DroneScale: Drone Load Estimation Via Remote Passive RF Sensing

Phuc Nguyen[†‡], Vimal Kakaraparthi[‡], Nam Bui[‡], Nikshep Umamahesh[‡], Nhat Pham[§‡]
Hoang Truong[‡], Yeswanth Guddeti[★], Dinesh Bharadia[★], Richard Han[‡]
Eric Frew[‡], Daniel Massey[‡], and Tam Vu[§‡]
vp.nguyen@uta.edu;[firstname.lastname]@colorado.edu
[firstname.lastname]@cs.ox.ac.uk;[yguddeti,dineshb]@eng.ucsd.edu
[†]University of Texas at Arlington, [‡]University of Colorado Boulder,
[§]University of Oxford, [★]University of California San Diego

ABSTRACT

Drones have carried weapons, drugs, explosives and illegal packages in the recent past, raising strong concerns from public authorities. While existing drone monitoring systems only focus on detecting drone presence, localizing or fingerprinting the drone, there is a lack of a solution for estimating the additional load carried by a drone. In this paper, we present a novel passive RF system, namely DroneScale, to monitor the wireless signals transmitted by commercial drones and then confirm their models and loads. Our key technical contribution is a proposed technique to passively capture vibration at high resolution (i.e., 1Hz vibration) from afar, which was not possible before. We prototype DroneScale using COTS RF components and illustrate that it can monitor the body vibration of a drone at the targeted resolution. In addition, we develop learning algorithms to extract the physical vibration of the drone from the transmitted signal to infer the model of a drone and the load carried by it. We evaluate the DroneScale system using 5 different drone models, which carry external loads of up to 400g. The experimental results show that the system is able to estimate the external load of a drone with an average accuracy of 96.27%. We also analyze the sensitivity of the system with different load placements with respect to the drone's body, flight modes, and distances up to 200 meters.

CCS CONCEPTS

• **Computer systems organization** → **Sensors and actuators**.

KEYWORDS

Drone security, drone load estimation, RF sensing systems

ACM Reference Format:
Phuc Nguyen[†‡], Vimal Kakaraparthi[‡], Nam Bui[‡], Nikshep Umamahesh[‡], Nhat Pham[§‡], Hoang Truong[‡], Yeswanth Guddeti[★], Dinesh Bharadia[★], Richard Han[‡], and Eric Frew[‡], Daniel Massey[‡], and Tam Vu[§‡]. 2020. DroneScale: Drone Load Estimation Via Remote Passive RF Sensing. In *The 18th ACM*

Figure 1: DroneScale system analyzes the RF signal transmitted from a drone and estimate its model and load.

1 INTRODUCTION

There has been a significant rise in drone-related mishaps in places such as airports, residential areas, schools, prisons, borders etc. in recent years [25]. The intentions behind these incidents can range anywhere from naively benign to extremely fatal. For example, Gatwick Airport of England was shutdown for more than 24 hours in late 2018, due to a drone incursion, disrupting hundreds of flights across the world [78]. An unregistered drone flew over a packed stadium during a football match causing a panic among security officials [1].More serious drone incidents involve them delivering drugs [3, 84],weapons [33], explosives [79, 85], illegal packages [23, 57, 81] across political borders, prisons etc.Even if a drone is registered with a unique MAC address, there exist many software solutions to modify the address.U.S. Customs and Border Protection has reported as many as 170 narcotic delivery attempts using drones in the past five years [2]. The Border Patrol would benefit from knowing whether a drone is just being used for observation or is being used to smuggle goods. Similarly, there has been a spike in reported incidents of delivery of phones, tobacco, drugs, and weapons to prisons [3, 23, 57, 84]. Staff at such facilities would again benefit by knowing whether the drone is being used for the delivery of contraband.

Current drone monitoring systems focus primarily on drone detection, e.g. Dedrone [24], DroneDetector [30], ARDRONIS [8],

DroneShield [31], Falcon Shield [34], AUDS [9], ADS-ZJU [75] and Matthan [59], localization [58], and fingerprinting [67]. DroneScale differs from these works by introducing the concept of automated drone load estimation, providing the capability to remotely "weigh" the drone as if on a *scale.* Prior work has manifested that the drone's transmitted RF signal contains features that indicate the presence of a drone [59], wherein it was found that for example the drone's propellers generate a vibration that can be detected in the RF signal. This intuition linking propellor dynamics with the RF signal inspired our work on drone load estimation. Matthan [59] and similar software-defined radio-based systems can only detect drone presence, their hardware and software architectures are not sufficient to precisely track the modulation in frequency of drone's communication caused by external load. More importantly, DroneScale uses frequency-based analysis to detect loads so it is more resilient to noises compared to temporal-based analysis as in Matthan system.

In this paper, we present DroneScale, the first passive RF system that is able to passively estimate external load carried by commercial-unmodified drones (unmodified structure, propellers, camera, and battery). DroneScale monitors the RF communication signal transmitted by a drone and uses it to extract its physical behaviors to infer the model of the drone as well as its carried load. In particular, DroneScale first passively listens to the frequency band of interest and detects active frequencies to extract the drone's body vibration profile. Next, it monitors the RF frequency variation of the drone's signal caused by the drone's body vibrations at the 1Hz resolution level. This is achieved using the relationship between the drone's body vibration and the corresponding thrust force required to carry an external load. Lastly, it estimates the drone's external load with 100g resolution based on the drone's vibration profile with its current system's performance. DroneScale's concept is illustrated in Fig. 1. However, realizing DroneScale is a formidable task due to the following challenges:

- External load detection of a drone is an unexplored problem. It is unclear what the unique signatures are, in order to detect the load carried by a drone.
- The vibration of a drone creates millimeter amplitude vibrations which generate a few Hz of RF frequency shifting of the drone signal. These signals are buried under the noise level due to the high phase noise and flicker noise of existing RF architectures.
- Existing RF architectures, due to baseband downconversion approach and ADC architecture, are not designed to monitor signals at low frequencies at which drone signals appear (<1 kHz).
- The monitored signals include noise from the environment. Separating them is a challenging task that requires thoughtful design algorithms to make the system perform reliably.
- It is unclear how parameters such as distance, load placement, flight modes, etc. are going to affect the accuracy of DroneScale.

Contributions. We make the following contributions. (1) We identify the relationship between the specifications of a drone and its body vibration profile. (2) We identify the model of a drone and the load carried by it based on its vibration signature. (3) We develop a new RF hardware technique allowing the system to automatically identify the best downconversion frequency to obtain the highest sensitivity at the frequency range of interest. DroneScale is able to monitor the RF shifting with 1Hz resolution. (4) We devise an
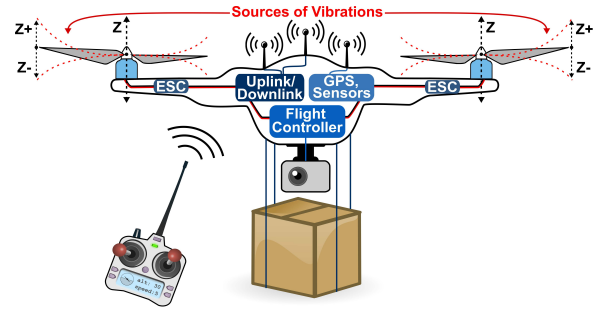

Figure 2: Typical commercial drone control architecture and the drone vibration sources

algorithm to characterize the drone by estimating its model and external load. (5) We verify that the proposed system can detect the drone load with 96.27% of accuracy. (6) We prove that the proposed system could be used to estimate the load carried by known drones at up to 200m distance.

2 LOAD VS. VIBRATION RELATIONSHIP

Before discussing the DroneScale system, we examine key concepts underlying a multi-rotor drone's vibration and observations on the impact of external load to the drone's vibration profile. This section aims to answer the following questions: (1) *Why are multi-rotor drones the only type of drones discussed herein?*, (2) *Why does a drone vibrate and what are the components that make it vibrate?*, (3) *Do different drones vibrate differently?*, (4) *Does carrying more load change the vibration profile of a drone?*, (5) *What are the potential approaches to monitor the load carried by a drone from afar?*

■ **Multi-rotor drones are so common**. Drones can be made in different forms including helicopter, airplane, multi-rotor or even a balloon. In that, helicopter and multi-rotor forms are the most common forms of drones due to their simplicity of manufacturing. A multi-rotor drone [53] has multiple rotors with a much simpler flying control mechanism compared to that of a helicopter [15, 16, 41]. Instead of changing its wing's pitch and speed using a complex rotor, as found in helicopters, to maintain balance and maneuver, a multi-rotor operates by simply changing its motors' speed, eliminating the need for any complex mechanical parts. Since it has multiple similar rotors arranged symmetrically, a multi-rotor can keep its balance more easily even when it carries additional load (e.g. camera, packages). Therefore, most of today's commercial drones are of the multi-rotor type and the same trend is predicted for the near future. In this paper, we focus on estimating the external load carried by multirotor UAVs.

■ **Typical drone components and vibration sources.** The control architecture of a commercial drone consists of three main components: (1) Flight Controller (FC) block, (2) RF communication (RC) block, (3) Media Streaming block as illustrated in Fig. 2. While media streaming can be optional, FC and RC blocks are required for the drone to take off and perform maneuvers. FC includes numerous sensors like vision, ultrasound, IR, Lidar etc. sensors, depending on the sophistication of the drone, but the crucial ones for smooth flight are the Inertial Measurement Units (IMUs), gyroscopes, accelerometers, and GPS, which ensure the stability of a drone during flight. FC has sophisticated software to seamlessly make the sensors work together. The Drone Propulsion block consists of motor
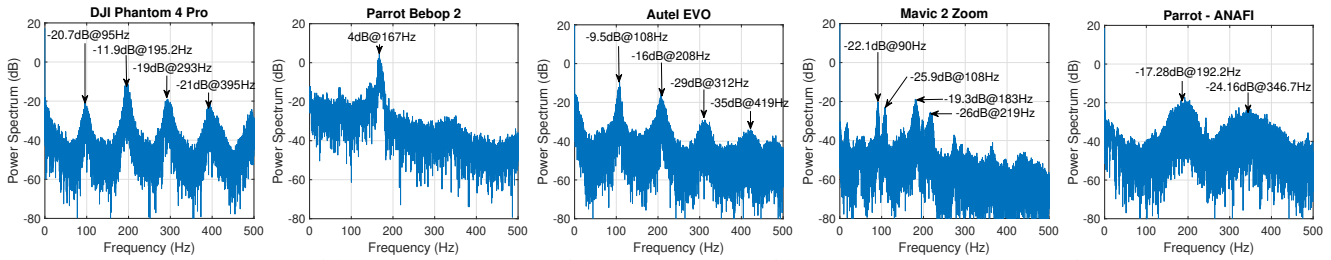
Figure 3: Vibration profile of (a) DJI Phantom Pro 4, (b) Parrot Bebop 2, (c) Autel Robotics EVO, and (e) DJI Mavic 2 Zoom, and (d) Parrot - ANAFI 4K Quadcopter captured by IMU sensor

controllers and Electronic Signal Controllers (ESCs) which control reversing, braking, speed and acceleration of the motors. A typical drone motor uses current control, through field-oriented control of ESCs, to control the torque of 3-phase stepper motors with high accuracy and bandwidth [35]. A motor controller is equipped with a microprocessor that communicates with the RC block and FC to control the ESC, and thereby the motor, appropriately. Drone body motion can be classified into two classes: body shifting and body vibration [59]. Body shifting is caused by external factors like controller signal and subsequent stabilization action of the FC block. These vibrations tend to be in the range of few 10s of Hz. The second type of drone body motion is body vibration. The body of a drone is vibrating at a frequency in the range of a 100s Hz [59], which is caused by additional forces created when the propeller moves back and forth on the z axis as shown in Fig. 2. The propeller creates both axial and radial vibrations along the drone's arms whose effects are manifested on the central plate of the drone body and other critical locations, where most of the drone's electronic components are placed [82]. Lastly, the RC block consists of a standard r/c radio receiver to receive pilot control signals through a minimum of four control channels. As the RC block is often placed close to the central plate of the drone, with its transceiver antennas placed at the center of the drone or drone's legs, the drone's body vibration affects the communication channel of the drone and generates distinguishable signatures that can be used to monitor the drone's physical behavior (See Sec. 3).

