NÖHÜ Müh. Bilim. Derg. / NOHU J. Eng. Sci., 2025; 14(1), 340-349

Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi
Niğde Ömer Halisdemir University Journal of Engineering Sciences

Araştırma makalesi / Research article

www.dergipark.org.tr/tr/pub/ngumuh / www.dergipark.org.tr/en/pub/ngumuh

Deep learning classification and object detection in helicopter images: Performance analysis of GoogleNet, AlexNet and YOLOv9c architectures

Helikopter görüntülerinde derin öğrenme ile sınıflandırma ve nesne tespiti: GoogleNet, AlexNet ve YOLOv9c mimarilerinin performans analizi

İrem Hatice Doğan[1], Ozan Arslan[2], Ayşe Betül Tat[3], Burhan Şahin[4], Ege Erberk Uslu[5,*], İbrahim Yülüce[6], Orhan Dağdeviren[7]

[1,2,3] *İnönü University, Computer Engineering Department, 44280, Malatya, Türkiye*
[4] *Turkish Aerospace, 06980, Ankara, Türkiye*
[5,6,7] *Ege University, Computer Engineering Department, 35100, İzmir, Türkiye*

Abstract

Helicopter imaging classification and detection are crucial for autonomous navigation, military operations, search and rescue, and civil aviation management. This study utilized two helicopter image datasets, applying data augmentation techniques such as random resizing, cutting, horizontal rotation, rotation, and color adjustments, along with histogram equalization for contrast enhancement. Twenty-four helicopter classes were trained using GoogleNet and AlexNet architectures, while the YOLOv9c model was employed for object detection. The results revealed that the GoogleNet classification model achieved an 81% F1 score, and AlexNet reached 73%. In contrast, the YOLOv9c model demonstrated an average mean Average Precision (mAP) of 87%. These findings indicate that CNN architectures and YOLO are effective for helicopter image classification and detection, highlighting their potential applications in military, search and rescue, and civil aviation contexts.

Keywords: Helicopter Image classification, Object detection, Convolutional neural networks, Data augmentation

Öz

Helikopter görüntülerinin sınıflandırılması ve tespiti, otonom navigasyon sistemlerinin, askeri operasyonların, arama kurtarma görevlerinin ve sivil havacılık yönetiminin önemli bileşenlerindendir. Bu çalışmada helikopter görüntüleri için iki farklı veri seti kullanılmıştır. Sınıflandırma için rastgele yeniden boyutlandırma, kesme, yatay döndürme, döndürme ve renk değişiklikleri gibi veri artırma teknikleri uygulanmıştır. Görüntülerin kontrastı da histogram eşitleme yöntemi kullanılarak yeniden düzenlenmiştir. 24 sınıf helikopter veri seti GoogleNet ve AlexNet mimarileri kullanılarak eğitilmiştir. Nesne tespiti için YOLOv9c mimarisi kullanılmıştır. Deneysel sonuçlar, GoogleNet tabanlı sınıflandırma modelinin test setinde %81 F-1 skoru elde ettiğini ve AlexNet modelinde genel F1 skorunun %73 olduğunu göstermektedir. YOLOv9c modeli ise ortalama %87 mAP oranları elde etmiştir. Bu sonuçlar, bir tür derin öğrenme modeli olan CNN mimarilerinin ve YOLO nesne tespitinin helikopter görüntüsü sınıflandırma ve tespitinde iyi olduğunu gösteriyor. Çalışma, helikopter ve bileşenlerini tespit etme ve sınıflandırmada iyi performans gösteren modellerin askeri, arama kurtarma ve sivil havacılık dahil olmak üzere çeşitli alanlarda kullanılabileceğini göstermiştir.

Anahtar Kelimeler: Helikopter Görüntü sınıflandırma, Nesne algılama, Evrişimli sinir ağları, Veri arttırma

1 Introduction

Rescue missions, autonomous navigation systems, civil aviation management, aircraft identification, and classification and detection of helicopter images are crucial. Flying activities, particularly military, rescue, and surveillance missions, are greatly enhanced by implementing robust classification systems, which also increase safety. Helicopters bring some challenges for image classification due to their different designs, structural differences, and various operational environments and conditions. Varying observation angles and other lighting conditions and background clutter increase these challenges. Advanced image classification algorithms have try to solve these challenges.

Significant advances in image classification [1,2] and prediction [3,4] have occurred in recent years. Developing Convolutional Neural Networks (CNNs) and refining transfer learning techniques are leading these advances. CNNs is a suitable architecture for image classification. This is because it automatically learns hierarchical features from

* Sorumlu yazar / Corresponding author, e-posta / e-mail ege.erberk.uslu@ege.edu.tr (E.E. Uslu)
Geliş / Recieved: 29.09.2024 Kabul / Accepted: 26.12.2024 Yayımlanma / Published: 15.01.2025
doi: 10.28948/ngumuh.1556995

raw pixel data. Due to this feature, CNNs are highly ideal for the helicopter classification task.

Finding and recognizing helicopters during military operations relies heavily on object detection. The You Only Look Once (YOLO) architecture has taken the lead in object recognition due to its accuracy and speed [5]. Thanks to its integrated design, which computes bounding boxes and class probabilities from full images in a single evaluation, it is well-suited for helicopter and structure recognition. By integrating YOLO object detection with CNN classification, we can detect helicopter components and identify different types of helicopters. This research provides a deep-learning strategy for component recognition and classification in helicopter images.

