Vol: 2, Issue: 2, 2024
Pages:150-171
Received: 04 November 2024
Accepted: 13 January 2025
© 2024 Karabük University

CURRENT TRENDS IN COMPUTING
Research Article
https://doi.org/10.71074/CTC.1578917

FINE-GRAINED CLASSIFICATION OF MILITARY AIRCRAFT USING PRE-TRAINED DEEP LEARNING MODELS AND YOLO11

HASAN KARACA[1], NESRIN AYDIN ATASOY[2*]

[1] *The Institute of Graduate Programs, Computer Engineering Department, Karabük University, 78050, Karabük, Türkiye*
[2] *Computer Engineering Department, Karabük University, 78050, Karabük, Türkiye*

ABSTRACT. **This research examines the potential of pre-trained deep learning models for the fine-grained classification of military aircraft, to achieve accurate identification and extraction of unique tail numbers. The study uses a publicly available dataset comprising 43 classes of military aircraft, with a total of 24,164 images for training and 6,042 images for testing. The performance of five distinct pre-trained convolutional neural network (CNN) architectures, including DenseNet121, MobileNetV2, ResNet50, ResNet101, and VGG19, is evaluated and compared. Furthermore, the paper examines the effectiveness of the YOLO11 model family for aircraft classification,with the YOLO11x-cls model achieving the highest accuracy of 95.9, demonstrating its superior performance. particularly emphasizing the YOLO11x-cls model's superior performance. The study analyses the training results and confusion matrix of the YOLO11x-cls model, demonstrating its accuracy and ability to generalize well to unseen data. This work contributes to the advancement of AI-powered image recognition for military aviation applications, potentially improving data collection, monitoring, and analysis processes.**

1. INTRODUCTION

Aircraft are an important part of military forces when it comes to performing duties related to national defense and security. In this advanced technological area, Artificial Intelligence (AI) and Optical Character Recognition (OCR) have combined to provide an unprecedented boost in processing textual information, while persistently gaining interest in the art of classifying the aircraft themselves and reading off their wing numbers. OCR technology, being the process of extraction of written content from the visual appearance of data to textual form, has undergone a profound change by integrating AI capabilities [1]. Besides the critical role that military aircraft play in national defense, their efficient classification, and precise identification, including the reading of the wing numbers precisely, is also an essential aspect of

E-mail address: hasankaraca9163@yandex.com , nesrinaydin@karabuk.edu.tr(*).
Key words and phrases. Fine-Grained Classification, Deep Learning, Convolutional Neural Networks, YOLO11, ResNet50.

ensuring maximum operational effectiveness. AI and OCR came together to cause a revolution in processing textual information and turned out to be important tools for dealing with complications regarding military aircraft data [2].

This integration has augmented not only the accuracy and speed of OCR but also its application in various fields such as finance, healthcare, and legal services [3]. Among the innovative techniques is Layout Agnostic Alignment (LAA) [4], which solves the problem of harmonizing document layouts across different systems using OCRs. Very recently, the integration of OCR and Text-based Visual Question Answering has reached a milestone, underlining the exacting integration of both technologies seamlessly. This includes the design of specific deep neural network models for dedicated tasks like automatic license plate recognition with BLPnet. Newer efforts like Clip-OCR and Master Object (COME) [5] have moved the goalposts further in the case of representations for text and images by contrastive learning and representation learning through multi-modal feature extraction. Moreover, post-correction of the errors generated by OCR and optimization of document recognition and data extraction [6, 7] have acted to stress the emerging primacy of AI-based OCR. In particular, the recent innovative research in the unsupervised ranking of name entities from garbled OCR text [8] and OCR-based product classification in the retail sector [9] has widened the horizon for the spread of OCR applications. Finally, the statistical learning models built recently for correction of OCR errors are extremely promising in further raising accuracy and reliability [10].

Another related area in which the recent works have significantly brought about a revolution in image processing and classification related to aircraft is the damage detection in aircraft engine bore scope images using deep learning where a new benchmark was established regarding the accuracy of inspection. The Scattering Characteristics Analysis Network (SCAN) has significantly altered the way the type of aircraft classification was done in few-shot image settings where high-quality Synthetic Aperture Radar images are available. Besides, several deep learning approaches have been revealing their encouraging performance for small aircraft detection. In particular, a modified ResNet-50 architecture applied to the large-satellite image processing and the Scattering Topology Network-ST-Net significantly reduces the processing time, thereby improving object recognition in the Synthetic Aperture Radar (SAR) images. Also, deep learning frameworks have been designed in the case of automatically detecting aircraft in remotely sensed satellite images to find small objects in a complex scene. On the other hand, machine learning models using radar data from small unmanned aerial systems have opened ways for scalable traffic management and safety improvements. One of the recent end-to-end aircraft detection algorithms outperformed other methods by a margin: [11]. Aircraft classification research has focused on Principle Component Analysis (PCA) and feature fusion techniques. This has vastly improved the performance of feature classification. Specific research on identifying aircraft types by Mask R-CNN enables the accurate identification and classification of aircraft types from high-resolution satellite images: [12]. Comparative studies have finally established the efficacy of deep learning methods for object detection, further enhancing the accuracy of detection [13].

Deep learning represents a class of machine learning methodologies using neural networks that can perform complex tasks on vast volumes of data [14]. It employed several robust pre-trained models

including ResNet50 [15], ResNet101 [16], ShuffleNet [17], Xception [18], GoogLeNet [19], Inception-V3 [20], MobileNet-V2 [21], Inception-ResNet-V2 [22] and NASNet-Mobile [23]. These pre-trained models achieved a lot of success in deep learning and, hence, are considered to show excellent performance in several parts of computer vision tasks, object recognition, natural language processing [24], and other artificial intelligence applications.

Additionly Gao and Wen-jun presented the IDBO-KELM model, which remarkably enhanced the accuracy of identification of aerodynamic parameters due to the reduction of errors in transonic regions [25], hence proving its potential in precise aircraft performance analysis. Proposed the MPSA-DenseNet model, a multi-task learning model with attention mechanisms that had achieved high accuracy for complex datasets classification tasks [26]. This methodology can also be extended to aerial data analysis. Also applied the Harris Hawks Optimization algorithm in feature selection to optimize model efficiency by minimizing feature sets while retaining high predictive accuracy [27]. This is important for computationally intensive domains in aircraft classification. Further evidence is derived from health diagnostics applications [28], where Rough Neutrosophic Attribute Reduction was combined with DL-based techniques to show how deep learning frameworks are really strong in handling big and complex datasets and improving decision-making processes therein. Collectively, these works present an overview of the developments around deep learning and machine learning models, emphasizing aspects related to accuracy, efficiency, and adaptability that make them highly relevant for advanced classification tasks, including aircraft identification.

2. METHODOLOGY

2.1. **Data:**

The dataset shown in Figure 1 used in this study was obtained from Kaggle and is named ”Military Aircraft Detection Dataset” [29]. The dataset consists of 43 classes, containing 24,164 images in the training set and 6,042 in the test set. Each image belonging to a specific aircraft is stored in a folder with the name of the corresponding class. Notably, there is no dedicated test dataset; therefore, a separation has been implemented that allocates twenty percent of the images from each class as test data, while the remaining eighty percent constitutes the training data.

Table 1 provides a tabular format that has different object categories, presumably aircraft, with numerical values provided in two separate columns labeled as ”Train” and ”Validation.” The tabular structure represents one format of a dataset to train and then validate types of aircraft through machine learning.

Each row in the table represents a unique category, potentially corresponding to a specific aircraft model or type. The ”Train” column indicates the number of instances available in the training set for a given category. The training set serves as the basic data used to instruct the model to recognize and distinguish between the different categories. Conversely, the ”Validation” column denotes the number of instances present within the validation set, which serves as a means to assess the model’s accuracy when confronted with previously unseen data, thereby ensuring its ability to effectively generalize to novel instances.

