3-2023

Cellphone-Acoustics Based sUAS Detection and Tracking

Ryan D. Clendening

CELLPHONE-ACOUSTICS BASED SUAS DETECTION AND TRACKING

THESIS

Ryan D. Clendening, B.S.C.E., Second Lieutenant, USAF

AFIT-ENG-MS-23-M-017

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENG-MS-23-M-017

CELLPHONE-ACOUSTICS BASED SUAS DETECTION AND TRACKING

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Computer Engineering

Ryan D. Clendening, B.S.C.E., B.S.C.E.

Second Lieutenant, USAF

March 23, 2023

AFIT-ENG-MS-23-M-017

CELLPHONE-ACOUSTICS BASED SUAS DETECTION AND TRACKING

THESIS

Ryan D. Clendening, B.S.C.E., B.S.C.E.
Second Lieutenant, USAF

Committee Membership:

Richard Dill, Ph.D
Chair

Brett J. Borghetti, Ph.D
Member

Douglas D. Hodson, Ph.D
Member

Abstract

Small Unmanned Aircraft Systems (sUAS) are an accessible technology that has become an increasingly large threat to US critical systems. They are small, lightweight, and challenging to detect, which has allowed sUAS to provide reconnaissance, drop explosives, and even crash into sensitive targets. This threatening technology demands using fault-tolerant, low-cost, replaceable, and accurate sensing resources, which counter the ubiquitous nature of sUAS [1]. Therefore, the methods developed in this thesis detect and track sUAS using accessible sensing resources, such as cellphones. This research effort stems from an Air Force Research Laboratory (AFRL) data collection in which sUAS fly over a constellation of cellphones that record timestamped acoustics data. In the first effort, we develop an acoustics sensor network-based sUAS detection methodology. It uses an Ensemble Voting Pipeline (EVP) that fuses time-synchronized, low-fidelity acoustics data from a constellation of 28 cellphones scattered throughout an airfield to make real-time drone detection decisions. This effort achieves a detection F1-Score of 0.846 in simulated test scenarios. The pipeline also outperforms the sUAS detection performance of each individual cellphone within the sensor network, which has an average detection F1-score of 0.582.

In the latter effort, a deep learning model is trained using acoustics data from the collection to predict sUAS range from a cellphone. A 2-Dimensional Convolutional Neural Network (2DCNN) predicts sUAS with a macro-F1 score of 0.7492 across four distinct range classes. Combined, these two efforts demonstrate the merits of using accessible sensing resources to achieve high-fidelity sUAS detection and tracking results.

To my wife. Without her support and encouragement, I would not be where I am today.

Acknowledgments

I want to thank my advisor, Maj Richard Dill, for his continued support throughout the entire thesis process and for enabling me to become a better writer, thinker, and engineer. I would also like to thank my committee members, Dr. Douglas Hodson and Dr. Brett Borghetti for their aid and expertise throughout the entire thesis process. Lastly, I would like to thank Dr. Peter Zulch, Dr. Darren Haddad, and Dr. Brett Smolenski for their Escape II dataset, technical support, and expertise in sUAS acoustics. This thesis would not have been accomplished without the help and mentorship of these gentlemen.

Table of Contents

CELLPHONE-ACOUSTICS BASED SUAS DETECTION AND TRACKING

I. Introduction

This chapter introduces the research domain, briefly summarizes the research objectives, and outlines the thesis. Section 1.1 discusses the airspace threat Small Unmanned Aircraft Systems (sUAS) pose, and Section 1.2 highlights counter-measures for sUAS. In Section 1.3, the primary research efforts, hypotheses, and goals are presented. Lastly, Section 1.4 provides an overview of the following chapters.

1.1 The Threat of sUAS

Within the past decade, the prevalence and accessibility of sUAS have changed aerospace conflict forever. Before the influx of sUAS in the consumer market, unpiloted aircraft were costly and primarily used in large-scale military operations [2]. However, due to recent rapid increases in innovation and manufacturing cost reduction, the same sUAS model can now be found anywhere from a child's birthday party to the front lines in the Russian-Ukrainian War. The exponentially increasing number of sUAS presents a variety of security concerns that demand military solutions.

The security concerns of sUAS pose threats to infrastructure, airspace, and military personnel. In 2015, an sUAS accidentally landed on the White House lawn, forcing the surrounding area to be put on lockdown [3]. In 2017, an sUAS collided with a commercial twin-propeller airplane while on the final descent, endangering the lives of everyone within the aircraft [4]. In 2016, ISIS attached an explosive to a sUAS during the battle of Mosul, demonstrating the first exploited use of hobbyist sUAS in military combat [5]. However, in recent events, sUAS are no longer a tool

used by non-state combatants or a confused amateur pilot but a powerful asset in a large-scale conflict. During the Ukraine conflict, both sides have used commercially accessible sUAS to provide reconnaissance, adjust artillery fire, and drop explosives on unsuspecting troops on the ground (see Figure 1). Ukrainian forces have received donations to try and create an sUAS army, while Russia has similarly put the assets to use for military operations [6]. These threats highlight the US military's challenges in protecting critical infrastructure.

The threats that sUAS present have prompted leading Department of Defense (DoD) Strategic Leaders to publish "Counter-Small Unmanned Aircraft Systems Strategy." The Doctrine highlights the changing domain of sUAS and the need for the Joint Force to prepare to develop solutions that span the entire "Doctrine, Organization, Training, Materiel, Leadership and Education, Personnel, Facilities—Policy (DOTMLPF-P) spectrum [1]." Additionally, the Doctrine emphasizes the need for our Forces to "detect, identify, deter, and, if necessary, defeat threat sUAS [1]." However, to do so, sUAS defense methods must be established to accurately detect, locate,

Figure 1: sUAS carrying an explosive payload [7]

classify, and, if necessary, overpower these systems.

The main goal of sUAS defense is to deny hostile sUAS access to contested airspace. Nevertheless, active and passive defense methods must be employed to achieve airspace protection. Military forces employ a variety of weaponry to defend against sUAS threats. Electronic warfare weapons disrupt, jam, and take over sUAS communications. In the Russian-Ukrainian conflict, Russian forces have jammed the sUAS communication signal, forcing the vehicle to return home or hijacked the sUAS link connected to the control systems mid-flight [6]. Other counter-sUAS weapons include small rockets [8], Electronic weaponry (Figure 2), guns, and lasers [9]. However, for these technologies to effectively deny, disrupt, or destroy sUAS threats, sUAS must be accurately detected and tracked.

Thus, the cornerstone of sUAS defense is accurate sUAS detection, classification, and tracking. These three goals are the principal components by which sUAS can be defeated. The US Air Force Science and Technology Strategy demands the need for "lower-cost sensors integrated on distributed platforms that provide resilience through

Figure 2: Electronic-sUAS Weapon being employed by a US Marine during a training exercise in 2018 [10]

numbers and redundancy and complement more exquisite sensors on standoff platforms [11]." Standard methods to achieve sUAS awareness rely on different sensing methods, including radar, optics, Radio Frequency (RF), acoustics, or a combination of these techniques [9]. However, each sensing modality has inherent costs that present unique challenges and fail to meet the objectives of the U.S. Air Force Science and Technology Strategy. Radar-based methods can detect and track sUAS but are expensive and may not reliably detect the shapes of various sUAS. RF-based solutions can detect and classify sUAS communication when in perceivable range but cannot track sUAS location. Optics-based methods can detect, track, and classify sUAS but require line-of-sight to perceive an sUAS, thus being constrained to near-optimal environmental conditions. Lastly, acoustics-based solutions can detect, classify, and track sUAS; however, acoustic sensors may be hindered by low signal-to-noise ratios and a lack of sensor platform mobility. Each of these sensing modalities is used for counter-sUAS strategies, but these sensing platforms tend to be expensive, have a centralized point of failure, and are difficult to mobilize. However, the future of warfare demands that the DoD develops resilient and fault-tolerant sensor networks to complement these expensive sUAS sensing platforms. These systems can allow the Joint Force to accelerate towards a warfare strategy that capitalizes on autonomous, edge computing technology.

1.2 sUAS Defense Solution

The research presented in this thesis supports the US Air Force Science and Technology Strategy goals by developing data-driven methods to detect and track sUAS using low-fidelity acoustics data from Commercial-off-the-shelf (COTS) cellphones. The research is broken into two distinct efforts: a Machine learning (ML)-based Ensemble Voting Pipeline (EVP) that provides resilient, fault-tolerant sUAS detection

from a sensor constellation of 28 cellphones and a deep-learning model that is capable of estimating sUAS range from a single cellphone.

1.2.1 Research Data

The data used throughout this research effort comes from the Escape II Data Collection, conducted by Air Force Research Laboratory (AFRL), where multi-sensor data is collected on various scenarios, including sUAS, vehicle and human movement scenarios. Many radar-based, acoustics-based, electro-optics-based, and vision-based sensors record the scenarios; however, this research effort specifically uses the acoustics data from the cellphones that record acoustics data during the sUAS scenario. The cellphones record the acoustics data using RedVox, a multi-modal data collection app [12]. All cellphones record acoustics data with a sample rate of 8KHz and are time-stamped to the microsecond. Additional details regarding the scenario can be seen in Chapter II and Chapter III.

1.2.2 Effort 1: sUAS Detection

The first research effort, "Sensor Network-Based sUAS Detection using Low-fidelity Audio from Cellphones," demonstrates the merits of ensemble-based predictions from inexpensive edge computing devices to achieve accurate sUAS detection over large airspace. First, a ML model is developed using data from the data collection to distinguish sUAS presence from background noise. The data is formatted into the Mel Frequency Cepstral Coefficients (MFCC) space and put into 256ms frames, which returns 40 coefficients. The model is trained using a subset of data from the data collection, which is partitioned to an *sUAS* and *Noise* class, determined by the GPS range from an sUAS to a cellphone. Any acoustics frame in which the range is within 80m of a cellphone is considered *sUAS*, whereas any frame with a range

outside of 80m is considered *Noise*. After model training, the model is copied to all devices within the cellphone constellation. Then a EVP is implemented that aggregates predictions from the constellation of sensors to determine sUAS presence. The EVP consists of a Majority Voting Scheme, a Weighting Function, and erroneous Prediction Finite State Machine (FSM). The EVP is evaluated using nine real-world flights from the data collection. The distributed sensor network that uses the EVP outperforms a single cellphone's F1 scores by 0.264 (relative to the cellphone) and resiliently detects sUAS despite sensor errors and dropout with an F1-score of 0.846. This research effort establishes a real-time sUAS detection methodology that produces resilient and accurate results.

1.2.3 Effort 2: sUAS Range Estimation

The second research effort, "Cellphone-Based sUAS Range Estimation: A Deep-Learning Approach," uses low-fidelity acoustics data to estimate sUAS range from a single cellphone with a deep learning model. This effort presents a sUAS tracking method that is scalable across a large constellation of devices and provides an accurate range estimation over a variety of sUAS types. Four sUAS datasets are used in this research effort, IF, Matrice, Phantom, and Combined. The IF, Matrice, and Phantom datasets contain data from single sUAS models, whereas the Combined dataset is a superset of the three individual datasets. Each dataset has 500ms audio samples with an associated sUAS range classification, which is separated into four distinct range classes ($y \leq 40m$, $40m < y \leq 60m$, $60m < y \leq 80m$, or $y > 80m$). Twenty percent of each of the individual sUAS datasets is sequestered for performance evaluation. Three different deep learning model architectures, 2-Dimensional Convolutional Neural Network (2DCNN), 1-Dimensional Convolutional Neural Network (1DCNN), and 2-Dimensional Convolutional Recurrent Neural Network (2DCRNN),

are compared using the Combined dataset, and the best-performing model is further evaluated using the sequestered testsets. The 2DCNN achieves an average macro-F1 score of 0.7492 across the four range classes and on three different sUAS model testsets. It demonstrates that sUAS model-agnostic range estimation features are extracted. This research effort establishes a low-cost sUAS tracking method that does not require expensive, fixed acoustic sensing methods.

The current publication status of the two efforts is seen in Table 1. "Sensor Network-Based sUAS Detection using Low-fidelity Audio from Cellphones" is awaiting PA approval and "Cellphone-Based sUAS Range Estimation: A Deep-Learning Approach" is currently in the review phase of conference publication. However, both papers are completed works and are presented within the thesis as such.

Table 1: Publication Results

Title	Type	Venue	Status
Sensor Network-Based sUAS Detection Using Low-fidelity Audio from Cellphones	Journal	Proceedings of the ACM on Interactive, Mobile, Wearable, and Ubiquitous Technologies	Pending PA Approval
Cellphone-Based sUAS Range Estimation: A Deep-Learning Approach	Conference	World Congress In Computer Science, Computer Engineering, and Applied Computing	Pending Acceptance

1.3 Research Objectives

This research aims to demonstrate a low-cost method for providing sUAS awareness that is mobile, easily accessible, and resilient to device errors. The two research efforts combine to demonstrate that cellphone acoustics can detect and track sUAS, which is an essential part of sUAS defense. In alignment with Air Force Strategic Goals for the coming decade, this research ensures that the DoD has low-cost sensors that are scalable, fault-tolerant, and complement the results from high-fidelity sUAS awareness sensors.

1.4 Document Overview

This document is organized as follows. Chapter II and Chapter III detail the two research efforts and present the results from the two studies. Finally, Chapter IV provides conclusions.

II. Paper: Sensor Network-Based sUAS Detection using Low-fidelity Audio from Cellphones

The following paper, "Sensor Network-Based sUAS Detection using Low-fidelity Audio from Cellphones," is waiting on PA approval to be submitted to the ACM Journal on Interactive, Mobile, Wearable, and Ubiquitous Technologies.

Sensor Network-Based sUAS Detection using Low-fidelity Audio from Cellphones

RYAN CLENDENING, RICHARD DILL, BRETT BORGHETTI, and DOUGLAS HODSON, Air Force Institute of Technology, USA

Due to their low-cost, accessibility, and hard-to-detect nature, small Uncrewed Aircraft Systems (sUAS) are at the forefront of airspace security and military operations. They also offer an avenue by which malicious actors can undermine the safety and security of critical systems, such as airports, sports stadiums, power plants, and other restricted areas. In this paper, we create a novel cellphone acoustics-based ensemble voting pipeline (EVP) that fuses time-sync'd, independent machine learning (ML) model predictions from a scattered network of cellphones' audio sensors to detect sUAS presence. This proof-of-concept EVP consists of an ML model copied onto all cellphones, a weighting function, and a Prediction Finite State Machine (PFSM). The EVP provides resilient sUAS detection accuracy using real-world data in nine simulated test scenarios, achieving an sUAS detection F1-score (a measure of both precision and recall) of 0.846. It outperforms the sUAS detection F1-Score of a single cellphone within the sensor network by 45.6% and provides resilient sUAS detection despite cellphone mispredictions and dropout.

CCS Concepts: • **Computer systems organization** → *Neural networks*; **Sensor networks**; Fault-tolerant network topologies.

Additional Key Words and Phrases: sUAS Detection, Information Fusion, Deep Learning, Acoustics Processing

ACM Reference Format:
Ryan Clendening, Richard Dill, Brett Borghetti, and Douglas Hodson. 2023. Sensor Network-Based sUAS Detection using Low-fidelity Audio from Cellphones. 1, 1 (February 2023), 23 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Small Uncrewed Aircraft Systems (sUAS) pose security risks and have changed how malicious actors can achieve desired goals. Following the 2022 Russian invasion of Ukraine, both sides used sUAS for tactical reconnaissance and offensive missions. For example, the Ukrainian army used commercially available sUAS to complement their more advanced weapon systems, and Russian forces used sUAS to "strike and provide reconnaissance" throughout the conflict [25]. In response to the militarized application of commercial sUAS, state governments and private companies have developed counter-sUAS platforms; however, these are often expensive, immobile, and present a single-point of failure.

This research effort aims to address these weaknesses by accomplishing two objectives: first, evaluating the merits of an ensemble-based approach to sUAS detection from sparsely scattered cellphone microphones, and second, demonstrating how using edge computation devices can improve sUAS detection resiliency and accuracy. We develop an ensemble voting pipeline (EVP) that makes sUAS detection predictions from 20 sparsely scattered cellphones' audio data. All cellphones collect time-synchronized, low-fidelity acoustics data from their microphones to make independent sUAS presence decisions using a machine learning (ML) model. The EVP

Authors' address: Ryan Clendening, ryan.clendening@afit.edu; Richard Dill, richard.dill@afit.edu; Brett Borghetti, brett.borghetti@afit.edu; Douglas Hodson, douglas.hodson@afit.edu, Air Force Institute of Technology, 2950 Hobson Way, WPAFB, Ohio, USA, 45433.

XXXX-XXXX/2023/2-ART $15.00
https://doi.org/XXXXXXX.XXXXXXX

receives, assesses, and weighs each independent decision, to ultimately yield a final group prediction. While cellphone-to-EVP communications data was previously collected and then replayed offline, our novel method - which combines various cellphones' ML predictions to detect sUAS - can be used to implement a real-time wireless acoustic sensor network (WASN) for applications beyond acoustics-based sUAS detection. Our results show that an acoustic sensor network that uses the EVP for sUAS detection outperforms a single sensor's detection F1 score (a measure of both precision and recall) by over 45% and provides resilient sUAS predictions despite two sensors producing entirely erroneous predictions.