We created two separate datasets for object recognition and classification using images of helicopters. To minimize the chance of overfitting, we used several data augmentation methods to the dataset of helicopter images we created for the classification assignment. These augmentations include rotation, color changes, random resizing cropping, and horizontal flipping. These operations enlarge the training dataset and enhance the model's capacity for generalization. The enlarged dataset is split 70-20-10% into training, validation, and test sets, respectively, to thoroughly assess the model's functionality. In addition, we tailored a smaller dataset to the object detection job. We did not use data augmentation for this work.

The foundation of the classification model is GoogleNet [6], which has emerged as a frontrunner due to its efficiency and excellent performance in picture recognition tasks. The pre-trained GoogleNet model was fine-tuned and trained on the 24-class helicopter dataset we developed using transfer learning. In addition, the YOLO architecture was used for the object detection task by fine-tuning on the other developed dataset.

As an classifier, GoogleNet was our top pick. Everything from batch size and epoch number to learning rate and patience time to early stopping is detailed. Step, weight decay, and learning rate were tweaked with the help of the LR planner. In order to train the model, we used weight decay in conjunction with the Adam optimization technique. The 24-class classification matrix, which includes metrics for recall, accuracy, and F1-score for each helicopter class, is employed to evaluate the classification model's efficacy. The classification results obtained serve as evidence of the model's ability to distinguish between helicopter models. The model's 81% overall accuracy on the test set is a dependable indicator of its performance, as demonstrated by mathematical analysis.

The second model that was implemented was AlexNet [7]. Researchers and programmers have implemented them extensively since their introduction in 2012. This eight-layer model served as the initial illustration of the capacity of deep neural networks to identify images. Subsequent iterations have been significantly influenced by its architectural principles, which have since become industry standards. Many of the methods used in this approach, such data augmentation and dropout, are now standard procedures in neural network training. There are 21,680,216 parameters in total for AlexNet based on our data. Adam optimizer is used as Alexnet's optimizer. Also Adam was selected since it accounts for both momentum and Root Mean Square Propagation (RMSprop) . The learning rate variable was chosen as 0.0001. Although choosing a lower value slightly improves the performance, it significantly increases the running time, so this ratio is ideally preferred. Like the previous algorithm, the model runs on a 24-class dataset. Both models have a runtime of 50 epochs.

For object recognition, the YOLOv9c [8] model was used. The model was not altered in any way. Successful findings were achieved when YOLOv9c's fine-tuning performance was assessed on a test set of helicopter images. The fine-tuned model demonstrated adequate accuracy in identifying different aircraft parts. For each class, mean accuracy (mAP), recall, and overall accuracy scores were noted, and assessments were based on these data. Specifically, the model scored 96.60% accuracy, 97.70% recall, and 98.30% mAP for the "helicopter" class, which refers to the location detection of the helicopter. The model also demonstrated robust performance metrics for the landing bar, cockpit, tail, and propeller, among other components. The model's exceptional resilience in these numerous component identification tasks distinguishes it.

The proposed research represents the whole methodology that considerably enhances the helicopter image classification. In particular, we make use of a convolution neural network referred to as CNN, which has gained wide reputation for image processing. Using transfer learning, we can plug into those models that have already learned essential features from huge sets of data and raise the accuracy and efficiency of our work in classification. We have also tried different methods in order to enhance our dataset. Data augmentation is a used for creating modified versions of already existing images by means of rotation, scaling, and flipping. This increases diversity within the training data, hence helping to avoid overfitting and improving the capability of the model for better generalization on test images. Also, our research does not stop at mere categorization. We furthered the key job of object detection using the YOLOv9c model with the help of our dataset. YOLO is utilized for object detection in real time. This can detect lots of items in one image very fast and precisely. By fine-tuning the YOLOv9c model on our dataset, this makes the model much more cognizant and observant of those objects important in the research.

Following the introduction, the paper is organized as follows: Section 2 reviews previous work on the topic. Section 3 provides a detailed account of the data collection process, including the sources of the data, the criteria for data selection, and any preprocessing steps. Section 4 describes the methodologies used. Section 5 presents the model architectures used in the study and Section 6 describes the data training process. Section 7 contains the model evaluation results. Finally, Section 8 summarizes the conclusions of the study and discusses potential future research directions

2 Related works

The first paper we are going to review is a paper written in 2020. This paper introduces remote sensing imagery in relation to a machine recognition application for aircraft type recognition, a new benchmark dataset called MTARSI. The dataset contains more than 9,000 images of 20 aircraft types with various characteristics such as complex backgrounds with different spatial resolutions, various poses, positions, illuminations and time periods. The authors argue that existing datasets used for aircraft detection are not suitable for remote sensing imagery and that previous studies have been applied with different datasets and settings, making it difficult to make comparisons. MTARSI aims to provide researchers with a standardized benchmark for developing and evaluating aircraft type recognition algorithms. They try to classify images with 10 different algorithms. As a result of this classification, the best algorithm for their dataset was EfficientNet with a success score of 89.79%. Other well-known and frequently used algorithms Resnet and GoogleNet achieved 89.61% and 86.53% respectively [9].

In the second paper, authors proposed a novel aircraft recognition scheme that utilizes a modular extreme learning machine (ELM) classifier. The scheme extracts three types of moment invariants (Hu, Zernike and Wavelet) as features from aircraft images and uses them as input to three separate modular neural networks. Each modular network consists of multiple single hidden layer feed-forward networks trained on different clustered data subsets using ELM. The final classification output is obtained by combining the outputs of each modular network according to their weighted sum. The scheme is evaluated on six different aircraft models and achieves a higher recognition accuracy compared to single ELM classifiers and other classification algorithms [10].