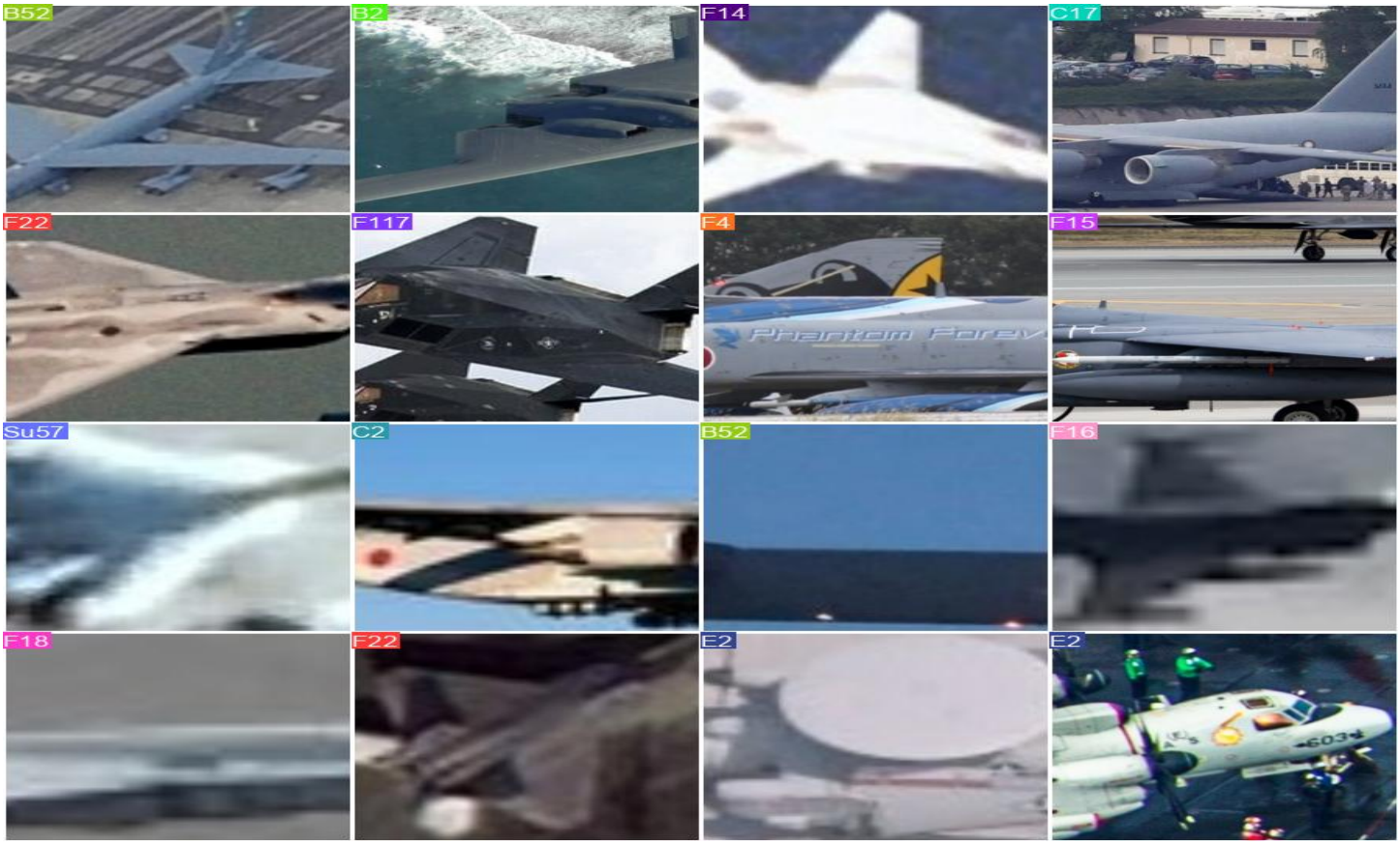


FIGURE 1. **Military aircraft class examples from different angles.**

This dataset is to be used in a k-fold cross-validation framework. K-fold cross-validation is a procedure where the data is divided into a number of K equally sized folds or subsets. Then, the model is going to be trained on k-1 of these folds, while the remaining fold will be used for its validation. It repeats K times where each fold acts once as a validation set. This technique enables the testing of the model's performance against independent datasets, thus becoming helpful when there is a lack of data resources.

The classes listed above represent some of the known model variants for aircraft models, such as F16, Rafale, and B52. Numerical values show how many images or data points within a particular category are available to train and validate the model for gaining knowledge and improving the model classification capability.

The information in Table 1 clearly elucidates how many training and validation data as shown Figure 2 each class of aircraft has. The composition of the dataset has been rich by applying data augmentation techniques as per the poor count of instances for some classes of military aircraft. Though there are classes with more than 890 instances in their training data, some classes have less than 400 instances.

TABLE 1. **Distribution of aircraft classes**

Class Name	Train	Validation	Class	Train	Validation	Class	Train	Validation
MQ9	561	140	B2	631	158	J20	562	141
JAS39	639	160	C130	611	153	F22	513	129
V22	546	137	YF23	424	106	F16	1339	335
Rafale	662	165	SR71	519	130	Vulcan	299	75
Mig31	554	139	U2	516	130	A400M	366	92
C17	666	167	C2	549	137	B1	500	126
AG600	480	119	F18	890	223	F117	284	71
F14	640	160	B52	645	161	F15	1147	287
F35	741	185	Su57	541	136	E7	146	37
Tornado	599	149	C5	583	145	XB70	137	35
EF2000	351	88	Tu95	500	124	AV8B	345	87
P3	459	115	E2	600	151	Be200	224	56
Tu160	513	129	US2	448	111	A10	550	138
F4	378	95	Su34	540	135	-	-	-
Mirage2000	585	146	RQ4	236	59	-	-	-

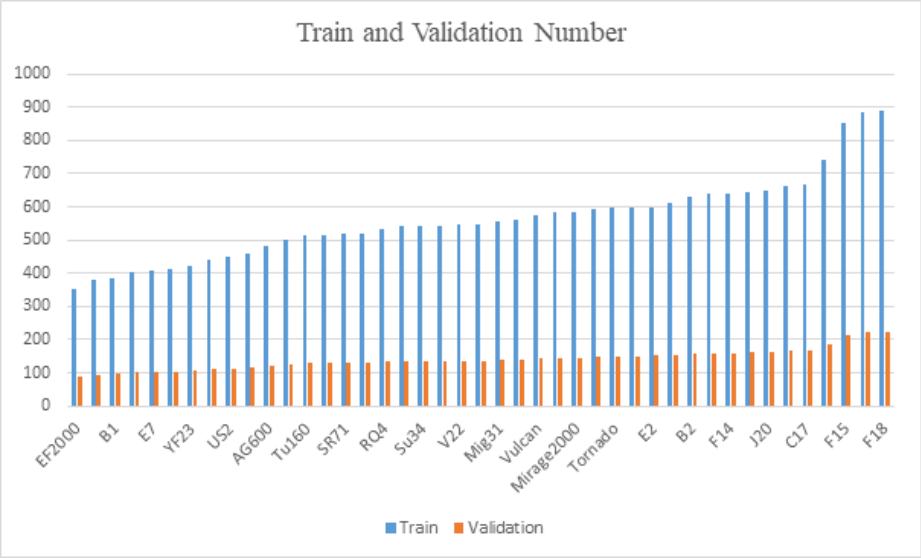


FIGURE 2. **Train and validation number of each class.**

2.2. **Methods:**

These models utilized for military aircraft classification in the current study are DenseNet121, MobileNetV2, ResNet50, ResNet101, and VGG19. These will be fine-tuned using ImageNet weights, the

weights that were developed during pre-training using the weights from the ImageNet dataset as training data. ImageNet is a dataset that has been trained for several object classification tasks and very frequently is used to fine-tune pre-trained models.

