Machinery & Energetics
Vol. 15, No. 2. 2024

Journal homepage:
https://technicalscience.com.ua/en

UDC 681.518:621.311.25
DOI: 10.31548/machinery/2.2024.33

Ruslan Yermolenko*
PhD in Physical and Mathematical Sciences, Associate Professor
Taras Shevchenko National University of Kyiv
01033, 60 Volodymyrska Str., Kyiv, Ukraine
https://orcid.org/0009-0000-0854-4668

Denys Klekots
Student
Taras Shevchenko National University of Kyiv
01033, 60 Volodymyrska Str., Kyiv, Ukraine
https://orcid.org/0000-0002-4251-2958

Olga Gogota
PhD in Physical and Mathematical Sciences, Researcher
Taras Shevchenko National University of Kyiv
01033, 60 Volodymyrska Str., Kyiv, Ukraine
https://orcid.org/0000-0003-4108-7256

Development of an algorithm for detecting commercial unmanned aerial vehicles using machine learning methods

Abstract. This study aimed to train algorithms for detecting commercial unmanned aerial vehicles using machine learning techniques. Neural network architectures YOLOv8 and MobileNetV3 were used to detect unmanned aerial vehicles in images and videos. The models used were pre-trained on the ImageNet dataset and then refined on the SimUAV dataset containing images of four types of drones (Parrot A.R. Drone 2.0; DJI Inspire I; DJI Mavic 2 Pro; and DJI Phantom 4 Pro), different sizes and in eight different background locations. The study confirmed that the combination of the YOLOv8 and MobileNetV3 architectures has significant potential for detecting commercial unmanned aerial vehicles in various types of images. The trained models demonstrated high performance in the recognition and classification of unmanned aerial vehicles, achieving an average detection accuracy (at an IoU threshold of 50%) of 0.747 and 0.909 for the MobileNetV3_Small and MobileNetV3_Large models, respectively. This demonstrates the high efficiency and accuracy of the models in detecting objects on the test data. The results of the study also included the values of the binary cross-entropy metric, which were 0.308 and 0.216, respectively, indicating the high accuracy of the models in object classification and confirming the high efficiency and reliability of these models in working with objects on the test data. During the study, the MobileNetV3_Large model showed more accurate results than MobileNetV3_Small, which indicates its higher efficiency in detecting and classifying aircraft. The obtained results confirm the prospects of applying machine learning methods in the field of monitoring and security systems, which reliably detect and track unmanned aerial vehicles in various conditions. The high performance of the trained models demonstrates their effectiveness in real-world operating conditions, making them a valuable tool for solving important control and supervision tasks

Keywords: convolutional neural network; object classification; YOLOv8; MobileNetV3; computer vision; binary cross-entropy

Article's History: Received: 16.02.2024; Revised: 10.05.2024; Accepted: 29.05.2024.

Suggested Citation:
Yermolenko, K., Klekots, D., & Gogota, O. (2024). Development of an algorithm for detecting commercial unmanned aerial vehicles using machine learning methods. *Machinery & Energetics,* 15(2), 33-45. doi: 10.31548/machinery/2.2024.33.

*Corresponding author

INTRODUCTION

The growing popularity and availability of unmanned aerial vehicles (UAVs) introduce new technologies and open many new opportunities in various industries, such as commercial activities, industrial production, and research and exploration. However, at the same time, the growth in the use of drones has also raised serious security constraints and concerns, as some drones can be used by malicious actors for illegal or harmful purposes, creating a need for improved methods of detecting and controlling them. This problem creates a need for effective solutions for fast and reliable detection of unmanned aerial vehicles, even in remote or difficult conditions. Existing methods are often insufficiently efficient or slow to effectively monitor such devices. Therefore, there is a need to develop new technologies and algorithms aimed at detecting and identifying unmanned aerial vehicles, which will ensure effective control and prevent security threats.

Machine learning methods have great potential in this area, as they can be used to automate the process of detecting and identifying objects in images. The use of machine learning algorithms can be used to create models that can analyse and interpret large amounts of data with high speed and accuracy (Pidpalyi, 2024). Such models can perform a variety of tasks, including detecting objects of different types and classes. An important aspect of using machine learning methods is their ability to adapt and learn from new data. This allows the models to continuously improve and adapt to changing environmental conditions, which is critical in the case of drone detection. In addition, the use of machine learning methods allows automation of the detection process and provides faster response times, which is important in the case of detecting potentially dangerous devices.

One of the most important areas of application of neural networks is their use in image processing, which opens wide opportunities for solving various tasks in the field of computer vision (Najjar & Baskaya, 2022). L. Chen *et al.* (2021) analysed the effectiveness of different convolutional neural network architectures for image classification and object recognition tasks, which highlighted the advantage of deep architectures over traditional machine learning methods, especially in tasks such as object recognition. Y. Wang *et al.* (2020) studied the impact of different optimisation algorithms and activation functions on the accuracy of models in image processing tasks, showing that the choice of optimal parameters can significantly affect the quality and efficiency of neural networks in image and video analysis tasks. M. Hammami *et al.* (2020) also focused on developing new data augmentation methods to improve model training in the face of insufficient image data, which improved their overall performance. However, such studies are often limited to specific aspects or contexts and do not incorporate all implementation options, which requires additional study and analysis.

The use of neural networks opens opportunities for their adaptation to new conditions and tasks, making them more flexible and versatile in various applications. A. Taherkhani *et al.* (2020) conducted a detailed analysis of the ability of neural networks to transfer learning to different tasks and datasets, identifying the key factors affecting their versatility and efficiency. M.M. Bejani & M. Ghatee (2020) addressed the impact of different regularisation methods on the overall performance and versatility of neural networks. K.M.R. Alam *et al.* (2020) analysed the possibility of using ensembles of neural networks to improve the accuracy and reliability of results. Such studies, however, focus only on certain features of the use of machine learning technologies, which may limit their applicability in real-world applications and require more detailed research.

The development of neural networks also allows them to not only reproduce existing knowledge but also learn from new data and experiences, ensuring their ability to continuously improve and adapt to changing conditions (Janulin *et al.*, 2022). S. Tishchenko & E. Kuznetsov (2024) investigated the capabilities of neural networks in the context of solving prediction and classification tasks in dynamic learning, emphasising their ability to effectively adapt to changes in the environment. N.O. Kushnir *et al.* (2022) addressed the mechanisms of neural networks' self-tuning during training, enabling automatic adjustment of their parameters to achieve optimal performance and accuracy. C.J. Lin *et al.* (2021) analysed the role of neural networks in resource-constrained learning environments, where they have shown the ability to efficiently use computing power and data, making them attractive for use in limited hardware. However, each case of using neural networks is unique and requires an individual approach, considering specific conditions, data, and the task at hand (Song *et al.*, 2018). Each research or application of neural networks has its peculiarities that should be addressed when developing and analysing models.

The study aims to investigate the effectiveness of MobileNetV3_Small and MobileNetV3_Large models in detecting commercial unmanned aerial vehicles in images, in different conditions and application modes. The main tasks will include training these models on a suitable dataset and evaluating them for object detection and classification.