In this work, they developed SFSA (Scatter Features Spatial-Structural Association Network), a new deep learning model for aircraft recognition in SAR (synthetic aperture radar) images. It combines information from electromagnetic scattering features and image space features to improve recognition accuracy. It extracts strong scattering points (SSPs) from aircraft images and models their spatial relationships as a graph. A Graph Convolution Network (GCN) is used to extract structural features from this graph, and a modified Visual Geometry Group Network (VGGNet) is applied to retrieve image space features. Finally, it combines these two types of features to provide better recognition performance. The effectiveness of the proposed method has been validated on a SAR aircraft dataset [11].

Wang Y. et al. propose a hybrid attention network model (BA-CNN) for aircraft image recognition. This model utilizes a two-channel ResNet34 architecture for feature extraction and enhances its ability to capture fine details in aircraft images. The model integrates a hybrid attention mechanism combining channel and spatial attention modules. This mechanism enables the model to focus on important features in both channel and spatial dimensions, strengthening its discrimination capability. The BA-CNN model is trained and tested on the Fine-Grained Visual Classification of Aircraft (FGVC-Aircraft) dataset for fine-grained image classification. The results show that the BA-CNN model achieves 89.2% recognition accuracy, outperforming other state-of-the-art methods. The hybrid attention mechanism proved effective in improving the model's discriminative feature learning and recognition accuracy [12].

Using CNNs and a generalized Multiple Instance Learning (MIL) framework, a novel approach for detailed aircraft detection in remote sensing images is proposed in another recent study. The main difficulty with comprehensive identification is that it is sometimes necessary to provide thorough component descriptions in order to detect minute variations within subcategories. With this approach, an airplane is seen as a bag of samples (head, tail, and wing, for example), and the goal is to create a model that can identify different kinds of aircraft only by looking at these components. Explicit part descriptions are not necessary. A pre-trained CNN serves as the feature extractor in this technique, along with components for instance transformation and MIL pooling. The CNN's patch-level features are transformed into instance-level features via the instance transformation component, which improves the model's capacity to identify distinguishing elements. By averaging the sample-level scores, the MIL pooling component makes a prediction on the kind of aircraft. In comparison to conventional CNN-based techniques and conventional MIL networks, experiments conducted on a combined aircraft dataset demonstrate that our method increases accuracy and lowers computing costs. The aircraft's unique areas are efficiently targeted by the generalized MIL framework, which suppresses background noise and captures diverse properties of different subcategories [13].

For various tasks involving object recognition and categorization, the works provided in [14-17] may be examined.

3 Data collection

As a first step in the data collection procedure, we sought reliable sources. In this context, leading websites were preferred and helicopter data were collected and processed using a range of technological tools and, importantly, scientific techniques. The Selenium library was used to collect and automate the information on these websites. To handle the dynamic structures of the websites and to analyze complex website architectures, we developed and used customized algorithms in the study. These algorithms revealed the visual data of the helicopters. We applied extensive error checking procedures and validation techniques to ensure the consistency and integrity of the visual data. Visual data that did not meet the quality requirements were not included in the dataset. As a result of these procedures, we obtained comprehensive and accurate information about the helicopters. The visual data of the helicopter components were labeled using the Roboflow platform and the helicopter class data were labeled using file hierarchy logic. All of our data was prepared in accordance with our algorithms, and we aimed to make it possible to achieve high performance.

4 Methodology

For helicopter classification, AlexNet and GoogleNet models were preferred. AlexNet provides fast training and inference thanks to its simple and efficient architecture and builds a solid foundation with its basic feature extraction capabilities. GoogleNet, on the other hand, offers more advanced feature extraction and high classification accuracy thanks to its deep architecture and inception modules. The simplicity and depth offered by AlexNet and GoogleNet provide the required performance in classifying helicopter images. In this study, two separate datasets were used for these two different tasks. While the YOLOv9c model is used for object detection, the GoogleNet and AlexNet models ared used for classification. In order to compare the performance of the models, detailed comparisons were made.

4.1 Datasets

In this study, two datasets were prepared. The first dataset was created for the YOLO model, while the other dataset was created for the models we will use.

4.1.1 YOLOv9c dataset

The dataset used with the YOLOv9c model consists of various images of helicopter components and is divided into five different classes: helicopter, landing bar, cockpit, tail and propeller. There are 29 images for each class, totaling 145 image samples. The images were resized to 640x640 pixels to fit the input size of the model. This was done to ensure that the model can perform consistent feature extraction in each image.

4.1.2 Model dataset

The dataset used for our models consists of a larger image archive. The dataset includes 24 different helicopter models and consists of 2494 images in total. The dataset is divided into three parts: 70% for training, 20% for validation and 10% for testing. Out of 2494 image data, 1745 images were used in the training set, 498 in the validation set and 251 in the test set. The analysis of the dataset used for the models was analyzed as class distribution, image size distribution and image proportion distribution.

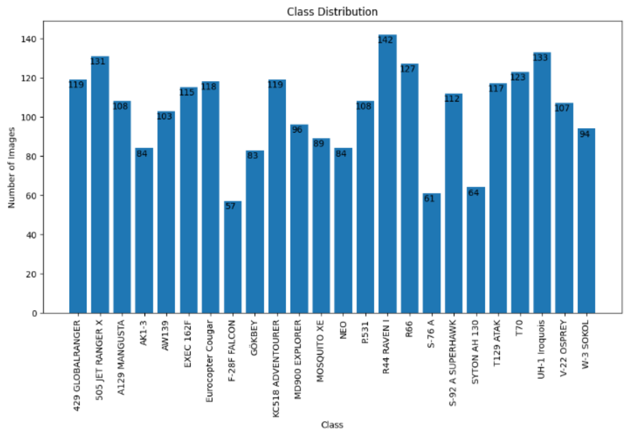

Figure 1. Helicopter class distribution

Figure 1 shows the number of images for each helicopter class. The number of images belonging to each class in the dataset is quite close to each other. A balanced distribution among the classes in the dataset increases the capacity of the model to learn each class. However, some classes have a higher number of images than others, as seen in R44-RAVEN-1 and UH-1 Iroquouis. This may cause the model to learn these classes better. In addition, the number of images of the F-28F FALCON helicopter is quite low compared to the other types, but the model's performance may decrease in classes with a small number of images. This imbalance can be overcome by using data augmentation techniques for classes with few images.