This activity is conducted in two stages - model selection and ensemble performance evaluation. In the model selection stage, a typical machine learning workflow is followed. Data is first collected and prepared for ingestion by several models for acoustics-based sUAS detection. The dataset contains acoustic recordings of 22 sUAS flights captured on 20 cellphone microphones. These recordings are partitioned into audio frames of length 256ms. Each segment of acoustics data is processed into its 40 Mel-Frequency Cepstral Coefficients (MFCCs). Additionally, GPS truth data from the cellphones and sUAS is captured for labeling the segment. Each frame is labeled with either *sUAS* or *Noise* classes to indicate whether the sUAS is within 80m of the cellphone. This data is used to train several models: linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), a radial basis function support vector machine (RBF-SVM), and a dense neural network (ANN) model. Once these models have been trained, their performance is evaluated on an unseen portion of the data to obtain a performance ranking.

After the first stage, the EVP performance is determined - a process that consists of several steps. The best-performing architecture is selected for use within the EVP, which has the following process. Each cellphone receives a 256ms audio frame, converts the audio to the MFCC feature space, and inputs the MFCCs into a model which makes a prediction (*sUAS* or *Noise*). The EVP gathers the independent cellphone predictions and applies a weight to each representing the device's average log energy relative to the sum of all devices' log energy. Next, the EVP aggregates the weighted votes to determine sUAS presence. This activity is repeated as the ensemble receives and processes a temporally-ordered sequence of audio frames. A memory-based prediction finite-state machine (PFSM) reduces the probability of transition from one label to another, improving the stability of sequential predictions by lessening the system's sensitivity to short-duration prediction errors. The ensemble's output is a time-series sequence of predictions of *sUAS* or *Noise* for each frame.

The outline of the paper is as follows. Section 2 details sUAS acoustics theory, multi-sensor fusion techniques, and the MFCC transform. Section 3 discusses sUAS sensing modalities and acoustics-based detection methods. Section 4 details the data collection, ML model selection experiment, pipeline design, and EVP evaluation methods. Lastly, Section 5 reports the results from the ML model selection experiment and evaluates the detection performance and resiliency for the EVP.

2 BACKGROUND

This section explains concepts necessary to understand the machine learning models and EVP proposed in Section 4. The following topics are covered: sUAS acoustics, machine learning, information fusion theory, and the Mel-Frequency domain.

2.1 sUAS Acoustics

sUAS emit sound from the rotating motors and propellers that produce lift and velocity. Sound is generated, resulting in a fundamental frequency (or frequencies) between 0-2kHz range [21]. Additionally, the harmonics produced from the propeller blade rotation help distinguish sUAS sounds from other noises. As shown in Equation 1, the fundamental frequency of an sUAS depends on the propeller rotations per minute, and the number of blades on the sUAS propeller [21]. sUAS that have motors rotating at different speeds (e.g., auto-stabilizing) have multiple fundamental frequencies due to the variations in propeller RPMs.

$$f_{fund} = RPM \cdot \frac{BladeCount}{60} \tag{1}$$

The physical vibration of sUAS produces additional acoustic noise, which tends to be at high frequencies (3KHz-4KHz) [13].

Figure 1 shows a spectrogram that illustrates the frequency of an sUAS. Harmonics are periodic from 400Hz to nearly 1500Hz. Although other noises may appear similar, ML can be trained to detect sUAS from environmental noises accurately.

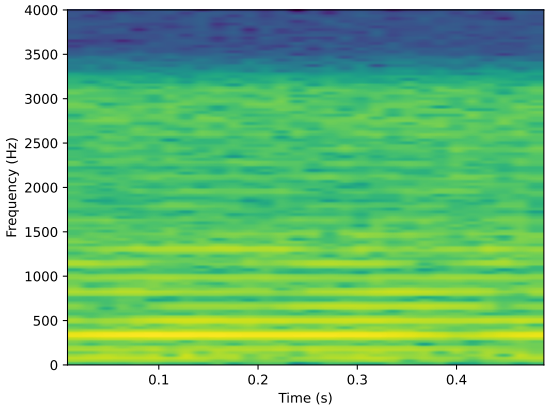

Fig. 1. Spectrogram of IF1200 sUAS

2.2 Machine Learning Algorithms

Four ML algorithms (ANNs, SVMs, LDA, and QDA) are evaluated to determine which model is suitable for the EVP; thus, a brief background is presented on each of the four algorithms.

An artificial neural network (ANN) is a supervised deep learning algorithm used to detect, predict, or classify a given phenomenon based on a dataset [1]. Neural networks, inspired by biological neurons in the brain, use multiple layers of perceptrons to learn a phenomenon. Each network layer consists of several perceptrons with an associated weight matrix containing the connection weights from the previous perceptrons in the layer. Additionally, a bias vector and activation function ensure non-linearity between network layers, an attribute that makes ANNs promising for learning complex decision boundaries [6].

SVMs are a family of ML algorithms in which a separating hyperplane maximizes the margins between points in different classes (for example - sUAS produced sounds versus other noises). SVMs typically work well for applications with input data with large feature spaces. In pattern recognition style problems, SVMs classify complex phenomena well due to their use of Kernel functions. These functions quantify the similarity between observations, mapping the original input space to a feature space where the data is linearly separable [10]. They are used to enlarge the features from their original domain into a higher dimensionality domain, where the SVM's separating hyperplane can better linearly separate the classes in the data.

Radial Basis Function (RBF) kernels transform the data into an infinite-dimensional space without explicitly working in the transformed feature space. This provides a computationally efficient kernel that can find the class decision boundary on non-linear decision boundaries [10, 20].

LDA and QDA are ML models that fit a classification discrimination boundary based on each class's data. However, each makes a different assumption about the nature of data feature distribution. LDA forces each class to represent the features using the same co-variance among the distribution of feature values in the data. QDA allows each class to have a different co-variance among the features. These techniques work well when the relationship between the data features and class is simple, and these models are easier to interpret. Including these model types in the model selection process helps researchers understand the nature of the data-to-class relationship.

In our research, we implement ANN, RBF-SVM, LDA, and QDA and compare each performance to determine which to use in the EVP. Section 4.4 further details the composition of the ML models, the model selection process, and the results.

2.3 Multi-Sensor Fusion Technique: Ensemble Voting

Sensor data fusion is defined as "the combination of data from multiple sensors (either of the same or different types)... to achieve more specific inferences than could be achieved by using a single, independent sensor" [14]. Although a variety of sensor fusion techniques exist to combine data from a constellation of devices, the research in this effort uses *information fusion* (aka *late fusion*). In related wireless acoustic sensor fusion works, researchers have fused data from acoustic sensors to detect and localize gunshots [5, 19]. The information fusion method that is employed in this effort is ensemble voting.

Ensemble voting aggregates the predictions of multiple sources (e.g., independent ML models) to improve the decision quality. Ensemble voting can be represented as a series of Bernoulli Trials, in which each trial represents a predictor and has two possible outcomes: success or failure. Thus Equation 2 demonstrates the probability of at least x correct predictions with n total predictors and a probability of correct prediction, p, and the probability of an incorrect prediction, $1 - p$.

$$P(r \geq x) = \sum_{r=x}^{n} nCr \cdot p^{r} \cdot (1-p)^{n-r} \tag{2}$$

Therefore, assuming a network of better-than-chance models where all models predict the presence of an sUAS based on independent decisions, the probability of success approaches 1 asymptotically as the number of predictors increases [7]. In this effort, the EVP leverages ensemble voting of sUAS detection predictions from the constellation of cellphone audio sensors. The EVP performs better than a single sensor while remaining scalable and resilient, further demonstrated in Sections 4 and 5.

2.4 Feature Extraction

Designed in the early 2000s to represent how humans perceive sound, Mel-Frequency Cepstral Coefficients (MFCCs) mathematically capture the acoustic variations between low and high frequencies in a low-dimensional way [16]. Related research efforts have depended on MFCCs to distinguish the larger UAS, which emit higher frequencies, from smaller UAS devices [13, 18, 20, 23, 26].

The Mel-scale spaces frequency bins linearly under 1kHz and logarithmically above 1kHz. Using the Mel-scale has shown promise in machine learning applications where lower frequencies contain more information than high-frequency components [13]. The approximation of a Mel from a frequency in Hz (f) to f_{mel} is seen in equation 3.

$$f_{mel} = 2595 \cdot log_{10}(1 + \frac{f}{700}) \tag{3}$$

The MFCCs for a time series audio frame can be calculated through the following transformation steps. First, a windowing function is applied to the data frame to reduce the effects of spectral leakage (e.g., Hanning window). Next, the Discrete Fourier Transform (DFT) is applied to the frame to transform it into the magnitude spectrum. Equation 4 calculates the DFT, which yields the magnitude spectrum, X(k), where N is the frequency points used in the DFT, and k is the number of bins between 0 and N-1 [17].

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{\left(\frac{-j2\pi nk}{N}\right)}; \quad 0 \leq k \leq N-1 \tag{4}$$

After the conversion to the magnitude spectrum, $X(k)$ is multiplied by a series of overlapping triangular filter banks, where $H_m(k)$ represents the weight given to the k^{th} energy spectrum bin, corresponding to the m^{th} output band. Equation 5 presents the calculation of the Mel Spectrum from the Magnitude Spectrum. Again, N is the frequency points used in the DFT transform, and M is the total number of triangular Mel weighting filters [17].

$$s(m) = \sum_{k=0}^{N-1} |X(k)|^2 \cdot H_m(k); \quad 0 \leq m \leq M-1 \tag{5}$$

The Discrete Cosine Transform (DCT) is then applied to the transformed Mel Frequency Coefficients, $s(m)$, resulting in Mel Frequency Cepstral Coefficients representing the overall power spectral envelope shape of the signal. Equation 6 presents the calculation to convert from Mel Frequency Coefficients to MFCCs, in which C is the number of MFCCs, M is the number of triangular filters, and $c(n)$ are the cepstral coefficients [17].

$$c(n) = \sum_{m=0}^{M-1} log_{10}(s(m)) \cdot cos\left(\frac{\pi \cdot n(m-0.5)}{M}\right); \quad n = 0, 1, 2, ..., C-1 \tag{6}$$

Equation 6 results in a fixed number of Mel Frequency Cepstral Coefficients, representing the power spectral envelope of a given acoustic frame [17].

This section establishes the necessary foundations for sUAS acoustics, ML, ensemble learning, and the ML feature format. Next, we present related sUAS detection works.

3 RELATED WORKS

Acoustic-based approaches exploit the produced sUAS noise for sUAS detection, tracking, and localization. Due to the persistent propeller noise, acoustic-based methods can distinguish sUAS acoustic footprints and localize the sounds using statistical and ML methods. However, most acoustics-based approaches are susceptible to noise pollution, have range limitations, and usually require high-fidelity microphones to achieve positive results. In addition, many efforts report results from test scenarios that do not fully represent real-world conditions. In contrast, our research focuses on acoustic-based sUAS detection using sparsely scattered, low-fidelity microphones across a large amount of airspace to achieve resilient sUAS detection. Current acoustic-based sUAS detection methods are now presented to demonstrate the research gap that our effort fills.

3.1 Detection Methods

Jeon et al. compared the detection accuracy of Gaussian Mixture Models (GMM), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN) in noisy, urban environments [11]. The researchers formatted the acoustic data with the Mel-frequency spectrogram. They discovered that the RNN model achieved the best detection accuracy in noisy urban environments, with an F1-score of 0.8009 for known sounds.

Shi et al. investigated using hidden Markov models (HMM) for sUAS detection in noisy environments [23]. The researchers used MFCCs to extract feature vectors for the HMM. Experimentally, Shi et al. discovered that

even at a low Signal-to-Noise Ratio (SNR), their model detected sUAS more than 80% of the time. Shi et al.'s main contribution is the usefulness of HMM for sUAS detection. They also demonstrated the ability to detect sUAS in noisy environments (5dB SNR).

Sedunov et al. developed an sUAS detection and localization system using an array of microphone nodes [21]. Their primary research emphasized long-range sUAS detection and localization using high-fidelity, directional microphones. Instead of using ML for sUAS detection, Sedunov et al. exploited the fact that multi-rotor UASs spin each rotor at different rotations per minute (RPM). Without the need for a database, their algorithm detected sUAS based on if there is more than one set of harmonics. Their algorithm achieved up to 97% sUAS detection accuracy compared with other types of aircraft (i.e., planes and helicopters). This research achieved high-level detection without bearing the cost of ML and showed that high-fidelity acoustic sensors could accurately localize sUAS targets.

Seo et al. developed a CNN to detect COTS sUAS using normalized Short-time Fourier transform (STFT) to create 2-dimensional images from sUAS acoustic data [22]. The researchers recorded outdoor sUAS data from two hovering sUAS and split the data into 2ms segments with 50% overlap. They then injected white Gaussian noise into the data for testing. Seo et al. achieved a 98.97% detection rate with a 100-epoch CNN and favorable SNR conditions. They demonstrated that STFT could be used for a CNN to achieve high detection rates.

Emadi, et al. researched sUAS detection and classification with different deep learning techniques [1]. Due to the lack of audio samples, the researchers augmented training data by recording audio from a smartphone in an indoor setting and creating additional training data with a Generative Adversarial Network (GAN). Their best-performing model was a CNN that used 1s spectrograms and achieved a detection F1-score of 0.9590. Their effort demonstrated the advantages of using augmentation techniques for sUAS detection and sUAS classification.

Casabianca et al. developed a late data fusion ensemble voting scheme of CNNs and CRNNs for sUAS detection [4]. The researchers arbitrarily adjusted the different neural network hyperparameters to create an ensemble of 10 predictors. The researchers trained the models using a training dataset taken entirely from online sources in which they then artificially mixed background noise (e.g., airplane sounds) to simulate real-world data. After training the various models, they evaluated the ensemble on unseen real-world data from a single cellphone positioned 1 meter from a hovering sUAS. They achieved an ensemble detection rate of 91.044% on this unseen data.

Lastly, Kolamunna et al. developed a recurrent neural network (RNN) for both sUAS detection, and classification [13]. The researchers addressed the lack of sUAS data by augmenting sUAS and noise data from online resources. Their augmentation technique of using online sources inspired the YouTube Video augmentation technique in this research effort. The researchers use a cascaded deep learning approach, where a model is first trained to detect an sUAS; once seen, it is prepared to classify the make and model of the sUAS. In evaluation, their detection model achieved F1-scores between 0.96-0.98 for closed set (types of sUAS seen by the model) performance and an open set F1-score performance of 0.88. Their work demonstrated the merits of a cascaded approach to sUAS detection and a rigorous dataset engineering process (augmentation, peak normalization, Doppler shifting, MFCC concatenation).

These research efforts demonstrate various machine-learning-based sUAS detection methods, summarized in Table 1. However, all efforts rely on high-fidelity audio from online sources or near-optimal recording environments. In contrast, this research uses cellphones that sample at 8KHz from an omnidirectional microphone. The sUAS are recorded from a wide range of distances and a considerable variation in cellphone type. Additionally, the data collection takes place at an airfield where there is environmental noise (i.e., airplane propellers). Our effort also deploys the ML model to a constellation of different cellphones in various real-world sUAS flights. Therefore, this effort seeks to demonstrate that despite these non-optimal environmental conditions, sampling rates, and sensing devices, low-fidelity cellphones can be aggregated into an EVP to achieve high-fidelity acoustic-based sUAS detection results.

Table 1. Performance of Related Works

RW and Citation	Sample Rate	Preprocessing	Model Type	Capabilities	Deficiencies
Jeon et al. [11]	44.1KHz	MFCC	RNN	Resilient in noisy environments	Not generalized across different microphones
Shi et al. [23]	44.1KHz	MFCC	HMM	Resilient in noisy environments	Reliant on high-fidelity data
Sedunov et al. [21]	48KHz	-	Spectrogram	No ML model or database	Requires extremely high-fidelity equipment
Seo et al. [22]	44.1KHz	STFT	CNN	CNN for sUAS detection	Unable to distinguish between similar noises
Al-Emadi et al. [1]	16KHz	Spectrogram	CNN	GAN for sUAS augmentation	Not tested in real-world scenarios
Casabianca et al. [4]	44.1KHz	Mel-Spectrogram	CNN/CRNN	Ensemble Voting Method	Trained and evaluated on unrealistic scenarios
Kolamunna et al. [13]	44.1KHz	MFCC	RNN	Cascaded Approach to sUAS detection and classification	Only evaluated with synthetically mixed audio

4 METHODOLOGY

In this section, all research methods are presented. First, the data collection, data collection scenarios, and the training dataset are discussed. We then explain the two stages of the research effort: the ML model selection and then the EVP design. Lastly, the EVP performance and resiliency experiment methodology are detailed.

4.1 Data Collection

The acoustics dataset is sourced from the Escape II data collection, a multi-week effort conducted by Air Force Research Labs [27]. The data collection transpires at an active airfield; thus, environmental noise (e.g., airplanes, cars, crickets) is present throughout the data collection.

Two different sUAS aircraft are flown during the data collection, and both aircraft have notable features that make them distinguishable. The Da-Jiang Innovations (DJI) Matrice 600 is a hexacopter and weighs approximately 20 pounds. The other, the DJI Phantom 4, is a quadcopter that weighs three pounds. Therefore, the two sUAS have very different acoustic footprints.

Twenty-eight cellphones (Samsung Galaxy S Series and Apple iPhones) are labeled and positioned along the flight path in three distinct clusters. The cellphones record acoustic, infra-sound, accelerometer, and gyroscopic sensor data via the RedVox application[24]. The multi-sensor data is captured and timestamped at microsecond intervals; however, this research is limited to GPS and acoustic data [9]. Table 2 lists the software and hardware configurations used throughout the collection.

