TELKOMNIKA Telecommunication Computing Electronics and Control
Vol. 24, No. 6, December 2026, pp. 2060~2073
ISSN: 1693-6930, DOI: 10.12928/TELKOMNIKA.v24i6.27036

Enhancing the accuracy of drone sound detection using a deep transfer learning model

Fajar Anggara[1], Dedik Romahadi[1], Aberham Genetu Feleke[2]
[1]Department of Mechanical Engineering, Faculty of Engineering, Universitas Mercu Buana, Jakarta, Indonesia
[2]School of Mechanical Engineering, Beijing Institute of Technology, Beijing, China

Article Info

Article history:

Received Mar 1, 2025
Revised Aug 22, 2026
Accepted Sep 8, 2026

Keywords:

Continuous wavelet transform
Deep learning
Drone sound detection
GoogLeNet
Short-time Fourier transform
Transfer learning

ABSTRACT

Unmanned aerial vehicles (UAVs) are increasingly accessible, escalating risks from covert surveillance and hazardous material transport. Acoustic drone detection helps safeguard critical infrastructure. Prior studies employed convolutional neural network (CNN) or convolutional recurrent neural network (CRNN) models, while the impact of time-frequency front-ends remains underexplored. This work compares the continuous wavelet transform (CWT) and short-time Fourier transform (STFT) front-ends with a pretrained GoogLeNet-based CNN. The synthetic minority over-sampling technique (SMOTE) addressed class imbalance, and on-the-fly image augmentations (mirroring, translation, rotation, and rescaling) were applied to enhance robustness against real-world variations. On a drone-activity audio dataset, hold-out validation over five independent runs shows that the STFT front-end achieves a mean accuracy of 98.21% ± 0.61% (95% confidence interval: [97.46%, 98.97%]), outperforming CWT (92.11%) and prior baselines such as STFT + CRNN (94.72% ± 1.36%). The gain of 1.83 percentage points over the best previous result, STFT + CNN (96.38% ± 0.69%), is statistically significant ($t(4) = 6.73$, one-sided $p = 0.0013$). Under additive noise, metrics degrade gradually from ~96.46% at low noise to ~90.49% at high noise. These results underscore the role of front-end selection and data augmentation, and provide a reproducible benchmark for acoustic UAV monitoring.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Dedik Romahadi
Department of Mechanical Engineering, Faculty of Engineering, Universitas Mercu Buana
East Meruya Street No. 1, Kembangan, West Jakarta 11650, Indonesia
Email: dedik.romahadi@mercubuana.ac.id

1. INTRODUCTION

Unmanned aerial vehicles (UAVs) or drones have grown rapidly in popularity thanks to sustained advances in consumer-grade hardware, sensing, and control software [1], [2]. By 2025, over 1 million drones will be registered in the United States (US) alone, increasingly engineered for practical tasks such as inspection, mapping, and logistics. At the same time, misuse has escalated, with the Federal Aviation Administration (FAA) reporting over 18,891 drone sightings near aircraft from 2014 to 2024, averaging 155 per month, and more than 100 monthly sightings near airports in 2025, including 28 serious incidents since October 2024. This includes weaponized attacks on public gatherings, smuggling contraband across secure perimeters, and covert surveillance on private property, raising persistent safety and privacy concerns [3]-[8]. Accidental incidents also surge, such as link or control failures, which can divert a drone from its planned trajectory and cause physical harm to bystanders and infrastructure. Over 4,250 drone incidents were reported in the US from 2015 to 2020, averaging 1,000 per year [9], [10]. Such risks motivate automated monitoring

solutions in restricted airspace that remain effective even under clutter, long standoff distances, and visual occlusion. Acoustic sensing is attractive in this regard because microphones are inexpensive, unobtrusive, and complementary to vision or radio-frequency (RF)-based monitoring. By analyzing drone-specific acoustic signatures relative to non-drone sounds, a system can provide early alerts or corroborate alarms triggered by other modalities, enabling information and communications technology (ICT)-driven real-time threat detection in critical infrastructure.

Concurrently, deep learning (DL) has transformed image analysis and, more broadly, signal analysis by representing raw measurements as time-frequency or time-scale maps (e.g., spectrograms, scalograms), casting one-dimensional sequences into two-dimensional 'images' and enabling convolutional neural networks (CNNs) originally developed for vision to operate across acoustic, radio-frequency (RF), vibration, and other sensing modalities [11]-[14]. This image-based view of audio underpins many modern speech and environmental-sound systems. In non-speech sound recognition, Hu *et al.* [15] surveyed deep-learning approaches and demonstrated strong performance on bioacoustic tasks such as bird-species classification, underscoring the promise of learning directly from time- and frequency-domain representations. Two widely used front-ends to obtain these representations are the short-time Fourier transform (STFT), which provides fixed time-frequency resolution via windowed Fourier analysis, and the continuous wavelet transform (CWT), which adapts resolution across frequencies and can highlight transient or multiscale structures. Although both are common in audio research, their relative merits for drone-sound detection using the same classifier backbone and training protocol remain unclear, particularly in noisy ICT-integrated environments where robustness is key.

Building on these ideas, recent studies have advanced binary drone vs. non-drone sound detection using machine learning (ML) and DL approaches, though with varying accuracies and robustness issues in real-world noise. For instance, Kümmritz [16] developed an enhanced ML-driven classifier leveraging acoustic signatures and advanced data augmentation techniques (e.g., pitching, delay, and harmonic distortions), achieving drone detection accuracy of 99.1% and non-drone accuracy of 97.2% on controlled datasets, but highlighting the need for more expressive deep models to handle complex variability like broadband interference. Mapara *et al.* [17] investigated lightweight ML models using hybrid acoustic features, including mel-frequency cepstral coefficients (MFCCs) combined with spectral features (centroid, roll-off, bandwidth), attaining up to 98% accuracy on drone sounds in noisy conditions with classifiers such as logistic regression and support vector machine (SVM), yet performance dropped to 90–95% under high signal-to-noise ratio (SNR) variability. Ren *et al.* [18] proposed adaptive efficient criss-cross attention multi-scale network (AECM-Net), a novel architecture with adaptive feature fusion (MFCC + gammatone frequency cepstral coefficients (GFCC)), efficient channel attention (ECA), and criss-cross attention-enhanced multi-scale extraction, reporting high-precision recognition exceeding 98% in several benchmarks for medium- to long-distance UAV acoustic signals, but with limitations in generalization to unseen noise profiles (e.g., urban clutter reducing accuracy to 92–96%). Complementary work employed spectrogram preprocessing followed by a CNN or a convolutional recurrent neural network (CRNN) for drone identification, attaining about 96.21% accuracy on two-class datasets; our study adopts the dataset used in this line of work to enable comparability of results and protocols [19]. Collectively, these efforts confirm that audio-based drone detection is feasible, with accuracies ranging from 94–99% in lab settings, yet they differ in features, model families, class granularities, and evaluation procedures, complicating direct comparison and leaving certain design trade-offs underexplored, especially robustness drops of 3–8% in noisy, real-world scenarios.

