Detection of Military Aircraft Using YOLO and Transformer-Based Object Detection Models in Complex Environments

Araştırma Makalesi/Research Article

Fatih ŞENGÜL[1], Kemal ADEM[2]

[1]Department of Computer Engineering, Sivas University of Science and Technology, Sivas, Türkiye.
[2]Department of Computer Engineering, Sivas Cumhuriyet University, Sivas, Türkiye.

210102002@sivas.edu.tr, kemaladem@cumhuriyet.edu.tr
(Geliş/Received:12.09.2024; Kabul/Accepted:08.01.2025)
DOI: 10.17671/gazibtd.1549034

Abstract— Computer vision and deep learning techniques are widely applied in object detection tasks across various domains, including defense technologies. Accurate and efficient detection of military aircraft plays a critical role in strengthening air defense systems and enabling effective strategic decision-making. This study evaluates the performance of YOLOv7, YOLOv8, and RT-DETR models in detecting military aircraft using a dataset consisting of 19.514 images spanning 43 aircraft models. The dataset incorporates images captured from various angles and diverse backgrounds, such as urban, rural, and coastal areas, ensuring realistic testing conditions. However, class imbalance is observed, with certain aircraft models, such as the F14 and F16, being more represented than others, which may affect model generalization. To address these challenges, hyperparameters were optimized, and performance metrics, including mean Average Precision (mAP) and recall, were analyzed. Experimental results show that YOLOv8 achieved 94% mAP and 88.1% recall, YOLOv7 reached 90.2% mAP and 82.7% recall, while RT-DETR demonstrated consistent performance with 92.7% mAP and 90.4% recall. These findings highlight the strengths and limitations of the evaluated models and provide inferences for improving detection systems in defense applications.

Keywords— military aircraft detection, YOLOv7, YOLOv8, RT-DETR

Karmaşık Ortamlarda YOLO ve Transformer Tabanlı Nesne Tespit Modelleri ile Askeri Uçak Tespiti

Özet— Bilgisayarla görme ve derin öğrenme teknikleri, savunma teknolojileri de dahil olmak üzere çeşitli alanlardaki nesne algılama görevlerinde yaygın olarak uygulanmaktadır. Savaş uçaklarının doğru ve verimli bir şekilde tespit edilmesi, hava savunma sistemlerinin güçlendirilmesinde ve etkili stratejik karar alma süreçlerinin desteklenmesinde kritik bir rol oynamaktadır. Bu çalışmada, 43 uçak modelini kapsayan 19.514 görüntüden oluşan bir veri kümesi kullanılarak YOLOv7, YOLOv8 ve RT-DETR modellerinin savaş uçaklarını tespit etme performansı değerlendirilmektedir. Veri kümesi, çeşitli açılardan ve kentsel, kırsal ve kıyı alanları gibi farklı arka planlardan çekilen görüntüleri içermekte ve gerçekçi test koşulları sağlamaktadır. Bununla birlikte, F14 ve F16 gibi belirli uçak modellerinin diğerlerine göre daha fazla temsil edildiği ve model genellemesini etkileyebilecek sınıf dengesizliği gözlemlenmiştir. Bu zorlukların üstesinden gelmek için hiperparametreler optimize edilmiş ve ortalama Ortalama Hassasiyet (mAP) ve geri çağırma dahil olmak üzere performans ölçütleri analiz edilmiştir. Deneysel sonuçlar, YOLOv8'in %94 mAP ve %88,1 geri çağırma, YOLOv7'nin %90,2 mAP ve %82,7 geri çağırma değerlerine ulaştığını, RT-DETR'nin ise %92,7 mAP ve %90,4 geri çağırma ile tutarlı bir performans sergilediğini göstermektedir. Bu bulgular, değerlendirilen modellerin güçlü yönlerini ve kısıtlamalarını vurgulamakta ve savunma uygulamalarında tespit sistemlerinin iyileştirilmesi için çıkarımlar sağlamaktadır.

Anahtar Kelimeler— savaş uçağı tespiti, YOLOv7, YOLOv8, RT-DETR

1. INTRODUCTION

Computer vision, a key branch of computer science, has rapidly advanced over time. Researchers have continually worked to develop more effective and efficient systems to tackle the challenges in this field. Morphological methods have emerged as significant strategies, particularly for addressing core issues in computer vision. The expansion of digital platforms has led to a substantial increase in visual data, driving demand for data processing and information extraction [1]. This surge in data has accelerated research in computer vision, with the ultimate goal of developing algorithms that operate as swiftly and accurately as the human eye [2]. The growing need for automation has also played a critical role in these advancements, enhancing efficiency and safety in high-risk environments through autonomous systems [3][4][5]. Research in computer vision typically falls into three main areas: segmentation, classification, and object detection [6]. These areas encompass more specific tasks, such as semantic segmentation, scene classification, and pixel-based classification. Semantic segmentation, for instance, distinguishes object boundaries within the same category, while pixel-based classification is particularly effective for hyperspectral remote sensing images, though it requires significant processing power [7][8][9]. Object detection, which involves identifying, classifying, and locating objects within an image, is particularly challenging in applications requiring detailed accuracy [10]. Object detection is critical in fields ranging from military operations to healthcare diagnostics [11][12][13][14]. However, the development of effective algorithms often encounters challenges such as low spatial resolution and complex image data. Additionally, reliance on human interpretation can introduce potential errors. Detected objects in images typically include a variety of structures, both man-made and natural, making object detection a complex task. Significant advancements in object detection have been driven by deep learning techniques, which have improved detection under large datasets and complex conditions [15]. The increasing computational power of GPUs has also been crucial in advancing these technologies, representing a critical step towards overcoming the challenges in object detection. As imaging technologies have advanced, the detection of military aircraft has become increasingly critical. Numerous studies have contributed to developing methods for accurately identifying military aircraft.

Early efforts focused on creating models for fighter jet detection using physical prototypes of aircraft such as the P51 Mustang, G1-Fokker, MiG25-F, and Mirage 2000, achieving a recognition accuracy of 91% and a response time of 3 seconds [16]. Building on this foundation, subsequent research introduced novel approaches, such as a 3D model for carrier-based aircraft detection, achieving a detection accuracy of 99.92% in real reconnaissance images [17]. Advancements in remote sensing also enabled methods that used Convolutional Neural Networks (CNNs) for aircraft classification, achieving an accuracy of 98.29% [18]. Further developments included an enhanced YOLOv3-based object detection system, which improved precision to 91.49%, surpassing the original YOLOv3's 85.61% [19]. Real-time fighter jet detection was achieved using the YOLOv4 algorithm, with mAP and fps improvements to 86.92% and 29.62, respectively [20]. Object detection techniques continued to evolve with the development of SCMask R-CNN, which combined object recognition and segmentation, resulting in an AP value of 96.8% [21]. To address the challenge of detecting small aircraft, a Multi-Scale Detection Network (MSDN) was proposed, achieving an F1-score above 96% and an AP value exceeding 90% [22].

Further advances included the DAFF-Net model for detecting fighter jets within remote sensing images, which achieved an mAP value of 83.83% [23]. This progress continued with the development of YOLOv5-Aircraft, integrating enhancements that led to a 3.74% increase in mAP and a 6.93% improvement in speed [24]. The comparative analysis of deep learning-based models for aircraft detection provided valuable insights, with one study demonstrating that the FNDCNNTL model achieved nearly 100% accuracy [25].

