Exploring Neural Network Methods for Software of Military Object Detection in UAV Images

Oleksii Bychkov[1], Kateryna Merkulova[1,†], Yelyzaveta Zhabska[1*,†] and Ivan Ivanenko[1,†]

[1] *Taras Shevchenko National University of Kyiv, Volodymyrska str. 64/13, Kyiv, 01601, Ukraine*

Abstract

This paper is dedicated to the study and comparison of military object detection and classification methods in video streams obtained from unmanned aerial vehicles. The main objective is to identify the most effective approach based on predefined quality assessment criteria, particularly for identifying military objects, for further use in software development.

Three detection and classification methods, namely Faster R-CNN, SSD, and YOLO, were used for the study. Three quality criteria for detection and classification methods were developed and described. An algorithm was proposed to determine the required number of images for calculating metrics with a predefined error ε. It was determined that for calculating a metric with an error on the order of 10^{-5}, 250 images are sufficient in the context of the task at hand. For each detection and classification method, corresponding metrics were calculated with the given error, followed by a comparative analysis of these methods based on the three metrics.

During the comparative analysis, none of the methods demonstrated the highest results across all three quality criteria. Therefore, priorities were assigned to each metric based on the specific nature of the task. After analyzing the qualitative metrics of each detection and classification method and considering the chosen priorities, it was concluded that the most effective approach for identifying military objects in UAV video streams is the method based on the YOLO model.

Keywords

UAV, image processing, object detection, pattern matching, image classification, neural networks

1. Introduction

In recent years, a significant number of software solutions have been developed for processing video streams from unmanned aerial vehicles (UAVs), primarily for object detection and analysis purposes. These systems are widely employed across various domains – from agriculture to the defense sector. However, due to considerable differences in task specificity, object types, and imaging conditions, there is an increasing demand for specialized software focused on the detection, identification, and classification of military-related objects.

This need has become particularly relevant in the context of ongoing armed conflict, where UAVs play a critical role in modern warfare — supporting aerial reconnaissance, situational monitoring, and target acquisition. Accordingly, the automated, accurate, and real-time detection of potentially dangerous or suspicious objects in UAV video streams is a key capability for improving the efficiency of military operations, reducing the cognitive load on operators, and enhancing overall situational awareness.

The automation of detecting and classifying military objects in the video stream from UAVs during wartime is one of the key factors for ensuring national security and effective control. Building a video stream processing system that comes from UAVs, creating an automated target recognition

Information Technology and Implementation (IT&I-2025), November 20-21, 2025, Kyiv, Ukraine

* Corresponding author.

† These authors contributed equally.

bos.knu@gmail.com (O. Bychkov); kate.don11@gmail.com (K. Merkulova); y.zhabska@gmail.com (Y. Zhabska); super-ivan-ivanenko@knu.ua (I. Ivanenko)

0000-0002-9378-9535 (O. Bychkov); 0000-0001-6347-5191 (K. Merkulova); 0000-0002-9917-3723 (Y. Zhabska); 0009-0008-0658-9801 (I. Ivanenko)

system, is an extremely important task. Currently, the most complex information processing system is the UAV operator's brain. However, the constant need for heightened attention, significant eye strain, and working at night imposes a heavy burden on the human operator [4].

This topic is highly relevant in wartime conditions, as drones play a key role in modern combat operations. Therefore, fast and accurate detection and classification of military objects in the video stream from UAVs is a crucial task that addresses the urgent security and defense needs of our nation.

2. Analysis of Related Solutions and Problem Definition

Currently, there are a number of software products for processing video streams from UAVs, designed for object detection. Programs like Pix4Dmapper [5], DroneDeploy [6], and AgroScout [7], which were thoroughly reviewed in [4], are actively used in various sectors, ranging from agriculture to military industries. Analyzing these software solutions reveals that they are comprehensive tools focused on specific domains, namely:

- Pix4Dmapper – focuses on surface analysis.
- DroneDeploy – specializes in creating maps and 3D models.
- AgroScout – used for plant disease diagnostics and crop monitoring.

Clearly, the development of software with similar functionalities would not be a novel and unique solution. Therefore, the software being developed will target a different domain—detection and classification of various types of objects related to military applications. This issue is especially relevant under wartime conditions, as drones have become an essential tool in modern warfare. Thus, the rapid identification of suspicious objects in the video stream from UAVs is a critical task in the current context.

