International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Development of Target Detection and Identification System for Combat Military Vehicles Via Artificial Intelligence and Decision Support Matrix

Reşat Ali Tütüncüoğlu[1], Selda Güney[2]

[1]Department of Defense Technologies and Systems Institute of Science and Engineering - Başkent University Ankara, TÜRKİYE
[2]Department of Electrical-Electronics Engineering Institute of Science and Engineering - Başkent University Ankara, TÜRKİYE

Abstract
Modern military intelligence systems are improving day by day. In the context of ground forces, combat vehicle-type classification technology serves as a valuable supporting asset. Understanding the capabilities and limitations of different combat vehicle types is crucial for developing effective tactics and strategies. The implementation of artificial intelligence-supported military vehicle type classification technology has raised increasing concerns as image processing, pattern recognition, and deep learning have advanced. You Only Look Once (YOLO) has demonstrated numerous considerable advantages in object detection and image classification. This method accelerates target detection because it can predict objects in real-time. The high accuracy of detection and assessment assists operational personnel in the field. The YOLO prediction method produces accurate results with minimal background errors and facilitates the understanding of generalized object representations. This paper utilizes YOLO to demonstrate combat vehicle type detection, specifically employing the YOLOv8m model, which is well-suited for mobile deployments. This study focuses on a process that begins with detecting targets using images obtained through electro-optical systems, aimed at supporting the activities of identifying and defining targets, which are essential components of target management systems. The process involves developing a Target Detection and Identification (TDI) system based on deep learning, incorporating steps such as pre-processing, segmentation, feature extraction, classification, additional data extraction, and decision support. This system is designed to interpret the results obtained from these processes and provide actionable recommendations. Furthermore, it assists in addressing the weapon target assignment problem in the land environment.

Keywords: Image processing, combat military vehicles, classification, assessment, artificial intelligence (AI), decision support.

1 INTRODUCTION

Combat vehicles are military vehicles specifically designed for use in armed conflicts. They come in various shapes and sizes and are categorized based on their primary functions and characteristics.

Understanding the capabilities and limitations of combat vehicles is a cornerstone of military strategy and tactics. This knowledge allows commanders to leverage the strengths of each type while mitigating their weaknesses, thereby maximizing the chances of success in combat.

Specifically, the classification of opposing combat vehicles, also known as the identification and understanding of adversary vehicles on the battlefield, is crucial for several reasons. These include target prioritization, tactical decision-making, firepower planning, force protection, intelligence gathering, reconnaissance, and situational awareness.

AI-driven enemy combat vehicle classification enhances military capabilities by providing rapid and accurate intelligence, improving decision-making, and reducing risks to human operators. It acts as a force multiplier that can significantly impact the outcomes of military operations on the modern battlefield.

2 RELATED WORKS

Object classification is a large field of research in computer vision and machine learning that aims to classify objects present in images into meaningful categories [1]. The convolutional neural networks (CNN) can be used at the core of everything from photo annotation to popular self-driving studies. they are working intensively behind the scenes in everything from agriculture to military security [2]. There is an increasing demand of object detection and automatic vehicle type recognition systems in the last 10 years as a result of the development of the cheaper sensor and camera technologies [3]. Because there is a huge amount of visual data that cannot be evaluated by human resource by self and this data has to be classified very fast and correlated to evaluation.

A disparity exists between object detection algorithms and classification algorithms in the field of computer vision. Object detection algorithms are designed to identify and delineate specific objects within an image by creating bounding boxes around the objects of interest. In this context, these algorithms are tasked with accurately positioning the objects within the image frame. Notably, object detection algorithms may generate multiple bounding boxes, each representing a distinct object within the image. It is crucial to highlight that the exact number of objects and, consequently, bounding boxes are unknown in advance, adding complexity to the detection process.

To solve this kind of problem would be to take different regions of interest from the image, and use a Convolutional Neural Network (CNN) to classify the presence of the object within that region. The problem with this approach is that the objects of interest might have different spatial locations within the image and different aspect ratios [4].

Several algorithm techniques have been suggested and investigated by different researchers to detect and classify vehicle types. There are one-stage and two-stage target detection algorithms. Region-based Convolutional Neural Networks (R-CNN), Fast R-CNN and Faster R-CNN algorithms are two-stage target detection algorithms [5]. YOLO is the one-stage algorithm and directly calculates the classification and regression from different positions [6].

In recent years, the processing ability of computing devices and the number of existing labelled image data have increased the popularity and success of CNN architectures. The CNN architectures have been widely used in visual object classification tasks.