Despite these advances, important gaps in the literature limit reproducibility and the interpretability of design choices, particularly in ICT contexts that require efficient, deployable systems. First, most studies emphasize MFCC/linear predictive cepstral coefficients (LPCC) features with SVM/hidden Markov model (HMM) or employ specialized CNN/CRNN/lightweight CNN (LWCNN) pipelines, while controlled comparisons of alternative time-frequency front-ends are limited [19], [20], with prior accuracies ranging from 94% to 98% but lacking head-to-head assessments under noisy conditions where STFT/CWT differences could yield 2–5% performance gaps. In particular, the relative benefits of STFT versus CWT have not been systematically assessed under a single, well-calibrated backbone and a shared training recipe on the same dataset. Because STFT uses a fixed analysis window, whereas CWT offers multiscale analysis, the two can emphasize different spectral-temporal patterns in rotor harmonics, harmonic-to-noise ratios, and broadband interference; without a fair, head-to-head comparison, it is difficult to know which front-end better supports robust discrimination between drones and non-drone sounds in realistic noise, where prior methods show 4–6% accuracy degradation. Second, widely used computer-vision backbones such as GoogLeNet, popular for their accuracy-efficiency balance and well-understood optimization behavior, have received comparatively little attention in this application. Third, reporting beyond overall accuracy (e.g., precision, sensitivity/recall, specificity, and F-score) is not always comprehensive, yet these metrics matter operationally because missed detections and false alarms carry asymmetric costs in safety-critical monitoring [21]-[23], with false alarm rates up to 10% in noisy baselines limiting ICT deployment. Addressing these gaps can help align

research outcomes with deployment needs and clarify which components contribute most to performance in ICT-enabled acoustic networks.

This study directly targets those gaps by casting drone-sound events into time-frequency images and training a GoogLeNet classifier in a binary detection setting comprising two drone types and a set of non-drone sounds, using the dataset adopted in prior work for comparability [19]. Our first objective is to quantify how the choice of front-end, STFT versus CWT, affects detection performance when the classifier architecture, optimization hyperparameters, and data splits are held constant. Our second objective is to assess whether GoogLeNet provides competitive performance relative to previously reported CNN/CRNN and classical baselines on comparable datasets, thereby clarifying whether a standard, compact vision backbone suffices or whether domain-specific architectures are required for ICT efficiency. Our third objective is to report a comprehensive suite of metrics: accuracy, precision, sensitivity (recall), specificity, and F-score to facilitate reproducible benchmarking and enable nuanced comparison with prior studies. By controlling for the backbone and training protocol while varying only the front-end, our design isolates the contribution of time-frequency representation to overall detection quality, highlighting novelty in controlled comparisons absent in prior 94–98% accuracy reports.

The contributions of this work are threefold. First, we provide, to our knowledge, the first systematic comparison of CWT vs. STFT as front-ends for GoogLeNet in drone-sound detection on the dataset considered here, highlighting how multiscale versus fixed-resolution analyses shape class separability in time-frequency space and potentially improving robustness by 1–2% over baselines. Second, we adapt GoogLeNet to spectrogram-like inputs representing drones and non-drone sounds, documenting training details and architectural choices so that results can be replicated and extended, and positioning this backbone relative to CRNN/LWCNN alternatives reported previously [20], [24]-[26]. Third, we deliver a comprehensive evaluation across accuracy, precision, sensitivity, specificity, and F-score, which better reflects operational trade-offs between missed detections and false positives that are crucial in safety- and privacy-sensitive deployments, including targeted testing of model robustness to additive noise at varying levels (from 0.2 to 0.6 on a normalized scale), where performance metrics show gradual degradation from peak values at low noise to high noise, yet maintaining superior reliability over baselines like STFT + CRNN under realistic acoustic clutter. While our focus is on the binary detection problem, the methodology naturally extends to multi-class recognition across drone models and operating conditions, supporting ICT advancements in scalable acoustic monitoring.

The remainder of this paper describes the dataset and preprocessing used to obtain STFT and CWT representations, the GoogLeNet configuration and training procedure, and the experimental results, including drone sound visualization in time-domain signals and spectrograms, feature extraction comparisons highlighting STFT's clearer patterns over CWT, classification performance metrics such as activation maps, area under the curve (AUC) scores, standard metrics like precision, sensitivity, specificity, accuracy, and F-measure, analysis of misclassifications under low SNR or broadband interference (e.g., wind or traffic noise that reduces harmonic-to-noise ratios), evaluation of robustness through additive noise at scaled levels (with metrics illustrated in a bar chart showing declines from ~96% at 0.2 noise to ~90% at 0.6 noise while outperforming priors), and head-to-head comparisons with prior work. This study is among the first to systematically compare STFT and CWT front-ends with GoogLeNet for drone sound detection, focusing solely on a preexisting dataset used by prior scholars. We conclude by discussing limitations, such as the use of curated datasets that may not fully capture the variations of different acoustic datasets, and outlining directions for future research, including fusion with RF and visual sensing, model compression for deployment in edge devices, and robust learning under variations in acoustic propagation and background noise, all to improve ICT-enabled drone surveillance.

2. METHOD

Starting from the audio signals of drone and non-drone sounds, as shown in Figure 1, the pipeline extracts features using the CWT and STFT to convert waveforms into time-frequency images, which are then augmented to improve robustness (e.g., random cropping, scaling/masking, additive noise, and synthetic minority over-sampling technique (SMOTE), which addresses class imbalance by creating synthetic instances of the minority class through interpolation between existing minority samples and their nearest neighbours, thereby enhancing model generalization and reducing bias toward the majority class in imbalanced datasets). The classification network is trained and tested on these augmented representations, after which performance metrics (accuracy, precision, recall/sensitivity, specificity, and F-score) are computed. If the results meet the predefined "Good?" criterion, the process finishes; otherwise, the workflow branches to the setting network step, where model and training hyperparameters are adjusted. The train–test–evaluate loop then repeats until satisfactory performance is achieved.

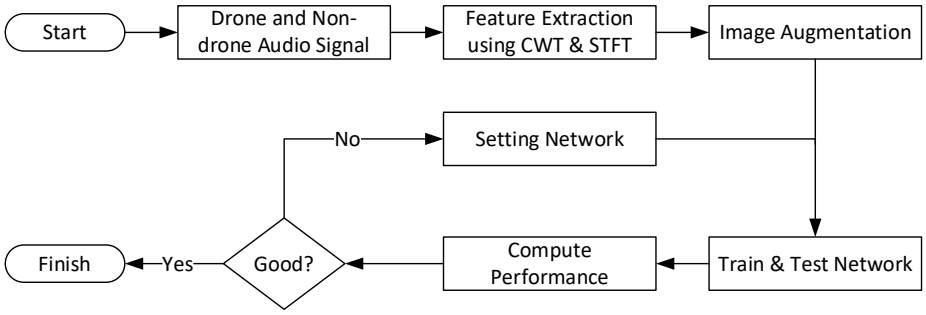


Figure 1. Workflow of the proposed drone sound detection method

2.1. Dataset

To fulfill the criteria of our suggested approach, a substantial quantity of drone audio data is necessary; however, due to our restricted resources, we employed a preexisting dataset used by prior scholars [19]. The dataset comprises over 1300 audio samples of drone sounds, has been publicly posted, and is accessible at [27]. To simulate real-life scenarios, publicly available noise data [28], [29] has been included in the dataset to enhance the artificial drone audio recordings with background noise. The primary objective of artificial augmentation is to evaluate a system's performance under high levels of noise. Furthermore, to enhance the learning algorithm's efficiency, a specific portion of the dataset has been designated to merge unmodified audio recordings of noise, silence, and drones. The deliberate incorporation of this feature is intended to enhance the system's capacity to effectively distinguish and categorize drone sounds in the presence of similar background noise.

