Automatic target recognition module development for fire control system based on machine learning

Victoria Vysotska [1,†], Roman Romanchuk[1,*,†], Mariia Nazarkevych[1,†], Oleksandr Lavrut [2,†], Tetiana Lavrut [2,†] and Maksym Pysarchuk[3,†]

[1] *Lviv Polytechnic National University, Stepan Bandera 12, 79013 Lviv, Ukraine*

[2] *Hetman Petro Sahaidachnyi National Army Academy, Heroes of Maidan 32, 79026 Lviv, Ukraine*

[3] *Organization and Planning Department of the Training Center "Partnership for Peace" of the International Peacekeeping and Security Center, 79026 Lviv, Ukraine*

Abstract

Target recognition is a priority in military affairs. This task is complicated by the fact that it is necessary to recognize moving objects, different topography and landscape create obstacles for recognition. Combat actions can take place at different times of the day, accordingly, the lighting angle and general lighting must be taken into account. It is necessary to detect the object in the video by segmenting the video frames and to recognize and classify it. In the work, the authors propose the development of a target recognition module as a component of the fire control system within the framework of the proposed information technology through artificial intelligence use. The YOLOv8 pattern recognition model family was used to develop the target recognition module. The data was collected from open sources, in particular, from video footage posted in open sources on the YouTube platform. The main task of data pre-processing is the classification of three classes of objects on video or in real-time - APC, BMP, and TANK. The dataset is formed using the Roboflow platform based on marking tools and, subsequently, augmentation tools. The data set consists of 1193 unique images - approximately equally for each class. The training was conducted using Google Colab resources. 100 epochs were taken to train the model. The analysis is carried out according to mAP50 (mean Average Precision as 0.85), mAP50-95 (0.6), precision (0.89) and recall (0.75) metrics. Large losses are present because the background was not taken into account in the study - training the module based on validated data (images) of the background without the technique. This will be the next step. It is also necessary to expand the classification of objects of military equipment.

Keywords

security, privacy, moving objects recognition, targets identification, machine learning, YOLO

MoMLeT-2024: 6th International Workshop on Modern Machine Learning Technologies, May, 31 - June, 1, 2024, Lviv-Shatsk, Ukraine

* Corresponding author.

† These authors contributed equally.

Victoria.A.Vysotska@lpnu.ua (V. Vysotska); roman.v.romanchuk@lpnu.ua (R. Romanchuk); mariia.a.nazarkevych@lpnu.ua (M. Nazarkevych); alexandrlavrut@gmail.com (O. Lavrut); Lavrut_t_v@i.ua (T. Lavrut); maximpysarchuck@gmail.com (M. Pysarchuk)

0000-0001-6417-3689 (V. Vysotska); 0009-0004-4352-1073 (R. Romanchuk); 0000-0002-6528-9867 (M. Nazarkevych); 0000-0002-4909-6723 (O. Lavrut); 0000-0002-1552-9930 (T. Lavrut); 0009-0001-9086-4608 (M. Pysarchuk)

1. Introduction

Today, the leading armies of the world strive to increase the capabilities of their main models of equipment and weapons by modernizing the existing fleet or developing the latest models. Automatic target recognition (Automatic Target Recognition Unit) is the ability of an algorithm or device to recognize targets or objects based on data received from sensors, including video surveillance, for example from unmanned aerial vehicles (UAVs), such as drones, or from video - recorders on combat vehicles. On the other hand, due to the increased use of UAVs for reconnaissance by the enemy, the security and privacy of many critical locations may be compromised. Therefore, they are also a legitimate target for fire control detection.

Target recognition information technology is a key component in cruise missiles and UAVs, as well as in the development of combat robots or sapper robots. Automatic target recognition is used not only in military affairs but also, for example, in the organization of searching for people/objects (at sea, in the area of natural disasters, fires, etc.).

The task of automatic target recognition in combat conditions is complicated by several factors, in particular:

1. Possible movement of the recognized object;
2. The movement of the object (combat vehicle or UAV), from where the video surveillance and further recognition of the targets originates;
3. Different weather conditions;
4. Different topography and landscape, including forest strips;
5. Presence of other objects that are potentially not targets (buildings, downed/destroyed combat vehicles, parts of structures such as bridges, etc.);
6. Lighting;
7. Potentially, the object being recognized is not an enemy;
8. Part of the recognized object is hidden behind obstacles;
9. The observation angle for different objects is different (for UAVs from top to bottom, for combat vehicles not only forward/around, but upwards for UAVs, for example).

As you can see, lighting conditions, different sizes of objects, moving backgrounds and various background contrasts significantly affect the quality, efficiency and speed of object recognition from video surveillance [1-2]. It is necessary not only to detect the object in the video by segmenting the video frames, but also to recognize and classify it (for example, a drone or a bird, a car or a building, etc.), and this is usually during the movement of both the surveillance object and the object-observer under adverse conditions in real-time. Detecting flying objects or objects moving in adverse conditions in the video is different from standard object detection because the size of a stationary/moving/flying and/or partially hidden object behind another object is constantly changing in frames depending on its distance and the movement of the observing object. It has problems such as low resolution, changes in lighting between day and night and unstable background, different weather conditions. Also, the accuracy of recognition depends on the quality of the surveillance camera, the selection of which during hostilities is not a controlled process. The

complexity of observation increases when recognizing from 2D (in front of the camera) to 3D (from above from the bottom at different angles at different heights) moving objects, taking into account scaling and proportions. Similarly, it reduces the accuracy of recognizing objects that are visually similar to each other and differ in small features or their absence when viewed from different angles or when the hull is partially hidden behind other natural objects or buildings (for example, some modifications of the T series tanks). Therefore, the detection and recognition of stationary/moving/flying/moving and/or partially hidden objects by other objects in the state of immobility/movement of the observer in different weather conditions, landscapes, lighting and at different heights have a large scope of observation and high mobility. There is a strong need for such applications in the real world because of the size differences within the same object type and the spatial resolution of the sensor.

