



An Approach for Aircraft Detection using VGG19 and OCSVM

Marwa A. Hameed¹ and Zainab A. Khalaf^{2*}

¹Computer Sciences, College of Computer Science & Information Technology, Basrah University, Basrah, Iraq

²Department of Computer Science, College of Education of Pure Sciences, Basrah University, Basrah, Iraq

Received 4 Nov. 2023, Revised 6 Apr. 2024, Accepted 12 Apr. 2024, Published 1 Jul. 2024

Abstract: Aircraft detection is an essential and noteworthy area of object detection that has received significant interest from scholars, especially with the progress of deep learning techniques. Aircraft detection is now extensively employed in various civil and military domains. Automated aircraft detection systems play a crucial role in preventing crashes, controlling airspace, and improving aviation traffic and safety on a civil scale. In the context of military operations, detection systems play a crucial role in quickly locating aircraft for surveillance purposes, enabling decisive military strategies in real time. This article proposes a system that accurately detects airplanes independent of their type, model, size, and color variations. However, the diversity of aircraft images, including variations in size, illumination, resolution, and other visual factors, poses challenges to detection performance. As a result, an aircraft detection system must be designed to distinguish airplanes clearly without affecting the aircraft's position, rotation, or visibility. The methodology involves three significant steps: feature extraction, detection, and evaluation. Firstly, deep features will be extracted using a pre-trained VGG19 model and transfer learning principle. Subsequently, the extracted feature vectors are employed in One Class Support Vector Machine (OCSVM) for detection purposes. Finally, the results are assessed using evaluation criteria to ensure the effectiveness and accuracy of the proposed system. The experimental evaluations were conducted across three distinct datasets: Caltech-101, Military dataset, and MTARSI dataset. Furthermore, the study compares its experimental results with those of comparable publications released in the past three years. The findings illustrate the efficacy of the proposed approach, achieving F1-scores of 96% on the Caltech-101 dataset and 99% on both Military and MTARSI datasets.

Keywords: Aircraft Detection, Deep Learning, Object Detection, VGG19, OCSVM.

1. INTRODUCTION

At the core of computer vision lies the principle of object detection, a process that involves the identification and categorization of objects by employing rectangular bounding frames. This method serves the dual purpose of localizing and classifying the identified objects, making it an integral part of the broader field of computer vision. Object detection is closely intertwined with related tasks, including object classification, semantic segmentation, and instance segmentation, collectively contributing to the understanding of visual data [1], [2]. The significance of object detection extends across a diverse spectrum of practical applications, encompassing domains such as autonomous driving, robotics, and video surveillance. It plays a pivotal role in enabling machines to perceive and interact with their surroundings effectively [3].

Recently, there was an increasing scholarly focus on the utilization of object detection techniques to the detection of aircraft within the domain of computer vision. This application serves critical functions in contexts like airport control, search operations for crashed aircraft, monitoring the dynamics of potentially hostile aircraft, and finds rele-

vance in both civil and military sectors [3], [4].

Deep learning, as an advanced methodology within computer vision, has emerged as a powerful tool for extracting precise image characteristics and analyzing data. This specific category of machine learning techniques, referred to as deep learning, introduces complexity into models, ultimately enhancing their capabilities. Deep learning has demonstrated the potential to significantly improve accuracy across various domains, encompassing classification, segmentation, as well as object detection [5].

In recent times, deep learning played a pivotal role in autonomously extracting feature representations from data, leading to significant advancements in object detection [6].

A notable contribution by Liu et al. involved the development of an aircraft detection method utilizing corner clustering and Convolutional Neural Networks (CNN). The methodology comprised two primary phases: region detection and classification. It initiated with the identification of potential regions through the application of mean-shift clustering on corners observed in binary images. Subsequently,



CNNs were employed for feature extraction and classification of regions likely to contain aircraft. The aircraft's precise location was ultimately ascertained through subsequent refinement and evaluation. The study utilized optical remote sensing images from the Remote Sensing Object Detection (RSOD) dataset that yielded a classification accuracy of 98.29% [7].

In another comparative evaluation, Alganci et al. assessed the efficacy of three object detection models: Faster Region-based Convolutional Neural Networks (R-CNN), YOLO-v3, and Single Shot MultiBox Detector (SSD) specifically for aircraft detection within very high-resolution (VHR) satellite imagery. The evaluation aimed to handle the challenge of limited labeled data by leveraging the Dataset for Object deTection in Aerial Images (DOTA), which comprised satellite image patches from various sources. Among the models, YOLO achieved a precision rate of 99.6%, while SSD exhibited a precision rate of 87.9%, and Faster R-CNN attained 81.2% [8].

Shi et al. introduced a two-stage airplane detection approach called DPANet (Deconvolution operation with Position Attention mechanism). This method incorporated deep neural networks with deconvolution operations and a Position Attention mechanism to enhance aircraft recognition in aerial imagery. Evaluations were performed on the DOTA and DIstill Observations to Representations (DIOR) datasets, resulting in a notable average precision (AP) increase to 85.95% [4].

Furthermore, Wang et al. proposed the Efficient Weighted Feature Fusion and Attention Network (EWFAN) for aircraft detection. This unique deep neural network integrated a module for weighted feature fusion and spatial attention mechanisms. The experiment utilized large-scale Gaofen-3 SAR images with a resolution of 1 m to assess the efficiency of the proposed architecture, ultimately achieving a detection rate of 95.4% and a false alarm rate of 3.3% [9].

Hu et al. introduced a cascade framework inspired by the YOLOv5 architecture for aircraft detection in remote sensing images. Their dataset, sourced from various satellite platforms, included 17,506 instances of aircraft belonging to 13 different types. The proposed approach achieved a mean average precision of 83.7%. However, the accurate detection of small objects against low-resolution and complex backgrounds remains a challenge. Additionally, the accuracy of direction prediction and recall rate is deemed inadequate, indicating the need for further improvements [10].

