ISSN: 2454-9940

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

E-Mail :
editor.ijasem@gmail.com
editor@ijasem.org

www.ijasem.org

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN 2454-9940

www.ijasem.org

Vol 19, Issue 1, 2025

Review of An Improved UAV Identification and Detection Using Deep Learning

*Dr. S V Saboji[1], Suresh Talwar[2]

[1] Professor, Basaveshwar Engineering College, Bagalkot, 587104

[2]AssistantProfessor, St. Martin's Engineering College, Secunderabad, Telangana – 500100

*Corresponding Author

Email: sabojiskumar@yahoo.com

Abstract— Urban greenery, including trees, plays a crucial role in enhancing the livability of cities. Accurate identification and assessment of urban trees are important for urban planning, environmental conservation, and management. Traditional methods of tree identification often relied on manual inspection, which can be time-consuming and labor-intensive. By training models on extensive datasets of high-resolution UAV imagery, this research endeavors to develop a system capable of autonomously and accurately identifying and classifying urban trees. Deep learning algorithms excel at learning complex patterns and features, making them well-suited for this task. This advancement holds great promise for enhancing urban planning and environmental conservation efforts by providing a more efficient and accurate means of tree identification in urban areas

I. INTRODUCTION

In the realm of urban development, the integration of green spaces, particularly trees, plays a pivotal role in elevating the overall quality of life within cities. The precise identification and assessment of urban trees carry significant implications for effective urban planning, environmental preservation, and resource management. Historically, the conventional methods employed for tree identification relied heavily on manual inspection, a process notorious for its time-consuming and labor-intensive nature [1].

However, the landscape of tree identification has undergone a transformative shift with the advent of high-resolution Unmanned Aerial Vehicle (UAV) imagery and the application of deep learning techniques. This technological evolution has ushered in a new era where the automation of tree identification becomes a tangible reality. The central challenge at hand involves crafting a system capable of accurately recognizing and categorizing individual trees from the intricate details embedded in high-resolution UAV imagery [2]. This necessitates the training of a deep learning model to discern distinctive features of various tree species and distinguish them amidst the diverse elements present in the imagery.

Against the backdrop of expanding urban areas and a burgeoning consciousness of environmental issues, the need for precise data on urban tree populations becomes increasingly imperative. This data, serving as a valuable resource, empowers urban planners, environmentalists, and policymakers to make well-informed decisions pertaining to tree conservation, urban design, and the management of green spaces [3].

The crux of the "Deep Learning-Based Urban Tree Identification from High-Resolution UAV Imagery" project lies in harnessing advanced deep learning algorithms to substantially enhance the accuracy and efficiency of tree identification in urban settings. Through the systematic training of models on extensive datasets comprising high-resolution UAV imagery, the research aims to cultivate a system endowed with the autonomy to accurately identify and classify urban trees. The inherent strength of deep learning algorithms lies in their ability to decipher intricate patterns and features, rendering them exceptionally well-suited for the intricacies of this task.

This [4] technological leap forward holds immense promise, not only for the realm of urban planning but also for bolstering environmental conservation endeavors. By providing a more streamlined, efficient, and accurate mechanism for tree identification in urban areas, this advancement stands to significantly augment our capacity to navigate the complexities of urban development while safeguarding the ecological balance. the complexities of urban development while safeguarding the ecological balance.

1.1 Problem Statement

The urban landscape, enriched by the presence of greenery and trees, is integral to fostering a high quality of life in cities. However, the accurate identification and assessment of urban trees pose significant challenges crucial to effective urban planning, environmental conservation, and management. Traditional approaches, reliant on manual inspection, are burdened by time-consuming and labor-intensive processes[5].

The [6] advent of high-resolution Unmanned Aerial Vehicle (UAV) imagery and the application of deep learning techniques present an opportunity to revolutionize this paradigm. The primary obstacle lies in crafting a system capable of precisely identifying and classifying individual trees within high-resolution UAV imagery. This necessitates the development of a deep learning model trained to discern unique features of diverse tree species while distinguishing them from other elements within the imagery.