R-CNN; to bypass the problem of selecting a huge number of regions, Ross Girshick [7] et al. proposed a method where we use selective search to extract just 2000 regions from the image and he called them region proposals. Therefore, now, instead of trying to classify a huge number of regions, you can just work with 2000 regions [8]. Problems with R-CNN; It still takes a very big amount of time to train the network. It

cannot be implemented in real-time. The selective search algorithm is a fixed algorithm. Therefore, no learning is happening at that stage. This could lead to the generation of bad candidate region proposals [4].
Fast R-CNN; some of the problems that are faced at R-CNN is solved by feeding the input image to the CNN to generate a convolutional feature map. The reason why "Fast R-CNN" is faster than R-CNN is that CNN doesn't need to be fed with 2000 region proposals each time. Instead of previous methodology, the convolution process is performed only once per image, and a feature map is created from this one.
Faster R-CNN; both of the above algorithms use selective search to find out the region proposals. Selective search is a slow and time-consuming process affecting the performance of the network. Therefore, Shaoqing Ren et al [9] came up with an object detection algorithm that eliminates the selective search algorithm and lets the network learn the region proposals.
Except for the You Only Look Once (YOLO) algorithm, all previously described object detection algorithms use regions to localize the object in the image. The network does not look at the complete image. Instead, parts of the image which have high probabilities of containing the object. In YOLO, a single convolutional network predicts the bounding boxes and the class probabilities for these boxes [10]. YOLO has the natural advantage of speed, better intersection over union in bounding boxes, and improved prediction accuracy compared to real-time object detectors.
YOLO is orders of magnitude faster than other object detection algorithms. The limitation of YOLO algorithm is that it struggles with small objects within the image [11]. This is due to the spatial constraints of the algorithm.
A Station-Temporal Fusion Network with two parallel feature extraction branches for detecting moving small targets has a successful precision result [12].
YOLOv5 [13] lightweight-improved object detection deep learning model integrated with the dehazing module (image restoration) is devised to detect the ship in the foggy image. Experimental results demonstrate the effectiveness of this approach in enabling efficient and accurate ship detection under foggy conditions [14].
YOLOv8, offers a diverse range of models, each specialized for specific tasks in computer vision. These models are designed to cater to various requirements, from object detection to more complex tasks like instance segmentation, pose/keypoints detection, oriented object detection, and classification [15]. The experimental results show that YOLOv8 optimized model improves small object detection with higher accuracy [16].
An image recognition method based on the combination of EfficientDet and Generative Adversarial Network (GAN), in which the gauged image features extracted from the EfficientDet model are used as the input of the GAN for the game learning of image categories and features [17].
Researcher G. Joccher has conducted a study using the YOLOv5 algorithm to estimate the distance of certain objects based on their box ratios. This study aims to accurately estimate the distances of objects in the image based on their size and ratios using the object detection capabilities of YOLOv5 [18].
Real time object distance and measurement using deep learning Canny Edge Algorithm and OpenCV has been achieved for small objects and small distance (1m≤ X)[19].
M. Lan et al. developed the YOLOv8 network model for distance measurement operations and parsed the optimized data, and incorporated it into the binocular distance system, proposing a fusion sensory distance model. In addition, the model provides high portability on executable embedded development boards by edge detection [20].

In our study, we extract the features of objects using the YOLO algorithm. By employing the transfer learning method, the newly obtained features are utilized as input for distance calculations. The trigonometric calculation method we employed enables us to achieve measurements that are both very simple and rapid. The need for this type of technology within this particular segment is to identify opposing force vehicles without revealing our presence. Moreover, a thorough review of the literature reveals that there are no studies of this kind specifically focused on the military tactical domain.

3 Framework Design and Methodology

This section outlines the methodology employed for the targeted system. It carefully examines the exploration and comprehension of the architecture while seeking to determine the optimal approach for achieving the desired output.

Initially, detailed information about the targets is obtained using a single-stage detector algorithm. Subsequently, the target's distance is calculated using an algorithm that employs the features extracted through the transfer learning method and the prepared data tables. Following the estimation of the target distance, this information is interpreted in conjunction with the generated decision support matrices, which provide recommendations to the user. This approach allows the model to demonstrate a high level of generalization while requiring less data.