MATERIALS AND METHODS

The key algorithms used in this study were neural network architectures for detecting objects in images and videos, such as YOLOv8 and MobileNetV3 (in particular, MobileNetV3_Small_ImageNet and MobileNetV3_Large_ImageNet). To use a combination of these models, the keras_cv library was used as it provides convenient access to these models and their functionality (The YOLOV8Detector model, n.d.). These models were managed through an application programming interface (API), which ensured ease of use and integration with other system components. The model used for training was pre-trained on the widely used ImageNet dataset (ImageNet Large Scale..., n.d.), which gave it a basic understanding of the different images and patterns in the images. After the preliminary training, the model was further trained on the SimUAV dataset (n.d.), which was divided into three parts: 10% for validation, 10% for testing and 80% for training. This dataset included 29,568 colour images of four models of unmanned aerial vehicles, presented in 640×640 pixels format. The images were artificially generated in different locations and under different lighting conditions to provide diversity and versatility in object recognition for the model.

Considering the pre-training of both models, the study decided to focus on retraining only the last layers in the YOLOv8 model. This strategy implied that all layers after tf.concat_5 would be retrained while maintaining the previously trained weights for the rest of the model. This approach retained the information gained from the previous training, while also allowing the model to tune its parameters to better match the specific characteristics of the SimUAV dataset. During the retraining process, the corresponding APIs of the keras_cv library were used, which provided convenient access to the model functionality and allowed for efficient implementation of changes in model weights following the specific requirements and characteristics of the SimUAV dataset. The retraining process involved adapting the model's weights, addressing the output formats, input and output layer sizes, and optimisation parameters. This strategy has significantly reduced the time required to train the model while ensuring high-quality results and compliance with the requirements of the UAV detection task.

The task of object detection implies the presence of two output levels in the model, namely classification and regression, each of which uses a separate loss function. For the classification layer, a binary cross-entropy loss function was used to facilitate the model correctly classifying objects in the images. For the regression level, the ratio of the intersection area of the rectangles described around the object to the area of their union was used (also known as interaction over union (IoU) loss function), which allowed the model to accurately determine the boundaries of the objects. The final loss function was calculated as the sum of the loss functions of the classification and regression levels. To evaluate the efficiency and accuracy of the model, the mean average precision (mAP) metric for the threshold value of the IoU was used, thus assessing the overall ability of the model to detect objects in images. After training, the model with the best mAP was selected, evaluated on the validation dataset, and used to evaluate the metrics on the test dataset.

RESULTS

There are many different methods and technologies used to detect drones, which can be divided into two main categories: non-optical and optical. Non-optical approaches include the use of sound, radar and radio frequency methods. Analysing the acoustic characteristics of drones' rotors can detect their presence, but this method is not always effective in noisy environments such as airports and cities. Radars are widely used to detect large aerial objects but require special modifications to detect drones and can suffer from noise. Radio-frequency methods allow drone signals to be intercepted and analysed, which is an effective way to detect over long distances.

Optical approaches, in turn, are based on image processing. Traditional methods include manual feature extraction from images, such as histograms of oriented gradients (HOG) and others. However, these methods, despite being quite fast, usually have low accuracy. In contrast, deep learning methods use neural networks to automatically extract features, which results in higher-quality object recognition in images (Rahmad *et al.*, 2020). In the research field of computer vision, there are usually three main categories of tasks: classification, detection and segmentation. The construction of a bounding box surrounding the drone in the frame requires information about what and where the object is in the image, which belongs to the object detection category.

Two-stage detectors are complex models divided into two stages for more efficient detection of objects in images. The first stage involves the creation of proposed areas where the required facilities could potentially be located. This process is carried out employing convolutional neural networks such as RCNN, Fast RCNN, and Faster RCNN (Fig. 1), which use base networks such as VGG-16 or ResNet to generate feature maps (Nguyen & Huynh, 2023).

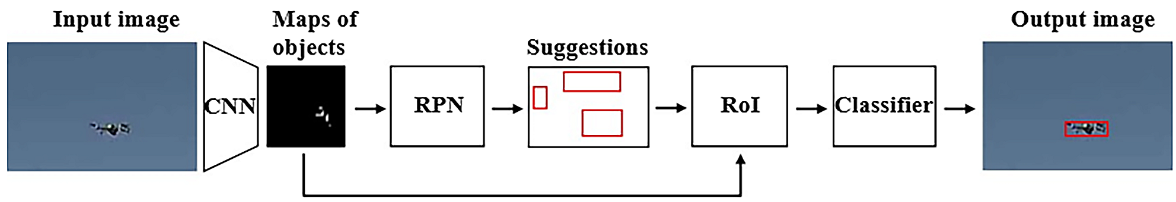

Figure 1. Scheme of a two-stage structure of object detection on the example of Faster R-CNN
Source: compiled by the authors

Then, the network responsible for generating region-of-interest (RoI) suggestions determines the locations where objects can be found. The proposed regions are then merged with the corresponding feature maps using maximum merging to create fixed-size feature maps. The obtained results are transferred to a classifier that determines the class of the object and builds a restrictive framework around it (Raja *et al.*, 2020).

Two-stage detectors usually provide higher accuracy than single-stage models, but their speed may not be sufficient for real-time applications. However, the ability to accurately detect objects makes them an important tool in the field of computer vision. One-step object detectors are innovative models that abandon the traditional two-step approach to object detection and instead provide bounding boxes and object classes simultaneously. This opens up new perspectives in the world of computer vision, allowing for faster and more efficient real-time object detection. One-stage detectors differ from their two-stage counterparts in that they operate without a phase of region proposal. Complex deep learning networks are used to simultaneously predict the position of objects and their classes. This provides faster and more efficient solutions to object detection tasks on large volumes of data. Two examples of one-step detectors are You Only Look Once (YOLO) and Single Shot Detector (SSD). Both methods use different strategies to solve object detection problems (Shuai & Wu, 2020; Jiang *et al.*, 2022). The SSD strategy relies on feature-matching functions to predict bounding boxes and object classes. On the contrary, YOLO technology divides the image into a grid of cells at three different scales, each of which is responsible for predicting anchor boxes and class probabilities (Fig. 2).

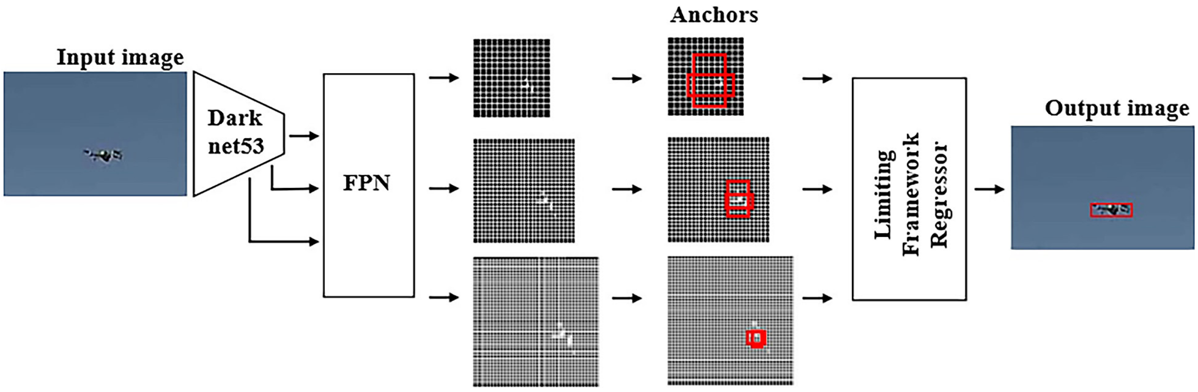

Figure 2. Scheme of the one-stage structure of object detection on the example of YOLOv3
Source: compiled by the authors

It is worth noting that YOLOv8 is an advanced model in the field of object recognition. It features a complex architecture, which includes several key components, each of which plays an important role in the data processing process (Fig. 3).