Figure 2 shows the width and height distribution of the images in the dataset. There is a wide range of image sizes. However, most of them are concentrated around 500 pixels. This indicates that the images need to be resized to standard sizes. This will make them suitable for the input size of the GoogleNet model to be used.

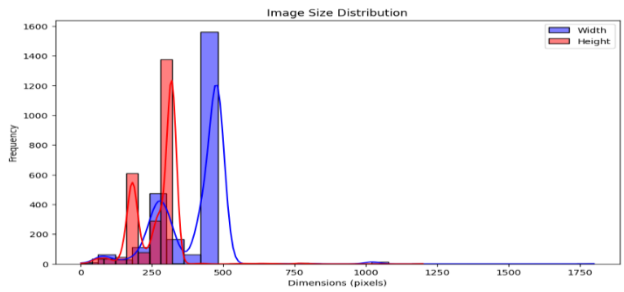

Figure 2. Image sizes distribution

The 2,494 valid images in the dataset, which cover a wide range of dimensions, are summarized in Table 1 and Figure 3. The average picture width of around 399.06 pixels and height of 268.06 pixels show that most images have a rather low resolution. Yet, the dataset does exhibit notable variety, suggesting the presence of a range of image sizes, with standard deviations of 122.46 pixels for width and 79.20 pixels for height. High-resolution samples have the largest size, 1800x1200 pixels, whereas defective or improperly processed images have the lowest dimensions, 1x1 pixels. The interquartile range, which includes widths of 299–480 pixels and heights of 189–320 pixels, reflects the size of most of the dataset. To ensure consistency and applicability for machine learning applications, these data highlight the need of preprocessing steps like scaling.

Table 1. Image Dimensions Summary

Feature	Width	Height
Count	2494.00	2494.00
Mean	399.06	268.06
Std	122.46	79.20
Min	1.00	1.00
25%	299.00	189.00
50%	464.00	310.00
75%	480.00	320.00
Max	1800.00	1200.00

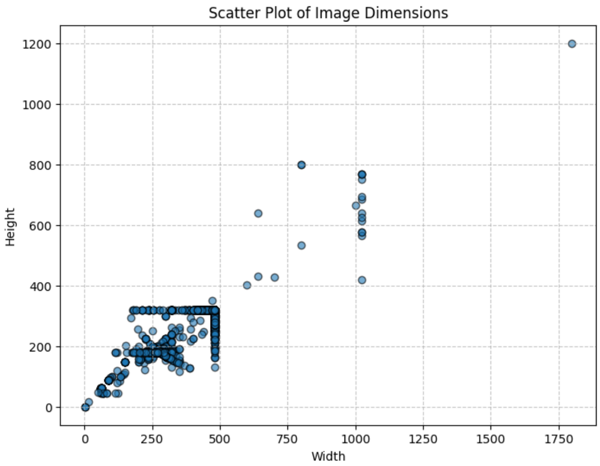

Figure 3. Image dimension analysis

Histogram equalization was applied to this dataset to address significant variations in brightness and contrast, as illustrated in Table 2. The dataset shows an average brightness of 144.13 and a contrast of 57.68, reflecting moderate levels overall. The wide range of values—from complete darkness (minimum brightness of 0) and flat images (minimum contrast of 0) to fully saturated (maximum brightness of 255) and highly dynamic images (maximum contrast of 106.76)—highlights inconsistencies that could negatively affect subsequent tasks like classification or object detection. The extremes in brightness and contrast indicate that certain images may demonstrate poor visibility or insufficient intensity variation to distinguish features effectively. The analysis helps us to better discover the characteristics of the our dataset and optimize training process of the model.

Table 2. Brightness and Contrast Summary

Feature	Brightness	Contrast
Count	2494.00	2494.00
Mean	144.13	57.68
Std	30.63	12.99
Min	0.00	0.00
25%	121.32	49.00
50%	138.78	58.32
75%	157.14	66.63
Max	25.00	106.76

A considerable degree of imbalance in the dataset is indicated by the class balance ratio of 0.2071. While the standard deviation shows notable differences in class frequencies, the mean class count shows a balanced distribution of pictures across classes. This disparity might lead to biased model training, where the model performs poorly in underrepresented classes and well in overrepresented ones. To solve this problem, preprocessing methods such as data augmentation for classes are used. This improves the model's resilience and prediction performance in real-world applications by strengthening class representation and guaranteeing that it generalizes well across all classes.

All images have been resized to 224x224 pixels so that the model can process each image consistently. An example image of our dataset is given in Figure 4.

Figure 4. Sample dataset images

These images are of the 429 Globalranger model. The sample images are shown from our random dataset, and the colors of the helicopters, their positions, and the direction of shooting the images are different from each other. This provides real scenario examples for both models. Our goal is to perform the classification process correctly regardless of the image.