It has been shown that these models classify military aircraft with high accuracy. The goals of this study were to evaluate the performance of various deep learning architectures on tasks of military aircraft classification. Models with different architectures, such as DenseNet121, MobileNetV2, ResNet50, ResNet101, and VGG19, have been tried out to determine which is best for this particular task.

2.2.1. ***DenseNet121***. Among such famous CNN architectures is DenseNet121, which stands out because of its dense connectivity pattern. While in a traditional CNN, each layer feeds only its subsequent layer, DenseNet introduces direct connections from every layer to every other layer in a feed-forward fashion. The successive reuse of features through the dense connectivity makes it possible for gradients to propagate efficiently in the network, hence alleviating the vanishing gradient problem of deep networks. DenseNet121 explicitly contains 121 layers and is also built by stacking dense blocks composed of several convolutional layers with batch normalization and ReLU activation, followed by a transition layer for the purpose of reducing dimensionality. This architecture results in better parameter efficiency and feature extraction; hence, it is quite suitable for image classification tasks.

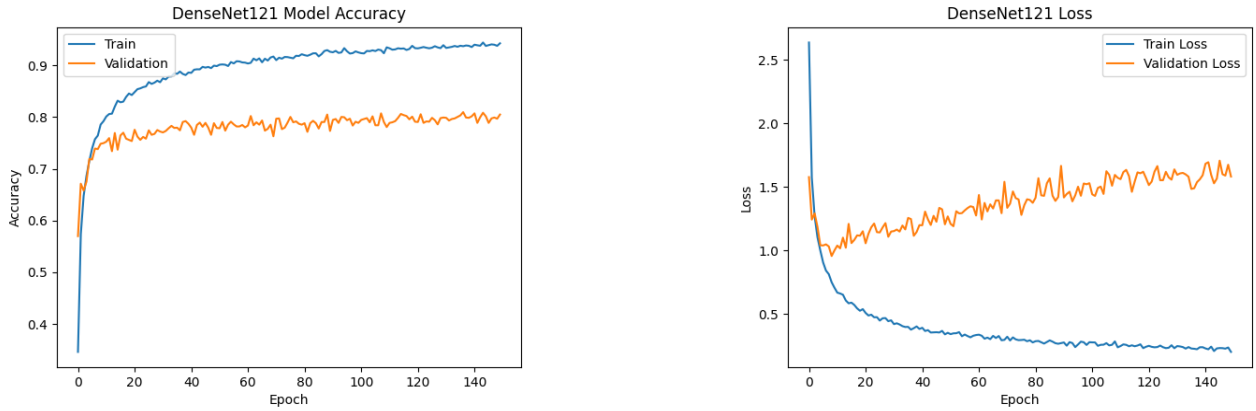
2.2.2. ***MobileNetV2***. MobileNetV2 is a light-weight CNN architecture that was proposed targeting mobile and embedded vision applications. It aimed at striking a good balance between model size and performance. Among the striking features include depthwise separable convolutions, whereby the standard convolution is split into two separate layers: depthwise convolution and pointwise convolution. This separation causes a dramatic reduction in the computational cost while keeping the capability of the network for representation intact. MobileNetV2 further uses the concept of an inverted residual with linear bottlenecks for swiftness. The bottlenecks expand the number of channels, apply depthwise convolution, and then project the features back to a lower-dimensional space to reduce the computational costs.

2.2.3. ***ResNet50***. ResNet50 is a part of the ResNet family, which initially introduced skip connections or shortcuts; this allowed the gradients to flow more directly through the network. That helped to mitigate the vanishing-gradient problem so that very deep networks could be trained. It has 50 layers and is constructed by a series of residual blocks. Considering the residual block, each of the blocks includes two convolutional layers with batch normalization and ReLU activation. Each block adds the input to the output, before passing on the result through the activation function. This enables the network to learn residual functions—that is, those where the input is close to the output. This architecture enables deeper networks to be trained, since optimization becomes considerably easier.

2.2.4. ***ResNet101***. ResNet101 is an extension of ResNet50 with 101 layers. It follows the same basic principles as ResNet50 but with a deeper architecture, which can capture more complex features and patterns in the data. The additional layers in ResNet101 allow for more refined feature extraction, potentially leading to improved performance on challenging tasks. However, deeper networks also require more computational resources and may be more prone to overfitting, so careful tuning and regularization are essential.

2.2.5. ***VGG19***. VGG19 is a variant of the VGG (Visual Geometry Group) network, known for its simplicity and effectiveness. It consists of 19 layers and is characterized by its uniform architecture, with a stack of convolutional layers followed by max-pooling layers, culminating in fully connected layers for classification. VGG networks are praised for their easy-to-understand architecture and strong performance, especially in capturing fine details in images. However, VGG19 is deeper and has more parameters than earlier VGG models, which can make it computationally expensive and prone to overfitting, especially on smaller datasets.

3. EXPERIMENTAL STUDIES

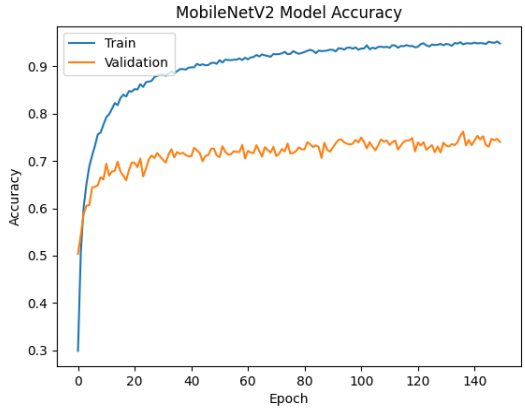


(A) Accuracy Plot (B) Loss Plot

FIGURE 3. **DenseNet121 Accuracy and Loss Plots.**

3.1. **DenseNet121.** The Figure 3 displays the training and validation accuracy and loss of a DenseNet121 model over 150 epochs. The training loss rapidly decreases in the first few epochs, from an initial value of approximately 2.5 to around 1.0 by epoch 10. It then continues to decrease gradually until it reaches a minimum of around 0.3 at epoch 150. Similarly, the validation loss also decreases in the first few epochs, from an initial value of around 2.0 to approximately 1.0 by epoch 10. However, after about 50 epochs, the value starts to increase again and reaches a maximum of approximately 1.8 at epoch 100. Subsequently, it decreases to around 1.5 at epoch 150. Optimization of the model was conducted utilizing the Adam optimizer with an initial learning rate of 0.001. The loss function applied was categorical cross-entropy, a method frequently employed in addressing multi-class classification problems.

3.2. **MobileNetV2.** Figure 4 shows the training and validation loss of a MobileNetV2 model over 150 epochs is displayed in the graph. Initially, the training loss decreases rapidly and then slows as it reaches a minimum value of approximately 0.3. The validation loss also drops quickly during the first few epochs but begins to increase after about 50 epochs, signaling potential overfitting. The provided details also

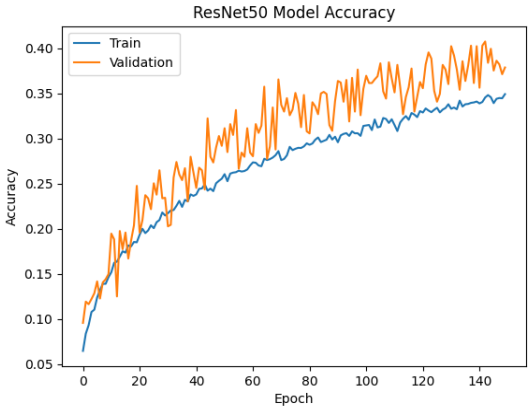


(A) Accuracy Plot

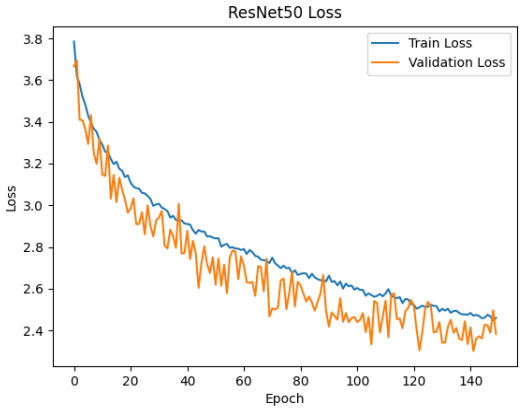
(B) Loss Plot

FIGURE 4. **MobileNetV2 Accuracy and Loss Plots.**

cover training and validation accuracy over the same period. Training accuracy starts at around 0.3 and quickly rises to about 0.95 by epoch 50. The validation accuracy begins similarly at 0.3 but climbs more slowly, reaching around 0.85 by epoch 150. The notable gap between training and validation accuracy suggests that the model is overfitting to the training data, excelling at recognizing training patterns but struggling to generalize to new data. The model was optimized using the Adam optimizer, initialized with a learning rate of 0.001. The categorical cross-entropy loss function, commonly used for multiclass classification tasks, was employed.