The Backbone plays a key role in extracting information from the input data in the YOLOv8 model. This model uses CSP Darknet53, which is an improved version of the previous models. Darknet53 is a basic architecture based on the idea of cross-stage partial connections (CSP). This architecture is noted for its ability to efficiently process large amounts of data and extract important features. The use of CSP Darknet53 in YOLOv8 provides faster data processing and improved object detection accuracy using advanced data processing technologies. Such a network can keep important features from the input images and transfer them for further analysis and classification of objects (Zou *et al.*, 2023).

"Neck" is used to combine features from different levels of the spine. For this purpose, the Path Aggregation Network (PANet) technology is used, which is the basis for the optimal integration of information from different levels. PANet provides adaptive feature weighting, where each feature is assigned a weight according to its importance. This allows the model to focus on the most informative features, increasing its ability to recognise objects. The panel effectively combines information from different layers of the ridge and uses it for further analysis and classification of objects in the images.

"Head" is an important component of the YOLOv8 model responsible for predicting bounding boxes and classifying objects. This model uses several heads that operate at different scales, allowing the model to accurately detect objects of different sizes and shapes. Each head predicts bounding boxes and object classes for a specific image scale. The YOLOv8 uses three heads, each responsible for a different scale: 82 layers for detecting large objects, 94 layers for medium-sized objects, and 106 layers for small objects (Diwan *et al.*, 2023).

"Auxiliary Branches" in the YOLOv8 model is a component aimed at improving the accuracy of small object detection. By starting object detection at an early stage of processing, auxiliary branches allow the model to confidently recognise even the smallest details. This is important for tasks where small objects can be critical to the accuracy and completeness of the analysis. Another important technology in the development of lightweight and efficient deep learning models for image processing on mobile devices is MobileNetV3. This series of models is designed to optimise computing resources and power consumption, delivering high performance on devices with limited specifications. The MobileNetV3 is characterised by its high performance and lightweight design, making it ideal for a variety of applications on mobile devices such as smartphones and embedded systems (Fig. 4).

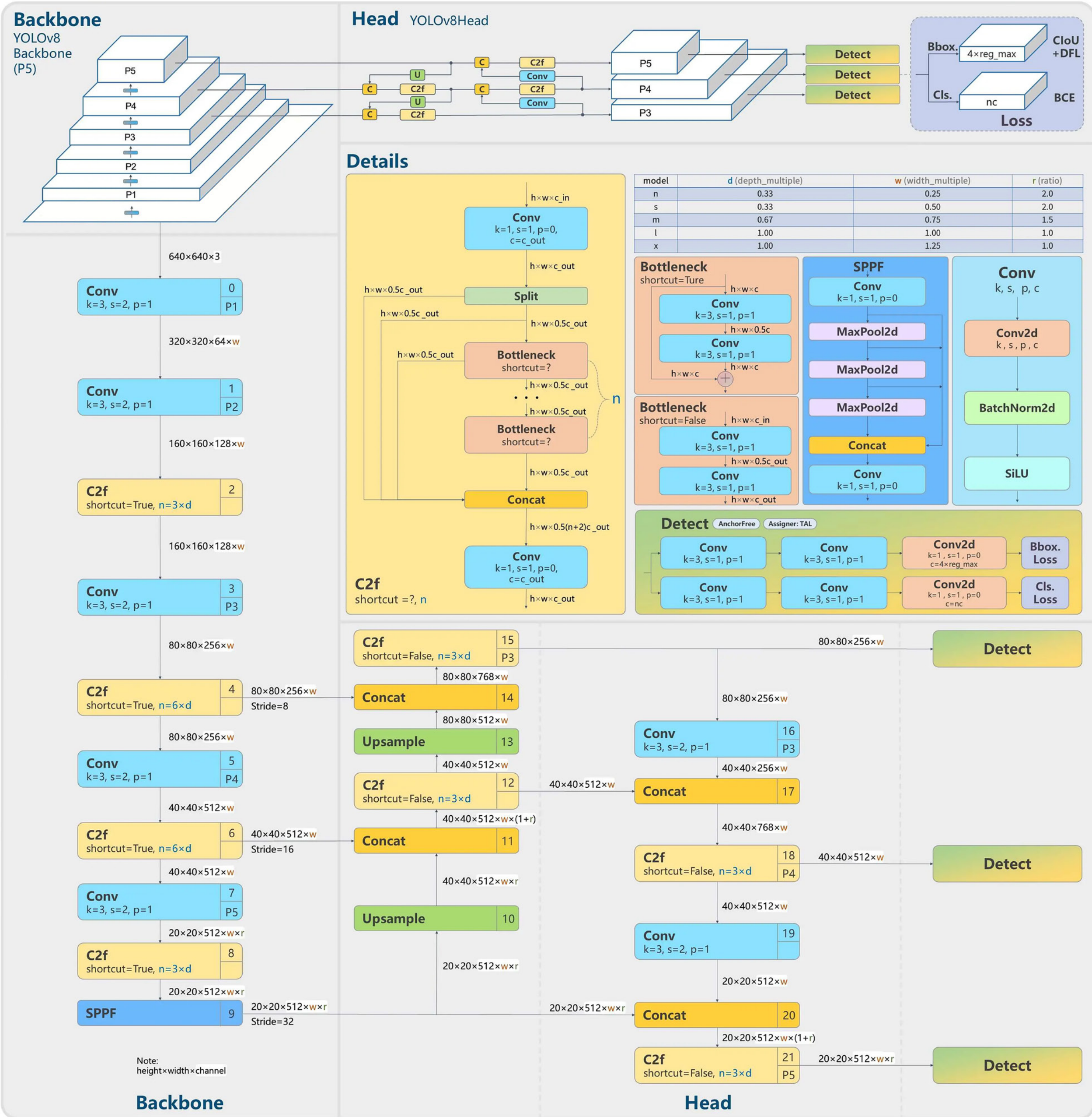

Figure 3. Structure of the YOLOv8 model

Source: PaddleYOLO (n.d.)