Xiao et al. introduced an adjustable deformable network (ADN) incorporating peak feature fusion (PFF) to recognize airplanes in SAR pictures. The primary objective of the PFF approach is to maximize the utilization of the solid scattering properties of aircraft. This involves extracting the most prominent features and combining them. The Harris detector and eight-domain pixel detection of local maxima

extract peak features. Subsequently, multichannel blending is utilized to improve the visibility of airplanes in different imaging situations. The ADN comprises an adaptive spatial feature fusion (ASFF) module and a deformable convolution module (DCM). The ASFF module addresses scale inconsistencies, enhancing the feature pyramid's ability to characterize features across scales and improving multi-scale aircraft identification performance. The DCM dynamically calculates and determines the 2-D differences in feature maps, enhancing the representation of aircraft with different shapes in geometric modeling. The experimental findings on the GaoFen-3 (GF3) dataset indicate that PFF-ADN is highly effective, as evidenced by its F1-score of 91.11% and average precision of 89.34% [11].

A novel framework was presented by Chen et al. for detecting aircraft within Remote Sensing Images (RSIs). They employed a region suggestion technique relying on circular intensity filtering to find probable targets at different scales. In addition, they utilized the Vector of Locally Aggregated Descriptors (VLAD) method to characterize the rotation-invariant Fourier Histogram of Oriented Gradients (HOG) feature. This approach provides a more condensed representation and improved descriptive capability while disregarding the rotation behavior of the target. The optical Remote Sensing Indices (RSIs) utilized in this research were obtained via the RSOD dataset, that consists of 446 remote sensing images, including a total of 4993 aircraft objects. The images used in this study were obtained from Google Earth and Tianditu. They had different levels of detail, with spatial resolutions ranging from 0.5 to 2 meters. The image from Google Earth was 1072 x 975 pixels, while the one from Tianditu had a size of 1116 x 659 pixels. The results demonstrated that the proposed technique could achieve faster and more accurate detection of aircraft targets in RSIs, thereby enhancing overall performance, with an average precision of 93.4%. Analysis of the failure results revealed that in intricate background scenarios, including the concourse of an airport building, might potentially result in misdirected targets. Conversely, the similarity in color between the airplane fuselage and the ground, caused by irregular lighting and aircraft paint, could result in potential missed detections [12].

Liu et al. introduced the YOLO-extract algorithm that improves the model architecture of the YOLOv5 method. The strategy eliminates the feature layer and prediction head, which have limited feature extraction capabilities, replacing them with a new feature extractor with higher abilities within the network. Furthermore, YOLO-extract leverages the concept of residual networks to integrate Coordinate Attention into the network seamlessly. The algorithm further enhances the capacity of the shallow layer to extract feature and position information by pairing mixed dilated convolution with the redesigned residual structure, optimizing the model's feature extraction ability for targets of various scales. The experiment results on the DOTA dataset indicate that YOLO-extract exhibits quicker conver-

gence and reduces computational workload by 45.3GFLOPs and parameter count by 10.526M, compared to the YOLOv5 algorithm. Additionally, YOLO-extract enhances the mean average precision (mAP) by 8.1% and triples the detection speed. However, the extraction of various aircraft target features from remote sensing satellite images faces challenges due to weather conditions, including skylight circumstances, clouds, and fog. Additionally, the scarcity of datasets related to aircraft types poses challenges for aircraft type detection [13].

Zhang et al. introduced an innovative methodology for identifying aircraft in Synthetic Aperture Radar (SAR) images with a low signal-to-clutter-noise ratio (SCNR). This approach utilized coherent scattering enhancement and a fusion attention mechanism. They also improved the Faster R-CNN by implementing a novel pyramid network that incorporates mechanisms for local and contextual attention. The contextual attention mechanism enables the network to extract pertinent contextual information from the image, whereas the local attention mechanism adaptively emphasizes significant features by enhancing their unique attributes. By effectively integrating local and contextual attention, the network can detect aircraft. Considerable experimentation was undertaken utilizing the TerraSAR-X SAR datasets to establish benchmarks. The experimental findings indicate that under conditions of low SCNR, the suggested aircraft detection method attains an average precision of 91.7% [14].

This study presents a method specifically developed to accurately detect airplanes, regardless of model, type, or color modifications. Detection of airplanes in automated activities is a significant difficulty due to their substantial variations in scale, direction, and visual resemblance to other objects. To address these challenges, an airplane detection system must be engineered to achieve robust discrimination, independent of factors such as rotation, pose, or resolution of the airplanes.

The paper is structured into four primary sections. Section 2 provides an in-depth discussion of the key algorithms and techniques utilized in this study. Section 3 presents the proposed system. The ensuing section, Section 4, delves into a comprehensive presentation of the experimental results. This is followed by a comparative analysis and discussion with recent related works in Section 5. Finally, Section 6 offers a concise summary of the paper's conclusion.

2. METHODS

This section primarily concentrates on the presentation of the VGG19 model, an exploration of the one-class Support Vector Machines (OCSVM) algorithm and transfer learning concept.

A. VGG19 model

The VGG19 model, introduced by Simonyan and Zisserman in 2014, is a convolutional neural network consisting of

19 layers. VGG19 comprises five architectural blocks. The convolutional and pooling layers are present in the first and following blocks. Two convolutional layers and one pooling layer are present in the third and fourth blocks, respectively. Four convolutional layers are in the final block. 3×3 modest filters are employed [15]. It includes 16 convolution layers and three fully connected layer. VGG19 is trained using the ImageNet dataset, which consists of more than one million images categorized into 1000 distinct categories [16].

