Foes or Friends: Embracing Ground Effect for Edge Detection on Lightweight Drones

Chenyu Zhao[1]*, Ciyu Ruan[1]*, Jingao Xu[2†], Haoyang Wang[1], Shengbo Wang[1], Jiaqi Li[1], Jirong Zha[1], Zheng Yang[5], Yunhao Liu[5], Xiao-Ping Zhang[1,3,4], Xinlei Chen[1,3,4†]
{zhaocyhi,softword77,xujingao13,wanghaoyang0428,shengbo1.wang,jackielithu,zhajirong,hmilyyz}@gmails.com
yunhao@mails.tsinghua.edu.cn,xiaoping.zhang@sz.tsinghua.edu.cn,chen.xinlei@sz.tsinghua.edu.cn
[1]Shenzhen International Graduate School, Tsinghua University, China
[2]Computer Science Department, Carnegie Mellon University, USA
[3]Pengcheng Laboratory and [4]RISC-V International Open Source Laboratory, China
[5]School of Software, Tsinghua University, China

Abstract

Drone-based rapid and accurate environmental edge detection is highly advantageous for tasks such as disaster relief and autonomous navigation. Current methods, using radar or cameras, raise deployment costs and burden lightweight drones with high computational demands. In this paper, we propose AirTouch, a system that transforms the ground effect from a stability "foe" in traditional flight control views, into a "friend" for accurate and efficient edge detection. Our key insight is that analyzing drone sensor readings and flight commands allows us to detect ground effect changes. Such changes typically indicate the drone flying over an edge, making this information valuable for edge detection. We approach this insight through theoretical analysis, algorithm design, and implementation, fully leveraging the ground effect as a new sensing modality without compromising drone flight stability, thereby achieving accurate and efficient scene edge detection. Extensive evaluations demonstrate that our system achieves a high detection accuracy with mean detection distance errors of 0.051m, outperforming the baseline performance by 86%.

CCS Concepts

• **Computer systems organization** → **Sensors and actuators**; • **Applied computing** → **Aerospace**; • **Computing methodologies** → **Neural networks**.

*These authors contributed equally to this research.
†Xinlei Chen and Jingao Xu are the corresponding authors.

ACM MobiCom '24, November 18–22, 2024, Washington D.C., DC, USA

ACM ISBN 979-8-4007-0489-5/24/09
https://doi.org/10.1145/3636534.3690699

Keywords

Quadrotors, Sensing Modality, Physical Knowledge aided AI

ACM Reference Format:
Chenyu Zhao[1]*, Ciyu Ruan[1]*, Jingao Xu[2†], Haoyang Wang[1], Shengbo Wang[1], Jiaqi Li[1], Jirong Zha[1], Zheng Yang[5], Yunhao Liu[5], Xiao-Ping Zhang[1,3,4], Xinlei Chen[1,3,4†]. 2024. Foes or Friends: Embracing Ground Effect for Edge Detection on Lightweight Drones. In *International Conference On Mobile Computing And Networking (ACM MobiCom '24), November 18–22, 2024, Washington D.C., DC, USA.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3636534.3690699

1 Introduction

Rapid and accurate terrain exploration of unknown areas is crucial for disaster response [1, 2], rescue-and-relief [3, 4], and autonomous navigation [5–7]. A key aspect involves detecting terrain edges, such as sudden changes in height (e.g., steps, cliffs) and variations in ground materials (e.g., water, soil, solid rock). With prior knowledge of these edges, systems can plan paths for people and robots more logically, efficiently, and safely [8–10]. To boost efficiency and cut costs in large-scale edge detection [11], mainstream systems leverage swarms of lightweight drones [12] (a.k.a., UAV) to execute the task collaboratively as they fly and scan the entire scene [13–16].

Existing drone-based edge detection solutions fall into two categories: (*i*) *wireless-signal based methods* leverage Wi-Fi [17], UWB [18], mmWave or terahertz (THz) radar [19–21], LiDAR [22], and acoustic signals [23], etc., to detect edges by analyzing signal ToF or phase changes [24]. These require sophisticated hardware, increasing costs and limiting widespread usage. (*ii*) *vision-based solutions* employ computer vision techniques or neural networks to identify edges using RGB or RGB-D images [25, 26]. While accurate, they demand substantial computational power, restricting their usage on resource-constrained lightweight drones [27, 28]. Additionally, visual sensors struggle in low-light or high-dynamic situations, compromising their robustness.

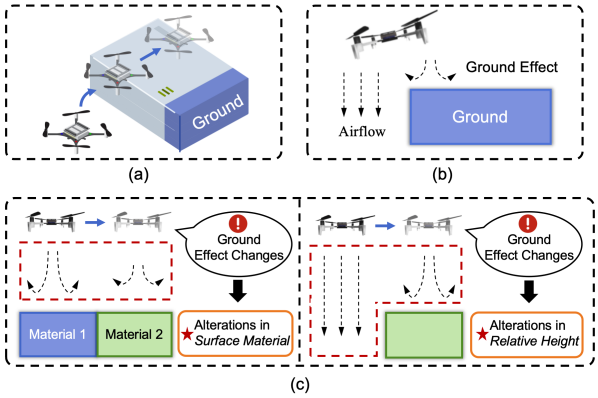


Figure 1: (a) Drone utilizes ground effect for edge detection. (b) When drones fly close to the ground, the increased upward lift is caused by the ground effect. (c) By detecting variations in ground effect, we can infer abrupt changes in height or alterations in the surface material, thereby identifying scene boundaries.

In this work, we aim to introduce a novel approach for edge detection on lightweight drones, which will serve as a significant complement to the aforementioned two envelopes, especially in situations where computational resources are scarce and scene visibility is low. As illustrated in Fig.1, when drones fly close to the ground (i.e., a surface): the airflow generated by the rotating rotors bounces off the surface below the drone, creating additional upward lift and leading to disturbances in a drone's flight. This phenomenon, widely known as **ground effect** (GE) [29, 30], varies with the drone's altitude above the surface and the type of surface material.

The key insight behind this work is to translate the *ground effect* into a fresh *sensing modality* for edge detection on lightweight drones - as shown in Fig.1(b), by identifying ground effect changes, we can deduce sudden alterations in the drone's relative height above a surface or in the surface material itself, pinpointing the edges within the scene. The process has similarities with sensing the surface by touching it using air. However, translating this insight into a practical system still faces two challenges:

• **The target discrepancy between sensing and flight control complicates ground effect profiling.** For the ground effect, sensing tasks treat it as a *friend*, aiming to detect drone flight instability through abrupt changes in sensor readings (e.g., Inertial Measurement Unit (IMU) samples) to measure it. However, drone flight control systems view it as a *foe*, striving to minimize its impact on flight stability. This leads to sensor (e.g., IMU) readings being extensively smoothed out after those complex proportional–integral–derivative (PID) operations [31], challenging the effective profiling of the ground effect.

• **The noisy sensing data overwhelms the vital feedback related to the ground effect.** The dynamic measurements from inexpensive, low-power sensors on lightweight drones make it difficult to extract actual ground-effect-related fluctuations from those complex noises in raw data. The circumstance is further complicated by the flight control system's smoothing and attenuating functions on the ground effect.

Remark. Under the premise that the flight control module treats the ground effect as a *foe* and tries to negate its impact on drone stability, accurately and efficiently measuring and profiling the attenuated ground effect from noisy sensor data is crucial for edge detection.

To tackle the above challenges, we design and implement **AirTouch**, the first system that treats the ground effect as a *friend* and offers methods to extract related data from onboard sensors, despite the flight control module's influence. Benefiting from AirTouch, the ground effect can be leveraged as a new sensing modality for tasks such as edge detection. In general, AirTouch excels in the following three aspects.

• **On the sensory input front.** We demonstrate that leveraging the onboard IMU readings and motor commands from flight controllers could effectively profile the ground effect. By examining the complex physical dynamics and drone stability control, we uncover how drone attitudes (i.e., measured by the IMU) and control signals (i.e., indicated by motor commands) interrelate and complement each other. This combination offers a full insight into the ground effect, even with the flight control module's adjustments.

• **On the algorithm front.** We propose a ground effect-informed environmental edge detection pipeline, which comprises (*i*) a fluctuation components feature extraction method and a cascaded cross-spectrum feature fusion technique to facilitate the extraction of ground-effect-related information from noisy IMU measurements and motor commands; (*ii*) a compact neural network (NN) designed to detect edges from the extracted features; and (*iii*) an aerodynamics-instructed physical filter to further enhance edge detection accuracy.

• **On the implementation front.** To further boost computational efficiency and enable lightweight drones to run the proposed NN in real-time, we apply techniques such as neural unit pruning and weight quantization on the NN before deployment. Additionally, during the NN training, we introduce a meticulously designed Disturbance Force-Informed loss function, incorporating binary cross-entropy loss, to expedite network convergence and make the network learn fine-grained bias.

We evaluate the performance of AirTouch by conducting extensive experiments and comparing it with the baseline using a real-world testbed. Based on a lightweight drone and its onboard IMU and motors, we conducted abrupt height discontinuity edge detection and material interface transition edge detection, respectively. The results demonstrate that

our system achieves a high detection accuracy with mean detection distance errors of 0.051m. Furthermore, our system surpasses the baseline performance by 86% with the same available sensor information. Note that AirTouch is open-source on GitHub[1].

The main contributions of this paper are as follows:

- We propose AirTouch, as far as we are aware, the first system that translates the traditionally negative ground effect into a new, positive sensing modality for accurate and efficient environmental edge detection.
- We demonstrate that combining IMU sampling and motor commands provides an effective sensing paradigm to characterize the ground effect under the influence of the flight control system. On this basis, we present a comprehensive neural network-based pipeline for profiling extracting, and utilizing the ground effect for sensing tasks from noisy sensory input.
- We develop a prototype system and evaluate the AirTouch system through real-world data and in-field experiments on a lightweight drone by deploying our system on onboard computing chips. Extensive evaluation results show the effectiveness of our system in impressive edge detection accuracy on low-cost drones and low-energy consumption sensors.