As urban areas expand and environmental concerns heighten, the demand for accurate data on urban tree populations becomes increasingly urgent [7]. Such data holds intrinsic value for urban planners, environmentalists, and policymakers, enabling them to make well-informed decisions concerning tree conservation, urban design, and the management of green spaces.
The proposed project, titled "Deep Learning-Based Urban Tree Identification from High-Resolution UAV Imagery," seeks to harness advanced deep learning algorithms to substantially enhance the precision and efficiency of tree identification in urban landscapes [8]. Through the systematic training of models on extensive datasets comprising high-resolution UAV imagery, the research endeavors to create a system with the autonomy to accurately identify and classify urban trees. Leveraging the capacity of deep learning algorithms to comprehend intricate patterns and features, this advancement holds great promise in significantly improving urban planning and environmental conservation efforts [9]. By providing a more streamlined and accurate means of tree identification, the project aims to address the challenges posed by the expanding urban environment and contribute to sustainable urban development[10].

1.2 Research Motivation

The motivation behind the research project, "Deep Learning-Based Urban Tree Identification from High-Resolution UAV Imagery," stems from the critical role that urban greenery, particularly trees, plays in shaping the livability of cities. In the context of urban planning, environmental conservation, and effective management, the accurate identification and assessment of urban trees emerge as pivotal factors.
Historically, traditional methods of tree identification relied heavily on manual inspection, a process known for its inherent drawbacks of being both time-consuming and labor-intensive. The advent of high-resolution Unmanned Aerial Vehicle (UAV) imagery and the integration of deep learning techniques present a transformative opportunity to revolutionize the cumbersome manual processes. The primary challenge at the heart of this motivation is the need to develop a sophisticated system capable of autonomously and accurately identifying and classifying individual trees within high-resolution UAV imagery.
As urban areas continue to expand and concerns about environmental sustainability grow, the demand for precise data on urban tree populations becomes increasingly urgent. This information holds immense value for key stakeholders, including urban planners, environmentalists, and policymakers, who rely on accurate data to make informed decisions regarding tree conservation, urban design, and the effective management of green spaces.
The crux of the research motivation lies in recognizing the potential of advanced deep learning algorithms to significantly enhance the accuracy and efficiency of tree identification in urban environments. By leveraging these algorithms and training models on extensive datasets of high-resolution UAV imagery, the research aims to create a system that not only automates the identification process but also ensures a high level of accuracy and reliability. The innate ability of deep learning algorithms to discern intricate patterns and features makes them particularly well-suited for the complexities involved in distinguishing between different tree species and other elements within the imagery.
Ultimately, the research aspires to contribute to the enhancement of urban planning and environmental conservation efforts. By providing a more efficient and accurate means of tree identification in urban areas, the project holds great promise in addressing the evolving challenges presented by urban expansion and environmental concerns, ultimately fostering more sustainable and ecologically conscious urban development. Alipour-Fanid et al. [11] proposed a machine learning-based delay-aware UAV detection and operation mode identification system over encrypted WiFi traffic. They utilized machine learning algorithms to detect and classify UAVs based on encrypted WiFi traffic, achieving notable results. Zhao et al. [12] introduced an approach using Auxiliary Classifier Wasserstein Generative Adversarial Networks (AC-WGANs) for UAV classification. Their method achieved a high accuracy of 95% for UAV classification at 5 dB and above, using compressed signals and AC-WGANs for classification.
Nemer et al. [13] presented a hierarchical learning approach for RF-based UAV detection and identification. They achieved an impressive accuracy of 99.2% for detecting operation modes of UAVs. However, their evaluation metric was flat, and they did not consider interference from other ISM devices. Swinney and Woods [14] proposed a deep residual learning approach for UAV flight mode classification. Their method achieved an accuracy of 91%, utilizing power spectral density (PSD) images of UAV signals and ResNet50 for feature extraction. Ezuma et al. [15] focused on detecting and classifying UAVs using RF fingerprints in the presence of WiFi and Bluetooth interference. They achieved promising results using RF fingerprints and machine learning techniques.
Medaiyese et al. [16] proposed a semisupervised learning framework for UAV detection, aiming to improve detection accuracy using unlabeled data. Their approach could potentially enhance UAV detection systems. Soltani et al. [17] discussed RF fingerprinting of unmanned aerial vehicles using nonstandard transmitter waveforms. Their work contributes to understanding and identifying UAVs based on RF signatures. Akter et al. [18] presented a sequential convolutional neural networks (CNN) approach for RF-based UAV surveillance systems. Their method utilizes CNNs for effective UAV detection and identification. Al-Emadi and Al-Senaid [19] proposed a drone detection approach based on radio-frequency using convolutional neural networks. Their method provides a novel approach to UAV detection using RF signals and CNNs this is also another most serious challenge.