The VGG19 architecture follows a sequential pattern of convolutional layers, interspersed with max-pooling layers, to reduce spatial dimensions and increase the depth of learned features. The convolutional layers employ rectified linear unit (ReLU) activations, promoting faster convergence and better training efficiency. The final layers consist of fully connected layers with softmax activation, enabling classification into multiple classes [17], as shown in Figure 1.

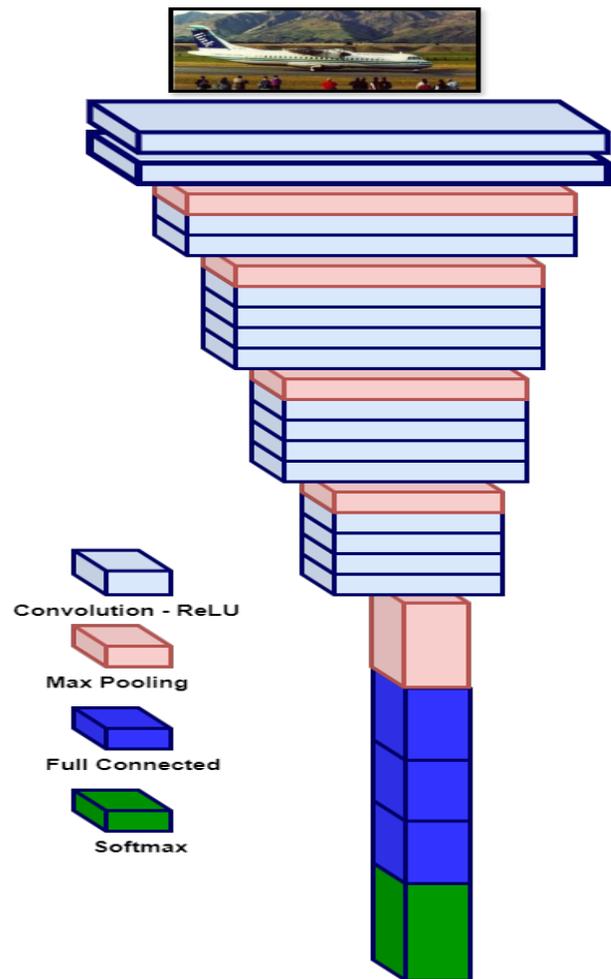


Figure 1. The Architecture of VGG19

VGG19 has been a pivotal model in the field of deep

learning, showcasing the significance of depth in neural network architectures. It has become a popular choice for numerous computer vision applications, including object detection and localization, due to its ability to acquire complex features [18], [19], [20]. In this study, the VGG19 model is employed to extract features from provided images of airplanes.

B. One-Class Support Vector Machine (OCSVM)

The OCSVM is a machine learning algorithm primarily used for anomaly detection or novelty detection tasks [21]. Unlike traditional Support Vector Machines (SVM), which are designed for binary classification, One-Class SVM is focused on learning and identifying a single class or a specific region in the feature space that represents "normal" or typical instances [22].

In essence, One-Class SVM constructs a decision boundary or a hyperplane that encapsulates the majority of the training data, aiming to enclose the "normal" data points while minimizing the outliers or anomalies. This objective is accomplished by identifying the hyperplane that optimizes the margin among the point of origin as well as the nearest points of data belonging to the target class, successfully segregating it from the remaining data points [22], [23]. During the testing or inference phase, the algorithm then identifies instances that fall outside the defined boundary as potential anomalies or deviations from the established norm [24].

Mathematically, the objective of One-Class SVM is to find a hyperplane represented by the equation [25]:

$$\mathbf{w}^T + \phi(\mathbf{x}) - b = 0 \quad (1)$$

where:

The weight vector, denoted as \mathbf{w} , is orthogonal to the hyperplane. Additionally, $\phi(\mathbf{x})$ represents a feature mapping for the input data. The variable "b" represents the bias term [25].

The hyperplane effectively partitions the feature space by maximizing the margin, which refers to the distance from the hyper plane and the nearest data point, commonly referred to as the support vector [23]. To accommodate a certain level of error or deviation from the boundary, a slack variable ξ is introduced, allowing some data points to fall within the margin or on the wrong side of the hyperplane. The optimization problem for One-Class SVM can be formulated as [25]:

$$\min \frac{1}{2} \mathbf{w}^2 + \frac{1}{\nu n} \sum_{i=1}^n \xi_i - \rho \quad (2)$$

Subject to:

$$\mathbf{w}^T + \phi(\mathbf{x}_i) - b \geq \rho - \xi_i, \quad \xi_i \geq 0; \quad i = 1, \dots, n \quad (3)$$

In these equations, the hyper parameter ν regulates the maximum proportion of margin errors, whereas " ρ " represents the radius for the hyper sphere, which encompasses the transformed average data points [22].

C. Transfer Learning (TL)

Deep learning, also known as deep structured learning, represents an advanced category of machine learning methods that has significantly impacted the field of artificial intelligence. The term "deep" refers to the network's depth, indicating the presence of multiple layers, which enables it to learn intricate patterns and features from data. Convolutional Neural Networks (CNNs) are a prominent deep learning technique widely utilized for feature extraction and data classification across various domains. A typical CNN architecture comprises several layers, including input, convolutional, pooling, fully connected, and output layers [5].

Transfer learning (TL) is a valuable technique employed to address data limitations by leveraging knowledge from a different domain. Instead of training a neural network from scratch, TL involves utilizing a pre-trained model's weights and architecture, which have been optimized on a large dataset. This approach allows for the extraction of common features and structures present in images, thereby facilitating the recognition of distinctive features specific to a particular dataset [15].