The remainder of this paper is structured as follows: §2 presents the core intuition and prerequisites underlying the AirTouch system. After introducing the system overview in §3, we elaborate on the two main components of the system in §4 and §5. In §6, we introduce the implementation. In §7, we evaluate our system. In §8 and §9, we have a discussion section and a related work section. In the last, we conclude this paper in §10.

2 Core intuitions and primers

2.1 Ground Effect and Edge Detection

AirTouch is rooted in the features of the ground effect for a new sensing modality to detect the edge. When the drone closely sweeps past a surface, it suffers an extra airflow rebounded by the surface beneath its body. The drone's flight attitude fluctuates due to the airflow caused by ground effect. While the ground effect is considered unwanted in a drone's routine flight, we leverage this physical phenomenon as exceptional feedback aids the sensing procedure. Moreover, the variant surfaces have different attributes for the rebounded airflow, such as the reflection direction and absorption intensity. Therefore, the change in the drone's state under effect contains the discrepancy of variant materials, and the boundary of different levels of ground effect depicts the edges.

[1] https://github.com/ChenyuZhaoTHU/AirTouch

2.2 Aerodynamics of Quadrotor-like drones

As a flight agent, the drone has states containing global position $\mathbf{p} = [p_x, p_y, p_z]^\top \in \mathbb{R}^3$, velocity $\mathbf{v} \in \mathbb{R}^3$, body angular velocity $\boldsymbol{\omega} \in \mathbb{R}^3$ and attitude rotation matrix $R \in SO(3)$. Then, we can describe the following dynamics:

$$\begin{aligned} m\mathbf{a} &= m\mathbf{g} + R\mathbf{f_u} + \mathbf{f_w}, \\ \mathbf{a} &= \dot{\mathbf{v}}, \mathbf{v} = \dot{\mathbf{p}}, \\ \mathbf{J}\dot{\boldsymbol{\omega}} &= \mathbf{J}\boldsymbol{\omega} \times \boldsymbol{\omega} + \boldsymbol{\tau_u} + \boldsymbol{\tau_w}, \\ \dot{R} &= RM(\boldsymbol{\omega}), \end{aligned} \tag{1}$$

where $\mathbf{a}$ is the acceleration of the drone's movement, m and $\mathbf{g} = [0,\ 0,\ -g]$ are mass and gravity acceleration vector, respectively. $M(\cdot)$ indicates skew-symmetric mapping. $\mathbf{f_u} = [0, 0, \Psi]^\top$ and $\mathbf{f_w}$ are the forces from four rotors thrust and unknown disturbance force, respectively. To simplify the formulating, here we assume that there is no natural wind, so the only unknown disturbance force comes from the ground effect. $\boldsymbol{\phi} = [\Psi, \tau_{u,x}, \tau_{u,y}, \tau_{u,z}]^\top$ denotes the output wrench, which determines the control of quadrotors. $\mathbf{u} = [n_1^2, n_2^2, n_3^2, n_4^2]^\top$ are actuation signal, while n_1, n_2, n_3, n_4 are motor rotation speeds. Accordingly, $\boldsymbol{\tau_u} = [\tau_{u,x}, \tau_{u,y}, \tau_{u,z}]^\top$ and $\boldsymbol{\tau_w}$ are the torques from four rotors and disturbance. The thrust Ψ can be derived from $\boldsymbol{\phi} = H_0\mathbf{u}$, with

$$H_0 = \begin{bmatrix} k_T & k_T & k_T & k_T \\ 0 & k_T l_r & 0 & -k_T l_r \\ -k_T l_r & 0 & k_T l_r & 0 \\ -c_Q & c_Q & -c_Q & c_Q \end{bmatrix}, \tag{2}$$

where k_T is thrust coefficient and l_r is the length of rotor arm, and c_Q represents torque coefficient. The critical factors in the drone stable flight problem are $\mathbf{f_w}$ and $\boldsymbol{\tau_a}$ from the unknown wind disturbance from GE. Disturbance force $\mathbf{f_w} = [f_{w,x}, f_{w,y}, f_{w,z}]^\top$ and disturbance torques $\boldsymbol{\tau_a} = [\tau_{w,x}, \tau_{w,y}, \tau_{w,z}]^\top$ comes from complicated aerodynamics interactions between quadrotors and environment, especially to the ground. In general, the larger the rotor output power or the closer the drone is to the ground, the more effect intensity the disturbance will have [32].

3 System Overview

From the top perspective, we design and deploy the AirTouch system to accurately detect the edges of different surfaces. The system captures implicit features from noisy raw data with minor and highly dynamic attributes. As Fig.2 illustrated, AirTouch excludes non-relevant components by extracting distinct characteristics of edges and fusing multiple features using *Fluctuation Component Extraction* (§4.2) and *Cascaded Cross-Spectrum Feature Fusion* (§4.3) at first. Then *Aerodynamics-Informed Double Phase Physical Filter* (§5.1) conducts two functionalities. The first is alleviating

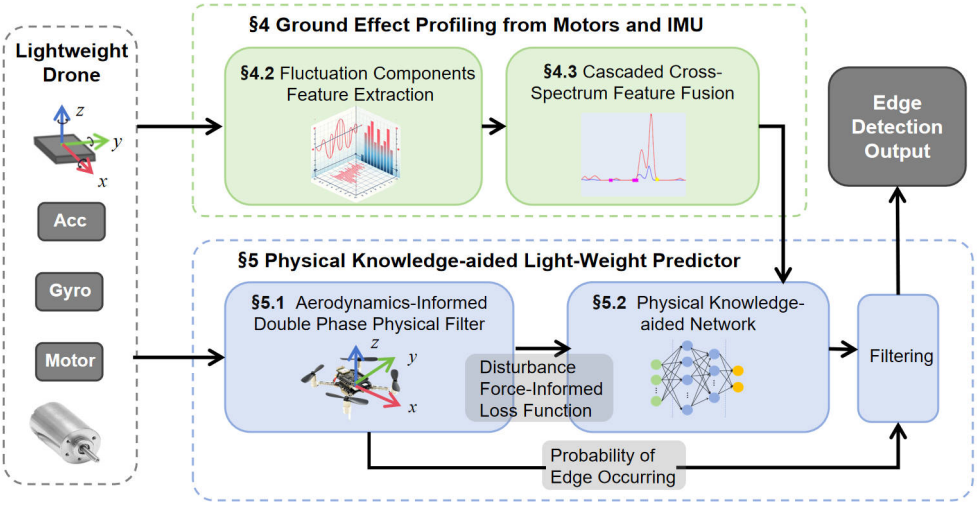


Figure 2: System Overview. The system is designed based on a physical-informed neural network to capture the discrepancy of the drone's state under the impact of the ground effect. It is a new sensing modality with physical phenomena and the proprioceptive sensing method.

noisy data distraction for precision improvement with Disturbance Force-Informed Loss in *Physical Knowledge-aided Network*(§5.2). The second is filtering out false-prone edge detection from the network output. Then, the final output can be transformed into edge detection results.

4 Ground Effect Profiling from Motors and IMU

Instead of relying on additional sensors, AirTouch utilizes a novel proprioceptive sensing method to detect different edges by ground effect. It is non-trivial to profile the ground effect to leverage this sensing method to distinguish the edge. Therefore, we decided to use IMU and motor signals for the tasks. *Combination of Two Modalities* (§4.1) elaborates the reason by delving into the intricate physical phenomena and the flight controller working pipeline of the drone to examine its precise impact thoroughly. Then, the next challenge is how to represent the different attributes of the ground effect. A concise and interpretable representation of data captures the essential characteristics of minor GE and will enhance the detection result. Aiming at this, we proposed two submodules: *Fluctuation Components Feature Extraction* (FC-FE) (§4.2) and *Cascaded Cross-Spectrum Feature Fusion* (CCS-FF) (§4.3).

4.1 Combination of Two Modalities

Why both IMU and motors? To sense the ground effect on drones, one intuitive approach is to utilize an IMU to monitor the drone's attitude changes. However, relying solely on

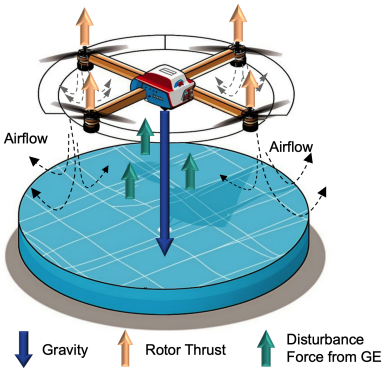


Figure 3: The forces exerted on the drone under ground effect. Although forces along other axes also exist, here only the principle change components of forces, along the z-axis, are shown.

a single modality of IMU data lacks universality from the perspective of practical engineering applications. A drone is equipped with a flight controller (FC) and one of its functions is attitude stability control. FC monitors and adjusts the drone's attitude in response to flight commands and environmental changes to maintain balance and stability. A sudden attitude change and imbalance will be captured by the IMU sensor and tentatively eliminated by adjusting the motor speeds [33]. Different FCs exhibit varying performances of resisting the ground effect. For instance, with high control precision and fast control loop frequencies, even under the ground effect, the attitude may not undergo significant changes, resulting in subtle signal variations in IMUs.