4.2 Data augmentation

A detailed data enrichment approach was put in place to improve the generalizability of the model and to stop it from overfitting. This methodology is aimed at inducing variability and resilience in the training dataset by applying transformations that are split into a sequence of very carefully defined steps. Random Resized Cropping was used to standardize images to a 224 × 224-pixel resolution while simultaneously maintaining the spatial diversity present in the dataset. This provided the model with the capability to identify features regardless of their positional or scalar changes. To enhance invariance to horizontal orientations, Random Horizontal Flipping was applied with 50% probability. Such a transformation helped to include mirrored patterns, therefore strengthening the model. Random Rotation was used to increase rotational invariance with random angular rotations of the images up to 75° (including 15, 30, 45, and 60) to ensure that the model could adequately detect and classify objects regardless of their rotational positioning. To simulate different lighting and color conditions, Color Jittering was introduced, where parameters like brightness, contrast, saturation, and hue were systematically changed. Contrast and saturation were adjusted from 0.1 to 0.3, while brightness was adjusted from 0.2 to 0.6. The hue was changed from 0.1 to 0.6. By doing so, it resulted in a realistic range of color and light conditions that increased the effectiveness of the model in different scenarios. To make sure that the input distribution stayed the same after these changes, the dataset was made homogeneous with a mean of [0.485, 0.456, 0.406] and a standard deviation of [0.229, 0.224, 0.225]. These augmentation techniques resulted in an expansion of the original dataset to 19,944 images, which were then separated into 251 test samples, 498 validation samples, and 19,195 training samples. This augmentation strategy helped the model greatly, since it reduced overfitting and made the model more resilient against variability in the data, in order to be more applicable in a variety of real-world scenarios.

4.3 Histogram equalization

In order to increase the contrast of the images of the models' dataset and to obtain clearer images, a histogram equalization (HE) technique was used. As shown in Figure 5, the histogram equalization process improved the contrast of the images by balancing their pixel intensity distributions, both overall and by color channel.

Figure 5. Original Image and After HE

This will enable the model to extract more and more accurate information from the images, which will positively affect its overall performance. The more balanced and high contrast images obtained after the histogram equalization process will be used more effectively in the learning process of the model. This will improve classification accuracy and overall model performance.

5 Model architectures

The YOLOv9c model is a deep learning model that was designed with object detection in mind. The YOLOv9c [8] model architecture is an upgraded new series of the YOLO series that addresses critical issues in deep learning methods, such as the information bottleneck and the accumulation of errors in deep networks, by incorporating the Programmable Gradient Information (PGI) framework. YOLOv9c is built using the fundamental elements from YOLOv7. Especially concerning architecture and efficiency, the YOLOv9c model shows notable improvements over previous models. These improvements allow the model to run more complex object detection tasks now. They have improved architectural simplicity and computer performance. Such optimizations are crucial for deploying deep learning models in real-world applications where computational resources are often limited, and are essential for helicopter detection and classification, a real-world application. In the training process of the model, pre-trained YOLOv9c weights were used. This allowed for a faster and more efficient training process, taking advantage of the fact that the model had been previously trained on a large dataset. The final layers of the model were adapted to fit the number of classes of our dataset.

In this study, GoogleNet architecture is used to classify helicopter types. The GoogleNet architecture is a deep convolutional neural network with an extensive layer structure and additional classification layers [6]. The main components of the model are as follows: The GoogleNet model, pre-trained on ImageNet, is highly effective in learning basic image features and is used as the main body. The fully connected layers are customized for classification and the fully connected layers in the original GoogleNet model are reconstructed to fit the number of classes. To avoid overlearning, 40% dropout was applied and the last fully connected classification layer was adjusted to set the output size of the model to the number of classes in the dataset, i.e. 24. In addition, two auxiliary classifiers from the original structure of the GoogleNet model were used to allow the main model to generalize better and gradients to reach deeper layers. These auxiliary classifiers are similarly constructed with dropout and fully connected layers. This structure allows the model to both perform better on training data and increase its generalization capability.

AlexNet [8] architecture is used as the second model. AlexNet architecture has an important place among deep convolutional neural networks and has shown high success in large-scale image recognition tasks. The main components of the model are as follows: Starting with a convolution layer with 11x11 filters, the network learns more complex features in deep layers by performing extensive feature extraction.

The model architecture is structured as follows: The first layer applies an 11x11 convolution with 96 filters, followed by batch normalization and then a 3x3 max pooling operation. The second layer conducts a 5x5 convolution using 256 filters, similarly followed by batch normalization and another max pooling step. Subsequent layers, namely the third, fourth, and fifth, implement 3x3 convolutions with 384, 384, and 256 filters respectively. These layers are designed to progressively capture more complex features within the data. The architecture then transitions to fully connected layers, consisting of two layers, each with 4096 neurons. To prevent overfitting, each of these layers incorporates a 50% dropout regularization. The final layer employs a softmax activation function to classify across 24 categories, providing a probability distribution over these classes. This configuration has been carefully tuned through extensive experimentation to optimize performance on training datasets while enhancing the model's ability to generalize to new data. All parameters described have been selected to achieve superior results based on rigorous empirical testing.

6 Data training process

Experimental studies for classification include the learning rate, batch size, epoch size, early stop and patience hyperparameters. In the study, transfer learning models were used and the model was fine-tuned. The hyperparameters and values of the most efficient classification model in these experiments were determined as learning rate 0.001, epoch size 32, epoch number 50 and patience time 5. The learning rate was adjusted by using a decay factor of 0.1 every 5 epochs and StepLR scheduler. The study employed a model trained with a weight decay of 1×10^{-4} and the Adam optimization algorithm. The model's performance was assessed through a 24-class classification setup, measuring accuracy, recall, and F1-score for each helicopter category. For object detection, the YOLOv9c model was utilized without any alterations. This model underwent training for 100 epochs, while the Googlenet model was trained for 50 epochs. The validation dataset played a crucial role in evaluating the model's performance throughout the training phase and in the learning process.