In TL, the pre-trained model's lower layers, which capture general features, are typically preserved, while the upper layers, responsible for more task-specific features, are fine-tuned to adapt to the new dataset. This transfer of knowledge from the source task to the target task enables the model to achieve good performance with minimal data. However, it is important to note that the pre-trained model's last few layers, initially designed for the source task, may require re-training to better suit the target task's characteristics. [26].

Transfer learning is a technique that entails employing pre-trained deep networks, which have been trained on large datasets, to tackle particular tasks with restricted data. Using the target dataset, it is customary to fine-tune the concluding layers of the pre-trained network. This enables the model to transfer information from the source task to the targeting task. Nevertheless, it is important to note that the ultimate layers, which were initially developed on the source task, might not be acceptable with the target goal [27]. Implementations of transfer learning that utilize pre-trained models tend to be more efficient and reliable compared to models trained from scratch [28].

In this study, two transfer learning approaches are employed:

1) Feature Extraction Approach

The feature extraction technique involves utilizing VGG19 network for the purpose of feature extraction, retaining its original design and learned weights. This approach entails using the network up to a predefined layer as an arbitrary feature extractor, with the outputs of these layers serving as features for further processing [28]. The VGG19 model, known for its 16 convolutional layers and three output layers, leverages its convolutional layers for feature extraction, preserving the original features of input images in the form of feature maps [17].

2) Fine-Tuning Approach

In the fine-tuning strategy, the pre-trained VGG19 architecture is adapted by replacing the original fully connected layers with newly initialized ones. These new fully connected layers are then trained to predict the input classes [29]. Additionally, the last three layers of the VGG19 model are replaced with OCSVM to enhance the model's performance in specific tasks.

3. THE PROPOSED SYSTEM

The fundamental concept presented in this paper involves utilizing the VGG19, which is modeled as a feature extractor through the 16 convolutional layers. Subsequently, these extracted features are then employed for airplane detection using OCSVM.

The architecture for aircraft detection in the proposed system comprises five primary stages, depicted in Figure 2. These stages encompass datasets, preprocessing, feature extraction, detection, and evaluation. Pre-processing plays a vital role in priming the aircraft images for subsequent processing. During the feature extraction phase, deep learning filtering is employed to extract and filter out irrelevant features. These features are then fed into the detection phase to accurately detect airplanes. In the last stage, the system's performance is assessed by evaluating the results obtained. Further elaboration on the system's specifics is provided in the subsequent subsections.

A. Dataset Stage

Three dataset used in this experiment. These are Caltech-101 dataset, Military Aircraft dataset, and MTARSI dataset that collected from the Kaggle website.

1) Caltech -101 Dataset

The Caltech -101 [26] is a widely used computer vision dataset designed for object recognition tasks. It was created by the California Institute of Technology (Caltech) and contains a diverse set of images belonging to 101 different object categories [10]. It comprises around 9146 images distributed among 101 object classes and an additional background clutter class. The number of images in each class ranging between 40 to 800 images, as in the aircraft category, which includes 800 images. The images have a uniform dimension of 300 x 200 pixels.

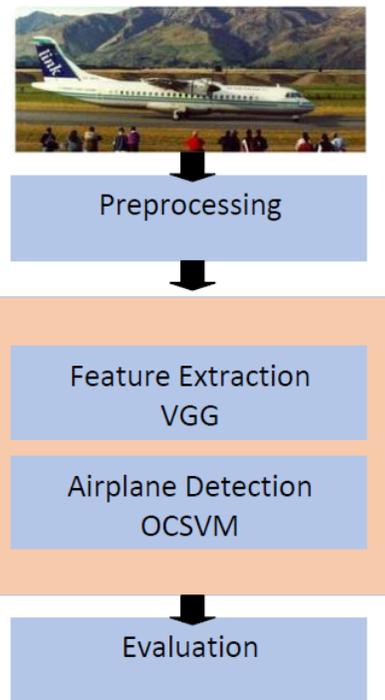


Figure 2. The Proposed System Flowchart

2) Military Aircraft Dataset

This is a remote sensing image Military Aircraft Recognition dataset. It consists of a total of 3,842 images, encompassing 20 distinct types of military aircraft. Each image within the dataset has been meticulously annotated with both horizontal boundary boxes and orientated bounding boxes, resulting in a comprehensive collection of 22,341 instances.

3) MTARSI Dataset

MTARSI, short for Multi-type Aircraft of Remote Sensing images [27] represents the initial publicly accessible dataset encompassing detailed aircraft classification designed for remote sensing photos. A group of seven esteemed professionals specializing in remote sensing image interpretation diligently annotated each instance of the provided images. Hence, the dataset exhibits significant credibility [28]. MTARSI comprises 9,385 remote sensing images derived from satellite imagery provided by Google Earth. The collection of aircraft images includes 36 various airports and 20 distinct type of aircrafts [29].

Additionally, one more dataset was generated based on the aforementioned Caltech-101 dataset. This generated dataset involves converting images into object boundaries using the **Canny** edge detection technique, a method known for its effectiveness in identifying edges with reduced noise and accurate edge localization in digital images. Figure 3 illustrates the main steps of the Canny algorithm.