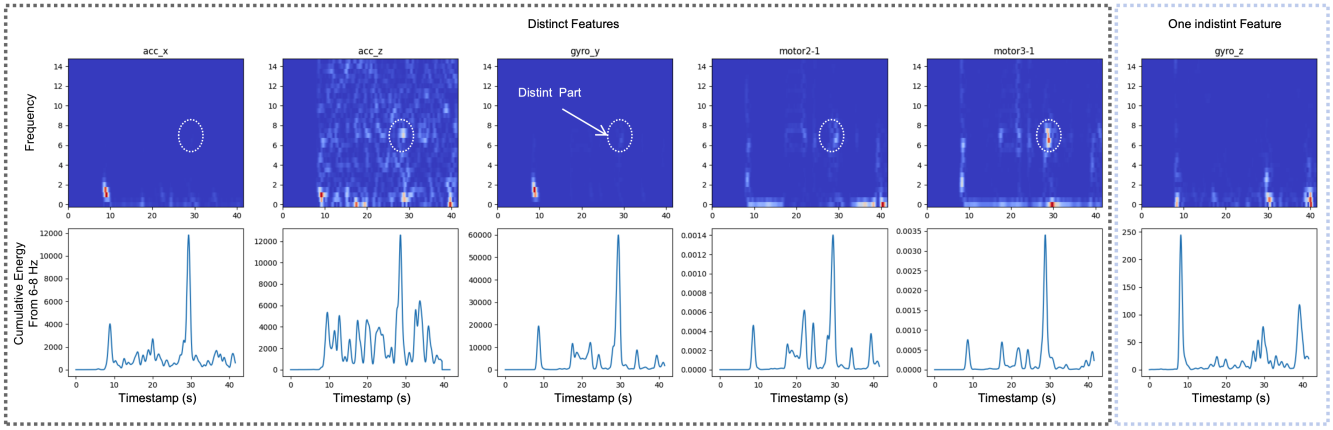


Figure 4: The left five plots show the distinct features, and the right one is an example of an indistinct feature. From the heatmaps of the power spectrum, the target periods in the left features coherently have distinct power increases between 6-8 Hz, while it may not be distinct or stable among other features.

Thus, motor signals and IMU's attitude information **are coupled and complementary to each other**, together contributing to the whole effect of the ground effect. To ensure that our ground effect-based detection method supports drones with different flight controllers, we profile the ground effect using both IMU and motor modalities from both a physical force analysis perspective and an input-data characteristic perspective.

From a physical force analysis perspective, we derive the disturbance force $\mathbf{f_w}$ as the direct impact measurement of the ground effect. As is shown in Fig.3, we exclude the gravity and thrust force from net force using Newton's second law:

$$\begin{gathered} \mathbf{f_w} = m\mathbf{a} - m\mathbf{g} - R\mathbf{f_u}, \\ \Psi = H_0 W(\mathbf{m}),\ \dot{R} = RM(\boldsymbol{\omega}), \\ M(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -gyro_z & gyro_y \\ gyro_z & 0 & -gyro_x \\ -gyro_y & gyro_x & 0 \end{bmatrix}, \end{gathered} \tag{3}$$

where $W(\cdot)$ is the transform mapping from motors' PWM signals $\mathbf{m} = [m_1, m_2, m_3, m_4]^\top$ to actuation signal $\mathbf{u}$. Apart from system parameters, data sources needed in the above derivation depend on $\mathbf{a}$ and $\boldsymbol{\omega}$ measured by IMU and $\mathbf{m}$ sent to motors.

Additionally, considering the data characteristic, two modalities contribute to data-driven edge detection. From the feature extraction results introduced later, we discover that the influence of the drone attitude and motor signals has dynamicity, which means the intensity of the ground effect varies across different detection trials. For this concern, redundancy in data sources could reduce the detection failure rate. At the same time, IMU and motors are delayed and feedback to each other. Therefore, to enable a comprehensive understanding of the drone's attitude and empower edge detection performance, our system contains the thorough ground effect's features, by leveraging both motor signals and IMU signals as the source data.

What is data dimension? The attitude change induced by GE is captured by the flight data of the drone, including the 3-axis acceleration rate $\mathbf{a} = [acc_x, acc_y, acc_z]^\top$ from the acceleration measurement, 3-axis angular acceleration rate $\boldsymbol{\omega} = [gyro_x, gyro_y, gyro_z]^\top$ from the gyroscope, and four motors' PWM signals $\mathbf{m} = [m_1, m_2, m_3, m_4]^\top$ calculated from command signals. Here, $\mathbf{m}$ is positively proportional to the motors' rotation speeds $[n_1, n_2, n_3, n_4]^\top$. Fig.8 illustrates a set of the raw data collected by the drone.

4.2 Fluctuation Components Feature Extraction (FC-FE)

We proposed the FC-FE algorithm, a frequency spectrum analysis-based method to extract features of high sensitivity and accuracy. It is based on the fluctuation of the attitude and motor signals. The temporal data source is $\mathbf{d} = [acc_x, acc_y, acc_z, gyro_x, gyro_y, gyro_z, m_{2-1}, m_{3-1}, m_{4-1}]^\top$. For normalization, m_{2-1}, m_{3-1}, and m_{4-1} are the differences between other three motors and m_1. Note that, for the convenience of explanation, we **take the height edge detection as the example case** in §4 and §5. We first examine the heat maps of power amplitude versus time to explore the useful frequency components over time in all 9 feature categories, which are transformed by Short-Time Fourier Transform (STFT) [34]. We analyze the sum power amplitude component P within a distinct frequency band F. P satisfies the fact that it is distinguishing at the edge. As illustrated in the left five heat maps in Fig.4, the power amplitude from 6 to 8 Hz is always distinctly stronger when the drone sweeps past the edge in

most data categories. The consistency lies in the rigid contact of IMU sensors and motors during the drone fluctuation. This particular power cluster reflects the attitude oscillation of the drone under the ground effect, which is determined by many factors, such as the drone's weight, size, propeller size, performance, and bias of the flight controller, etc. Theoretically, such frequency range differs from different drone types and our tests with larger drones coincide with the observation. The frequency range can be easily collected and configured on the system.

We calculate the cumulative power amplitude within F at each timestamp and take the results as features to distinguish the edge. Therefore, the smaller the temporal and frequency resolutions are, the better the performance of real-time capability and detection precision the system will have. Note that the resolutions of frequency and temporality correlate to the window size and overlap size of STFT.

4.3 Cascaded Cross-Spectrum Feature Fusion (CCS-FF)

We design Cascaded Cross-Spectrum to fuse multiple features of each sensor into one synthetic feature, and then three synthetic features for acceleration rate, angular rate, and motor signal will be generated. Inspired by the cross-spectrum, which is a method to assess the correlation between the components of two signals at the same frequency, CCS-FF extracts the correlation within one sensor's different dimensions to represent a comprehensive characteristic.

Before introducing CCS-FF, we first introduce the cross-spectrum [35]. If there is a strong correlation between the two signals at a particular frequency, the amplitude of the cross-spectrum at the corresponding frequency will be large. $G_{xy}(f)$ is the cross-spectrum of two signals $x(t)$ and $y(t)$, whose frequency domain represents $X(f)$ and $Y(f)$. The magnitude component $|G_{xy}(f)|$ quantifies the degree of correlation between the two signals at the frequency f. $G_{xy}(f)$ is defined as the product of $X(f)$ and the conjugate of $Y(f)$, as shown in the following Eq.4,

$$G_{xy}(f) = X(f)Y^*(f). \tag{4}$$

Meanwhile, as a signal's power spectral density, a fundamental concept used to characterize the power distribution of a signal across various frequencies in the frequency domain can be expressed as Eq.5; thus, the relationship between $|G_{xy}(f)|$, $G_{xx}(f)$ and $G_{yy}(f)$ satisfies the relationship as defined in Eq.6.

$$G_{xx}(f) = X(f)X^*(f). \tag{5}$$

$$|G_{xy}(f)|^2 = |G_{xx}(f)||G_{yy}(f)|. \tag{6}$$

To this end, the correlation between signals can be calculated using the product of their power spectral densities. Extending this concept to more than two signals involves cascading their cross-spectrum. For n signals with frequency domain representations $X_1(f)$,$X_2(f)$,...$X_n(f)$, we define the cascaded cross-spectrum $P_{ccs}(f)$ and use its magnitude to measure the degree of correlation between the n signals, which can be represented as if n is an even number,

$$\begin{aligned} |P_{ccs}(f)| &= |X_1(f)X_2^*(f)X_3(f)...X_n^*(f)| \\ &= \sqrt{|G_{x_1x_1}(f)||G_{x_2x_2}(f)|...|G_{x_nx_n}(f)|}. \end{aligned} \tag{7}$$

Note that if n is an odd number, the last term of the first row should be $X_n(f)$. Based on the aforementioned derivation, the power spectral densities of the extracted features are multiplied to serve as the synthetic fusion feature providing correlation information to represent the comprehensive characteristic. To be specific, 9-dimensional temporal data $\mathbf{d}$ has been transformed into 3-dimensional synthetic NN input data $\mathbf{c} = [\mathbf{a_s}, \boldsymbol{\omega_s}, \mathbf{m_s}]^\top$.

5 Physical Knowledge-aided Light-Weight Predictor

To facilitate the drone's real-time and accurate acquisition of edge information, we employed a lightweight neural networks (NN) model for edge prediction. As discussed in the previous section, the coupling of motor and IMU data challenges the precise capture of the ground effect. Compared to traditional methods, NN can better capture the spatio-temporal relationship between drone sensor fluctuations and ground effect. By modeling the ground effect, extracting and fusing features in §4, we obtained the input for the NN (§5.2). Moreover, we utilized the *Aerodynamics-Informed Double Phase Physical Filter* proposed in §5.1 to guide the network in learning data more efficiently. Given the limited computing and storage resources of the drone, we also conducted lightweight processing on the model in §5.3.