II. SYSTEM MODELS

EXISTING SYSTEM

3.1.1 Machine Learning

Before we take a look at the details of various machine learning methods, let's start by looking at what machine learning is, and what it isn't. Machine learning is often categorized as a subfield of artificial intelligence, but I find that categorization can often be misleading at first brush. The study of machine learning certainly arose from research in this context, but in the data science application of machine learning methods, it's more helpful to think of machine learning as a means of building models of data. Fundamentally, machine learning involves building mathematical models to help understand data. "Learning" enters the fray when we give these models tunable parameters that can be adapted to observed data; in this way the program can be considered to be "learning" from the data. Once these models have been fit to previously seen data, they can be used to predict and understand aspects of newly observed data. I'll leave to the reader the more philosophical digression regarding the extent to which this type of mathematical, model-based "learning" is similar to the "learning" exhibited by the human brain. Understanding the problem setting in machine learning is essential to using these tools effectively, and so we will start with some broad categorizations of the types of approaches we'll discuss here.

3.1.2 Categories of Machine Leaning

At the most fundamental level, machine learning can be categorized into two main types: supervised learning and unsupervised learning.

Supervised learning involves somehow modeling the relationship between measured features of data and some label associated with the data; once this model is determined, it can be used to apply labels to new, unknown data. This is further subdivided into classification tasks and regression tasks: in classification, the labels are discrete categories, while in regression, the labels are continuous quantities. We will see examples of both types of supervised learning in the following section.

Unsupervised learning involves modeling the features of a dataset without reference to any label and is often described as "letting the dataset speak for itself." These models include tasks such as clustering and dimensionality reduction. Clustering algorithms identify distinct groups of data, while dimensionality reduction algorithms search for more succinct representations of the data. We will see examples of both types of unsupervised learning in the following section.

3.1.3 Need for Machine Learning

Human beings, at this moment, are the most intelligent and advanced species on earth because they can think, evaluate, and solve complex problems. On the other side, AI is still in its initial stage and have not surpassed human intelligence in many aspects. Then the question is that what is the need to make machine learn? The most suitable reason for doing this is, “to make decisions, based on data, with efficiency and scale”.

Lately, organizations are investing heavily in newer technologies like Artificial Intelligence, Machine Learning and Deep Learning to get the key information from data to perform several real-world tasks and solve problems. We can call it data-driven decisions taken by machines, particularly to automate the process. These data-driven decisions can be used, instead of using programing logic, in the problems that cannot be programmed inherently. The fact is that we can’t do without human intelligence, but other aspect is that we all need to solve real-world problems with efficiency at a huge scale. That is why the need for machine learning arises.

3.1.4 Limitations in Machines Learning

While Machine Learning is rapidly evolving, making significant strides with cybersecurity and autonomous cars, this segment of AI as whole still has a long way to go. The reason behind is that ML has not been able to overcome number of challenges. The challenges that ML is facing currently are −

Quality of data − Having good-quality data for ML algorithms is one of the biggest challenges. Use of low-quality data leads to the problems related to data preprocessing and feature extraction.

Time-Consuming task − Another challenge faced by ML models is the consumption of time especially for data acquisition, feature extraction and retrieval.

Lack of specialist persons − As ML technology is still in its infancy stage, availability of expert resources is a tough job.

No clear objective for formulating business problems − Having no clear objective and well-defined goal for business problems is another key challenge for ML because this technology is not that mature yet.

Issue of overfitting & underfitting − If the model is overfitting or underfitting, it cannot be represented well for the problem.

Curse of dimensionality − Another challenge ML model faces is too many features of data points. This can be a real hindrance.

Difficulty in deployment − Complexity of the ML model makes it quite difficult to be deployed in real life.

a. Graphical User Interface (GUI): The project utilizes the Tkinter library to create a GUI for user interaction. Tkinter provides buttons and text areas for users to upload datasets, preprocess data, train models, and make predictions.

b. Dataset Upload: Users can upload a dataset containing high-resolution UAV images of urban areas. This dataset includes images of areas with trees and without trees.

c. Data Preprocessing: After uploading the dataset, the system preprocesses the data by extracting features from each image. This preprocessing step involves resizing images, extracting bounding box coordinates, and normalizing the bounding box data.

d. Training VGG Model: The project trains a deep learning model using a combination of VGG16 (a pre-trained

ISSN 2454-9940
www.ijasem.org
Vol 19, Issue 1, 2025

convolutional neural network) and Faster RCNN (an object detection algorithm). This model learns to identify trees in the UAV images based on extracted features and bounding box information.