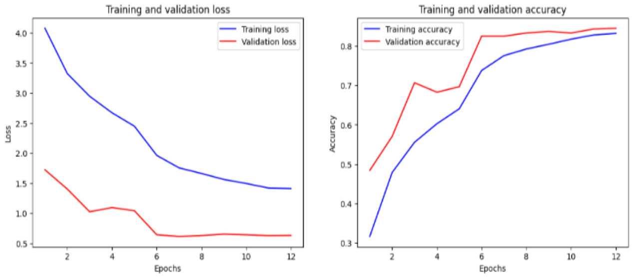

Figure 6. Training, validation loss and accuracy graphs

Figure 6 shows the training and validation loss curves of the GoogleNet model. The curve decreases consistently over the epochs, showing that the model is learning efficiently. The training loss curve consistently decreased as the epochs progressed. This shows that the model is learning the training data well. The validation loss curve similarly decreases, indicating that the model performs well on the overall data and there is no overlearning. The training accuracy of the GoogleNet model has been continuously increasing, showing that the model's performance on the training data has improved and was measured at 83%. The validation accuracy increased in a similar way, showing the generalization ability of the model. The value was observed to be 85%. The value and the pattern of increase here reveals that the model also performs well on test data, i.e. data that the model has not seen before.

Figure 7 shows the change in accuracy and loss values for AlexNet during the training process of the model. At the beginning of training, the accuracy value starts at around 20%. As the number of epochs increases, the accuracy value gradually increases. Especially during the first 20 epochs, a significant increase in accuracy is observed. This increase indicates that the model starts to adapt to the data. When the number of epochs reaches 40, the accuracy exceeds 80% and stabilizes in the last epochs. This shows that the performance of the model is continuously improving throughout the training process and it classifies the data better. The loss value is quite high at the beginning of the training and decreases rapidly in the first epochs. This rapid decrease indicates that the model's errors are rapidly decreasing at the beginning and the parameters of the model are optimized.

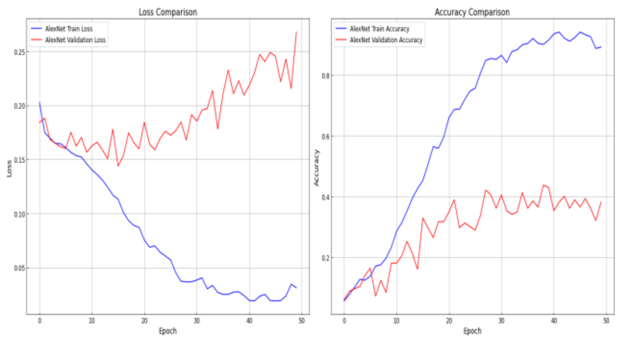

Figure 7. Validation loss and accuracy graphs

When the number of epochs approached 20, the loss value decreased significantly to around 0.05. When the number of epochs reached 40, the loss value dropped below 0.025 and remained at this level for the last epochs. This decrease in the loss value indicates that the learning process of the model is progressing successfully and that it adapts well to the training data. Based on this graph, it can be concluded that the AlexNet model has successfully carried out the training process and has been able to cope with the classification task given to it on the dataset and the overall performance of the model is high.

7 Model evaluation results

For each class, the performance of the YOLOv9C model was measured using criteria such as precision, recall and mAP (Mean Average Precision). The overall precision for this model was 85.2%. The classification results with GoogleNet were evaluated with precision, sensitivity and F-1 score metrics and the overall F-1 score was 81%. The results obtained by both models on a class basis are detailed below.

7.1 YOLOv9c results

The YOLOv9c model's general accuracy came out to be 85.2%. This result indicates that the model performs satisfactorily for the object detection challenge. The helicopter class performed the best, with 96.6% accuracy, 97.7% recall, and 98.3% mAP50. However, with 74.1% accuracy and 59.3% recall, the Propeller class had less performance than the other classes. Table 3 presents all the data findings.

Table 3. YOLOv9c results

Class	Precision	Recall	mAP50	mAP50-95
Helicopter	0.966	0.977	0.983	0.62
Landing Bar/Gear	0.827	0.828	0.843	0.34
Cockpit	0.897	0.931	0.930	0.397
Tail	0.826	0.862	0.865	0.338
Propeller	0.741	0.593	0.724	0.303
Overall	0.852	0.838	0.869	0.400

Figure 8 displays, with confidence value given, the many helicopter components found using the Yolov9c model.

Although the cockpit and propeller are identified with confidence levels of 75% and 65%, respectively, the general form of the aircraft has a confidence level of 85%, with a confidence level of 76% for the tail and 83% for the landing gear, respectively. These many confidence levels correspond to the degree of precision of the method in identifying and grouping every component.

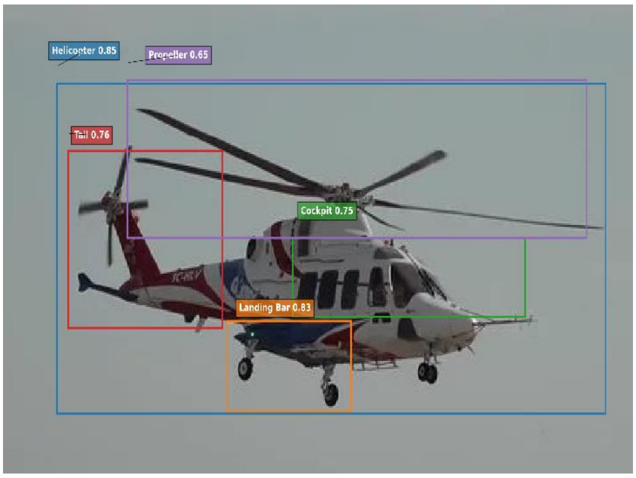

Figure 8. Training, validation loss and accuracy graphs

7.2 GoogleNet results

The performance metrics of the GoogleNet model in Table 4 show that it achieved high accuracy in the classification task. The overall accuracy of the model is 81%, with precision and recall values generally at high levels. In particular, 100% precision and recall values were achieved for the A129 Mangusta, AW139 and MD900 Explorer classes. However, lower precision and recall values were observed in the KC518 Adventourer and 505 Jet Ranger X classes. This may be due to the fact that these classes have less distinctive features than the other classes or that they have less data in the dataset.