Additionally, the application of R-FCN using Google Earth images reached a detection accuracy of 98.01%, outperforming SSD and Faster R-CNN models [26]. Advances in detection methods continued with the development of TransEffiDet, an aircraft detection method based on EfficientDet and Transformer modules, achieving an mAP value of 86.6% [27]. In military vehicle detection, Tiny YOLOv3 and Quantized SSD Mobilenet v2 showed superior performance in edge devices [28]. Further refinement in military aircraft recognition was achieved through the integration of VACR techniques with Back Propagation Neural Networks (BPNN), leading to a training accuracy of 95.33% and a testing accuracy of 87% [29]. Additionally, the lightweight CNN framework CGC-NET demonstrated its effectiveness in remote sensing images, achieving a 91.06% F-score and outperforming other models [30].

Continued innovation was evident in the development of the YOLOv5-Aircraft model, which achieved a 3.74% increase in mAP and a 6.93% speed improvement over previous versions [31]. The optimization of YOLOv5 led to the YOLM model, which reached an mAP score of 88.7% on the FAIR1M dataset, outperforming other base models [32]. Comparative analyses highlighted the effectiveness of Faster R-CNN, which achieved the highest mAP value of 97%, making it suitable for high-precision scenarios [33]. Similarly, the scaled YOLOv4 model achieved 96% accuracy in practical applications using high-resolution Worldview-3 data [34]. The YOLO-extract algorithm, optimized from YOLOv5,

further enhanced detection capabilities, achieving a 95.9% mAP value [35]. The CNTR-YOLO algorithm improved detection accuracy by 3.3% over YOLOv5, reaching a 70.1% average accuracy on the MAR20 dataset [36]. Finally, the GCD-DETR model for UAV detection marked a significant advancement, achieving high accuracy rates of 95.6% and 97.8% on UAV datasets [37]. The enhancement of YOLOv8 and Faster R-CNN further demonstrated the continuous improvement in object detection, with YOLOv8 achieving a general accuracy of 96.7% mAP, surpassing Faster R-CNN in overall performance [38].

In recent years, various studies have focused on the detection of small aerial objects. One such study modified the YOLOv8 model by integrating Multi-Scale Image Fusion (MSIF) and a P2 layer, achieving an Average Precision (AP) of 0.189 in the Drone-vs-Bird Detection Challenge, demonstrating effectiveness in detecting small and fast-moving objects at 45.7 FPS for 640x640 resolutions [39]. Another study introduced a YOLO-based segmented dataset containing 20,925 images, including 12,474 drones and 8,451 birds, to address the challenge of distinguishing drones from birds. The dataset features detailed segmentation and diverse environmental conditions, providing a valuable resource for training deep learning models in UAV detection and classification tasks [40]. Another study utilized the YOLOv4 model to develop a drone detection system, achieving 85% accuracy by classifying military drones under the 'aeroplane' category in the COCO dataset [41]. Another study evaluated three deep learning approaches in the Drone vs. Bird Detection Challenge, with the best model achieving an average precision of 80%, demonstrating robustness against small object sizes, distant targets, and moving cameras [42].

Despite significant advancements in military aircraft detection, challenges remain, particularly in the reliance on satellite images with limited perspectives and the need for extensive computational resources. This study addresses these gaps by evaluating the performance of state-of-the-art object detection models, including YOLOv7, YOLOv8, and RT-DETR, across diverse and complex scenarios. By analyzing the effects of different hyperparameters, this research provides insights into the strengths and limitations of these models, contributing to their potential application in defense technology.

2. MATERİAL AND METHODS

This section examines the dataset utilized in this manuscript, explaining the theoretical foundations relevant to the topics covered. The methods applied in this study and the experiments conducted are also detailed.

2.1. Dataset

The dataset utilized in this study is centered on the detection of military aircraft and consists of 43 classes of visual data, available as an open-source resource on Kaggle. It includes 19.514 images, covering a broad spectrum of military aircraft types and models. Each class represents a specific type or model of military aircraft, with images taken from various angles and set against diverse backgrounds, enhancing the model's adaptability to real-world conditions. The images capture jets under different seasonal and temporal conditions, with varying weather and background settings, promoting the development of more robust algorithms. This variety ensures that the model is effective across multiple scenarios, from snowy landscapes to tropical islands, thereby broadening its applicability. However, the dataset is not evenly distributed across all classes, as illustrated in Figure 1. Classes such as the F14 and F16 are more heavily represented, with over 1,000 images each, while others, like the F35 and Rafale, have fewer images. This imbalance could cause some classes to be more easily recognized, while others might be underrepresented, posing challenges in developing a balanced detection model.

This dataset is a valuable resource for research on the automatic detection of military aircraft, commonly used in academic studies to evaluate algorithm performance on real-world data. Experiments conducted with images from various backgrounds and angles improve the algorithms' adaptability and effectiveness, which is critical in military and defense applications. Accurate detection of military aircraft can enhance air defense systems, monitor enemy aircraft, and ensure civilian air traffic safety.

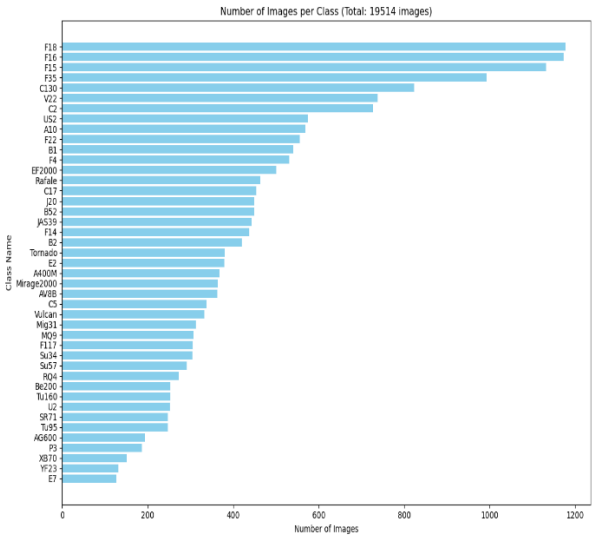

Figure 1. Dataset Distribution

Object detection has become a pivotal research area due to its increasing importance across various fields, aiming to automatically identify and localize specific objects within visual data. This technology underpins critical applications ranging from autonomous vehicles and security surveillance to medical diagnostics and retail analytics. Object detection involves classifying objects within an image and localizing them with bounding boxes, thereby providing information about the object's identity and location [43].

The implementation of object detection is primarily based on deep learning models, which learn features from large volumes of labeled image data. Early models utilized techniques like sliding windows followed by feature extraction, but recent deep learning approaches offer more direct and efficient detection. CNN-based models, for example, provide faster and more accurate results compared to traditional machine learning algorithms [44]. In this study, advanced object detection methods, particularly RT-DETR and various versions of the YOLO algorithm, are employed for their effectiveness in overcoming challenges in object detection and contributing to advancements in the field.

2.2.1. Convolution-Based Object Detection Models

Convolution-based object detection is a foundational technique in computer vision systems, enabling the detection, identification, and classification of objects within images. This process is built upon deep learning methodologies, particularly Convolutional Neural Networks (CNNs), which have revolutionized the field of image processing. Object detection algorithms are designed to locate and categorize objects within images by leveraging the layered structure of CNNs and their capacity for learning complex visual features [45]. The initial stage of convolution-based object detection involves feature extraction within convolutional layers. These layers apply filters and activation functions to progressively abstract visual features from raw pixel values, such as edges, textures, and shapes [46]. Typically, activation functions like ReLU enhance the model's learning capability, allowing for more complex function modeling [47]. Once features are extracted, models like the Region Proposal Network (RPN) identify potential object regions, predicting bounding boxes that define the approximate location and size of objects within the image [48].