5.1 Aerodynamics-Informed Double Phase Physical Filter

Aerodynamics-Informed Double Phase Physical Filter consists of two functionalities in two phases. One functionality is that it provides a fine-grained bias serving as a physical knowledge-based loss item in the NN training phase for alleviating noisy data distraction, to improve detection precision. The other one is that it minimizes the occurrence probability of false detection caused by edge-irrelevant disturbances in the global phase. The core intuition is leveraging the inherent disturbance force on the drone, hidden in the distinction between ground effects separated by the edge. Utilizing *Fast Responded Constant False Alarm Rate Algorithm* (FR-CFAR), the enlightenment of the extra force change could be detected.

Inspired by the aerodynamics analysis introduced in §2.2, we leverage the unknown disturbance force $\mathbf{f_w}$ exerted on

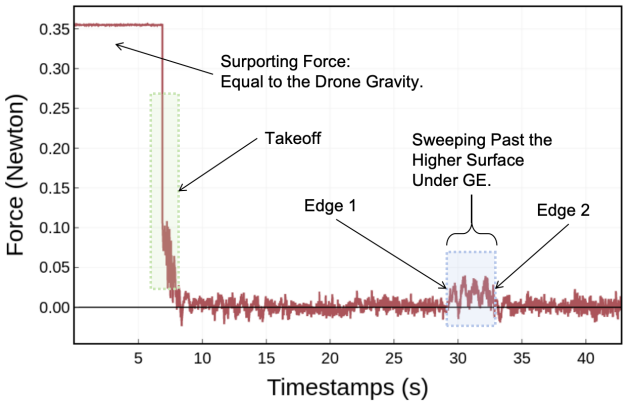


Figure 5: Disturbance force along the z-axis in height edge detection in Aerodynamics-Informed Double Phase Physical Filter. The magnitude of extra upward lift $f_{w,z}$ is the data source of the physical filter. The convex part of the force represents that the drone suffers an extra upward lift due to the ground effect.

the drone body to get a glimpse of the height edge detection process. The $\mathbf{f_w} = [f_{w,x}, f_{w,y}, f_{w,z}]^\top$ is induced by the extra disturbance of the rebound airflow solely and here we only focus on the upward force $f_{w,z}$ because the distinct component of force change is along z axis in height edge detection. It is similar in cases of material edge detection, where the distinct change of force may occur along other axes. $\mathbf{f_w}$ can be derived from the acceleration rate $\mathbf{a}$ and the rotation speeds of four motors n_1, n_2, n_3, and n_4.

As is shown in Fig.5, the force in the z axis is plotted over the process of idle, takeoff, hover, and traveling across the platform. In the beginning, the $f_{w,z}$ is equal to the supporting force from the ground, whose value is the gravity of the drone. Then, when the drone is hovering or flying in the air, this extra force is around zero. Only when the drone is sweeping over the platform, $f_{w,z}$ have significant increasing peaks roughly from 29.37 to 32.81 seconds in the plot, roughly consistent with the ground truth.

To apply physical embedding for the neural network and exclude the false edge detection caused by data noise or non-relevant flight operation, a strategy is necessary to capture the mutation of the disturbance. However, limited by the environmental parameter accuracy and sensor precision, the f_w measurement may have offset and noise. This makes it impossible to notice the mutation with a simple constant threshold. Therefore, we propose the Fast-Respond Constant False Alarm Rate method to adjust the threshold and recognize the force mutation.

Fast-Respond Constant False Alarm Rate (FR-CFAR). As is illustrated in Fig.6, we propose an FR-CFAR algorithm based on procedures of the Cell-Averaging Constant False Alarm Rate (CA-CFAR) to select the target points, which provides a constant false detection probability [36]. FR-CFAR detects target signals in background noise while ensuring a constant probability of false positives, and also has instant detection ability. The difference between a normal CA-CFAR and our FR-CFAR is the omission of the lagging window, which requires queue time for the following signal that has not been generated yet. Therefore, without waiting for the lagging window data, FR-CFAR can respond with the detection result instantly.

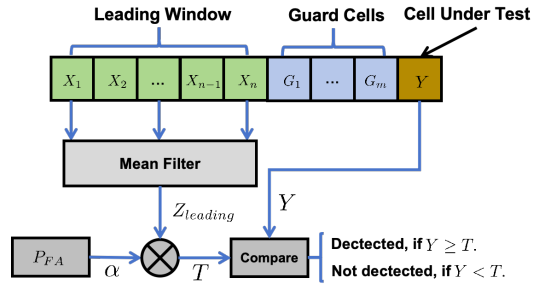


Figure 6: The main procedure of Fast Responded Constant False Alarm Rate (FR-CFAR) algorithm.

The key scheme is that the algorithm proposes a threshold level calculated by estimating the noise floor level around the judged signal sample, cell under test (CUT). If the signal sample's magnitude exceeds the threshold, the CUT Y is considered the target. A group of cells surrounding the Cell Under Test (CUT) is selected to determine this, and the average power level is calculated. $[G_1, ..., G_m]$, the cells directly adjacent to the CUT, known as "guard cells", are typically excluded from this calculation to prevent interference from the CUT itself, thus preventing a more clean estimate from adjacent interference. The detection threshold T satisfies the following relationships:

$$\begin{aligned} Z_{leading} &= \frac{1}{N}\sum_{i=1}^{N} X_i, \\ T &= \alpha \cdot Z_{leading}, \\ \alpha &= N * (P_{FA}^{-\frac{1}{N}} - 1), \end{aligned} \tag{8}$$

where $Z_{leading}$ is the mean signal value of the N-cell leading window to estimate the noise level, X_i denotes the signal received in the i th cell in the leading window, and threshold factor α is adjusted to maintain the false alarm rate at a predetermined probability P_{FA} [37]. Once the CUT Y exceeds the dynamic detection threshold T, the disturbance force alerts a high probability of edge occurring. Correspondingly, this filter improves detection precision for NN training in the first physical embedding phase by alleviating noisy data distraction. For the second phase, the final NN Predictor detection without alerts from disturbance force will be filtered and ignored.

5.2 Physical Knowledge-aided Network

The input of our model consists of three-dimensional fused features denoted as $\mathbf{c} = [\mathbf{a_s}, \boldsymbol{\omega_s}, \mathbf{m_s}]^\top$ within a specific time window, resulting in an input size equal to three times the window size. The model is designed for binary classification, where the detection of edges is accomplished by identifying transitions between the two encoded categories.

To address the specific requirements of edge detection and energy conservation in our task, we employ a standard NN architecture. This architecture incorporates convolutional layers followed by max-pooling layers, which serve to extract relevant features from the input. To introduce non-linearity and enhance the model's capability to learn complex patterns in the data, Rectified Linear Unit (ReLU) activation functions are applied after the convolutional layers. The output layer of the model consists of two units with softmax activation, enabling the model to generate class probabilities for each category.

Disturbance Force-Informed (DF) Loss Function. The incorporation of a physics-informed loss function aims to enhance the model's comprehension of ground effect dynamics. When a drone encounters an edge, a previously unknown disturbance force undergoes an abrupt change, serving as a distinctive indicator for edge detection. By integrating this force in conjunction with a threshold, our physics-embedded methodology effectively captures relevant features associated with ground effects. This approach improves the precision of edge detection and yields valuable insights into the flight environment of the drone.

For instance, considering a drone traversing the edge of an abrupt height change, a state variable switches between 1 and 0 depending on the surface. The edge marks the transition between these states, with distinct disturbance forces experienced on each surface. The disturbance force-informed loss function employed is computed as follows:

$$\mathcal{L}_F = \begin{cases} e^{|\hat{y}-1|} - 1, & ||f_w|| \geqslant T. \\ e^{|\hat{y}|} - 1, & \text{others}\ . \end{cases}, \tag{9}$$

where $\mathcal{L}_F$ denotes the physics-informed loss function, which is designed to measure the model's prediction error for disturbance forces. The $\hat{y}$ represents the predicted state, and the T is the threshold used to determine if the disturbance force f_w exceeds a critical value. The disturbance force f_w and T can be calculated in §5.1.

For this binary classification problem, we utilize the binary cross-entropy as our fundamental loss function:

$$\mathcal{L}_S = -y \log \hat{y} - (1 - y) \log(1 - \hat{y}), \tag{10}$$

where y is the sample's actual label (0 or 1), and $\hat{y}$ is the predicted probability value output by the model.

Considering the two loss functions, the overall loss function for edge detection is:

$$\mathcal{L} = \mathcal{L}_S + \lambda \cdot \mathcal{L}_F, \tag{11}$$

where λ is the coefficient that weighs the contribution of the disturbance force-informed loss function and can be adjusted easily in different applications.

5.3 Pruning and Quantization

For the lightweight design of our model, we employed pruning and quantization to reduce complexity, enhance efficiency, and improve performance, making it well-suited for resource-constrained platforms.

Weight pruning is a technique aimed at reducing the computational complexity of neural networks by eliminating redundant connections. In our pruning procedure, weights are systematically pruned based on an adjustable threshold. Specifically, we first sort the absolute values of all weights in ascending order to obtain a vector. We then calculated the threshold, denoted as θ, used to prune weights based on their magnitude and dynamically adjusted during training. This systematic approach, in which weights are either retained or set to zero based on their magnitude compared to the threshold, enables effective weight pruning and leads to model compression.

Quantization is the establishment of a mapping relationship between floating-point data and fixed-point data. We denote floating-point real numbers as r and quantized fixed-point integers as q. The conversion formula between floating-point and integer is given by:

$$r = S(q - Z),\ q = \text{round}(r \times S) + Z, \tag{12}$$

where S is the scale factor representing the proportionality between real and integer values, and Z is the zero point representing the integer equivalent of 0 in real numbers. The *round* function rounds the result of $r \times S$ to the nearest integer.