7.3 AlexNet results

The model shows an overall high accuracy, as evidenced by the strong concentration of values along the diagonal. This indicates a good ability to correctly classify most helicopter types. However, certain classes pose challenges for the model. For the 505 Jet Ranger X helicopter model, the number of correctly labeled images is 5, while the number of incorrectly labeled images is shown to be 23. This has a negative impact on the success of the model. These deviations are possible due to the quality of the images of the models, the shooting direction or the similarities between the models.

In Table 5, the performance metrics of the AlexNet model show that, like GoogleNet, it provides high accuracy in the classification task. The overall accuracy of the model is 75%, with precision and recall values generally at high levels. The highest accuracy value was observed for the Eurocopter Cougar helicopter. The lowest accuracy was observed for the R44 Raven I. While there were high achievements in some classes, there were some classes where the achievements dropped by half compared to the other classes.

The performance of the YOLOv9c, GoogleNet and AlexNet models in the detection and classification of helicopters and helicopter components is presented in detail. The overall accuracy of the YOLOv9c model is 85.2%, which shows that the model is effective in the object detection task. The overall accuracy of the GoogleNet model is 81% and 75% for the AlexNet model. Overall, our models performed reliably in the classification task.

Table 4. GoogleNet results

Class	Precision	Recall	F1-Score
429 GLOBALRANGER	0.67	0.53	0.59
505 JET RANGER X	0.53	0.50	0.52
A129 MANGUSTA	1.00	0.92	0.96
AK1-3	1.00	0.88	0.93
AW139	1.00	1.00	1.00
EXEC 162F	0.53	0.67	0.59
Eurocopter Cougar	0.86	1.00	0.92
F-28F FALCON	1.00	1.00	1.00
GÖKBEY	1.00	1.00	1.00
KC518 ADVENTOURER	0.42	0.45	0.43
MD900 EXPLORER	1.00	1.00	1.00
MOSQUITO XE	0.90	0.90	0.90
NEO	1.00	0.40	0.57
P.531	1.00	1.00	1.00
R44 RAVEN I	0.67	1.00	0.80
R66	0.69	0.65	0.67
S-76 A	0.50	0.50	0.50
S-92 A SUPERHAWK	1.00	0.86	0.92
SYTON AH 130	0.70	0.88	0.78
T129 ATAK	0.92	1.00	0.96
T70	1.00	0.82	0.90
UH-1 Iroquois	0.91	1.00	0.95
V-22 OSPREY	0.92	1.00	0.96
W-3 SOKOL	0.91	0.91	0.91
Overall	0.81	0.81	0.81

Table 5. AlexNet results

Class	Precision	Recall	F1-Score
429 GLOBALRANGER	0.74	0.67	0.70
505 JET RANGER X	0.60	0.58	0.59
A129 MANGUSTA	0.71	0.79	0.75
AK1-3	0.59	0.87	0.70
AW139	0.76	0.73	0.75
EXEC 162F	0.75	0.54	0.63
Eurocopter Cougar	0.96	0.87	0.91
F-28F FALCON	0.78	0.58	0.67
GÖKBEY	0.92	0.75	0.83
KC518 ADVENTOURER	0.50	0.69	0.58
MD900 EXPLORER	0.73	0.80	0.76
MOSQUITO XE	0.77	0.56	0.65
NEO	0.81	0.72	0.76
P.531	0.80	0.67	0.73
R44 RAVEN I	0.51	0.87	0.65
R66	0.66	0.82	0.73
S-76 A	0.89	0.67	0.76
S-92 A SUPERHAWK	0.81	0.85	0.83
SYTON AH 130	0.88	0.74	0.80
T129 ATAK	0.86	0.77	0.81
T70	0.72	0.78	0.75
UH-1 Iroquois	0.76	0.86	0.81
V-22 OSPREY	0.69	0.65	0.67
W-3 SOKOL	0.82	0.56	0.67
Overall	0.75	0.72	0.73

Table 6. Model comparison results

Class	Precision	Recall	F1-Score
GoogleNet	0.81	0.81	0.81
AlexNet	0.7510	0.7309	0.7323

Table 6 compares the helicopter identification performance of two deep learning architectures, GoogleNet and AlexNet. The results show that GoogleNet achieves a precision score of 0.81, a recall of 0.81 and an F1 score of 0.81, while AlexNet achieves a precision score of 0.7510, a recall of 0.7309 and an F1 score of 0.7323. The high F1 scores and similar recall values for both models suggest that both are effective at detecting true positives. However, GoogleNet's higher sensitivity score indicates that it is better at distinguishing between true and false positives, resulting in fewer false alarms. This difference in sensitivity can be attributed to the unique architecture and design choices of GoogleNet, which incorporates initialization modules and batch normalization. Overall, the results suggest that GoogleNet may be the more appropriate choice for this particular task, but when considering the performance differences between the two models, AlexNet's lower memory usage and easy integration may compensate for this performance difference.