Subsequently, the extracted regions are resized and classified using Region of Interest (RoI) Pooling, transforming them into fixed-size vectors for further processing. The final stages involve classification and regression layers, which assign class labels and refine the placement of bounding boxes using techniques such as softmax classification and linear regression. This allows the model to accurately predict both the class and location of each object in the image. The success of these techniques depends significantly on well-constructed training datasets and the use of data augmentation and regularization methods to prevent overfitting. These practices ensure the model's robustness and its ability to make accurate predictions under varying conditions. Convolution-based object detection is widely used across industries like automotive, healthcare, security, and retail, enabling automated and precise task execution without human intervention. This continuous evolution of AI and computer vision technologies leads to increasingly innovative applications across these sectors.

2.2.2. YOLO (You Only Look Once)

YOLO (You Only Look Once) is a pioneering deep learning architecture developed for real-time object detection, introduced by Joseph Redmon and colleagues in 2016 [49]. Unlike traditional methods, YOLO performs detection in a single pass through the network, dividing the image into grids and predicting bounding boxes and class probabilities simultaneously. This approach allows YOLO to achieve both speed and accuracy, making it ideal for real-time applications. YOLO's architecture is based on convolutional neural networks (CNNs), comprising multiple convolutional layers, pooling layers, and fully connected layers. These layers are designed to extract features, learn relationships, and make predictions necessary for object detection. By processing the entire image at once, YOLO leverages global information to reduce false positives and enhance accuracy [49]. The evolution of YOLO has seen the development of various versions, each improving upon the last. YOLOv2 and YOLOv3 introduced enhancements in accuracy and the ability to detect objects at multiple scales [50]. This study focuses on YOLOv7 and YOLOv8, the latest advancements in the YOLO series, selected for their improved architectures and performance.

2.3.3. YOLOv7 and YOLOv8

YOLOv7 features a deeper and wider CNN structure, utilizing multi-scale feature maps and improved bounding box regression techniques to enhance accuracy. It incorporates advanced data augmentation and custom cross-connection modules, optimizing the model for real-time applications with complex backgrounds. YOLOv8, the most innovative in the series, adopts a multi-layer perceptron architecture that excels in detecting complex geometric structures and textures. This model is particularly effective in low-light and noisy environments, with optimized feature extraction and information flow.

Both YOLOv7 and YOLOv8 are designed for real-time applications, but YOLOv8's more complex architecture offers superior performance in handling intricate tasks. Table 1 details the hyperparameters used for these models, which were carefully tuned to achieve optimal results during training and testing.

Table 1. Hyperparameters and Descriptions for YOLO Models

Parameter	Value	Description
lr0	0,0002	Initial speed for weight updates in the model
lrf	0,001	Learning rate used in the final stages of training
Momentum/Beta1	0,937	Cumulative effect of previous gradient updates
Weight Decay	0,0005	Prevents model overfitting
Box	0,05	Weight of bounding box loss
Cls	0,3	Weight of class loss
Obj	0,7	Weight of object loss
Iou_t	0,20	IoU training threshold
Anchor_t	4.0	Anchor box alignment threshold
Fl_gamma	0,0	Focal loss gamma

In this study, YOLOv7 and YOLOv8 models were employed for object detection and classification, with hyperparameters meticulously adjusted to maximize performance. The selected hyperparameters, as outlined in Table 1, were determined through extensive experimentation to achieve the best possible results under various conditions.

2.2.4. RT-DETR

The RT-DETR (Real-Time Detection Transformer) method introduces significant innovations in object detection by utilizing a transformer-based architecture, particularly effective in complex visual environments [51]. Unlike CNN-based approaches, RT-DETR employs a global prediction approach, simplifying the detection process by predicting objects collectively rather than individually. This method reduces training complexity and improves model efficiency, making it capable of handling overlapping and multi-scale objects [52]. The RT-DETR architecture comprises three main components: a backbone, a transformer, and a feed-forward network (FFN). Typically using ResNet for feature extraction, the transformer processes these features through multiple attention layers, capturing relationships between objects [53][54]. The final predictions for object classification and localization are produced by the FFN, utilizing cross-attention mechanisms to combine object queries with image features, resulting in accurate and scalable detection outcomes [55]. RT-DETR's transformer component represents a key innovation, allowing for global context understanding by linking different parts of the feature map through attention mechanisms. This capability is particularly advantageous in complex scenarios, where overlapping objects are detected with higher accuracy [56]. The model outputs a set of predicted classes and bounding boxes, optimized using bipartite matching and the Hungarian algorithm to align predictions with ground truth efficiently [57]. The performance of RT-DETR has been demonstrated on datasets like MS COCO, with significant improvements in object detection accuracy, particularly in challenging categories [58]. Modifications such as deformable attention mechanisms have further enhanced its capabilities, particularly in dense and complex scenes [59]. Despite its advantages, the model's training process can be computationally intensive, a challenge addressed in subsequent research focusing on optimization [60].

For this study, RT-DETR was employed with carefully selected hyperparameters to optimize performance during training and testing. The choice of parameters like the optimizer, learning rates, and cost weights played a crucial role in achieving accurate and efficient object detection [55]. Data augmentation techniques such as HSV adjustments and geometric transformations were also applied to enhance the model's robustness [59].

Table 2. Hyperparameters and Descriptions for RT-DETR Models

Parameter	Value	Description
optimizer	AdamW	The optimization algorithm of the model
base learning rate	0,0001	The initial learning rate of the model
learning rate of backbone	0,00001	Learning rate for the backbone network
weight decay	0,0001	Weight decay to prevent overfitting
number of AIFI layers	1	Number of Adaptive Addition and Subtraction Layers
number of RepBlocks	3	Number of repeating blocks
embedding dim	256	The dimension of embedding vectors
feedforward dim	1024	The dimension of the feedforward network
nheads	8	Number of heads in the multi-head attention mechanism
number of feature scales	3	Number of different feature scales
number of decoder layers	6	Number of layers in the decoder
number of queries	300	Maximum number of objects the model can process simultaneously
bbox cost weight	5.0	Weight of the bounding box cost function
GIoU cost weight	2.0	Weight of the Generalized IoU cost function
class loss weight	1.0	Weight of the class loss
bbox loss weight	5.0	Weight of the bounding box loss
GIoU loss weight	2.0	Weight of the Generalized IoU loss

Each parameter was fine-tuned to ensure optimal model performance in real-world conditions. Understanding these adjustments is key to improving future implementations.