■ **Different models of drones vibrate differently.** Different drones have different architecture (propeller size and shape, material, length from propeller to central mass etc.) and weight that require the drone propellers to have different angular velocities to generate sufficient thrust force for the drone to perform its maneuvers (See. Sec. 4). To validate this hypothesis, we conducted an experiment where a high quality IMU sensor (MicroStrain LORD 2DM-GX5-25 [51]) is placed at the center of mass of the drones' bodies. The experiment is conducted on 5 common commercial drones in the market including DJI Phantom 4 Pro [29], Parrot Bebop 2 [61], Parrot - ANAFI 4K Quadcopter [60], Autel Robotics EVO [10], and DJI Mavic 2 Zoom [28]. The vibrations captured by the IMU are illustrated in Fig. 3. As can be seen in the Figure, five different drones generate varying vibration profiles. In particular, the main vibration frequencies of DJI Phantom 4 Pro are 95Hz, 195.2Hz, 293Hz, 395Hz and 495Hz. The main vibration frequency of Parrot Bebop 2 is 167Hz. 108Hz, 208Hz, 312Hz, and 419Hz are the main vibration frequencies of an Autel EVO. Next, 90Hz, 108Hz, 183Hz and 219Hz are the main vibration frequencies of a Mavic 2 Zoom drone. Lastly, Parrot ANAFI has 192.2Hz and 346Hz as its main

vibration frequencies. These vibrations affect their transmitted RF signal. DroneScale exploits this information to detect the model of a drone (See Sec. 6.1). One might argue that the drone might be duped if the intruder increases or reduces the angular velocity to change the vibration profile. However, increasing or reduce angular velocity will lead to an increase or reduce total thrust force, which pushes the drones out of its equilibrium stage. As a result, the drone will constantly fly up or down.

■ **Vibration profile of a drone changes when the drone carries impacted external load.** Carrying more load will require the drone to increase the angular velocity of all propellers, thereby increasing the vibration frequency of the drone. To validate this hypothesis, we conducted another set of experiments where the IMU sensor MicroStrain LORD 2DM-GX5-25 is attached to a drone when it carries multiple loads from 50g to 600g. Starting from 100g, we found that the drone's vibration signature starts changing. As an example, when the DJI carries no load and 300g load, the vibration frequency increased due to the higher requirement in thrust force. The vibration frequency increases from 95Hz to 123Hz when the drone carrying 300g of external load. In addition, at the frequency range of 200-300 Hz, carrying 300g load resulted in 60Hz of frequency shift. So, instead of observing at one fixed range of frequency, our algorithms look at the whole vibration spectrum from 0 to 500Hz to extract the drone vibration profile. Similar results are observable across the five different drone models mentioned above.

■ **Estimating the drone's load from afar: An unexplored problem.** Up to this point, we establish the possibility of detecting the model of a drone and whether it carries any impacted load. The next problem is to do it remotely. While a camera, microphone, and radar can offer potential solutions, they have critical weaknesses when applied to drone load estimation. Camera-based approaches can be used to monitor the drone's visual shape and architecture to detect if it carries any additional component or not (e.g., load container). However, the camera can't tell whether the container is empty or full. Microphone based approaches can be used to monitor the sound generated by the drone when it carries additional load; the more load it is carrying, the higher the acoustic frequency that can be captured. However, system performance may suffer from environmental noise and the short distance for acoustic techniques. A radar-based solution can be applied to monitor the drone [66], but this technique requires operation at very high frequencies (K band and W band). In this project, we will prove that tracking drone vibrations through the transmitted communication signal is the most suitable solution because it supports (1) long distance sensing, (2) its performance is more resilient to noise from the surrounding environment, and (3) it is passive.

3 ANALYTICAL MODEL OF DRONE SIGNAL AFFECTED BY VIBRATION

Drone vibration modulates its transmitted RF signal in (1) amplitude, (2) phase, and (3) frequency. While amplitude and phase can be used to detect drone presence, they are highly impacted by environmental noise and would be challenging to be used for high-resolution sensing such as drone model detection or external load monitoring. In this paper, we focus on the impact of vibration to the drone signal in the frequency domain. We now present the relationship between drone maneuvers, its propellers' angular velocities, its vibrations, and the corresponding RF variation.

3.1 Drone Maneuvers, Propeller's Angular Velocity, and Vibrations

■ **Relationship between drone structure and its propeller's angular velocity.** Drones have rotors fixed in parallel to their bodies and they are placed in a square formation with equal distance from the center mass of the drone. A drone is controlled by adjusting the angular velocities of the rotors which are spun by electric motors. The quadcopter is the most common design because of its simple structure. In a quadcopter, there are six states, defined by three position vectors $\xi = |x; y; z|$ and angles pitch (ϕ), roll (θ), yaw (ψ), which specify the rotation about the three axes, but only four control inputs, the angular velocities of the four rotors $\omega_{i=1:4}$ [71]. A drone can be controlled by multiple methods including PID control, back stepping control, nonlinear H∞ control, LQR control, and nonlinear control with nested saturation.

To stabilize the quadcopter, a PID controller is often used because of its simple structure and ease of implementation. The general form of the PID controller is as follows:

$$e(t) = x_d(t) - x(t); u(t) = K_P \cdot e(t) + K_I \int_0^t d\tau + K_D \frac{de(t)}{dt}. \quad (1)$$

where $e(t)$ is the difference between the desired state $x_d(t)$ and the present state $x(t)$, $u(t)$ is the control input, τ is the control torque, and K_P, K_I, and K_D are the parameters for the proportional, integral, and derivative elements of the PID controller [27].

The interactions between the states and the total thrust **T** and the torques τ created by rotors are visible from the quadcopter dynamics. **T** affects the acceleration in the direction of the z-axis and holds the quadcopter in the air. Torque τ_ϕ has an affect on angle ϕ, torque τ_θ affects angle θ, and torque τ_ψ contributes to angle ψ. Assuming $S_x = sin(x)$ and $C_x = cos(x)$, the PID control equations for a quadcopter are:

$$\mathbf{T} = (g + K_{z,D}(\dot{z}_d - \dot{z} + K_{z,P}(z_d - z)))(\mathbf{m}/(C_\phi C_\theta)), \quad (2)$$

$$\tau_\phi = (K_{\phi,D}(\dot{\phi}_d - \dot{\phi}) + K_{\phi,P}(\phi_d - \phi))I_{xx}, \quad (3)$$

$$\tau_\theta = (K_{\theta,D}(\dot{\theta}_d - \dot{\theta}) + K_{\theta,P}(\theta_d - \theta))I_{yy}, \quad (4)$$

$$\tau_\psi = (K_{\psi,D}(\dot{\psi}_d - \dot{\psi}) + K_{\psi,P}(\psi_d - \psi))I_{yy}, \quad (5)$$

where the gravity g, the mass **m** and moments of inertia I of the quadcopter are considered. The angular velocities of the rotors ω_i can be calculated as follows:

$$\omega_1^2 = \frac{\mathbf{T}}{4k} - \frac{\tau_\theta}{2kl} - \frac{\tau_\psi}{4b}; \omega_2^2 = \frac{\mathbf{T}}{4k} - \frac{\tau_\phi}{2kl} + \frac{\tau_\psi}{4b}$$
$$\omega_3^2 = \frac{\mathbf{T}}{4k} + \frac{\tau_\theta}{2kl} - \frac{\tau_\psi}{4b}; \omega_4^2 = \frac{\mathbf{T}}{4k} + \frac{\tau_\phi}{2kl} + \frac{\tau_\psi}{4b} \quad (6)$$

where **k** is the lift constant, **l** is the distance from rotor to central mass, and **b** is the drag constant. Stable angular velocity is unique to a drone model. Changing the angular velocities of drones will push them out of their balance. Hence, from Eqs. 2 and 6, we can infer that a drone's characteristics include its mass (**m**), and structure (**k,l**), which have a significant effect on angular velocities of its propellers ($\omega_{1:4}$). In the following discussion, we will explain how these angular velocities affect the drone's body vibration.

■ **Relationship between propeller's angular velocity and drone vibration.** As discussed in Sec. 2, a drone's body vibration is generated by its propellers' blade vibration. Hence, studying the relationship between angular velocity of the propellers ($\omega_{i=1:4}$) and the drone's body vibration frequency (f_{vib}) is critical to understand how the vibration is transmitted from the drone's propellers to the drone's body. The vibration generated by propellers creates both axial and radial forces at the motor fixtures. These two forces are transmitted from the motor fixtures to the central plate of the drone via the drone's arm. This is proved in the literature with the aid of an accurate computational model of a drone where the spectrum of vibrations captured at the central plate of the drone is in the range of 0-1200 Hz for propeller rotation speeds of 0-36000 rpm [82]. The combined impact of these forces on the drone body is proven to be directly related to the drone vibration velocity v_{vib}, which is the rate of change of peak to peak displacement due to vibration d_{vib}. E.Kuantama et.al [47] analyze the propeller vibration caused due to the action of thrust force and material elasticity of the blades. They use carbon fiber propellers with diameters of 13-inch to do vibration velocity calculation at the propeller hub. The vibration velocity of a 13-inch propeller varies from 0.8 m/s to 3.1 m/s with angular velocity of the propeller varying from 1000 to 5000 rpm. In addition, the drone vibration frequency can be calculated as following: $f_{vib} = \frac{v_{vib}}{\pi d_{vib}}$ (Hz). For all the calculations in the following results we consider a d_{vib} of 5mm. This is because the value is close to the average amplitude of vibration, calculated using an IMU[51] fixed at the center of all the drones used in this research. Considering a d_{vib} of 5mm for a drone's body frame, caused by a 13-inch propeller, the corresponding vibration frequency can be calculated at different angular velocities as shown in Table 1 [47].