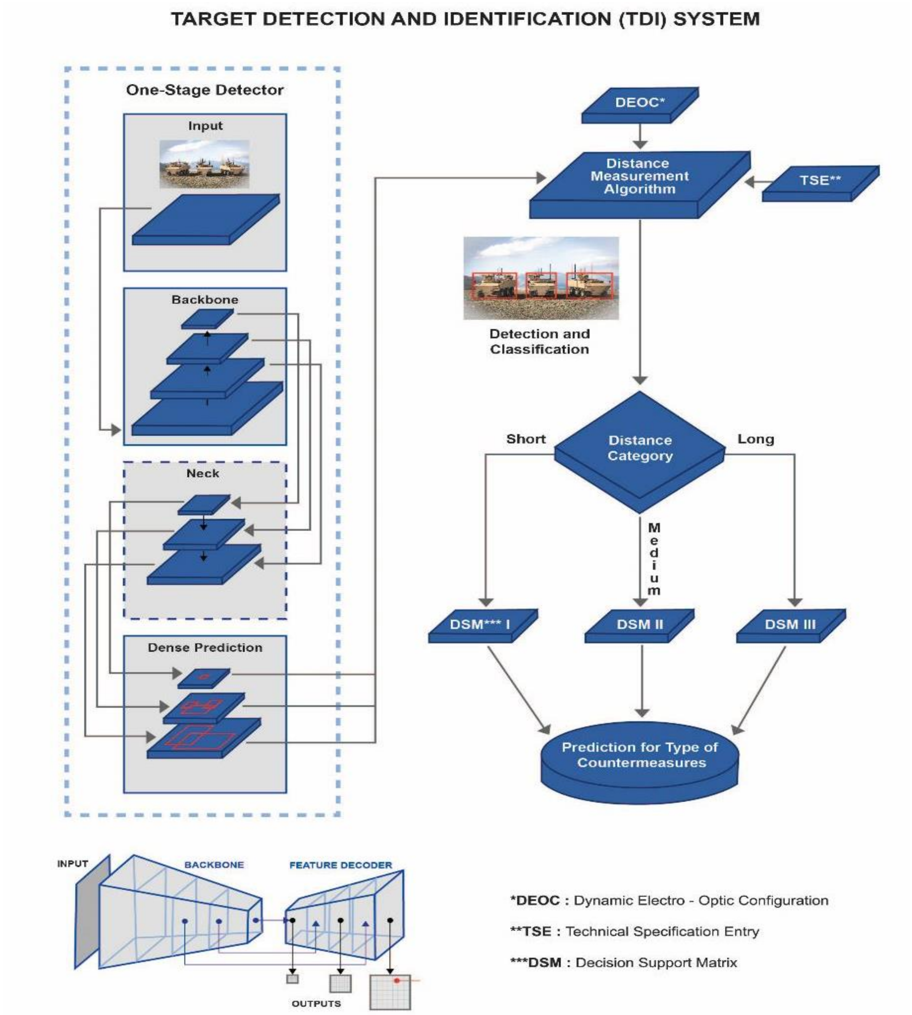


Figure 1Main Algorithm and Steps

3.1 Dataset

For the initial experimentations part, the dataset was prepared by Roboflow Universe [21]. In the early stage of experimentation, the initial dataset consists of originally 730 images. To increase the number of data, "data augmentation" techniques are used (flip and rotate). By applying the rotation (between -20° and +20°) and bounding box rotation (between -15° and +15°), the new data set consists of 1730 images.

- After reviewing the results, the new data is added, then to increase the number of data, again we used "data augmentation" techniques (rotate, shear, grayscale).

Figure 2 Augmentation Techniques (Rotate, Shear, and Grayscale)

- By applying the rotation, bounding box rotation, shear, and grayscale, the new augmented data set consists of 5984 images (see data split below).

Table 1. Data Split

Train Set	Valid Set	Test Set
%75	%15	%10
4489	897	598

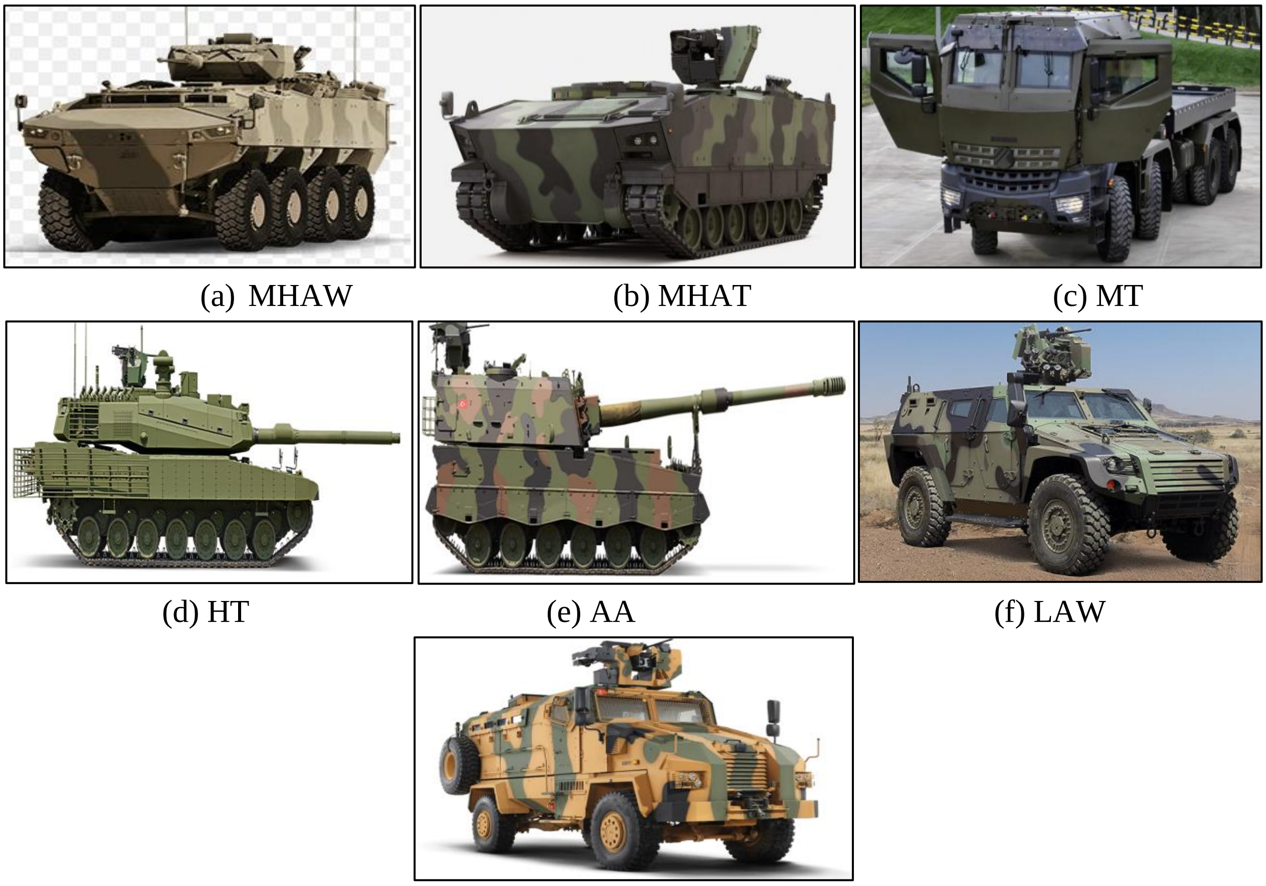
- Average annotations per image is 1,2 per image, average image sizes are 0,41mp, median image ratios are 640x640).