2.3. Performance Metrics

To evaluate the performance of object detection models, the most commonly used metrics include Precision, Recall, and mean Average Precision (mAP). These metrics are based on fundamental concepts such as True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN), which are typically organized in a confusion matrix. Precision, measures the ratio of correctly identified positive examples to the total number of examples predicted as positive. High precision indicates that the model produces few false positives, and it is calculated using Equation (1).

$$P = \frac{TP}{TP + FP} \tag{1}$$

Recall, calculates the ratio of correctly identified positive examples to the total number of actual positive examples. High recall suggests that the model successfully detects most of the positive instances, as shown in Equation (2).

$$R = \frac{TP}{TP + FN} \tag{2}$$

mean Average Precision (mAP), evaluates the balance between precision and recall across different classes. It is computed as the average of the precision-recall curve areas for each class, as detailed in Equations (3) and Equations (4).

$$P = \sum_{n} \big(R(n) - R(n-1)\big)\, P(n) \tag{3}$$

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i \tag{4}$$

These metrics are crucial for understanding how well a model performs in real-world scenarios and for comparing different models. Precision and recall often exhibit a trade-off, where improving one may decrease the other. Therefore, balancing these metrics is essential for developing an effective object detection model.

Additionally, the F1 Score is frequently used to balance precision and recall. It is the harmonic mean of precision and recall, as defined in Equation (5).

$$F1\ Score = 2 * \left(\frac{P * R}{P + R}\right) \tag{5}$$

The Confusion Matrix, as shown in Table 3, provides a detailed breakdown of the model's predictions, illustrating the relationship between actual and predicted classifications. It includes True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN), offering insight into specific types of errors made by the model.

Table 3. Confusion Matrix

Actual / Predicted	**Positive Prediction**	**Negative Prediction**
Actual Positive	**True Positive (TP)**	**False Negative (FN)**
Actual Negative	**False Positive (FP)**	**True Negative (TN)**

These metrics and the confusion matrix are essential tools for analyzing model performance in-depth and guiding the model development process.

3. RESULTS and DISCUSSION

In this study, the performance of the object detection model was evaluated using commonly accepted metrics such as F1 score, precision, recall, and mean average precision (mAP). These metrics, explained in detail in Section 3, were used to analyze the model's effectiveness in accurately detecting and classifying objects.

3.1. YOLOv7

YOLOv7 is a prominent object detection model known for its real-time detection capabilities and high accuracy. The model's performance was thoroughly assessed using various metrics and graphs to determine its strengths and areas for improvement. Figure 1 presents the F1 score-confidence curve, where the F1 score, a harmonic mean of precision and recall, is plotted against different confidence levels. The average F1 score across all classes was 0.86 at a confidence level of 0.563, indicating the model's ability to balance precision and recall. The variation in F1 scores among different classes suggests that while the model performs consistently well for certain classes, its performance fluctuates depending on the confidence threshold.

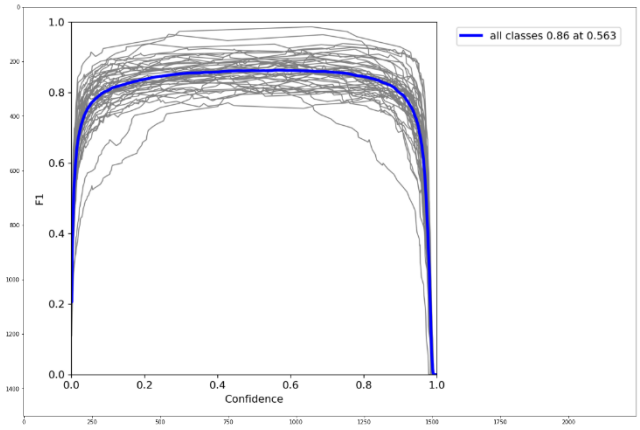

Figure 1. F1 Score-Confidence Curve for YOLOv7 Model

Figure 2 illustrates the precision-recall curve for the YOLOv7 model, showing the trade-off between precision and recall across different classes. The model achieved a mean average precision (mAP) of 0.902 at an IoU threshold of 0.5, indicating a strong balance between precision and recall in detecting objects with a significant overlap. Figure 3 depicts the precision-confidence curve, highlighting the model's precision across varying confidence levels. The model reached a

precision of 1.00 at a confidence level of 0.982, demonstrating its capability to produce highly accurate detections at near-perfect confidence levels. However, the curve also indicates that precision varies across classes, emphasizing the model's strong performance in certain categories and room for improvement in others.

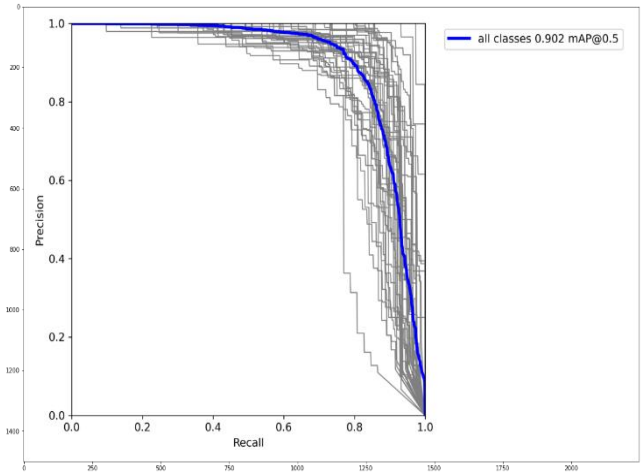

Figure 2. Precision-Recall Curve for YOLOv7 Model

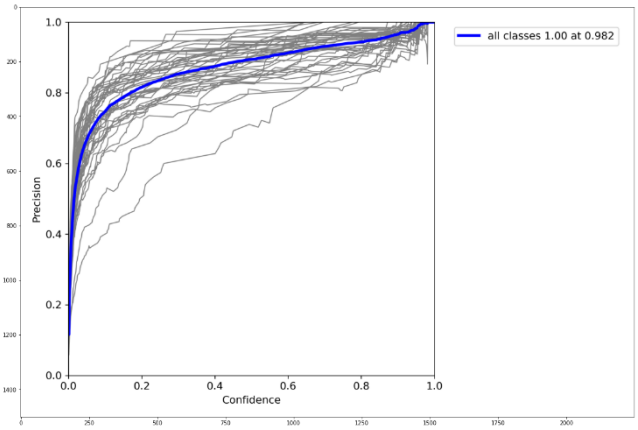

Figure 3. Precision-Confidence Curve for YOLOv7 Model

Figure 4 shows the confusion matrix for YOLOv7, displaying the accuracy of the model's predictions across different classes. The diagonal elements represent correctly classified instances, while off-diagonal elements indicate misclassifications. High accuracy rates in classes such as AG600 and Mig31 reflect the model's reliability, whereas lower accuracy in classes like F117 suggests areas where further model refinement is needed. The experimental results indicate that the YOLOv7 model can effectively detect and classify various aircraft types with high accuracy across different conditions. These findings are supported by the visual examples provided in Figure 1 and Figure 2 , showcasing the model's robust detection capabilities in real-world scenarios.