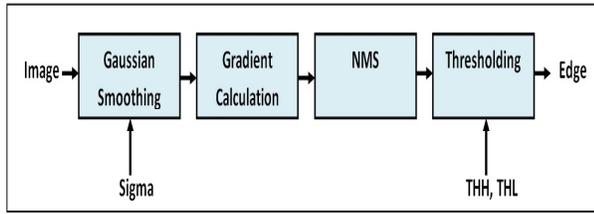


Figure 3. The Traditional Canny Method of Edge Detection

- **Gaussian Smoothing (G)** reduces noise in the image, which can be caused by factors such as sensor imperfections or compression artifacts. It also helps in suppressing small, insignificant edges that might be the result of noise. This is important because the Canny algorithm aims to detect strong, meaningful edges. Canny technique commonly employs a two-dimensional Gaussian function as shown in equation (1) to smooth and remove noise from images [30].

$$G(x, y) = \frac{\exp\left[-\frac{(x^2+y^2)}{2\sigma^2}\right]}{2\pi\sigma^2} \quad (4)$$

The symbol "σ" represents the parameter of the Gaussian filter, which governs the smoothing degree of image.

- **Gradient Calculation** involves computing the gradient magnitude and direction for each pixel in the image. The gradient provides information about how the intensity of the image changes at each point and helps identify regions of rapid intensity change, which often correspond to edges. Edges can be detected in locations where the magnitudes of the picture gradients exhibit significant values. The magnitudes are obtained by convolving the image with the gradient masks [31]. The partial derivatives about the x and y axes can be represented as $P_x(i, j)$ and $P_y(i, j)$, respectively, with the image $I(x, y)$. The conversion from rectangular coordinates to polar coordinates involves transforming the given coordinates $P_x(i, j)$ and $P_y(i, j)$ into the gradient amplitude $M(i, j)$ and the gradient direction $\theta(i, j)$ for a pixel. $M(i, j)$ denotes the edge strength of any point (i , j), while $\theta(i, j)$ denotes the normal vector of any point (i , j) [32].

$$M(i, j) = \sqrt{P_x(i, j)^2 + P_y(i, j)^2} \quad (5)$$

$$\theta(i, j) = \arctan\left(\frac{P_y(i, j)}{P_x(i, j)}\right) \quad (6)$$

- **Non-Maximal Suppression** plays a crucial role in the Canny edge detection method by aiding in the refinement of detected edges through thinning and retaining only the most prominent ones. This step ensures that the final edge map contains only thin, single-pixel-wide edges by suppressing non-maximal

gradient values in the gradient magnitude image. To refine the edge features, it is necessary to suppress all values along the gradient line, with the exception of the local maxima [31]. The NMS method can assist in ensuring that each edge is one pixel wide. The Canny technique utilizes 3 x 3 neighboring regions, encompassing eight directions each, for interpolating the gradient magnitude based on the direction of the gradient. A potential edge point is identified when the magnitude $M(i, j)$ exceeds the cumulative interpolation results along the gradient's direction. In contrast, the point is classified as non-edge if the magnitude is smaller. The method yields the candidate edge image as a result [30].

- **Double Thresholding** is a pivotal step in the Canny algorithm that aims to classify the edges into strong, weak, and non-edge pixels based on gradient magnitude values. This step involves applying two thresholds to the gradient magnitude image to distinguish between different levels of edge strength. Two thresholds are used: a high threshold (THH) and a low threshold (THL). A pixel is classified as a firm edge if its gradient exceeds a threshold value, denoted as THH. If the gradient value falls below the predetermined threshold level (THL), excluding or erasing the pixel corresponding to such a gradient value is imperative [30], [31]. Figure 4 shows an example of each dataset images.

B. Preprocessing Stage

Preprocessing plays a pivotal role in the preparation of images for subsequent processes [33]. It involves a series of transformations applied to an initial image, aimed at improving its quality and rendering statistical analysis more consistent and comparable [34]. In this study, various preprocessing techniques are employed to optimize the data, including resizing the images to a standard size of (224,224). Additionally, for the purpose of data augmentation, methods such as data cropping and rotation are applied.

C. Feature Extraction Stage

Feature extraction stands as a critical phase in airplane detection, as the model's efficacy significantly hinges on the quality and pertinence of the extracted features [35], [36]. This stage is dedicated to capturing pertinent visual characteristics from images that aid in distinguishing airplanes from other objects or backgrounds. As mentioned before the VGG19 model comprised of sixteen convolutional layers, is employed for this purpose.

D. Detection Stage

The detection phase commences right after the completion of the feature extraction stage. To achieve this, the final three layers of the VGG19 model configuration are replaced with OCSVM. OCSVM is a robust and well-regarded detector known for its effectiveness in object detection, particularly in the case of aircraft.



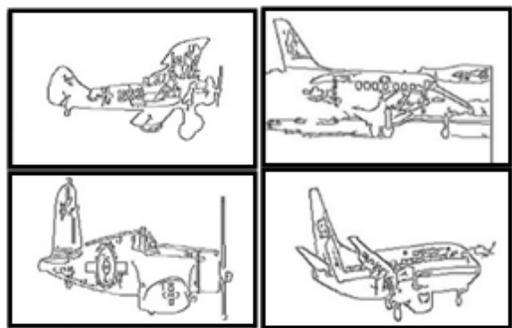
((a))



((b))



((c))



((d))

Figure 4. An Example of Used Datasets ((a)) CalTech-101 Dataset, ((b)) Military Aircraft Dataset, ((c)) MTARSI Dataset, ((d)) Canny Object Boundaries Dataset

E. Evaluation Stage

The efficacy of the proposed method was assessed by utilizing various assessment measures in the experimental results. The criteria employed include F1-score and accuracy. The following are the most fundamentals [33], [37]:

True Positives (TP) refers to the rate at which the model accurately classifies airplanes as positive.

True Negatives (TN) refers to the rate at which the model accurately classifies airplanes as negative.

False Positives (FP) represents the number of airplanes erroneously identifies as positive when, in reality, they are actually negative.