ω (krpm)	2	2.5	3	3.5	4	4.5	5
$v_{vib}(m/s)$	1.1	1.2	1.4	1.7	1.9	2	2.3
$f_{vib}(Hz)$	70.02	76.39	89.1	108.22	120.95	127.32	146.42

Table 1: Rotor angular and vibration velocity caused by a DJI's blade

The vibration frequency spectrum of a drone in reality is much broader and more complex due to various factors such as propeller imbalance, phase synchronization, geometry and material of the drone, flight controller response etc. [47, 82] Loaded conditions, especially, would result in higher vibration frequencies due to higher thrust force generation. Up to this point, the relationship between drone architecture and its vibration profile is investigated. In the next section, we will investigate how this vibration profile changes the signal amplitude which is captured by an RF receiver.

3.2 Vibration Creates Frequency Modulation

RF communication is one of the key components of drone control, as discussed in Sec. 2. Most of the existing drones use COTS RF for their communication. 2.4GHz and 5 GHz are typical ISM frequency ranges that are being used [83]. All drones used for this research use 2.4GHz ISM frequency range.
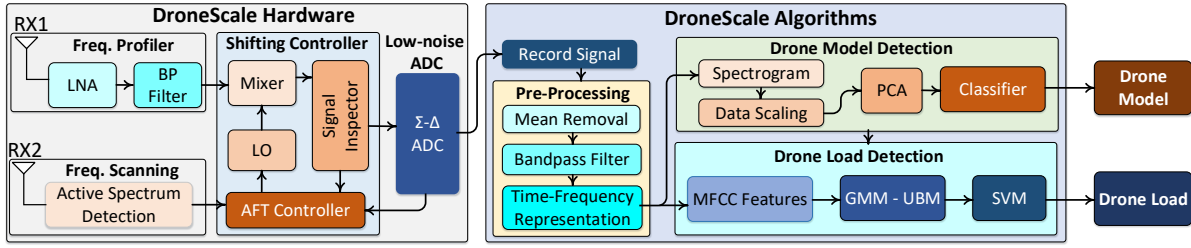
4

Figure 4: DroneScale System Overview

Drone vibrations create a change in displacement between itself and the DroneScale receiver causing a frequency shift in the received signal of the monitoring system. Such a frequency shift can be calculated by the following equation [65]: $\Delta f \approx \frac{2v_{vib}}{c}f_c$, where f_c is the carrier frequency, Δf is the frequency shift. Let us assume an f_{vib} of 200 Hz based on our IMU experiments. When d_{vib} = 5 mm, f_c = 2.4 GHz, $v_{vib} = \pi d_{vib}f_{vib}$, c = 3 * 10^8 m/s, the frequency shift Δf is around 50.3 Hz. This result implies that 1 Hz of drone's body vibration corresponds to 0.25 Hz in RF frequency shift. This also implies that if DroneScale detects the RF shifting at 1Hz level, it can differentiate drones' vibrations that are 4Hz away from each other. Fortunately, different models create different vibration signatures as shown in Fig 3. Based on the IMU results presented in Fig 3, we can say that an RF receiver system with 1Hz resolution would differentiate the RF signals of all the drones used.

The problem becomes much more complex when we consider how much the drone's RF shifts due to an external load. From our experiments we observe that carrying a 300g load generates 65.35Hz of physical vibration frequency at the vibration band between 200-300 Hz for a drone. The frequency shifting caused by additional load can be calculated as follows: $\Delta f' \approx \frac{2v'_{vib}}{c}f$, where $v'_{vib} = v_{vib}+\Delta v_{vib}$ is the vibration speed increased by load, and $\Delta f_{load} = \Delta f' - \Delta f$ is the shifting frequency. So, when the drone carries 300g, the frequency shift is 65.35 Hz. Hence, there is a 15.05 Hz difference when compared with no load (50.3 Hz). Theoretically, every gram of external load would cause 15.05/300 = 0.05 Hz of an RF shift. Hence, a 100g of external load would create an RF shift of 5 Hz. Again, 1Hz resolution would be ideal to differentiate an external load with a 100g resolution.

4 DRONESCALE'S SYSTEM OVERVIEW

To monitor drone vibration with high-resolution using passive RF, the receiver system needs to be able to detect the frequency variations of a 2.4GHz signal with less than or equal to 1Hz resolution as discussed in the previous section. It is challenging to obtain such resolution with existing RF architectures such as software-defined radios. In addition, even if the RF system can monitor the RF signal variations with extremely high resolution, it is difficult to distinguish between the signal variations caused by a drone's vibration and other environmental noises.

■ **DroneScale Hardware** is a passive RF sensing system that is able to detect 1Hz RF variations caused by the drone vibration. Two key components that allow DroneScale to obtain such high-resolution sensing include (1) a new downconversion controller and (2) a low-noise ADC. The downconversion controller controls the LO to downconvert the receiver to receive the signal at a specific frequency band that has the highest Signal-to-Noise ratio (SNR).

Without this downconversion approach, the drone signal appears at low frequency (<1kHz), which is buried under flicker noise. In addition, instead of using SAR ADC architecture like most existing RF devices, DroneScale exploits $\Sigma - \Delta$ ADC architecture has extremely low phase noise that can be used to capture frequency fluctuation of the signal (See Sec. 5).

■ **DroneScale Algorithms** are designed to accurately identify the drone model and its load based on the signal collected from the hardware. In particular, it first pre-processes the data, cleans out the noise, and then extracts the vibration signals (See. Sec. 6). The extracted signal is transformed using a Short Time Fourier Transform (STFT), to the frequency domain, for drone model detection, and to a mel-frequency cepstrum, for external load detection. When the model is detected, DroneScale compares the captured vibrations profiles using the GMM-UBM model with MFCC features to estimate how much external load the drone is carrying. In the following sections, we will discuss these components in detail.

5 RF HARDWARE DESIGN FOR HIGH-RESOLUTION MONITORING

5.1 Noise Characteristics of RF Systems

Instead of analyzing the drone signal at the high-noise frequency range (less than 1kHz), we downconvert the drone signal to a higher frequency band to remove the impact of a high noise (flicker noise) area near DC. Before discussing how this mechanism was implemented, let's discuss the characteristics of the noise experienced in today's RF receivers. There are several noise sources that can compromise the sensitivity of a common RF receiver. They can be grouped into three types based on their frequency characteristics, namely, (1) *white noise*, (2) *phase noise*, and (3) *flicker (1/f) noise* as illustrated in Fig. 5 (a) [44, 63, 80].

First, white noise has a flat power spectral density over the Nyquist spectrum. Its sources include the ADC (Analog-to-Digital Converter) quantization noise, thermal noise of the electronics, and noise inside the MOS (Metal-Oxide-Silicon) transistors. Quantization noise is introduced by the ADC when an analog sample is approximated by a discrete digital value. Thermal movements of electrons inside resistors and capacitors creates the thermal noise of the receiver. With a high-speed ADC, which is usually employed in existing RF receivers, the thermal noise of the sampling track-and-hold capacitor could be a significant source dominating the white noise spectrum [63]. Noise Spectral Density (NSD) is the quantitative metric to measure the noise floor (white noise) of an ADC [44]. It is defined as the noise power that is sampled at the input of an ADC per unit of bandwidth (i.e. 1Hz). There are four factors that we would need to consider when calculating NSD such as (1) effective noise-free resolution (ENOB), (2) sampling frequency

(fs), (3) full-scale signal power V_{rms}^2/R_{in} (V_{rms}: root mean square of the maximum voltage, R_{in}: input resistance of the ADC), and (4) architecture of the ADC. Generally, with a Nyquist-rate ADC that has a flat noise floor across the Nyquist spectrum, we can calculate NSD (dBm/Hz)through this equation:

$$NSD = 10 * log(\frac{V_{rms}^2}{R_{in.}}) + 30 - (6.02 * ENOB + 1.76) - 10 * log_{10}(fs/2) \quad (7)$$

Second, phase noise has a lower frequency component near the input signal and a broadband component that adds to the white noise floor. It is created because of the imperfection (i.e. jitter) of the ADC's sampling clock [12, 22]. The sampling clock jitter will spread the input signal energy to nearby frequencies thereby reducing the overall spectral resolution. The signal-to-noise ratio (SNR) of an input frequency (f_i) limited by the clock jitter (t_j) can be calculated through this equation, i.e. $SNR = -20 * log(2\pi f_i t_j)$ [12]. Thus, with the same clock jitter, phase noise increases with the frequency of the input signal and vice versa.
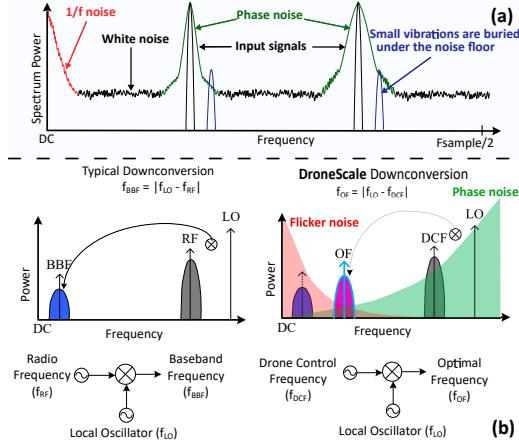

Figure 5: Typical noises distribution (a) and DroneScale frequency downconversion concept (b)

Last, flicker noise arises from the population and depopulation of individual charges carrier 'trap' in semiconductor devices [63]. Flicker noise is dominant near DC and its PSD increases inversely with frequency [64]. To effectively characterize drones, several factors have to be considered to create a sensitive, low noise, and high-speed RF receiver. Those factors include building a high resolution ADC and deriving a new downconversion frequency approach, which will be discussed next.

5.2 Bringing Drone Signals to Low-noise Bands

To downconvert the signal to an intermediate frequency for further processing, a mixer is used. A frequency mixer is a 3-port electronic circuit. Two of the ports are 'input' ports and the other port is an 'output' port. The ideal mixer 'mixes' the two input signals and the output signal frequency is either the sum or difference of frequencies of the inputs: $f_{out} = f_{in1} \pm f_{in2}$. The three ports of a mixer include the Local Oscillator (LO) port, the Radio Frequency (RF) port, and the Intermediate Frequency (IF) port. The LO port is typically driven with either a sinusoidal continuous wave (CW) signal or a square wave signal. The relationship between input and output frequencies is given by $f_{IF} = |f_{LO} - f_{RF}|$. A typical downconversion operation is shown in Fig. 5 (b) (left). The flicker

noise level of a mixer is high at low frequencies of a spectrum and decreases as the frequency increases. Common RF mixers such as the CMOS Gilbert cell mixers can have significant flicker noise power up to 50-100KHz [48, 52]. Using traditional approach would bring the drone signal to low IF frequency (1kHz), which is buried under the noise floor. Phase noise, on the other hand, increases with frequency. High phase noise level can lower the spectral resolution and reduce the sensitivity of our system. As a result, we would need to find the optimal location, which does not suffer from both flicker noise and phase noise in our frequency spectrum, to analyze frequency modulations caused by drone body vibrations as illustrated in Fig. 5 (b) (right).