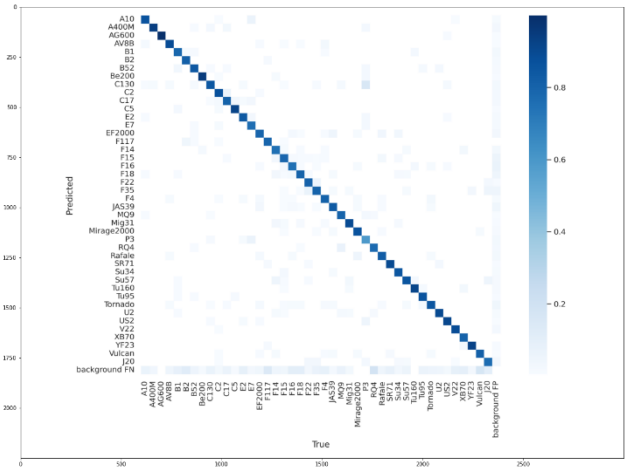

Figure 4. Confusion Matrix for YOLOv7

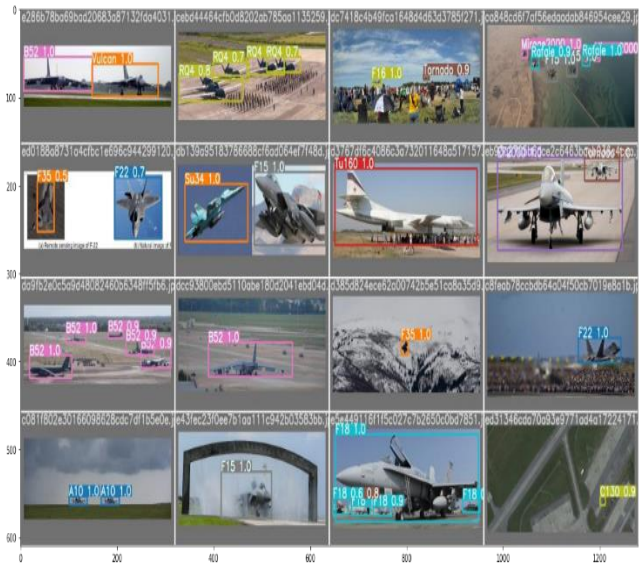

Figure 5. Sample Detection Results Using YOLOv7

The visual results presented in Figure 5 provide a tangible representation of YOLOv7's detection capabilities, aligning well with the quantitative metrics discussed earlier. The model successfully identifies and localizes various aircraft, including those with overlapping features or challenging backgrounds, reaffirming its effectiveness in real-world applications.

3.2. YOLOv8

The YOLOv8 model's performance was rigorously analyzed through a variety of metrics and visual representations. This evaluation emphasized its strengths in accuracy and reliability, while also identifying specific areas where further enhancements could be made. Figure 6 presents the F1 score-confidence curve, where the relationship between F1 scores and confidence levels is illustrated. The average F1 score across all classes reached 0.90 at a confidence level of 0.695, indicating the model's strong performance at this level. The variability in F1 scores among different classes suggests that while the model performs well for certain classes, its performance varies depending on the confidence threshold. Figure 7 shows the precision-recall curve, which highlights the model's ability to balance precision and recall. The model achieved a mean average precision (mAP) of 0.940 at an IoU threshold of 0.5, reflecting its high accuracy and reliability across different classes.

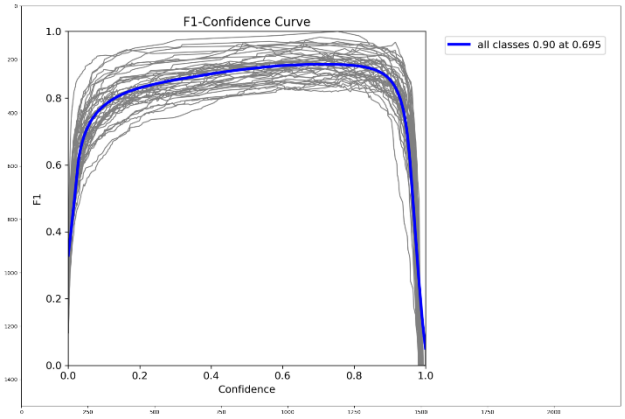

Figure 6. F1 Score-Confidence Curve for YOLOv8 Model

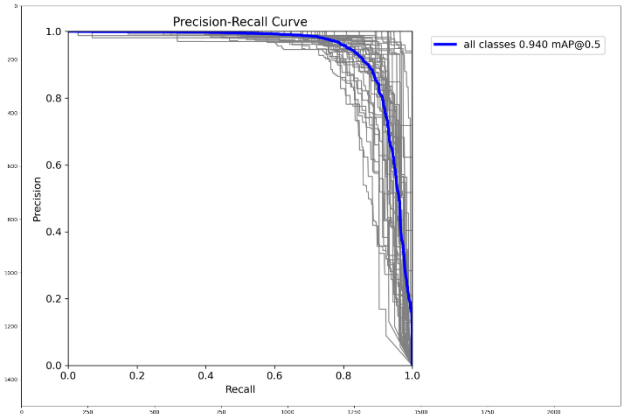

Figure 7. Precision-Recall Curve for YOLOv8 Model

Figure 8 depicts the precision-confidence curve, demonstrating that the model reached a precision of 1.00 at a confidence level of 1.00, indicating near-perfect accuracy at high confidence levels. This suggests that the model is highly reliable in making accurate predictions when it operates at maximum confidence.

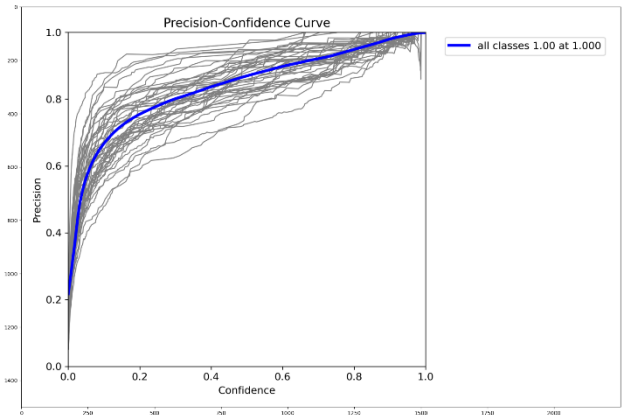

Figure 8. Precision-Confidence Curve for YOLOv8 Model

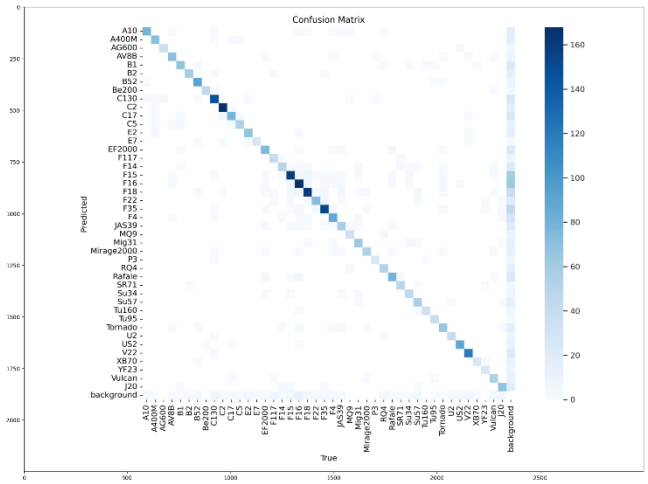

Figure 9. Confusion Matrix for YOLOv8

Figure 9 displays the confusion matrix, which illustrates the accuracy of the model's predictions across various classes. The matrix highlights the model's strong performance in distinguishing between different classes, with high accuracy rates for many classes. However, it also reveals areas where the model could benefit from further refinement.

The experimental results confirm that the YOLOv8 model is highly effective in detecting and classifying various types of aircraft with remarkable accuracy, even under diverse and challenging environmental conditions. This effectiveness is not only evident in the quantitative metrics, such as the high mean Average Precision (mAP) and precision-recall balance, but also in the qualitative assessment of the model's detection capabilities. Figure 6 and Figure 7 illustrate the F1 score-confidence and precision-recall curves, respectively, highlighting the model's ability to maintain strong performance across varying confidence thresholds and object scales. Furthermore, Figure 10 presents visual examples of the YOLOv8 model's successful detections across different scenarios, showcasing its robustness and reliability in real-world applications. These examples underscore the model's proficiency in accurately identifying and localizing aircraft, even in complex scenes with varied backgrounds and lighting conditions.