Thus, the purpose of the work is to develop a method of target recognition in real-time as a component of the fire control system, due to the use of artificial intelligence.

2. Related works

In recent years, during the full-scale war in Ukraine, with the gradual improvement of drone control technology, UAV remote sensing images and videos have become an important source of operational data. At the same time, equipping combat vehicles with video recorders for video surveillance with elements of artificial intelligence and machine learning for real-time object recognition will increase the level of security of combatants with appropriate timely responses to the results of target recognition.

Video frames → video frame segmentation → detection of potential objects → detection of moving objects → recognition of objects as potentially dangerous → object classification → object identification.

Today, neural networks and deep learning are commonly used for tasks such as image segmentation [3-5], object detection [6-8], and image classification [9-11]. Most of the currently applied deep neural network models, such as PSPNET [4], U-NET [5], RESNET [10], and VGG [11], are developed based on manually collected image (non-video) datasets under favourable conditions, such as MS-COCO [12], VOC2012 [13], VOC2007 [14].

There are two general scenarios for the application of methods of detecting objects by remote sensing by a drone or based on video surveillance from a car, in particular, data processing is assumed:

- after flight/trip using stationary computers (requires high detection and identification accuracy).
- in real-time during the flight/ride, when the onboard computer on the drone or in the car, respectively, synchronously processes the video data in real-time. The parameters of the model must be within a certain scale to meet the requirements for the operation of the embedded equipment. Once the operating conditions are met, the detection accuracy of the method should also be as high as possible.

Therefore, applied neural network-based object detection methods must meet different requirements for each scenario.

Thus, neural network methods for detecting objects in drone remote sensing video or video surveillance in combat vehicles must be able to adapt to the specific characteristics of this data. They should be designed to meet post-flight/trip data processing requirements, which can provide high accuracy and recall speed, or they should be designed as smaller-scale parameter models that can be deployed in embedded hardware environments for real-time processing on drones/ cars. In this work, we propose the application of a neural network based on the YOLOv8 architecture for the automatic recognition of objects as potential targets of a fire control system.

Currently, numerous methods of object detection based on neural networks have been proposed, in particular, using the YOLO series [15-22]. Unlike the two-step methods, the one-step method combines object location and classification in one step, achieving real-time object detection on both desktop and embedded hardware. These methods not only achieve good identification results, but also offer several improvements in areas such as training data augmentation methods, network training methods, loss functions, activation functions, and network model structures.

YOLOX, a neural network model with one-step object detection, is proposed in [18]. In [19], the authors proposed a neural network model with one-step target detection. In [20], the authors investigated the optimal speed and accuracy of object detection based on YOLOv4. The authors in [23] described CSPDarkNet as the backbone structure of the network, improving the learning ability of convolutional neural networks, and allowing the network to maintain the accuracy of feature map extraction. In [24], the authors proposed the CrowdDet method based on a neural network for detecting dense and mutually closed targets in images. In the neck part of the network, the SPPF module and the PAFPN module have been introduced [25]. The author still uses CSPDarkNet [23] as the backbone network, but introduces SiLU as an activation function that solves the gradient dispersion problem when the input of the ReLU function is negative and the output is 0 [26-27].

There are many studies based on different versions of YOLO, but still, the most important problem in object identification is the effective detection of small objects and the accuracy of classification of various moving objects [28-33] under different environmental conditions (for example, an object in different gradations of green colour on the background, as well as a different spectrum of the green colour [34]) in a video stream of different image quality [35-45] with subsequent storyboarding and segmentation of the corresponding images for identification and classification [46-63].

Remote sensing images often have large dimensions, complex backgrounds, and a significant presence of small objects. The proposed solution is focused on optimizing the accurate detection of small objects at a distance and objects in motion under various environmental conditions. The advantage of YOLO series networks is the use of multi-level detection heads, which allow the detection of objects of different sizes from different levels of feature vectors. Our approach mainly focuses on detecting small objects as well as moving objects using feature vectors from lower layers that have higher spatial resolution. To achieve this goal, we use a machine learning module to optimize their semantic

characteristics. Detection heads can obtain feature vectors with both high spatial resolution and accurate semantic information, thus increasing overall identification accuracy.