In particular, the traditional approach downconverts the RF signal to baseband before pre-processing. The LO frequency and the RF frequency are designed to be identical. For example, to capture drone vibration at IF frequency, a Wi-Fi transmitter that operates at 2.4 GHz can be downconverted using 2.4GHz LO frequency. However, this brings the modulated signal to DC central where the signal is highly impacted by white noise and flicker noise of a mixer as illustrated in Fig. 5 (a). To overcome this problem, we propose a method to shift the frequency far away from the noisy near-DC frequency range by adjusting the LO signal. For example, using a 2.3GHZ LO frequency helps bring the modulated signal to 1.01MHz which is still detectable using an ADC with a sampling frequency of 2.02 MHz. However, shifting the signal too far from low frequency bands will result in an increase of phase noise from the ADC. To overcome this problem, our proposed system calculates an optimal downconversion frequency (f_{OF}), which strikes a balance between flicker noise and phase noise levels to increase the Signal-to-Noise ratio of the received modulated signal. The LO is configured to downconvert the received modulated signal to f_{OF} without prior knowledge about manufacturing specification details of a mixer and an ADC that define their noise.

f_{OF} is calculated by the noise level (dBFS) at different input signal tones using SNR degradation theory [6]. In the current setup, drones are streaming data to the remote control with 4MHz bandwidth. The starting point of the spectrum (f_{drone}) is used to configure the LO. In addition, the f_{OF} of our receiver architecture is at 25kHz. Hence, the LO is configured to operate at $f_{LO} = f_{drone}$ + 25kHz. From the above theory f_{LO} can be formulated as:

$$f_{LO} = f_{drone} + f_{OF} \quad (8)$$

5.3 ADC Architecture for Drone Monitoring

While the mechanism mentioned above brings the drone signal to a low-noise band for analysis, the captured signal is still buried under the noise floor of conventional high-speed ADCs used by most current RF receivers. We now discuss how sigma-delta ADC architectures should replace existing successive-approximation (SAR) architectures for high resolution (i.e., <= 1Hz) RF monitoring.

There are three popular ADC architectures for data acquisition in the market, namely (1) pipelined, (2) successive-approximation (SAR), and (3) sigma-delta ($\Sigma - \Delta$), as shown in Fig. 6 (a) [46]. Pipelined and SAR ADCs offer high sampling rate and are widely used in RF receivers. However, to trade off with their high sampling rate, they tend to have low resolution and high noise floor. From our in-lab experiment, these factors can directly compromise the spectral resolution and the ability to capture the variations in a
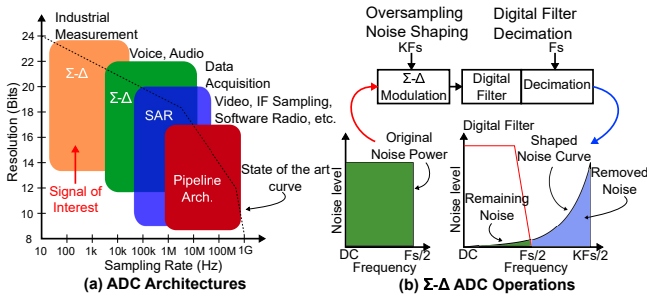
Figure 6: ADC architectures

drone's vibration with external loads, which are typically at low frequency (i.e. $< 1kHz$). To overcome this challenge, we employ the sigma-delta architecture, which has high resolution and low noise floor. In a sigma-delta ADC, low noise is achieved by utilizing oversampling, noise shaping, digital filtering, and decimation as illustrated in Fig. 6 (b) [20]. As mentioned in Sec. 5.1, white noise is spread out uniformly over the Nyquist spectrum. Hence, oversampling widens the Nyquist spectrum, thereby, reducing the white noise energy in the spectrum of interest. A noise shaping function, i.e. Sigma-Delta modulator, is then applied to the signal to shape the noise function so that most of the noise energy is shifted to the high frequency of the wider Nyquist spectrum leaving a low noise energy in the spectrum of interest. This noise level is then removed by a digital filter. The signal is decimated to the required sampling rate before outputting the results. Additionally, Sigma-Delta architecture also provides with higher bit-resolution (i.e. 24-32 bits) than SAR or Pipelined counterparts further reducing the effective noise floor. The main disadvantage of the Sigma-Delta architecture is the low sampling rate due to the additional processing required for noise shaping and modulation functions. Thus, our signals need to be shifted to a low frequency range to be effectively captured. For a sigma-delta ADC with noise shaping modulation, the noise floor of the spectrum of interest is also relatively flat. Thus, it is also applicable within that spectrum. For example, we see that the average noise floor of a high-speed sigma-delta ADC ADS1675 (V_{rms} = 4.24V, R_{in} = 400 Ohm, fs = 4MHz, $ENOB_{4MHz}$ = 17.06) employed in our design could be as low as -150.94 (dBm/Hz). In Sec. 7.1, we show that our customized hardware can achieve much lower noise floor than USRP devices. Hence, it enables the ability to capture 1Hz vibrations which is not possible with USRPs.

5.4 DroneScale Prototype

We implemented the system as shown in Fig. 7 using a Texas Instruments TI ADS1675 4MSPS, 24-Bit Analog-to-Digital Converter, Delta-Sigma architecture. This is a high-speed, high-precision analog-to-digital converter using an advanced delta-sigma ($\Delta\Sigma$) architecture, and performs up to 4MSPS with high AC performance and DC accuracy. We use a Pasternack band pass filter VBF-2435+ from Mini-Circuit that allows 2.340-2.530 GHz to pass through with an insertion loss 1.61 dB at 2.440 GHz and VSWR is 1.35:1. A Double Balanced Mixer PE86X1020 is used for mixing the upcoming signal and the LO. This mixer utilizes Schottky Ring Quad Diodes with optimized balun structures. The signal is captured using an Ettus LP0965 antenna which operates from 850 MHz to 6.5 GHz for short distance and in-lab experiments. For field testing, we use a Pasternack PE51049 19 dBi 2.4 GHz directional antenna. Subsection 7.1
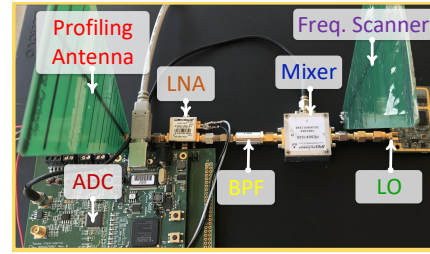

Figure 7: DroneScale prototype

describes the system performance and its ability to track 1Hz variation in details. Moreover, DroneScale has to know what the active frequency of the drone's communication signal is. While some drones use COTS Wi-Fi transceivers for communication, which can be captured through preambles that are publicly broadcast, other drones like DJI implement their own pre-established connection between the drone and its controller where their preamble can't be seen. We exploited a software-defined radio hardware solution where the device scans across different spectra to identify the active frequency and provides the value of f_{drone} in Eq.8.

6 DRONE MODEL AND LOAD ESTIMATION ALGORITHMS

In this section, we discuss the signal pre-processing techniques that allow the system to remove noise and only focus on the region of interest. We then explain the features used to detect the model and external load of the drone. Lastly, we present the details of the learning algorithm that was developed to realize that idea. The signal is passed through a band-pass filter to remove the unexpected noise. A drone's RF signal is classified in two stages, with the first stage being its model detection and the second stage being its external load estimation. Doing a 2-stage classification is necessary because each drone model has a unique vibration profile, as shown in Fig. 3, and as a result each drone's vibration profile varies in a unique way depending on its model and physical construction when additional load is added.

6.1 PCA Dimensionality Reduction with STFT Features for Model Detection

From observing the vibration profiles of different drones in Fig 3, we notice that only a handful of points in a wide vibration spectrum help us to classify the drones' vibration profiles from each other. Hence, eliminating the features which remain the same in all of the RF signal samples would simplify the classification task. Principal Component Analysis (PCA) helps in eliminating such redundant features effectively, thereby improving classification efficiency. We use Short Time Fourier Transforms (STFT) of the time series data of the drone signal captured by our sensitive receiver system. STFTs are windowed such that each sample consists of 10240 datapoints which capture the frequency shift of a drone's signal due to its body vibration with enough resolution, as discussed in Sec.3.2, on a time-frequency scale. Each datapoint is intended to be a feature that is used for the classification. To get a well-represented dataset that captures the variance in a high dimensional feature space like such, we need a massive sample size which is not practical for our application. Such a disproportion between the sample size and number of features in a dataset will lead to sparsity in the
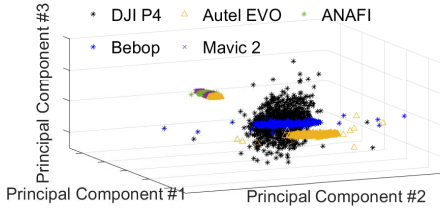
Figure 8: PCA output for model estimation using 3/85 features for representation

dataset when projected in a high dimensional space, popularly known as the curse of dimensionality [18]. The optimal number of features for a sample size n lies between $\sqrt{n}$ and n-1 depending on the correlation between the features [43]. A general method to determine the degree of sparsity in a dataset is to calculate the covariance matrix S of the dataset and examine the eigenvalue distribution for each eigenvector of S. A big range in the eigenvalues of S suggest a sparse dataset that can be optimized for features. Covariance matrix, by nature, reduces the number of features to n. The covariance matrix is given by: $S = \frac{1}{n-1} \sum_{i=1}^{n} \sum_{j=1}^{n} (X_i - \mu_i)(X_j - \mu_j)^T$, where X_i is the ith feature vector, μ_i is the mean of the ith feature vector given by $\mu = \frac{1}{n} \sum_{j=1}^{n} x_{i,j}$ and n is the sample size.

Next, we perform eigendecomposition of the covariance matrix which gives us n eigenvectors and eigenvalues. The eigenvalues calculated for the drone signal dataset in the dimension space of order n, defined by n eigenvectors, suggest that more than 95% of those dimensions are redundant (eigenvalues≈0) when compared with the rest. A very small number of features bear most of the information regarding the distribution of the dataset, quantified by variance. A widely accepted dimensionality reduction method that extends this approach is called the Principal Component Analysis (PCA) [68]. PCA takes k eigenvectors with the highest eigenvalues and projects the features onto a k - dimensional space defined by the chosen eigenvectors with the help of a transformation matrix T. The choice of k is made by monitoring the contribution of variance of each added dimension to the dataset, where the variance in a dimension is calculated using $\sigma_k^2 = \frac{1}{n} \sum_{i=1}^{n} (x_{i,k} - \mu_k)^2$. , where $x_{i,k}$ is the ith sample in k^{th} dimension, μ_k is the mean of the sample size in the kth dimension, n is the sample size. The percentage of variance explained by each dimension is given by: $\sigma_k^2(\%) = \frac{\sigma_k^2}{\sum_{i=1}^{k} \sigma_i^2}$.