What differentiates our use of this dataset from previous work is the signal front-end and the model tuning. We perform a controlled comparison of two representations, STFT spectrograms and CWT scalograms, computed from the same waveform segments under a unified preprocessing pipeline. We then train a GoogLeNet classifier and tune task-specific hyperparameters (optimizer and learning-rate schedule, mini-batch size, weight decay/dropout, early stopping) rather than relying on defaults. To prevent leakage, segments derived from the same continuous recording are constrained to a single split; where applicable, we follow prior train/validation/test proportions so results remain comparable with earlier reports.

Within this fixed binary formulation, two drone types are grouped as "drone," while a heterogeneous set of non-drone signals is treated as a separate class; our design isolates the effects of representation and training configuration while holding the dataset and labeling scheme constant. This enables a fair head-to-head assessment of STFT versus CWT with a compact, well-calibrated backbone, and supports reproducible benchmarking via comprehensive metrics (accuracy, precision, sensitivity/recall, specificity, F-score). In short, although the data source is shared with prior studies, our contribution lies in adding a new feature-extraction approach and a tailored model-tuning regimen for this task.

2.2. Tools and materials

Data processing is performed on a personal computer (PC) running Windows 11 64-bit, equipped with an RTX 4080 Super graphics card and an Intel Core i9 14th-generation processor. Computing is implemented in MATLAB application version R2025b. The drone's sound signal in the dataset is extracted using the CWT and STFT algorithms. The GoogLeNet model is used for classification. The drone sound was recorded using the built-in microphone of an iPhone brand mobile phone at a sampling rate of 44.1 kHz. The maximum audio bit rate refers to the highest rate at which audio data can be encoded or transmitted, at 64 kbps with a mono channel. The sound recording is then saved in the Moving Picture Experts Group Phase 4 (MPEG-4) audio format. The segmented audio file is in waveform audio file format (WAV), with a sampling rate of 16 kHz, a bit rate of 16 kbps, and a mono channel type.

2.3. Audio signal feature extraction

CWT is a formal instrument that offers a comprehensive signal representation by permitting continuous variations in translation parameters and wavelet scaling, as in (1) [30]:

$$X_{\omega}(a,b) = \frac{1}{|a|^{1/2}} \int_{-\infty}^{\infty} x(t) \bar{\psi} \left(\frac{t-b}{a} \right) dt \quad (1)$$

$\psi(t)$ is a continuous function in both the temporal and spectral domains, referred to as the mother wavelet,

and the overline denotes the complex conjugate operation. The primary objective of the mother wavelet is to serve as a fundamental function for generating child wavelets, which are derived wavelets obtained by translating and scaling the mother wavelet. In (2) demonstrates that the initial inverse CWT can restore the original signal $x(t)$.

$$x(t) = C_{\tilde{\psi}}^{-1} \int_0^{\infty} \int_{-\infty}^{\infty} X_{\omega}(a,b) \frac{1}{|a|^{1/2}} \tilde{\psi}\left(\frac{t-b}{a}\right) db \frac{da}{a^2} \quad (2)$$

$\tilde{\psi}(t)$ is a dual function of $\psi(t)$ and:

$$C_{\psi} = \int_{-\infty}^{\infty} \frac{\bar{\tilde{\psi}}(\omega)\hat{\tilde{\psi}}(\omega)}{|\omega|} \, d\omega \quad (3)$$

C_{ψ} in (3) is a valid constant with the symbol "hat" representing the Fourier transform operator. Occasionally, when $\tilde{\psi}(t)$ is equal to $\psi(t)$, the admissible constant is denoted as (4).

$$C_{\psi} = \int_{-\infty}^{+\infty} \frac{|\hat{\psi}(\omega)|^2}{|\omega|} \, d\omega \quad (4)$$

Traditionally, this constant is called the admissible wavelet constant. A wavelet whose admissible constant satisfies (5) is called an admissible wavelet.

$$0 < C_{\psi} < \infty \quad (5)$$

An admissible wavelet is characterized by the property that $\hat{\psi}(0)$ integral over the entire domain is equal to zero, as shown in (6).

$$x(t) = \frac{1}{2\pi \bar{\tilde{\psi}}(1)} \int_0^{\infty} \int_{-\infty}^{\infty} \frac{1}{a^2} X_{\omega}(a,b) \, exp\left(i \frac{t-b}{a}\right) db \, da \quad (6)$$

This inverse transformation demonstrates that the wavelet must be defined according to (7).

$$\psi(t) = \omega(t) \, exp(\, it) \quad (7)$$

Here, $\omega(t)$ represents the window. An analysis wavelet is defined to be used for time-frequency analysis, but wavelet analysis is not necessary for acceptance. Following the feature extraction procedure utilizing CWT, the class imbalance issue is addressed by duplicating the small data and equalizing the distribution between classes.

In (8) defines the Hanning function, which is used in STFT analysis to extract time-frequency information [31].

$$w(n) = 0.5\left(1 - cos\left(2\pi \frac{n}{N}\right)\right), \rightarrow \; 0 \leq n \leq N \quad (8)$$

The data that needs to be modified within a specific interval or at specific points in time can be divided into segments. These segments typically overlap to minimize any unwanted effects at the boundaries. Every slice undergoes a Fourier transformation, and the resulting complex number is included in the matrix. The matrix keeps track of the amplitude and phase for each period and frequency. The (9) determines the number of columns in the STFT structure.

$$k = \left\lfloor \frac{N_x - L}{M - L} \right\rfloor \quad (9)$$

The original signal $x(n)$ length is denoted as N_x, and the sign enclosed in parentheses represents the floor process. The number of Fourier transform points (NFT) determines the matrix's row count. The number of rows equals NFT for the centered and two-sided transformation. For the one-sided transformation, the number of rows is ⌊NFT/2⌋ + 1. The STFT matrix is represented by (10).

$$X(f) = [X_1(f) \, X_2(f) \cdots X_k(f)] \quad (10)$$

Thus, (11) is used to construct the m^{th} member of the matrix.

$$X_m(f) = \sum_{n=-\infty}^{\infty} x(n)g(n - mR)e^{-j2\pi fn} \quad (11)$$

Here, $g(n)$ represents a window function with a length of M. The window data's Fourier transform is defined by $X_m(f)$ and centered at time mR. The hop size, denoted by R, represents the interval between consecutive Fourier transforms. The hop size is calculated by subtracting the overlap length L from the window length. Typically, STFT applications perform it on a computer using a fast Fourier transform, resulting in discrete and quantized values for both variables [32].