3.2 Operational Dataset Types

It's important to note that the specific classification of combat vehicles can vary by country and military organization, and new technologies may lead to the development of new categories or modifications to existing ones. Additionally, some vehicles may serve multiple roles, depending on their configuration and the mission at hand.

The main indicators we take into account when choosing types of data sets are as follows. First of all, land vehicles are divided into wheeled and tracked then military equipment is classified as heavy, medium and light [22]. Afterwards, it is stated that type and number information is the most important information in surveillance reports [23&24]. So that combat vehicles that were considered to be most effective against friendly troops when encountered on the battlefield were selected as a type classification.

This data set consist of 7 (seven) types that are mostly used in the armies. These are Medium and Heavy Armoured Wheeled (MHAW) Carriers, Medium and Heavy Armoured Tracked (MHAT) Carriers, Military Trucks (MT), Heavy (Main Battle) Tanks (HT), Light Armoured Wheeled (LAW) Carriers, Armoured Artillery (AA), and Mine Resistant Ambush Protected Vehicles (MRAP).



(a) MHAW (b) MHAT (c) MT
(d) HT (e) AA (f) LAW
(g) MRAP

Figure 3 Sample Images from the Dataset (a-g)

The selected images are labelled by subject matter experts to make supervised training. Also, the test data composed free from the trained datasets.

3.3 Main Algorithm

YOLOv8, specifically, is an improvement and refinement of the YOLO object detection algorithm [15]. YOLOv8 is a computer vision model architecture developed by Ultralytics, the creators of YOLOv5.
A comparative analysis of the widely used v5 and v7 is carried out by Olorunshola et al [25] shows significant contribution compared to other works earlier mentioned in the literature review. The experiment demonstrates the effectiveness of the v5 model compared with v7.
Also, the results of the YOLOv8 to v5, v10 and v11 algorithms with the same dataset were obtain to check the performance of the model. For precision and inference speed, YOLOv8 outperforms the rest of the algorithms' results.
YOLOv8 is the one of the latest iterations in the YOLO series of real-time object detectors, offering cutting-edge performance in terms of accuracy and speed. Building upon the advancements of previous YOLO versions, YOLOv8 introduces new features and optimizations that make it an ideal choice for various object detection tasks in a wide range of applications.

- Advanced Backbone and Neck Architectures: YOLOv8 employs state-of-the-art backbone and neck architectures, resulting in improved feature extraction and object detection performance.

- Anchor-free Split Ultralytics Head: YOLOv8 adopts an anchor-free split Ultralytics head, which contributes to better accuracy and a more efficient detection process compared to anchor-based approaches.
- Optimized Accuracy-Speed Trade off: With a focus on maintaining an optimal balance between accuracy and speed, YOLOv8 is suitable for real-time object detection tasks in diverse application areas.
- Variety of Pre-trained Models: YOLOv8 offers a range of pre-trained models to cater to various tasks and performance requirements, making it easier to find the right model for your specific use case [15].

3.4 Model Structure

The YOLOv8 architecture can be broadly divided into three main components [15]:

- Backbone: This is the convolutional neural network (CNN) responsible for extracting features from the input image. YOLOv8 uses a custom CSPDarknet53 backbone, which employs cross-stage partial connections to improve information flow between layers and boost accuracy.
- Neck: The neck, also known as the feature extractor, merges feature maps from different stages of the backbone to capture information at various scales. YOLOv8 Architecture utilizes a novel C2f module instead of the traditional Feature Pyramid Network (FPN). This module combines high-level semantic features with low-level spatial information, leading to improved detection accuracy, especially for small objects.
- Head: The head is responsible for making predictions. YOLOv8 employs multiple detection modules that predict bounding boxes, objectness scores, and class probabilities for each grid cell in the feature map. These predictions are then aggregated to obtain the final detections.