Table 2. Sensor Configuration

Sensor Configuration	
Device Count	13 Apple iPhone 15 Samsung Androids
Apple App Version	4.0.2.4
Android App Version	3.3.1.2
Filetype	Redvox API 1000
Sampling Rate	8KHz
Bitrate	192Kbps
Audio Format	PCM Floating Point

4.2 Data Collection Scenarios

The data collection is separated into two distinct scenarios: short flight (*SF*) and long flight (*LF*). The *SF* scenario consists of 16 passes, where an sUAS flies approximately 410m paths from one end of the runway to the other, seen in Figure 2. Each sUAS is at 30.48m and travels 10-20 knots along the flight path. Each pass takes approximately 1-minute to complete. Although the sUAS speed varies between passes, the variation is consistent within each

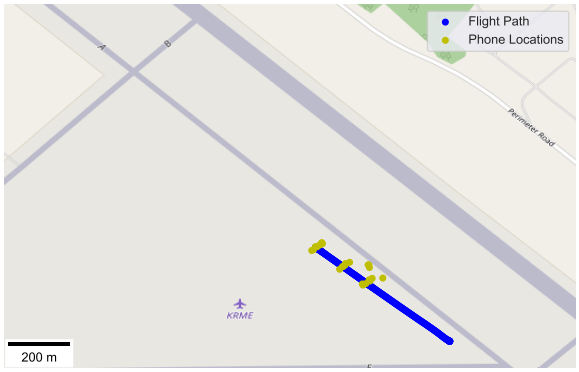

Fig. 2. Short Flight Scenarios

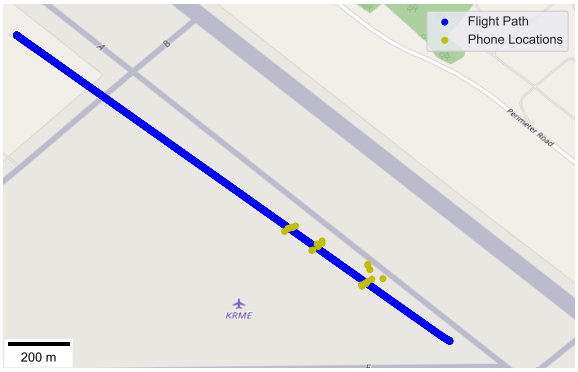

Fig. 3. Long Flight Scenarios

pass. The Matrice flies 6 passes while the Phantom flies the other 10 passes in the *SF* scenario. The *LF* scenario consists of 18 passes, in which an sUAS flies roughly 1.5km. The sUAS is at 30.48m above ground level (AGL) and remains at a consistent 15 knots throughout each pass. The Matrice and Phantom complete the *LF* scenario, with 8 passes from the Phantom and 10 from the Matrice. The sensor configuration is adjusted for this scenario, and the cellphone clusters are spread over 300m, seen in Figure 3. Six passes from the *LF* are used in the training dataset, while 12 are used for tuning and evaluating the EVP on the cellphone constellation. In figures 2 and 3, the red circles represent each cellphone's class separation between *sUAS* and *Noise* classes.

4.3 Training Dataset

The ML training dataset, composed of .wav files sampled at 8KHz, contains all *SF* passes and six *LF* passes. Additionally, the training dataset includes cellphone audio data from 24 of the 28 devices used in the collection. The remaining four phones never cord usable audio due to software issues during the collection. Since the objective is to generalize sound across different internal microphones (Android and iPhone), the dataset includes the acoustic recordings from all 24 usable cellphones.

In preparation for ML model training, the .wav files are first converted to time-series arrays, then divided into 256ms audio frames. The class boundaries are generated using the GPS range [1] from each cellphone to the sUAS for a given 256ms acoustics audio frame. All acoustics frames where the sUAS falls within 80m range of a cellphone are labeled *sUAS*, whereas every frame where the sUAS is outside of 80m is labeled *Noise*. Assuming that each sUAS flies at 15Kn, an sUAS traverses each cellphone's 80m detection area in approximately 10 seconds. Additionally, any frame where the sUAS is within $+/-2$m from the class separation boundary is removed (the intuition is that the sUAS flies on average 15Kn, which equates to 1.975m in 256ms). Finally, the dataset is partitioned into a 20% training-test-split, and the dataset is then z-transformed (zero-mean and unit variance) based on the distributions of the training data. Each frame is then converted to MFCC format (discussed in Section 2.4), returning a 40×1 coefficient vector.

Table 3 lists the training dataset composition. The dataset contains an equal amount of both classes, which is chosen to ensure that the model properly learns both the sUAS and Noise classes. A 20% testset is sequestered before ML model training to evaluate the performance of the various ML models. Section 4.4 provides additional information on the model training process.

[1]Although GPS positioning has limitations, in this effort, we assume that the GPS positions are accurate.

Table 3. Dataset Composition

Type	Time (s)	Frames
sUAS	3,497.2s	13,661
Noise	3,784.4s	14,783
Total	7281.6s	28,444

These design decisions ensure that the dataset provides enough variation to allow the ML model to properly generalize sUAS noise across different sUAS types, recording devices, detection distances, and SNRs. Although other efforts have used augmentation techniques like frequency shifting and injecting noise into training samples, we determine that these are not required because the different cellphone perspectives have inherent frequency shifts caused by the Doppler effect and the environmental noise (i.e., airplanes flying over the runway) provides sufficient SNRs.

4.4 Stage One: Model Selection

In the first stage of our effort, we make an initial ML model selection. The selected ML model should be computationally efficient and achieve sUAS detection results with low-dimensional input representations so that the model can be copied to cellphones and used in a real-world environment. Additionally, a null model is evaluated as a baseline and predicts the majority class of the training dataset (*Noise*). All models use the 40-dimensional MFCCs that are discussed in Section 2.4. Additionally, we evaluate the models using a sequestered 20% test-set, and the best-performing model based on the established criteria is selected for use in the EVP.

Both LDA and QDA are selected as extremely fast-performing models. The more complex but likely higher-performing models are a tuned RBF-SVM and an ANN. The tunable parameters of the SVM are C and gamma. A low value of C introduces more bias but decreases the variance of the model. In contrast, a high value of C does the opposite by introducing additional variance but decreasing bias. The gamma parameter dictates the influence of a single point; thus, a high gamma value suggests each point has a small amount of influence, which results in overfitting, whereas a small gamma indicates that each point has a lot of influence, thus causing the model to under-fit [15].

Additionally, three variations of the ANN compare a variety of network configurations, as seen in Tables 4, 5, and 6. ANNs are selected as the neural network architecture because the Mel coefficients are assumed to be independent and thus do not have temporal and spatial dependencies that may warrant more complex deep learning architectures (i.e., convolutional neural networks and recurrent neural networks). Version One is a wide but shallow ANN, which requires a large number of trainable parameters. Version Two is a shallow but deep ANN, which requires a small number of trainable parameters but has many layers. Lastly, ANN Version Three is a mixture of the two other networks and has deep and wide network characteristics. All three dense networks are exclusively made of scaled-exponential linear unit (Selu) activation functions, which is a type of activation function that produces linear mappings when an input value is above zero but a non-zero mapping when an input value is below zero. Equation 7 shows the Selu activation mapping with scaled-value λ.

$$selu(x) = \lambda \begin{cases} x & if x > 0 \\ \alpha e^{x} - \alpha & if x \leq 0 \end{cases} \tag{7}$$

This activation function reduces the vanishing gradient problem that Rectified Linear Units encounter and allows deep networks to converge to a solution. Additionally, all dense layers kernel weights are initialized to a LeCun Normal distribution. The LeCun Normal distribution has a zero mean and variance seen in Equation 8, where fan_{in} equals the number of input units to the dense layer. Combined, these hyper-parameters guarantee that

Table 4. ANN Version One

ANN Architecture	Size	Activation	Trainable Parameters
Input	40	Selu	
Dense	1024	Selu	41984
Alpha Dropout	0.5	-	-
Dense	512	Selu	524900
Alpha Dropout	0.4	-	-
Dense	256	Selu	131328
Alpha Dropout	0.3	-	-
Output	1	Sigmoid	257

Table 5. ANN Version Two

ANN Architecture	Size	Activation	Trainable Parameters
Input	40	-	-
Dense	80	Selu	3280
Alpha Dropout	0.3	-	-
Dense	80	Selu	6480
Alpha Dropout	0.2	-	-
Dense	80	Selu	6480
Dense	40	Selu	3240
Dense	20	Selu	820
Dense	20	Selu	420
Dense	20	Selu	420
Dense	10	Selu	210
Dense	10	Selu	110
Dense	5	Selu	55
Output	1	Sigmoid	6

the outputs of all dense layers in the network will self-normalize, alleviating the vanishing gradient problem [6, 12]. Therefore, this Selu activation-LeCun kernel initialization is in all layers of the three varieties of ANNs, as this commonly outperforms other activation functions. Additionally, alpha dropout regularizes each network by randomly setting input activations to the Selu activation low threshold, α'. However, alpha dropout preserves the mean and standard deviation of the inputs to a layer, ensuring self-normalization is not violated [12].

$$mean = 0, \sigma^2 = \frac{1}{fan_{in}} \tag{8}$$

The ML model selected for use in the EVP is the ANN Version Three. The rationale for selection and further results are presented in Section 5.1.

4.5 Stage Two: Ensemble Voting Pipeline Design

The second stage of the effort is the EVP. An overview of the ensemble voting pipeline (EVP) design flow is provided in the following paragraphs. The EVP begins by collecting audio information from all usable cellphones

Table 6. ANN Version Three

ANN Architecture	Size	Activation	Trainable Parameters
Input	40	-	-
Dense	160	Selu	6560
Alpha Dropout	0.3	-	-
Dense	80	Selu	12880
Alpha Dropout	0.2	-	-
Dense	40	Selu	3240
Dense	20	Selu	820
Dense	20	Selu	420
Dense	10	Selu	210
Output	1	Sigmoid	11

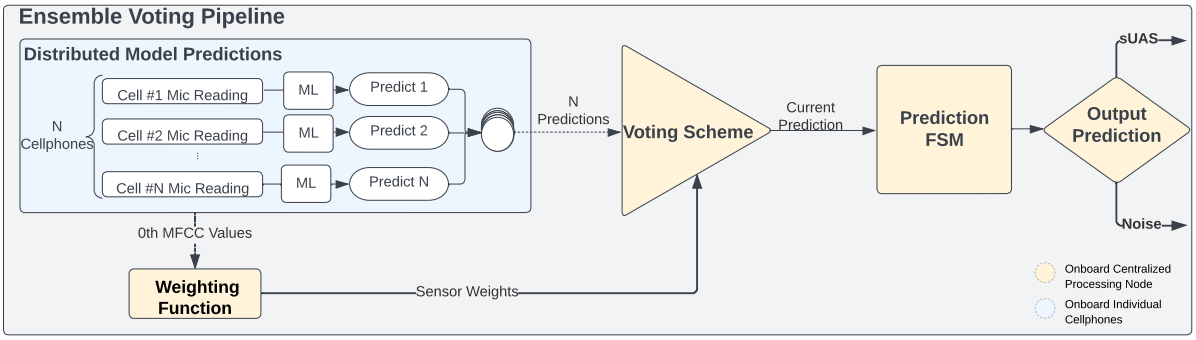

Fig. 4. sUAS Ensemble Voting Pipeline System Design

(N) in the constellation. Every 256ms, each cellphone records the local audio environment at its current location. Each device then converts the audio into the MFCC feature space, z-transforms the audio frame, and makes a prediction with the onboard ML model (*sUAS* or *Noise*). In real-world applications, this can be implemented via a WASN, a proven method for communicating audio-related information in various research efforts [2, 3]; however, the EVP simulates the data acquisition step. The EVP, serving as a centralized processing node, aggregates the results, weights each device's vote using the zeroth MFCC, calculates the weighted sum of the cellphones' predictions, and then makes a prediction using the voting scheme. The output passes through the erroneous prediction finite state machine (PFSM), which acts as a de-noising filter that ensures mispredictions caused by noisy or faulty sensor readings do not routinely affect EVP predictions. The finite state machine then produces the final prediction (*sUAS* or *Noise*). The following paragraphs discuss the design and EVP data flow process in more detail.

4.5.1 Distributed Model Predictions. The first sub-component of the pipeline gathers predictions from each cellphone every 256ms. Each phone records 256ms of audio, transforms the audio frame to a z-transformed MFCC format, and then makes an independent prediction using the ML model confidence threshold determined through tuning (see Section 5.2.1). These predictions are gathered and passed to the ensemble voting component. Additionally, the z-transformed zeroth MFCC of each cellphone is relayed to the Weighting Function.

4.5.2 Weighting Function. The *Weighting Function* sub-component's goal is to weight the predictions of cellphones that are likely correct. The mobile device constellation is sparse; consequently, only a minority subset of the phones can detect the sUAS during any given frame. The weighting function assumes cellphones that perceive a high energy level likely have a more valuable sUAS detection prediction. The zeroth MFCC coefficient represents the average log energy of the acoustic signal; thus, large zeroth coefficients suggest high sound levels.

All sensors' MFCC values pass through a Softmax function, which normalizes the sensor's weight log energy level relative to other devices in the constellation such that all values are positive, between zero and one. The sum of the normalized values is one. Equation 9 displays the softmax calculation, where x_i represents the i_{th} cellphone's Zeroth MFCC [8].

$$\sigma(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}} \tag{9}$$

4.5.3 Voting Scheme. After all the cellphones have made predictions and been weighted, the *Voting Scheme* sub-component aggregates the predictions to determine the system's overall prediction. Each cellphone prediction is multiplied by the cellphone's weight, as determined by the *Weighting Function*. Therefore, cellphone microphone readings with lower energy levels than others provide lower weight and vice versa. The current sUAS prediction is determined in Equation 10, in which $\sigma(x_i)$ represents the voting weight and $vote_i$ is either 1 (*sUAS*) or 0 (*Noise*). A value above the threshold signifies that an sUAS is present, and a value below the threshold signifies noise. The sUAS prediction threshold is an EVP hyper-parameter tuned to the cellphone constellation and environmental characteristics. More threshold tuning information is in sub-Section 4.7.

$$pred_{curr} = \sum_{i=1}^{n} (\sigma(x_i) * vote_i) > thresh \tag{10}$$

The voting scheme increases the overall probability of sUAS detection through the benefits of binomial probability. If cellphone predictions are treated as weak learners with detection accuracy above 50%, the aggregated sum of the predictions provides more accurate results than any individual cellphone. Therefore, the voting scheme is designed to ensure that the constellation of devices provides more accurate sUAS prediction results than any individual cellphone's predictions.

4.5.4 Prediction Finite State Machine. The final sub-component of the sUAS prediction pipeline is the *PFSM*, designed to reduce erroneous predictions. The *PFSM* adds temporal stability to the EVP's prediction and reduces the chances of erroneous predictions from the voting scheme in a sequence of predictions. The *PFSM* method is developed using the heuristic that an sUAS detected in one 256ms frame is likely to be the same one present in the subsequent frame, and vice versa for noise. Therefore, assuming the *PFSM* is at a steady state, the EVP must predict *sUAS* or *Noise* four times before the PFSM allows a transition to the alternate prediction.

The PFSM has eight states representing the system's output prediction. This number is determined during the EVP tuning in Sub-section 4.7. The *PFSM* input is the prediction of the voting scheme sub-component, and the output is the final EVP prediction. Figure 5 presents the *PFSM*, where each arrow represents the EVP's current $prediction$ that is passed from the voting scheme, and the states in Figure 5 represent the EVP's $OutputPrediction$. The *PFSM* is designed to reduce the prediction variance caused by spurious short-duration prediction errors, thus requiring repeated identical predictions to transition from one prediction state to the other.

4.6 Test Scenarios and Truth Data Determination

Nine real-world flights from the *LF* scenario evaluate the EVP. Additionally, three passes from the *LF* scenario tune the confidence threshold value for the ML model, the voting threshold for the EVP, and the number of PFSM

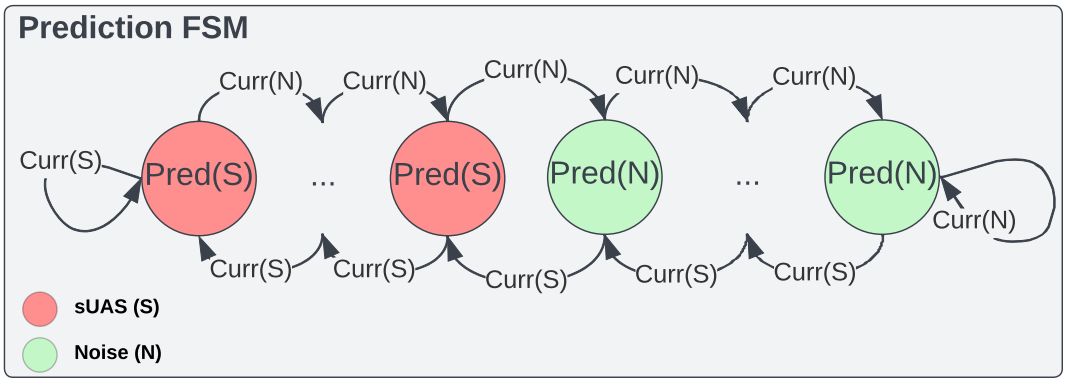

Fig. 5. sUAS Prediction FSM: Arrows demonstrate what most recent prediction is, whereas circles represent the actual EVP prediction.