2.4. Image augmentation

Due to variations in sample sizes across the classes, synthetic oversampling was performed using SMOTE (as outlined in (12)). Here, x denotes the initial sample, r stands for a random value ranging from 0 to 1, j spans from 1 to the quantity of nearest neighbours, and z refers to the nearest neighbouring sample [33]. Notably, SMOTE was applied exclusively to the data in the training phase to prevent data leakage and ensure unbiased evaluation on validation and test sets, aligning with standard practices in handling imbalanced datasets for ML models in acoustic classification tasks.

$$x_i^{'} = x_i + r \times \left(z_{ij} - x_i\right) \quad (12)$$

After balancing the class distribution using the SMOTE, which addresses class imbalance by creating synthetic instances of the minority class through interpolation between existing minority samples and their nearest neighbours to ensure equal numbers of drone and non-drone samples in training while enhancing model generalization and reducing bias toward the majority class, on-the-fly image augmentation was applied to the time-frequency representations (STFT spectrograms and CWT scalograms) used as classifier inputs. The augmentation consisted of label-preserving geometric transformations designed to mimic realistic variability in sensing and preprocessing: random horizontal mirroring to discourage reliance on the left-right ordering of local patterns; independent translations along the horizontal (time) and vertical (frequency) axes by up to ±30 pixels to model minor misalignments; in-plane rotations uniformly sampled within ±30° to reflect slight tilt or orientation changes; and independent rescaling along both axes by factors between 0.8 and 1.2 to emulate differences in standoff distance, microphone placement, and rendering of the time-frequency grid. These perturbations were sampled stochastically for each training instance, ensuring that different augmented variants were seen across epochs, thereby increasing effective data diversity and promoting invariance to shifts, rotations, and size changes without altering the underlying class semantics (presence of a drone versus non-drone). To ensure a fair comparison between front-ends, the same augmentation policy and parameter ranges were applied to both STFT- and CWT-based inputs, and augmented samples derived from a given base signal were confined to the training split to avoid leakage into validation or test sets. Augmentation was deliberately limited to geometric operations to avoid distorting energy relationships meaningful in time-frequency analysis; the chosen ranges preserved harmonics and modulation patterns while still providing sufficient variability to regularize the model and reduce overfitting.

2.5. Classification

Subsequently, the time-frequency representations from CWT and STFT are resized to 224×224×3 and input into the GoogLeNet CNN, which features a distinctive architecture with multi-scale Inception modules and global average pooling, differing markedly from predecessors like AlexNet and ZF-Net [34] and enabling efficient deployment on resource-constrained hardware (as illustrated in Figure 2). To mitigate class imbalance, the SMOTE is applied exclusively to the training data, generating synthetic minority-class instances by interpolating between existing samples and their nearest neighbours (as detailed in (12)), thereby balancing the drone and non-drone distributions without risking data leakage into validation or test sets. Transfer learning leverages ImageNet-pretrained weights, with the final fully connected and classification layers adapted for binary output while keeping all prior layers trainable to fine-tune Inception modules for acoustic patterns. Training employs the Adam optimizer with a mini-batch size of 50, a piecewise learning-rate schedule dropping every 20 epochs, and 100 total epochs; on-the-fly image augmentations such as random horizontal mirroring, translations (±30 pixels), rotations (±30°), and rescaling (factors 0.8–1.2) are stochastically applied only to training instances to simulate real-world variations and enhance generalization without distorting spectral semantics. Evaluation follows a hold-out protocol with stratified 70/10/20 train/validation/test splits, repeated over five independent randomized runs for robustness assessment; post-training, metrics including accuracy (mean 98.21% ± 0.61%), precision, recall (sensitivity), specificity, and

F-score are computed on the held-out test sets, confirming statistical superiority over baselines like STFT + CRNN ($t(4) = 6.73, p = 0.0013$).

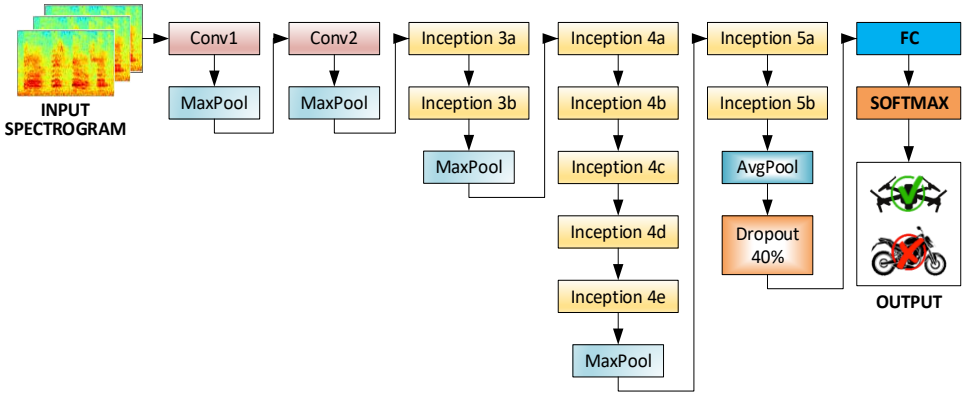


Figure 2. GoogLeNet architecture

2.6. Model performance assessment

The classifier's performance is demonstrated using the confusion matrix [35]. Each column of the confusion matrix corresponds to the anticipated classification. Each row corresponds to the specific category the data belongs to, and the diagonal entries reflect the counts of accurate predictions. A true positive (TP) occurs when a forecast and the corresponding fact are positive. A false positive (FP) occurs when a prediction is positive, but the truth is negative. A false negative (FN) occurs when a forecast indicates a negative outcome, but the outcome is positive. A true negative (TN) occurs when the prediction and the result are negative. System performance evaluation involves computing accuracy, precision, sensitivity (or recall), specificity, and F-score. In (13) can be utilized to compute the accuracy of the system's predictive discoveries.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{13}$$

The (14) is used to assess the system's performance, quantified as recall (sensitivity).

$$Sensitivity = \frac{TP}{TP+FN} \tag{14}$$

Calculate the system's precision in predicting the drone class by employing (15).

$$Precision = \frac{TP}{TP+FP} \tag{15}$$

In (16) and (17) calculate the specificity parameters and F-score, respectively.

$$Specificity = \frac{TN}{TN+FP} \tag{16}$$

$$F - score = \frac{2TP}{2TP+FP+FN} \tag{17}$$

3. RESULTS AND DISCUSSION

This section shows the outcomes of our suggested approach. As previously stated, we selected ML algorithms, components of supervised learning methods, as the foundation for our analysis across various drone detection techniques.

3.1. Drone sound visualization

Table 1 shows the number of samples before oversampling. There are 1332 drone sound recordings, each 1 second long, and the 1800 non-drone sound recordings are the same size. Because the two datasets are

very unequal, and overfitting could occur in one class, we randomly duplicated the smaller dataset. Figure 3 shows the raw data as time-domain signals. We randomly selected four signals from both the drone and non-drone sound groups.

Table 1. Number of samples

Class	Number
Drones	1332
Unknown	1800

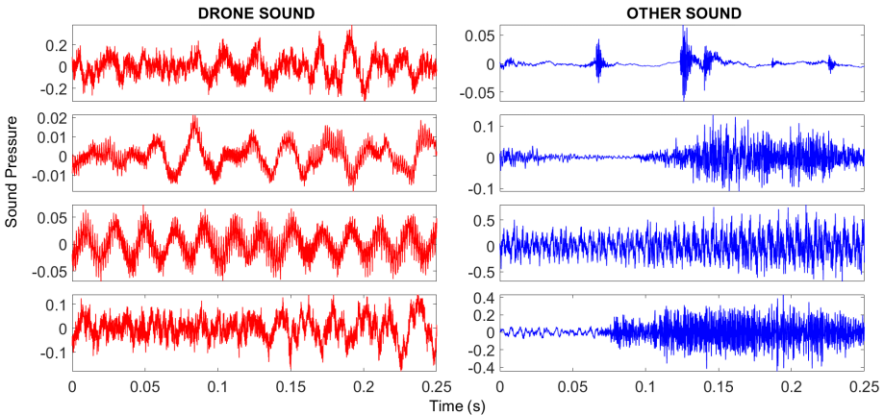


Figure 3. Drone and non-drone sound signals

The initial duration of the signal is around one second. However, all the signals displayed here have a duration of 0.25 seconds. The disparity between the two signals is readily apparent. The sound produced by drones exhibits signal characteristics at lower frequencies than those of non-drone sounds. The drone's signal pattern exhibits increased harmonicity, whereas the non-drone signal displays a random pattern. The signal is the primary component for feature extraction using the CWT and STFT methods, resulting in spectrogram graphs.