3.5 Aditional Feature Extraction by Using Distance Measurement Algorithm

A picture can be used to localize the target in an image and associate it with the distance. A total of four numbers are obtained in the Bounding Box. The Width and Height variables are used in the formula for measuring the distance of object and describing the details of the detected object/objects. Width and Height will vary depending on the distance of the object to the camera.

The values detected by the YOLOv8 algorithm and marked with a bounding box are used to estimate the distance based on the type of target detected, utilizing the calculation method described in Equation 1. To determine the distance, revised formulas have been adapted to the algorithm (see distance measuring formulas in Equations 2 and 3). The focal length (f) depends on the specific type of electro-optic device used.

$$\frac{f}{d} = \frac{r}{R} \quad (1)$$

$$f = d \times \frac{r}{R} \text{ pixels} \quad (2)$$

$$d = f \times \frac{R}{r} \text{ cm} \quad (3)$$

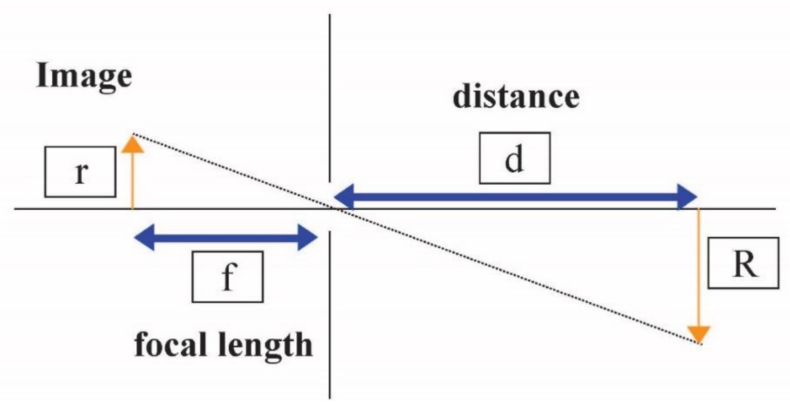


Figure 4 Trigonometric Logic of Distance Measuring

To achieve this, the width and height matrices (R) of the most commonly used vehicle types were created. When vehicle lengths are used, the false prediction rate due to the vehicle's stance position tends to increase. However, height generally remains consistent; therefore, average height values were adopted for different vehicle types. Additionally, when these values are adjusted according to the specific regions in which they will be applied, the estimation error rate decreases. The image height (r) of the identified vehicle is calculated by the YOLO algorithm.

3.6 Prediction by Using Decision Support Matrix

After the combat vehicles are detected and identified, estimating their distance will force the decision maker to decide what precautions to take or what type of weapon system to use to neutralize the target.

The decision support matrix, which is created by examining the Land Combat Vehicles and the measures to be taken against these vehicles, makes recommendations to the user according to the information obtained from system.

Table 2 DSM I Recommendation

Type of Combat Vehicles	Minimum Ammunition /Weapon System Type
Short Distance (50-700m)	**Recommendation**
Light Armoured Wheeled (LAW) Carriers	Medium Armour Piercing Ammunition, RPG
Medium and Heavy Armoured Wheeled (MHAW) Carriers	A/T Ammunition with Short Distance Weapon System
Medium and Heavy Armoured Tracked (MHAT) Carriers	A/T Ammunition with Short Distance Weapon System
Military Trucks (MT)	A/T Ammunition with Short Distance Weapon System
Heavy (Main Battle) Tanks (HT)	RPG
Armoured Artillery (AA)	RPG
Mine Resistant Ambush Protected Vehicles (MRAP)	RPG

4 Model Training and Evaluation

4.1 Training Results

In this section, we present the results of combat vehicle image classification in the test datasets.

The YOLOv8m model was used for the training and test purpose, and a YAML file was defined to configure the paths for the dataset and the number of classes to train. The model was trained for 241 epochs with a batch size of 32, and “The Roboflow” Platform was used to visualize and track data in real time.