Figure 10. Sample Detection Results Using YOLOv8

3.3. RT-DETR

The RT-DETR model, known for its effective use of attention mechanisms and real-time detection capabilities, was subjected to an extensive performance analysis. The study utilized a range of metrics and visual tools to assess its accuracy across various confidence levels and classes, highlighting both its robust performance and areas needing refinement.

Figure 11 illustrates the F1 score-confidence curve for the RT-DETR model, showing the relationship between F1 scores and confidence levels. The model achieved a high F1 score of 0.93 at a confidence level of 0.637, indicating a strong balance between precision and recall at this confidence level.

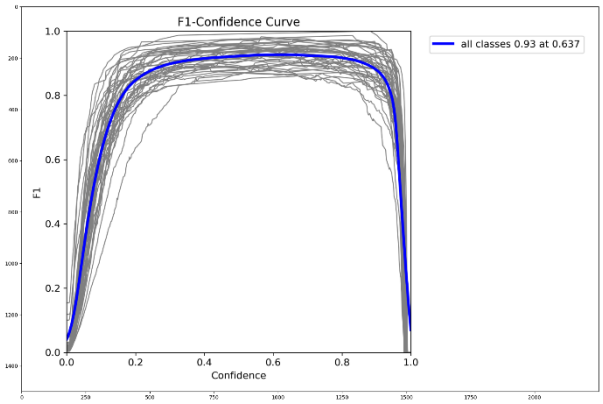

Figure 11. F1 score-Confidence curve for the RT-DETR model

Figure 12 presents the precision-recall curve for the RT-DETR model. The model attained a mean Average Precision (mAP) of 0.927 at an IoU threshold of 0.5, demonstrating its reliable performance in detecting and correctly identifying most objects.

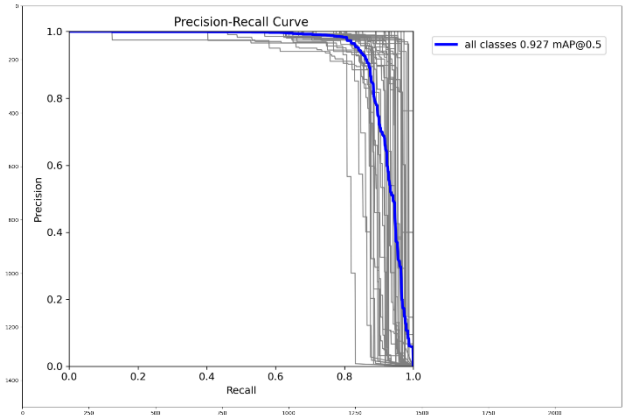

Figure 12. Precision-Recall Curve for RT-DETR Model

Figure 14 shows the precision-confidence curve, where the model achieved a precision of 1.00 at a confidence level of 1.000, indicating that at this level, all predicted objects were correctly classified, highlighting the model's high reliability at maximum confidence.

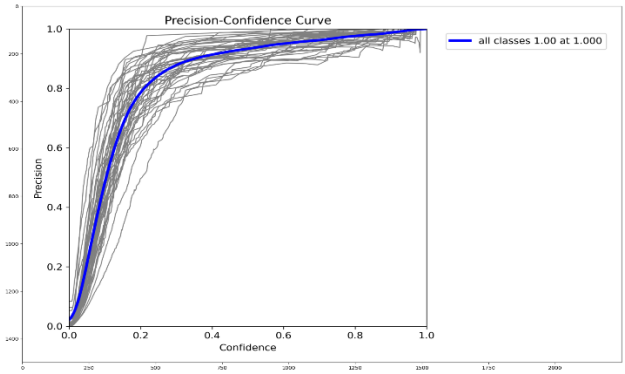

Figure 13. Precision-Confidence Curve for RT-DETR Model

Figure 14 displays the confusion matrix for the RT-DETR model, showing the model's accuracy in distinguishing between various classes. The matrix's diagonal shows high accuracy for several aircraft classes, but also highlights areas where the model struggled, particularly with visually similar aircraft.

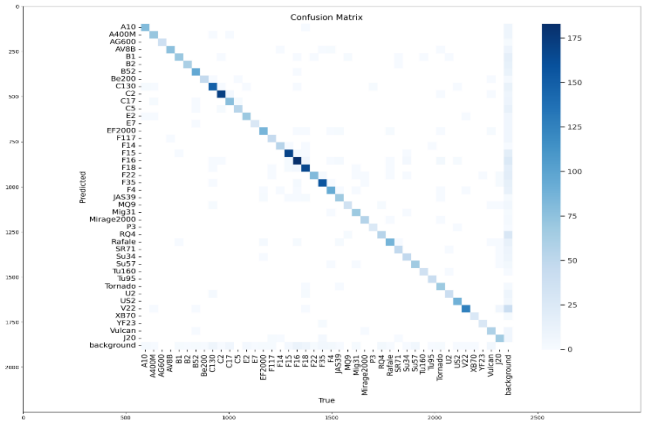

Figure 14. Confusion Matrix for RT-DETR

In addition to these performance metrics, Figure 16 provides visual examples of the RT-DETR model's successful detections across different scenarios, further demonstrating its effectiveness in real-world applications.

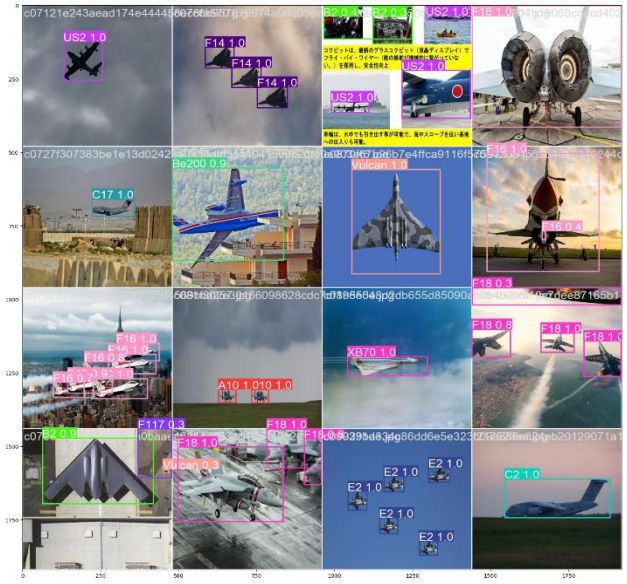

Figure 15. Sample Detection Results Using RT-DETR

The RT-DETR model, despite its architectural differences from traditional CNN-based models like YOLO, demonstrated commendable performance in detecting and classifying various types of aircraft across diverse environmental conditions. Figure 11 and Figure 12 highlight the model's robust F1 score and precision-recall performance, respectively, while Figure 13 emphasizes the model's high precision at maximum confidence levels. The model's ability to accurately distinguish between various classes is further illustrated in Figure 14 through the confusion matrix. Finally, Figure 15 showcases the model's strong detection capabilities with visual examples of successful aircraft detections in real-world scenarios.