After weight pruning and quantization, the lightweight CNN model is obtained with a model compression ratio:

$$CR = \frac{S_o}{S_c} = \frac{32}{(1 - s_f) \times b} \tag{13}$$

where the original model size is S_o, S_c is the size of the model after pruning and quantization, the pruning sparsity level is s_f, and the number of quantization bits is b.

6 Implementation

To experimentally validate our system for precise edge detection, we selected the Crazyflie 2.1 nano-quadrotor, including an STM32F405 MCU and a 250mAh LiPo battery. It weighs 35.6 grams and offers 22.65 GOPS of processing power and 512KB of RAM. Additionally, our test drone is equipped with

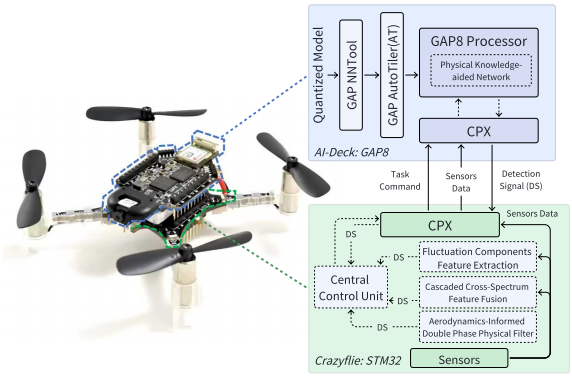


Figure 7: Implementation framework and modules on the devices.

an AI deck, a Multi-ranger deck, and a Flow deck. The AI deck, equipped with a GAP8 microcontroller, enhances the drone's memory and computational capabilities, which is crucial for implementing our NN model. The Multi-ranger deck detects objects around the Crazyflie, while the Flow deck tracks the drone's movements.

We implemented the *Fluctuation Components Feature Extraction* (§4.2), *Cascaded Cross-Spectrum Feature Fusion* (§4.3) and *Aerodynamics-Informed Double Phase Physical Filter* (§5.1) on the STM32 and then fed the fusion data into our *Physical Knowledge-aided Network* (§5.2) on the AI deck, as is shown in Fig.7. The AI deck communicates with the STM32 on the Crazyflie to obtain the fusion data for the actual model inference implementation.

7 Evaluation

To this end, we implement a prototype of AirTouch and perform experiments with different scenes and parameters. We first introduce the experimental setup and evaluate the overall performance of our system. Next, we present two case studies demonstrating our system's effectiveness and accuracy in detecting different types of edges. Additionally, we explore the system's resilience and evaluate specific components' contributions. Finally, we measured the model's weight size, inference speed, and energy consumption to demonstrate the efficiency of our system.

7.1 Experiment Setup

Parameter Setup and Dataset: We measured Crazyflie's mass m, propeller radius D, the distance between rotor axes d, thrust constant C_T, and air pressure ρ. The drone is shown in Fig.10(a). Also, we tested the thrust constant k_T from the real world using the relationship $k_T = C_T \rho D^4$. The data sampling rate is 100Hz for acceleration rate, angular acceleration rate, and motor signals. For STFT, we set the window size and overlap size to 199. In FR-CFAR, we configure the following

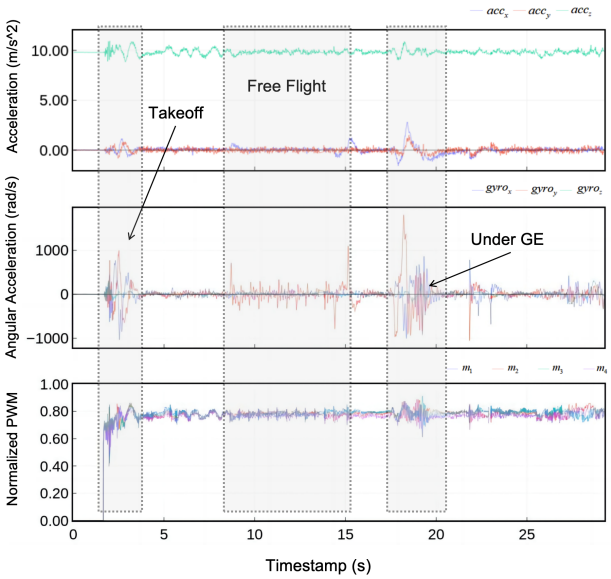


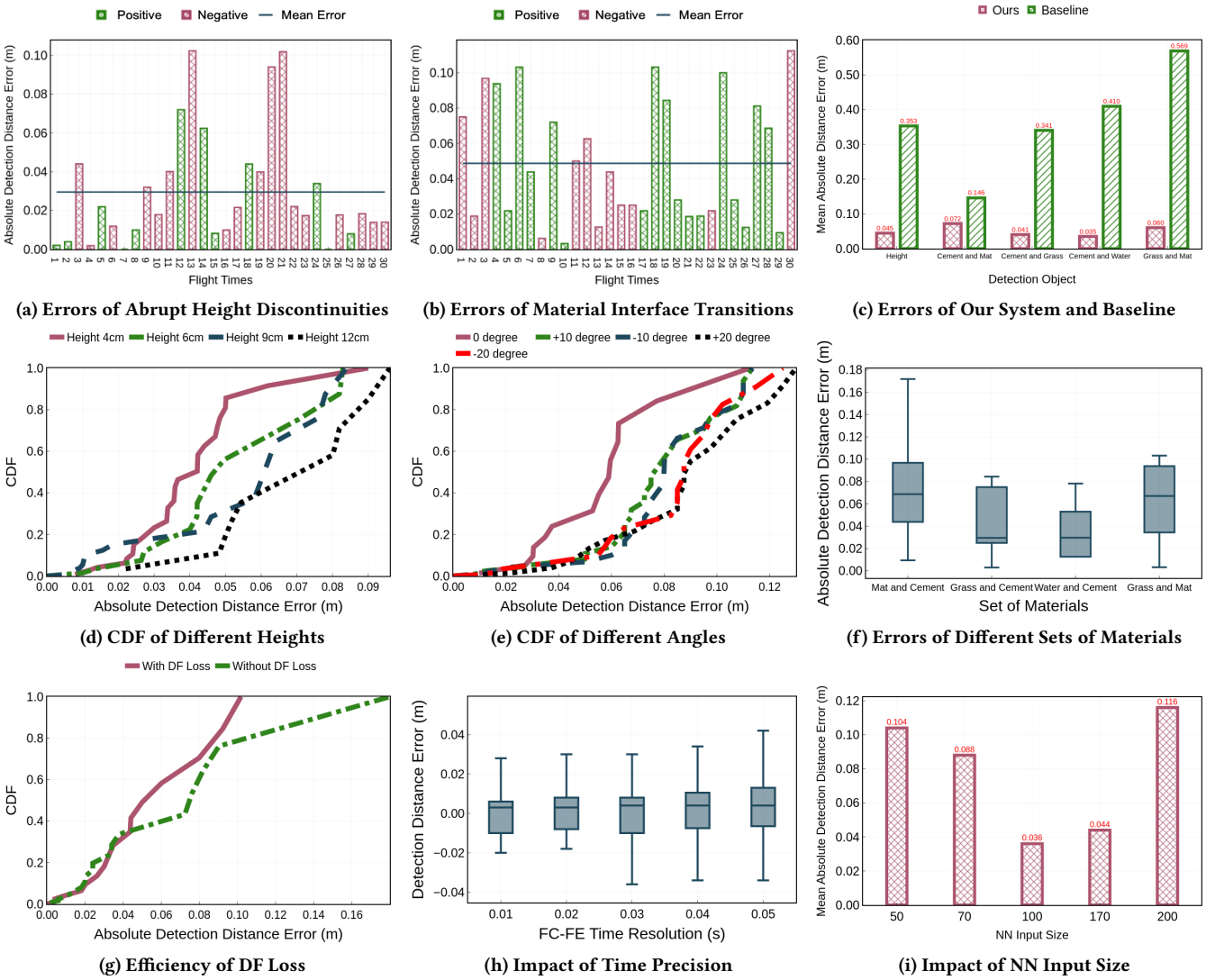
Figure 8: The raw data used in the system. The data contains the acceleration rates, angular acceleration rates, and the motor PWM signals from takeoff, normal flight, to sweeping past the platform surface.

parameters: $P_{FA} = 10^{-6}$, leading window size of 50, and guard cell size of 15. For the dataset, we arranged the testbed of the height discontinuity platform, which is a drone landing platform, and several surfaces of different materials to collect the training dataset. We collect 200 samples for height and 300 for material edge detection.

Network Training: The proposed network is implemented in TensorFlow. We trained the models for 200 epochs with a fixed batch size of 32. The input window size is 100, corresponding to a total sequence length of 100 timestamps. The output represents the probability of the drone either reaching or not reaching an edge and is encoded using one-hot encoding, resulting in an output size of 2.

Evaluation Metrics and Ground Truth: The performance of our system is evaluated using the **detection distance error**, which measures the discrepancy between the detected edge and the actual edge. The distance error is calculated by converting the time error of detection using the drone's velocity. We align the ground truth of the precise edge location with the IMU and motor signals by calculating the distance between the drone and the landmark representing the edge. This distance is recorded by the drone's multi-range deck.

Baseline: We tested our system against a low computational cost frequency spectrum transform method. It first transforms the raw data using STFT and accumulates the power at every timestamp. Then, it calculates the sum of correlations using Pearson Correlation Coefficient [3], and the target detected edge is at the point when the second-order derivative is equaled to zero.



(a) Errors of Abrupt Height Discontinuities (b) Errors of Material Interface Transitions (c) Errors of Our System and Baseline

(d) CDF of Different Heights (e) CDF of Different Angles (f) Errors of Different Sets of Materials

(g) Efficiency of DF Loss (h) Impact of Time Precision (i) Impact of NN Input Size

Figure 9: Performance. (a) The absolute detection errors in 30 standard flights for the height edge detection. (b) The absolute detection errors in 30 standard flights for the material edge detection. (c) Comparison with the baseline on different materials. (d) The detection errors with different heights to the surface. (e) The detection errors with different angles to the edge. (f) The comparison among different sets of materials. (g) The efficiency of DF Loss Function. (h) Influence of Time Resolution. (i) The performance influence of NN input layer size.