False Negatives (FN) is the number of airplanes erroneously predicted as negative when, in reality, they are actually positive. The metrics can be defined as following:

$$P(\text{Precision}) = \frac{TP}{TP + FP} \tag{7}$$

$$R(\text{Recall}) = \frac{TP}{TP + FN} \tag{8}$$

$$F1\text{-score} = \frac{2 \times P \times R}{P + R} \tag{9}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

4. EXPERIMENTAL RESULTS

The suggested system was executed on a personal computer according to the given specifications: Windows 10 Pro operating system, an Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz 1.99 GHz, installed RAM 8.00 GB, and system type 64-bit operating system. The proposed system was designed using Python programming language, version 3.10, within a Jupyter Notebook environment, chosen for its interactive and collaborative features.

For this study, the VGG19 and OCSVM methods were selected to harness the advantages of CNN and SVM techniques and demonstrate their efficacy in aircraft detection. The training data undergoes feature extraction using VGG19 to determine class values. This involves putting all images through the CNN network to extract relevant features, which are then fed to OCSVM to perform the detection process.

Three datasets were chosen to showcase their outcomes, which were subsequently examined in depth to measure efficacy of the proposed methodology. Each data set was separated manually into two subsets (70:30) : the training set and the test set.

In the experimental evaluation, three detection models were applied: VGG19, OCSVM, and the proposed system.

To evaluate and compare the detection performance of the proposed system, we utilized three datasets: Caltech-101, the Military dataset, and MTARSI. In order to achieve optimal outcomes and guarantee accurate evaluation of the proposed system, It was configured with precise parameters, as outlined in Table I below.

TABLE I. Related Parameters of the Proposed System

Parameter	Value
kernal	linear
gamma	0.01
nu	0.01
cache size	100

Figure 5 illustrates the training and validation loss as well as the accuracy of the proposed system using the Caltech-101 dataset.

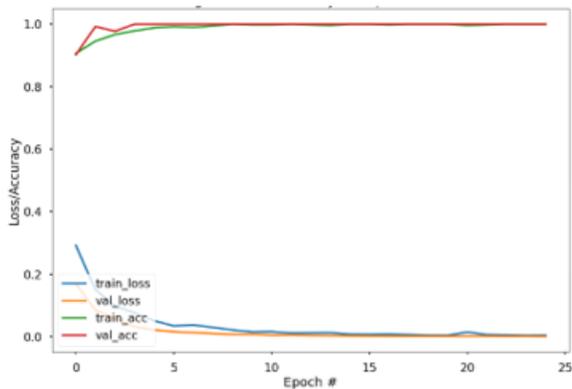


Figure 5. The Loss and Accuracy of Training Phase Utilizing Caltech-101

The effectiveness of the proposed system was assessed by combining VGG19 with OCSVM. The results are detailed in Table II.

TABLE II. Experimental Results of the Suggested System using F1-score

Datasets	Caltech-101	Military	MTARSI
OCSVM	94%	95%	96%
VGG19	88%	96%	97%
Proposed System	96%	99%	99%

Table II reveals that the proposed system outperformed both the VGG19 and OCSVM systems, achieving higher F1-score results for all datasets. For the Caltech-101 dataset, the F1-scores were 96% for the proposed system, 88% for both VGG19 and 94% for OCSVM. In the case of the military aircraft dataset, the obtained F1-scores were 99%, 96%, and 95% for the proposed system, VGG19, and OCSVM respectively. Meanwhile, for the MTARSI dataset, the proposed system exhibited an F1-score of 99% as opposed to 97% for VGG19 and 96% for OCSVM.

Figure 6 showcases an example of the proposed system's results, demonstrating accurate and efficient detection of aircraft in the testing image across all datasets.

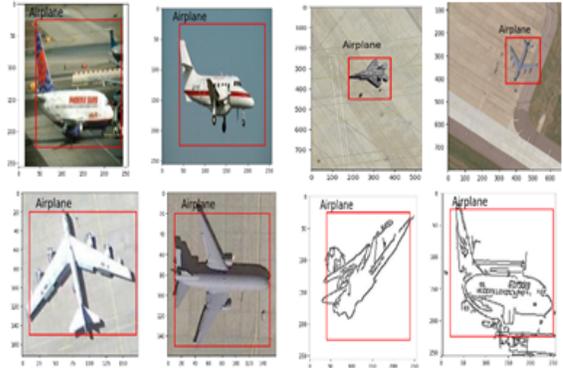


Figure 6. The Results of the Proposed System for Airplane Detection

5. COMPARISON AND DISCUSSION

To measure the efficiency of the proposed method with the chosen datasets, a comprehensive comparative analysis was conducted. This analysis involved contrasting the experimental results with recent research findings concerning the Caltech-101 and MTARSI datasets. The comparison encompassed factors such as dataset size, and the employed detection methodology. Table III presents a holistic view of these comparisons with relevant studies focused on aircraft detection.

TABLE III. A Comparison of the Proposed System with Recent Related Works: F1-score and Accuracy metrics

Reference	Dataset	Detection method	Evaluation metrics
[30]	MTARSI	VGG16	87.5% AC
[38]	Caltech	R-CNN	90.4% F1
[39]	Caltech	(MEsSP)	78.4% F1
[40]	Caltech	Modified Fuzzy C-Mean	70.9% F1
[41]	MTARSI	VGG16	72.1% AC
Proposed System	Caltech	VGG19-OCSVM	96% F1
	MTARSI		99% F1

Table III clearly presents the performance results on the Caltech-101 dataset. Akanksha et al. [38] achieved an F1-score of 90.4% using R-CNN. In contrast, Rafique et al. [39] attained a lower F1-score of 78.4% with their proposed approach by using Maximum Entropy scaled Super Pixels segmentation (MEsSP). Jalal et al. [40] used Modified Fuzzy C-Mean and Maximum Entropy and achieved an F-score of 70.9%. Remarkably, the method introduced in this

paper achieved an outstanding 96% F1-score, signifying a significant enhancement over previous methodologies.