The percentage of variance explained by a dimension becomes less than 0.1% beyond a certain value of k, which is observed to be the optimal selection of dimensions for a dataset. The dataset is scaled to have zero mean and unit variance before performing PCA so that the eigen-decomposition results are accurate. Training and test sets were divided for a 10 - fold cross validation. The Transformation matrix T is derived from the training set and applied to the test set during each fold of cross validation, after deriving and applying an appropriate scaling matrix in a similar fashion. Fig.8 represents the distribution of the drone signal from different drones after PCA reduction, in 3 of the 85 dimensions used for classification. The data for the figure is taken from one of the training sets used in the 10-fold cross validation process. The 3 dimensions shown in the figure explain 76.3% of variance in the dataset. The other dimensions account for 22.2% of variance. An addition of more dimensions has no effect on the explained variance percentage.

Following the dimensionality reduction, we use the Random Forests Classifier [13] with 250 decision trees to classify the drone signal dataset according to its model. Any standard classifier such as Support Vector Machines(SVM), Decision Trees, KNN etc. can be optimized and used for the classification task, but Random Forests is chosen due to its many optimization options that strike a balance between bias and variance of the training sets.

6.2 GMM-UBM Supervector Normalization with MFCC Feature for Load Estimation

We utilize the Mel-frequency cepstrum (MFCC) feature to extract the signatures of the vibration behaviors of the drones carrying different weights. We then apply the GMM-UBM supervector to normalize the MFCC feature so that it can be classified by any learning mechanism.

Feature Extraction: MFCC is widely used in speech recognition and music retrieval. The technique converts signal data to the Mel-scale domain and then computes the filter bank coefficients. We propose the technique of using the Gaussian Mixture Model with the Universal Background Model to normalize the set of features into a signal vector, named supervector, to process the data before using a classifier.

GMM-UBM supervector. Gaussian Mixture Model (GMM) is a statistical approach to identify the distribution of a given data set. Given a data set $X = X_1, X_2, .., X_n$ where X_i is a feature vector, we find a model using GMM that best fits the distribution of X. GMM is represented by a set of parameters $\lambda^m = \{\omega^m, \mu^m, \Sigma^m\}$, where ω^m, μ^m, Σ^m are the weight, mean and covariance of an m-th gaussian component respectively. The probability of a sample x is a weighted sum of M Gaussian components: $p(x|\lambda) = \sum_{m=1}^{M} \omega^m g(x|\mu^m, \Sigma^m)$, where g is the Gaussian function of m-th mixture, calculated as:
$g(x|\mu^m, \Sigma^m) = \frac{1}{\sqrt{(2\pi)|\Sigma^m|^{\frac{1}{2}}}} exp - \frac{1}{2}(x - \mu^m)(\Sigma^m)^{-1}(x - \mu^m)$. A general practice is to choose M to be 3.

Expectation Maximization Algorithm: To fit a GMM model to the data set X, we adopt the Expectation Maximization (EM) algorithm. According to [26], values of the parameters are updated for each iteration as follows: $\overline{\omega^m} = \frac{1}{n} \sum_{i=1}^{n} Pr(m|X_i, \lambda)$, the updated mean value of m-th mixture $\overline{\mu^m} = \sum_{i=1}^{n} Pr(m|X_i, \lambda)X_i / Pr(m|X_i, \lambda)$. In addition, the updated variance of m-th mixture is calculated as: $\overline{(\Sigma^m)^2} = \sum_{i=1}^{n} Pr(m|X_i, \lambda)X_i^2 / Pr(m|X_i, \lambda) - \overline{\mu^m}$, where $Pr(m|X_i, \lambda) = \frac{\omega^m g(X_i|\mu^m, \Sigma^m)}{\sum_{j=1}^{M} \omega^m g(X_i|\mu^j, \Sigma^j)}$ is the probability of mth-gaussian given the data X_i and parameter set λ. We apply the EM algorithm to all the training samples to generate an overall GMM, named Universal Background Model (UBM). Then, with each data sample, we adapt the UBM model using the Maximum a Posterior (MAP) adaptation to generate a unique GMM model for this sample set. The use of UBM model keeps all the generated GMMs under the same scenario, thus, reducing the outliers.

Maximum a Posteriori Estimation: Given a MFCC feature set $X = X_1, X_2, .., X_n$ and a UBM model λ_{UBM}, the required statistics variables including n^m, $E^m(X)$, and $E^m(X^2)$ can be calculated as: $n^m = \frac{1}{n} \sum_{i=1}^{n} Pr(m|X_i, \lambda_{UBM})$; $E^m(X) = \frac{1}{n^m} \sum_{i=1}^{n} Pr(m|X_i, \lambda_{UBM})X_i$; $E^m(X^2) = \frac{1}{n^m} \sum_{i=1}^{n} Pr(m|X_i, \lambda_{UBM})X_i^2$.

The statistical variables above are used to update the Gaussian mixture parameters. In particular, n^m corresponds to the weight
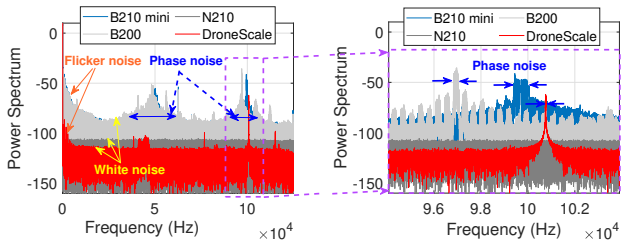
Figure 9: Comparing the received signal captured by DroneScale hardware and USRPs.

value, $E^m(X)$ and $E^m(X^2)$ are used to update mean and variance respectively: $\hat{\omega}^m = [\alpha_\omega^m \frac{n_m}{T} + (1 - \alpha_\omega^m)\omega^m]$; $\hat{\mu}^m = [\alpha_\mu^m E^m(X) + (1 - \alpha_\mu^m)\mu^m]$; $(\hat{\Sigma}^m)^2 = [\alpha_\Sigma^m E^m(X^2) + (1 - \alpha_\Sigma^m)((\Sigma^m)^2 + (\mu^m)^2) - (\hat{\mu}^m)^2]$ To be convenient, the value of $\alpha_\omega^m, \alpha_\mu^m$, and α_Σ^m are set to be equal to $\frac{n^m}{n^m+r}$, with r being the relevance factor to avoid dividing by 0 [70].

Support Vector Machine with modified GMM-UBM kernel: Among various approach, Support Vector Machine is the most conventional method that optimizes a hyper-plane to efficiently classify data. We use GMM Kullback-Leibler divergence kernel (GMM KL) [55]. Given two GMM models λ_P and λ_Q, the GMMKL kernel is defined as $K_{GMMKL}(\lambda_P, \lambda_Q) = \frac{1}{2} \sum_{m=1}^{M} [(\sqrt{\omega_{UBM}^m}(\Sigma_{UBM}^m)^{-1/2}\mu_P^m)^T = (\sqrt{\omega_{UBM}^m}(\Sigma_{UBM}^m)^{-1/2}\mu_Q^m)]$. The defined GMMKL kernel is then used for classification.

7 PERFORMANCE EVALUATION

In this section, we evaluate the performance of DroneScale's hardware and algorithms, and DroneScale's ability to detect drone model and load through in-lab and field-test experiments. The experiments are designed to answer the following questions: (1) *How sensitive is the developed system while detecting the frequency fluctuations? What are the accuracies when DroneScale estimates external loads carried by different drones?* (2) *What is the accuracy of DroneScale when detecting a drone's model?* (3) *Does the placement of the drone's external load impact the system's accuracy?*, (4) *Can the system work when the drone operates in autonomous mode?*, and (5) *Does the distance impact the accuracy of the estimation?*

7.1 Validation of DroneScale Hardware Sensitivity to Frequency Variations

We conduct four in-lab experiments to assess the hardware sensitivity of DroneScale to frequency variations.

7.1.1 Tracking single tone signal. Note that the low-noise region of DroneScale hardware starts from 25kHz. 100kHz is the f_{OF} for tracking a single tone signal. In the first experiment, an RF transmitter transmits a 2400MHz signal being less than 20m away from four different receivers, namely DroneScale, USRP B210 mini, B200, and N210-CBX daughter board. USRP's hardware is configured to listen to 2399.9 MHz frequency so that the received signal will appear in the 100kHz area. This would help the signal visible to USRP's receiver because it is out of flicker noise generated by their mixer and it is then easier to compare to DroneScale's hardware. DroneScale profiles the signal and controls its LO to operate at 2399.9 MHz. The transmitted signal is then also shifted to
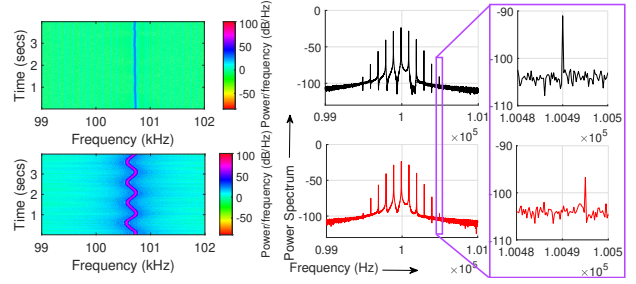

Figure 10: Signal captured by DroneScale system when there is no frequency modulation (top-left), 1Hz frequency modulation (bottom-left), 100Hz FM rate modulation (top-right), and 101Hz FM rate modulation (bottom-right)

100kHz for further analysis. The received signal captured by them is shown in Fig. 9. While N210 is not able to capture the 100kHz single tone signal, B210 mini, B200 are able to capture 100kHz signal with extremely high phase noise. Meanwhile, the noise level of the DroneScale's hardware is significantly lower compared to that of USRPs. This result illustrates our system's ability to track the frequency fluctuations at high resolution.

7.1.2 Detecting signals that are 1Hz away from each other. In the second experiment, the RF transmitter transmits signals that are 1Hz apart from each other including 2400 MHz, 2400.000001 MHz, 2400.000002 MHz, 2400.000003 MHz, ..., 2400.000005 MHz. The LO operates at 2399.9 MHz to downconvert the signal of interest to the 100kHz band. Of all the four receivers, only DroneScale is able to differentiate the signals that are 1Hz away from each other. B210 mini, B200, and N210 are not able to detect signals that are 1Hz away from each other due to their high noise floor. This confirms that our system is able to monitor the signal at low frequency (such as 1kHz) while they are not observable by the USRPs.

7.1.3 Detecting RF bandwidth variation of 1Hz. In the third experiment, the transmitter transmits a single tone signal at 2400 MHz modulated by an FM modulation of 100 Hz deviation and 1Hz FM rate. This means that the signal is fluctuating with 100Hz bandwidth at the fluctuation frequency of 1Hz. Fig. 10 (top left) shows the single tone signal when our hardware downconverts the signal of interest to 100kHz. When the transmitter modulates the signal using FM with 100Hz deviation and 1Hz FM rate, the captured signal is shown in Fig. 10 (bottom left). This implies DroneScale hardware's ability to capture the signal fluctuation at 1Hz resolution.