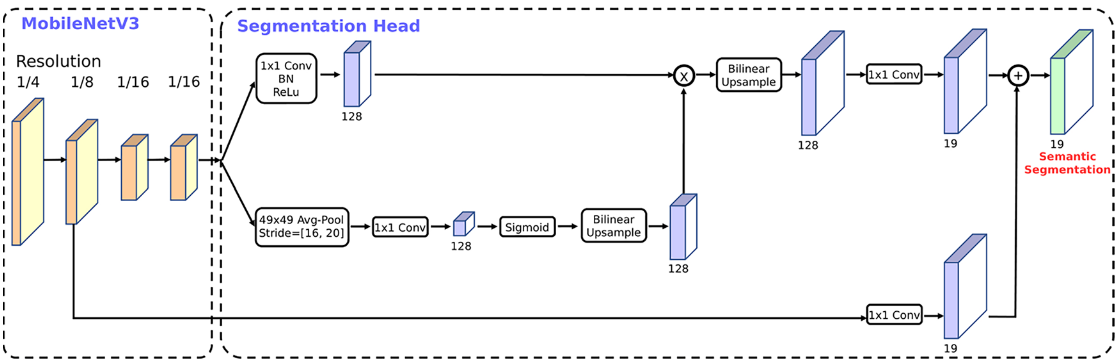

Figure 4. Structure of the MobileNetV3 model

Source: S.H. Tssng (2020)

Architectural innovations introduced in MobileNetV3 include the use of two key concepts: Squeeze-and-Excitation (SE) modules and structural adaptability modules, which allow for adaptive modification of the network architecture to meet the specifics of input data and tasks. SE modules are used to focus the network's attention on the most important features of the input data. This is achieved through two operations: "squeeze" and "excitation". The first operation performs group aggregation of features, which reduces their size and number. In the second operation, using a fully connected network, each group of features is assigned a weight that reflects its importance. Separate SE layers are used for each MobileNetV3 module, tailoring the network's attention to different contexts and details of the input images (Zhao & Wang, 2022).

The structural adaptability modules in MobileNetV3 can be used to modify to meet the requirements of a specific application. This is achieved by including different types of blocks depending on the characteristics of the input data and the expected output. For instance, for a classification task, an architecture with more connections and layers can be used, while for a segmentation task, an architecture with more layers and filters can be used. This provides greater flexibility and efficiency in a variety of use cases. In this study, a combination of the YOLOv8 and MobileNetV3 architectures was implemented and pre-trained on the ImageNet dataset. To implement this, the corresponding API of the keras_cv library was used, which provided convenient access to these models and their functionality. Due to the preliminary training of both models on a large amount of data, it was decided to retrain only some of the last layers of the YOLOv8 model (all layers after tf.concat_5).

During the retraining, the corresponding APIs of the keras_cv library were used. The retraining process involved changing the model's weights to meet our requirements and specifications, incorporating the output formats, input and output layer sizes, and optimisation parameters. This significantly reduced the time required to train the model while maintaining the high quality of the results. To improve the efficiency of the model in detecting UAVs, especially when they are small in the image, the image interlacing method can be used. This avoids missing objects on the borders of tiled images. For example, if the camera resolution is 1920×1080 pixels, and the size of the input image in the YOLOv8 model is 640×640 pixels. During operation, each image in the dataset can be cut into 8 tiles, with 2 tiles arranged vertically and 4 tiles arranged horizontally. The width and height of the tile in this case is 640 pixels, which corresponds to the size of the input image in the YOLOv8 model. The calculation of overlaps between distributed tiles is as follows:

$$l_o = \frac{n l_{tile} - l_{image}}{n-1}, \tag{1}$$

where l_o – the length of the overlap; n – the number of tiles; l_{tile} – the length of the tile; l_{image} – the length of the original image.

The formula (1) establishes that the overlap length between vertically adjacent tiles is 200 pixels, which is a fairly optimal solution since it is an integer. However, using the same formula, the overlap length between horizontally adjacent tiles is approximately 213.4 pixels, which creates problems in practical applications due to the difficulty of achieving such accuracy. Instead, the length of the three overlaps can be manually set to 200, 200, and 240 pixels, respectively. This will make the tile grid more accurate and simplify calculations, which is critical in applications where accuracy and simplicity are important factors. After splitting the image into tiles, the coordinates of the object labels must be updated to reflect the new positions. Information about the objects in the source images is stored in the YOLO format, where each image has a corresponding .txt file with annotations, including the object class, x-y coordinates, and width and height. The x, y, w, h were calculated using formulas (2-5), addressing the pixel coordinates:

$$x = \frac{x_{min} + x_{max}}{2 w_{image}}, \tag{2}$$

$$y = \frac{y_{min} + y_{max}}{2 h_{image}}, \tag{3}$$

$$w = \frac{x_{max} - x_{min}}{w_{image}}, \tag{4}$$

$$h = \frac{y_{max} - y_{min}}{h_{image}}. \tag{5}$$

Each object label of the source image contains the x and y coordinates of the upper left corner of the bounding box, as well as its width (w) and height (h). The x-coordinate defines the position relative to the left edge of the image, increasing towards the right. The y-coordinate defines the position relative to the top edge of the image, increasing in the downward direction. These values are expressed as a percentage of the total width (w_{image}) and height (h_{image}) of the image (Fig. 5).

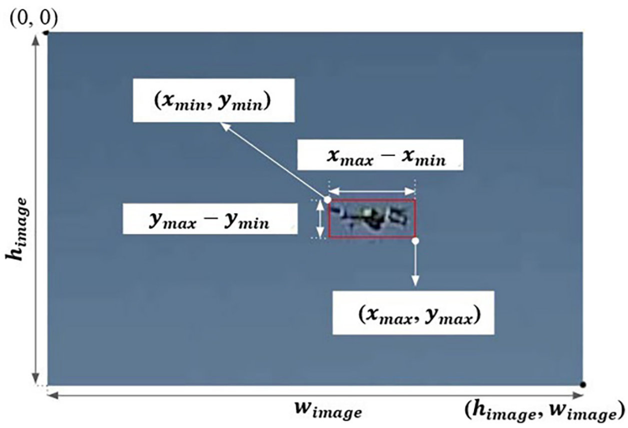

Figure 5. Calculation scheme in the YOLO model
Source: compiled by the authors

Emphasis was placed on calculating the model's performance in real-time. Since drone tracking requires the rapid processing of many images, several key performance metrics have been identified. To determine the effectiveness of the output level responsible for classification, binary cross-entropy was used, which is the optimal choice for binary classification tasks. This metric measures the distance between the predicted and actual values, addressing the probability of each class for each input sample, and is calculated using the formula (6):

$$L_{classification} = -\frac{1}{N}\sum_{i=1}^{N}(y_y * \log(p_i) + (1-y_i) * \log(1-p_i)), \quad (6)$$

where N – the number of samples in the training set; y_i – the actual class label for sample i; p_i – the predicted probability that sample i belongs to class 1.

The binary cross-entropy is used to assess how accurately the model classifies objects into two classes: present (in this case, a UAV) or absent. The value of this metric is minimised if the predicted and actual probabilities are the same, which means that the model correctly classifies objects. High values of binary cross-entropy indicate that the model cannot effectively separate classes, which leads to low classification accuracy. For the regression level, intersection over Union (IoU) was used (He et al., 2021; Yuan et al., 2020). This metric is calculated as the ratio of the intersection area to the area of the union of the two boundary regions: true boundary regions and predicted boundary regions (7):

$$IoU = \frac{\text{Cross-sectional area}}{\text{The area of the association}}. \quad (7)$$

The true boundaries are manually annotated based on the dataset, while the predicted boundaries are the output of the model that processes the input images (Fig. 6).