3. Research Methods

The task of object detection and classification in a video stream is not a trivial one. Therefore, it is evident that there is currently no exact analytical solution. However, there are already a variety of methods and algorithms that show good results in this area under certain conditions. Identifying the required military objects in the UAV video stream can be a challenging task, and choosing the best detection and classification algorithm is crucial for achieving high-accuracy recognition results.

3.1. Methods based on neural networks

Particularly noteworthy today are the methods based on neural networks, as in recent years they have shown incredible results in various fields of human activity, including tasks related to object detection and classification in video streams. For this reason, it was decided to use one of these methods for the future application. Based on personal experience and open sources from the internet, three of the most popular and widely used methods for object detection and classification based on neural networks were chosen: Faster R-CNN [8], SSD [9], and YOLO [10].

3.1.1. Faster R-CNN

In the first version of R-CNN, there were three distinct stages. First, 2000 region proposals were generated using selective search. Then, these regions were resized to a fixed size. In the final stage, a support vector machine (SVM) with trained weights was applied for classification.

R-CNN turned out to be not as powerful due to the use of selective search, which significantly slowed down the process. Additionally, a large number of cached data had to be stored for the trained network.

Many of these problems were addressed with the release of Fast R-CNN. The entire architecture now consists of a single module, greatly simplifying the training process. One of the key innovations in the developed model was the addition of the ROI (Region of Interest) Pooling layer, designed to produce feature vectors of fixed length. This layer transforms each proposed region into a grid, after which a max-pooling operation is applied to each cell. It is worth noting that Fast R-CNN still uses the selective search algorithm.

In Faster R-CNN, the Region Proposal Network (RPN) was introduced to generate candidate regions, while Fast R-CNN was used for object detection within these regions. These two stages were combined into a single network by sharing features. The RPN takes an image as input and returns a set of coordinates of rectangular regions (which are candidates for classification), along with probability scores indicating the likelihood of an object being present in those regions. RPN is a fully connected convolutional network, meaning it does not contain any fully connected layers. This conceptual solution replaced the selective search algorithm in Fast R-CNN.

3.1.2. YOLO

The R-CNN family of algorithms uses region proposals, which provide good accuracy but can be very slow for certain domains. Another family of algorithms, which has been developing in parallel for object detection, does not use region proposals.

YOLO (You Only Look Once) is a single-stage object detector that achieves both speed and accuracy. This neural network is designed for object detection and is distinguished by its ability to quickly and accurately identify objects in images and videos. The model can process data in real-time. This is achieved because it does not perform the process of object localization at multiple levels of the image, which is typically common in other object detection architectures.

The main difference between this architecture and others is that while some systems apply CNN multiple times to different fragments of the image, YOLO applies CNN once to the entire image at once.

3.1.3. SSD

The SSD (Single Shot Multibox Detector) model utilizes the idea of a pyramidal hierarchy of network outputs for identifying objects at different scales. The image passes sequentially through convolutional layers, which reduce its dimensions. The output signal from the last layer of each size is used to make decisions regarding object detection, forming what is known as the "pyramidal feature" of the image. This allows for object identification at different scales, as the dimensionality of the outputs from the early layers strongly correlates with bounding boxes for small objects, while the outputs from the later layers correlate with bounding boxes for larger objects.

Unlike YOLO, this model does not divide the image into a grid of a fixed size. Instead, it predicts the shifts of key bounding boxes. The boxes at different levels are scaled in such a way that each output layer dimension is responsible for objects of its scale. This means that large objects can only be detected at higher levels, while small objects are detected at lower levels.

3.2. Research Metodology

Thus, the detection and classification methods that will be investigated within this article have been briefly reviewed. Now, let's move on to describing the overall research methodology based on the results that will help determine which method performs best for solving our specific task—namely, the detection and classification of specified military types of objects in video streams from UAVs.

First and foremost, it is essential to define what is meant by the term "quality criterion" within the context of this article. To put it simply, a quality criterion (for a method, technology, solution, or algorithm) is a characteristic or property of the method that can be unambiguously interpreted in a numerical form.

At this stage, it is necessary to determine the parameters to focus on when selecting quality criteria. During the research process, these criteria will be applied to methods for detecting and classifying military objects in video streams from UAVs. Based on personal experience and information from open sources, the most relevant quality indicators for such methods are as follows:

1. Ratio of correctly identified objects to the total number of objects – this criterion allows to assess how effectively the model identifies objects in the video stream.
2. Intersection over Union (IoU) – this metric measures the accuracy with which the model determines the location of objects in the video stream.
3. Average object localization time – this criterion reflects the processing and localization speed of objects by the investigated method.