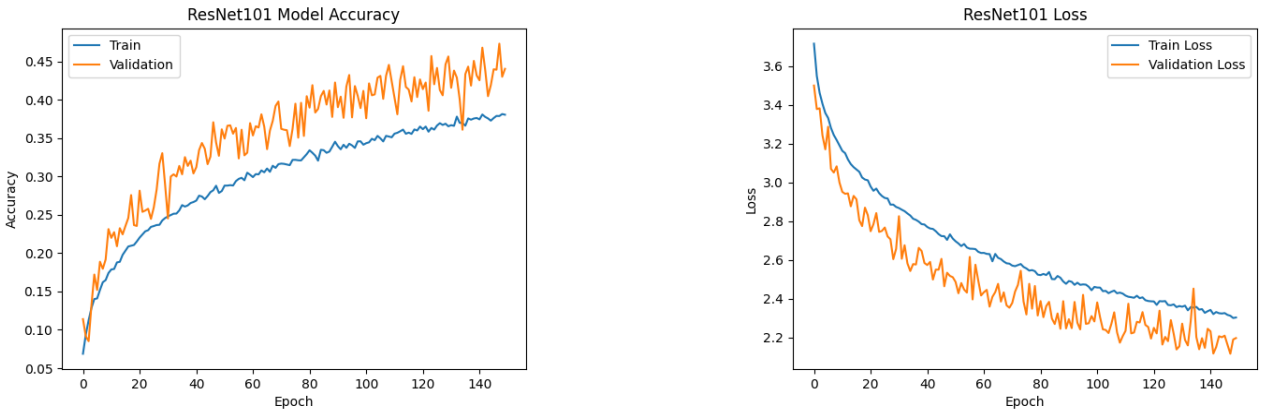
(A) Accuracy Plot



(B) Loss Plot

FIGURE 5. **ResNet50 Accuracy and Loss Plots.**

3.3. **ResNet50.** Figure 5 depicts the training and validation accuracy of ResNet50, after it has been trained for 150 epochs. The accuracy of the training starts at around 0.05 and goes all the way up to approximately 0.38. That of the validation also starts at approximately 0.05 but increases to around 0.35, though it sometimes fluctuated up and down in that process. Since the gap between training and validation accuracy is relatively small, overfitting does not happen in this model. This graph represents the training and verification loss for the ResNet50 model during 150 epochs. The training loss starts higher at approximately 3.8 and decreases linearly to about 2.4; also, the smooth validation loss starts from roughly 3.8 and goes down to about 2.4, with fluctuations during most of this training process. The relatively small difference between the training and verification loss suggests that this is not an overfitting model.



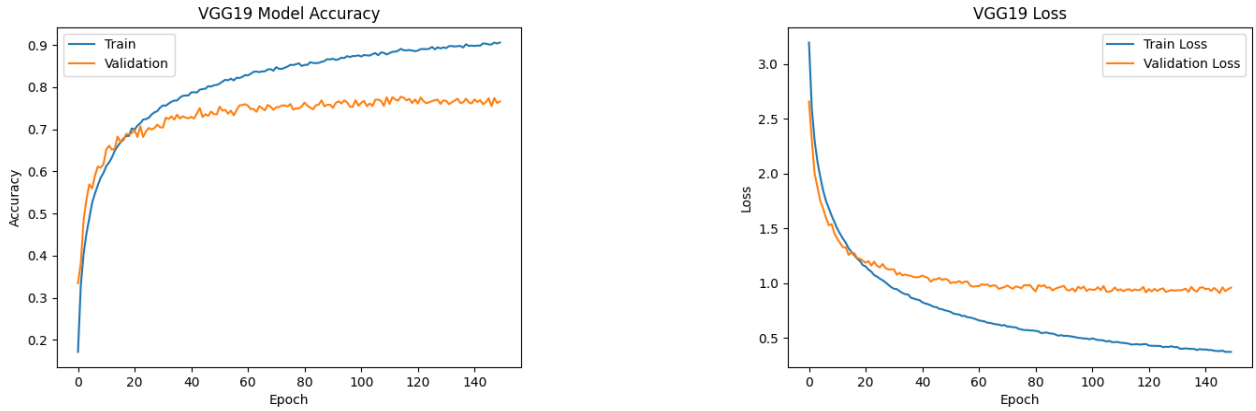
(A) Accuracy Plot (B) Loss Plot

FIGURE 6. **ResNet101 Accuracy and Loss Plots.**

3.4. **ResNet101.** These following graphs represent the accuracy and loss of the ResNet101 model concerning 150 epochs of training and validation in Figure 6. It is crystal clear that the training and validation accuracy both increase with time; however, the accuracy of training always outpaces that of validation. This would support the interpretation that the model overfits to the training dataset, learning the patterns in the training dataset perhaps too well and thereby limiting its eventual performance on new data. In similar fashion, while the training loss decreases steadily as time progresses, although validation loss is becoming less regular and skewed higher than training loss. That would mean the model has overfit to the training data. If improving the model's generalization ability, some techniques like regularization or data augmentation could be done.

3.5. **VGG19.** VGG19 was proposed by Simonyan and Zisserman in 2014 and also follows a deep CNN model architecture with stacks of convolution layers followed by fully connected layers. It is a very

simple yet effective network. VGG19 contains 19 layers, amounting to approximately 143.7 million parameters, and has achieved state-of-the-art performance for most image classification challenges.



(A) Accuracy Plot (B) Loss Plot

FIGURE 7. **VGG19 Accuracy and Loss Plots.**

Top graphic describes the accuracy curve, while the bottom graph depicts the loss curve, segregated for training and validation phases for VGG19 architecture in Figure 7.

It is observed from the accuracy figure that the model first increases in the initial epochs and then saturates at approximately 90 percent for the training set. Also, the validation accuracy increases, but it does so at a more gradual pace and stabilizes a little over 80 percent. That's good because it reflects appropriate generalization to unseen data.

This is reflected in the loss graph, wherein the training loss drops rapidly within the first few epochs before it stabilizes. The validation loss does decrease but at a much slower rate compared to the training loss, and it plateaus at a higher value; this is expected, as the validation set is not used for training.

In Figure 8 confusion matrix indicates that the model correctly classifies most images in the test set, although there are some misclassifications. There are instances where the model misclassifies images. For instance, some images of F-16s are misclassified as F-22s. However, this is a common occurrence with machine learning models. Overall, the results indicate that the VGG19 model can accurately classify images and generalize well to unseen data, making it a valuable deep learning model.

The confusion matrix reveals significant misclassifications between classes C130 and C17, with 14 occurrences of class C130 being erroneously labeled as C17 and 10 occurrences of class C17 being incorrectly identified as C130. This finding suggests that the model encounters difficulties in differentiating between these two classes, likely due to their visual similarities. A similar pattern is observed with classes F15 and F18, where 35 instances of F15 are misclassified as F18. This recurring misclassification suggests that these classes also possess visually similar attributes, posing challenges to the model's ability to accurately differentiate them.