Turning to the MTARSI dataset, Wu et al. [30] introduced MTARSI as the first public database for aircraft remote-sensing images. Their VGG-16 model achieved an accuracy of 87.5%. Similarly, Mo et al. [41] employed VGG-16 and reached an accuracy of 72.10%. Conversely, the approach suggested in this research demonstrated exceptional performance, achieving a flawless 99% F1-score, surpassing the performance of other methods.

This study is limited in its focus on detecting a single airplane within an image, without addressing scenarios involving multiple airplanes, which are common in real-world aerial imagery. This limitation reduces the generalizability of the proposed approach, particularly in complex situations where detecting and localizing multiple instances of the same object class is necessary. Future research could focus on extending the proposed method to effectively handle the detection of multiple airplanes within a single image, thereby enhancing its practical utility in applications such as aerial surveillance and reconnaissance.

6. CONCLUSION

This paper presents an approach for detecting airplanes, utilizing deep learning techniques and a transfer learning methodology. Initially, VGG19 model was employed for feature extraction. Then, the feature vector that has been acquired is afterward inputted into the OCSVM algorithm for aircraft detection. Following numerous experiments, the results obtained from the proposed system demonstrated notable performance improvements compared to traditional unmodified systems (OCSVM, VGG19) and recent related works.

REFERENCES

- [1] Y. Xiao, Z. Tian, J. Yu, Y. Zhang, S. Liu, S. Du, and X. Lan, "A review of object detection based on deep learning," *Multimedia Tools and Applications*, vol. 79, pp. 23 729–23 791, 2020.
- [2] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *Proceedings of the IEEE*, vol. 111, no. 3, pp. 257–276, 2023.
- [3] F. Chen, R. Ren, W. Xu, and T. Van de Voorde, "Airplane recognition from remote sensing images with deep convolutional neural network," in *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2020, pp. 264–267.
- [4] L. Shi, Z. Tang, T. Wang, X. Xu, J. Liu, and J. Zhang, "Aircraft detection in remote sensing images based on deconvolution and position attention," *International Journal of Remote Sensing*, vol. 42, no. 11, pp. 4241–4260, 2021.
- [5] M. Abdulla and A. Marhoon, "Agriculture based on internet of things and deep learning," *Iraqi Journal for Electrical and Electronic Engineering*, vol. 18, no. 2, pp. 1–8, 2022.
- [6] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *International journal of computer vision*, vol. 128, pp. 261–318, 2020.
- [7] Q. Liu, X. Xiang, Y. Wang, Z. Luo, and F. Fang, "Aircraft detection in remote sensing image based on corner clustering and deep learning," *Engineering Applications of Artificial Intelligence*, vol. 87, p. 103333, 2020.
- [8] U. Alganci, M. Soydas, and E. Sertel, "Comparative research on deep learning approaches for airplane detection from very high-resolution satellite images," *Remote sensing*, vol. 12, no. 3, p. 458, 2020.
- [9] J. Wang, H. Xiao, L. Chen, J. Xing, Z. Pan, R. Luo, and X. Cai, "Integrating weighted feature fusion and the spatial attention module with convolutional neural networks for automatic aircraft detection from sar images," *Remote Sensing*, vol. 13, no. 5, p. 910, 2021.
- [10] Q. Hu, R. Li, Y. Xu, C. Pan, C. Niu, and W. Liu, "Toward aircraft detection and fine-grained recognition from remote sensing images," *Journal of Applied Remote Sensing*, vol. 16, no. 2, pp. 024 516–024 516, 2022.
- [11] X. Xiao, H. Jia, P. Xiao, and H. Wang, "Aircraft detection in sar images based on peak feature fusion and adaptive deformable network," *Remote Sensing*, vol. 14, no. 23, p. 6077, 2022.
- [12] X. Chen, J. Liu, F. Xu, Z. Xie, Y. Zuo, and L. Cao, "A novel method of aircraft detection under complex background based on circular intensity filter and rotation invariant feature," *Sensors*, vol. 22, no. 1, p. 319, 2022.
- [13] Z. Liu, Y. Gao, Q. Du, M. Chen, and W. Lv, "Yolo-extract: Improved yolov5 for aircraft object detection in remote sensing images," *IEEE Access*, vol. 11, pp. 1742–1751, 2023.
- [14] X. Zhang, D. Hu, S. Li, Y. Luo, J. Li, and C. Zhang, "Aircraft detection from low scnr sar imagery using coherent scattering enhancement and fused attention pyramid," *Remote Sensing*, vol. 15, no. 18, p. 4480, 2023.
- [15] A. Karacı, "Vggcov19-net: automatic detection of covid-19 cases from x-ray images using modified vgg19 cnn architecture and yolo algorithm," *Neural Computing and Applications*, vol. 34, no. 10, pp. 8253–8274, 2022.
- [16] M. Bansal, M. Kumar, M. Sachdeva, and A. Mittal, "Transfer learning for image classification using vgg19: Caltech-101 image data set," *Journal of ambient intelligence and humanized computing*, pp. 1–12, 2023.
- [17] N. L. Tun, A. Gavrilov, N. M. Tun, H. Aung *et al.*, "Remote sensing data classification using a hybrid pre-trained vgg16 cnn-svm classifier," in *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*. IEEE, 2021, pp. 2171–2175.
- [18] H. Zhang and X. Hong, "Recent progresses on object detection: a brief review," *Multimedia Tools and Applications*, vol. 78, pp. 27 809–27 847, 2019.
- [19] V. Sonde, P. Shirpurkar, M. Giripunje, and P. Ashtankar, "Experimental and dimensional analysis approach for human energy required in wood chipping process," in *International Conference on Advanced Machine Learning Technologies and Applications*. Springer, 2020, pp. 683–691.