Thus, all three criteria cover the main characteristics of the object detection and classification method, namely: the ability to identify, localization accuracy, and recognition speed. It is also important to note that each of these metrics will not be calculated for a single image but for a sample of size N, meaning the average value will be used. Below is a description of each of the proposed quality criteria.

3.2.1. The ratio of correctly identified objects to the total number of objects

The full name of the first quality criterion essentially describes its content. For convenience, to avoid repeating the full name of this metric, let's denote it, for example, by the symbol R. To calculate the quality criterion R for the detection and classification method, the following formula is used:

$$R = \frac{\sum_{i=1}^{N} m_i}{\sum_{i=1}^{N} k_i}, \tag{1}$$

where *N* is the total number of images, m_i is the number of objects that the method correctly identified in the *i*-th image, and k_i is the actual number of objects in the *i*-th image.

Before applying the formula above, it is necessary to determine in which cases the method is considered to have correctly detected and classified an object in an image [11]. To do this, it is advisable to use the Intersection over Union (IoU) quality evaluation metric, which will be discussed in more detail later. The value of the IoU metric can range from 0 to 1, with values greater than 0.5 generally considered to indicate a good prediction of the object detector, while values below this threshold indicate ineffective prediction [12, 13]. Therefore, when counting the correctly identified objects, for each object, the IoU value is calculated. If IoU > 0.5, the object is included in the count, and if the value is lower, it is ignored. To calculate the number of correctly identified objects m_i in the *i*-th image, the following formula is used:

$$m_i = \sum_{j=1}^{k_i} \begin{cases} 1, & IoU_j > 0.5, \\ 0, & IoU_j \leq 0.5, \end{cases} \tag{2}$$

where k_i is the actual number of objects in the *i*-th image, IoU_j is the IoU metric value calculated for the *j*-th object in the *i*-th image. Therefore, when calculating the quality criterion *R*, formula 2 will be integrated into formula 1.

3.2.2. Intersection over Union

Intersection over Union is an evaluation metric used to measure the accuracy of an object detector (in our case, various military objects) on a specific dataset. Any algorithm that provides predicted bounding boxes for objects in an image can be evaluated using the Intersection over Union metric [14]. The IoU metric is calculated using the formula:

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union}, \tag{3}$$

where *Area of Overlap* is the area of the intersection between the predicted and actual bounding boxes, and *Area of Union* is the area of the union between the predicted and actual bounding boxes.

Considering formula 3, it can be noted that the possible values of the IoU metric range from 0 to 1, including these extreme values. It is generally considered that IoU > 0.5 indicates a good prediction of the object detector [15]. Now, let's construct a new formula to calculate the average value of the IoU quality criterion, IoU_c:

$$IoU_c = \frac{\sum_{i=1}^{N} IoU_i}{N}, \tag{4}$$

where IoU_i is the value of the IoU quality metric calculated for the i-th image, and N is the total number of images in the sample.

Since one image may contain more than one object, the averaged value should also be calculated for each individual image [16]. The calculation of IoU_i for the i-th image is performed using the formula:

$$IoU_i = \frac{\sum_{j=1}^{m_i} IoU_j}{m_i}, \tag{5}$$

where IoU_j is the value of the IoU quality metric calculated for the j-th object on the i-th image, and m_i is the number of correctly localized objects on the i-th image.

3.2.3. Average object localization time

This quality evaluation criterion is designed to demonstrate the speed of the method under investigation. In short, this criterion refers to the time required to identify a single object in an image using a specific identification method [17]. For ease of notation, the T symbol will be used to represent this quality criterion. The following formula is used to calculate the average value of criterion T:

$$T = \frac{\sum_{i=1}^{N} T_i}{N}, \tag{6}$$

where Ti is the average time to identify an object on the i-th image, and N is the total number of images in the sample. To briefly describe formula number 6, let's calculate the metric T separately for each image, then sum all the obtained values, and finally divide the result by the total number of images [18].

The calculation of the quality evaluation metric T_i for the i-th image is done using the formula:

$$T_i = \frac{\sum_{i=1}^{m_i} b_i - a_i}{m_i}, \tag{7}$$

where b_i is the time at which the object identification process ends on the image, a_i is the time at which the object identification process starts on the image, and m_i is the number of correctly identified objects on the i-th image [19].

3.3. Error Estimation

To perform a comparative analysis of the described methods based on the proposed quality criteria, it is necessary to compute them for a sample of images of size N [20]. This is done to ensure that the obtained results are as objective as possible.