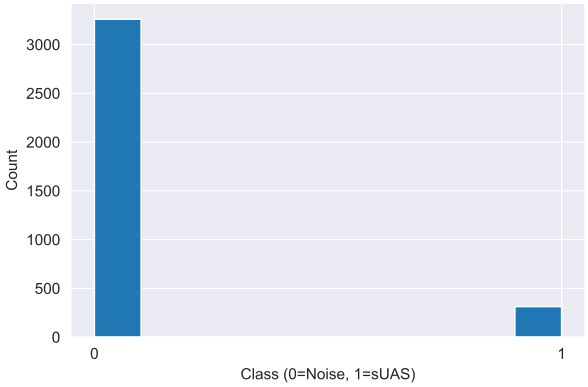

Fig. 6. Histogram of Indiv. Cellphone Truth Data Frames

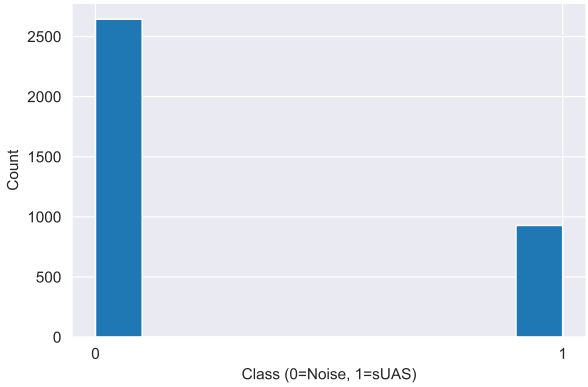

Fig. 7. Histogram of EVP Truth Data Frames

states. Each of the *LF* passes are approximately 90-115s in length; thus, the pipeline is evaluated using a total of 3,188 256ms frames. Additionally, the network confidence threshold is tuned using 1,594 256ms frames.

Truth labels are produced for each cellphone in the *LF* scenarios to evaluate and tune the EVP system. The distance from a cellphone to the sUAS during a given pass determines the truth label. An sUAS less than 80m from a cellphone is labeled *sUAS*, and one outside the 80m range is labeled *Noise*. Each cellphone's score metrics are evaluated using only its truth data. The EVP system is evaluated using the combined truth data from all cellphones, which means that an sUAS within 80m of any cellphone is labeled as *sUAS* for the EVP system. The truth data distribution of the nine test scenarios of an individual device compared to the overall EVP is in Figures 6 and 7. Although the distribution of *Noise* to *sUAS* data is not equal in comparison, we evaluate the EVP performance using distribution-agnostic metrics (i.e., F1 score) to ensure that truth data imbalance between the EVP and individual devices does not imply misleading results.

Figure 9 shows the truth-labeled detection area for the EVP and individual cellphones. The red circle represents the 80m area around a cellphone. Additionally, the overlapping regions of the circles demonstrate multiple devices

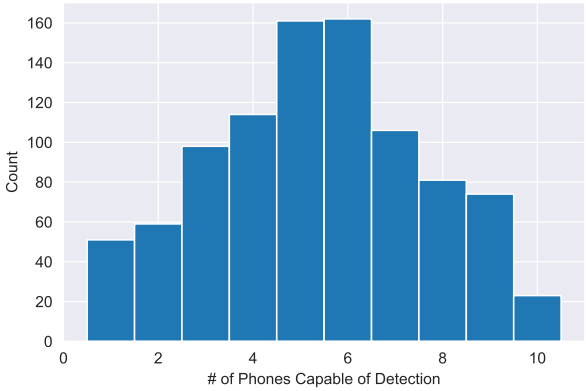

Fig. 8. Histogram of Detectable Devices

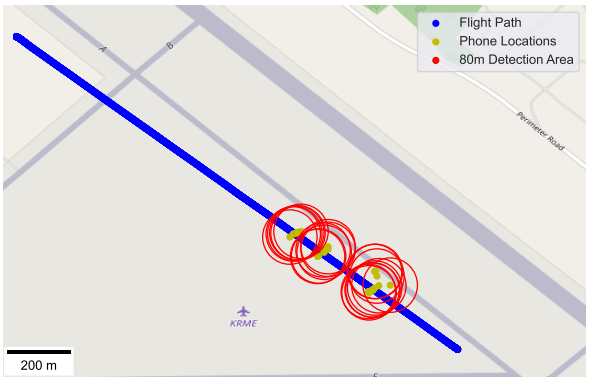

Fig. 9. Testing Scenario Detection Areas

Table 7. Hyper-parameter Tuning Values

Hyperparameter Name	Possible Value Set
ML Confidence Threshold	[0.40-0.92]
EVP Voting Threshold	[0.15-0.5]
PFSM States	0,2,4,8

within 80m of the sUAS. Figure 8 shows the distribution of overlapping detection areas in the truth data. On average, six cellphones detect the sUAS simultaneously, with a maximum of 10 and a minimum of one detectable device.

It is worth noting that although all phones are positioned to record during every scenario, software errors caused a subset of the phones not to record acoustics data. Therefore, the unusable cellphone data is not included during each test scenario. On average, 17 of the 28 cellphones record usable acoustics data for the given test scenarios, with a maximum of 20 and a minimum of 15.

4.7 EVP Hyper-parameter Tuning

Before testing the pipeline performance on nine *LF* test scenarios, the hyper-parameters of the pipeline are tuned to ensure that the confidence threshold of the ML model, ensemble vote threshold, and PFSM are set to the near-optimal values. The EVP is tuned to maximize accuracy using the truth-labeled sUAS detection values from Sub-Section 4.6. The tuning values are seen in Table 7. Three *LF* passes are exclusively used for hyper-parameter tuning. Then, the pipeline repeatedly runs from these scenarios over all possible confidence thresholds, voting thresholds, and state pairings. The F1-score of the EVP is used as the tuning metric to determine which hyper-parameter configuration to use.

4.8 System Evaluation

The EVP is evaluated on real-world flights from the Escape II *LF* scenario to measure EVP performance and resiliency against sensor dropout and erroneous predictions. All experiments report the averaged accuracy, precision, recall, F1-score, and F1 performance improvement compared to individual cellphone predictions. The equations for each metric used throughout the evaluation are in Table 8.

Table 8. Metric Equations

Metric	Equation
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
Recall	$\frac{TP}{TP+FN}$
Precision	$\frac{TP}{TP+FP}$
F1	$\frac{2 \cdot precision \cdot recall}{precision+recall}$

4.8.1 EVP Performance Versus Individual Cellphones. The first experiment compares the EVP performance to the individual cellphones' predictions over the nine test scenarios. The test methodology is straightforward. The EVP's performance is evaluated using truth data (e.g., sUAS within 80m of any cellphone is labeled *sUAS*, and outside of 80m is *Noise*). These results are compared to the average performance of all individual cellphones' predictions. Independent truth data is used for each cellphone within the constellation that records audio and makes predictions. All results are conveyed in terms of averaged F1-score, accuracy, recall, and precision. These metrics compare the aggregated individual cellphones' performances to the performance of the EVP. The EVP aims to outperform the average F1-score of individual cellphones by over 25%.

4.8.2 EVP Sub-component Comparison. The second experiment evaluates the benefits of the weighting and PFSM sub-components. The experiment methodology is identical to the first experiment, except that the EVP baseline performance is compared to the EVP configurations without *Weighting*, without the *PFSM*, and without either *Weighting* or *PFSM*. The results are recorded in terms of averaged F1-score, accuracy, recall, and precision. Without the weighting function, the ensemble vote threshold is set so that if 20% of the total amount of cellphones predict *sUAS*, the EVP predicts *sUAS*. The 20% threshold can also be seen as the null weighting value, meaning that all devices are equally weighted. Only the current prediction is considered in the configuration without the PFSM, and there is no prediction momentum. Without either sub-component, the EVP makes predictions based solely on the ensemble voting scheme without weighting or temporal stability. We expect the full EVP to outperform all EVP configurations with sub-components removed, as the sub-components are vital to ensuring that accurate sUAS detection is achieved.

4.8.3 EVP Resiliency Evaluation. The last experiment measures the EVP resiliency against cellphone dropout and mispredictions. Resiliency is vital because sensors are prone to failure, errors, and potentially malicious attacks. We measure this in two ways: cellphone dropout and cellphone mispredictions. We evaluate both using Algorithm 1. In the dropout experiment, we artificially remove a random subset of cellphones from each test scenario and re-evaluate the EVP without those devices included but with the EVP truth labeling remaining fixed. In the seven test conditions, we remove various quantities of cellphone devices $c \in \{1, 2, 3, 4, 5, 6, 10\}$ from the constellation. We then re-evaluate the EVP on the test scenarios five times to reduce outlier situations. In the dropout test, the $resiliency_{drop}$ function returns the EVP F1 score of the test scenario, t, with c cellphones removed from the constellation. The expectation is that a slight decrease in the number of functioning cellphones does not yield a noticeable variation in EVP performance because the cellphones operate independently and there are usually overlapping devices within 80m of sUAS during the detection frames.

The second part of Experiment #3 assesses resiliency against malicious ML model mispredictions, such as data or cyber-attacks. A subset of the cellphone predictions are inverted during each test scenario to mimic this

situation. The mispredicting cellphone case quantities are taken from $c \in \{1, 2, 3, 4, 5, 6, 10\}$ mispredicting devices. In each of these cases, the cellphones which predict incorrectly are randomly selected, and each *LF* scenario used for EVP evaluation is re-evaluated five times. The algorithm for this test is identical to the last, in Algorithm 1. However, now the $resiliency_{mispred}$ is a function that evaluates the EVP with c random phones' predictions inverted during the test scenario, t, and returns the EVP F1 score for that pass. The presumption is that the EVP will remain accurate while the number of mispredicting devices averages less than one per the median number of capable of detecting devices (6), as seen in Figure 8. However, as the number of incorrect predictors nears the median number of detectable devices, performance is expected to decrease due to the Bayesian nature of the ensemble-based approach.

Algorithm 1 Resiliency Test Algorithm

$cases \leftarrow [1, 2, 3, 4, 5, 6, 10]$
$results \leftarrow \{1 : [], 2 : [], 3 : [], 4 : [], 5 : [], 6 : [], 10 : []\}$ $(Dictionary)$
for $c \in cases$ **do**
  $case_{f1} \leftarrow []$
  for $t \in test_{scenarios}$ **do**
    $pass_{f1} \leftarrow []$
    for $k \in [1, \ldots, 5]$ **do**
      if $Test_{dropout} == TRUE$ **then**
        $pass_{f1}.append(resiliency_{drop}(c, t))$
      else
        $pass_{f1}.append(resiliency_{mispred}(c, t))$
      end if
    end for
    $case_{f1}.append(avg(pass_{f1}))$
  end for
  $results\{c\} \leftarrow avg(case_{f1})$
end for

5 RESULTS

In this section, we present the research results. We break the results into our two research stages. First, we present the research effort's first stage: the ML model selection tuning and results. Then, stage two evaluates the ML model within the EVP. The EVP tuning results are detailed, and then the EVP evaluation results are presented, including the EVP performance evaluation, sub-component evaluation, and the EVP resiliency test.

5.1 Stage One: Model Selection Results

After tuning the SVM and training all models, the four ML models are compared using a 20% sequestered testset from the training dataset established in Section 4.3. The model is selected by considering performance, prediction speed, and memory usage, as these are all essential characteristics to consider if the ML model would be copied onto devices in a real-world environment.

Table 9 reports the performance of all models. The RBF-SVM, tuned using 5-fold cross-validation to have a C-value of 100 and a gamma-value of 0.1, outperforms all other models, with the ANN (Version 3) performing second best. Therefore, these two models are further compared using prediction speed and memory usage.

Table 9. Performance Evaluation Results

Model	Accuracy	Precision	Recall	F1 Score
Null	0.506	0.00	0.00	0.00
LDA	0.678	0.691	0.630	0.659
QDA	0.708	0.748	0.616	0.675
RBF-SVM	0.882	0.889	0.898	0.892
ANNv1	0.830	0.877	0.763	0.816
ANNv2	0.823	0.844	0.786	0.814
ANNv3	0.841	0.849	0.824	0.837

Table 10. Performance Evaluation Results

Model	Memory Usage	Prediction Time
RBF-SVM	6,714KB	2.17ms
ANNv3	294KB	0.125ms

The ANN performs predictions 2.045ms faster than the SVM on the evaluation machine. The prediction speed is calculated using a program-based timer that records each ML model's total time required to predict N test frames and divided by the N frames. Additionally, the ANN uses 6,420KB less storage than the RBF-SVM. The memory usage is determined by saving both models to disk and comparing the reported memory usage in kilobytes (KB). The results of these comparisons are in Table 10.

After a decision analysis, it is determined that the RBF-SVM's slow and large memory footprint is not worth the slight improvement in performance compared to the ANN. Therefore, the ANN (Version 3) is used as the ML model distributed across the constellation of cellphones and integrated into the fusion pipeline.

Compared to related works that have developed ML-based sUAS detection models, the model performs worse than some models that use much higher fidelity audio while performing similarly to others. However, the ANN uses a low-dimensional representation of the input data (40 Coefficients) and low-fidelity audio (8KHz). Still, it achieves accurate detection results that are comparable to STFT and spectrogram-based models (e.g., [1] and [22]) that have a higher-dimensional input representation. This comparison demonstrates that the low-fidelity audio from commercial-off-the-shelf cellphones can produce high-fidelity sUAS detection models. Next, the ANN is added to the EVP. The following sub-section discusses the EVP performance results and resiliency results.

5.2 Stage Two: EVP Evaluation

After selecting ANNv3 for implementation within the EVP, ANNv3 is copied to the EVP, and EVP performance is evaluated. First, we present the EVP hyper-parameter tuning results and then examine the EVP performance, sub-component, and resiliency evaluation results. Combined, these experiments demonstrate the value of using an ensemble-based approach for sUAS detection that is resilient against microphone errors and dropout and capable of making real-time sUAS detection decisions.

5.2.1 Pipeline Hyper-parameter Tuning. The results from tuning the EVP hyper-parameters determine that the best confidence threshold is 0.90 with a 0.20 voting threshold. Additionally, the PFSM tuning results set the FSM to 8 states, which means that if the EVP has been predicting *Noise*, the voting scheme would need to predict *sUAS* four consecutive times (1.024s) to change the EVP output prediction. The high confidence threshold signifies

Table 11. Individual versus Full EVP Performance Comparison (Average across 9 test scenarios)

-	Accuracy	Recall	Precision	F1
Individual	0.936	0.623	0.572	0.582
Full EVP	0.926	0.829	0.884	0.846

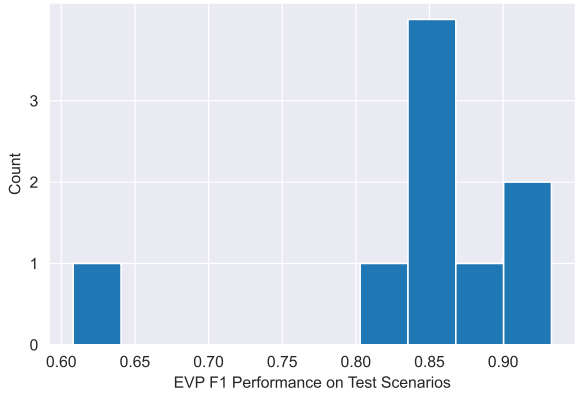

Fig. 10. Histogram of EVP F1 Performance on Test Scenarios

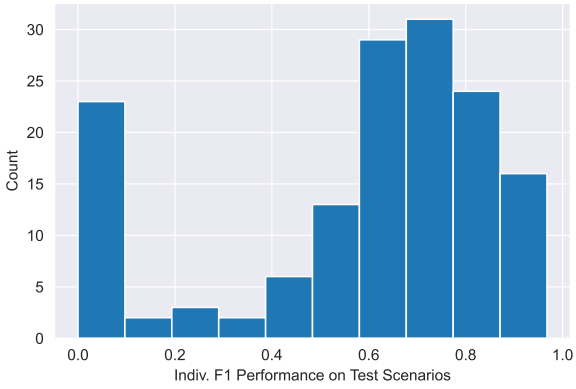

Fig. 11. Histogram of Cellphones' F1 Performances on Test Scenarios

that the ML model must make extremely confident predictions to predict the sUAS class (thus reducing the false positive rate). The values for the confidence level, voting threshold, and PFSM states remain set for the remainder of the pipeline evaluation process.

5.2.2 EVP Performance Results. Across the nine test scenarios, the EVP outperforms the averaged F1-score of individual devices by 25%. Additionally, these results show that the ensemble of weak predictors improves overall sUAS detection. Table 11 shows the EVP results and individual cellphones. Figures 10 and 11 show the histograms of the F1 performance distributions of all individual cellphones and the EVP in the test scenarios.

Table 12 demonstrates the EVP performance compared to related sUAS detection works. The EVP achieves comparable detection results while relying on lower-fidelity data from cellphone microphones and is resilient against individual cellphone errors. Additionally, the EVP is evaluated using unseen testing scenarios recorded in a real-world testing environment. Thus, the performance results provide a realistic perspective on how the EVP would perform in the real world.