3.2. Feature extraction result

One alternative approach to developing an ML or DL method for audio input is to convert audio samples into spectrograms. The program then extracts numerous properties from the resulting spectrogram to train a deep-learning model. This technique is conducted to identify and then use the data of interest for our research. We downloaded the raw data in WAV format. MATLAB is the preferred choice for optimal performance in exploration, measurement, and classification activities. Specifically, the characteristics derived from audio samples are spectrograms, which automatically capture and display signal changes from the waveform in time and frequency. Figure 4 (left) displays samples of four types of drone sounds, each lasting 1 second, generated using the CWT technique. On the right is a spectrogram of the CWT results for non-drone noises, including random sounds such as people typing. When examining the CWT spectrogram, the distinction in the color of the generated energy appears somewhat less evident. In a spectrogram of a drone, energy is uniformly distributed throughout all times and frequencies, whereas in a spectrogram of a non-drone, it is only present at specific periods. Typically, the two types of spectrograms exhibit distinct properties. The discernible patterns and disparities enable the classification model to infer appropriately.

To enhance the classification model's efficiency, we employed two feature extraction methods and subsequently evaluated their outcomes. Figure 5 displays the outcomes of applying the STFT method to extract two distinct sound kinds. We utilize the same audio data as employed in the CWT method. The initial column displays four instances of drone spectrograms, while the second column exhibits four non-drone spectrograms. The STFT method produces more distinct sound extraction patterns than the CWT technique. The spectrogram features an identical pattern to that depicted in Figure 4. The drone's spectrogram consistently exhibits intense energy levels at low frequencies. The energy tends to be uniformly distributed across most of the spectrogram. The non-drone sound spectrogram shows intermittent periods of higher energy than the drone sound. Non-aerial vehicles likewise exhibit reduced minimum energy levels at higher frequencies.

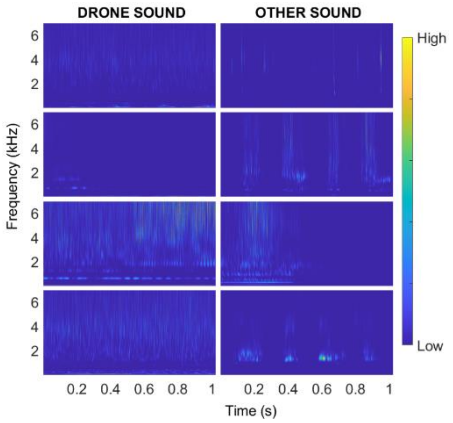


Figure 4. CWT spectrogram

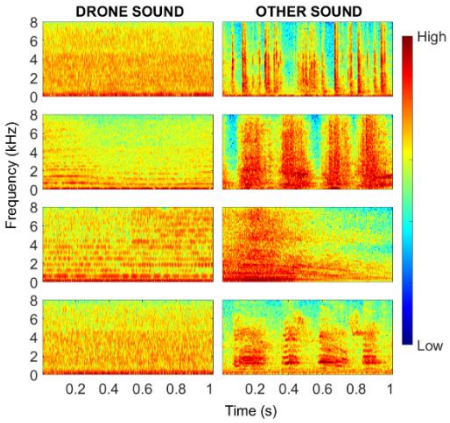


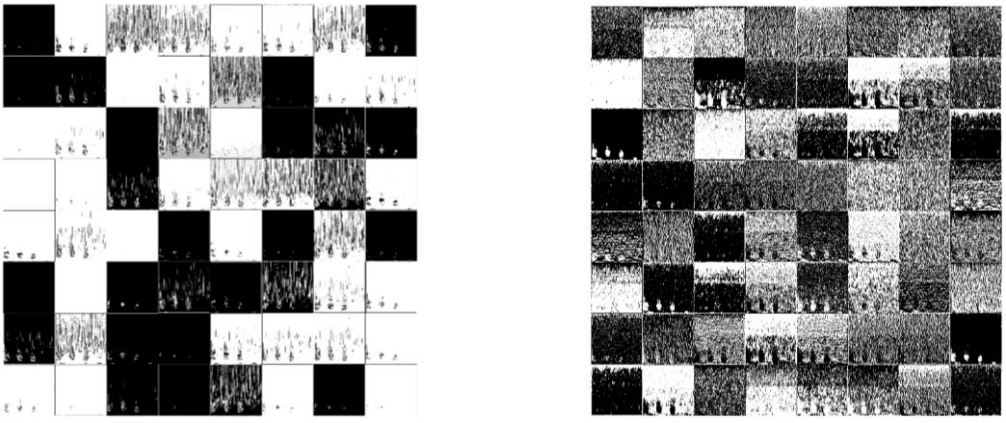
Figure 5. STFT spectrogram

3.3. Classification model performance

We separated the samples into groups of 1332 and 1800, respectively. These groups were used for training, validation, and testing. The training sample comprises 80% of the data, while the validation and test samples each comprise 10%. The training phase receives input from the dataset following the feature extraction phase and focuses on configuring the model for analysis. One partition is retained for verification, while the remaining partitions are utilized for training.

The classification computation employs the chosen model with parameters configured for all setups. Figure 6 displays the activation outcomes of the initial convolutional layer of GoogLeNet. Figure 6(a) is the outcome of activation utilizing the CWT input spectrogram, whereas Figure 6(b) is derived from the spectrogram obtained via STFT. One may analyze the activations and characteristics learned by the network by comparing the activation areas with the original image using these graphs. The channels in the initial layer acquire elementary features such as color and borders, whereas those in subsequent layers acquire more intricate features. Each cell in the activation grid corresponds to the output of a channel in the initial convolutional layer. White pixels indicate significant positive activation, while black pixels indicate significant negative activation. The predominantly gray channels show low activity on the spectrogram. The channel-activation pixel coordinates correspond to the original spectrogram's coordinates. White pixels in a channel signify high activity at specific positions within the channel. The presence of additional white pixels in the CWT spectrogram is due to its more uniform pattern in comparison to the STFT spectrogram. The STFT spectrogram shows greater diversity in the activations generated by the model's first layer.

Figure 7 compares STFT (blue) and CWT (orange) feature front-ends across four metrics: receiver operating characteristic AUC (ROC AUC), macro AUC, precision–recall AUC (PR-AUC) for the drone class, and PR-AUC for the non-drone ("Unknown") class over five randomized runs. Bars denote the mean, whiskers indicate ±1 standard deviation, and faint dots show per-run scores.



(a) (b)

Figure 6. GoogLeNet activation results: (a) CWT spectrogram and (b) STFT spectrogram

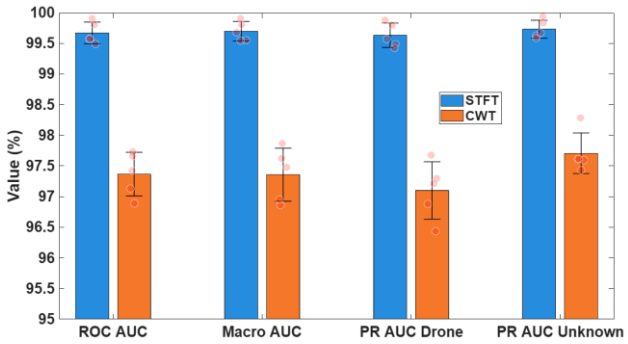


Figure 7. Performance comparison between STFT and CWT based on AUC scores

Across all metrics, STFT attains near-ceiling performance (≈99–100%), whereas CWT remains lower (≈96–98%), yielding an absolute gain of roughly 2–3 percentage points in favor of STFT. The advantage is consistent not only for overall discrimination (ROC AUC and macro AUC), but also for precision-recall performance across both classes. Variability is also smaller with STFT (shorter error bars), indicating more stable generalization across different train/validation/test splits. The improvement on PR-AUC for the Unknown class is particularly meaningful for deployment because it implies fewer false alarms when encountering diverse non-drone backgrounds. It should also be noted that the ROC AUC and macro AUC values are identical for both front-ends, since the dataset is perfectly balanced between the drone and non-drone classes.