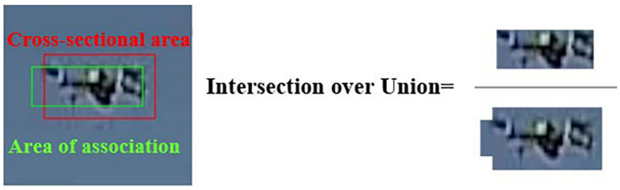

Figure 6. IoU calculation scheme

Source: compiled by the authors

This indicator determines how accurately the projected boundaries of an object coincide with the real ones. The mean average precision (MaP) was also performed, with results with an IoU value greater than 0.5 (50%) addressed. On the other hand, -ln(IoU) is used as a loss function for object detection tasks (Yu *et al.*, 2016). The final loss function was calculated as the sum of the loss functions for classification and regression (8):

$$L_{end} = L_{classification} - \ln(IoU). \quad (8)$$

While training the object classification and detection model in this study, four types of drones (Parrot A.R. Drone 2.0; DJI Inspire I; DJI Mavic 2 Pro; and DJI Phantom 4 Pro), on eight locations (street, trees, grass, mountain lake, palace, winter town, and temple), were used to train the classification part of the model. At the same time, the marked rectangle coordinates were used to train the regression part of the model, which predicts the exact location of the drones in the image. Furthermore, for training the models, the NVIDIA L4 GPU was used through Google Collab. One of the advantages of using Google Collab was the availability of a configurable development environment and pre-installed TensorFlow version 2.15. More detailed characteristics of the medium used are given in Table 1.

Table 1. Characteristics of the environment used in Google Collab

Characteristic	Value
Number of virtual CPUs	16
GPU	NVIDIA L4
RAM	64 Gb
GPU memory	24 GB GDDR6
SSD drive	375 GB

Source: compiled by the authors

The batch size was set to 160 when training the MobileNetV3_Large_ImageNet and MobileNetV3_Small_ImageNet models. To optimise the parameters, the Adam algorithm was used as one of the most efficient for training neural networks. The learning rate parameter of the Adam optimizator was set to 0.048 during MobileNetV3_Large_ImageNet model traning and 0.024 for MobileNetV3_Small_ImageNet. These parameters were carefully selected to ensure optimal learning computation speed. Using appropriate values of the package, optimiser and training speed helped to ensure the stability of the training process and improve the quality of the resulting models. The image size was set at 640×640 pixels, ensuring a balance between detection quality and processing speed. Other hyperparameters were left by default (Bhargavi, 2021; Rui *et al.*, 2021).

During the training of the models, their state was saved at each epoch for further analysis of their performance. The so-called COCO metrics were used to evaluate the model, in particular, the mean average precision for a threshold value of the interaction over union of 50% (MaP@[IoU=50%]) (Wood & Chollet, 2022). The comparison of the MaP@[IoU=50%] level was carried out at different points using the weights of the models saved at different stages of training. The graphs in Figure 7 show the loss functions as a function of the training epoch for both models under study.

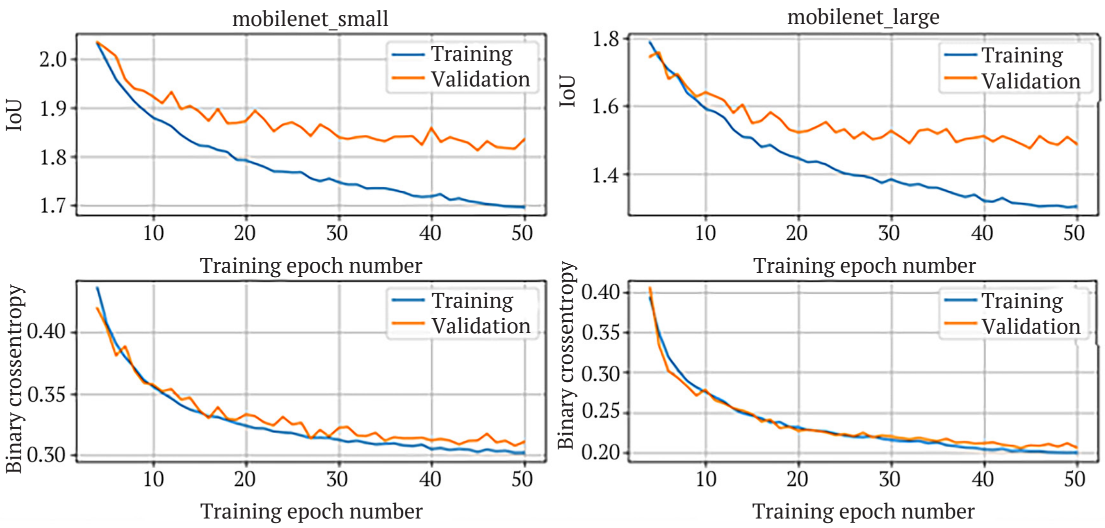

Figure 7. Loss functions for models with increasing number of epochs
Source: compiled by the authors

When determining the effectiveness of a model, it is necessary to address the difference between its training and validation performance. A graphical representation of this difference indicates that validation results may be shelved even though training performance may be improving. Figure 8, in turn, shows the dependence of the MaP@[IoU=50%] metric estimated on the validation dataset on the training epoch.

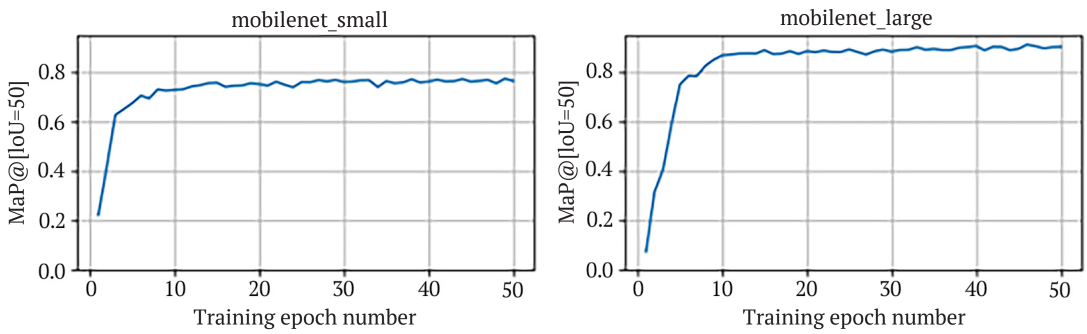

Figure 8. The dependence of MaP@[IoU=50%] estimated on the validation dataset on the number of training epochs
Source: compiled by the authors

As can be seen from the graphs in Figure 8, the accuracy of the MobileNetV3_Large model is higher. Noting these results, it should be noted that this may be the result of the greater complexity of the model itself. This can include more parameters or a more complex architecture, allowing it to better adapt to different types of data and conditions. Table 3 shows the values of the metrics evaluated on the test dataset for the models with the highest validation score of MaP@[IoU=50%]. The results of object classification on the test dataset are shown in Figure 9.