These results also highlight a few key findings. The first is the performance degradation of the individual cellphones compared to the initial ML model test-set performance. Although the test-set-split performance reported in Table 9 suggests that the ANN can achieve an F1-score of 0.837, in real-world test scenarios, the ANN copied onto the cellphones achieves an average of 0.582 F1-score. The degradation rationale is due to the variation in training and testset range distributions. Figures 12, 13 demonstrate the *sUAS* class range distributions for the training dataset versus the test scenarios. Although all data that is used for training, testing the ML models, and evaluating the EVP comes from the same data collection, most of the training data frames are at

Table 12. Performance compared to Related Works

	Model Type	Features	Data Quality	Accuracy	Precision	Recall	F1-Score
Jeon [11]	RNN	MFCC	44.1KHz	-	0.7953	0.8066	0.8009
Shi [23]	HMM	MFCC	44.1KHz	1.00	-	-	-
Salman [20]	RBF-SVM	Mult. Feat. Types	44.1KHz	0.9990	1.00	0.9980	0.9990
Seo [22]	CNN	STFT	44.1KHz	0.8997	0.7938	0.9638	0.8706
Al-Emadi [1]	CNN	Spectrogram	16KHz	0.9638	0.9624	0.9560	.9590
Casabianca [4]	CNN	Mel-Spectrogram	44.1KHz	0.91044	-	-	0.91747
Kolamunna [13]	RNN	MFCC	44.1KHz	0.94-0.97	-	-	0.96-0.98
EVP	ANN	MFCC	8KHz	0.926	0.829	0.884	0.846

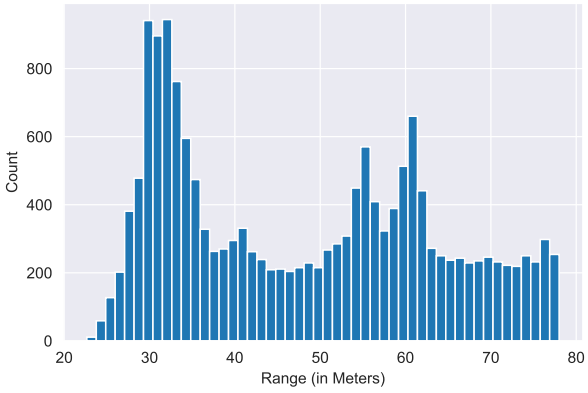

Fig. 12. Histogram of Training Dataset sUAS Ranges (*sUAS* Class)

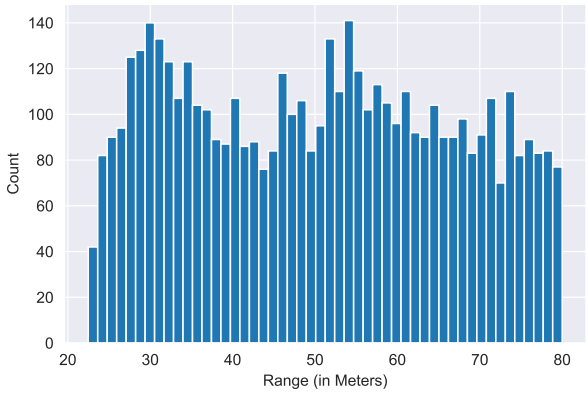

Fig. 13. Histogram of Test Scenario sUAS Ranges (*sUAS* Class)

fixed intervals (30m and 50-60m ranges), leading to a non-uniform distribution in the training dataset. However, in the continuous nature of the test scenarios, there is a much more uniform distribution of ranges.[2]

Another key finding that explains the performance variation is the cellphones themselves. Although the ML model trains on all cellphones, Figure 14 shows the variability in cellphones' predictions. Eight cellphones fail to record audio or correct GPS values during test scenarios; thus, they are omitted. Others record audio but have erroneous results.

5.2.3 Sub-component Comparison Results. We now evaluate the performance improvements after adding sub-components to the EVP. Table 13 shows the performance of the EVP without the Weighting Function, the PFSM, or both. These results show that without weighing the EVP votes by the log energy of each cellphone's acoustic frame, the EVP makes less precise predictions. Additionally, without the PFSM, the EVP makes less stable predictions, decreasing performance. Without either sub-component, the ensemble voting scheme slightly

[2]A large variety of data augmentation techniques were used to try and reduce this variation in performance (pitch-shifting training data, amplitude-shifting training data, and supplementing additional audio from online sources) [13]. However, all augmentation techniques did not improve EVP (tested using QDA, SVM, and ANN) performance in the evaluation scenarios. We suspect this is caused by the variation in internal microphones and the low-fidelity data provided.

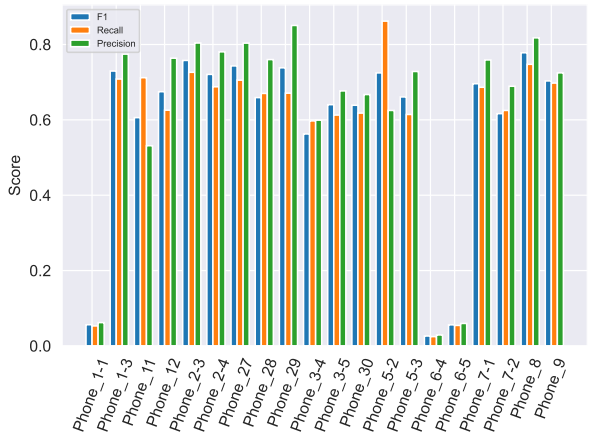

Fig. 14. Test Scenario Performance of Cellphones

Table 13. Sub-component Performance Comparison (Average across test scenarios)

Weighting	FSM	Accuracy	Recall	Precision	F1
N	Y	0.868	0.512	0.969	0.649
Y	N	0.902	0.809	0.813	0.806
N	N	0.877	0.537	0.976	0.681
Y	Y	0.926	0.829	0.884	0.846

improves the performance. Combined, these results demonstrate the value of the weighting function and PFSM in the EVP.

5.2.4 Resiliency Evaluation Results. The EVP resiliency results are now presented. We first demonstrate the EVP's performance despite cellphone dropout and then examine the EVP's performance despite erroneous predictions.

Figure 15 presents the F1 performance of the EVP when cellphones are disabled. This figure demonstrates that cellphone dropout (which in a real-world scenario could be caused by software errors or malicious destruction of sensors) does not cause a significant decrease EVP performance. The EVP can withstand cellphone dropout and make accurate sUAS detection predictions for the constellation. Even with the loss of 10 unusable cellphones, the F1 performance only degrades slightly below 0.75.

Likewise, Figure 16 demonstrates that EVP performance remains resilient when up to two of the predicting cellphones consistently make incorrect predictions. In contrast, the average F1-score of individual cellphones linearly decreases as the number of incorrect predictors increases. However, three incorrectly predicting cellphones severely decrease the EVP performance. This critical point can be explained using the Bernoulli Trial equation, seen in Equation 11. Assuming the mean number of sUAS detecting cellphones during any given *sUAS* labeled 0.256ms frame is six and the mean number of cellphones within a test scenario is 17, three mispredicting devices suggest that roughly one out of six devices is likely mispredicting. Assuming cellphones detect sUAS with 66% balanced accuracy and a simple majority of these devices are required to predict sUAS, this yields a 47.5% detection rate. Thus, performance degradation rapidly occurs as the probability of detection decreases beyond 50% (i.e., the number of mispredicting cellphones approaches the number of detectable devices for a truth-labeled *sUAS* frame).

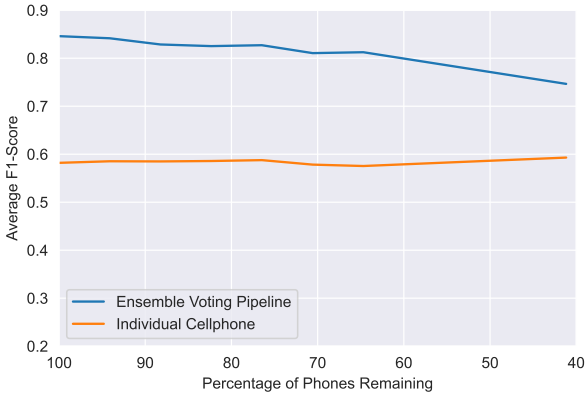

Fig. 15. EVP Performance Despite Cellphone Removal

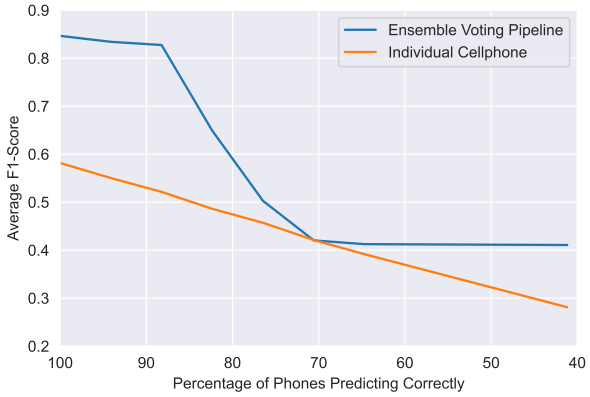

Fig. 16. EVP Performance Despite Erroneous Predictors

$$P(n \geq 3) = \sum_{n=3}^{6} 6Cn \cdot 0.65^{n-1} \cdot 0.35 \cdot 0.35^{6-n} = 0.475 \tag{11}$$

These results show that the EVP performs with an average F1-score of 0.846, which rivals other sUAS detection works that require high-fidelity audio sensors and have achieved limited testing in real-world environments. Additionally, our methods can remain accurate when up to two sensors consistently make incorrect predictions or when six cellphones are disabled during the test scenarios. Overall, these results suggest the value of using low-fidelity sensors to achieve both accurate and resilient sUAS detection.

6 CONCLUSION

sUAS pose a dangerous threat from state and non-state actors that demand that easily accessible and fault-tolerant sensing methods be developed to match the low-cost and available nature of sUAS. This article presents a novel cellphone acoustics-based sUAS detection method, Ensemble Voting Pipeline. The proposed, tested, and evaluated EVP produces accurate sUAS detection results with acoustics data from cellphone microphones. The pipeline is composed of an ML model copied across all cellphones and three additional sub-components (Ensemble Voting Scheme, weighting function, and an erroneous prediction FSM). Using nine real-world test scenarios, the pipeline achieves an average F1-score of 0.846 sUAS detection which outperforms the performance of individual cellphones' average F1-score of 0.582. These results confirm the overall research hypothesis that using the EVP with the distributed cellphone sensor network allows for improved sUAS detection compared to individual cellphone microphones and is resilient against sensor disruptions and errors. These results demonstrate the EVP's ability to achieve resilient and accurate sUAS detection while remaining scalable, fault-tolerant, and not having a single sensor point of failure. In turn, these methods can provide resilient airspace awareness in real-world settings that suppress the threat of sUAS in sensitive areas.

6.1 Future Work

Future work anticipates re-collecting data with more standardized devices and higher fidelity audio to try and mitigate this issue. However, the inaccurate nature of the cellphones helps to demonstrate further the power of an ensemble-based detection methodology that is resistant to individual errors by any single predictor.

ACKNOWLEDGMENTS

The views expressed in this document are those of the author and do not reflect the official policy or position of the United States Air Force, the United States Department of Defense, or the United States Government. This material is declared a work of the US Government and is not subject to copyright protection in the United States. Additionally, we would like to thank our sponsor AFRLRI, for the data, support, and expert knowledge on acoustics and the sUAS domain.

REFERENCES

[1] Sara Al-Emadi, Abdulla Al-Ali, and Abdulaziz Al-Ali. 2021. Audio-Based Drone Detection and Identification Using Deep Learning Techniques with Dataset Enhancement through Generative Adversarial Networks. *Sensors* 21, 15 (2021). https://doi.org/10.3390/s21154953

[2] Pau Arce, David Salvo, Gema Piñero, and Alberto Gonzalez. 2021. FIWARE Based Low-Cost Wireless Acoustic Sensor Network for Monitoring and Classification of Urban Soundscape. *Comput. Netw.* 196, C (sep 2021), 10 pages. https://doi.org/10.1016/j.comnet.2021.108199

[3] Selene Caro-Via, Ester Vidaña-Vila, Gerardo José Ginovart-Panisello, Carme Martínez-Suquía, Marc Freixes, and Rosa Ma Alsina-Pagès. 2022. Edge-Computing Meshed Wireless Acoustic Sensor Network for Indoor Sound Monitoring. *Sensors* 22, 18 (2022). https://doi.org/10.3390/s22187032

[4] Pietro Casabianca and Yu Zhang. 2021. Acoustic-Based UAV Detection Using Late Fusion of Deep Neural Networks. *Drones* 5, 3 (2021), 14. https://doi.org/10.3390/drones5030054

[5] S. Deligeorges, C. Lavey, G. Cakiades, J. George, Y. Wang, F. N Ez, and F. Doyle. 2015. A mobile acoustic sensor fusion network using biologically inspired sensors and synchronization. In *2015 18th International Conference on Information Fusion (Fusion)*. IEEE, Washington, DC, 1717–1723.

[6] Aurélien Géron. 2017. *Hands-on machine learning with Scikit-Learn and TensorFlow : concepts, tools, and techniques to build intelligent systems.* O'Reilly Media, Sebastopol, CA, 284–288.

[7] Aurélien Géron. 2017. *Hands-on machine learning with Scikit-Learn and TensorFlow : concepts, tools, and techniques to build intelligent systems.* O'Reilly Media, Sebastopol, CA, 191–195.

[8] Aurélien Géron. 2017. *Hands-on machine learning with Scikit-Learn and TensorFlow : concepts, tools, and techniques to build intelligent systems.* O'Reilly Media, Sebastopol, CA, 142.

[9] Milton Isla, Anthony Christe, and Tyler Yoshiyama. 2022. *Redvox-python-sdk*. Red Vox. https://github.com/RedVoxInc/redvox-python-sdk

[10] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2021. *An introduction to statistical learning : with applications in R* (2 ed.). Springer, Berlin, Germany, 367–402.

[11] Sungho Jeon, Jong-Woo Shin, Young-Jun Lee, Woong-Hee Kim, YoungHyoun Kwon, and Hae-Yong Yang. 2017. Empirical study of drone sound detection in real-life environment with deep neural networks. In *2017 25th European Signal Processing Conference (EUSIPCO)*. IEEE, Kos, Greece, 1858–1862. https://doi.org/10.23919/EUSIPCO.2017.8081531

[12] Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. 2017. Self-Normalizing Neural Networks. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, California, USA) *(NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 972–981.

[13] Harini Kolamunna, Thilini Dahanayaka, Junye Li, Suranga Seneviratne, Kanchana Thilakaratne, Albert Y. Zomaya, and Aruna Seneviratne. 2021. DronePrint: Acoustic Signatures for Open-set Drone Detection and Identification with Online Data. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5 (3 2021). Issue 1. https://doi.org/10.1145/3448115

[14] Martin Liggins, David Hall, and James Llinas. 2009. *Handbook of Multisensor Data Fusion* (2 ed.). CRC Press, New York, New York, 1.

[15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[16] D. Prabakaran and S. Sriuppili. 2021. Speech Processing: MFCC Based Feature Extraction Techniques- An Investigation. *Journal of Physics: Conference Series* 1717, 1 (jan 2021), 012009. https://doi.org/10.1088/1742-6596/1717/1/012009

[17] K. Sreenivasa Rao and K. E. Manjunath. 2017. *Speech Recognition Using Articulatory and Excitation Source Features* (1st ed.). Springer Publishing Company, Incorporated, New York, NY, USA, 85–92.

[18] Saad Ur Rehman and Muhammad Amjad Iqbal. 2019. Feature extraction and classification of UAV's acoustic signal using 4-microphones array in a real noisy environment. In *Eleventh International Conference on Signal Processing Systems*, Kezhi Mao (Ed.), Vol. 11384. International Society for Optics and Photonics, SPIE, Chengdu, China, 113840E. https://doi.org/10.1117/12.2559543

[19] János Sallai, Will Hedgecock, Péter Völgyesi, András Nádas, Gyorgy Balogh, and Ákos Lédeczi. 2011. Weapon classification and shooter localization using distributed multichannel acoustic sensors. *J. Syst. Archit.* 57 (2011), 869–885.

[20] Soha Salman, Junaid Mir, Muhammad Tallal Farooq, Aneeqa Noor Malik, and Rizki Haleemdeen. 2021. Machine Learning Inspired Efficient Audio Drone Detection using Acoustic Features. In *2021 International Bhurban Conference on Applied Sciences and Technologies (IBCAST)*. IEEE, Islamabad, Pakistan, 335–339. https://doi.org/10.1109/IBCAST51254.2021.9393232

[21] Alexander Sedunov, Darren Haddad, Hady Salloum, Alexander Sutin, Nikolay Sedunov, and Alexander Yakubovskiy. 2019. Stevens Drone Detection Acoustic System and Experiments in Acoustics UAV Tracking. In *2019 IEEE International Symposium on Technologies for Homeland Security (HST)*. IEEE, Woburn, MA, USA, 1–7. https://doi.org/10.1109/HST47167.2019.9032916

[22] Yoojeong Seo, Beomhui Jang, and Sungbin Im. 2018. Drone Detection Using Convolutional Neural Networks with Acoustic STFT Features. In *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, Auckland, New Zealand, 1–6. https://doi.org/10.1109/AVSS.2018.8639425

[23] Lin Shi, Ishtiaq Ahmad, Yujing He, and Kyunghi Chang. 2018. Hidden Markov model based drone sound recognition using MFCC technique in practical noisy environments. *Journal of Communications and Networks* 20 (10 2018), 509–518. Issue 5. https://doi.org/10.1109/JCN.2018.000075

[24] RedVox Sound. 2022. *Infrasound Recorder*. RedVox Sound. Retrieved September 17, 2022 from https://apps.apple.com/us/app/infrasound-recorder/id969566810

[25] Chris Vallance. 2022. Ukraine sent dozens of 'dronations' to build Army of Drones. https://www.bbc.com/news/technology-62048403

[26] Bowon Yang, Eric T. Matson, Anthony H. Smith, J. Eric Dietz, and John C. Gallagher. 2019. UAV Detection System with Multiple Acoustic Nodes Using Machine Learning Models. In *2019 Third IEEE International Conference on Robotic Computing (IRC)*. IEEE, Naples, Italy, 493–498. https://doi.org/10.1109/IRC.2019.00103

[27] Peter Zulch. 2022. *2021 ESCAPE II Data Collection*. Technical Report. AFRL/RIGC.