The determination of the required number of images N for calculating the quality criterion R with a given error ε is carried out using the formula:

$$\varepsilon = |f(n + step) - f(n)|, \tag{8}$$

where ε is the given error for computing *f*, *f(N)* is the metric value for a specific object identifier using a sample of N images, n is the current number of images, and *step* is a fixed increment that increases n for each iteration.

4. Research Results

For further research, three models were used: Faster R-CNN, SSD, and YOLO. The first two models were implemented using the TensorFlow framework, a popular open-source machine learning framework developed by Google. It is used for creating, training, and deploying various machine learning and deep learning models. Meanwhile, the YOLO model was implemented using the PyTorch framework, developed by Facebook and widely used by researchers and engineers worldwide. During training, monitoring was conducted to assess the effectiveness of the model training process. This control involved analyzing loss functions such as Classification loss and Localization loss.

Classification loss is a loss function applied during the training of a neural network for classification tasks. It measures the difference between predicted and actual object classes, helping to adjust the model during training [21].

Localization loss is a loss function used to train a neural network to determine the location of objects in an image. It evaluates the error between predicted and actual object coordinates, improving localization accuracy during the model's training process [22].

Below, Figures 1 and 2 show the graphs of the above-mentioned loss functions for each of the trained models.

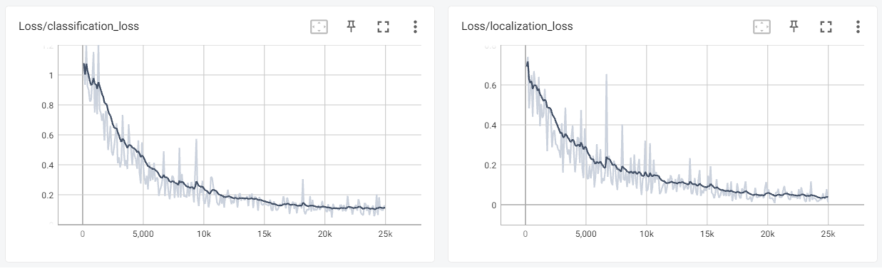

Figure 1: Training process of the SSD model.

In Figure 3, the corresponding graphs reflecting the training process of the model are shown. The graphs have a slightly different style because a different framework was used for the implementation of this model.

The provided graphs for the three models reflect the overall trend of their training. To evaluate and compare the obtained quantitative results, Table 1 is presented, which shows the loss function values for each model after the completion of the training process.

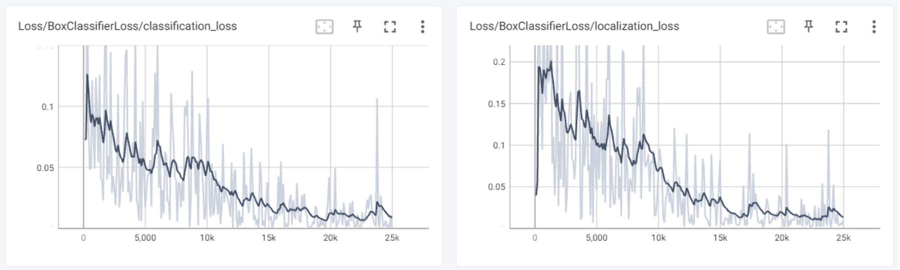

Figure 2: Training process of the Faster R-CNN model.

The smaller the loss function value, the more effectively the model performs the given task. In other words, a lower loss function value indicates that the model's predictions are closer to the expected results. However, this statement should not be taken too literally. In general, if the loss function value is between 0 and 1, it is considered a good result. On the other hand, if the loss function values are almost zero, it could indicate overfitting. In this case, the model will perform perfectly on the training dataset but will perform poorly on new data, as it has adapted too much to the training set. Therefore, the table above does not reflect the real situation, as it shows data related to the training process.

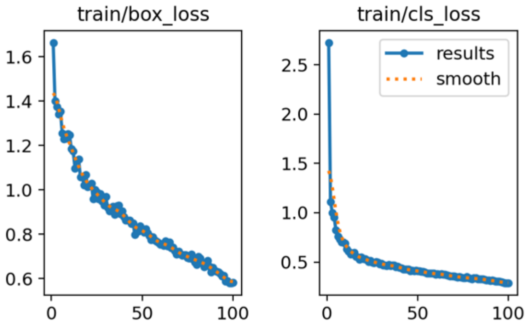

Figure 3: The training process of the YOLO model.

Thus, it is now necessary to check the performance of the implemented methods in practice. This can be done using the proposed quality criteria, which were discussed earlier. At this point, all the necessary formulas to calculate the quality criteria are in place, meaning the theoretical base is ready.