Table 3. Metrics for the models under study

Metric	MobileNetV3_Small	MobileNetV3_Large
Binary cross entropy	0.308	0.216
-ln(IoU)	1.818	1.529
MaP@[IoU=50%]	0.747	0.909

Source: compiled by the authors

Figure 9. Results of correct object classification using the trained model
Source: compiled by the authors

Several key technologies that have a significant impact on its functionality and efficiency should be considered when analysing the MobileNetV3_Large model in detail. One such technology is the mixed-depthwise convolution blocks technique, which reduces the number of computational operations and parameters while maintaining high accuracy. In addition, MobileNetV3_Large uses optimisation mechanisms such as linear bottlenecks and squeeze-and-excitation to reduce computational costs and improve model accuracy by focusing on important areas of the image. It's also worth noting the use of the Mish-activated function, which negates problems with vanishing and vanishing gradients.

Thus, the pre-training on ImageNet provided a basic understanding of the original features and common patterns in the images, but the combination of this pre-training with MobileNetV3 technologies allowed the model to show higher performance in object classification.

DISCUSSION

Scientific research in the development and application of neural networks for image and video processing is of great importance in the rapidly evolving and advancing world of technology. Such research can open broad prospects for solving various problems in the field of computer vision. Neural networks can make significant progress in object detection, image classification, and the analysis and processing of large amounts of video data. They are an important tool that can help increase automation and efficiency in many industries. As the market for unmanned technologies such as UAVs is rapidly developing, the use of neural networks can enable their detection in images and video with high accuracy and speed, which is becoming a key factor in ensuring safety and effective airspace management. Efficient algorithms and models developed based on neural networks can be a key element in security systems, as well as in commercial applications where unmanned vehicles can be used for surveillance, research and transport needs (Zarichuk, 2023).

It is important to note that the development and application of neural networks for detecting and classifying unmanned aerial vehicles requires an individual approach to each case. Each type of drone may have its unique characteristics, such as size, shape, speed and movement patterns, which require the use of appropriate technologies to detect and identify them. This individual approach will allow for the most efficient use of resources and ensure adaptation to various usage scenarios. Given the wide range of factors that influence the detection and classification of UAVs, it is important to pay sufficient attention to creating more accurate, fast and reliable systems to ensure effective airspace management.

The results of this study show the significant potential of using the YOLOv8 and MobileNetV3 algorithms in the detection of commercial unmanned aerial vehicles in various types of images. This was confirmed by the MaP@[IoU=50%] value of 0.909 for the MobileNetV3_Large model. This demonstrates the high efficiency and accuracy of this algorithm in recognising aircraft under different conditions and in different types of images. At the same time, for the MobileNetV3_Small model, this indicator reached 0.747, which also indicates the possibility of its use in this area, given its features. The achieved aircraft detection and classification rates are high and demonstrate great potential in the application of these models for solving airspace surveillance and security tasks. YOLOv8 along with MobileNetV3 can be used to build comprehensive detection systems that provide reliable and fast real-time object recognition.

S.V. Viraktamath *et al.* (2021) also analysed the impact of various parameters of neural networks on their accuracy and efficiency in object recognition, showing that the correct selection of parameters can significantly improve model performance. A.M. Ghoreyshi *et al.* (2020) improved algorithmic approaches to processing large amounts of data, which has increased the speed of video data processing and improved the accuracy of object detection. However, it is worth noting that this study has solved some problems that remained unaddressed in the above-mentioned works. In particular, the use of YOLOv8 and MobileNetV3 has made it possible to achieve significant improvements in the accuracy of object detection and display, which is a significant achievement. In addition, this study focused on the specific tasks of detecting unmanned aerial vehicles, which allowed for a deeper understanding of their specifics and needs.

When analysing the results, it is important to note the binary cross-entropy index, which for the MobileNetV3_Large model was 0.216. This indicator indicates low training losses and high efficiency of the model in solving classification tasks. This makes the MobileNetV3_Large model perfect for image processing and object detection tasks. The value of this indicator for the MobileNetV3_Small model, which was 0.308, is also an important result. This indicator indicates lower efficiency and accuracy of the MobileNetV3_Small model compared to MobileNetV3_Large in classification tasks. Thus, the analysis of this indicator can conclude the efficiency and potential of each model in the context of solving specific image processing tasks. O.S. Bubryak & K.R. Potapova (2023) also conducted similar analyses and studies in the field of image processing, identifying and confirming the high efficiency of using neural network algorithms in object detection and classification. At the same time, the study by S. Thakur *et al.* (2021) examined various aspects of using machine learning for image analysis and processing, which is also important for the development of this field. It should be noted that these works could complement the present study, contributing to a more complete understanding of the possibilities and limitations of using neural networks in the field of image processing and object classification.

The results also demonstrated the -ln(IoU) values for MobileNetV3_Small and MobileNetV3_Large models, which reflects their accuracy in detecting and localising objects in images. The -ln(IoU) values for these models were 1.818 and 1.529, respectively. This confirms the high accuracy and efficiency of both models in solving object detection tasks. A high IoU score (which is equal to low -ln(IoU) score) indicates that the models correctly determine the size and position of objects in the image, making them very useful for applications that require accurate object localisation. N.N. Hasibuan *et al.* (2021) determined that optimising some algorithms and parameters of the YOLOv8 and MobileNetV3 models can increase the speed of image and video processing, which is a significant achievement in this area. M.Ş. Gündüz & G. Işık (2023) also added to the knowledge of the model's response to different types of images, revealing support for a wide range of image formats and sizes without significant performance loss. At the same time, W. Liu *et al.* (2022) brought new information on the impact of weather conditions and other factors on the performance of models of this series in different operating conditions, which expands the understanding of its operating limits and features. All the above research contributes to a deeper and more complete understanding of the capabilities and limitations of the YOLOv8 and MobileNetV3 models in real-world applications, ensuring their high suitability for a wide range of uses in a variety of environments.

Thus, such research using machine learning and computer vision algorithms is of great importance in the overall technological landscape. They can be used to develop and design more accurate and efficient computer vision systems that can be used for object detection, pattern recognition, motion tracking, and other tasks. The rapid development of unmanned aerial vehicles, video surveillance and monitoring systems requires continuous improvement in image and video processing techniques to ensure the highest level of safety and performance. Research in computer vision and machine learning addresses these challenges by developing new algorithms and models that can work in real-time and reliably detect objects even in difficult conditions, improving safety, efficiency and process automation in various fields of activity.

CONCLUSIONS

The study results confirmed that the combination of YOLOv8 and MobileNetV3 algorithms has significant potential in detecting commercial unmanned aerial vehicles in images when working on a test dataset. Preliminary training of these algorithms on the ImageNet dataset, followed by refinement and training on the SimUAV dataset containing 29,568 images, achieving high accuracy in recognising and classifying UAVs of various types and sizes. The results of the study demonstrated that the MobileNetV3_Large model has better accuracy than the MobileNetV3_Small. After 50 iterations, the models achieved MaP@[IoU=50%] of 0.909 and 0.747, respectively, indicating the high efficiency and accuracy of this algorithm in detecting objects on the test data.