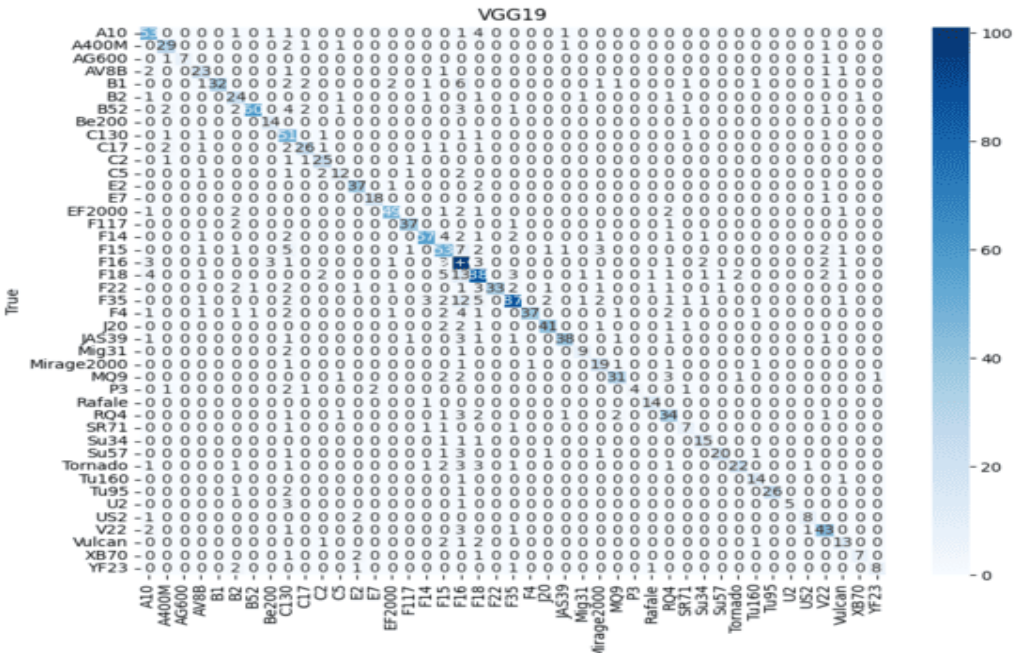


FIGURE 8. **Confusion matrix of VGG19.**

There is a notable level of misclassification between certain classes, such as F22 and F35, with observed values of 33 and 25, respectively. Though the model has some capacity to distinguish between these classes, a considerable overlap in their features leads to classification errors. Furthermore, the model incorrectly identifies Mig31 as Mig29 in 31 instances, illustrating the challenge of differentiating between similar classes. On the other hand, the model shows low misclassification rates for classes like A10 and A400M, and Tornado and Tu160, with values close to zero, indicating high accuracy for these presumably visually distinct classes. To enhance classification accuracy for often misclassified categories, several strategies can be employed. Increasing the size and diversity of training data for specific classes, such as C130, C17, F15, and F18, can improve the model's ability to distinguish among them. Moreover, employing data augmentation techniques, such as rotating, scaling, and flipping images, can help in creating a more robust training set.

3.6. **YOLO11.** The performance of the YOLO11x-cls model is evaluated in comparison with that of other YOLO11 classifier models, including YOLO11n-cls, YOLO11s-cls, YOLO11m-cls, and YOLO11l-cls. Table 2 of the paper presents a comprehensive comparison of these models based on key metrics, including accuracy, precision, recall, and F1-score.

The best performance is exhibited by the YOLO11x-cls model, which reaches class accuracy of 95.9, precision of 94.1, Recall of 95.6, and an F1-score as high as 94.8. Definitely the best results among all

TABLE 2. **YOLO11 models metrics**

Model	Accuracy	Precision	Recall	F1	Params (M)
YOLO11n-cls	0.919	0.900	0.908	0.904	1.6
YOLO11s-cls	0.935	0.918	0.933	0.925	5.5
YOLO11m-cls	0.950	0.937	0.941	0.939	10.4
YOLO11l-cls	0.955	0.936	0.952	0.944	12.9
YOLO11x-cls	**0.959**	**0.941**	**0.956**	**0.948**	**28.4**

four variants of YOLO 11. However, they are achieved at the expanse of a dramatic increase compared to the number of this model's parameters: 28.4 million, more than that of YOLO11L-cls (12.9M) by more than two times, and more than that one of YOLO 11n-cls in 17 times (1.6M). The same model, YOLO11x-cls, can serve as an example illustrative of tradeoffs between model complexity and achievements.

By comparing these models, YOLO11n-cls outperforms YOLO11scls with a moderate increase in network parameters, from 1.6M to 5.5M, yielding substantial precision improvement, from 91.9 to 93.5, and further improving the F1-score from 90.4 up to 92.5. A similar trend presents when going from YOLO11s-cls to more powerful YOLO11m-cls (increasing parameters to 10.4M), this further increases the accuracy to the level of 95.0 and F1 up to 93.9. However, this increase diminishes as the scaling factor increases. Considering YOLO11l-cls for example, it has a very limited increase of +0.4 in accuracy and +0.4 in F1-score compared to YOLO11x-cls, while the number of parameters increased 2.2 times.

Considering only the YOLO11 family of classifiers, the much larger YOLO11x-cls achieves state-of-the-art, but the two variants described here, the YOLO11m-cls and YOLO11l-cls, are offering top performance with considerably fewer parameters and hence may be excellent candidates for a practical realization. This clearly points towards at least further investigating some methodologies for optimizing such models—e.g., model pruning, quantization, knowledge distillation—for limited computing power contexts.

TABLE 3. **Military aircraft classification models metrics**

AI Model Used	**Classes**	**Methods**	**Accuracy**
Linear SVM [30]	20	CNN, data augmentation	96.8%
Artificial Neural Networks [31]	4	Sound signal processing, NN	96.2%
Feedforward NN [32]	4	Image processing, NN	97.0%
Signal Processing AI [33]	5	Radar signal feature extraction	95.0%
YOLO11x-cls **(Proposed)**	43	CNN, data augmentation	95.9%