In deep learning, feature extraction methods SIFT (Scale Invariant Feature Transform) and HOG (Histograms of Oriented Gradients) performed this task by applying some machine learning algorithms on top of the classifier. Some deep learning methods are applied to colour images, while others are applied to IR images. Real-time processing of IR images is simpler because it requires less memory and computing power. It also does not affect different lighting conditions. However, it is practically impossible to collect a training dataset for some subject areas (for example, military equipment during the war). Algorithms of the YOLO family, based on the CNN architecture, are widely used and well-known algorithms for solving object detection problems. YOLO v4 and YOLO v5 are mostly used models. YOLO v4, being a modified version of YOLO3, uses a cross-stage partial network (CSPNet) in the Darknet, creating a new feature extractor backbone called CSPDarknet53. To increase the efficiency of the algorithm, YOLOv4 uses a bag of freebies and a bag of special offers. Total loss of IOU (CIOU), dropout lock regularization and many expansion approaches. Mish activation, Diou-NMS and modified pathway aggregation networks are included in the speciality package. But YOLOv5 is different from previous versions. Here PyTorch is used instead of Darknet. It uses CSPDarknet53 as structural support. This pipeline removes the redundant gradient information seen in large pipelines and incorporates gradient transformation into feature maps, which speeds up inference, improves accuracy, and reduces model size by reducing the number of parameters. It enhances the flow of information using a Path Aggregation Network (PANet), resulting in three different feature map outputs for multiscale prediction. This improves the model's ability to effectively predict small and large objects. The image is sent to PANet for feature fusion after input to CSPDarknet53 for feature extraction. The processing speed of YOLO v4 and v5 ranges from 45 to 150 frames per second. However, unlike the faster R-CNN, it has lower recall error and higher localization. Since each grid can only offer two bounding boxes, it also has trouble detecting nearby objects and small objects. The latest addition to the YOLO object detection family is the YOLO v8 model. It is the fastest and most accurate real-time object detector available today. All YOLOv8-based tools outperform previous object detectors in terms of speed and accuracy.

3. Models and methods

The paper discusses a hybrid approach using CNN-LSTM to improve the detection performance of military equipment with a moving background and different distances, as well as its TANK/APC/BMP. The main contributions of the article.

1. Collection of images from open sources for YouTube platforms.
2. Hybrid CNN-LSTM model with hyperparametric tuning using Bayesian optimization for object detection.
3. Detailed analysis of the YOLOV8 model on different ranges of images and determination of their accuracy with a certain confidence value.

Object detection algorithms in deep learning are mainly divided into regional and regression. The main task of detecting objects of military equipment is to detect an object in a frame where objects of different target classes are present; therefore, object classification is a prerequisite for object detection using a bounding box.

Regression-based algorithms are mainly used for real-time object detection. These are one-step frameworks based on global regression/classification that directly map image pixels to bounding box coordinates, reducing time consumption. One of the fastest object recognition models is YOLO, which can analyse frames at up to 150 FPS for small networks. Although YOLO is not the most accurate model in terms of Mean Average Accuracy (mAP), it performed reasonably well during training.

As part of the detection of objects of military equipment, an experiment was conducted on automatic recognition and identification of targets of the fire control system. This was implemented using various object detection models. The experiment is focused on the implementation of the YOLO v8 algorithm for comparison with other versions.

Algorithms based on regional offers. A regional convolutional neural network (R-CNN) is proposed for CNN matching. Compared with models without deep CNNs, R-CNN significantly improved the detection performance for the mean average precision (mAP). It has several disadvantages, including costly training in terms of money and time, and worst of all, high latency (detection time). Building on the performance of R-CNN, fast R-CNN improves accuracy by speeding up training and testing. Fast R-CNN dramatically reduces training and testing time, but regional proposals are still generated using traditional methods that require a lot of time for pre-processing. A faster R-CNN is proposed as a solution to the region proposal bottleneck problem, which makes region proposals through a neural network. Fast/faster R-CNN and other object detectors using regional proposal networks have shown superior performance in many tests. However, they are not always successful in finding small objects. Current approaches are poorer in terms of repeatability and generalizability when real-world circumstances are constantly changing because they depend on specific image data.

Classification of stationary/moving/flying and/or partially hidden objects of military equipment by another object under adverse conditions in real-time. Detecting flying objects or objects moving in adverse conditions in the video is different from standard object detection because the size of a stationary/moving/flying and/or partially hidden object behind another object is constantly changing in frames depending on its distance and the movement of the observing object. It has problems such as low resolution, changes in lighting between day and night and unstable background, different weather conditions. Also, the accuracy of recognition depends on the quality of the surveillance camera, the selection of which during hostilities is not a controlled process. The complexity of observation increases when recognizing from 2D (in front of the camera) to 3D (from above from the bottom at different angles at different heights) moving objects, taking into account scaling and proportions. Similarly, it reduces the accuracy of recognizing objects that are visually similar to each other and differ in small features or their absence when viewed from different angles or when the body is partially hidden behind other natural objects or buildings.

Deep learning-based object detection methods for various tasks, which include lane detection, intelligent vehicle systems, detection of moving objects, including military equipment, etc.

The observation module for automatic recognition and identification of targets of the fire control system M is represented by a simulation model via a tuple:

$$M = < I, O, R, U, N, \alpha, \beta, \gamma >, \quad (1)$$

where I is a set of input data in the form of a video stream from a video camera, $I = \{i_1, i_2, i_3, i_4\}$; O is a set of initial data in the form of recognition and identification of objects of military equipment, $O = \{o_1, o_2, o_3\}$; R is basic rules for processing the input data of the video stream, $R = \{r_1, r_2, r_3, r_4, r_5\}$; U is parameters for processing the input data of the video stream, $U = \{u_1, u_2, u_3, u_2, u_3\}$; N is a neural network for learning the recognition, identification and classification of military equipment objects such as TANK/BMP/APC; α is operator of analysis and storyboarding of input data of the video stream; β is image processing operator through segmentation and analysis of segmented objects; γ is operator of recognition, identification and classification of objects of military equipment such as TANK/BMP/APC.