e. Model Evaluation: After training the model, the system evaluates its performance using metrics such as accuracy, precision, recall, and F1-score. These metrics provide insights into the model's classification performance.

f. Visualization: The project includes visualization components, such as graphs showing accuracy and loss over epochs during model training. Additionally, the system displays a confusion matrix to visualize the model's classification performance.

g. Prediction on Test Images: Users can select test images to predict whether trees are present in the images. The trained model analyzes the images and generates predictions along with bounding box coordinates. Users can visualize the predictions overlaid on the test images.

h. User Feedback: The GUI provides a text area where the system outputs messages, such as progress updates, model evaluation results, and predictions. This feedback helps users understand the system's status and results.

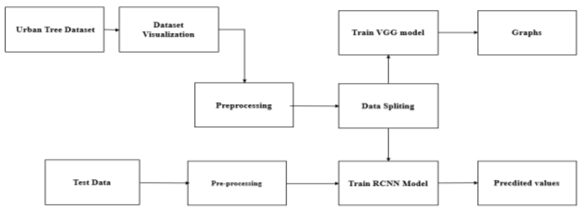

Figure 4.1: Block Diagram of Proposed System.

Data preprocessing

Image preprocessing is a critical step in computer vision and image analysis tasks. It involves a series of operations to prepare raw images for further processing by algorithms or neural networks. Here's an explanation of each step in image preprocessing:

Step 1. Image Read: The first step in image preprocessing is reading the raw image from a source, typically a file on disk. Images can be in various formats, such as JPEG, PNG, BMP, or others. Image reading is performed using libraries or functions specific to the chosen programming environment or framework. The result of this step is a digital representation of the image that can be manipulated programmatically.

Step 2. Image Resize: Image resize is a common preprocessing step, especially when working with machine learning models or deep neural networks. It involves changing the dimensions (width and height) of the image. Resizing can be necessary for several reasons:

a. Ensuring uniform input size: Many machine learning models, especially convolutional neural networks, require input images to have the same dimensions. Resizing allows you to standardize input sizes.
b. Reducing computational complexity: Smaller images require fewer computations, which can be beneficial for faster training and inference.
c. Managing memory constraints: In some cases, images need to be resized to fit within available memory constraints.

When resizing, it's essential to maintain the aspect ratio to prevent image distortion. Typically, libraries like OpenCV or Pillow provide convenient functions for resizing images.

Step 3. Image to Array: In this step, the image is converted into a numerical representation in the form of a multidimensional array or tensor. Each pixel in the image corresponds to a value in the array. The array is usually structured with dimensions representing height, width, and color channels (if applicable).

For grayscale images, the array is 2D, with each element representing the intensity of a pixel. For color images, it's a 3D or 4D array, with dimensions for height, width, color channels (e.g., Red, Green, Blue), and potentially batch size (if processing multiple images simultaneously).

The conversion from an image to an array allows for numerical manipulation and analysis, making it compatible with various data processing libraries and deep learning frameworks like NumPy or TensorFlow.

Step 4. Image to Float32: Most machine learning and computer vision algorithms expect input data to be in a specific data type, often 32-bit floating-point numbers (float32). Converting the image array to float32 ensures that the pixel values can represent a wide range of intensities between 0.0 (black) and 1.0 (white) or sometimes between -1.0 and 1.0, depending on the specific normalization used.

ISSN 2454-9940
www.ijasem.org
Vol 19, Issue 1, 2025

This step is essential for maintaining consistency in data types and enabling compatibility with various machine learning frameworks and libraries. It's typically performed by dividing the pixel values by the maximum intensity value (e.g., 255 for an 8-bit image) to scale them to the [0.0, 1.0] range.

Step 5. Image to Binary: Image binarization is a process of converting a grayscale image into a binary image, where each pixel is represented by either 0 (black) or 1 (white) based on a specified threshold. Binarization is commonly used for tasks like image segmentation, where you want to separate objects from the background.

The process involves setting a threshold value, and then for each pixel in the grayscale image, if the pixel value is greater than or equal to the threshold, it is set to 1; otherwise, it is set to 0.

Binarization simplifies the image and reduces it to essential information, which can be particularly useful in applications like character recognition or object tracking, where you need to isolate regions of interest.