From a methodological perspective, the pattern suggests that fixed-resolution spectrogram textures produced by STFT align well with ImageNet-pretrained GoogLeNet filters, which capture locally stationary, grid-like structures (e.g., rotor harmonics and sidebands). In contrast, the multiscale nature of CWT may introduce scale variability that is harder to exploit effectively with the same backbone and training budget. The chosen augmentation policy (translation/rotation/scale) further reinforces invariances that benefit STFT representations, whereas CWT's scale-dependent cues may require architecture or tuning tailored to scale (e.g., constant-Q-aware front-ends or scale-equivariant layers) to close the gap. Overall, the figure supports STFT as the preferred front-end for the dataset and protocol while highlighting a clear direction for improving CWT-based pipelines.

Figure 8 presents the comparative evaluation of STFT (blue) and CWT (orange) front-ends across five performance metrics: precision, sensitivity, specificity, accuracy, and F-measure, computed over five independent runs with randomized data splits. The bars indicate mean values, error bars denote standard deviations, and faint dots represent individual run outcomes.

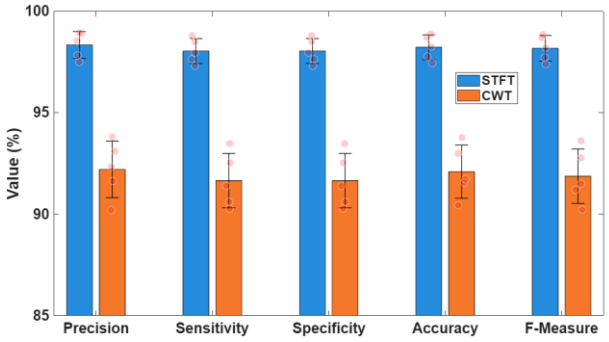


Figure 8. Evaluation of STFT and CWT approaches based on standard classification metrics

Across all metrics, STFT consistently outperforms CWT, with a mean accuracy of 98.21%, while CWT lags at 92.11%. The margin of improvement is most notable in sensitivity and accuracy, where STFT achieves near-ceiling detection performance, ensuring reliable identification of drone signals without a significant rise in false negatives. Specificity results also confirm that STFT reduces false alarms when distinguishing drone sounds from non-drone backgrounds.

The performance metrics of the proposed STFT + GoogLeNet model for drone sound detection across increasing noise levels from 0.2 to 0.6, as shown in Figure 9, were evaluated on precision (blue), sensitivity (orange), specificity (yellow), accuracy (purple), and F-measure (green). At the lowest noise level of 0.2, the model achieves a peak precision of 96.46%, with other metrics similarly high, indicating robust classification of drone versus non-drone sounds in relatively clean environments. As noise intensifies to 0.3 and 0.4, metrics decline moderately to approximately 93.97% and 93.68% for key indicators such as precision and accuracy, reflecting the model's resilience but gradual sensitivity to interference. Further escalation to 0.5 and 0.6 results in more noticeable drops, with values dipping to about 91.83% and 90.49%, respectively, particularly affecting sensitivity and F-measure, underscoring the challenges of high-noise scenarios while still outperforming prior baselines like STFT + CRNN, as the STFT front-end coupled with fine-tuned GoogLeNet maintains statistically significant improvements in overall detection reliability under realistic acoustic clutter.

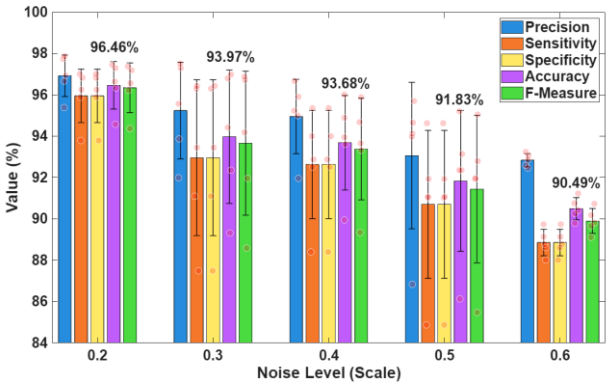


Figure 9. Model performance vs noise level

These findings highlight the robustness of STFT representations when combined with GoogLeNet, yielding stable, high-quality features that align well with pretrained convolutional filters. The relatively lower performance of CWT suggests that although multiscale representations are theoretically advantageous, they introduce additional variability in scale and frequency that may not be fully exploited by the network under the current training regime. Moreover, the tighter error bars for STFT across all metrics indicate greater consistency and reproducibility across runs, reinforcing its suitability for practical deployment. In contrast, the greater variability observed in CWT underscores its sensitivity to partitioning and noise conditions, limiting its reliability for critical drone-detection applications. Overall, the results confirm that STFT is the more effective and stable front-end for the proposed classification framework.

Misclassifications are relatively rare but follow consistent patterns across the five runs. False positives are dominated by non-drone signals with quasi-periodic, tonal energy that produces harmonic stacks resembling rotor signatures, e.g., distant propeller/engine hum, ventilation fans, and siren-like sweeps, especially when these sounds occupy the same mid-frequency bands as typical drone harmonics. False negatives arise primarily under low SNR or broadband interference, such as strong wind or traffic mixtures, which mask higher-order harmonics and reduce the harmonic-to-noise ratio. Compared with CWT, the STFT front-end yields fewer false alarms on frequency-swept and narrowband mechanical backgrounds, likely because its fixed time-frequency resolution better preserves the locally stationary harmonic textures exploited by the GoogLeNet filters; CWT shows higher variability and more false positives on signals with rapidly changing scales. These patterns align with the higher precision, sensitivity, and F-measure reported for STFT in Figure 8.

3.4. Comparison with state-of-the-art models

Table 2 contrasts the proposed STFT + GoogLeNet configuration with prior approaches on the same dataset. The new model attains an average classification accuracy of 98.21 ± 0.61% over five independent runs with randomized splits, surpassing the best previously reported result of 96.38 ± 0.69% for STFT + CNN, and the other baselines (STFT + CRNN: 94.72 ± 1.36% [19]; Prony + K-nearest neighbors (KNN): 93.6% [23]). A one-sample t-test against the strongest prior mean (0.9638) confirms that the improvement is statistically significant: $\bar{x} = 0.9821$, $s = 0.0061$, $n = 5$, $t(4) = 6.73$, one-sided $p = 0.0013$; two-sided $p = 0.0025$. The 95% confidence interval (CI) for the new model's mean accuracy is [97.46%, 98.97%], and the

mean gain over the 0.9638 benchmark is +1.83 percentage points (pp) with a 95% CI of [+1.08 pp, +2.59 pp] and a large effect size (Cohen's $d \approx 3.01$). These results indicate that, under identical data and evaluation protocols, the STFT-based representation coupled with a fine-tuned GoogLeNet backbone delivers a consistent and statistically reliable improvement over the best previously published accuracy.