The main processes of the surveillance model for automatic recognition and identification of fire control system targets are "Video Stream Processing", "Image Processing", "Machine Learning" and "Object Classification".

The process of "Processing a video stream" will be described by a superposition:

$$M_{AU} = \mu \circ \beta \circ \alpha, \quad (2)$$

$$M_{AU} = \mu(\beta(\alpha(i_1, i_2, i_3), r_1, u_1), u_2), \quad (3)$$

where μ is the operator for recognizing any potential objects in the image (buildings, bridges, military equipment, etc.); i_1 is a set of data from the video stream and images of the original; i_2 is a set of images of military equipment; i_3 is a set of landscape background data; r_1 is the rules for framing the video stream into an image; u_1 is a set of conditions for forming images from a video stream; u_2 is a set of image analysis requirements, including noise filtering.

The process of "Image processing" will be described by superposition:

$$M_{CU} = \chi \circ \beta \circ \alpha, \quad (4)$$

$$M_{CU} = \chi(\beta(\alpha(C_{AU}, i_2, i_3, i_4), o_1, r_2, u_3), r_3), \quad (5)$$

where χ is the operator for recognizing potential objects of military equipment in the image; i_2 is set of images of military equipment; i_3 is a set of landscape background data; i_4 is dictionaries of validated images of military equipment; o_1 is the set of all recognized objects in the image; r_2 is image segmentation rules; r_3 is image segment analysis rules; u_3 is a set of image processing conditions.

The process of "Machine learning" will be described as:

$$M_{UL} = \omega \circ \gamma \circ \beta \circ \alpha, \quad (7)$$

$$M_{UL} = \omega(\gamma(\beta(\alpha(C_{CU}, i_1), o_2, i_4), u_4), r_4), \quad (8)$$

where ω is the identification operator of a recognized object of military equipment on multiple images cut from the video stream; i_1 is a set of data from the video stream and images of the original; i_4 is dictionaries of validated images of military equipment; o_2 is the set of all recognized objects of military equipment in the image; r_4 is machine learning rules of the neural network for identification of military equipment; u_4 is a set of conditions for recognition and identification of objects of military equipment.

The process of "Classification of objects" will be described as:

$$M_{US} = \lambda \circ \gamma \circ \beta \circ \alpha, \quad (9)$$

$$M_{US} = \lambda(\gamma(\beta(\alpha(C_{US}, i_1), o_3, i_4), u_5), r_5), \quad (10)$$

where λ is the operator of the classification of the identified object of military equipment on multiple images cut from the video stream; i_1 is a set of data from the video stream and images of the original; i_4 is dictionaries of validated images of military equipment; o_3 is the set of all identified objects of military equipment in the image; r_5 is rules for the classification of military equipment; u_5 is a set of requirements for the classification of recognized objects of military equipment.

The analysis is carried out using classification or clustering, which segments according to certain criteria. Although the collection of information occurs automatically, it is still necessary to implement such studies according to the recognition, identification and classification of objects in unfavourable conditions in motion and with poor image quality, and the corresponding processing of the results. The effectiveness of processing the corresponding background and objects on it also significantly affects the research results (for example, green on a green background or part of an object hidden behind another object). One of the most important criteria of such technology is the ability to collect data depending on the period of the day and season, and their periodicity due to a change in the background due to the results of active hostilities in a certain area.

4. Experiments, results and discussion

In the work, the authors propose the development of a target recognition module as a component of the fire control system within the framework of the proposed information technology through artificial intelligence use.

The YOLOv8 pattern recognition model family was used to develop the target recognition module. This is the latest version of Ultralitics' popular real-time object detection and image segmentation product. YOLOv8 comes bundled with the following pre-built models:

- Image classification models pre-trained on ImageNet database with 224 image resolution.
- Instance segmentation control points trained on the COCO segmentation dataset with 640 image resolution.
- Object detection control points trained on the COCO detection dataset with 640 image resolution.

The output is performed at almost 105 frames per second on the GPU of the average modern laptop, while the ad-large model runs at an average speed of 17 frames per second.

YOLOv8 uses the PuTorch framework, a framework for developing deep neural networks from Facebook. YOLOv8 has several advantages over other tools, including:

1. Most services support the provision of computing power.
2. A large number of methods of application and use of models.
3. High level of accuracy, confirmed by tests on COCO and Roboflow 100 datasets.

As mentioned above, YOLOv8 achieves high accuracy on the COCO dataset. For example, the YOLOv8m model - achieves 50.2% mAP when measured on COCO. When compared against Roboflow 100, a dataset that specifically evaluates model performance in different domains, YOLOv8 performed significantly better than YOLOv5. In addition, YOLOv8 provides developers with a significant list of features. Unlike other models that split tasks between many different Python files, YOLOv8 comes with a CLI interface that makes training the model more intuitive.

The Google Collaboratory service, better known as "Colab", was used for training. “Colab” is a cloud version of Jupyter Notebook. Using Colab does not require installing and running or upgrading your computer hardware to meet Python's CPU/GPU intensive requirements. In addition, Colab provides free access to computing infrastructure such as storage, RAM, computing power, graphics processing units (GPUs) and tensor processing units (TPUs).