Next, the number of images required to calculate the quality criteria needs to be determined. There is a lower limit to the number of images in the sample, which ensures sufficient objectivity of the obtained results when using the given number of images for calculations. In other words, a sample that is too small would lead to results that lack objectivity. To quantify the necessary objectivity of the calculations, the allowable error value ε will be used. The question now is how many images N are required to calculate the quality criterion f with a specified error ε. This issue was discussed in the second section, where it was stated that the maximum value of ε, corresponding to sufficient objectivity, is considered to be 10^{-5}.

Table 1

Final training results of the models

Experiment	classification_loss	localization_loss
Faster R-CNN	0.01114	0.01048
SSD	0.1289	0.07844
YOLO	0.579	0.2881

To calculate the required number of images N that would satisfy the error of 10^{-5}, an algorithm was programmed, and its flowchart can also be seen in the second section. As a result, it was found that for calculating the given quality criteria, 250 images would be sufficient.

As a result of conducting all necessary experiments, Table 2 was formed.

Table 2

Results of quality criteria calculations for identification methods

	R	IoUc	T
Faster R-CNN	0.94706	0.94158	0.09325
SSD	0.90882	0.89421	0.0478
YOLO	0.96176	0.90083	0.03345

Based on the obtained results, a comparative analysis of the selected detection and classification methods will be conducted. This includes both qualitative and quantitative comparisons using the computed quality criteria. The qualitative comparison of detection and classification methods based on a specific metric involves determining which method has the best value for this metric (whether it is the highest or the lowest). After the qualitative comparison, a quantitative comparison is conducted to determine how much one detection and classification method outperforms another based on the chosen metric. The quantitative comparison was carried out using the *P* indicator:

$$P = \frac{A - B}{B} * 100\%, A \geq B, \tag{9}$$

where *P* is the number that shows by what percentage *A* exceeds *B*, *A* is the metric value for the first detection and classification method, and *B* is the metric value for the second detection and classification method.

Let’s consider the quality criterion Intersection over Union (IoU), which measures the accuracy of predicting the object’s location by detection and classification methods. In Figure 4, graphs are presented that illustrate the dependence of the average IoU value for each method on the number of images *N* required to calculate this indicator.

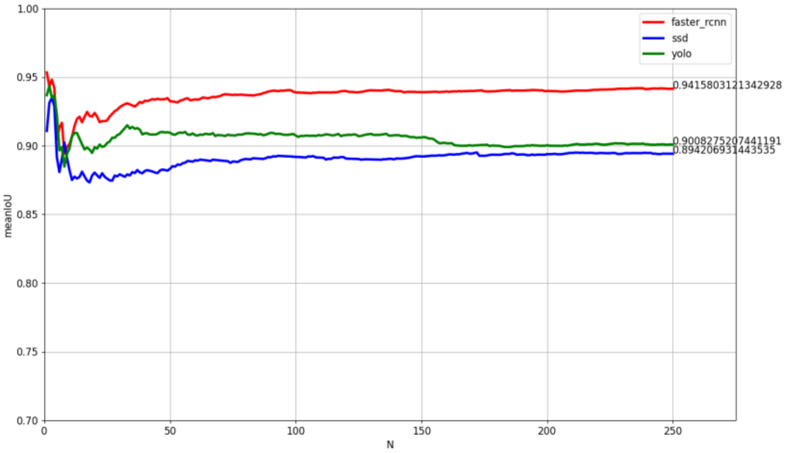

Figure 4: *IoU(N)* function graphs for three methods.

After substituting the corresponding values into Formula 9, the following results are obtained:

$$P_{IoU} = \frac{0{,}94158 - 0,90083}{0,90083} * 100\% \approx 4{,}5\%, \tag{10}$$

$$P_{IoU} = \frac{0{,}94158 - 0{,}89421}{0{,}89421} * 100\% \approx 5{,}3\%. \tag{11}$$

During the quantitative comparison of detection and classification methods using the IoU metric, it was found that the Faster R-CNN model predicts the location of the specified military targets 4.5% more accurately compared to the YOLO model, and 5.3% more accurately than the model based on the SSD architecture.

To compare the methods based on the *R* metric, which reflects the ratio of the number of correctly localized military targets to the total number of targets, let’s consider the physical meaning of this metric, which numerically characterizes the method's ability to correctly identify the given objects, in particular, vehicles. Figure 5 shows three graphs that illustrate the relationship between the *R* metric value for each detection and classification method and the number of images *N* required for the calculation of the metric.