To illustrate the ability of DroneScale in capturing expansion and collapse of RF bandwidth, we conduct the fourth experiment where the transmitter is configured to transmit 2400 MHz signal with FM modulation at (1) 100Hz deviation at 100 Hz FM rate and (2) 100Hz deviation at 101 Hz FM rate (i.e., bandwidth expansion of 1Hz). If DroneScale hardware is able to detect bandwidth expansion of 1Hz, this implies its ability to capture 1Hz resolution of RF variations. Fig. 10 (right) shows the Fourier Transform of the received signal in two cases. As can be seen in the Figure, the sideband of 100 Hz FM rate signal (top right) and 101 Hz FM rate signal (bottom right) are distinguishable. This confirms that the system is able to monitor the bandwidth expansion at 1Hz resolution. These results also conclude that existing USRP architecture cannot be used to
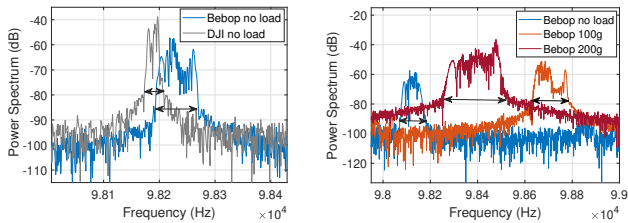
Figure 11: (Left) DJI and Bebop vibration profile captured by RF signal, (Right) Bebop vibration profile when carrying external load captured by RF signal

Figure 12: Tested drones

track drone's load, hence, we will not discuss their performance in terms of load's detection further.

7.1.4 Detecting bandwidth expansion variation caused by drone vibration. In the fifth experiment, we attach an antenna of a USRP transmitter onto the drones. This transmitter is configured to transmit the signal at 2400 MHz. DroneScale hardware controls the LO to shift the IF signal to the 100kHz region (i.e., LO is tuned to 2399.9MHz) where it obtains maximum Signal-to-Noise Ratio. Fig. 11 (left) shows the signals transmitted by the antenna when mounted on DJI Phantom and Parrot Bebop drones, as captured by the DroneScale hardware. The difference in the bandwidth of the monitored signals captures the impact of the drones' vibrations. We place different loads (100g, 200g) onto the Bebop drone and observe the signal transmitted from the attached antenna. Fig. 11 (right) shows the vibration profile extracted from the DroneScale hardware. As can be seen in the Figure, the bandwidth of the monitored signal is captured and can clearly be differentiated from each other. This confirms the feasibility of monitoring drone load using the RF signal. In the following discussion, we will discuss how the system performs with the communication signals transmitted by the drones themselves under various conditions.

7.2 Drone's Load Estimation Evaluation

7.2.1 Evaluation Methodology. We evaluate the system with 5 different models of drones, namely, Parrot ANAFI [60], DJI Phantom 4 Pro [29], DJI Mavic 2 Zoom [28], 2 × Parrot Bebop 2 [61], Autel EVO [10] as shown in Fig. 12. Two bebop drones were used during the experiments. As they have identical vibration profiles, their data is combined. All drones communicate with their controller at 2.4GHz frequency. The drones are attached with multiple precision weights [5] mounted underneath them during flight. Each drone is loaded with weights in the range of 0-400g. The drone is flying at a distance of up to 200m from the receiver. We could not go beyond 200m during testing due to the limitations of our testing
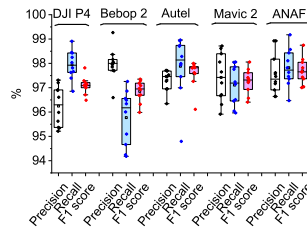

Figure 13: Accuracy of model detection

Figure 14: Drone load estimation accuracy-10 folds

Figure 15: ANAFI load estimation accuracy

Figure 16: Bebop 2 load estimation accuracy

facility and the COVID-19 lockdowns. A drone's body vibration introduces frequency, amplitude and phase modulation into it's communication signal. As we base our system on the frequency modulation detection, we limit the effects of distance. For most commercial drones, the communication signal can be transmitted upto 4km away from it. Because of that, we strongly believe that DroneScale will be able to perform reasonably well for at least hundreds of meters distance. The data is collected in a period of 2 months making for 9596 seconds of flight data. The location of data collection is a residential area with more than 10 active signals being captured from Wi-Fi access points around the neighborhood by DroneScale at all times. With a full charge, each drone could fly for about 20 minutes. It takes around 1.5 hours to fully charge each battery. DroneScale collects the samples at 5MHz and then downsamples the rate to 250 kHz. The drone signal is downconverted to 25kHz for further analysis. The band from 25kHz to 125kHz with the lowest noise range is used to track the drone's signal. The dataset is populated with files consisting of 31250 time-series data points, amounting to $1/8^{th}$ of a second each. The drones were flying 10-15 meters away from the receiver architecture. DJI Phantom 4 and Parrot Bebop 2 were tested with two different load placements, for each load category, to test the effect of load distribution on the classification results. Additional data was collected for DJI Phantom 4 and Parrot Bebop 2 to do sensitivity analysis. For that purpose, DJI Phantom 4 was also flown at a distance of 200 meters to perform the same experiments. Similarly, Bebop 2 was flown in an autonomous mode, using GPS, resulting in rapid variation of distance of the drone from the receiver during flight. The dataset is well balanced with negligible differences in sample sizes of each category.

7.2.2 Drone model detection results. The first step implemented in DroneScale after collecting a drone's RF communication data is to classify the drone according to its model before moving on to its external load estimation. The dataset used for model detection has 4800, 2688, 4680, 4640 and 2656 samples of RF data for Phantom, Bebop, EVO, Mavic, and ANAFI drones respectively. The data collection is done such that every combination of drone model and external load is more or less uniformly distributed. But because
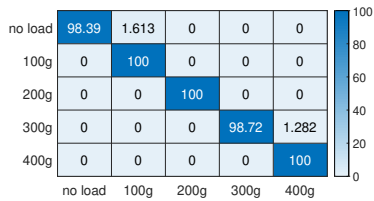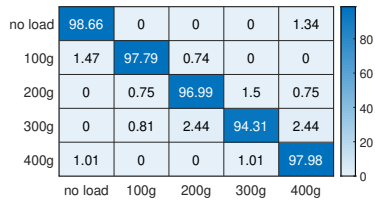
10

Figure 17: Autel EVO load estimation accuracy

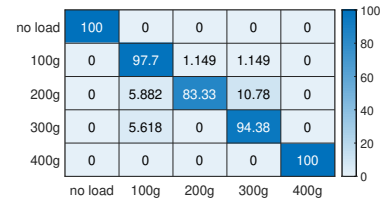**Figure 18: DJI Phantom 4 Pro load estimation accuracy**

Figure 19: Mavic 2 load estimation accuracy
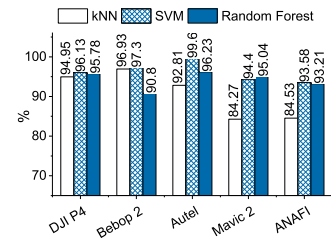

Figure 20: Load estimation using different algorithms
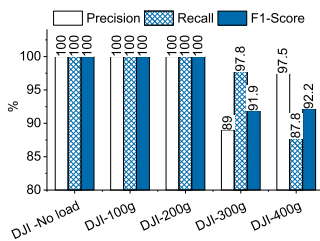

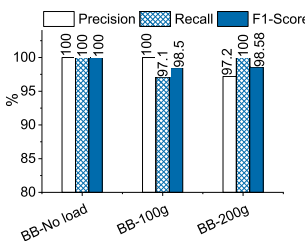**Figure 21: Estimation accuracy at 200m distance**

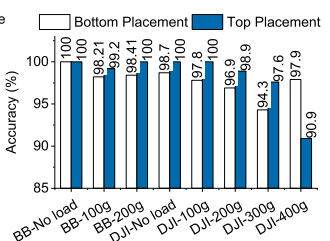**Figure 22: Estimation accuracy in autonomous mode**

Figure 23: Estimation accuracy at different placement

of the loading capabilities of Bebop and ANAFI (0-200g instead of 0-400g), we would have a discrepancy in the distribution of the dataset when we combine the load categories for model detection. To train the model detection algorithm (Sec 6.1), we made sure to have a uniform distribution of 2600 samples in each model category while making sure that every external load category is evenly distributed in a model's sample size. A 10-fold cross validation [13] is run on the data to ensure the robustness of the classification. The validation process divides the dataset into 10 uniformly distributed parts, where 9 parts are trained on while one part acts as the testing set during each fold of validation. The results presented in Fig.13 show that the scores have a very small variance confirming their validity. The average accuracy score for model detection is observed to be 97.2%, showing the robustness of DroneScale during the first stage of drone's RF signal classification.

7.2.3 Load estimation results. Following the model detection of a drone, we evaluate the drones' RF signals to be classified based on their external loads. In this experiment, DJI Phantom 4 Pro, DJI Mavic 2, and Autel EVO carry loads of 0-400g with a 100g resolution, while, Bebop 2 and ANAFI carry 0-200g with 100g resolution. While Bebop and ANAFI can carry up to 250g, 200g is a safe mounted load. Similarly, DJI Phantom 4, Autel EVO, and DJI Mavic can carry up to 600g, but 400g is a safe mounted load when they do not generate abnormal behavior during flights. We used 4800, 4680, 4640 samples for 5 load categories of DJI Phantom 4, Autel EVO, DJI Mavic, and 2688, 2656 samples for 3 load categories of Bebop, ANAFI to train the load classification algorithm. The loads were placed underneath these drones in all the experiments of this section. The load range for each drone was chosen based on its specifications to ensure balance and safety during flight. A 10-fold cross-validation process similar to the first stage of classification is employed. The average load estimation accuracies of ANAFI, Bebop 2, Autel, DJI Phantom 4, and Mavic 2 are 93.65%, 97.3%, 99.65%, 97.14%, 94.7%, respectively. The scores with low variance throughout the 10 folds of validation confirms the validity of the results (See Fig.14). The average of the confusion matrices from 10-folds for all the drones is presented in Figures 15 - 19. The accuracy scores are high even when the

load estimation algorithm is implemented with classifiers other than SVM (See Fig. 20). More specifically, while Random forest obtains 90.27% and kNN obtains only 83.5% average accuracy, our SVM classification algorithm using GMM-UBM kernel with MFCC features obtains 96.27% accuracy. This proves the robustness of our proposed techniques discussed in Sec. 6.2.

7.3 Sensitivity Analysis

7.3.1 Impact of distance. While the above results were obtained when the drone is flying with a distance less than 20m, we also evaluate the impact of distance between the drone and DroneScale up to 200m with the DJI Phantom drone. This distance limit is set due to the limitation of our testing facility and the impact of COVID-19 lockdown on outdoor testing. Fig. 21 shows the accuracy of estimating the load carried by a DJI drone when it is at a 200m distance. The dataset consists of five load classes with near-uniform distribution, similar to that of shorter distance experiments. The total sample size for this experiment is 4576, which is very close to the closer-range counterpart. The average accuracy of the 10-fold cross-validation scores is 96.87% which shows that the impact of a distance of up to 200m from DroneScale is negligible. This result is promising in terms of demonstrating DroneScale's performance in estimating the drone's load at distance.