The methods used during the study of the formed dataset.

1. Flip – Horizontal (flipping the image object horizontally).
2. Rotation – Between -15° and +15° (rotation of the image object – clockwise or counterclockwise by a degree from -15° to +15°).
3. Brightness – Between -25% and +25% (changing the brightness of the image to increase the resistance of the model to changes in lighting and camera settings – from -25% to +25%).
4. Cutout – 3 boxes with 10% size each (cut out a part of the image – 3 boxes of 10% size each).
5. Bounding Box: Blur – Up to 2.5px
6. Bounding Box: Noise – Up to 15% of pixels.

The last two points are used to expand the level of the bounding box when forming/generating new training data, only changing the content of the bounding boxes of the original image. Image upscaling is the process of increasing the size of a dataset by manipulating the existing training data. Zooming in helps the model to better generalize to a wide range of contexts. For example, you can change the brightness or darkness of an object relative to its background. Or perhaps blur the subject against its background for tasks that often involve shooting fast-moving subjects. Modifications to the bounding box alone led to systematic improvements, especially for models that were small datasets (several thousand photos). You can also change the colours of only objects in the OCR image, blur only moving objects, such as military equipment in various shades of green, rotate

objects, such as objects in the overhead view, and flip the orientation of objects to create mirroring effects similar to those present in most camera situations.

Before the development of the automatic target recognition module of the fire control system, the workspace is organized and the rules for accessing the data store and processing relevant data from it are defined. The Google Collaboratory service, better known as "Colab", was used for training. “Colab” is a cloud version of Jupyter Notebook. The main task of data preprocessing is the classification of three classes of objects on video or in real-time - APC, BMP, and TANK. Next, the corresponding data set was created and filled. The data was collected from open sources, in particular, from video footage posted in open sources on the YouTube platform (videos from the promotion of Rashka technology in the first days of the war on the territory of Ukraine and from military parades of Russian equipment). This process included searching for images and videos of the above-mentioned objects and marking the corresponding objects. The dataset is formed using the Roboflow platform based on marking tools and, subsequently, augmentation tools. The data set consists of 1193 unique images - approximately equally for each class. After applying image preprocessing and argumentation methods, the data set has the following form (Table 1):

```
train: ../train/images
val: ../valid/images
test: ../test/images
nc: 3
names: ['bmp', 'btr', 'tank']
```

Table 1
Distribution of data in the data set

Set type	Absolute value	Relative value
Train Set	2490	87%
Valid Set	225	8%
Test Set	138	5%

The training was conducted using Google Colab resources. 100 epochs were taken to train the model. The statistical results of neural network training are shown in Fig. 1-2. The analysis was carried out according to mAP50 (mean Average Precision), mAP50-95, precision and recall metrics (Fig. 1).

AP (Average precision) is a popular metric for measuring the accuracy of object detectors such as Faster R-CNN, SSD, etc. Average Precision calculates the average precision for recall in the range 0 to 1. It is a measure of the model's precision, taking into account only "easy" detections. mAP50-95: Average of the average accuracy calculated at different IoU thresholds ranging from 0.50 to 0.95. It gives a complete picture of the performance of the model at different levels of detection complexity.

Precision measures how accurate your predictions are. For example, what percentage of your predictions are correct? Recall measures how well you find all positive samples. For example, we can find 75% of all possible positive cases in our best predictions. As we can see from Fig. 1 The Precision metric gives a larger swing at the beginning and becomes more similar to the mAP50 at the end as the number of trials increases. The mAP50-95 has bad

values (0.5-0.6) at the end as the number of trials increases. The Recall metric has a relatively constant value in the range of 0.85-0.75 after half of the conducted experiments (epochs). This is not a good enough result and needs further research and model training on a larger dataset of actual data. The Precision metric gives slightly better results - in the range of 0.85-0.89.