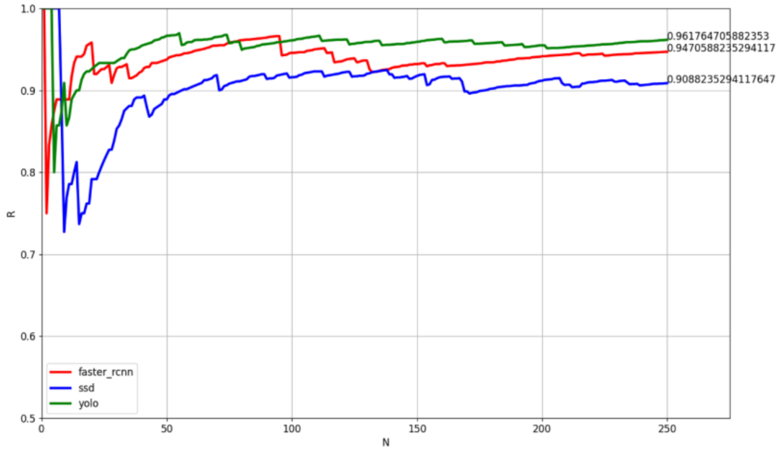

Figure 5: Graphs of the *R(N)* function for three methods.

After substituting the corresponding values into the formula, the following result is obtained:

$$P_R = \frac{0{,}96176 - 0{,}94706}{0{,}94706} * 100\% \approx 1{,}6\%, \tag{12}$$

$$P_R = \frac{0{,}96176 - 0{,}90882}{0{,}90882} * 100\% \approx 5{,}8\%. \tag{13}$$

After the quantitative comparison of detection and classification methods using the R metric, it can be concluded that the method based on the YOLO model has a 1.6% better ability to detect military targets compared to the Faster R-CNN method and a 5.8% better performance compared to the method based on the SSD architecture.

The last quality criterion for comparing methods is the average detection and classification time for a single military target. Figure 6 shows the graphs demonstrating the dependence of the T metric value for each method on the number of images N required for its calculation.

In this case, formula 9 is not optimal, as at first glance it is obvious that one number is significantly larger than the other. For a more visual comparison, it is better to determine how many times one value exceeds the other:

$$\frac{0{,}09326}{0{,}03345} \approx 2{,}79, \tag{14}$$

$$\frac{0{,}0478}{0{,}03345} \approx 1{,}43. \tag{15}$$

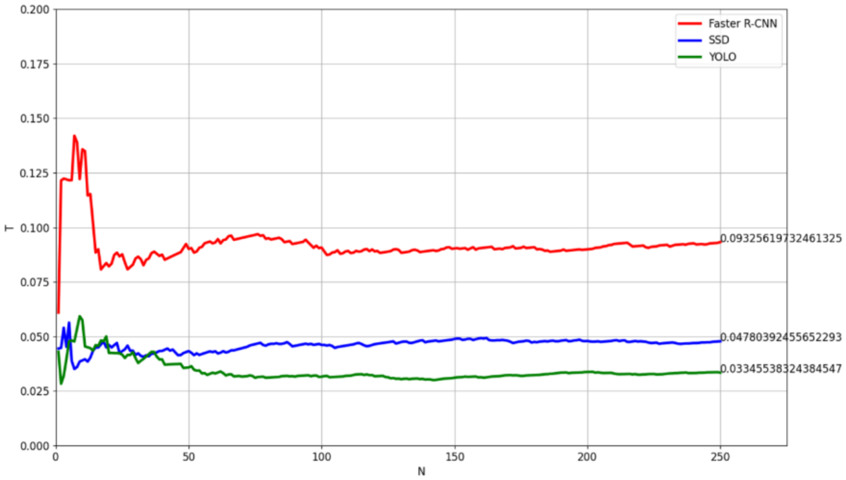

Figure 6: Graphs of the *T(N)* function for three methods.

Thus, as a result of the quantitative comparison of the three detection and classification methods using the T metric, it can be concluded that the YOLO-based method detects and classifies a military target on average 2.79 times faster than the Faster R-CNN-based method and 1.43 times faster than the SSD-based method.