- [20] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *Digital Signal Processing*, vol. 126, p. 103514, 2022.
- [21] F. Harrou, A. Dairi, B. Taghezouit, and Y. Sun, "An unsupervised monitoring procedure for detecting anomalies in photovoltaic systems using a one-class support vector machine," *Solar Energy*, vol. 179, pp. 48–58, 2019.
- [22] Y.-P. Zhao, G. Huang, Q.-K. Hu, and B. Li, "An improved weighted one class support vector machine for turboshaft engine fault detection," *Engineering Applications of Artificial Intelligence*, vol. 94, p. 103796, 2020.
- [23] A. Al Shorman, H. Faris, and I. Aljarah, "Unsupervised intelligent system based on one class support vector machine and grey wolf optimization for iot botnet detection," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 7, pp. 2809–2825, 2020.
- [24] H.-J. Xing and L.-F. Li, "Robust least squares one-class support vector machine," *Pattern Recognition Letters*, vol. 138, pp. 571–578, 2020.
- [25] H.-J. Xing and W.-T. Liu, "Robust adaboost based ensemble of one-class support vector machines," *Information Fusion*, vol. 55, pp. 45–58, 2020.
- [26] N. M. Almoosawi and R. S. Khudeyer, "Resnet-34/dr: a residual convolutional neural network for the diagnosis of diabetic retinopathy," *Informatica*, vol. 45, no. 7, 2021.
- [27] S. S. Basha, S. K. Vinakota, V. Pulabaigari, S. Mukherjee, and S. R. Dubey, "Autotune: Automatically tuning convolutional neural networks for improved transfer learning," *Neural Networks*, vol. 133, pp. 112–122, 2021.
- [28] S. Aggarwal, S. Gupta, A. Alhudhaif, D. Koundal, R. Gupta, and K. Polat, "Automated covid-19 detection in chest x-ray images using fine-tuned deep learning architectures," *Expert Systems*, vol. 39, no. 3, p. e12749, 2022.
- [29] J. A. Alhijaj and R. S. Khudeyer, "Integration of efficientnetb0 and machine learning for fingerprint classification," *Informatica*, vol. 47, no. 5, 2023.
- [30] Z.-Z. Wu, S.-H. Wan, X.-F. Wang, M. Tan, L. Zou, X.-L. Li, and Y. Chen, "A benchmark data set for aircraft type recognition from remote sensing images," *Applied Soft Computing*, vol. 89, p. 106132, 2020.
- [31] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," in *2004 conference on computer vision and pattern recognition workshop*. IEEE, 2004, pp. 178–178.
- [32] B. Zhao, W. Tang, Y. Pan, Y. Han, and W. Wang, "Aircraft type recognition in remote sensing images: Bilinear discriminative extreme learning machine framework," *Electronics*, vol. 10, no. 17, p. 2046, 2021.
- [33] Z. A. Khalaf, S. S. Hammadi, A. K. Mousa, H. M. Ali, H. R. Alnajjar, and R. H. Mohsin, "Coronavirus disease 2019 detection using deep features learning," *International Journal of Electrical & Computer Engineering (2088-8708)*, vol. 12, no. 4, 2022.
- [34] H. Moradmand, S. M. R. Aghamiri, and R. Ghaderi, "Impact of image preprocessing methods on reproducibility of radiomic features in multimodal magnetic resonance imaging in glioblastoma," *Journal of applied clinical medical physics*, vol. 21, no. 1, pp. 179–190, 2020.
- [35] H. Jia, Q. Guo, J. Chen, F. Wang, H. Wang, and F. Xu, "Adaptive component discrimination network for airplane detection in remote sensing images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 7699–7713, 2021.
- [36] M. Weiss, S. Staudacher, D. Becchio, C. Keller, and J. Mathes, "Steady-state fault detection with full-flight data," *Machines*, vol. 10, no. 2, p. 140, 2022.
- [37] Z. A. Khalaf and I. Sheet, "News retrieval based on short queries expansion and best matching," *J. Theor. Appl. Inf. Technol.*, vol. 97, no. 2, pp. 490–500, 2019.
- [38] E. Akanksha and P. R. Koteswara Rao, "A feature extraction approach for multi-object detection using hog and ltp," *International Journal of Intelligent Engineering & Systems*, vol. 14, no. 5, 2021.
- [39] A. A. Rafique, M. Gochoo, A. Jalal, and K. Kim, "Maximum entropy scaled super pixels segmentation for multi-object detection and scene recognition via deep belief network," *Multimedia Tools and Applications*, vol. 82, no. 9, pp. 13 401–13 430, 2023.
- [40] A. Jalal, A. Ahmed, A. A. Rafique, and K. Kim, "Scene semantic recognition based on modified fuzzy c-mean and maximum entropy using object-to-object relations," *IEEE Access*, vol. 9, pp. 27 758–27 772, 2021.
- [41] H. Mo and G. Zhao, "Ric-cnn: Rotation-invariant coordinate convolutional neural network," *Pattern Recognition*, vol. 146, p. 109994, 2024.



Marwa A. Hameed : She completed her undergraduate studies in computer science at the College of Science in 2011. Currently, she is pursuing a master's degree in Computer Sciences at the College of Computer Science & Information Technology, Basrah University, Basrah, Iraq.



Dr. Zainab A. Khalaf : She is a lecturer in the Department of Computer Science at the University of Basrah. She completed her PhD at the University of USM in Malaysia in March 2015. Her research interests include artificial intelligence, natural language processing (NLP), data mining, speech recognition, and computer vision.