The study also showed that the value of the binary cross-entropy for the MobileNetV3_Large model is 0.216, confirming the low level of training losses and high efficiency of the model in solving classification tasks. Additionally, the study notes that the IoU for this model is 1.529, indicating its high accuracy in detecting and localising objects in images. Thus, the results obtained indicate the superiority of the MobileNetV3_Large model and confirm its potential in solving computer vision problems in the example of commercial UAV detection. It is worth noting that despite the high performance of the research results, it is important to consider the limited availability of the data used, which may limit the universality of the model. In addition, pre-training a model on publicly available datasets such as ImageNet can lead to the transfer of the limitations of these datasets to the resulting model, which can affect its versatility and accuracy in specific applications. All of this should be addressed in further research and consideration of applications of the model.

One of the areas for such research could be to expand the dataset for model training, which would enhance the model's ability to detect and classify various objects. It is also worth considering improving the preview and filtering algorithms that help avoid false positives and improve model accuracy. Such research will further improve such models and facilitate their application in real-world conditions.

ACKNOWLEDGEMENTS

None.

CONFLICT OF INTEREST

None.

REFERENCES

[1] Alam, K.M.R., Siddique, N., & Adeli, H. (2020). A dynamic ensemble learning algorithm for neural networks. *Neural Computing and Applications*, 32(12), 8675-8690. doi: 10.1007/s00521-019-04359-7.
[2] Bejani, M.M., & Ghatee, M. (2020). Theory of adaptive SVD regularization for deep neural networks. *Neural Networks*, 128, 33-46. doi: 10.1016/j.neunet.2020.04.021.
[3] Bhargavi, K. (2021). Deep learning architectures and tools: A comprehensive survey. In *Deep learning applications and intelligent decision making in engineering* (pp. 55-75). Hershey: IGI Global. doi: 10.4018/978-1-7998-2108-3.ch002.
[4] Bubryak, O.S., & Potapova, K.R. (2023). System of transportation vehicles detection on the basis of linear method SVM and neural network YOLO. In *Proceedings of IX International Scientific and Practical Conference "Science and Innovation of Modern World"* (pp. 206-210). London: Cognum Publishing House.
[5] Chen, L., Li, S., Bai, Q., Yang, J., Jiang, S., & Miao, Y. (2021). Review of image classification algorithms based on convolutional neural networks. *Remote Sensing*, 13(22), article number 4712. doi: 10.3390/rs13224712.
[6] Diwan, T., Anirudh, G., & Tembhurne, J.V. (2023). Object detection using YOLO: Challenges, architectural successors, datasets and applications. *Multimedia Tools and Applications*, 82(6), 9243-9275. doi: 10.1007/s11042-022-13644-y.
[7] Ghoreyshi, A.M., AkhavanPour, A., & Bossaghzadeh, A. (2020). Simultaneous vehicle detection and classification model based on deep YOLO networks. In *2020 International Conference on Machine Vision and Image Processing (MVIP)* (pp. 1-6). Piscataway: Institute of Electrical and Electronics Engineers. doi: 10.1109/MVIP49855.2020.9116922.
[8] Gündüz, M.Ş., & Işık, G. (2023). A new YOLO-based method for real-time crowd detection from video and performance analysis of YOLO models. *Journal of Real-Time Image Processing*, 20(1), article number 5. doi: 10.1007/s11554-023-01276-w.
[9] Hammami, M., Friboulet, D., & Kechichian, R. (2020). Cycle GAN-based data augmentation for multi-organ detection in CT images via Yolo. In *2020 IEEE International Conference on Image Processing (ICIP)* (pp. 390-393). Abu Dhabi: Institute of Electrical and Electronics Engineers. doi: 10.1109/ICIP40778.2020.9191127.
[10] Hasibuan, N.N., Zarlis, M., & Efendi, S. (2021). Detection and tracking different type of cars with YOLO model combination and deep sort algorithm based on computer vision of traffic controlling. *Synchronous: Journal and Research of Informatics Engineering*, 6(1), 210-221.
[11] He, J., Erfani, S., Ma, X., Bailey, J., Chi, Y., & Hua, X.S. (2021). Alpha-IoU: A family of power intersection over union losses for bounding box regression. *Advances in Neural Information Processing Systems*, 34, 20230-20242.
[12] ImageNet Large Scale Visual Recognition Challenge (ILSVRC). (n.d.). Retrieved from https://image-net.org/challenges/LSVRC/index.php.
[13] Janulin, M., Vrublevskyi, O., & Prokhorenko, A. (2022). Energy minimization in city electric vehicle using optimized multi-speed transmission. *International Journal of Automotive and Mechanical Engineering*, 19(2), 9721-9733. doi: 10.15282/ijame.19.2.2022.08.0750.
[14] Jiang, P., Ergu, D., Liu, F., Cai, Y., & Ma, B. (2022). A review of Yolo algorithm developments. *Procedia Computer Science*, 199, 1066-1073. doi: 10.1016/j.procs.2022.01.135.
[15] Kushnir, N.O., Loktikova, T.M., Morozov, A.V., & Yurchenko, V.O. (2022). The use of convolutional neural networks in image object recognition and classification tasks. *Technical Engineering*, 1(89), 93-100. doi: 10.26642/ten-2022-1(89)-93-100.
[16] Lin, C.J., Lin, C.H., & Wang, S.H. (2021). Integrated image sensor and light convolutional neural network for image classification. *Mathematical Problems in Engineering*, 2021, article number 5573031. doi: 10.1155/2021/5573031.
[17] Liu, W., Ren, G., Yu, R., Guo, S., Zhu, J., & Zhang, L. (2022). Image-adaptive YOLO for object detection in adverse weather conditions. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(2), 1792-1800. doi: 10.1609/aaai.v36i2.20072.
[18] Najjar, L., & Baskaya, F.A.T. (2022). Drone-based SVP utilization for revitalizing under-valued public open spaces: Case of Istanbul. *DISEGNARECON*, 15(29). doi: 10.20365/disegnarecon.29.2022.17.
[19] Nguyen, H.L., & Huynh, K.T. (2023). New model for low-end computers: ResNet and VGG-16. *Science and Technology Development Journal*, 26(1), 2672-2680. doi: 10.32508/stdj.v26i1.4030.
[20] PaddleYOLO. (n.d.). Retrieved from https://github.com/RangeKing/PaddleYOLO.
[21] Pidpalyi, O. (2024). Future prospects: AI and machine learning in cloud-based SIP trunking. *Bulletin of Cherkasy State Technological University*, 29(1), 24-35. doi: 10.62660/bcstu/1.2024.24.
[22] Rahmad, C., Asmara, R.A., Putra, D.R.H., Dharma, I., Darmono, H., & Muhiqqin, I. (2020). Comparison of Viola-Jones Haar Cascade classifier and histogram of oriented gradients (HOG) for face detection. *IOP Conference Series: Materials Science and Engineering*, 732, article number 012038. doi: 10.1088/1757-899X/732/1/012038.

[23] Raja, R., Kumar, S., & Mahmood, M.R. (2020). Color object detection based image retrieval using ROI segmentation with multi-feature method. *Wireless Personal Communications*, 112(1), 169-192. doi: 10.1007/s11277-019-07021-6.

[24] Rui, C., Youwei, G., Huafei, Z., & Hongyu, J. (2021). A comprehensive approach for UAV small object detection with simulation-based transfer learning and adaptive fusion. *ARXIV.* doi: 10.48550/arXiv.2109.01800.