5. Conclusion

After conducting a comparative analysis of the studied methods, it is difficult to definitively determine which one is the best for detecting and classifying military targets captured by UAVs, as none of the methods demonstrated the best results across all three quality criteria. In other words:

1. The Faster R-CNN-based method predicts the location of military targets most accurately (best for the IoU metric).
2. The YOLO-based method identifies military targets the fastest (best for the T metric).
3. There is no clear difference in terms of detection and classification capability between Faster R-CNN and YOLO, as both perform equally well. However, when looking at the numbers, YOLO shows slightly better results than Faster R-CNN in the *R* quality criterion. This advantage, though, is minimal (only 1.6%) and could be considered within the margin of calculation error.

As observed from the results, the SSD-based method did not outperform in any of the three quality criteria, so it is not considered a viable candidate for the future implementation of a military target detection and classification system for UAV video streams.

Thus, a more detailed analysis of the results for the Faster R-CNN and YOLO methods is necessary to determine the most relevant method based on the computed quality criteria. Since neither method showed the best results across all three criteria, the next step is to prioritize each quality criterion. This will allow for a selection based on the context of the task to be solved using these methods.

The *R* quality criterion, which evaluates the ability of the method to detect and classify military targets, is the most important and should be given the highest priority. This is because methods with a higher *R* value are capable of detecting more camouflaged military targets, such as people in military uniforms blending with the environment. This is particularly important in the context of territory monitoring and control using drones. The *T* quality criterion, which characterizes the speed of the method, should be given second priority because the methods will operate not with static images, but with real-time video streams from drones. To process and analyze all frames from the video stream in time, high-speed performance is essential. If the speed is insufficient, frames that may contain military targets will be skipped, resulting in missed detections. Therefore, the IoU quality criterion, which assesses the accuracy of object localization, takes third priority, as it is important for evaluating placement accuracy, but it does not carry as much weight as the ability to identify and classify objects.

Based on the established priorities, the most relevant method for the given context will be determined. According to the *R* quality criterion, which holds the highest priority, the Faster R-CNN and YOLO methods showed nearly identical results, making it impossible to determine a clear winner at this point. Moving to the *T* metric, which has the second priority, the clear winner is the YOLO-based method, as it is almost three times faster than the Faster R-CNN-based method based on the obtained results. Considering the final priority criterion, IoU, it was noted that Faster R-CNN performed 4.5% better than YOLO, but this advantage is minimal and not significant, as the IoU criterion holds the lowest priority. Therefore, given that the YOLO-based method was nearly three times faster than the Faster R-CNN-based method, the YOLO-based method was selected for the further development of the military target detection and classification system in UAV video streams.

Declaration on Generative AI

During the preparation of this work, the authors used GPT-4 to check grammar and spelling.

References

[1] M. Zeng, S. He, Q. Zeng, Y. Niu, R. Zhang, PA-YOLO: Small Target Detection Algorithm with Enhanced Information Representation for UAV Aerial Photography, in IEEE Sensors Letters. doi:10.1109/LSENS.2025.3550406.

[2] Y. Wang, W. Wei, Z. Xin, D. Li, Research on Small Face Detection in UAV Command and Control System, 2022 International Conference on Cyber-Physical Social Intelligence (ICCSI), Nanjing, China, 2022, pp. 69-72. doi:10.1109/ICCSI55536.2022.9970688.

[3] V. Petrivskyi, V. Shevchenko, O. Bychkov, O. Pokotylo, Models and Information Technologies of Coverage of the Territory by Sensors with Energy Consumption Optimization, In: Mathematical Modeling and Simulation of Systems, MODS 2021, Lecture Notes in Networks and Systems, vol. 344, Springer, Cham, 2021. doi:10.1007/978-3-030-89902-8_2.

[4] K. Merkulova, Y. Zhabska, I. Ivanenko, Software for UAV Images Processing for Object Identification, 20th International Scientific Conference "Dynamical System Modeling and Stability Investigation", DSMSI 2023 - Volume 1: Mathematical Foundations of Information Technologies, CEUR Workshop Proceedings, vol. 3687, pp. 25-34, 2023. URL: https://ceur-ws.org/Vol-3687/Paper_3.pdf.

[5] Pix4Dmapper. URL: https://www.pix4d.com/product/pix4dmapper-photogrammetry-software/

[6] DroneDeploy. URL: https://www.dronedeploy.com/.

[7] AgroScout. URL: https://agro-scout.com/.

[8] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137-1149, 1 June 2017. doi:10.1109/TPAMI.2016.2577031.

[9] Y. Zhou, S. Cong, Improved Transformer-Based SSD Detector for Airborne Object Detection, 2022 4th International Conference on Frontiers Technology of Information and Computer (ICFTIC), Qingdao, China, 2022, pp. 18-22. doi:10.1109/ICFTIC57696.2022.10075226.