Received XX February XX; revised XX February XX; accepted XX February XX

III. Paper: Cellphone-Based sUAS Range Estimation: A Deep-Learning Approach

The following paper, "Cellphone-Based sUAS Range Estimation: A Deep-Learning Approach," was submitted to the 2023 IEEE International Conference on Acoustics, Speech and Signal Processing; it is pending acceptance.

CELLPHONE-BASED SUAS RANGE ESTIMATION: A DEEP-LEARNING APPROACH

Ryan D. Clendening[⋆⋆] *Richard Dill*[⋆] *Brett J. Borghetti*[⋆]
Brett Smolenski[⋆⋆] *Darren Haddad*[†] *Douglas D. Hodson*[⋆]

[⋆] Air Force Institute of Technology Dept. of Electrical & Comp. Eng. 2950 Hobson Way, WPAFB, OH
[⋆⋆]North Point Defense, Inc 184 Brooks Road, Rome, NY
[†]Air Force Research Labs Info. Exploit. Branch 525 Brooks Road, Rome, NY

ABSTRACT

Small Unmanned Aircraft Systems (sUAS) are accessible platforms that pose a security threat. These threats warrant affordable and accurate methods for tracking sUAS. We apply neural network-based methods to predict sUAS range from cellphone acoustic recordings; the data comes from twenty-eight cellphones recording three different sUAS that fly over the devices. The timestamped acoustics data is transformed into 0.5s Mel-spectrograms frames and 0.5s raw audio frames. Truth values are calculated using euclidean distance from the sUAS to a cellphone and split into four range classes. The data is sequestered into an 80/20 training-test split and is used to train three different architectures. The 2DCNN architecture outperforms the other architectures (1DCNN and 2DCRNN). The 2DCNN is then re-trained to generalize sUAS range with various sUAS models and achieves an average Macro-F1 score of 0.7492 across different sUAS models. The results show that deep-learning-based sUAS ranging with cellphones is an effective and low-cost method for accurately tracking sUAS.

Index Terms— sUAS Ranging, Deep-Learning, Acoustics Processing, Sound Source Tracking

1. INTRODUCTION

The accessibility of small Unmanned Aircraft Systems (sUAS) presents significant security risks to the public and military operations. In 2017, a Canadian passenger jet collided with a hobbyist drone, causing damage to the wing and risking passenger lives [1]. In the Russian-Ukraine Conflict, sUAS played a significant role in reconnaissance collection and artillery attacks [2]. The low profile and highly accessible nature of sUAS demands sUAS defense strategies that are affordable, scalable, and accurate. Although other more expensive tracking methods exist, this effort attempts to use affordable and scalable sensing devices to counter the ubiquitous nature of sUAS. These methods can be used to protect, defend, and track sUAS throughout restricted airspace.

This research investigates how cellphones can provide sUAS tracking capabilities by estimating the sUAS range from a mobile device's microphone. sUAS emit sound from the rotating motors and propellers that produce lift and velocity. Sound is generated, resulting in a fundamental frequency (or frequencies) between 0-2kHz range and its harmonics [3]. Additionally, the physical vibration of sUAS produces additional acoustic noise, which tends to be at high frequencies (3KHz-4KHz) [4]. We use four datasets of recorded sUAS flights, named after the recorded sUAS, with corresponding range truth data: IF, Matrice, Phantom, and Combined. Combined is the superset of the single sUAS datasets. A portion of the single sUAS datasets is sequestered for evaluation later. The acoustics data converts into 0.5s Mel-spectrogram format for the 2-dimensional models and 0.5s raw audio for the 1D model. The 2D convolutional neural network (2DCNN), 1D convolutional neural network (1DCNN), and the 2D convolutional recurrent neural network (2DCRNN) are each trained from Combined. The best-performing architecture is then re-trained using all four datasets and evaluated on three sequestered testsets. The hypothesis is that the 2DCNN is the best-performing architecture and can achieve greater than 70% balanced accuracy on all testsets. Additionally, we hypothesize that the 2DCNN trained using Combined outperforms each model trained from only a single sUAS dataset and can achieve an F1 score over 80% when an sUAS is within 40m. The additional metric, within 40m, is chosen because a cellphone needs to recognize when an sUAS is close to a sensor and in high-threat airspace.

Although other researchers have achieved accurate sUAS localization results from high-fidelity acoustic arrays, this research contributes a method to locate sUAS from low-fidelity acoustic sensors within cellphones that is scalable and accurate. The paper is organized as follows: Section 2 presents related sUAS tracking research. Section 3 offers the research methodology and model architectures. Lastly, Section 4 provides the results.

2. RELATED WORKS

Researchers commonly employ two acoustics source localization methods: Direction of arrival (DoA) and time difference of arrival (TDoA) [5]. DoA is calculated using multiple signal classification (MUSIC). TDoA is calculated using generalized cross-correlation (GCC) and can produce highly accurate localization results for systems with multiple nodes [6]. Researchers have proposed using deep-learning to supplement TDoA calculation; however, these techniques require fixed sensor locations [7]. A summary of sUAS localization efforts follows.

Sedunov et al. developed an sUAS detection and localization system using a collection of acoustic arrays [3]. Each array consisted of 15 custom-built microphones, and the arrays were spaced at a distance of 80-120m. The researchers applied the Steered-Response Phase Transform (SRP-PHAT) to produce the direction-of-arrival (DoA). Sedunov et al. achieved an average 4.7 degree DOA precision and 200m range.

Kyritsis et al. developed an sUAS localization technique using DoA estimation from a four-element acoustic array [8]. The

researchers determined that the maximum detectable range of the sUAS in a rural environment was 77m and that they could achieve accurate DoA estimation ($\leq 5°$).

Additionally, previous contributions have investigated sUAS ranging with a high-fidelity microphone. These efforts used high-quality recording equipment with relatively large bit depths and sample rates to achieve accurate sUAS range estimation [9, 10, 11].

Although limited, previous efforts on sUAS localization rely on sophisticated acoustic sensors and arrays to produce impressive results. In contrast, this research is the first effort to estimate sUAS range from ordinary cellphones void of sophisticated microphones. Furthermore, we introduce a machine learning method to estimate sUAS location without explicitly calculating TDoA or DoA.

3. METHODOLOGY

This section describes a novel method to estimate sUAS range without requiring statistical methods like TDoA or DoA. These methods provide a deep-learning network capable of estimating sUAS range from a single acoustic device (i.e., a cellphone). This contribution enables constellations of cellphones to provide persistent sUAS awareness without being limited to fixed, high-fidelity acoustic sensor configurations. We first state our research assumptions; then examine the dataset used. We then explain the features extracted as the inputs to the networks and the deep-learning models used in the experiments. Finally, we present an overview of the experiment design and objectives demonstrating that cellphone-based sUAS range estimation is achievable with a deep-learning-based approach.

This research makes the following assumptions.

- sUAS targets have different acoustic signatures
- sUAS acoustic signatures change in predictable ways depending on the range that their acoustic emissions propagate
- Accuracy decreases with an increase in environmental noise
- 80m is the maximum detection range for an sUAS

Three sUAS flight scenarios source the datasets. There are ten hover passes, 36 short passes, and 19 long passes. The sUAS fly at 100ft AGL (above ground level) for each flight, move between 10-20 kn, and fly directly over the sensor constellation during every pass. The sensor constellation contains three clusters of cellphones that span a range of approximately 300m. In hover flights, an sUAS hovers at 33m. In short flights, a single sUAS flies in a straight line for 410m. Lastly, in long flights, an sUAS flies 1.5km across the constellation of cellphones. The entire data collection is conducted at an active airfield; thus, environmental noise (i.e., propeller noise from airplanes) is present throughout all the data.

Twenty-eight cellphones are positioned across the sUAS flight path. The cellphone positioning ensures that the sUAS range is generalized for varying doppler effects, internal microphones, and device orientations. The phones capture acoustic data using RedVox, a multi-modal data collection tool [12]. RedVox records acoustic data with microsecond granularity. All acoustic data is sampled at 8KHz and converted to audio (i.e., .wav) files. Table 1 displays the cellphone and app configurations.

The cellphones collect acoustic data for three different sUAS: Inspired Flight 1200 [13], DJI Matrice 600 [14], DJI Phantom 4 Pro [15]. These sUAS differ in shape, weight, and acoustic signatures. The DJI Phantom 4 flies short and long flights, the Inspired Flight flies hover and short flights, and the Matrice flies hover, short, and long flights. Therefore, the range distributions across each sUAS vary slightly.

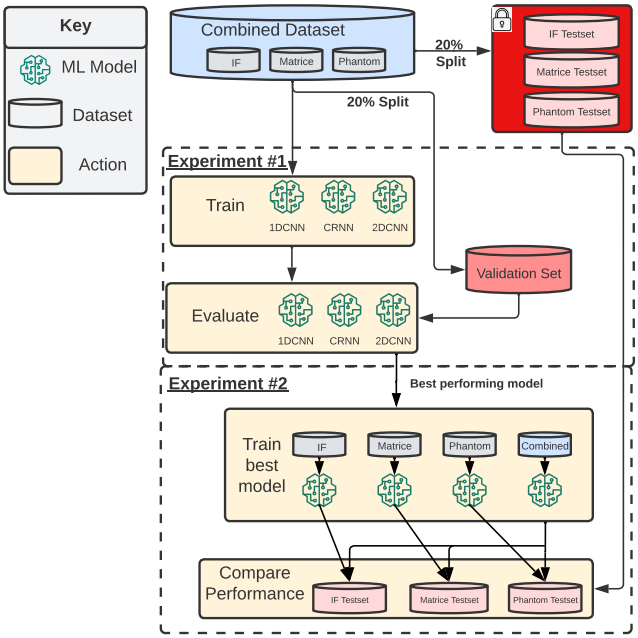

Fig. 1. Methodology Flow Diagram

Table 1. Sensor Configuration

Device Count	13 Apple Iphones 15 Samsung Androids
Apple App Version	V4.0.2.4
Android App Version	V3.3.1.2
File Type	Red Vox API 1000
Audio Format	PCM Floating Point

This research uses four datasets. Each dataset contains 0.5s raw audio samples and a range class for that given frame. Although frame length is fixed for this effort, we expect the sUAS range truth data to become more ambiguous as frame length per sample increases. Therefore, an increase in frame length would likely cause a decrease in classification performance. The first three datasets consist of flights flown by each sUAS model, the Inspired Flight 1200 [13], DJI Matrice 600 [14], and DJI Phantom 4 Pro [15]. Each of the three individual sUAS datasets has a 20% testset split that is sequestered before training. To maintain evaluation integrity, the training data of Combined exclusively contains the training data from the three other datasets (and vice versa for test data). The truth data is separated into four classes by distance in meters: $y \leq 40$ (Class 0), $40 < y \leq 60$ (Class 1), $60 < y \leq 80$ (Class 2), or $y > 80$ (Class 3). These classes are chosen to represent valuable proximity dividers for an sUAS in flight. If an sUAS is over 80m away, the cellphone receives little to no acoustic noise from the sUAS. However, if an sUAS is directly above a cellphone, the distance is less than 40m away (all flights are conducted at an altitude of 33m). Truth data within two meters of the class separations are removed to account for sUAS movement within the 0.5s frames. Table 2 provides class breakdown and dataset sizes.

The research effort conducts two experiments to evaluate if sUAS range estimation is achievable with convolutional deep-learning networks. These methods are outlined in the methodology flow diagram in Figure 1. The methodology depicted is as fol-

Table 2. Dataset Sizes and Class Distributions

Dataset	<40m	40-60m	60-80m	>80m	Total Time
IF	590.5s (40.5%)	250.5s (17.2%)	187.5s (12.9%)	429.5s (29.4%)	1458s
Matrice	1929.5s (37.6%)	657s (12.8%)	1048.5s (20.44%)	1493s (29.0%)	5128s
Phantom	1045s (25.7%)	850s (21.0%)	676s (16.7%)	1485.5s (36.6%)	4056.5s
Combined	3565s (33.5%)	1757.5s (16.5%)	1912s (18.0%)	3408s (32.0%)	10,642.5s

lows. Prior to the first experiment, a 20% testset is taken from each dataset. Then, Experiment One is conducted. The first experiment determines which model architectures best classify sUAS range from the Combined dataset. This experiment uses Combined as the baseline dataset to train and evaluate the 2DCNN, 2DCRN, and 1DCNN. Each architecture's score is reported in terms of macro F1-score, the arithmetic mean of each class's F1-score, which is a combination of recall and precision ($\frac{TP}{TP+\frac{1}{2}(FP+FN)}$). Three different model architectures are used throughout the experimentation process. The first is a 2DCNN, a specific type of neural network that has shown particular strength in sUAS acoustics tasks [16, 17]. The second architecture is a 2DCRNN, which has shown promise in various sound localization tasks and combines the strengths of a CNN with the temporal memory of recurrent neural networks [18, 19]. The last architecture is a 1D convolutional neural network (1DCNN). A 1DCNN exhibits similar performance compared to RNNs in various time series prediction tasks; however, a 1DCNN trains in a fraction of the time. 1DCNN is chosen for this research effort as it is much more computationally efficient to train than other RNN architectures while achieving positive results in raw audio classification problems [20].

The model architectures have different input formats. The 2DCNN and 2DCRNN receive data input in Mel-Spectrogram format, whereas the 1DCNN has raw formatted input. The Mel-spectrogram represents raw acoustics data in the frequency domain while preserving the time-domain. In line with previous research efforts [18, 21], Librosa, [22], converts the 0.5s raw acoustic frames as input into the 2DCNN and 2DCRNN. Each data frame size is 8x128, which contains 128 Mel-frequency bins with an FFT length of 2048 and a hop length of 512. However, the 1DCNN input is raw 0.5s audio clips (4000x1 array); thus, the data is not transformed into the Mel-Spectrogram form before inputting into the network. A 20% validation set split is used to determine the best-performing architecture. All architectures' validation results are compared to a naive alternative in which the model predicts the majority class in the training dataset. The architecture with the highest validation set performance is selected for Experiment Two.

Tables 3, 4, and 5 display the three model architectures; each layer of the respective networks is ordered sequentially from top to bottom. The convolution layers in the model architecture tables are represented by (number of filters)@(receptive field). Additionally, BN stands for batch normalization, which is applied prior to the activation function. In the dense layers, the number in the size column signifies the number of perceptrons within the layer.

Experiment Two determines how accurately the best-performing model generalizes across the three different sUAS types and evaluates if the model meets the hypothesized criteria established in Section 1. The best model is trained on each of the four datasets, and then each of the individual sUAS-trained models is compared to the Combined trained model. Three tests evaluate the testset performance of each individual sUAS-trained model to the model trained using Combined. F1 and balanced accuracy (the arithmetic mean of the recall scores of the four range classes) are used to compare the performance of the deep-learning models. The Individual sUAS dataset' testsets are withheld from Combined and preserved for model evaluation. These tests determine if the model trained using multiple sUAS models can extract model-agnostic sUAS range features that enhance the network's ability to generalize across the different sUAS models. Additionally, this experiment assesses the merits of deep-learning-based sUAS ranging with cellphones.

4. RESULTS

Experiments one and two evaluate the research objective of determining if acoustics-based sUAS range estimation is achievable using deep-learning. Additionally, these experiments demonstrate if deep-learning-based sUAS range prediction is generalizable across multiple sUAS targets. The first experiment reports that the 2DCNN deep-learning architecture best achieves sUAS range prediction. The second experiment's objective is to determine if the 2DCNN model generalizes sUAS range across various sUAS types and to evaluate how accurately our model predicts range. Experiment Two also assesses the guiding research hypothesis that a deep-learning model can achieve over 70% balanced accuracy and an F1 score within 40 meters above 80%.

Table 6 shows the validation set performance of the three model architectures. The 2DCRNN and the 1DCNN have similar validation set performance, while the 2DCNN outperforms both other architectures. These results confirm the hypothesis that the 2DCNN is best equipped for sUAS ranging. Thus, the 2DCNN is used in Experiment Two. The results of the 2DCRNN also confirm that the dimensionality of the input data does not effectively allow the 2DCRNN to learn time-based dependencies in the Mel-spectrogram. Other researchers that have developed 2DCRNNs for acoustics deep-learning have input shapes that are much wider in the time domain (128x128) [18, 23]; thus, there is much more information available in the time-domain axis of the Mel-spectrogram. Unfortunately, the acoustics data used in our experiment has an 8KHz sampling rate and does not have the time-domain resolution available to researchers with high-fidelity audio. Additionally, the dimensionality of a 4000x1 audio frame inherently presents challenges that make machine learning much more challenging than a concise representation of audio data. From Dai's paper that developed the 1DCNN for raw audio, the researchers concluded that, at best, the 1DCNN performed similarly to a 2DCNN on sound event classification tasks [20].