4.3 Dataset Splitting

In machine learning data pre-processing, we divide our dataset into a training set and test set. This is one of the crucial steps of data pre-processing as by doing this, we can enhance the performance of our machine learning model. Suppose if we have given training to our machine learning model by a dataset and we test it by a completely different dataset. Then, it will create difficulties for our model to understand the correlations between the models. If we train our model very well and its training accuracy is also very high, but we provide a new dataset to it, then it will decrease the performance. So we always try to make a machine learning model which performs well with the training set and also with the test dataset. Here, we can define these datasets as:

TrainingSet: A subset of dataset to train the machine learning model, and we already know the output.

Testset: A subset of dataset to test the machine learning model, and by using the test set, model predicts the output.

4.4 VGG Model

VGG16, is called as Visual Geometry Group at Oxford University, is a seminal Convolutional Neural Network (CNN) architecture introduced in 2014. While newer models boast better raw performance, VGG16's simplicity, interpretability, and effectiveness in various tasks make it a cornerstone of computer vision and deep learning education.

Key features:

a. Depth: VGG16 boasts 16 weight layers (excluding pooling and fully connected layers), exceeding earlier architectures like AlexNet, allowing for extraction of more complex features.

b. 3x3 Filters: Instead of larger filters in AlexNet, VGG16 employs smaller 3x3 filters throughout, reducing parameter count and potentially mitigating vanishing gradients.

c. Stacking Convolutional Layers: Feature extraction is achieved by stacking convolutional layers. Each stack follows a pattern:

d. 3x3 convolutional layers (number varies), each with different numbers of channels (64, 128, 256, 512)

e. ReLU activation for non-linearity

f. Zero-padding to preserve spatial information

g. Max-pooling layer for down sampling

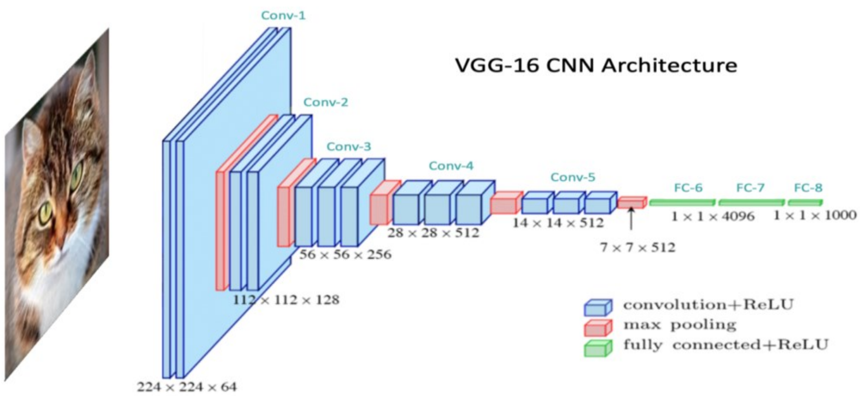

Figure 4.2: Architecture diagram of VGG16 model.

4.2 VGG16 Model Working:

ISSN 2454-9940
www.ijasem.org
Vol 19, Issue 1, 2025

The VGG16 works to extract information from images and classify them.
Preprocessing (Not part of VGG16 itself)

1. The input image (224x224 RGB) gets preprocessed before entering the network. This typically involves:
2. Mean subtraction: Removing the average pixel value from each channel to center the data.
3. Color channel conversion: Normalizing the pixel values (e.g., dividing by 255) and potentially converting to another color space.

Convolutional Blocks (Feature Extraction)

a. The core of VGG16 is the series of convolutional blocks, each extracting increasingly complex features:
b. Convolutional Layers:
c. Each block applies multiple (2-4) 3x3 convolutional filters to the previous layer's output.
d. Each filter learns to detect specific patterns in the image.
e. ReLU activation adds non-linearity, keeping only positive activations.
f. Zero-padding ensures spatial dimensions remain the same.
g. Max-Pooling Layer: Downsamples the feature map by taking the maximum value in a 2x2 region, reducing resolution and computational cost.
h. Retains key spatial information.

Fully Connected Layers (Classification)

1. After the final convolutional block, the network transitions to fully connected layers:
2. Flatten the feature maps into a one-dimensional vector.
3. Pass through 3 densely connected layers with increasing numbers of neurons (4096, 4096, 1000).
4. Use ReLU activation in the first two layers for non-linearity.

Output (Probability Distribution)

1. The final layer employs a Softmax function, outputting a probability distribution across 1000 classes (for ImageNet).
2. The class with the highest probability gets identified as the object in the image.

king of the outsourced data on behalf of the data owner. To execute the integrity checking of data the verifier needs to produce a challenging message and sends it to the cloud server. The cloud server needs to respond the computed proof for the selected file blocks to the verifier besides managing and storing these outsourced data of data owner. Here, it is assumed that cloud server is always equipped with Powerful computing capacity; data owner and the verifier have only constrained computational power or bandwidth.