7.2 Overall Performance

To evaluate the system's overall performance, we examined the accuracy of edge detection for abrupt height discontinuities and material interface transitions across 30 test groups. Fig.9(a) and 9(b) depict the edge detection distance errors for the cases. The positive values stand for the early detection before the drone exactly passes above the edge, while negative values present the lagging detection. Notably, abrupt height discontinuities can be regarded as one form of material interface transition, where one side of the surface is air. After analysis, the mean absolute detection distance error for abrupt height discontinuities was 0.0295m, while it was 0.0362m for material interface transitions.

We compare the edge detection errors of our method with the baseline. Our method outperforms the baseline in all sets of detection. The mean absolute error is 0.051 m, much lower than the baseline at 0.364m, with 86% performance improvement. This is attributed to the characteristics of the ground effect refined by our system's design, so without profiling the proper characteristics the baseline can only achieve coarse detection.

7.3 Case Study: Abrupt Height Discontinuities

When testing the edge detection performance for abrupt height discontinuities, we employed a platform setup constructed from an acrylic surface platform as illustrated in Fig.10(b), over which the drone flew horizontally. Due to the platform's elevation above the ground, the moment when the drone flies over the platform represents a scenario of abrupt height discontinuity at the edge. Furthermore, we tested the system's edge detection robustness by varying the drone's height above the ground and the angle at which it approached the edge during horizontal flight.

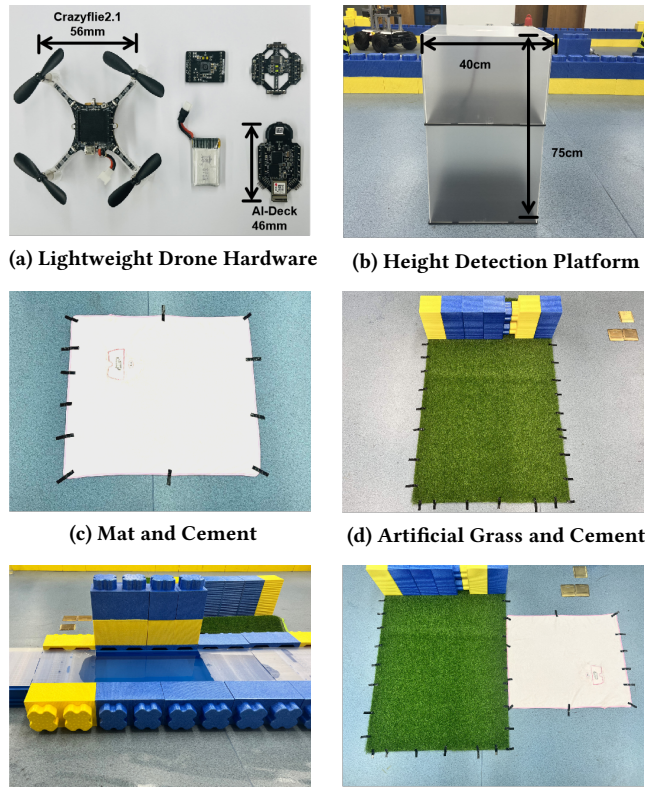
The height is defined as the vertical distance above the drone to the platform surface when sweeping past it. As depicted in Fig.9(d), detection accuracy decreases along with the height increases. At a height of 4cm, nearly 80% of the errors fall within 0.05m, while with the heights of 9cm and 12cm, less than 30% of the tests fall within this range. This trend is attributed to the increased altitude, which weakens the ground effect and the disturbance. The weaker the effect brings to the drone, the more challenging it is for detection.

We also explored the influence of the transition angle to the edge. The horizontal angle is defined as the angle between the drone's horizontal flight path and the horizontal normal to the edge of the platform. Fig.9(e) illustrates that detection accuracy decreases along with the angle increases. This may be because the asymmetry of the angle enhances the dynamic drone attitude, with obscure features. The overlap between the curves for +10 and -10 degrees, as well as +20 and -20 degrees, suggests that the magnitude of the horizontal angle significantly impacts the error, not the horizontal direction, due to the symmetric design of the drone.

7.4 Case Study: Material Interface Transitions

We conducted four sets of experiments to test the edge detection performance for Material Interface Transitions. These included edges between "mat" and "cement", "artificial grass" and "cement", "water" and "cement", and "artificial grass" and "mat" (Fig.10(c)-(f)). According to Fig.9(f), the edge detection error is the largest and most widely ranged for the "mat" and "cement" edges, with a median error of 0.070 m and a maximum error of 0.170 m. It is slightly lower but relatively stable for the "artificial grass" and "cement" edges. The "water" and "cement" edges exhibit the smallest and most concentrated detection errors. Approximately 75% of the data falls within the range of 0.01 m to 0.05 m. The edge detection error between "artificial grass" and "mat" falls between that of "mat" and "cement" and that of "artificial grass" and "cement".

The surface natures of different materials generate different ground effects in terms of intensity, frequency, and directional components, leading to different drone behaviors. If the discrepancy is more obvious, the detection will be relatively more accurate because it is easier to capture the edge-induced change (with higher SNR) and vice versa. For example, the artificial grass surface's pronounced texture and varying height compared to the smooth cement surface contribute to this distinction in ground effect, making the edge more discernible for the artificial grass.



(a) Lightweight Drone Hardware (b) Height Detection Platform (c) Mat and Cement (d) Artificial Grass and Cement (e) Water and Cement (f) Artificial Grass and Mat

Figure 10: Hardware and testbeds of experiments for dataset collection and infield experiments.

7.5 Impact of System Components

Effectiveness of DF Loss Function: To evaluate the effectiveness of the Disturbance Force-Informed (DF) Loss Function, we conducted a comparative analysis between models trained with and without the DF loss. As depicted in Fig.9(i), the introduction of the DF loss lead to a significant reduction in distance detection errors, with the majority of errors decreasing from 0.18 m to within 0.10 m. This improvement demonstrates a substantial accuracy improvement, reaching 44% compared to the model trained without the DF loss.

FC-FE Time Resolution: We also investigated the impact of time resolution in the Fluctuation Components Feature Extraction (FC-FE) process. The resolution can be adjusted

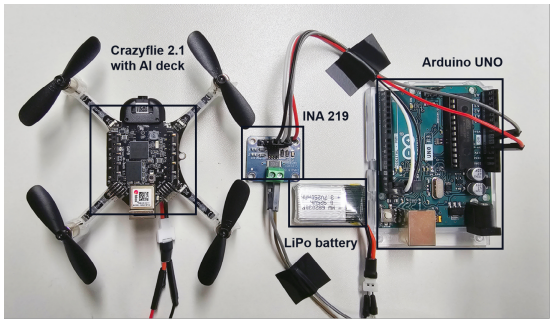


Figure 11: Power consumption measurement Setup using INA 219 and Arduino.

by modifying the window size and overlap size. As depicted in Fig.9(h), higher resolutions enhanced the detection performance, resulting in lower distance errors.

NN Input Window Size: To assess the influence of different input window sizes, we conducted experiments on a standardized dataset. As illustrated in Fig.9(i), the input size ranging from 100 to 170 exhibited the best results. Considering the trade-off between model performance and computational complexity, we chose a window size of 100 as it strikes a balance, maintaining satisfactory performance levels.

7.6 Model Optimization and Efficiency

After pruning and quantizing our NN model, we observed a reduction in model size from 141837 Bytes to 14375 Bytes. Additionally, the inference time was reduced from 0.00271s to 0.00110s. However, the model's accuracy on the training and test sets decreased from 97% to 86%. Notably, the accuracy reduction is deemed acceptable the model size has been reduced by nearly 10-fold, while the inference speed has increased by approximately 2.5 times.

To demonstrate the performance and energy efficiency of AirTouch, we conducted power consumption experiments by deploying our modules on Crazyflie's STM32 and the AI deck's GAP8 processor. As shown in Fig.11, we utilized the INA 219 and Arduino to measure the power consumption of the nano-quadrotor with AI deck during the deployment of our system on board. The INA 219 is a current sensor that provides accurate current and voltage measurements.

Table.1 shows that the network module consumes the most power, 33 mW, contributing significantly to the total system power consumption of 121 mW. However, the power consumption for driving the drone's propellers during flight is approximately 3000 mW. Therefore, the energy consumption of our proposed system is negligible.

8 Discussion

In this section, we provide explanations of some concerns.

Table 1: System power consumption

Module	Power (mW)
Basic drone module power	78
Fluctuation Components	6
Feature Extraction	
Cascaded Cross-Spectrum	2
Feature Fusion	
Aerodynamics-Informed Double	2
Phase Physical Filter	
Physical Knowledge-aided Network	33
Total Power	121

8.1 Sensing Range

In §7.3, the results show the sensing effectiveness degraded significantly above 12 cm. In fact, the sensing range is extendable with larger, heavier drones. From the theoretical aspect, according to recent studies of aerodynamics and ground effect [29, 38], larger drones, longer propellers, and heavier weights generate more intense airflows and GE, even when flying at the same altitude. As a result, the GE will dissipate over a greater distance for detection. Also from the observation in our prior experiments, a drone weighing 2.5 kg with a diagonal span of 450 mm and propellers of 100 mm could detect GE from approximately 3 meters above, showing the possibility of sensing range extension.