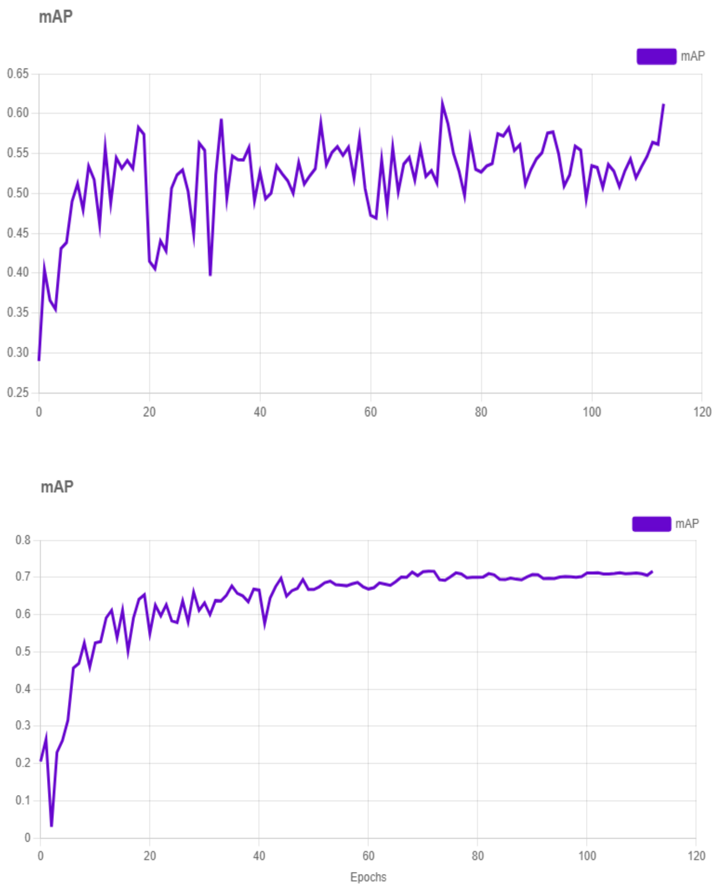


Figure 5 Initial Data set and Second Data set mAP results (a) ve (b)

When the results are compared, the dataset preparation is one of the most important part for the results. As the number of training epochs increases, the loss is decreasing. In the context of bounding box-class and object regression, lower loss values indicate that the predicted bounding boxes are getting closer to the ground truth.

A reduction in validation loss means that our model is generalizing well. It's not just fitting the training data but is also making better predictions on data it hasn't seen before. This suggests that the model is learning meaningful patterns in the data.

For precision and inference speed, Table 2 also, provides the comparison of YOLOv5, v10 and v11 results with YOLOv8. It can be seen that YOLOv8 outperforms the rest of the algorithms’ results.

Table 3 YOLO v8 versus YOLOv5, v10 and v11 at our Dataset

Class	Images	Mean Average Precision			
		YOLO 5m	YOLO 8m	YOLO 10m	YOLO 11m
All	**5984**	62,3%	83,6%	68,7%	70,5%

International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Speed Of Inference	79 FPS	117 FPS	80 FPS	100 FPS

When the confusion matrix is examined, it is observed that the lowest results are in LAW and MT type classifications. It is considered that the similarity of LAW and MT type vehicles to civilian vehicles in some cases increases the error rates.

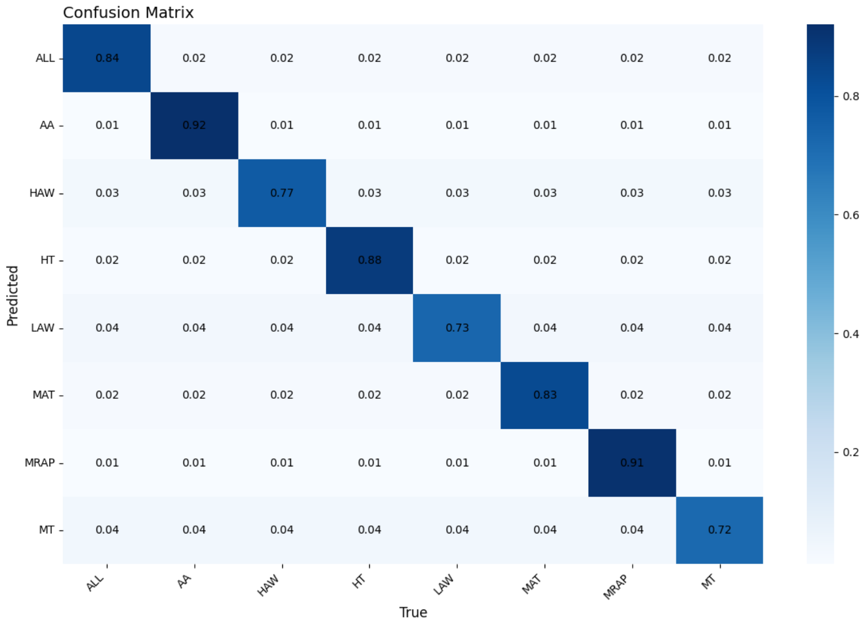


Figure 6 Confusion Matrix

The detection and classification of a single observed combat vehicle is carried out very quickly by the YOLOv8 algorithm. Following the main classification, the estimation of the distance using additional data obtained from the relevant tables, and the use of the appropriate decision support matrices, with the presentation of the results are calculated in milliseconds (See below first example for single target).
In cases where there are multiple targets, the required computation times are negligibly low (See below second example for multi target).