[10] Y. Song, Z. Xie, X. Wang, Y. Zou, MS-YOLO: Object Detection Based on YOLOv5 Optimized Fusion Millimeter-Wave Radar and Machine Vision, in IEEE Sensors Journal, vol. 22, no. 15, pp. 15435-15447, 1 Aug.1, 2022. doi:10.1109/JSEN.2022.3167251.

[11] O. Bychkov, K. Merkulova, Y. Zhabska, Information Technology of Person's Identification by Photo Portrait, 2020 IEEE 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), Lviv-Slavske, Ukraine, 2020, pp. 786-790. doi:10.1109/TCSET49122.2020.235542.

[12] Z. Li, F. Liu, W. Yang, S. Peng, J. Zhou, A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects, in IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 12, pp. 6999-7019, Dec. 2022. doi:10.1109/TNNLS.2021.3084827.

[13] C. A. R. Goyzueta, J. E. C. De la Cruz, W. A. M. Machaca, Integration of U-Net, ResU-Net and DeepLab Architectures with Intersection Over Union metric for Cells Nuclei Image Segmentation, 2021 IEEE Engineering International Research Conference (EIRCON), Lima, Peru, 2021, pp. 1-4. doi: 10.1109/EIRCON52903.2021.9613150.

[14] O. Bychkov et al., Using Neural Networks Application for the Font Recognition Task Solution, 2020 55th International Scientific Conference on Information, Communication and Energy Systems and Technologies (ICEST), Niš, Serbia, 2020, pp. 167-170. doi:10.1109/ICEST49890.2020.9232788.

[15] S. R. Sitaraman, M. V. S. Narayana, J. Lande, L. M, A. H. Shnain, Center Intersection of Union loss with You Only Look Once for Object Detection and Recognition, 2024 International

Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS), Hassan, India, 2024, pp. 1-4. doi:10.1109/IACIS61494.2024.10721907.

[16] I. Yurchuk, A. Pylypenko, Quantile-Based Statistical Techniques for Anomaly Detection, Proceedings of the XX International Scientific Conference “Dynamical System Modeling and Stability Investigation” (DSMSI-2023), CEUR Workshop Proceedings, vol. 3746, 2023, pp. 64-73. URL: https://ceur-ws.org/Vol-3746/Paper_7.pdf.

[17] O. Bychkov, O. Ivanchenko, K. Merkulova, Y. Zhabska, Mathematical Methods for Information Technology of Biometric Identification in Conditions of Incomplete Data, Proceedings of the 7th International Conference "Information Technology and Interactions" (IT&I-2020), CEUR Workshop Proceedings, vol. 2845, 2020, pp. 336-349. URL: https://ceur-ws.org/Vol-2845/Paper_31.pdf.

[18] Y. R. T. Bethi, S. Narayanan, V. Rangan, A. Chakraborty, C. S. Thakur, Real-Time Object Detection and Localization in Compressive Sensed Video, 2021 IEEE International Conference on Image Processing (ICIP), Anchorage, AK, USA, 2021, pp. 1489-1493. doi:10.1109/ICIP42928.2021.9506769.

[19] Y. E. Kang, W. Kang, T. Lee, H. S. Chwa, Paste-and-Cut: Collective Image Localization and Classification for Real-Time Multi-Camera Object Detection, 2023 14th International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Republic of Korea, 2023, pp. 740-742. doi:10.1109/ICTC58733.2023.10393851.

[20] S. Toliupa, A. Pylypenko, O. Tymchuk, O. Kohut, Generator for Testing Data Analytics Methods, Proceedings of the XX International Scientific Conference “Dynamical System Modelling and Stability Investigation” (DSMSI-2023), CEUR Workshop Proceedings, vol. 3687, 2023, pp. 11-24. URL: https://ceur-ws.org/Vol-3687/Paper_2.pdf.

[21] A. S. Pal, A. Panda, U. Garain, Label Dependency Aware Loss for Reliable Multi-Label Medical Image Classification, ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Hyderabad, India, 2025, pp. 1-5. doi:10.1109/ICASSP49660.2025.10888215.

[22] M. W. P. Maduranga, U. Oruthota, H. K. I. S. Lakmal, S. Kulatunga, RSSI-Based Indoor Localization Using Deep Learning with A Custom Loss Function, 2024 8th SLAAI International Conference on Artificial Intelligence (SLAAI-ICAI), Ratmalana, Sri Lanka, 2024, pp. 1-5. doi: 10.1109/SLAAI-ICAI63667.2024.10844973.