7.3.2 Impact of autonomous flight mode. Most of the drones still stream their data including video, audio, and sensor data continuously to the controller even though they are in autonomous flying mode. To evaluate the system performance when the drone is in autonomous flying mode, we use the Bebop flight planning app to assign the location where the drone flies back and forth during which the drone's RF signals are captured using our receiver system. The other experimental conditions and classification algorithms remain the same as discussed in Sec. 7.2.3. Fig. 22 illustrates the system performance when the Bebop drone is carrying up to 200g of load in autonomous flying mode. The estimation performance is almost similar (99.05% of average accuracy using 10-fold cross-validation). However, this works only when the sensor status and video streaming data is constantly being transmitted by the

11

drone to its controller device. When the drone operates in a total autonomous mode without transmitting any signal, camera/audio-based approach can be considered as an alternative solution.

7.3.3 Impact of load placement location on a drone. We evaluate the system with experiments where the loads are placed on the top of a drone instead of making them hang underneath to see if there is any impact of load placement location on the classification results. These experiments were conducted on DJI Phantom and Bebop drones. The experiments and classification were done without any changes to other parameters discussed in Sec. 7.2.3. While Fig. 16 and 18 show the results when the drones are attached underneath the drone body, Fig. 23 shows the average recall scores for each category, after 10-fold cross-validation, when the loads are placed on top of the drone. The average accuracy scores are 98.74% for DJI Phantom and 99.13% for Bebop, a slight improvement in the accuracy scores when compared to that of underneath loading placement experiments. This improvement is obtained because the load can be placed close to the central mass of the drone with top placement. Placing load perfectly underneath the drone is often challenging as its mounted location has to be adjusted so it doesn't block the drone's sensor (e.g., landing sensor).

8 RELATED WORK

Drone monitoring systems have attracted much attention in the research community recently. First of all, continuous-Wave (CW) radar systems can detect drones' activities in different frequency bands [32, 40, 49, 76, 77]. However, the radar-based approach is (i) sensitive to the relative orientation of drones with respect to the radar [39, 72], and (ii) hard to classify drones' activities from birds' due to the weak signatures from kinetic elements of the drone [69].

Camera-based approaches have also been designed for drone monitoring such as developing image processing method [36] or multiple camera arrays system [4, 21, 56, 62, 73] to track drones. However, there is no possible camera-based approach to the drone's load estimation. Acoustic detection methods extracting acoustic signatures from different drones are also well studied [11, 17, 38, 45, 54, 74]. Some research explores audio assisted video drone detection [14, 19, 50]. There are some intrinsic shortcomings, including vulnerability to distance and ambient noise, and a huge variety of acoustic drone signatures[7, 11, 42]. S. Ramesh et al. proposed a drone and motor fingerprint method utilizing the interplay between motors and each of its own unique acoustic patterns [67]. This is a promising approach that can also be leveraged for load estimation.

9 DISCUSSION

First, while the current results are encouraging, we wish to conduct more experiments with a wider selection of drone models as well as more drones of the same model. The vibration profile of some drone models may be similar (e.g., DJI and Autel EVO), so we need to understand whether DroneScale's efficacy holds across a broader set of models. Also, we need to understand whether there are enough differences within the same model, perhaps due to manufacturing variations, to affect DroneScale's performance. Towards this end, our initial experiments found a similar vibration profile and classification results of two drones of the same model (Bebop 2). Second, the proposed system detects load from a single drone at a given time. Third, due to the impact of COVID-19 lockdowns, we

could not test the system in a larger area and with weights other than precision weights. However, since DroneScale is largely based on frequency modulation, which is highly resilient to the noise over long distances, we are hopeful that our system will perform well at substantially longer distances. Fourth, DroneScale current hardware and software only work with unmodified drones with load detection resolution of 100g. Building a system that can estimate load of modified UAV is our next logical step, which requires our significant investigation to answer these questions. (1) which components can we modify from a COTS drone? From our experience, removing the camera is feasible with DJI P4, but it is impossible with other drones like Bebop 2, Anafi, EVO, and DJI Mavic 2. (2) What are the universal signatures across drones when they are modified? (3) How can we build a transfer learning model that can learn vibration characteristics from a brand new drone, and it is still usable to modified drones?. Fifth, DroneScale only can work with trained drones. However, a single drone can generate different vibrations signatures depending on the flight conditions, and there is no theoretical model that can calculate the vibrations from the drone architecture. This is, in fact, an interesting research direction where one can estimate how much vibration a drone can generate giving its specific propeller length, size, materials, architecture, rotor speeds and so on. In addition, the current system only identifies drone spectrum from 2.4GHz to 2.5GHz. This can be improved by using state of the art spectrum sensing systems such as SweepSense [37]. We also do not consider fixed wing UAV classification in this paper.

10 CONCLUSION

This paper presents a novel technique to detect and estimate a drone's load using passive RF sensing. We discuss a new RF hardware system that is able to monitor the drone signal with 1Hz resolution, which can track a few Hz frequency variation created by drone vibration. We conduct in-lab and field experiments to validate the system using 6 unmodified-drones of 5 different models. DroneScale estimates drone load with 96.27% average accuracy with 10-fold cross validation. DroneScale is also able to monitor drone load accurately at distance up to 200m, on autonomous flight mode, and with different load placement. Finally, as DroneScale can track physical motion with high resolution, its hardware and software design can be extended for other domain applications such as tracking micro vibration of industrial machines, monitoring human vital-signs with high-precision for healthcare applications, or precise spectrum detection in cognitive radio applications.

12

REFERENCES

[1] 9news. CU Boulder police investigating drone that flew over Folsom Field during game. https://www.denverpost.com/2017/10/19/drone-over-folsom-field/. October 19, 2017.

[2] 9news. How Many Drones Are Smuggling Drugs Across the U.S. Southern Border? https://www.airspacemag.com/flight-today/narcodrones-180974934/. July 02, 2019.

[3] AJC News. Georgia man admits he tried to use drone to drop pot into prison. https://tinyurl.com/ybaft7k4. July 01, 2020.

[4] C. Aker and S. Kalkan. Using deep networks for drone detection. In *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6. IEEE, 2017.

[5] Amazon. Precision Steel Calibration Weight. https://tinyurl.com/ydfulrac. July 01, 2020.

[6] Analog Dialogue. Understanding and Eliminating 1/f Noise. https://tinyurl.com/yb8h52d8. July 02, 2019.

[7] M. Z. Anwar, Z. Kaleem, and A. Jamalipour. Machine learning inspired sound-based amateur drone detection for public safety applications. *IEEE Transactions on Vehicular Technology*, 68(3):2526–2534, 2019.

[8] ARDRONIS. This Counter-Drone System Will Safely Hijack and Capture Rogue Drones. https://tinyurl.com/yapeqh22. July 01, 2020.

[9] AUDS. Anti-UAV Defense System (AUDS). https://tinyurl.com/ybqckm6v. July 01, 2020.

[10] Autel Robotics. Autel Robotics EVO. https://auteldrones.com/products/evo. July 01, 2020.

[11] A. Bernardini, F. Mangiatordi, E. Pallotti, and L. Capodiferro. Drone detection by acoustic signature identification. *Electronic Imaging*, 2017(10):60–64, 2017.

[12] B. Brannon and A. Barlow. Aperture uncertainty and adc system performance. *Applications Note AN-501. Analog Devices, Inc.(September)*, 2000.

[13] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.

[14] J. Busset, F. Perrodin, P. Wellig, B. Ott, K. Heutschi, T. Rühl, and T. Nussbaumer. Detection and tracking of drones using advanced acoustic cameras. In *Unmanned/Unattended Sensors and Sensor Networks XI; and Advanced Free-Space Optical Communication Techniques and Applications*, volume 9647, page 96470F. International Society for Optics and Photonics, 2015.

[15] G. Cai, B. M. Chen, K. Peng, M. Dong, and T. H. Lee. Modeling and control system design for a uav helicopter. In *2006 14th Mediterranean Conference on Control and Automation*, pages 1–6. IEEE, 2006.

[16] G. Cai, B. M. Chen, K. Peng, M. Dong, and T. H. Lee. Modeling and control of the yaw channel of a uav helicopter. *IEEE transactions on industrial electronics*, 55(9):3426–3434, 2008.

[17] E. E. Case, A. M. Zelnio, and B. D. Rigling. Low-cost acoustic array for small uav detection and tracking. In *2008 IEEE National Aerospace and Electronics Conference*, pages 110–113. IEEE, 2008.

[18] L. Chen. *Curse of Dimensionality*, pages 545–546. Springer US, Boston, MA, 2009.

[19] F. Christnacher, S. Hengy, M. Laurenzis, A. Matwyschuk, P. Naz, S. Schertzer, and G. Schmitt. Optical and acoustical uav detection. In *Electro-Optical Remote Sensing X*, volume 9988, page 99880B. International Society for Optics and Photonics, 2016.

[20] M. Clifford. Fundamental principles behind the sigma-delta adc topology: Part 1. *Technical Article for Analog Devices, Inc*, pages 1–4, 2016.

[21] A. Coluccia, M. Ghenescu, T. Piatrik, G. De Cubber, A. Schumann, L. Sommer, J. Klatte, T. Schuchert, J. Beyerer, M. Farhadi, et al. Drone-vs-bird detection challenge at ieee avss2017. In *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6. IEEE, 2017.

[22] H. M. Corporation. The impact of clock generator performance on data converters.

[23] Daily News. Mother-daughter team busted for using drone to smuggle contraband into prison. https://tinyurl.com/yavv4chn. July 01, 2020.

[24] Dedrone. Dedrone: Complete Airspace Security Solution. https://www.dedrone.com/. July 01, 2020.

[25] Dedrone. Worldwide Drone Incidents. https://tinyurl.com/y9d8vfh3. July 01, 2020.

[26] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

[27] I. C. Dikmen, A. Arisoy, and H. Temeltas. Attitude control of a quadrotor. In *2009 4th International Conference on Recent Advances in Space Technologies*, June 2009.

[28] DJI. DJI Mavic 2 Zoom. https://tinyurl.com/y98ev3q5. July 01, 2020.

[29] DJI. DJI Phantom 4 Pro. https://www.dji.com/phantom-4-pro. July 01, 2020.

[30] DroneDetector. Protect Yourself From Air, Ground, and Water-Based Threats. http://dronedetector.com/. July 01, 2020.

[31] DroneShield. DroneShield: Detect. Assess. Respond. https://www.droneshield.com/. July 01, 2020.

[32] J. Drozdowicz, M. Wielgo, P. Samczynski, K. Kulpa, J. Krzonkalla, M. Mordzonek, M. Bryl, and Z. Jakielaszek. 35 ghz fmcw drone detection system. In *2016 17th International Radar Symposium (IRS)*, pages 1–4. IEEE, 2016.