YOLOv7 demonstrated strong performance with a mAP@.5 score of 0.902, achieving precision values above 0.90 for classes like A400M, AG600, and Be200, alongside high recall rates in these categories. However, the model exhibited lower recall for certain classes, such as F117, indicating areas for potential improvement. YOLOv8 surpassed YOLOv7 in several metrics, achieving a mAP@.5 of 0.94 and maintaining precision values above 0.90 for most classes, with particularly high performance for classes like Tu160 and Tu95. The model also excelled in recall for classes such as C2 and US2, likely due to algorithmic optimizations and potentially more extensive training data.

The RT-DETR model outperformed the previous two, with a mAP@.5 of 0.93, and exhibited high precision and recall across nearly all classes. Notably, it achieved excellent results in EF2000, F35, and Rafale classes, as well as in P3 and E7, demonstrating its effectiveness across a broad range of classes.

As shown in Table 4. Model Performance Comparison, the RT-DETR model consistently delivered higher precision (0.952) and recall (0.904) compared to YOLOv8 (precision: 0.924, recall: 0.881) and YOLOv7 (precision: 0.907, recall: 0.827). Although YOLOv8 achieved the highest mAP@50 value at 0.94, RT-DETR excelled in overall precision and recall, indicating its superior performance in object detection tasks across a wide dataset. YOLOv7, while trailing behind, still produced effective results in specific classes.

Table 4. Model Performance Comparison

Model	Instances	P	R	mAP@ 50	mAP
YOLOv7	3578	0.907	0.827	0.902	0.829
YOLOv8	3578	0.924	0.881	0.940	0.877
RT-DETR	3578	0.952	0.904	0.927	0.879

This comparison highlights the RT-DETR model's superior accuracy and consistency in object detection tasks, particularly when high precision and recall are critical.

4. CONCLUSION AND RECOMMENDATIONS

This study has explored the potential of deep learning-based models for the automatic detection of military aircraft, a critical task in modern warfare and strategic surveillance operations. The study utilized an extensive dataset comprising 19.514 images across 43 different military aircraft classes. The primary objective was to accurately classify and detect these classes using YOLOv7, YOLOv8, and RT-DETR models. The evaluation of each model was conducted using metrics such as Precision, Recall, and mean Average Precision at IoU threshold 0.5 (mAP@.5).

Performance analyses revealed varying results across models, highlighting each model’s strengths and weaknesses. The YOLOv7 model demonstrated impressive overall performance but struggled with lower-than-expected Recall rates in classes like F117. The YOLOv8 model built upon the performance of YOLOv7, achieving higher overall mAP values and displaying superior Precision in most classes, particularly in the Tu160 and Tu95 classes. Meanwhile, the RT-DETR model provided more consistent and superior results across Precision and Recall metrics, proving highly reliable across almost all classes. However, all three models exhibited a need for improved performance in certain classes. Low recall rates in some classes suggest that the models may not adequately recognize objects within these classes, possibly due to insufficient training data for those categories. Additionally, high false positive rates in some cases indicate limitations in the models' generalization capabilities or imbalances in the dataset. This highlights the potential benefit of augmenting the dataset with additional samples from underperforming classes. To enhance model performance, several strategies are recommended. First, additional data should be collected for classes with lower performance, and efforts should be made to balance the dataset. Employing data augmentation techniques can also help improve model robustness. This study has demonstrated the significant potential of deep learning models for military aircraft detection. By implementing these recommendations, it is expected that model performance can be further enhanced, enabling broader and more effective applications.

Future studies are intended to enhance model performance by addressing class imbalance and improving generalization capabilities. Specifically, optimizing model architectures, increasing data diversity, and employing advanced data augmentation techniques could lead to significant performance improvements. Recent data augmentation methods, such as MixUp, CutMix, Mosaic, and Copy-Paste, can be utilized to address class imbalance and improve robustness. Moreover, it is essential to evaluate models in terms of speed and efficiency for real-time applications. Such advancements are expected to provide more reliable and effective solutions for the automatic detection of military aircraft, contributing significantly to both military and civilian applications.

REFERENCES

[1] K. Bayoudh, R. Knani, F. Hamdaoui, A. Mtibaa, "A survey on deep multimodal learning for computer vision: advances, trends, applications, and datasets", The Visual Computer, 38(8), 2939-2970, 2022.

[2] A. A. Khan, A. A. Laghari, S. A. Awan, "Machine learning in computer vision: a review", EAI Endorsed Transactions on Scalable Information Systems, 8(32), 2021.

[3] J. Zhao, R. Masood, S. Seneviratne, "A review of computer vision methods in network security", IEEE Communications Surveys & Tutorials, 23(3), 1838-1878, 2021.

[4] E. Dilek, M. Dener, "Computer vision applications in intelligent transportation systems: a survey", Sensors, 23(6), 2938, 2023.

[5] R. Szeliski, Computer Vision: Algorithms and Applications, Springer Nature, 2022.

[6] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, D. Terzopoulos, "Image segmentation using deep learning: A survey",IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(7), 3523-3542, 2022.

[7] W. Chen, Y. Li, Z. Tian, F. Zhang, "2D and 3D object detection algorithms from images: A Survey", Array, 100305, 2023.

[8] X. Zhuang, D. Li, Y. Wang, K. Li, "Military target detection method based on EfficientDet and Generative Adversarial Network", Engineering Applications of Artificial Intelligence, 132, 107896, 2024.

[9] S. Khalid, H. M. Oqaibi, M. Aqib, Y. Hafeez, "Small pests detection in field crops using deep learning object detection", Sustainability, 15(8), 6815, 2023.

[10] M. Abdel-Aty, Y. Wu, O. Zheng, J. Yuan, "Using closed-circuit television cameras to analyze traffic safety at intersections based on vehicle key points detection", Accident Analysis & Prevention, 176, 106794, 2022.

[11] J. Liu, Y. Jin, "A comprehensive survey of robust deep learning in computer vision", *Journal of Automation and Intelligence*, 2023.

[12] K. Roopa, T. V. Rama Murthy, P. C. Prasanna Raj, "Neural network classifier for fighter aircraft model recognition", *Journal of Intelligent Systems*, 27(3), 447-463, 2018.

[13] H. Zhu, H. Lung, N. Lin, "Carrier-based aircraft detection on flight deck of aircraft carrier with simulated 3-D model by deep neural network", 3rd International Conference on Computer Science and Software Engineering, 96–101, May 2020.

[14] Q. Liu, X. Xiang, Y. Wang, Z. Luo, F. Fang, "Aircraft detection in remote sensing image based on corner clustering and deep learning", Engineering Applications of Artificial Intelligence, 87, 103333, 2020.

[15] W. Ma, H. Chen, Y. Zhang, "An improved YOLOv3 model for aircraft detection in remote sensing images", IEEE Access, 8, 120129-120138, 2020.

[16] Y. Yang, G. Xie, Y. Qu, "Real-time detection of aircraft objects in remote sensing images based on improved YOLOv4", 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), 1156–1164, March 2021.

[17] Q. Wu, D. Feng, C. Cao, X. Zeng, Z. Feng, J. Wu, Z. Huang, "Improved mask R-CNN for aircraft detection in remote sensing images", Sensors, 21(8), 2618, 2021.

[18] L. Zhou, H. Yan, Y. Shan, C. Zheng, Y. Liu, X. Zuo, B. Qiao, "Aircraft detection for remote sensing images based on deep convolutional neural networks", *Journal of Electrical and Computer Engineering*, 2021(1), 4685644, 2021.