III. CONCLUSION

In conclusion, the project titled "Deep Learning-Based Urban Tree Identification from High-Resolution UAV Imagery" stands as a pivotal initiative in the realm of urban planning and environmental conservation. The pressing need for accurate and efficient identification of urban trees is met head-on by harnessing the power of advanced deep learning algorithms and high-resolution UAV imagery.

Traditionally, the manual inspection of trees in urban areas has been a labor-intensive process, hampering the speed and accuracy of data collection. However, the integration of cutting-edge technology allows for a revolutionary shift towards automation. Through extensive datasets and meticulous training, deep learning models are poised to recognize and classify individual trees with a precision that surpasses traditional methods.

The implications of this research extend far beyond the realms of academia, impacting urban planners, environmentalists, and policymakers alike. As urban areas expand and environmental concerns intensify, the ability to access accurate data on tree populations becomes paramount. The developed system promises to be a game-changer, offering an autonomous and highly accurate means of tree identification in urban environments.

still a hot topic in research.

REFERENCES

[1] B. H. Y. Alsalam, K. Morton, D. Campbell, and F. Gonzalez, "Autonomous UAV with vision based on-board decision making for remote sensing and precision agriculture," in Proc. IEEE Aerosp. Conf. Big Sky, MT, USA: IEEE, Mar. 2017, pp. 1–12.
[2] S. Coveney and K. Roberts, "Lightweight UAV digital elevation models and orthoimagery for environmental applications: data accuracy evaluation and potential for river flood risk modelling," Int. J. of Remote Sens., vol. 38, no. 8-10, pp. 3159–3180, 2017.
[3] A. Palmer, "Amazon wins FAA approval for prime air drone delivery fleet," 2020, accessed: 2021-16-03. [Online]. Available: https://tinyurl.com/um26c4ku
[4] J. Porter, "Alphabet's wing drones get FAA approval to make deliveries in the US," 2019, accessed: 2021-16-03. [Online]. Available: https://tinyurl.com/43zse424
[5] R. Altawy and A. M. Youssef, "Security, privacy, and safety aspects of civilian drones: A survey," ACM Trans. on Cyber-Physical Syst., vol. 1, no. 2, pp. 1–25, 2016.
[6] NASA, "What is unmanned aircraft systems traffic management?" 2020, accessed: 2020-10-12. [Online]. Available: https://www.nasa.gov/ ames/utm
[7] FAA, "UAS by the numbers," 2020, "Accessed: 2020-10-12". [Online]. Available: https://www.faa.gov/uas/resources/by_the_numbers/
[8] L. C. Danielle Furfaro and N. Musumeci, "Civilian drone crashes into army helicopter," 2017, accessed: 2021-16-03. [Online]. Available: https://tinyurl.com/tyv3ayak
[9] O. Medaiyese, M. Ezuma, A. P. Lauf, and I. Guvenc, "Wavelet transform analytics for RF-based UAV detection and identification system using machine learning," arXiv preprint arXiv:2102.11894, 2021. [10] M. F. Al-Sa'd, A. Al-Ali, A. Mohamed, T. Khattab, and A. Erbad, "RFbased drone detection and identification using deep learning approaches: An initiative towards a large open source drone database," Elsevier, Future Generation Comput. Syst., vol. 100, pp. 86–97, 2019.
[11] A. Alipour-Fanid, M. Dabaghchian, N. Wang, P. Wang, L. Zhao, and K. Zeng, "Machine learning-based delay-aware UAV detection and operation mode identification over encrypted WiFi traffic," IEEE Trans. on Inf. Forensics and Security, vol. 15, pp. 2346–2360, 2019.
[12] C. Zhao, C. Chen, Z. Cai, M. Shi, X. Du, and M. Guizani, "Classification of small UAVs based on auxiliary classifier wasserstein GANs," in Proc. IEEE Global Commun. Conf. (GLOBECOM), Abu Dhabi, United Arab Emirates, Dec. 2018, pp. 206–212.
[13] I. Nemer, T. Sheltami, I. Ahmad, A. U.-H. Yasar, and M. A. Abdeen, "RF-based UAV detection and identification using hierarchical learning approach," MDPI Sensors, vol. 21, no. 6, p. 1947, 2021.