Experiment Two evaluates how effectively the 2DCNN generalizes sUAS range across various sUAS models through three different comparisons. The first result of Experiment Two compares the 2DCNN trained with the IF dataset and the 2DCNN trained with Combined on the IF testset. Table 7 presents the Balanced Accuracy and F1 scores of the two models' performances. The 2DCNN trained on Combined improves IF range accuracy across all four classes. Additionally, massive mispredictions with 2DCNN (Combined dataset) are reduced (e.g., predicting 80m+, but the sUAS is within 40m). It achieves an F1 score within 40m of 0.90 and a balanced accuracy of 0.824, which is beyond the hypothesized success thresholds.

The second result compares the 2DCNN trained with the Matrice dataset and the 2DCNN trained with Combined on Matrice testset. Table 8 presents the balanced accuracy and F1 scores of the two models' performances. The 2DCNN trained with Combined outperforms the classification capability of the model trained with the Matrice dataset. It achieves an F1 score within 40m of 0.91 and a balanced accuracy of 0.793, which exceeds the performance thresholds.

The third result compares the 2DCNN trained with the Phantom

Table 3. 2DCNN Architecture

2DCNN Architecture	Size	Activation	Strides
Input	128x8x1	-	-
Batch Norm	-	-	-
2DConv	8@4x4	BN/Relu	-
Max Pooling	2x2	-	-
2DConv	16@2x2	BN/Relu	2x2
Dropout	0.3	-	-
2DConv	32@2x2	BN/Relu	2x2
Dropout	0.3	-	-
Flatten	-	-	-
Dense	512	Relu	-
Dense	256	Relu	-
Dropout	0.5	-	-
Output	4	Softmax	-

Table 4. 2DCRNN Architecture

2DCRNN Architecture	Size	Activation	Strides
Input	128x8x1	-	-
Batch Norm	-	-	-
2DConv	8@2x2	BN/Relu	1x1
2DConv	16@4x4	BN/Relu	3x1
Dropout	0.3	-	-
2DConv	32@4x4	BN/Relu	4x1
Max Pooling	12x1	-	
Dropout	0.3	-	-
Reshape	-	-	-
GRU	32	-	-
Dense	64	Relu	-
Dropout	0.5	-	-
Output	4	Softmax	-

Table 5. 1DCNN Architecture

1DCNN Architecture	Size	Activation	Strides
Input	4000x1	-	-
1DConv	64@80	BN/Relu	4
Max Pooling	4	-	-
1DConv	64@3	BN/Relu	2
Max Pooling	4	-	-
Dropout	0.3	-	-
1DConv	128@3	BN/Relu	-
Max Pooling	4	-	-
1DConv	256@3	BN/Relu	-
Max Pooling	2	-	-
Reshape	-	-	-
GAP	1	-	-
Output	4	Softmax	-

Table 6. Model Architecture Validation Set Performance

Architecture	Input Type	Macro F1-Score
2DCNN	Mel-Spect	0.72
2DCRNN	Mel-Spect	0.63
1DCNN	Raw Audio	0.58
Naive	-	0.13

Table 7. IF Testset Performance

2DCNN Trained w/	IF	Combined
<40m F1	0.89	0.90
40-60m F1	0.69	0.79
60-80m F1	0.65	0.73
80m+ F1	0.82	0.84
Balanced Accuracy	0.769	0.824

dataset and the 2DCNN trained with Combined on the Phantom testset. The Balanced Accuracy and F1 scores of the two models' performances are in Table 9. The model does not achieve the performance goals with an F1 score of less than 0.80 within 40m and a balanced accuracy score of less than 0.70. This performance degradation is likely caused by the design of the DJI Phantom. The Phantom is a small sUAS with low rotor power that yields a smaller acoustic footprint than the other sUAS have.

Although the 2DCNN trained using Combined only confirms the hypothesis on two of three sUAS models, there are important takeaways from the testset performances regarding the generalizability of sUAS range estimation and the usefulness of deep-learning-based sUAS tracking. The first is that sUAS ranging is generalizable across different sUAS types. The performance increase from training the model with Combined versus a single sUAS implies that the 2DCNN learns sUAS type-agnostic features in the convolution layers that improve ranging performance across all sUAS types. This concept challenges the notion of how humans perceive sound and further demonstrates the power of using deep-learning to recognize patterns that are not easily recognizable by human perception. These

Table 8. Matrice Testset Performance

2DCNN Trained w/	Matrice	Combined
<40m F1	0.89	0.91
40-60m F1	0.60	0.64
60-80m F1	0.73	0.75
80m+ F1	0.81	0.84
Balanced Accuracy	0.768	0.793

Table 9. Phantom Testset Performance

2DCNN Trained w/	Phantom	Combined
<40m F1	0.73	0.72
40-60m F1	0.57	0.60
60-80m F1	0.46	0.52
80m+ F1	0.75	0.75
Balanced Accuracy	0.626	0.647

results demonstrate that deep-learning is an effective method to localize sUAS with cellphones when presented with low-fidelity data and a sub-optimal data collection environment. These results imply that if given a large constellation of cellphones, an sUAS range estimation model distributed across all devices effectively distinguishes which devices are close (within 40m), moderately close (40-60m), moderately far (60-80m), and far (farther than 80m) of the sUAS. Combining the results, the sUAS can be effectively tracked within the constellation of cellphones. These methods provide an effective sUAS defense strategy that is de-burdened from relying exclusively on expensive sensing methods.

5. CONCLUSION

The threat of sUAS by state and non-state actors demands sUAS countermeasures with equally accessible defense resources. To meet this need, the research effort develops a deep-learning-based sUAS ranging method that can be used for sUAS tracking within a constellation of low-cost sensing devices. First, three different model architectures are trained and the best-performing model, the 2DCNN, is selected for further testing. Testsets from each of the three individual sUAS dataset evaluate the performance of the 2DCNN trained using Combined and the 2DCNN trained on individual sUAS datasets. The results from this test demonstrate that additional data from various sUAS help the model better achieve sUAS ranging on each sUAS model. The balanced accuracies and F1 scores within 40m for both the IF and Matrice sUAS models are above 0.70 and 0.80, respectively; however, the balanced accuracy and F1 score for the Phantom sUAS are not above either threshold. Despite this, these results demonstrate that a deep-learning-based approach to cellphone-based sUAS ranging is attainable and can be employed to track and detect sUAS without the burden of expensive, immobile sensing resources. In future efforts, the effects of background noise on classification performance will be analyzed. Additionally, more complex model architectures will be explored.

6. REFERENCES

[1] Travis M. Andrews, "A commercial airplane collided with a drone in canada, a first in north america," Oct 2017.

[2] Chris Vallance, "Ukraine sent dozens of 'dronations' to build army of drones," Jul 2022.

[3] Alexander Sedunov, Darren Haddad, Hady Salloum, Alexander Sutin, Nikolay Sedunov, and Alexander Yakubovskiy, "Stevens drone detection acoustic system and experiments in acoustics uav tracking," 2019.

[4] Harini Kolamunna, Thilini Dahanayaka, Junye Li, Suranga Seneviratne, Kanchana Thilakaratne, Albert Y. Zomaya, and Aruna Seneviratne, "Droneprint: Acoustic signatures for open-set drone detection and identification with online data," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 5, 3 2021.

[5] Xiufang Shi, Guoqiang Mao, Brian D. O. Anderson, Zaiyue Yang, and Jiming Chen, "Robust localization using range measurements with unknown and bounded errors," *Trans. Wireless. Comm.*, vol. 16, no. 6, pp. 4065–4078, jun 2017.

[6] Xianyu Chang, Chaoqun Yang, Junfeng Wu, Xiufang Shi, and Zhiguo Shi, "A surveillance system for drone localization and tracking using acoustic arrays," 8 2018, vol. 2018-July, pp. 573–577, IEEE Computer Society.

[7] Zewen Wang, Dexiu Hu, Yongjun Zhao, Zhaocheng Hu, and Zhixin Liu, "Real-time passive localization of tdoa via neural networks," *IEEE Communications Letters*, vol. 25, pp. 3320–3324, 10 2021.

[8] Alexandros Kyritsis, Rodoula Makri, and Nikolaos Uzunoglu, "Small uas online audio doa estimation and real-time identification using machine learning," *Sensors*, vol. 22, no. 22, 2022.

[9] Darren Haddad Kalianppna Gopalan, Brett Smolenski, "Acoustic detection of drone range using band energy features," 4 2022.

[10] Darren Haddad Kalianppna Gopalan, Brett Smolenski, "Detection and classification of drones using fourier-bessel series representation of acoustic emissions," 7 2022.

[11] Darren Haddad Matthew Tan, Brett Smolenski, "Real-time acoustic detection and identification of drones in operational conditions," 1 2022.

[12] Milton Isla, Anthony Christe, and Tyler Yoshiyama, "Redvox-python-sdk," 2022.

[13] Inspired Flight, "If1200," 2021.

[14] DJI, "Matrice 600," 2017.

[15] DJI, "Phantom 4 pro v2," 2022.

[16] Yoojeong Seo, Beomhui Jang, and Sungbin Im, "Drone detection using convolutional neural networks with acoustic stft features," 2018.

[17] Sara Al-Emadi, Abdulla Al-Ali, and Abdulaziz Al-Ali, "Audio-based drone detection and identification using deep learning techniques with dataset enhancement through generative adversarial networks," *Sensors*, vol. 21, 8 2021.

[18] Mariam Yiwere and Eun Joo Rhee, "Sound source distance estimation using deep learning: An image classification approach," *Sensors (Switzerland)*, vol. 20, 1 2020.

[19] Sharath Adavanne, Archontis Politis, Joonas Nikunen, and Tuomas Virtanen, "Sound event localization and detection of overlapping sources using convolutional recurrent neural networks," 6 2018.

[20] Wei Dai, Chia Dai, Shuhui Qu, Juncheng Li, and Samarjit Das, "Very deep convolutional neural networks for raw waveforms," 10 2016.

[21] Yagya Raj Pandeya, Dongwhoon Kim, and Joonwhoan Lee, "Domestic cat sound classification using learned features from deep neural nets," *Applied Sciences*, vol. 8, no. 10, 2018.

[22] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, 2015, vol. 8.

[23] Sharath Adavanne, Archontis Politis, Joonas Nikunen, and Tuomas Virtanen, "Sound event localization and detection of overlapping sources using convolutional recurrent neural networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 1, pp. 34–48, 2019.

IV. Conclusions

In this thesis, two efforts use accessible equipment (i.e., cellphone microphones) to track and detect Small Unmanned Aircraft Systems (sUAS). The first research effort evaluates the benefits of aggregating sUAS detection predictions from a constellation of cellphones. Using an Ensemble Voting Pipeline (EVP), persistent sUAS detection is achieved throughout a noisy, fault-ridden Wireless Acoustic Sensor Network (WASN).

The second effort uses a deep learning model to predict sUAS range using cellphone audio data formatted into Mel-Spectrograms. The effort determines that a deep learning model can predict the sUAS range (across four range classes) with a macro-F1 score of 0.7492. Additionally, the effort demonstrates that a deep learning architecture trained with various sUAS models can extract sUAS-agnostic features in the convolutional layers that enhance predictions on each sUAS.

Combined, these concepts help to enhance the Joint Force's ability to detect sUAS threats with inexpensive, accessible sensing equipment. As warfare continues to evolve, the rapidly developing threats of sUAS require awareness solutions that are just as agile, replaceable, and effective. Although other sUAS sensing methods exist, the efforts in this thesis evaluate sensing methods that complement the US's more "exquisite systems on stand-off platforms" [11]. Redundant and mobile sensor systems, such as these, can accelerate the Joint Force towards a future that uses low-cost edge computing and autonomous technology to keep situational awareness in adverse environments.

4.1 Future Work

In future work, we desire to investigate the value of inexpensive acoustic sensing equipment in the sUAS detection, tracking, and classification domain. The work

in this thesis demonstrates proof-of-concept methods that, if enhanced, could be employable in the real world to defend against sUAS threats. Additionally, the EVP developed in this thesis can be trained to detect and recognize more threats than just sUAS, allowing a sensor constellation to provide resilient battle-space awareness for air and ground targets (i.e., troop movement, gunshots, and fixed-wing aircraft).

In future efforts, the research accomplished in this thesis will be enhanced in several ways.

- Conduct a new data collection with standardized omnidirectional microphones on-board devices (e.g., Raspberry Pis with GPS and microphone hats or cell-phones with a higher sampling rate). This sensor standardization and re-collection of data will allow access to higher-fidelity audio data and a larger variety of different sUAS flights, improving Machine learning (ML) model performance while remaining a low-cost sUAS defense method.

- Investigate the ability of a deep learning model to recognize other sUAS characteristics: azimuth and elevation angle in relation to the sensor constellation, type of sUAS, or approximate weight of the payload that an sUAS carries. This effort will improve the specificity of deep-learning-based sUAS localization and flight characteristics.

- Enhance sUAS detection performance by fusing multi-modal sensors (such as imagery, radar, or radio frequency sensors). This will improve the resiliency of predictions in non-optimal sensing environments, as each sensing modality has inherent strengths that complement one another.

- Integrate the EVP from Effort 1 into a real-world WASN system to enable real-time sUAS recognition across a large area. This additional effort will turn the proof-of-concept EVP into a system that the Joint Force can use to further our

strategic sUAS and sensor objectives.

Appendix A. sUAS Cluster Estimation

In the main body of work, sUAS tracking is limited to a cellphone-agnostic perspective of estimating sUAS range from a cellphone. However, we now present an additional perspective on sUAS tracking, which uses a deep learning-based early data fusion approach to estimate which portion of the cellphone constellation is closest to the sUAS. We develop a 2DConvolutional Neural Network (CNN) that takes the entire constellation's cellphone audio as the input and predicts which cellphone cluster is closest to the sUAS at a given data frame. This method is effective in achieving accurate sUAS location prediction but requires strict assumptions that must be met to train a neural network. First, the cellphone cluster positions must remain the same (i.e., cellphones must remain within their cluster). Additionally, the flight path of the sUAS must cross all three cellphone clusters to have valuable data for model training.

The data used for this research effort comes from the Escape II Data Collection. All three scenarios, *H*, *SF*, and *LF*, are used because the cellphone cluster membership remains constant across all three scenarios.

The truth labeling for the neural network is determined by calculating the average range to the sUAS from all cellphones within each of the three cellphone clusters. The range from each cellphone to the sUAS is already known due to work conducted in "sUAS Ranging: A Deep Learning Approach," so calculating which portion of the constellation the sUAS is a trivial task. The target values for the network take on three possible values, each mutually exclusive, representing the cellphone cluster that the sUAS is closest to. Figure 3 shows a visualization of the class boundaries, which depicts the class separations between the clusters with blue lines. Additionally, Figure 4 depicts the class distribution of the three clusters. The data distribution is unbalanced because the *H* scenario takes place on top of cluster class two, and the middle cluster, cluster 1, is only the closest to an sUAS when the sUAS is in transit

over top of cluster 1. In training and score reporting, the class unbalance is accounted for by adding class weight penalties to the neural network training and reporting all scores in distribution-agnostic measures (e.g., balanced accuracy).

The data is formatted in a 28-channel deep, Mel-Spectrogram format, with 0.5s frames (4000 samples) that have a hop length of 512 samples, Fast Fourier Transform (FFT) length of 2048 samples, and overall dimensionality of 28x128x8. Additionally, the three clusters are grouped within the input data; thus, all cluster 1 cellphones are in channels 0-7, cluster 2 cellphones are in channels 8-15, and cluster 3 cellphones are in channels 16-27. This data is then fed into a 2-dimensional CNN.

The CNN architecture is created to demonstrate that an early data fusion approach to sUAS ranging is possible; thus, the model is not tuned and evaluated to the same levels of rigor as the primary research efforts. However, it follows a similar structure to the other neural networks built throughout this research effort. Table 2.