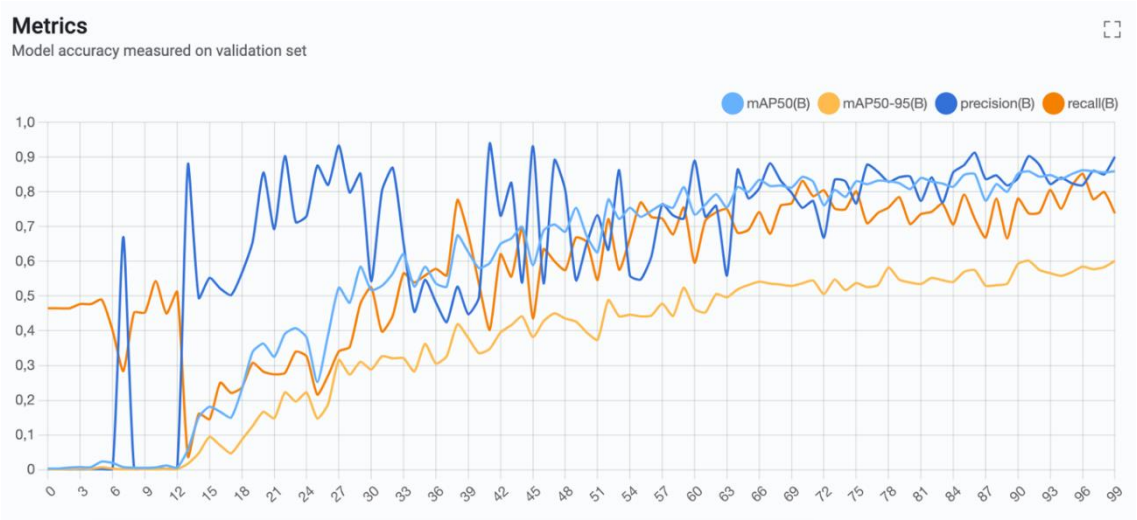

Figure 1: Accuracy graph of the model

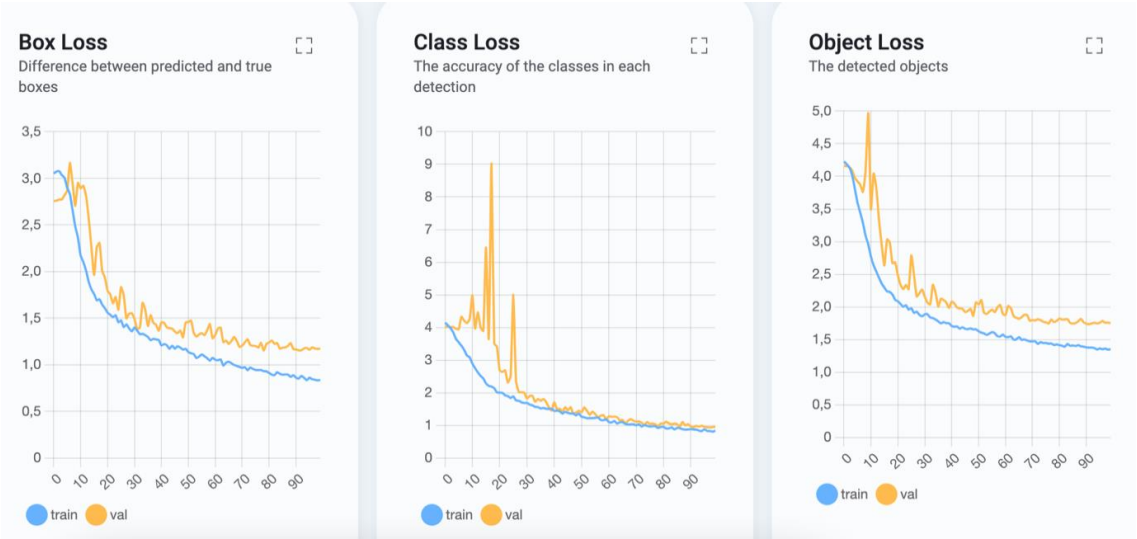

Figure 2: Graphs of model losses

Figures 3-5 show examples of system operation. Large losses are present because the background was not taken into account in the study - training the module based on validated data (images) of the background without the technique. This will be the next step. It is also necessary to expand the classification for objects of military equipment - which is exactly T-64, E-72 or T-90.

Figure 3: Example of object recognition of the BMR class

Figure 4: An example of TANK class object recognition

Figure 5: Example of object recognition of the BMR class

5. Conclusions

Target recognition is a priority in military affairs. This task is complicated by the fact that it is necessary to recognize moving objects, different topography and landscape create obstacles for recognition. Combat actions can take place at different times of the day, accordingly, the lighting angle and general lighting must be taken into account. It is necessary to detect the object in the video by segmenting the video frames and to recognize and classify it. The training was conducted using Google Colab resources. 100 epochs were taken to train the model. The analysis was carried out according to mAP50 (mean Average Precision), mAP50-95, precision and recall metrics.

The proposed method can be used to identify objects of a military nature and recognize targets to create (modernize) modern fire control systems of modern military equipment.

References

[1] S.S. Aote, et al., An improved deep learning method for flying object detection and recognition, SIViP, 2023. doi:10.1007/s11760-023-02703-y.

[2] Z. Zhang, Drone-YOLO: An Efficient Neural Network Method for Target Detection in Drone Images, Drones 7 (2023) 526. doi:10.3390/drones7080526.

[3] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask R-CNN, in Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017, pp. 2961–2969.

[4] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017, pp. 2881–2890.

[5] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in Proceedings of the Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015, Springer: Berlin/Heidelberg, Germany, pp. 234–241.

[6] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks, Adv. Neural Inf. Process. Syst. 28 (2015).

[7] R. Girshick, Fast R-CNN, in Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015, pp. 1440–1448.

[8] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in Proceedings of the Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014, pp. 580–587.

[9] G. Huang, Z. Liu, L. Van Der Maaten, Weinberger, K.Q. Densely connected convolutional networks, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017, pp. 4700–4708.

[10] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016, pp. 770–778.

[11] K. Simonyan, Zisserman, A. Very deep convolutional networks for large-scale image recognition, arXiv 2014, arXiv:1409.1556.