[19] M. Liu, Q. Hu, C. Wang, T. Tian, W. Chen, "Daff-Net: Dual attention feature fusion network for aircraft detection in remote sensing images", 2021 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 4196–4199, July 2021.

[20] L. Zhou, L. Zhang, N. Konz, "Computer vision techniques in manufacturing", IEEE Transactions on Systems, Man, and Cybernetics: Systems, 53(1), 105–117, 2022.

[21] E. Kiyak, G. Unal, "Small aircraft detection using deep learning", Aircraft Engineering and Aerospace Technology, 93(4), 671–681, 2021.

[22] H. M. A. Mohammed, M. Polat, A. A. Tahlıl, İ. Y. Özbek, "Multi-scale aircraft detection from satellite images", *Erzincan University Journal of Science and Technology*, 14(1), 322–330, 2021.

[23] Y. Wang, T. Wang, X. Zhou, W. Cai, R. Liu, M. Huang, et al., "TransEffiDet: aircraft detection and classification in aerial images based on EfficientDet and transformer", Computational Intelligence and Neuroscience, 2262549, 2022.

[24] P. Gupta, B. Pareek, G. Singal, D. V. Rao, "Edge device based military vehicle detection and classification from UAV", Multimedia Tools and Applications, 81(14), 19813–19834, 2022.

[25] A. D. W. Sumari, D. E. Adinandra, A. R. Syulistyo, S. Lovrencic, "Intelligent Military Aircraft Recognition and Identification to Support Military Personnel on the Air Observation Operation", *International Journal on Advanced Science, Engineering, and Information Technology (IJASEIT)*, 6(Accepted for Publication), 2022.

[26] T. Wang, X. Zeng, C. Cao, W. Li, Z. Feng, J. Wu, et al., "CGC-NET: Aircraft Detection in Remote Sensing Images Based on Lightweight Convolutional Neural Network", *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15, 2805–2815, 2022.

[27] S. Lou, J. Yu, Y. Xi, X. Liao, "Aircraft target detection in remote sensing images based on improved YOLOv5", IEEE Access, 10, 5184–5192, 2022.

[28] W. Liu, J. Tian, T. Tian, "YOLM: A remote sensing aircraft detection model", IGARSS 2022–2022 IEEE International Geoscience and Remote Sensing Symposium, 1708–1711, July 2022.

[29] B. Azam, M. J. Khan, F. A. Bhatti, A. R. M. Maud, S. F. Hussain, A. J. Hashmi, K. Khurshid, "Aircraft detection in satellite imagery using deep learning-based object detectors", Microprocessors and Microsystems, 94, 104630, 2022.

[30] P. Benjamin, B. Benjamin, G. Dimitri, S. Gérard, E. Eric, "Oriented aircraft object detector using Scaled YOLOv4 on very high resolution satellite and synthetic datasets", 2023 Joint

Urban Remote Sensing Event (JURSE), 1–4, May 2023.

[31] Z. Liu, Y. Gao, Q. Du, M. Chen, W. Lv, "YOLO-extract: Improved YOLOv5 for aircraft object detection in remote sensing images", IEEE Access, 11, 1742–1751, 2023.

[32] F. Zhou, H. Deng, Q. Xu, X. Lan, "CNTR-YOLO: Improved YOLOv5 Based on ConvNext and Transformer for Aircraft Detection in Remote Sensing Images", Electronics, 12(12), 2671, 2023.

[33] M. Zhu, E. Kong, "Multi-Scale Fusion Uncrewed Aerial Vehicle Detection Based on RT-DETR", Electronics, 13(8), 1489, 2024.

[34] A. Kumar, S. Singh, "AIR-SCAN: Aircraft Identification and Recognition using Deep Learning Scanning", 2024 11th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), 1–6, March 2024.

[35] K. He, X. Zhang, S. Ren, J. Sun, "Deep residual learning for image recognition", Proceedings of the IEEE conference on computer vision and pattern recognition, 770–778, 2016.

[36] R. Girshick, J. Donahue, T. Darrell, J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation", Proceedings of the IEEE conference on computer vision and pattern recognition, 580–587, 2014.

[37] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, "You only look once: Unified, real-time object detection", Proceedings of the IEEE conference on computer vision and pattern recognition, 779–788, 2016.

[38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al., "Attention is all you need", Advances in neural information processing systems, 5998–6008, 2017.

[39] J. H. Kim, N. Kim, C. S. Won, "High-speed drone detection based on yolo-v8", ICASSP 2023–2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 1–2, June 2023.

[40] S. K. Shandilya, A. Srivastav, K. Yemets, A. Datta, A. K. Nagar, "YOLO-based segmented dataset for drone vs. bird detection for deep and machine learning algorithms", Data in Brief, 50, 109355, 2023.

[41] S. Patil, S. M. Jaybhaye, M. M. Khalifa, S. Kharche, A. Khatib, A. Kshirsagar, "Drone detection using YOLO", AIP Conference Proceedings, 2938(1), December 2023.

[42] A. Coluccia, A. Fascista, A. Schumann, L. Sommer, A. Dimou, D. Zarpalas, et al., "Drone vs. bird detection: Deep learning algorithms and results from a grand challenge", Sensors, 21(8), 2824, 2021.

[43] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, S. Zagoruyko, "End-to-end object detection with transformers", European Conference on Computer Vision (ECCV), 213–229, 2020.

[44] Z. Sun, S. Cao, Y. Yang, K. M. Kitani, "Rethinking transformer-based set prediction for object detection", Proceedings of the IEEE/CVF International Conference on Computer Vision, 3611–3620, 2021.

[45] R. u, D. Wunsch, "Survey of clustering algorithms", IEEE Transactions on Neural Networks, 16(3), 645–678, 2005.

[46] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, et al., "Microsoft COCO: Common objects in context", Computer Vision – ECCV 2014, 740–755, 2014.

[47] N. Dalal, B. Triggs, "Histograms of oriented gradients for human detection", 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 886–893, June 2005.

[48] S. Ren, K. He, R. Girshick, J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks", Advances in Neural Information Processing Systems, 28, 2015.

[49] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, Y. Wei, "Deformable convolutional networks", Proceedings of the IEEE International Conference on Computer Vision, 764–773, 2017.

[50] S. J. Russell, P. Norvig, Artificial Intelligence: A Modern Approach, Pearson, 2016.

[51] C. M. Bishop, N. M. Nasrabadi, Pattern Recognition and Machine Learning, Vol. 4(4), Springer, New York, 2006.

[52] T. Hastie, R. Tibshirani, J. H. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer, New York, 2009.

[53] D. G. Lowe, "Distinctive image features from scale-invariant keypoints", *International Journal of Computer Vision*, 60, 91–110, 2004.

[54] Y. LeCun, Y. Bengio, G. Hinton, "Deep learning", Nature, 521(7553), 436–444, 2015.

[55] C. Cortes, V. Vapnik, "Support-vector networks", Machine Learning, 20(3), 273–297, 1995.

[56] A. Krizhevsky, I. Sutskever, G. E. Hinton, "ImageNet classification with deep convolutional neural networks", Communications of the ACM, 60(6), 84–90, 2017.

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

[58] J. Redmon, A. Farhadi, "YOLOv3: An incremental improvement", arXiv preprint arXiv:1804.02767, 2018.

[59] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection", arXiv preprint arXiv:2010.04159, 2020.