8.2 Wind Conditions

AirTouch works well without relying on a completely windless environment. On the evaluation front, the performance results in §7 do not completely exclude the winds although the system is tested indoors. In our experiments, varying wind conditions were created by frequently opening doors, windows, and using air conditioners. Therefore, the consistent performance results reported in this paper could demonstrate the system's robustness. On the technical front, AirTouch focuses on sudden changes in airflow rather than its absolute strength to detect edges. Relative steady winds, without abrupt changes during the instant moment over an edge, do not significantly impact the system's performance.

8.3 Comparison With Vision

AirTouch is a compact and complementary solution for edge detection and it shows various apparent advantages over vision-based methods. **First, in terms of resource efficiency**, AirTouch demonstrates a significant reduction in computational and memory requirements. According to our tests, it requires only 0.28 million FLOPs and 13KB of neural network parameters, amounting to 0.007% of the FLOPs and 22% of the parameter size found in state-of-the-art lightweight vision neural networks (i.e., FLOPs: 3900 million, parameter size: 58 KB) [39]. Although a low-resolution camera (320x320 grayscale) processes millions of bits per frame,

AirTouch uses only 32kbits—substantially less. Additionally, our use of advanced techniques like factorized optimization and group convolution holds promise for further reducing its resource footprint. **Second, regarding capability**, AirTouch is designed to complement, not replace, existing approaches. Visual modalities often fail in low-light conditions or when materials have similar colors and textures. In contrast, AirTouch can identify edges based on distinct physical properties, making it a valuable addition or even an emergency backup in cases of critical system failures.

8.4 Complex Edge Scenarios

For now, AirTouch can effectively detect quadrilateral edges but may encounter limitations with more complex shapes or edges between surfaces of similar natures. To address these challenges, future work may focus on several key areas. For irregular shape detection, employing advanced flight path planning techniques can improve sensing trajectories and facilitate better detection of varied geometries of targets. Besides, upgrading edge detection algorithms by training models on diverse datasets of intricate shapes also contributes to detection compatibility. For scenarios involving edges of similar materials, integrating supplementary sensing modalities like depth sensors or cameras, can provide additional data and improve the detection success rate.

9 Related Work

We review the most related works in this section.

Terrain exploration. Terrain exploration holds significant importance across various fields and activities, encompassing everything from scientific research to disaster response [40, 41]. In disaster management, whether responding to natural disasters like earthquakes[16, 42], floods[43], or wildfires [44], or human-made disasters like industrial accidents [45], terrain knowledge is crucial for rescue and relief operations [46, 47]. Autonomous robots also rely on terrain understanding for efficient and safe navigation [48]. Analyzing terrain features such as slopes, obstacles, and uneven surfaces enables effective path planning and obstacle avoidance [49, 50]. One critical aspect of terrain exploration entails the detection of terrain edges, encompassing a range of features such as abrupt changes in elevation like steps and cliffs, as well as variations in ground materials, including water bodies, soil compositions, or solid rock formations.

Drone-based edge detection. To enhance efficiency and reduce costs in large-scale edge detection, prevalent systems employ swarms of lightweight drones [14, 51]. These drones operate collaboratively, scanning the entire scene to achieve comprehensive coverage [52], with more flexibility compared with ground vehicles [53, 54]. However, current drone-based edge detection methods face challenges, categorized into two main types: (*i*) Wireless-signal based methods: These methods utilize signals like mmWave Radar [1], Wi-Fi [55], terahertz radar [20], LiDAR [56], and acoustic signals [57] for contactless edge detection. While effective, they often rely on infrastructure and struggle with operations in inaccessible areas [58]. (*ii*) Visual-sensor based methods: These algorithms aim to accurately detect object boundaries using computer vision techniques or neural networks[59–61]. However, they require significant computational resources, limiting their deployment on resource-constrained drones [62]. Furthermore, existing pioneer study [63] proposes multi-modal fusion sensing, achieving promising accuracy by integrating the information from both wireless and visual domains.

Ground effect of drones. Achieving precise control over drone positions is paramount, yet remains a considerable challenge [64, 65]. This challenge is predominantly attributed to the intricate interplay between rotor and wing airflows with the ground surface [66, 67]. The aerospace industry has recognized the ground effect for some time, acknowledging its potential to amplify lift forces while decreasing aerodynamic drag [68, 69]. Despite advantages, they also pose challenges to flight stability [70]. Consequently, mitigating the impacts of ground effect has been a persistent issue [71]. In contrast, this paper diverges from conventional approaches by harnessing the ground effect to detect edges rather than attempting to neutralize it. As far as we are aware, it is the first system to perform edge detection without the use of additional sensors.

10 Conclusion

In conclusion, this paper introduces AirTouch, a proprioceptive sensing system that transforms the traditionally negative ground effect into a positive sensing modality for environmental edge detection. By combining IMU and motor signals, AirTouch successfully captures and analyzes the ground effect, providing a novel and effective approach for precise and efficient sensing tasks. The presented system exhibits notable performance in edge detection accuracy on lightweight drones while considering energy consumption and limited computational resources. These achievements highlight the system's potential to enhance the capabilities of drones across diverse environments. Future research will concentrate on extending edge detection capabilities to mobile platforms, facilitating air-ground coordination and collaboration. Also, we will extend the AirTouch with multimodal fusion techniques to fully utilize its potential. As an initial work, this new sensing modality could benefit the community and inspire further research into its varied applications.

11 Acknowledgement

This paper was supported by the National Key R&D program of China (2022YFC3300703), the Natural Science Foundation of China under Grant 62371269, Shenzhen 2022 Stabilization Support Program (WDZC20220811103500001) and Meituan.

References

[1] Emerson Sie, Zikun Liu, and Deepak Vasisht. Batmobility: Towards flying without seeing for autonomous drones. In *Proceedings of the 29th ACM MobiCom*, pages 1–16, 2023.

[2] Xuecheng Chen, Zijian Xiao, Yuhan Cheng, ChenChun Hsia, Haoyang Wang, Jingao Xu, Susu Xu, Fan Dang, Xiao-Ping Zhang, Yunhao Liu, et al. Soscheduler: Toward proactive and adaptive wildfire suppression via multi-uav collaborative scheduling. *IEEE Internet of Things Journal*, 2024.

[3] Bin-Bin Zhang, Dongheng Zhang, Ruiyuan Song, Binquan Wang, Yang Hu, and Yan Chen. Rf-search: Searching unconscious victim in smoke scenes with rf-enabled drone. In *Proceedings of the ACM MobiCom*, pages 1–15, 2023.

[4] Chen-Chun Hsia, Yanggang Xu, Jiyuan Ren, and Xinlei Chen. Demo abstract: Carl: Collaborative altitude-adaptive reinforcement learning for active search with uav swarms. In *2024 23rd ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pages 249–250. IEEE, 2024.

[5] Chris Xiaoxuan Lu, Stefano Rosa, Peijun Zhao, Bing Wang, Changhao Chen, John A Stankovic, Niki Trigoni, and Andrew Markham. See through smoke: robust indoor mapping with low-cost mmwave radar. In *Proceedings of the ACM MobiSys*, pages 14–27, 2020.

[6] Chris Xiaoxuan Lu, Muhamad Risqi U Saputra, Peijun Zhao, Yasin Almalioglu, Pedro PB De Gusmao, Changhao Chen, Ke Sun, Niki Trigoni, and Andrew Markham. milliego: single-chip mmwave radar aided egomotion estimation via deep sensor fusion. In *Proceedings of the ACM SenSys*, pages 109–122, 2020.

[7] Yuxuan Liu, Haoyang Wang, Fanhang Man, Jingao Xu, Fan Dang, Yunhao Liu, Xiao-Ping Zhang, and Xinlei Chen. Mobiair: Unleashing sensor mobility for city-scale and fine-grained air-quality monitoring with airbert. In *Proceedings of the 22nd Annual International Conference on Mobile Systems, Applications and Services*, pages 223–236, 2024.

[8] Soon-Jo Chung, Aditya Avinash Paranjape, Philip Dames, Shaojie Shen, and Vijay Kumar. A survey on aerial swarm robotics. *IEEE Transactions on Robotics*, 34(4):837–855, 2018.

[9] Raul Mur-Artal and Juan D Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE transactions on robotics*, 33(5):1255–1262, 2017.

[10] Danyang Li, Yishujie Zhao, Jingao Xu, Shengkai Zhang, Longfei Shangguan, and Zheng Yang. edgeslam 2.0: Rethinking edge-assisted visual slam with on-chip intelligence. In *Proceedings of the IEEE INFOCOM*, 2024.

[11] Zuxin Li, Fanhang Man, Xuecheng Chen, Baining Zhao, Chenye Wu, and Xinlei Chen. Tract: Towards large-scale crowdsensing with high-efficiency swarm path planning. In *Adjunct Proceedings of the 2022 ACM International Joint Conference on Pervasive and Ubiquitous Computing and the 2022 ACM International Symposium on Wearable Computers*, pages 409–414, 2022.

[12] Xinlei Chen, Aveek Purohit, Shijia Pan, Carlos Ruiz, Jun Han, Zheng Sun, Frank Mokaya, Patric Tague, and Pei Zhang. Design experiences in minimalistic flying sensor node platform through sensorfly. *ACM Transactions on Sensor Networks (TOSN)*, 13(4):1–37, 2017.

[13] Jingao Xu, Hao Cao, Zheng Yang, Longfei Shangguan, Jialin Zhang, Xiaowu He, and Yunhao Liu. Swarmmap: Scaling up real-time collaborative visual slam at the edge. In *Proceedings of the USENIX NSDI*, pages 977–993, 2022.

[14] Haoyang Wang, Jingao Xu, Chenyu Zhao, Zihong Lu, Chen Cheng, Yuhan, Xuecheng Chen, Xiao-Ping Zhang, Yunhao Liu, and Xinlei Chen. Transformloc: Transforming mavs into mobile localization infrastructures in heterogeneous swarms. In *In the Proceedings of the IEEE INFOCOM*, 2024.