[12] T. Lin, M. Maire, S.J. Belongie, L.D. Bourdev, R.B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Doll'ar, C.L. Zitnick, Microsoft COCO: Common Objects in Context. arXiv 2014, arXiv:1405.0312. URL: http://xxx.lanl.gov/abs/1405.0312.
[13] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, Zisserman, A. The PASCAL Visual Object Classes Challenge (VOC2012) Results. URL: http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.
[14] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, The PASCAL Visual Object Classes Challenge (VOC2007) Results. URL: http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.
[15] G. Jocher, A. Chaurasia, J. Qiu, YOLO by Ultralytics. 2023. URL: https://github.com/ultralytics/ultralytics/blob/main/CITATION.cff.
[16] C.Y. Wang, A. Bochkovskiy, H.Y.M. Liao, YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 7464–7475.
[17] C. Li, et al. YOLOv6: A single-stage object detection framework for industrial applications, arXiv 2022, arXiv:2209.02976.
[18] Z. Ge, S. Liu, F. Wang, Z. Li, J. Sun, Yolox: Exceeding yolo series in 2021. arXiv 2021, arXiv:2107.08430.
[19] G. Jocher, et al. Ultralytics/Yolov5: V6.0—YOLOv5n 'Nano' Models, Roboflow Integration, TensorFlow Export, OpenCV DNN Support, 2021. URL: https://zenodo.org/record/5563715.
[20] A. Bochkovskiy, C.Y. Wang, H.Y.M. Liao, Yolov4: Optimal speed and accuracy of object detection, arXiv 2020, arXiv:2004.10934.
[21] J. Redmon, A. Farhadi, YOLO9000: Better, faster, stronger, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
[22] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016, pp. 779–788.
[23] C.Y. Wang, H.Y.M. Liao, Y.H. Wu, P.Y. Chen, J.W. Hsieh, I.H. Yeh, CSPNet: A new backbone that can enhance learning capability of CNN, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 14–19 June 2020, pp. 390–391.
[24] X. Chu, A. Zheng, X. Zhang, J. Sun, Detection in crowded scenes: One proposal, multiple predictions, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020, pp. 12214–12223.
[25] S. Liu, L. Qi, H. Qin, J. Shi, J. Jia, Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018, pp. 8759–8768.
[26] X. Glorot, A. Bordes, Y. Bengio, Deep sparse rectifier neural networks, in Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, Ft. Lauderdale, FL, USA, 11–13 April 2011, pp. 315–323.

[27] S. Elfwing, E. Uchibe, K. Doya, Sigmoid-weighted linear units for neural network function approximation in reinforcement learning, Neural Netw. 107 (2018) 3–11.
[28] V. Vysotska, K. Smelyakov, N. Sharonova, E. Vakulik, O. Filipov, R. Kotelnykov, Fast Color Images Clustering for Real-Time Computer Vision and AI System, CEUR Workshop Proceedings 3664 (2024) 161–177.
[29] M. Nazarkevych, V. Lytvyn, M. Kostiak, N. Oleksiv, N. Naconechnyi, Method of Dataset Filling and Recognition of Moving Objects in Video Sequences based on YOLO, CEUR Workshop Proceedings 3654 (2024) 265–276.
[30] M. Nazarkevych, M. Kostiak, N. Oleksiv, V. Vysotska, A.-T. Shvahuliak, A YOLO-based Method for Object Contour Detection and Recognition in Video Sequences, CEUR Workshop Proceedings 3654 (2024) 49–58.
[31] R.M. Peleshchak, V.V. Lytvyn, M.A. Nazarkevych, I.R. Peleshchak, H.Y. Nazarkevych, Influence of the Symmetry Neural Network Morphology on the Mine Detection Metric, Symmetry 16(4) (2024) 485.
[32] V. Vysotska, N. Sharonova, M. Shirokopetleva, O. Dolhanenko, A. Chupryna, S. Smelyakov, Research of Methods for Image Sharpness Evaluation in Photos of People, CEUR Workshop Proceedings 3664 (2024) 255–272.
[33] Z. Hu, D. Uhryn, Y. Ushenko, V. Korolenko, V. Lytvyn, V. Vysotska, System programming of a disease identification model based on medical images, in Proceedings of SPIE - The International Society for Optical Engineering, 2024, 12938, 129380F.
[34] S. Tsybulia, O. Lavrut, V. Lytvyn, T. Lavrut, M. Nazarkevych, V. Vysotska, Clustering Methods Analysis for Terrain Colors Characteristics Determination, CEUR Workshop Proceedings 3387 (2023) 103-116.
[35] V. Lytvyn, V. Vysotska, V. Mykhailyshyn, A. Rzheuskyi, S. Semianchuk, System Development for Video Stream Data Analyzing, Advances in Intelligent Systems and Computing 1020 (2020) 315-331. doi: 10.1007/978-3-030-26474-1_23.
[36] V. Motyka, Y. Stepaniak, M. Nasalska, V. Vysotska, People's Emotions Analysis while Watching YouTube Videos, CEUR Workshop Proceedings 3403 (2023) 500–525.
[37] O. Yakovleva, A. Kovtunenko, V. Liubchenko, V. Honcharenko, O. Kobylin, Face Detection for Video Surveillance-based Security System, CEUR Workshop Proceedings 3403 (2023) 69-86.
[38] K. Dergachov, L. Krasnov, O. Cheliadin, R. Kazatinskij, Video data quality improvement methods and tools development for mobile vision systems, Advanced Information Systems 4(2) (2020) 85–93. doi:10.20998/2522-9052.2020.2.13.
[39] V. Barsov, O. Plakhotnyi, O. Kosterna, Research of the method of increasing the object determination accuracy on the low-resolution video stream, Advanced Information Systems 5(2) (2021) 91–97. doi:10.20998/2522-9052.2021.2.12.
[40] E. Sabziev, Determining the location of an unmanned aerial vehicle based on video camera images, Advanced Information Systems 5(1) (2021) 136–139. doi:10.20998/2522-9052.2021.1.20.
[41] O. Tymochko, V. Larin, S. Osiievskyi, O. Timochko, A. Abdalla, Method of processing video information resource for aircraft navigation systems and motion control, Advanced Information Systems 4(1) (2020) 140–145. doi:10.20998/2522-9052.2020.1.22.