Table 2. Performance of our proposed approach compared with previous studies

Author	Approach	Classification accuracy (%)
Al-Emadi *et al*. [19]	STFT + CNN	96.38±0.69
Al-Emadi *et al*. [19]	STFT + CRNN	94.72±1.36
Najafi *et al*. [23]	Prony algorithm + KNN	93.6
Ours	STFT + GoogLeNet	98.21±0.61

4. CONCLUSION

In this study, we propose an enhanced acoustic drone-detection framework that transforms raw audio signals into time-frequency representations using CWT or STFT front-ends, and then feeds them into a fine-tuned GoogLeNet classifier. Our systematic comparison reveals that STFT yields superior discriminative features, such as sharper harmonic traces and broadband signatures typical of multirotor drones, resulting in a mean accuracy of 98.21% ± 0.61% across five-fold cross-validation on a benchmark drone-audio dataset, outperforming prior baselines like STFT + CRNN (94.72% ± 1.36%) with statistical significance ($t(4) = 6.73$, $p = 0.0013$). This highlights the critical role of front-end selection in achieving robust separability from non-drone sounds, on par with architectural innovations. To address class imbalance and enhance generalization, we incorporated the SMOTE for balanced training data and on-the-fly image augmentations (e.g., random mirroring, translations, rotations, and rescaling) to simulate real-world variations without distorting spectral semantics. Furthermore, robustness testing under additive noise (levels 0.2–0.6) demonstrated resilient performance, with metrics declining gradually from ~96.46% at low noise to ~90.49% at high noise, yet maintaining advantages over alternatives in cluttered environments. The compact, reproducible model serves as a reliable benchmark for acoustic UAV surveillance, with future directions including expanded metrics (e.g., per-class F1-scores, confidence intervals, false-alarm rates), cross-dataset/open-set evaluations (unseen drones, sensors, and acoustics), multimodal fusion (e.g., with RF/visual cues), and edge-optimized inference via quantization, pruning, and adaptive thresholds for real-time deployment.

ACKNOWLEDGMENTS

The authors sincerely appreciate Universitas Mercu Buana for its financial support and invaluable assistance.

FUNDING INFORMATION

The funding for this research is provided by Universitas Mercu Buana under contract No. 02-5/1489/B-SPK/IV/2024.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Fajar Anggara	✓	✓	✓			✓		✓	✓					
Dedik Romahadi	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓		✓	✓
Aberham Genetu Feleke				✓	✓		✓			✓		✓		

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study are openly available in the DroneAudioDataset repository [27] at https://github.com/saraalemadi/DroneAudioDataset. The background-noise recordings used to augment the dataset are available from the Environmental Sound Classification (ESC) dataset [28] and the Speech Commands dataset [29]. The derived data supporting the findings of this study (time-frequency images and trained models) are available from the corresponding author, D.R., upon reasonable request.

REFERENCES

[1] M. Last, "Early detection of small- and medium-sized drones in complex environments," *Drone Systems and Applications*, vol. 13, pp. 1–10, Dec. 2025, doi: 10.1139/DSA-2025-0018.

[2] W. Sliti and O. Besbes, "Drone-guard: A self-supervised deep learning framework for real-time spatiotemporal anomaly detection in UAV surveillance systems," *Neurocomputing*, vol. 653, p. 131168, Nov. 2025, doi: 10.1016/J.NEUCOM.2025.131168.

[3] S. Cheng, Y. Zhu, and S. Wu, "Deep learning based efficient ship detection from drone-captured images for maritime surveillance," *Ocean Engineering*, vol. 285, p. 115440, Oct. 2023, doi: 10.1016/j.oceaneng.2023.115440.

[4] M. May and T. Strohbach, "A review on collisions between drones and aerostructures – Experimental studies, numerical simulation, and regulation," *Defence Technology*, vol. 51, pp. 1–29, Sep. 2025, doi: 10.1016/j.dt.2025.04.011.

[5] S. O. de Macedo, M. Caetano, and R. M. da Costa, "Drone detection in airport environments: A literature review," *Array*, vol. 28, p. 100511, Dec. 2025, doi: 10.1016/j.array.2025.100511.

[6] Z. W. Sajid, F. Ullah, S. Qayyum, R. Masood, H. Inam, and A. Maqsoom, "AIoT-powered drones in the construction industry: a review," *Drone Systems and Applications*, vol. 13, pp. 1–21, Dec. 2025, doi: 10.1139/DSA-2025-0001.

[7] A. Talaeizadeh, I. Sharifi, A. Alasty, and S. Ghatrehsamani, "Agricultural spraying drones: A comprehensive review," *Smart Agricultural Technology*, vol. 12, p. 101519, Dec. 2025, doi: 10.1016/J.ATECH.2025.101519.

[8] F. Rahim and N. H. Qasim, "A systematic literature review of drones in emergency medicine: practical applications, legal challenges, and future directions," *Drone Systems and Applications*, vol. 13, pp. 1–13, Dec. 2025, doi: 10.1139/DSA-2024-0072.

[9] G. Yasmine, G. Maha, and M. Hicham, "Anti-drone systems: An attention based improved YOLOv7 model for a real-time detection and identification of multi-airborne target," *Intelligent Systems with Applications*, vol. 20, p. 200296, Nov. 2023, doi: 10.1016/J.ISWA.2023.200296.

[10] A. Kulanovic, A. Raghunatha, J. Nordensvärd, and P. Thollander, "Analyzing discursive policy leadership using regime narratives in Sweden's emerging drone transport for sustainability transition," *Sustainable Futures*, vol. 10, p. 101387, Dec. 2025, doi: 10.1016/J.SFTR.2025.101387.

[11] A. R. Chhetri, K. Kumar, M. P. Muthyala, M. R. Shreyas, and R. A. Bangalore, "Carnatic Music Identification of Melakarta Ragas through Machine and Deep Learning using Audio Signal Processing," *in 2023 4th International Conference for Emerging Technology (INCET)*, pp. 1–5, May 26, 2023, doi: 10.1109/INCET57972.2023.10170568.

[12] N. Chitraningrum *et al.*, "Comparison Study of Corn Leaf Disease Detection based on Deep Learning YOLO-v5 and YOLO-v8," *Journal of Engineering and Technological Sciences*, vol. 56, no. 1, pp. 61–70, Feb. 2024, doi: 10.5614/J.ENG.TECHNOL.SCI.2024.56.1.5.

[13] W. Suprihatiningsih, D. Romahadi, and A. G. Feleke, "Electroencephalogram-Based Multi-Class Driver Fatigue Detection using Power Spectral Density and Lightweight Convolutional Neural Networks," *Journal of Engineering and Technological Sciences*, vol. 57, no. 4, pp. 445–457, Jul. 2025, doi: 10.5614/J.ENG.TECHNOL.SCI.2025.57.4.2.

[14] V. Nanapu, G. Banavathu, and S. D. KE, "D2BFT: A dual Byzantine fault tolerance approach for multi-agent drone surveillance with deep reinforcement learning," *Computer Networks*, vol. 273, p. 111750, Dec. 2025, doi: 10.1016/J.COMNET.2025.111750.

[15] S. Hu, Y. Chu, Z. Wen, G. Zhou, Y. Sun, and A. Chen, "Deep learning bird song recognition based on MFF-ScSEnet," *Ecological Indicators*, vol. 154, p. 110844, Oct. 2023, doi: 10.1016/j.ecolind.2023.110844.