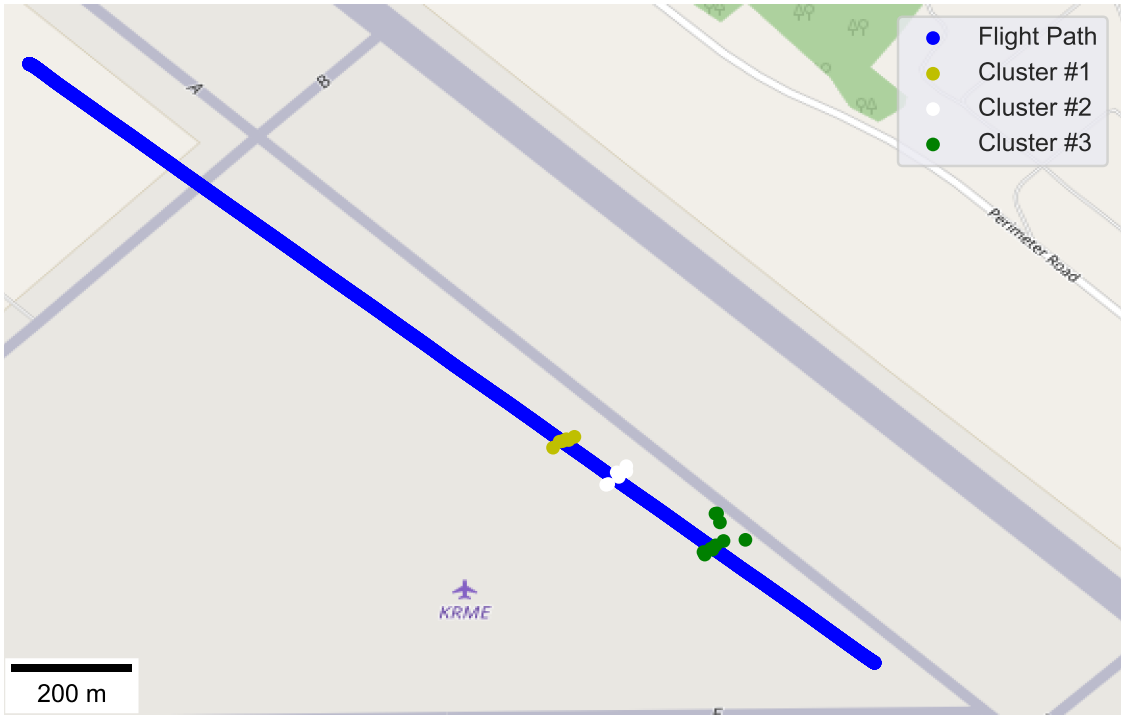

Figure 3: Early Data Fusion Approach Class Separation

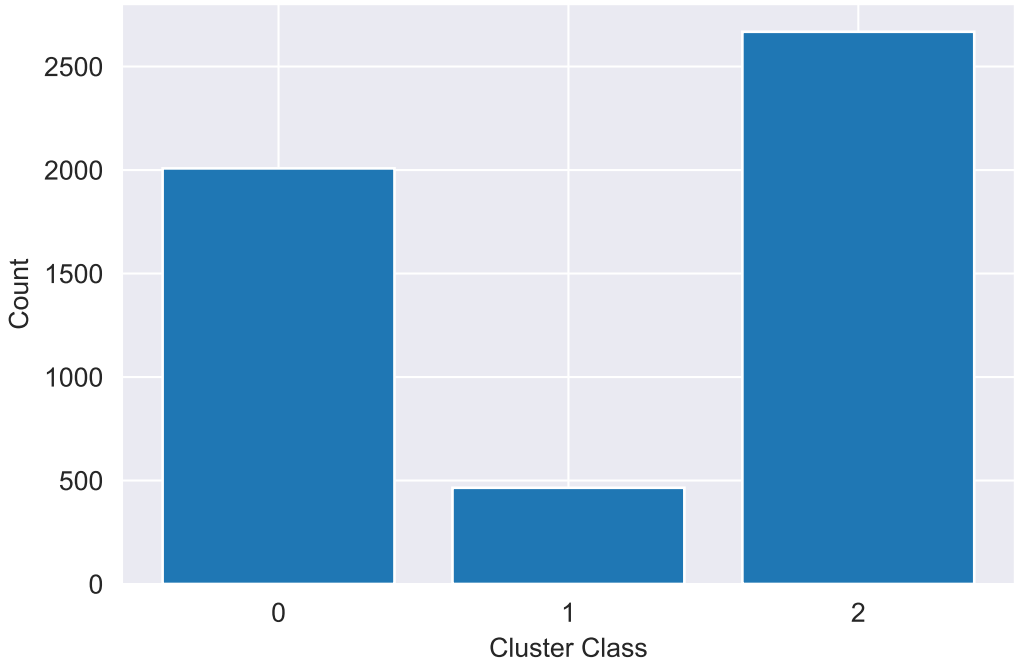

Figure 4: Early Data Fusion Class Distribution

displays the model architecture.

The model is tested using a 20% test-set split that is partitioned before model training. After training the model, the testset evaluation yields 96.29% accuracy across the three cluster classes and a balanced accuracy score of 94.39%. Figure 5 displays the confusion matrix of the predictions. These results demonstrate that the model can determine where an sUAS is in relation to the three distinct cellphone clusters. Although not as precise as methods such as Time Difference of Arrival (TDoA), the model accurately determines sUAS location relative to the constellation of devices.

Table 2: Early Data Fusion Neural Network

2DCNN Architecture	Size	Activation	Strides
Input	28x128x8x1	-	-
Batch Norm	-	-	-
2DConv	8@3x3	BN/Leaky Relu	-
2DConv	16@3x3	BN/Leaky Relu	2x2
2DConv	32@3x3	BN/Leaky Relu	2x2
Dropout	0.3	-	-
Flatten	-	-	-
Dense	256	Leaky Relu	-
Dense	128	Leaky Relu	-
Dropout	0.5	-	-
Output	3	Softmax	-

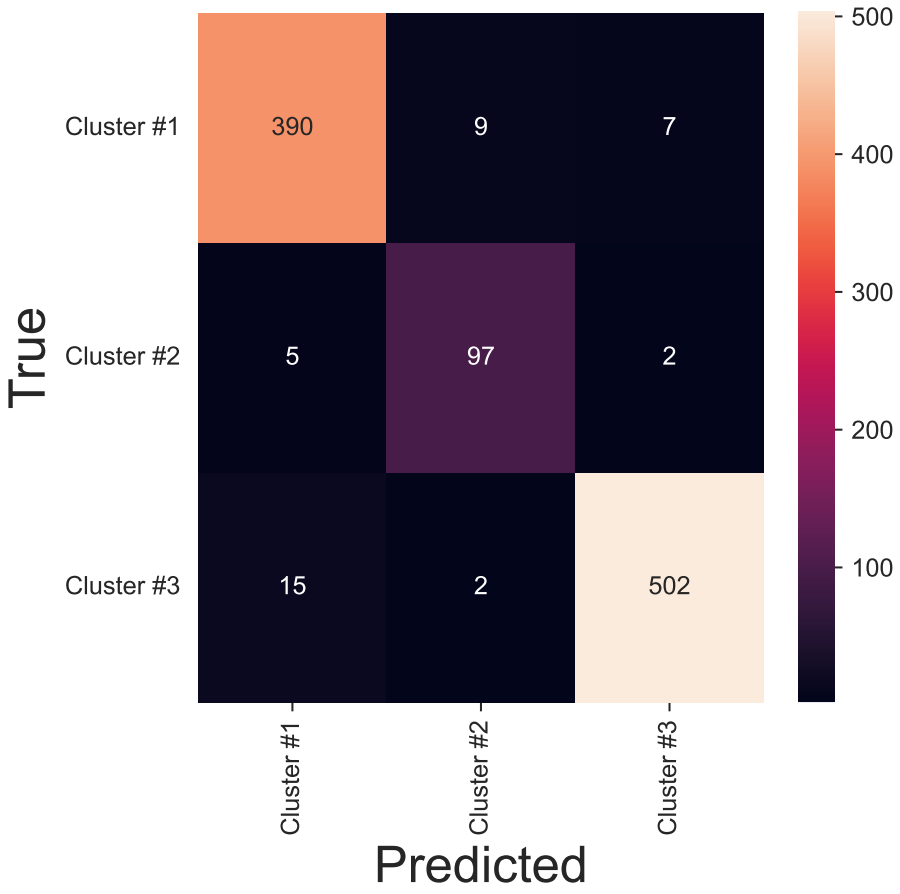

Figure 5: Early Data Fusion Confusion Matrix

Appendix B. sUAS Range Estimation: A Regression Approach

In Chapter III, sUAS range estimation is split into four range classes. However, prior to shifting to a multi-class classification approach for sUAS range estimation, we first designed a regression output layer for the CNN, as this seemed like a valid method for training the network.

The ML model design and the data used to train the regression model are a modified version of the Combined dataset from "Cellphone-Based sUAS Range Estimation: A Deep Learning Approach." The dataset only contains ranges up to 100m as distance estimations become ambiguous when an sUAS is not detectable. Additionally, a 20%-split is taken from the dataset to be used as a testset. Although class imbalance is an issue for training the network when presented as a multi-class classification model, it is even more apparent in the continuous domain. Figure 6 demonstrates the dataset data distribution which is (roughly) a three-peaked normal distribution. The non-uniformity of the data causes severe issues that result in the machine learning performance degradation of the regression model. These issues are now highlighted via an ML model comparison between a regression 2-Dimensional Convolutional Neural Network (2DCNN) and the multi-class classification 2DCNN used in "Cellphone-Based sUAS Range Estimation: A Deep Learning Approach."

The regression 2DCNN is trained using the modified Combined dataset and a loss function of Mean Squared Error (MSE). After training, the regression model is evaluated using the testset. These continuous valued predictions are then converted into pseudo-range classification bins to compare the regression model performance to the 4-class classification model performance.

Figures 7 and 8 demonstrate that the regression model performs significantly worse than the multi-class classification model. This is likely because to minimize loss, the

regression model is driven to a local minimum that makes conservative predictions (i.e., class 2 or class 3). The training data has a distribution with an average value of around 60m, and the network is driven to make conservative predictions to minimize MSE. If presented with a more even distribution of training data and high-quality audio, a regression network would likely outperform the capabilities of the multi-class classification network. However, this exploration demonstrates that due to the dataset constraints, a regression model is insufficient for estimating sUAS range.

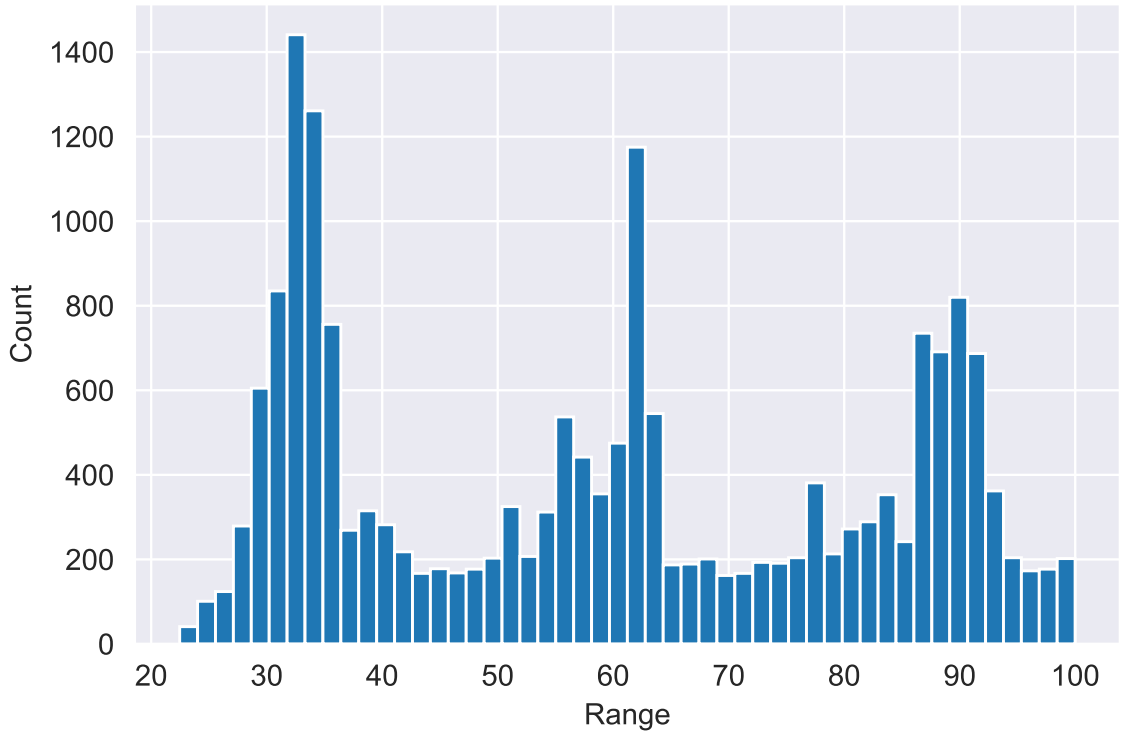

Figure 6: Combined Dataset Distribution

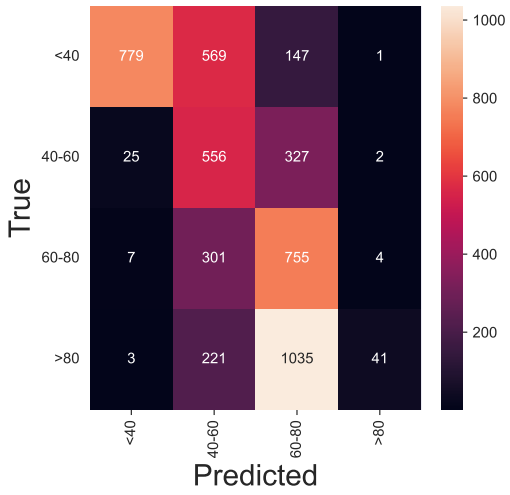

Figure 7: 2DCNN (Regression) Confusion Matrix

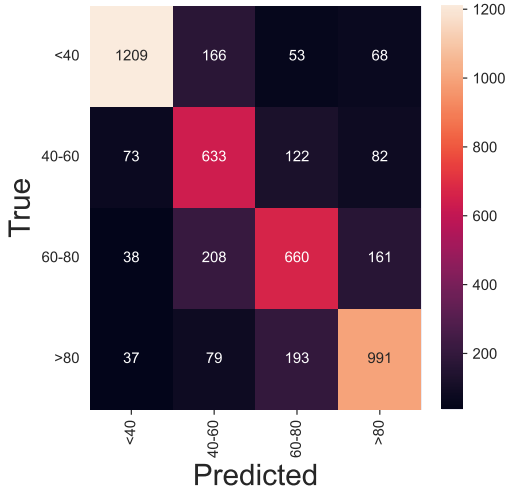

Figure 8: 2DCNN (Classification) Confusion Matrix

Bibliography

1. *Counter-Small Unmanned Aircraft Systems Strategy*, US Department of Defense, Washington, DC: USA, 2020. [Online]. Available: https://media.defense.gov/2021/Jan/07/2002561080/-1/-1/1/department-of-defense-couter-small-unmanned-aircraft-systems-strategy.pdf

2. J. Alkobi, "The evolution of drones: From military to hobby & commercial," Feb 2022. [Online]. Available: https://percepto.co/the-evolution-of-drones-from-military-to-hobby-commercial/#:~:text=Informed%20by%20military%20research%20and,its%20first%20commercial%20drone%20permit.

3. Z. J. Miller, "Drone that crashed at white house was quadcopter," Jan 2015. [Online]. Available: https://time.com/3682307/white-house-drone-crash/

4. T. M. Andrews, "A commercial airplane collided with a drone in canada, a first in north america," Oct 2017. [Online]. Available: https://www.washingtonpost.com/news/morning-mix/wp/2017/10/16/a-commercial-airplane-collided-with-a-drone-in-canada-a-first-in-north-america/

5. K. Allen, "Drones and violent nonstate actors in africa," Aug 2021. [Online]. Available: https://africacenter.org/spotlight/drones-and-violent-nonstate-actors-in-africa/

6. C. Vallance, "Ukraine sent dozens of 'dronations' to build army of drones," Jul 2022. [Online]. Available: https://www.bbc.com/news/technology-62048403

7. A. Messinis, "Iraq-conflict-mosul," 2017. [Online]. Available: https://www.gettyimages.com/detail/news-photo/picture-taken-on-march-14-2017-in-the-northern-iraqi-city-news-photo/653711132?adppopup=true

8. "Coyote uas — raytheon missiles & defense," 2022. [Online]. Available: https://www.raytheonmissilesanddefense.com/what-we-do/counter-uas/effectors/coyote

9. *Department of Defense Counter-Unmanned Aircraft Systems*, Congressional Research Service, Washington, DC: USA, 2022. [Online]. Available: https://sgp.fas.org/crs/weapons/IF11426.pdf

10. R. Daniel, "Testing a drone," 2018. [Online]. Available: https://www.defense.gov/Multimedia/Photos/igphoto/2001896713/

11. *Air Force Science and Technology Strategy*, Department of Defense, Washington, DC: USA, 2019. [Online]. Available: https://www.af.mil/Portals/1/documents/2019%20SAF%20story%20attachments/Air%20Force%20Science%20and%20Technology%20Strategy.pdf

12. M. Isla, A. Christe, and T. Yoshiyama, "Redvox-python-sdk," 2022. [Online]. Available: https://github.com/RedVoxInc/redvox-python-sdk

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
20–01–2023	Master's Thesis	Sept 2021 — Jan 2023

4. TITLE AND SUBTITLE

Cellphone-Acoustics Based sUAS Detection and Tracking

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Ryan D. Clendening

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENG-MS-23-M-017

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AFRL
WPAFB OH 45433-7765
DSN 312-785-0066, COMM 937-255-0066
Email: dominic.dumbra.1.ctr@us.af.mil

10. SPONSOR/MONITOR'S ACRONYM(S)

AFRL/RY

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Small Unmanned Aerial Systems (sUAS) are an easily accessible technology that has become an increasingly large threat to US critical systems. This threatening technology demands using fault-tolerant, low-cost, replaceable, and accurate sensing resources, which counter the ubiquitous nature of sUAS [1]. Therefore, the methods developed in this thesis detect and track sUAS using easily accessible sensing resources, such as cellphones. First, we develop an acoustics sensor network-based sUAS detection methodology. In the latter effort, a deep learning model is trained using the acoustics data from the data collection to predict sUAS range from a cellphone. Combined, these two efforts demonstrate the merits of using accessible sensing resources to achieve highly accurate sUAS detection and tracking results.

15. SUBJECT TERMS

artificial neural network (ANN), convolutional neural network (CNN), deep learning, information fusion, wireless sensor networks, small unmanned aerial systems, internet of things

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Maj Richard Dill, AFIT/ENG
U	U	U	UU	58	19b. TELEPHONE NUMBER *(include area code)* (937) 255-3636, ext 3652; richard.dill@afit.edu