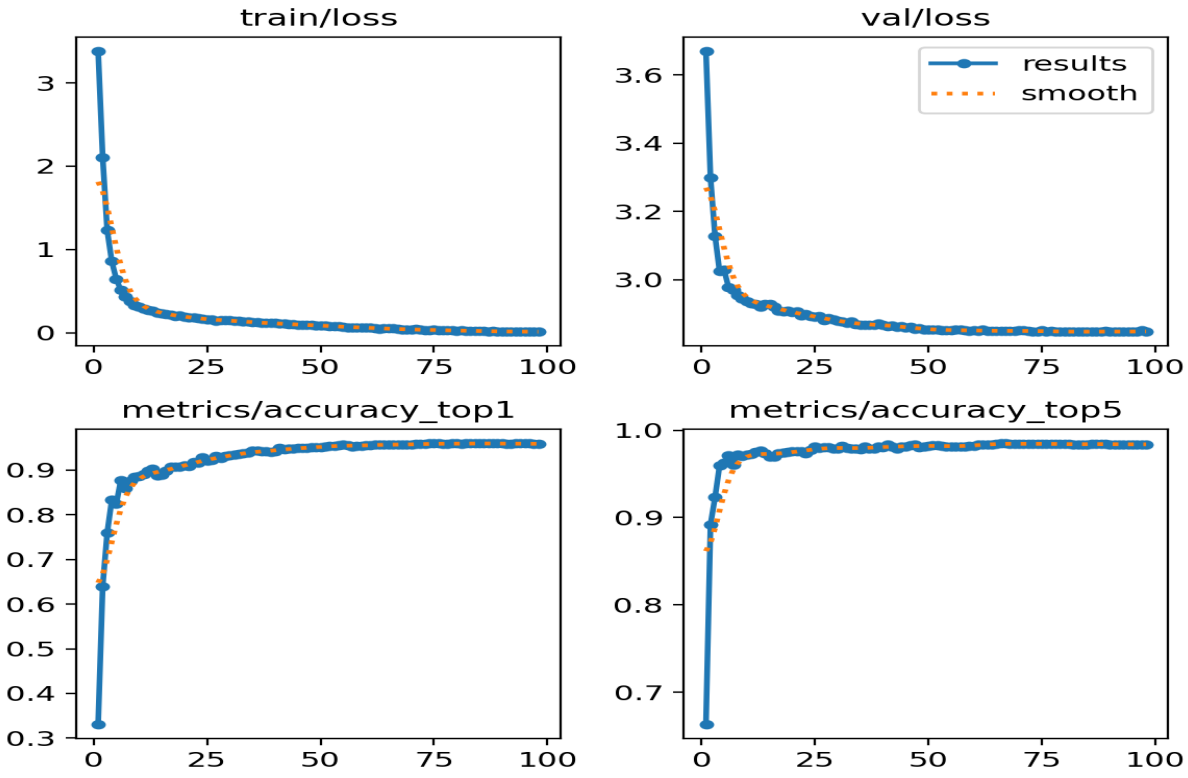


FIGURE 9. **YOLO11x-cls train metrics.**

As can be seen from the table, the YOLO11x-cls model outperforms all other models across all metrics. It exhibits the highest accuracy (0.959), precision (0.941), recall (0.956), and F1-score (0.948). These results indicate that the YOLO11x-cls model not only achieves high accuracy in classifying military aircraft but also demonstrates a remarkable ability to correctly identify positive instances (high recall) while minimizing false positives (high precision). The superior performance of the YOLO11x-cls model can be attributed to its larger size and complexity compared to other models. This enables the model to learn more complex features and patterns from the data, resulting in improved generalization and higher accuracy.

Figure 9 of the paper shows some training metrics for the YOLO11x-cls model. The focus of the above graph lies in the training loss and model accuracy over the epochs. It is observed that its training loss keeps decreasing step by step with growing epochs, a good signal for indicating this model learns well. More importantly, high training accuracy shows that the model learns from the training data and generalizes well. This would tend to suggest that the model is learning useful patterns in the training set and generalizing this by properly classifying military aircraft in the environment.

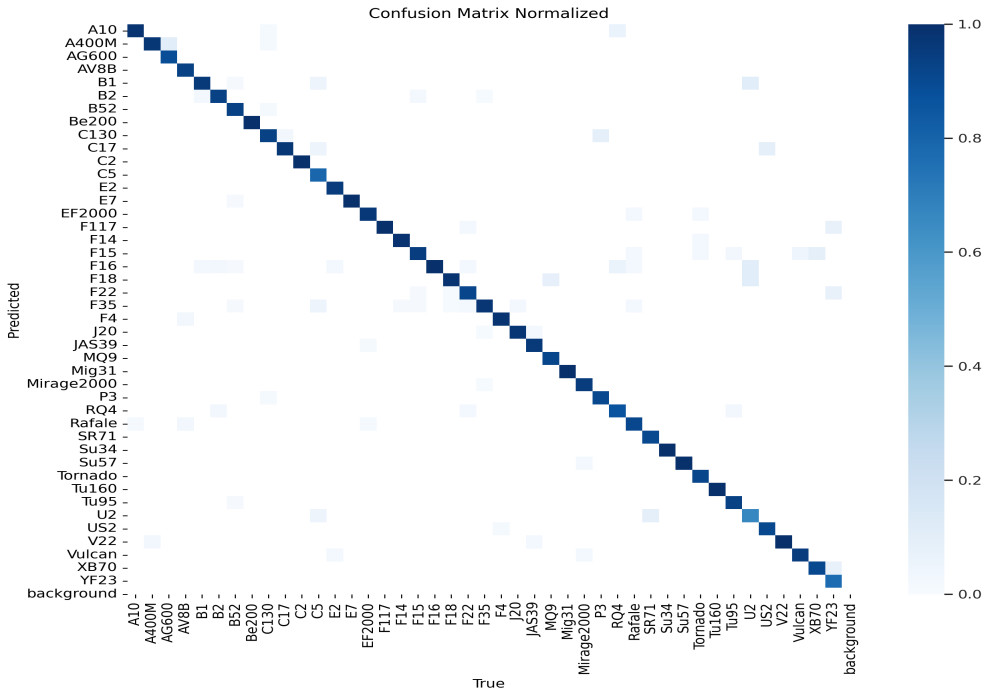


FIGURE 10. **Confusion metrics of YOLO11x-cls.**

Figure 10 is confusion matrix of the YOLO11x-cls model. From it, insight into the classification behavior of the model can be viewed. A confusion matrix depicts visually that the model has been able to classify most of the military aircraft in the test set correctly. It has, however, also revealed some misclassifications that are not out of the ordinary in any real-world machine learning application. A closer look into these misclassifications may hold a silver lining in the form of suggestions that could be obtained regarding the model's points of failure. For example, it could be very useful to study why some F-16 images were classified as F-22s with the aim of bringing improvement in the model's discrimination capability.

3.7. **Training Configuration for YOLO11.** YOLO11x classification was trained using the well-defined set of hyperparameters to have the best performance on the military airforce dataset. Training for 100 epochs, it has early stopping set with 10 epochs of patience to avoid overfitting by stopping when validation loss stops improving. Batch size 16 balances memory efficiency with gradient stability, and input image size 224x224 pixels was chosen for a good balance between computational efficiency and retaining enough spatial information. Optimization is guided by an initial learning rate (lr0) = 0.01 that is gradually decreased according to the learning rate factor (lrf) = 0.01. The momentum parameter is set to 0.937 for accelerated convergence. It stabilizes the gradient update, while weight decay equals 0.0005 and serves as a regularization factor against overfitting. During the first three epochs, the warmup policies

are used to smoothly increase the initial learning rate and momentum factor in favor of finding stability in training. Putting warmup momentum at 0.8 and warm up bias learning rate at 0.1 allows the model's bias to somewhat adapt more quickly.

The training regimen incorporates a suite of data augmentation techniques to enhance the robustness and generalization capacity of the classification model. Color jittering is applied through perturbations of hue, saturation, and value channels with respective magnitudes of 0.015, 0.7, and 0.4, introducing variations in image color characteristics. Geometric transformations, including translation up to 10% of the image dimensions and scaling by a factor up to 50%, are utilised, while rotation, shear, perspective, and vertical flips are intentionally deactivated. Horizontal flips are applied with a probability of 0.5. Mosaic augmentation, a technique that combines multiple images into a single training sample, is enabled.Furthermore, RandAugment is employed as an automated augmentation strategy to apply a diverse set of transformations. The training configuration is further augmented by the deliberate exclusion of random erasing (40% probability), cropping of the entire image during training, and the exclusion of mixup and copy-paste augmentations. These strategies are designed to diversify the training data, mitigate overfitting, and enhance the model's capacity for generalization to unseen data.

This work has designed an effective regularization that could balance the loss between the localization and classification tasks. The threshold for IoU is set to 0.7, and one can rest assured of getting high accuracy in overlap with the real results during the prediction stage. The box loss weighs 7.5; the classification loss is weighed at 0.5 while DFL takes 1.5-the loss weighs nicely manages the object detection and localization. Data augmentation involves a hue, saturation, and value adjustment in order to further generalize the model on hsv-h 0.015, hsv-s 0.7, and hsv-v is also 0.4. Additionally, the model does not use dropout, which was set to 0.0, and it is without label smoothing, while relying on weight decay and data augmentation for regularization. This combination of hyperparameters provides a robust and efficient training process, optimizing model generalization and convergence for the accurate classification of military aircraft.