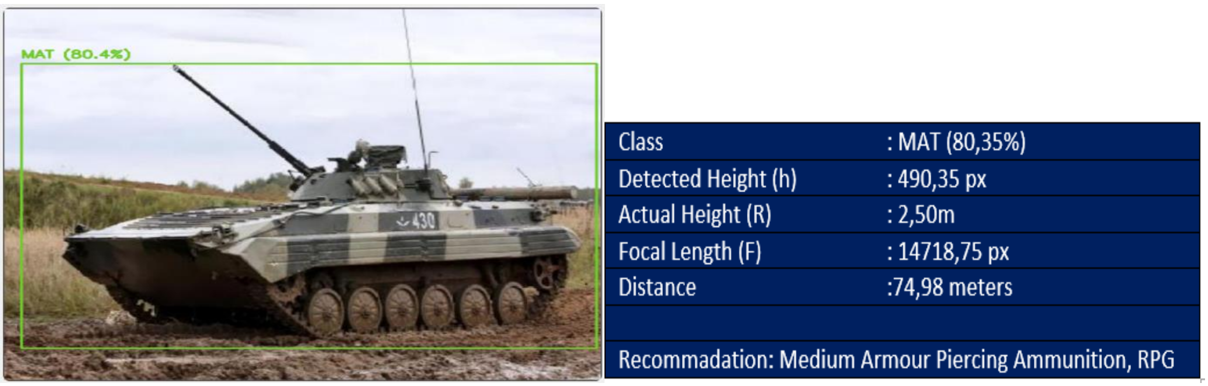


Figure 7 Single Target Detection & Identification with Recommendation

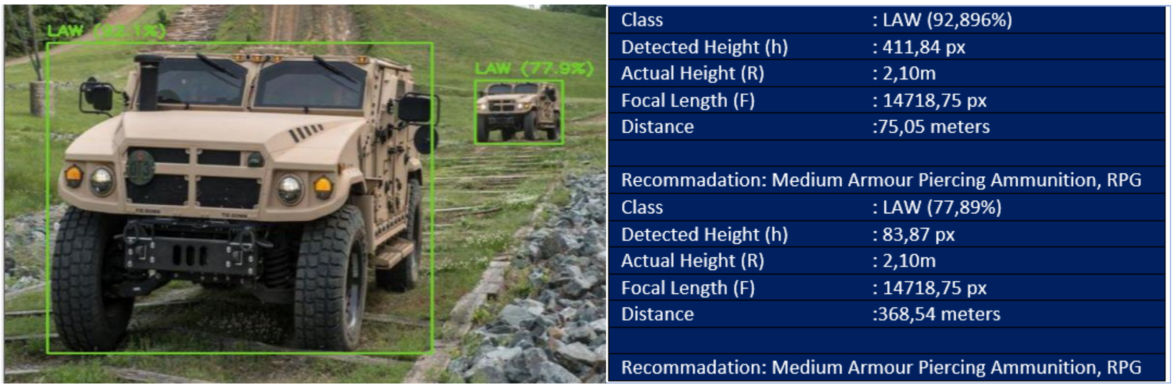


Figure 8 Multi Target Detection & Identification with Recommendation

4.2 Operational Test Results & Discussion

The trained model (algorithm, measuring distance and decision support matrix) is uploaded to the military type computer (Panasonic Toughbook CF33). Then, the success of the TDI System using the both tablet's camera and also vehicle's camera are tested on the field.

The type of the vehicles was MHAW type vehicles. The success of detection is 100% and 70% classification precision, the false precision is 20%. The false classifications are 1 (one) LAW and 1 (one) MRAP.

Also, printed pictures of vehicles for every class are prepared (ten pictures for every class) and tested by tablet's camera. While the success achieved within the scope of detection was close to the training results by -5%, it was observed that the classification success was approximately 13% lower than the training test results.

Table 4 Operational Test Results

Class/T&F	LAW	MHAW	MHAT	MT	HT	AA	MRAP
True	6	6	7	6	8	8	8
False	4	4	3	4	2	2	2

4.3 Operational Understanding of Training Results

Figure 13 MT, MHAW and MRAP Examples

MT, MHAW and MRAP type vehicle identification confusion results are investigated. It is observed that their sizes, heights from ground, and number of the wheels are different. Remember that the selected pictures are very distinctive examples.

MHAT and HT type of vehicle identification confusion results are investigated. It is observed that their bases are almost same. Specifically, the towers are resembling each other. Only, the diameter of the tank barrels is bigger than the diameter of MHAT barrels.

Figure 14 MHAT and HT Example

5 Conclusions

Due to the complexity and variability of the actual combat environment and the demand for real-time target recognition, this paper proposes an image recognition method based on the combination extracting additional features by adding calculation method.

It is extremely important to determine the distance of an object passively using an existing electro-optic device. Thanks to the applied method, there is no significant increase in detection and identification time. In addition, it is possible to determine the measures that can be taken against the enemy without being detected by the enemy.

In this study, additional information (distance estimation) was obtained by using the detection and classification results of military combat vehicles. It is aimed to provide support to the user in a critical time through the decision support matrix, where these results are evaluated as a whole by supporting them with additional data.

This paper proposes a model for categorizing vehicle types utilizing the most recent state-of the-art object detection model, YOLOv8. With an overall mAP 83,6%, the model's promising results. The model was very successful in recognizing the vehicles, as shown by the results. Any future identification system that has to accurately identify the type of vehicle can easily incorporate it. While the classes can be further broken down into a more detailed manner, dividing classes of vehicles as unarmoured trucks, air defence vehicles, wheeled artillery, etc.

It is evaluated that including distance measurements in future studies to determine the extent of separation in the structural differences of combat vehicles will increase the classification accuracy.