8 Results and future works

The paper proposes an approach using CNN and YOLO architectures to classify and object detection of helicopter images. Driven by the need for great precision and efficiency in practical uses such military operations, search and rescue missions, and civil aviation, the study comprises GoogleNet and AlexNet models to maximize performance under mentioned parameters, architectures and pipelines. In classification tests, the AlexNet-based model obtained a 73% F-1 score; the GoogleNet-based model obtained an F-1 score of 81%. Furthermore displaying a mean average precision (mAP) of 87% in object recognition, the YOLOv9c model highlighted the dependability of the suggested method. The study emphasizes the major contributions of data augmentation methods and transfer learning approaches in improving model performance. GoogleNet's advanced feature extraction via its deep architecture and inception modules complements AlexNet's efficient and simple design, ensuring high accuracy. This approach establishes a solid foundation for reliable helicopter component detection, proving its applicability across various domains. Future studies will seek to expand these approaches to include more complicated operating situations and a wider spectrum of aircraft. Including more varied and large-scale image collections in the dataset will help to improve the generalizability and robustness of the models. Advancing data augmentation methods is expected to help to improve model performance and resilience against various real-world conditions. To increase accuracy and efficiency in aircraft classification and detection tasks, next research will combine CNN architectures with more recent generation object detection algorithms. Moreover, the application of these models in emerging fields such autonomous navigation systems and intelligent air traffic management will be a focal point, where exact and reliable aircraft recognition is paramount. Continuous development and optimization of deep learning models are essential to provide more dependable and effective solutions for aircraft recognition and detection, so contributing to advancements in aviation safety and operational efficiency.

Conflict of interest

The authors declare no conflicts of interests.

Similarity rate (iThenticate): %14

References

[1] Anagün, Y., Işik, Ş., & Çakir, F. H. (2023). Surface roughness classification of electro discharge machined surfaces with deep ensemble learning. Measurement, 215,112855. https://doi.org/10.1016/j.measurement.2023.112855

[2] Kurt, Z., Işık, Ş., Kaya, Z., Anagün, Y., Koca, N., & Çiçek, S. (2023). Evaluation of EfficientNet models for COVID-19 detection using lung parenchyma. Neural Computing and Applications, 35(16), 12121–12132. https://doi.org/10.1007/s00521-023-08344-z

[3] Can, Z., Isik, S., & Anagun, Y. (2024). CVApool: using null-space of CNN weights for the tooth disease classification. Neural Computing and Applications, 36, 16567–16579. https://doi.org/10.1007/s00521-024 09995-2

[4] Gozukara, G., Anagun, Y., Isik, S., Zhang, Y., & Hartemink, A. E. (2023). Predicting soil EC using spectroscopy and smartphone-based digital images. Catena, 231, 107319. https://doi.org/10.1016/j.catena.2023.107319

[5] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. arXiv preprint arXiv:1506.02640. https://doi.org/10.48550/arXiv.1506.02640

[6] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1–9. https://doi.org/10.1109/CVPR.2015.7298594

[7] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, 25.

[8] Wang, C. Y., Yeh, I. H., & Liao, H. Y. M. (2024). Yolov9: Learning what you want to learn using programmable gradient information. arXiv preprint arXiv:2402.13616. https://doi.org/10.48550/arXiv.2402.13616

[9] Wu, Z. Z., Wan, S. H., Wang, X. F., Tan, M., Zou, L., Li, X. L., & Chen, Y. (2020). A benchmark data set for aircraft type recognition from remote sensing images. Applied Soft Computing, 89, 106132. https://doi.org/10.1016/j.asoc.2020.106132

[10] Rong, H. J., Jia, Y. X., & Zhao, G. S. (2014). Aircraft recognition using modular extreme learning machine. Neurocomputing, 128, 166-174. https://doi.org/10.1016/j.neucom.2012.12.064

[11] Zhao, C., Zhang, S., Luo, R., Feng, S., & Kuang, G. (2023). Scattering features spatial-structural association network for aircraft recognition in SAR

images. IEEE Geoscience and Remote Sensing Letters, 20, 1-5. https://doi.org/10.1109/LGRS.2023.3280442

[12] Wang, Y., Chen, Y., & Liu, R. (2022). Aircraft image recognition network based on hybrid attention mechanism. Computational Intelligence and Neuroscience, 2022(1), 4189500. https://doi.org/10.1016/j.measurement.2023.113098

[13] Huang, X., Xu, K., Huang, C., Wang, C., & Qin, K. (2021). Multiple instance learning convolutional neural networks for fine-grained aircraft recognition. Remote Sensing, 13(24), 5132. https://doi.org/10.3390/rs13245132

[14] Zhao, Y., Zhao, L., Li, C., & Kuang, G. (2020). Pyramid attention dilated network for aircraft detection in SAR images. IEEE Geoscience and Remote Sensing Letters, 18(4), 662-666. https://doi.org/10.1109/LGRS.2020.2981255

[15] Wu, Q., Feng, D., Cao, C., Zeng, X., Feng, Z., Wu, J., & Huang, Z. (2021). Improved mask R-CNN for aircraft detection in remote sensing images. Sensors, 21(8), 2618. https://doi.org/ 10.3390/s21082618

[16] Zhang, F., Du, B., Zhang, L., & Xu, M. (2016). Weakly supervised learning based on coupled convolutional neural networks for aircraft detection. IEEE Transactions on Geoscience and Remote Sensing, 54(9), 5553-5563. https://doi.org/10.1109/TGRS.2016.2569141

[17] Alkharji, N., Almazrouei, H., Alzaabi, S., Nassif, A. B., Elsalhy, M., & Talib, M. A. (2024). Aircraft-type classification using deep learning algorithms. *Proceedings of the 12th International Conference on Intelligent Systems (IS)*, Varna, Bulgaria, 1–6. https://doi.org/10.1109/IS61756.2024.10705261