[42] L. Moskvych, Y. Riepina, K. Shcherbinin, Application of Innovative Approaches to Video Segmentation in a Criminal Process, CEUR Workshop Proceedings 2870 (2021) 1792-1805.
[43] T. Kovaliuk, N. Kobets, G. Shekhet, T. Tielysheva, Analysis of Streaming Video Content and Generation Relevant Contextual Advertising, CEUR workshop proceedings 2604 (2020) 829-843.
[44] D. Reynolds, R. Messner, Video Copy Detection Utilizing the Log-Polar Transformation, International Journal of Computing 15(1), (2016) 8-13.
[45] M. Baharon, M. Abdollah, N. Abu, Z. Abidin, A. Idris, Secure Video Transcoding in Mobile Cloud Computing, International Journal of Computing 17(4) (2018) 208-218.
[46] W. Liu, J. Li, Z. Ye, O. Kochan, CN-Unet: A Robust Network Based on Deep Convolution for Medical Image Segmentation, CEUR Workshop Proceedings 3387 (2023) 29-39.
[47] V. Hnatushenko, P. Kogut, M. Uvarov, On Satellite Image Segmentation via Piecewise Constant Approximation of Selective Smoothed Target Mapping, Applied Mathematics and Computation 389 (2020). doi:10.1016/j.amc.2020.125615.
[48] B. Rusyn, V. Korniy, O. Lutsyk, R. Kosarevych, Deep Learning for Atmospheric Cloud Image Segmentation, in proceedings of IEEE 11 th International Conference on Electronics and Information Technologies, ELIT 2019, pp.188-191.
[49] B. Rusyn, R. Kosarevych, O. Lutsyk, V. Korniy, Segmentation of atmospheric cloud images obtained by remote sensing, in proceedings of International conferences on Advanced in Radioelectronics, Telecomunication and Computer Engineering TCSET 2018, pp.213-216.
[50] R. Kosarevych, et al., Spatial point patterns generation on remote sensing data using convolutional neural networks with further statistical analysis, Scientific Reports 12(1) (2022) 14341.
[51] R.J. Kosarevych, B.P. Rusyn, V.V. Korniy, T.I. Kerod, Image Segmentation Based on the Evaluation of the Tendency of Image Elements to form Clusters with the Help of Point Field Characteristics, Cybernetics and Systems Analysis 51(5) (2015) 704-713.
[52] R. Kosarevych, O. Lutsyk, B. Rusyn, Detection of pixels corrupted by impulse noise using random point patterns, Visual Computer 38(11) (2022) 3719–3730.
[53] B. Rusyn, O. Lutsyk, R. Kosarevych, T. Maksymyuk, J. Gazda, Features extraction from multi-spectral remote sensing images based on multi-threshold binarization, Scientific Reports 13(1) (2023) 19655.
[54] B. T. Truong, S. Venkatesh, Video abstraction: A systematic review and classification, ACM transactions on multimedia computing, communications, and applications 3(1), (2007) 3-es.
[55] J. Eggermont, et al., Optimizing computed tomographic angiography image segmentation using Fitness Based Partitioning, Lecture Notes in Computer Science 4974 (2008) 275–284.
[56] M. Nazarkevych, et al., Evaluation of the Effectiveness of Different Image Skeletonization Methods in Biometric Security Systems, International Journal of Sensors Wireless Communications and Control 11(5) (2021) 542-552. doi: 10.2174/2210327910666201210151809.

[57] M. Medykovskyy, P. Lipinski, O. Troyan, M. Nazarkevych, Methods of protection document formed from latent element located by fractals, in proceedings of IEEE Xth International Scientific and Technical Conference" Computer Sciences and Information Technologies"(CSIT), 2015, September, pp. 70-72.

[58] D. B. Goldman, B. Curless, D. Salesin, S. M. Seitz, Schematic storyboarding for video visualization and editing, Acm transactions on graphics (tog) 25(3) (2006) 862-871.

[59] M. U. Sreeja, B. C. Kovoor, Towards genre-specific frameworks for video summarisation: A survey, Journal of Visual Communication and Image Representation 62 (2019) 340-358.

[60] V. N. Mandhala, D. Bhattacharyya, B. Vamsi, N. Thirupathi Rao, Object detection using machine learning for visually impaired people, International Journal of Current Research and Review 12(20) (2020) 157-167.

[61] N. V. Nguyen, C. Rigaud, J. C. Burie, Digital comics image indexing based on deep learning, Journal of Imaging 4(7) (2018) 89.

[62] M. Nadeem, H. Shen, L. Choy, J. M. H. Barakat, Smart diet diary: real-Time mobile application for food recognition, Applied System Innovation 6(2) (2023) 53.

[63] B. Van Eden, B. Rosman, An overview of robot vision, in proceedings of IEEE Southern African Universities Power Engineering Conference/Robotics and Mechatronics/Pattern Recognition Association of South Africa (SAUPEC / RobMech/ PRASA), 2019, January, pp. 98-104.