3.8. **Classification Test on Real Data.** These testing results on your deep learning classifier illustrate that the model works excellent for aircraft type identification when the images are clear, frontal, or in ideal condition. The top-1 predictions have very high scores: 1.00 for F-16, C-130, and A-10, proving the strength of the model to pick out unique structural features like the configurations of wings, placement of engines, and shapes of fuselage. On challenging views of the aircraft, such as the F-22 at 0.59 confidence with alternative predictions of F-35 at 0.33, the model only shows slight ambiguity. While the classifier is robust in this matter, it struggles with classes that really do tend to look somewhat similar with the stealth factors of an aircraft taken into consideration. The meaningful secondary predictions in the top-5 results, such as F-18 with F-16 and Rafale, reflect the nuanced understanding of aircraft classes developed by the model but point to areas for improvement. To further improve the performance, more training on datasets that include diverse angles, lighting conditions, and occlusions would enhance the model's ability to tell apart visually similar aircraft.

This efficiency and applicability to real-world performance metrics further validate your classifier. The 11.8 ms/image preprocessing time just shows how well-optimized the pipeline in charge of preparing

(a) (b)

(c) (d)

FIGURE 11. **Performance evaluation of aircraft classification model: Top-5 predictions and confidence scores across diverse aircraft types [34] with processing metrics (Preprocess: 11.8 ms, Inference: 266.4 ms) on Dual Intel Xeon CPUs (2.20 GHz).**

the inputs before feeding into the model is. This gives a model inference of 266.4ms for the input shape [(1,3,224,224)] that becomes competitive in efficiency for any deep neural network and a rather complicated task of aircraft type recognition. Consequentially, the full processing time equals about 278.2 msec per image, as was noticed perfectly streamlined and fast end-to-end pipeline. This level of performance can be delivered by, but may not be limited to a hardware setup with 2 Intel Xeon CPUs-2.20 GHz since the system has multi-threading at the parallel processing level. Since this model has a very

FIGURE 12. **OCR output used without image preprocessing aircraft-1.**

high degree of confidence in addition with sub-second processing time will perform remarkably in real-life applications such as in military recognitions, auto surveillance and aircraft recognition systems. With further refinement, this classifier should be able to differentiate better between visually similar aircraft and optimize the inference time to perform well in complex and time-critical situations.

3.9. **Optical Character Recognition (OCR).** The presented image dataset demonstrates the application of OCR via the EasyOCR library on images of military aircraft tail numbers. An initial assessment shows that direct OCR application on the original images, which are characterised by varying lighting conditions and potentially low contrast, produces sub-optimal results with low confidence scores or inaccurate character recognition. This is due to the inherent challenges that OCR systems face when processing images that lack sharp transitions and distinct features. The presence of noise, blurred text and inconsistent lighting also contributes to the OCR's struggle to accurately decipher the alphanumeric sequences that make up the tail numbers. The result is an unacceptable level of recognition, indicating the need for pre-processing techniques.

The presented image dataset demonstrates the application of OCR via the EasyOCR on images of military aircraft tail numbers. An initial assessment shows that direct OCR application on the original images [35], which are characterised by varying lighting conditions and potentially low contrast, produces sub-optimal results with low confidence scores or inaccurate character recognition. This is due to the inherent challenges that OCR systems face when processing images that lack sharp transitions and distinct features. The presence of noise, blurred text and inconsistent lighting also contributes to the OCR's struggle to accurately decipher the alphanumeric sequences that make up the tail numbers. The result is an unacceptable level of recognition, indicating the need for pre-processing techniques.

FIGURE 13. **OCR result after histogram equalisation aircraft-1.**

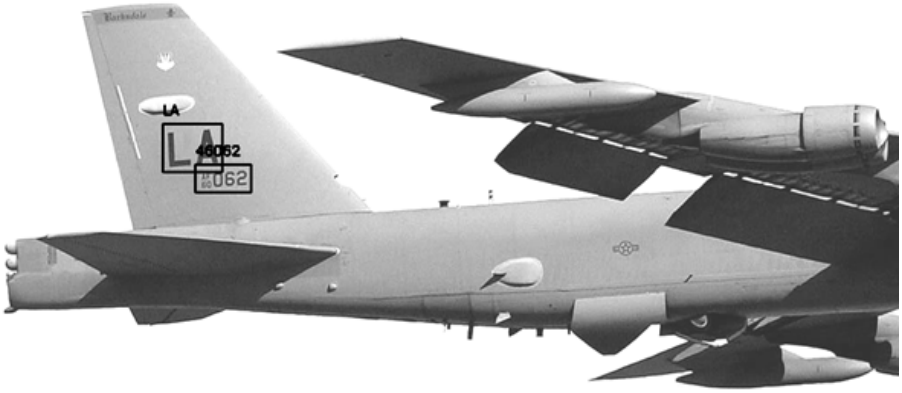


FIGURE 14. **OCR result of the image after histogram equalisation and contrast boosting process aircraft-1.**

To overcome the limitations of direct OCR, a pre-processing step using histogram equalisation is introduced. Histogram equalisation increases the contrast of the image by redistributing pixel intensities, effectively stretching the dynamic range of grey levels within the image. This adjustment significantly

FIGURE 15. **OCR output used without image preprocessing aircraft-2.**

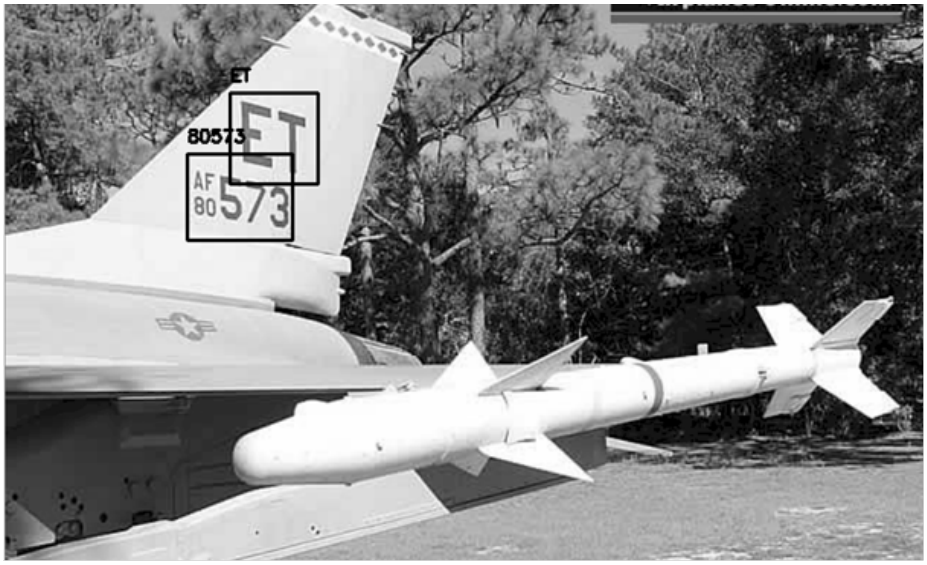


FIGURE 16. **OCR result of the image after histogram equalisation followed by contrast boosting aircraft-2.**

increases the contrast between the tail number characters and the background, improving the visual distinction of the text. After histogram equalisation, EasyOCR's performance shows a marked improvement, providing accurate character recognition with increased confidence scores. This confirms that

pre-processing techniques such as histogram equalisation play a key role in optimising the OCR process, particularly for images with difficult contrast or lighting conditions, and underlines the importance of pre-processing in text recognition applications.

4. Conclusion

This study focuses on the fine-grained classification of military aircraft using deep learning models that are pre-trained, focusing on uniquely identifying tail numbers correctly. Five CNN architectures were compared: DenseNet121, MobileNetV2, ResNet50, ResNet101, and VGG19. Whereas all the models built had a degree of success, the best performing family was the YOLO11 family, with a high accuracy achieved of 0.959 by YOLO11x-cls. The precision, recall, and F1-score were all superior with this model; hence, this classifier had great capability in recognizing the aircraft and generalizing to unseen data. This probably is due to the YOLO11x-cls being bigger in size and more complicated; hence, it can learn far more complicated features and patterns from data. The consistent decrease in training loss of the model and the high training accuracy are a witness to how effectively it gets to learn from the training data and does the right thing in classifying the aircraft correctly. More evidence can be seen with regard to the performance of the model from the confusion matrix; it performs high in classifying most of the aircraft in the test set.