[25] Shuai, Q., & Wu, X. (2020). Object detection system based on SSD algorithm. In *2020 International Conference on Culture-oriented Science & Technology (ICCST)* (pp. 141-144). Beijing: Institute of Electrical and Electronics Engineers. doi: 10.1109/ICCST50977.2020.00033.

[26] SimUAV. (n.d.). Retrieved from https://github.com/mlcaepiee/SimUAV/tree/main?tab=readme-ov-file.

[27] Song, T.-G., Pirahandeh, M., Ahn, C.-J., & Kim, D.-H. (2018). GPU-accelerated high-performance encoding and decoding of hierarchical RAID in virtual machines. *Journal of Supercomputing*, 74(11), 5865-5888. doi: 10.1007/s11227-017-1969-y.

[28] Taherkhani, A., Cosma, G., & McGinnity, T.M. (2020). AdaBoost-CNN: An adaptive boosting algorithm for convolutional neural networks to classify multi-class imbalanced datasets using transfer learning. *Neurocomputing*, 404, 351-366. doi: 10.1016/j.neucom.2020.03.064.

[29] Thakur, R.S., Chatterjee, S., Yadav, R.N., & Gupta, L. (2021). Image de-noising with machine learning: A review. *IEEE Access*, 9, 93338-93363. doi: 10.1109/ACCESS.2021.3092425.

[30] The YOLOV8Detector model. (n.d.). Retrieved from https://keras.io/api/keras_cv/models/tasks/yolo_v8_detector/.

[31] Tishchenko, S., & Kuznetsov, E. (2024). Neural networks for the task of image classification. *Science and Technology Today*, 3(31), 705-718. doi: 10.52058/2786-6025-2024-3(31)-705-718.

[32] Tsang, S.H. (2020). *MobileNetV3: Searching for MobileNetV3 (Image Classification)*. Retrieved from https://sh-tsang.medium.com/paper-mobilenetv3-searching-for-mobilenetv3-image-classification-5072d4d8703c.

[33] Viraktamath, S.V., Yavagal, M., & Byahatti, R. (2021). Object detection and classification using YOLOv3. *International Journal of Engineering Research & Technology*, 10(2), 197-202. Retrieved from https://www.ijert.org/research/object-detection-and-classification-using-yolov3-IJERTV10IS020078.pdf.

[34] Wang, Y., Li, Y., Song, Y., & Rong, X. (2020). The influence of the activation function in a convolution neural network model of facial expression recognition. *Applied Sciences*, 10(5), article number 1897. doi: 10.3390/app10051897.

[35] Wood, L., & Chollet, F. (2022). Efficient graph-friendly COCO metric computation for train-time model evaluation. *ARXIV.* doi: 10.48550/arXiv.2207.12120.

[36] Yu, J., Kiang, Y., Wang, Z., Cao, Z., & Hiang, T. (2016). UnitBox: An advanced object detection network. In *MM '16: Proceedings of the 24th ACM International Conference on Multimedia* (pp. 516-520). New York: Association for Computing Machinery. doi: 10.1145/2964284.2967274.

[37] Yuan, X., Xia, J., Wu, J., Shi, J., & Deng, L. (2020). Low altitude small UAV detection based on YOLO model. In *2020 39th Chinese Control Conference (CCC)* (pp. 7362-7366). Shenyang: Institute of Electrical and Electronics Engineers. doi: 10.23919/CCC50068.2020.9188588.

[38] Zarichuk, O. (2023). Hybrid approaches to machine learning in software development: Applying artificial intelligence to automate and improve processes. *Development Management*, 21(4), 53-60. doi: 10.57111/devt/4.2023.53.

[39] Zhao, L., & Wang, L. (2022). A new lightweight network based on MobileNetV3. *KSII Transactions on Internet and Information Systems (TIIS)*, 16(1), 1-15. doi: 10.3837/tiis.2022.01.001.

[40] Zou, S., Zhang, B., & Zhang, B. (2023). Research on improved image classification algorithm based on Darknet53 Model. In *Proceedings of the 11th International Conference "Communications, Signal Processing, and Systems"* (pp. 73-80). Singapore: Springer. doi: 10.1007/978-981-99-2362-5_10.

Руслан Єрмоленко

Кандидат фізико-математичних наук, доцент
Київський національний університет імені Тараса Шевченка
01033, вул. Володимирська, 60, м. Київ, Україна
https://orcid.org/0009-0000-0854-4668

Денис Клекоць

Студент
Київський національний університет імені Тараса Шевченка
01033, вул. Володимирська, 60, м. Київ, Україна
https://orcid.org/0000-0002-4251-2958

Ольга Гогота

Кандидат фізико-математичних наук, науковий співробітник
Київський національний університет імені Тараса Шевченка
01033, вул. Володимирська, 60, м. Київ, Україна
https://orcid.org/0000-0003-4108-7256

Розробка алгоритму виявлення комерційних безпілотних літальних апаратів за допомогою методів машинного навчання

Анотація. Дане дослідження було спрямоване на тренування алгоритмів виявлення комерційних безпілотних літальних апаратів з використанням методів машинного навчання. Під час дослідження для виявлення безпілотних літальних апаратів на зображеннях та відео були використані архітектури нейронних мереж YOLOv8 та MobileNetV3. Використовувані моделі були попередньо натреновані на датасеті ImageNet, а потім доопрацьовані на датасеті SimUAV, що містив зображення безпілотних апаратів чотирьох типів (Parrot A.R. Drone 2.0; DJI Inspire I; DJI Mavic 2 Pro; and Dji Phantom 4 Pro), різних розмірів та у восьми різних фонових локаціях. Результати даного дослідження підвердили, що поєднання архітектур YOLOv8 та MobileNetV3 має значний потенціал у виявленні комерційних безпілотних літальних апаратів на різних видах зображень. Навчені моделі продемонстрували високі показники в розпізнаванні та класифікації безпілотних літальних апаратів, досягаючи середньої точності виявлення (при пороговому значення IoU рівному 50%) на рівні 0,747 та 0,909, для моделей MobileNetV3_Small та MobileNetV3_Large, відповідно. Це свідчить про високу ефективність та точність моделей у виявленні об'єктів на тестових даних. Також результати дослідження включали в себе значення метрики бінарної перехресної ентропії, які склали відповідно 0,308 та 0,216, вказуючи на високу точність моделей у класифікації об'єктів, та підтверджуючи високу ефективність та надійність цих моделей в роботі з об'єктами на тестових даних. В ході дослідження модель MobileNetV3_Large показала більш точні результати порівняно з MobileNetV3_Small, що свідчить про її вищу ефективність у виявленні та класифікації літальних апаратів. Отримані результати підтверджують перспективність застосування методів машинного навчання в області систем моніторингу та безпеки, що дозволить надійно виявляти та відстежувати безпілотні літальні апарати в різних умовах. Високі показники натренерованих моделей свідчать про їх ефективність у реальних умовах експлуатації, що робить їх цінним інструментом для вирішення важливих завдань контролю та нагляду

Ключові слова: згорткова нейронна мережа; класифікація об'єктів; YOLOv8; MobileNetV3; комп'ютерний зір; бінарна перехресна ентропія