challenge. *Int J Comput Vis* 2010; 88: 303-338. DOI: 10.1007/s11263-009-0275-4.
- [31] Lin T-Y, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL. Microsoft COCO: Common objects in context. In Book: Fleet D, Pajdla T, Schiele B, Tuytelaars T, eds. *Computer Vision – ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*. Cham: Springer International Publishing Switzerland; 2014: 740-755. DOI: 10.1007/978-3-319-10602-1_48.
- [32] Selcuk B, Serif T. A comparison of YOLOv5 and YOLOv8 in the context of mobile UI detection. *Int Conf on Mobile Web and Intelligent Information Systems* 2023; 161-174. DOI: 10.1007/978-3-031-39764-6_11.
- [33] Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: unified, real-time object detection. *2016 IEEE Conf on Computer Vision and Pattern Recognition (CVPR) 2016*: 779-788. DOI: 10.1109/CVPR.2016.91.
- [34] Bernabé S, González C, Fernández A, Bhargale U. Portability and acceleration of deep learning inferences to detect rapid earthquake damage from VHR remote sensing images using intel OpenVINO toolkit. *IEEE J Sel Top Appl Earth Obs Remote Sens* 2021; 14: 6906-6915. DOI: 10.1109/JSTARS.2021.3075961.
- [35] Demidovskij A, Gorbachev Y, Fedorov M, Slavutin I, Tugarev A, Fatekhov M, Tarkan Y. OpenVINO deep learning workbench: Comprehensive analysis and tuning of neural networks inference. *2019 IEEE/CVF Int Conf on Computer Vision Workshops (ICCVW) 2019*: 783-787. DOI: 10.1109/ICCVW.2019.00104.
- [36] Farhadi A, Redmon J. YOLOv3: An incremental improvement. *Computer vision and pattern recognition*. Berlin, Heidelberg, Germany: Springer 2018; 1804: 1-6.
- [37] Thévenaz P, Blu T, Unser M. Image interpolation and resampling. In Book: Bankman IN, ed. *Handbook of medical imaging*. Academic Press Inc; 2000: 393-420. DOI: 10.1016/B978-012077790-7/50030-8.
- [38] Hosang J, Benenson R, Schiele B. Learning non-maximum suppression. *2017 IEEE Conf on Computer Vision and Pattern Recognition (CVPR) 2017*: 6469-6477. DOI: 10.1109/CVPR.2017.685.
- [39] Hosna A, Merry E, Gyalmo J, Alom Z, Aung Z, Azim MA. Transfer learning: a friendly introduction. *J Big Data* 2022; 9: 102. DOI: 10.1186/s40537-022-00652-w.
- [40] Andriyanov N, Papakostas G. Optimization and benchmarking of convolutional networks with quantization and OpenVINO in baggage image recognition. *VIII IEEE Int Conf on Information Technology and Nanotechnology (ITNT) 2022*: 1-4. DOI: 10.1109/ITNT55410.2022.9848757.
- [41] Gao T, Suto J. Acceleration of image classification and object tracking by the intel Neural Compute Stick 2 with power efficiency evaluation on Raspberry Pi 4B. *Sensors* 2025; 25(6): 1794. DOI: 10.3390/s25061794.
- [42] Mao H, Yao S, Tang T, Li B, Yao J, Wang Y. Towards real-time object detection on embedded systems. *IEEE Trans Emerg Top Comput* 2016; 6(3): 417-431. DOI: 10.1109/TETC.2016.2593643.
- [43] Coates A, Ng AY. Multi-camera object detection for robotics. *IEEE Int Conf on Robotics and Automation* 2010: 412-419. DOI: 10.1109/ROBOT.2010.5509644.
- [44] Yu PK. The algorithmic divide and equality in the age of artificial intelligence. *Fla L Rev* 2020; 72(2): 331-389.
- [45] Buolamwini J, Gebru T. Gender shades: Intersectional accuracy disparities in commercial gender classification. *Proc Mach Learn Res* 2018; 81: 77-91.

Authors' information

Ruslan Farkhetdinov (b. 2001), currently a master degree student at Institute of Information Technology and Intelligent Systems at Kazan Federal University, Russia. Research interests are computer vision and machine learning. E-mail: rurfarkhetdinov@stud.kpfu.ru

Bulat Abbyasov (b. 1997), received a master degree from the Institute of Information Technology and Intelligent Systems (ITIS), Kazan Federal University (KFU) in 2022. Currently, he is a third year PhD student and a Research Associate at ITIS KFU. E-mail: bulat.abbyasov@gmail.com

Alexander Eryomin (b. 2000), received a BSc degree from Siberian Federal University in 2022. In 2024, he received a master degree in Intelligent Robotics from the Institute of Information Technology and Intelligent Systems (ITIS) of Kazan Federal University (KFU). Currently he is a first year PhD student and a Research Associate at ITIS KFU. E-mail: aneremin@it.kfu.ru

Alexandra Dobrokvashina (b. 1998), received a master degree from the Institute of Information Technology and Intelligent Systems (ITIS) of Kazan Federal University (KFU) in 2021. Currently she works as a Research Associate at the Laboratory of Intelligent Robotic Systems (LIRS) at IT IS KFU. Her research interests are robotics, simulation, motion planning and navigation. E-mail: dobrokvashina@it.kfu.ru

Mikhail Svinin (b. 1959), currently a full professor at the College of Information Science and Engineering at Ritsumeikan University, Japan. His primary research areas include robotics, analytical mechanics, and control theory. He teaches courses in physics, differential equations, and systems biology. His research interests encompass robotics, haptic interfaces, and machine intelligence. He has authored over 200 scientific publications focusing on motion planning and mobile robotics. E-mail: svinin@fc.ritsumei.ac.jp

Evgeni Magid (b. 1975), currently a full professor, a Head of Intelligent Robotics Department and a Head of Laboratory of Intelligent Robotic Systems (LIRS) at Kazan Federal University, Russia. A full professor at HSE University, Russia. Senior IEEE member. Previously he worked at University of Bristol, UK; Carnegie Mellon University, USA; University of Tsukuba, Japan; National Institute of Advanced Industrial Science and Technology, Japan. He earned his Ph.D. degree from University of Tsukuba, Japan. He authors over 300 publications. Research interests are mobile robotics, path planning, search and rescue robotics, human robot interaction, medical robotics, heterogeneous robotic teams, image processing, and computer vision. E-mail: magid@it.kfu.ru

Code of State Categories Scientific and Technical Information : 28.23.15

Received April 07, 2025. The final version – May 19, 2025.