[33] FAA. FAA: Drones and Weapons, A Dangerous Mix. https://tinyurl.com/y8c227s5. July 01, 2020.

[34] FalconShield. Falcon Shield: Automatic radio-controlled drone identification solution. https://tinyurl.com/yaeed9gk. July 01, 2020.

[35] Fintan Corrigan. How Do Drones Work And What Is Drone Technology. https://www.dronezon.com/learn-about-drones-quadcopters/what-is-drone-technology-or-how-does-drone-technology-work/. July 02, 2019.

[36] S. R. Ganti and Y. Kim. Implementation of detection and tracking mechanism for small uas. In *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1254–1260. IEEE, 2016.

[37] Y. Guddeti, R. Subbaraman, M. Khazraee, A. Schulman, and D. Bharadia. SweepSense: Sensing 5 GHz in 5 Milliseconds with Low-cost Radios. pages 317–330, 2019.

[38] L. Hauzenberger and E. Holmberg Ohlsson. Drone detection using audio analysis. 2015.

[39] A. Herschfelt, C. R. Birtcher, R. M. Gutierrez, Y. Rong, H. Yu, C. A. Balanis, and D. W. Bliss. Consumer-grade drone radar cross-section and micro-doppler phenomenology. In *2017 IEEE Radar Conference (RadarConf)*, pages 0981–0985. IEEE, 2017.

[40] F. Hoffmann, M. Ritchie, F. Fioranelli, A. Charlish, and H. Griffiths. Micro-doppler based detection and tracking of uavs with multistatic radar. In *2016 IEEE Radar Conference (RadarConf)*, pages 1–6. IEEE, 2016.

[41] G. Hoffmann, H. Huang, S. Waslander, and C. Tomlin. Quadrotor helicopter flight dynamics and control: Theory and experiment. In *AIAA guidance, navigation and control conference and exhibit*, page 6461, 2007.

[42] A. Hommes, A. Shoykhetbrod, D. Noetel, S. Stanko, M. Laurenzis, S. Hengy, and F. Christnacher. Detection of acoustic, electro-optical and radar signatures of small unmanned aerial vehicles. In *Target and Background Signatures II*, volume 9997, page 999701. International Society for Optics and Photonics, 2016.

[43] J. Hua, Z. Xiong, J. Lowey, E. Suh, and E. R. Dougherty. Optimal number of features as a function of sample size for various classification rules. *Bioinformatics*, 21(8):1509–1515, 2005.

[44] Ian Beavers. Noise Spectral Density: A New ADC Metric? https://www.analog.com/en/technical-articles/noise-spectral-density.html.

[45] S. Jeon, J.-W. Shin, Y.-J. Lee, W.-H. Kim, Y. Kwon, and H.-Y. Yang. Empirical study of drone sound detection in real-life environment with deep neural networks. In *2017 25th European Signal Processing Conference (EUSIPCO)*, pages 1858–1862. IEEE, 2017.

[46] W. Kester. Which adc architecture is right for your application. In *EDA Tech Forum*, volume 2, pages 22–25, 2005.

[47] E. Kuantama, O. G. Moldovan, I. Țarcă, T. Vesselényi, and R. Țarcă. Analysis of quadcopter propeller vibration based on laser vibrometer. *Journal of Low Frequency Noise, Vibration and Active Control*, page 1461348419866292, 2019.

[48] D. Lee and M. Lee. Low flicker noise, odd-phase master lo active mixer using a low switching frequency scheme. *IEEE Journal of Solid-State Circuits*, 50(10):2281–2293, 2015.

[49] C. Liang, N. Cao, X. Lu, and Y. Ye. Uav detection using continuous wave radar. pages 1–5, 09 2018.

[50] H. Liu, Z. Wei, Y. Chen, J. Pan, L. Lin, and Y. Ren. Drone detection based on an audio-assisted camera array. In *2017 IEEE Third International Conference on Multimedia Big Data (BigMM)*, pages 402–406. IEEE, 2017.

[51] LORD. IMU MicroStrain LORD 3DM-GX5-25. http://tinyurl.com/yxm8bngy. July 01, 2020.

[52] F. M. Mahmoud and M. Abdelghany. Low flicker-noise rf cmos gilbert-cell mixer for 2.4 ghz wireless communication systems. In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, pages 1158–1161. IEEE, 2016.

[53] R. Mahony, V. Kumar, and P. Corke. Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. *IEEE Robotics and Automation magazine*, 19(3):20–32, 2012.

[54] J. Mezei and A. Molnár. Drone sound detection by correlation. In *2016 IEEE 11th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, pages 509–518. IEEE, 2016.

[55] P. J. Moreno, P. P. Ho, and N. Vasconcelos. A kullback-leibler divergence based kernel for svm classification in multimedia applications. In *Advances in neural information processing systems*, pages 1385–1392, 2004.

[56] T. Müller. Robust drone detection for day/night counter-uav with static vis and swir cameras. In *Ground/Air Multisensor Interoperability, Integration, and Networking for Persistent ISR VIII*, volume 10190, page 1019018. International Society for Optics and Photonics, 2017.

[57] News 5. Drone spotted dropping items into Cuyahoga County jail. https://tinyurl.com/y89db3z2. July 01, 2020.

[58] P. Nguyen, T. Kim, J. Miao, D. Hesselius, E. Kenneally, D. Massey, E. Frew, R. Han, and T. Vu. Towards RF-based Localization of a Drone and Its Controller. In *Proceedings of the 5th Workshop on Micro Aerial Vehicle Networks, Systems, and*

13

Applications, DroNet'19, pages 21–26, Seoul, Republic of Korea, June 2019. Association for Computing Machinery.

[59] P. Nguyen, H. Truong, M. Ravindranathan, A. Nguyen, R. Han, and T. Vu. Matthan: Drone presence detection by identifying physical signatures in the drone's rf communication. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, pages 211–224. ACM, 2017.

[60] Parrot. Parrot - ANAFI 4K Quadcopter. https://www.parrot.com/us/drones/anafi. July 01, 2020.

[61] Parrot. Parrot Bebop 2. https://www.parrot.com/us/drones/parrot-bebop-2. July 01, 2020.

[62] P. Petrides, C. Kyrkou, P. Kolios, T. Theocharides, and C. Panayiotou. Towards a holistic performance evaluation framework for drone-based object detection. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1785–1793. IEEE, 2017.

[63] Planet Analog. SIGNAL CHAIN BASICS (Part 104): Understanding noise in ADCs. https://www.planetanalog.com/signal-chain-basics-part-104-understanding-noise-in-adcs/.

[64] G. J. Pottie and W. J. Kaiser. *Principles of embedded networked systems design*. Cambridge University Press, 2005.

[65] Q. Pu, S. Gupta, S. Gollakota, and S. Patel. Whole-home gesture recognition using wireless signals. In *Proceedings of the 19th Annual International Conference on Mobile Computing & Networking*, MobiCom '13, pages 27–38, New York, NY, USA, 2013. Association for Computing Machinery.

[66] S. Rahman and D. A. Robertson. Radar micro-Doppler signatures of drones and birds at K-band and W-band. *Scientific Reports*, 8(1):17396, Nov. 2018. Number: 1 Publisher: Nature Publishing Group.

[67] S. Ramesh, T. Pathier, and J. Han. SoundUAV: Towards Delivery Drone Authentication via Acoustic Noise Fingerprinting. In *Proceedings of the 5th Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*, DroNet'19, Seoul, Republic of Korea, June 2019. Association for Computing Machinery.

[68] M. Ringnér. What is principal component analysis? *Nature biotechnology*, 26(3):303–304, 2008.

[69] M. Ritchie, F. Fioranelli, H. Griffiths, et al. Monostatic and bistatic radar measurements of birds and micro-drone. In *2016 IEEE Radar Conference (RadarConf)*, pages 1–5. IEEE, 2016.

[70] Rong Zheng, Shuwu Zhang, and Bo Xu. Text-independent speaker identification using gmm-ubm and frame level likelihood normalization. In *2004 International Symposium on Chinese Spoken Language Processing*, pages 289–292, 2004.

[71] F. Sabatino. *Quadrotor control: modeling, nonlinearcontrol design, and simulation*. 2015.

[72] A. Schroder, M. Renker, U. Aulenbacher, A. Murk, U. Boniger, R. Oechslin, and P. Wellig. Numerical and experimental radar cross section analysis of the quadrocopter dji phantom 2. In *2015 IEEE Radar Conference*, pages 463–468. IEEE, 2015.

[73] A. Schumann, L. Sommer, J. Klatte, T. Schuchert, and J. Beyerer. Deep cross-domain flying object classification for robust uav detection. In *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6. IEEE, 2017.

[74] Y. Seo, B. Jang, and S. Im. Drone detection using convolutional neural networks with acoustic stft features. In *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6. IEEE, 2018.

[75] X. Shi, C. Yang, W. Xie, C. Liang, Z. Shi, and J. Chen. Anti-Drone System with Multiple Surveillance Technologies: Architecture, Implementation, and Challenges. *IEEE Communications Magazine*, 56(4):68–74, Apr. 2018.

[76] D.-H. Shin, D.-H. Jung, D.-C. Kim, J.-W. Ham, and S.-O. Park. A distributed fmcw radar system based on fiber-optic links for small drone detection. *IEEE Transactions on Instrumentation and Measurement*, 66(2):340–347, 2016.

[77] K. Stasiak, M. Ciesielski, A. Kurowska, and W. Przybysz. A study on using different kinds of continuous-wave radars operating in c-band for drone detection. In *2018 22nd International Microwave and Radar Conference (MIKON)*, pages 521–526. IEEE, 2018.

[78] The New York Times. Gatwick Airport Shut Down by 'Deliberate' Drone Incursions. https://tinyurl.com/y7quw3jc. July 01, 2020.

[79] TJP. Explosive-laden drones targeted precision missile tech in Beirut - report. https://tinyurl.com/y7w7htfl. July 01, 2020.

[80] Tommy Neu. Why your high-speed ADC can never have enough SNR.

[81] U. S. Customs and Border Protection. Yuma Sector Agents Intercept Narcotics Dropped From Drones. https://tinyurl.com/yboplk3o. July 01, 2020.

[82] J. Verbeke and S. Debruyne. Vibration analysis of a uav multirotor frame. In *Proceedings of ISMA 2016 International Conference on Noise and Vibration Engineering*, pages 2401–2409, 2016.

[83] www.droneflyers.com. Basics of Radio Frequencies for FPV Quadcopter Drones. https://tinyurl.com/y75pzn2g. July 02, 2019.

[84] www.northwestgeorgianews.com. Report: Trio planned to use drone to to get tobacco, phones to inmate. https://tinyurl.com/y8hnamtj. July 01, 2020.

[85] www.wpxi.com. Man accused of using drone to drop explosives on ex-girlfriend's property. https://tinyurl.com/y9x6jw2s. July 01, 2020.

14