In future investigations, the dataset will be greatly expanded to encompass a more diverse range of aircraft categories. The integration of Optical Character Recognition (OCR) techniques will enable the automatic extraction of tail numbers from images, thereby improving both the precision and efficiency of data processing. The research will also evaluate various pre-trained models to determine the most effective options for this task, potentially enhancing both performance and accuracy. These initiatives are essential for advancing the core technology and ensuring the solution's capability to manage a wider array of aircraft identification cases.

Moreover, the research will explore the adaptation of these models for real-time use cases, like video surveillance systems. This entails evaluating their performance in real-time scenarios and optimizing them for ongoing monitoring and swift data processing. The primary objective of this investigation is to create a highly reliable AI-driven image recognition solution specifically designed for military aviation purposes. Such advancements are anticipated to greatly enhance processes of data collection, monitoring, and analysis, ultimately bolstering national defense and security capabilities. The outcomes of these studies will form the foundation for developing a thorough and efficient system capable of meeting the stringent requirements of military operations.

Declarations

- **Contribution Rate Statement:** Hasan KARACA has conducted the study and wrote the first draft, Nesrin AYDIN ATASOY has supervised, reviewed and edited the manuscript.
- **Conflict of Interest:** The authors report no declarations of interest.
- **Data Availability:** Dataset is available online.
- **Statement of Support and Acknowledgment:** None.

REFERENCES

[1] Mori, S., H. Nishida, and H. Yamada, Optical character recognition. 1999: John Wiley & Sons, Inc.

[2] Mekonnen, I., Automated Aircraft Identification by Machine Vision. 2017.

[3] Tomovic, S., K. Pavlovic, and M. Bajceta, Aligning document layouts extracted with different OCR engines with clustering approach. Egyptian Informatics Journal, 2021. 22(3): p. 329-338.

[4] Kobayashi, Y., et al., Basic research on a handwritten note image recognition system that combines two OCRs. Procedia Computer Science, 2021. 192: p. 2596-2605.

[5] Zeng, G., et al., Beyond OCR + VQA: Towards end-to-end reading and reasoning for robust and accurate textvqa. Pattern Recognition, 2023. 138: p. 109337.

[6] Onim, M.S.H., et al., BLPnet: A new DNN model and Bengali OCR engine for Automatic Licence Plate Recognition. Array, 2022. 15: p. 100244.

[7] Lv, G., et al., COME: Clip-OCR and Master ObjEct for text image captioning. Image and Vision Computing, 2023. 136: p. 104751.

[8] Imam, N.H., V.G. Vassilakis, and D. Kolovos, OCR post-correction for detecting adversarial text images. Journal of Information Security and Applications, 2022. 66: p. 103170.

[9] Irimia, C., et al., Official Document Identification and Data Extraction using Templates and OCR. Procedia Computer Science, 2022. 207: p. 1571-1580.

[10] Dutta, H. and A. Gupta, PNRank: Unsupervised ranking of person name entities from noisy OCR text. Decision Support Systems, 2022. 152: p. 113662.

[11] Oucheikh, R., T. Pettersson, and T. Löfström, Product verification using OCR classification and Mondrian conformal prediction. Expert Systems with Applications, 2022. 188: p. 115942.

[12] Mei, J., et al., Statistical learning for OCR error correction. Information Processing & Management, 2018. 54(6): p. 874-887.

[13] Shen, Z., et al. Deep learning based framework for automatic damage detection in aircraft engine borescope inspection. in 2019 International Conference on Computing, Networking and Communications (ICNC). 2019. IEEE.

[14] Sun, X., et al., SCAN: Scattering characteristics analysis network for few-shot aircraft classification in high-resolution SAR images. IEEE Transactions on Geoscience and Remote Sensing, 2022. 60: p. 1-17.

[15] Kiyak, E. and G. Unal, Small aircraft detection using deep learning. Aircraft Engineering and Aerospace Technology, 2021. 93(4): p. 671-681.

[16] Khan, S.N., et al. Rapid Aircraft Classification in Satellite Imagery using Fully Convolutional Residual Network. in 2020 International Conference on Emerging Trends in Smart Technologies (ICETST). 2020. IEEE.

[17] Kang, Y., et al., ST-Net: Scattering Topology Network for Aircraft Classification in High-Resolution SAR Images. IEEE Transactions on Geoscience and Remote Sensing, 2023. 61: p. 1-17.

[18] Hassan, A., et al. A deep learning framework for automatic airplane detection in remote sensing satellite images. in 2019 IEEE Aerospace Conference. 2019. IEEE.

[19] Dolph, C., et al. Aircraft Classification Using RADAR from small Unmanned Aerial Systems for Scalable Traffic Management Emergency Response Operations. in AIAA AVIATION 2021 FORUM. 2021.

[20] Chen, Z., T. Zhang, and C. Ouyang, End-to-end airplane detection using transfer learning in remote sensing images. Remote Sensing, 2018. 10(1): p. 139.

[21] Azam, F., et al., Aircraft classification based on PCA and feature fusion techniques in convolutional neural network. IEEE Access, 2021. 9: p. 161683-161694.

[22] Alshaibani, W., et al., Airplane Type Identification Based on Mask RCNN and Drone Images. arXiv preprint arXiv:2108.12811, 2021.

[23] Alganci, U., M. Soydas, and E. Sertel, Comparative research on deep learning approaches for airplane detection from very high-resolution satellite images. Remote sensing, 2020. 12(3): p. 458.

[24] LeCun, Y., Y. Bengio, and G. Hinton, Deep learning. nature, 2015. 521(7553): p. 436-444.
[25] Gao, Z., & Yi, W. (2025). Optimizing projectile aerodynamic parameter identification of kernel extreme learning machine based on improved Dung Beetle Optimizer algorithm. Measurement, 239, 115473.
[26] Song, T., Nguyen, L. T. H., & Ta, T. V. (2025). MPSA-DenseNet: A novel deep learning model for English accent classification. Computer Speech & Language, 89, 101676.
[27] Zhang, Y., Liu, R., Wang, X., Chen, H., & Li, C. (2021). Boosted binary Harris hawks optimizer and feature selection. Engineering with Computers, 37, 3741-3770.
[28] Prakash, N. N., Rajesh, V., Namakhwa, D. L., Pande, S. D., & Ahammad, S. H. (2023). A DenseNet CNN-based liver lesion prediction and classification for future medical diagnosis. Scientific African, 20, e01629.
[29] Data Statement Dataset is available at `https://www.kaggle.com/datasets/a2015003713/militaryaircraftdetectiondataset`
[30] Azam, F., Rizvi, A., Khan, W. Z., Aalsalem, M. Y., Yu, H., Zikria, Y. B. (2021). Aircraft classification based on PCA and feature fusion techniques in convolutional neural network. *IEEE Access*, *9*, 161683-161694.
[31] Barbarosou, M., Paraskevas, I., Ahmed, A. (2016). Military aircrafts' classification based on their sound signature. *Aircraft Engineering and Aerospace Technology: An International Journal*, *88*(1), 66-72.
[32] Karacor, A. G., Torun, E., Abay, R. (2011). Aircraft classification using image processing techniques and artificial neural networks. *International Journal of Pattern Recognition and Artificial Intelligence*, *25*(08), 1321-1335.
[33] Luo, S., Yu, J., Xi, Y., Liao, X. (2022). Aircraft target detection in remote sensing images based on improved YOLOv5. *IEEE Access*, *10*, 5184-5192.
[34] Fine-Grained Visual Classification of Aircraft, S. Maji, J. Kannala, E. Rahtu, M. Blaschko, A. Vedaldi, arXiv.org, 2013
[35] `https://www.airplanes-online.com/`