References

1. R. Szeliski, “Computer Vision: Algorithms and Applications,” 1st ed. New York, NY, USA: Springer-Verlag New York, Inc., 2010.
2. P. Sharma, S. Gupta, S. Vyas, M. Shabaz, “Object Detection and Recognition Using Deep Learning-based Techniques,” IET Commun. 00, 1– 11 (2022). https://doi.org/10.1049/cmu2.12513.
3. M.A. Berwo, A. Khan, Y. Fang, H. Fahim, S. Javaid, J. Mahmood, Z.U. Abideen, “Deep Learning Techniques for Vehicle Detection and Classification from Images/Videos: A Survey Sensors,” 2023, 23, 4832. https://doi.org/10.3390/s23104832.
4. R. Gandhi, “R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms” https://medium.com/towards-data-science/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-

algorithms-36d53571365e.
5. J. Hu, “Research on Fast Tracking Algorithm of Moving Target Based on Optical Flow Method,” Xi’An, City: Xidian University; 2014. [Google Scholar].
6. X. Liu, “End-to-end Target Detection and Attribute Analysis Algorithm Based on Deep Learning and its Application,” [D] Guangzhou: South China Universi-ty of Technology; 2017. [Google Scholar].
7. R. Girshick, “Fast R-CNN,” In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448. [Google Scholar].
8. R. Girshick, J. Donahue, T. Darrell, J. Malik, “Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation,” In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587. [Google Scholar].
9. Ren, Shaoqing et al. “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” IEEE Transactions on Pattern Analysis and Machine Intelligence 39 (2015): 1137-1149.
10. J. Redmon et al. “You Only Look Once: Unified, Real-Time Object Detection,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015): 779-788.
11. J. Redmon, J. and A. Angelova, “Real-time Grasp Detection Using Convolutional Neural Networks,” 2015 IEEE International Conference on Robotics and Automation (ICRA) (2014): 1316-1322.
12. S. Zhu, L.Ji, J.Zhu, S.Chen, H.Ren, “Spatio-temporal fusion with motion masks for the moving small target detection from remote-sensing videos” Engineering Applications of Artificial Intelligence, Volume 138, Part A, 2024, 109362, ISSN 0952-1976, https://doi.org/10.1016/j.engappai.2024.109362.
13. J.R Glenn et al. “ultralytics/yolov5: Initial Release.” (2020).
14. T.Liu, Z.Zhang, Z.Lei, Y.Huo, S.Wang, J.Zhao, J.Zhang, X.Jin, X.Zhang, “An approach to ship target detection based on combined optimization model of dehazing and detection”, Engineering Applications of Artificial Intelligence, Volume 127, Part B, 2024, 107332, ISSN 0952-1976, https://doi.org/10.1016/j.engappai.2023.107332.
15. G. Jocher, A. Chaurasia and J. Qiu, “Ultralytics YOLOv8”, V= {8.0.0}, 2023, https://github.com/ultralytics/ultralytics.
16. H. Huadong, W.Binyu, X.Jiannan, Z.Tianyu. “Improved small-object detection using YOLOv8: A comparative study” Applied and Computational Engineering. 41. 80-88, 2024 doi:10.54254/2755-2721/41/20230714.
17. X.Zhuang, D. Li, Y.Wang, K.Li, “Military target detection method based on EfficientDet and Generative Adversarial Network”, Engineering Applications of Artificial Intelligence, Volume 132, 2024, 107896, ISSN 0952-1976, https://doi.org/10.1016/j.engappai.2024.107896.
18. G.Jocher, “YOLOv5 by Ultralytics (Version 7.0)” [Computer software] 2020 https://doi.org/10.5281/zenodo.3908559
19. B. M. U, H. Raghuram and Mohana, “Real Time Object Distance and Dimension Measurement using Deep Learning and OpenCV” 2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS), Coimbatore, India, 2023, pp. 929-932, Doi: 10.1109/ICAIS56108.2023.10073888.
20. M. Lan, J. Wang and L. Zhu, "Perception and Range Measurement of Sweeping Machinery Based on Enhanced YOLOv8 and Binocular Vision," in IEEE Access, vol. 11, pp. 126398-126408, 2023, doi: 10.1109/ACCESS.2023.3331013.
21. Roboflow Annotation Tool, https://roboflow.com/annotate.
22. APP-6 (D) NATO Joint Military Symbology, Chapter 3 Land Symbols October 2017.
23. ATP 2-01.3 Intelligence Preparation of the Battlefield, Chapter 5, Step3 Evaluate the Treat,

"https://armypubs.army.mil" March 2019];

24. ATP 3-21.10 Infanty Rifle Company, Appendix F-Armored, Stryker, and Mounted Employment "(https://armypubs.army.mil)" May 2018.
25. O. Oluwaseyi, M. Irhebhude, A. Evwiekpaefe, "A Comparative Study of YOLOv5 and YOLOv7 Object Detection Algorithms," Journal of Computing and Social Informatics. 2. 1-12. 10.33736/jcsi.5070.2023.