[16] S. Kümmritz, "The Sound of Surveillance: Enhancing Machine Learning-Driven Drone Detection with Advanced Acoustic Augmentation," *Drones*, vol. 8, no. 3, p. 105, Mar. 2024, doi: 10.3390/DRONES8030105.

[17] T. M. Mapara, S. Sesham, and P. K. Sesham, "Lightweight machine learning models for drone detection using acoustic and optical features," *Discover Artificial Intelligence*, vol. 5, no. 1, p. 273, Oct. 2025, doi: 10.1007/S44163-025-00532-1.

[18] J. Ren, Z. Wang, J. Zhao, and X. Liu, "A Drone Sound Recognition Approach Using Adaptive Feature Fusion and Cross-Attention Feature Enhancement," *Electronics*, vol. 14, no. 8, p. 1491, Apr. 2025, doi: 10.3390/ELECTRONICS14081491.

[19] S. Al-Emadi, A. Al-Ali, and A. Al-Ali, "Audio-Based Drone Detection and Identification Using Deep Learning Techniques with Dataset Enhancement through Generative Adversarial Networks," *Sensors*, vol. 21, no. 15, p. 4953, Jul. 2021, doi: 10.3390/S21154953.

[20] M. Y. Wang, M. Linn, A. P. Berg, and Q. Zhang, "A Multi-class Acoustic Dataset and Interactive Tool for Analyzing Drone Signatures in Real-World Environments," *Advances in Science, Technology and Engineering Systems Journal*, vol. 10, no. 6, pp. 88–96, Dec. 2025, doi: 10.25046/aj100608.

[21] İ. Aydın and E. Kızılay, "Development of a new Light-Weight Convolutional Neural Network for acoustic-based amateur drone detection," *Applied Acoustics*, vol. 193, p. 108773, May 2022, doi: 10.1016/J.APACOUST.2022.108773.

[22] Z. Liu *et al.*, "Deep Learning-Based Acoustic Recognition of UAVs in Complex Environments," *Drones*, vol. 9, no. 6, p. 389, May 2025, doi: 10.3390/DRONES9060389.

[23] J. Najafi, S. Mirzakuchaki, and S. Shamaghdari, "Autonomous Drone Detection and Classification Using Computer Vision and Prony Algorithm-Based Frequency Feature Extraction," *Journal of Intelligent & Robotic Systems*, vol. 111, no. 1, p. 8, Jan. 2025, doi: 10.1007/S10846-024-02216-X.

[24] O. O. Medaiyese, M. Ezuma, A. P. Lauf, and I. Guvenc, "Wavelet transform analytics for RF-based UAV detection and identification system using machine learning," *Pervasive and Mobile Computing*, vol. 82, p. 101569, Jun. 2022, doi: 10.1016/J.PMCJ.2022.101569.

[25] D. T. Phan, "Comparison Performance of Spectrogram and Scalogram as Input of Acoustic Recognition Task," *Lecture Notes in Networks and Systems*, vol. 1283, pp. 660–673, Nov. 2025, doi: 10.1007/978-3-031-84457-7_41.
[26] M. H. Rahman, M. A. S. Sejan, M. A. Aziz, R. Tabassum, J. I. Baik, and H. K. Song, "A Comprehensive Survey of Unmanned Aerial Vehicles Detection and Classification Using Machine Learning Approach: Challenges, Solutions, and Future Directions," *Remote Sensing*, vol. 16, no. 5, p. 879, Mar. 2024, doi: 10.3390/RS16050879.
[27] S. Al-Emadi, A. Al-Ali, A. Mohammad, and A. Al-Ali, "Audio Based Drone Detection and Identification using Deep Learning," *in Proc. 2019 15th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Tangier, Morocco, Jun. 2019, pp. 459–464, doi: 10.1109/IWCMC.2019.8766732. [Online]. Available: https://github.com/saraalemadi/DroneAudioDataset
[28] K. J. Piczak, "ESC: Dataset for Environmental Sound Classification," *Proceedings of the 23rd ACM international conference on Multimedia. ACM*, pp. 1015–1018, Oct. 13, 2015. doi: 10.1145/2733373.2806390.
[29] P. Warden, "Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition," *arXiv:1804.03209*, Apr. 2018, doi: 10.48550/arXiv.1804.03209.
[30] I. Lupea and M. Lupea, "Continuous Wavelet Transform and CNN for Fault Detection in a Helical Gearbox," *Applied Sciences*, vol. 15, no. 2, p. 950, Jan. 2025, doi: 10.3390/app15020950.
[31] J. Zhou, B. Hao, Y. Li, and X. Yang, "Underwater Small Target Detection Method Based on the Short-Time Fourier Transform and the Improved Permutation Entropy," *Acoustics*, vol. 6, no. 4, pp. 870–884, Oct. 2024, doi: 10.3390/acoustics6040048.
[32] A. Jablonski and K. Dziedziech, "Intelligent spectrogram – A tool for analysis of complex non-stationary signals," *Mechanical Systems and Signal Processing*, vol. 167, p. 108554, Mar. 2022, doi: 10.1016/j.ymssp.2021.108554.
[33] D. Romahadi, A. G. Feleke, and R. P. Youlia, "Evaluation of Laplacian Spatial Filter Implementation in Detecting Driver Vigilance Using Linear Classifier," *International Journal of Technology*, vol. 15, no. 6, p. 1712, Dec. 2024, doi: 10.14716/ijtech.v15i6.7166.
[34] C. Szegedy *et al*., "Going deeper with convolutions," *in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, 2015, pp. 1–9, doi: 10.1109/CVPR.2015.7298594.
[35] M. Hasnain, M. F. Pasha, I. Ghani, M. Imran, M. Y. Alzahrani, and R. Budiarto, "Evaluating Trust Prediction and Confusion Matrix Measures for Web Services Ranking," *IEEE Access*, vol. 8, pp. 90847–90861, 2020, doi: 10.1109/access.2020.2994222.

BIOGRAPHIES OF AUTHORS



Fajar Anggara holds a Master of Engineering degree in Mechanical Engineering. His academic research is focused on fluid mechanics and heat transfer. The results of his research have been published in Scopus and other prominent publications through many international collaborators. He can be contacted at email: fajar.anggara@mercubuana.ac.id.



Dedik Romahadi is a lecturer in the Department of Mechanical Engineering at Universitas Mercu Buana, Jakarta, Indonesia, where he has served as a permanent faculty member since August 2018. He earned a Master of Science (M.Sc.) in Mechanical Engineering. His research focuses on signal processing, machine learning, and brain–computer interface systems, with particular interest in applying these methods to engineering diagnostics, intelligent monitoring systems, and electroencephalography-based human–machine interaction. He has published his work in a range of reputable journals and continues to develop practical, data-driven methods for real-world mechanical and industrial challenges. He can be contacted at email: dedik.romahadi@mercubuana.ac.id.



Aberham Genetu Feleke received a Ph.D. in Mechanical Engineering from Beijing Institute of Technology, Beijing, China, in 2019. He is currently working as a lecturer at Beijing Institute of Technology. He has 3 years of postdoctoral research experience and 7 years of teaching experience in higher education. He co-authored a book called "Advanced Control Engineering" and published more than twenty research papers in reputable journals and conferences. His research interests include signal processing, human intention prediction, and human-machine collaboration. He can be contacted at email: abrucag@gmail.com.