[15] Xinlei Chen, Carlos Ruiz, Sihan Zeng, Liyao Gao, Aveek Purohit, Stefano Carpin, and Pei Zhang. H-drunkwalk: Collaborative and adaptive navigation for heterogeneous mav swarm. *ACM Transactions on Sensor Networks (TOSN)*, 16(2):1–27, 2020.

[16] Xuecheng Chen, Haoyang Wang, Yuhan Cheng, Haohao Fu, Yuxuan Liu, Fan Dang, Yunhao Liu, Jinqiang Cui, and Xinlei Chen. Ddl: Empowering delivery drones with large-scale urban sensing capability. *IEEE Journal of Selected Topics in Signal Processing*, 2024.

[17] Guoxuan Chi, Zheng Yang, Jingao Xu, Chenshu Wu, Jialin Zhang, Jianzhe Liang, and Yunhao Liu. Wi-drone: Wi-fi-based 6-dof tracking for indoor drone flight control. In *Proceedings of the ACM MobiSys*, pages 56–68, 2022.

[18] Yi Liu, Zhuozhu Jian, Junbo Tan, Lunfei Liang, Houde Liu, and Xinlei Chen. Demo abstract: Range-slam: Uwb based realtime indoor location and mapping. In *2024 23rd ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pages 259–260. IEEE, 2024.

[19] Anfu Zhou, Shaoyuan Yang, Yi Yang, Yuhang Fan, and Huadong Ma. Autonomous environment mapping using commodity millimeter-wave network device. In *Proceedings of the IEEE INFOCOM*, pages 1126–1134. IEEE, 2019.

[20] Sayed Saad Afzal, Atsutse Kludze, Subhajit Karmakar, Ranveer Chandra, and Yasaman Ghasempour. Agritera: Accurate non-invasive fruit ripeness sensing via sub-terahertz wireless signals. In *Proceedings of the 29th ACM MobiCom*, pages 1–15, 2023.

[21] Yan Zhuo, Han Li, Chenlong Wang, and Xinlei Chen. Adaptive chirps domain window order of mm-wave radar for uav motion capture. In *2024 23rd ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pages 291–292. IEEE, 2024.

[22] Xumiao Zhang, Anlan Zhang, Jiachen Sun, Xiao Zhu, Y Ethan Guo, Feng Qian, and Z Morley Mao. Emp: Edge-assisted multi-vehicle perception. In *Proceedings of the 27th ACM MobiCom*, pages 545–558, 2021.

[23] Shirui Cao, Dong Li, Sunghoon Ivan Lee, and Jie Xiong. Powerphone: Unleashing the acoustic sensing capability of smartphones. In *Proceedings of the ACM MobiCom*, pages 1–16, 2023.

[24] Zhiyuan Xie, Xiaomin Ouyang, Li Pan, Wenrui Lu, Guoliang Xing, and Xiaoming Liu. Mozart: A mobile tof system for sensing in the dark through phase manipulation. In *Proceedings of the ACM MobiSys*, pages 163–176, 2023.

[25] Bing Li, Juan Pablo Munoz, Xuejian Rong, Qingtian Chen, Jizhong Xiao, Yingli Tian, Aries Arditi, and Mohammed Yousuf. Vision-based mobile indoor assistive navigation aid for blind people. *IEEE transactions on mobile computing*, 18(3):702–714, 2018.

[26] Hao Cao, Jingao Xu, Danyang Li, Longfei Shangguan, Yunhao Liu, and Zheng Yang. Edge assisted mobile semantic visual slam. *IEEE Transactions on Mobile Computing*, 2022.

[27] Deng-Ping Fan, Zheng Lin, Zhao Zhang, Menglong Zhu, and Ming-Ming Cheng. Rethinking rgb-d salient object detection: Models, data sets, and large-scale benchmarks. *IEEE Transactions on neural networks and learning systems*, 32(5):2075–2089, 2020.

[28] Yu-Huan Wu, Yun Liu, Jun Xu, Jia-Wang Bian, Yu-Chao Gu, and Ming-Ming Cheng. Mobilesal: Extremely efficient rgb-d salient object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):10261–10269, 2021.

[29] Stephen A Conyers, Matthew J Rutherford, and Kimon P Valavanis. An empirical evaluation of ground effect for small-scale rotorcraft. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 1244–1250. IEEE, 2018.

[30] Chenyu Zhao, Ciyu Ruan, Shengbo Wang, Jirong Zha, Haoyang Wang, Jiaqi Li, Yuxuan Liu, Xuzhe Wang, and Xinlei Chen. Demo abstract: Bio-inspired tactile sensing for mav landing with extreme low-cost

sensors. In *2024 23rd ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pages 261–262. IEEE, 2024.

[31] Chenyu Zhao, Haoyang Wang, Jiaqi Li, Fanhang Man, Shilong Mu, Wenbo Ding, Xiao-Ping Zhang, and Xinlei Chen. Smoothlander: A quadrotor landing control system with smooth trajectory guarantee based on reinforcement learning. In *Adjunct Proceedings of the 2023 ACM International Joint Conference on Pervasive and Ubiquitous Computing & the 2023 ACM International Symposium on Wearable Computing*, pages 682–687, 2023.

[32] Md Shah Alam and Jared Oluoch. A survey of safe landing zone detection techniques for autonomous unmanned aerial vehicles (uavs). *Expert Systems with Applications*, 179:115091, 2021.

[33] Stephen Xia, Minghui Zhao, Charuvahan Adhivarahan, Kaiyuan Hou, Yuyang Chen, Jingping Nie, Eugene Wu, Karthik Dantu, and Xiaofan Jiang. Anemoi: A low-cost sensorless indoor drone system for automatic mapping of 3d airflow fields. In *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, pages 1–16, 2023.

[34] Ruxin Wang, Long Huang, and Chen Wang. Distracted driving detection by sensing the hand gripping of the phone. In *Proceedings of the 27th ACM MobiCom*, pages 828–830, 2021.

[35] Douglas J Nelson. Cross-spectral methods for processing speech. *The Journal of the Acoustical Society of America*, 110(5):2575–2592, 2001.

[36] Ahsan Jalil, Hassan Yousaf, and Muhammad Iram Baig. Analysis of cfar techniques. In *2016 13th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, pages 654–659. IEEE, 2016.

[37] Alfonso Farina and Francisco A Studer. A review of cfar detection techniques in radar systems. *Microwave Journal*, 29:115, 1986.

[38] Antonio Matus-Vargas, Gustavo Rodriguez-Gomez, and Jose Martinez-Carranza. Ground effect on rotorcraft unmanned aerial vehicles: A review. *Intelligent Service Robotics*, 14(1):99–118, 2021.

[39] Xavier Soria, Yachuan Li, Mohammad Rouhani, and Angel D Sappa. Tiny and efficient model for the edge detection generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1364–1373, 2023.

[40] Philip Arm, Gabriel Waibel, Jan Preisig, Turcan Tuna, Ruyi Zhou, Valentin Bickel, Gabriela Ligeza, Takahiro Miki, Florian Kehl, Hendrik Kolvenbach, et al. Scientific exploration of challenging planetary analog environments with a team of legged robots. *Science robotics*, 8(80):eade9548, 2023.

[41] Young Been Kim, Shu Yang, and Dae Seok Kim. Sidewinder-inspired self-adjusting, lateral-rolling soft robots for autonomous terrain exploration. *Advanced Science*, page 2308350, 2024.

[42] Tung-Ju Hsieh. Understanding earthquakes with advanced visualization. *ACM SIGGRAPH Computer Graphics*, 44(1):1–13, 2010.

[43] Aaron Lowe, Pankaj K Agarwal, and Mathias Rav. Flood-risk analysis on terrains. *Communications of the ACM*, 63(9):94–102, 2020.

[44] Zhuozhu Jian, Zejia Liu, Haoyu Shao, Xueqian Wang, Xinlei Chen, and Bin Liang. Path generation for wheeled robots autonomous navigation on vegetated terrain. *IEEE Robotics and Automation Letters*, 2023.

[45] Michael Walker, Thao Phung, Tathagata Chakraborti, Tom Williams, and Daniel Szafir. Virtual, augmented, and mixed reality for human-robot interaction: A survey and virtual design element taxonomy. *ACM Transactions on Human-Robot Interaction*, 12(4):1–39, 2023.

[46] Tao Li, Ning Xie, Chunqiu Zeng, Wubai Zhou, Li Zheng, Yexi Jiang, Yimin Yang, Hsin-Yu Ha, Wei Xue, Yue Huang, et al. Data-driven techniques in disaster information management. *ACM Computing Surveys (CSUR)*, 50(1):1–45, 2017.

[47] Weichen Zhang, Yuxuan Liu, Xuzhe Wang, Xuecheng Chen, Chen Gao, and Xinlei Chen. Demo abstract: Embodied aerial agent for city-level visual language navigation using large language model. In *2024 23rd ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pages 265–266. IEEE, 2024.

[48] Xin Zhou, Xiangyong Wen, Zhepei Wang, Yuman Gao, Haojia Li, Qianhao Wang, Tiankai Yang, Haojian Lu, Yanjun Cao, Chao Xu, et al. Swarm of micro flying robots in the wild. *Science Robotics*, 7(66):eabm5954, 2022.

[49] Qingzhao Zhang, Xumiao Zhang, Ruiyang Zhu, Fan Bai, Mohammad Naserian, and Z Morley Mao. Robust real-time multi-vehicle collaboration on asynchronous sensors. In *Proceedings of the 29th ACM MobiCom*, pages 1–15, 2023.

[50] Xuecheng Chen, Haoyang Wang, Zuxin Li, Wenbo Ding, Fan Dang, Chengye Wu, and Xinlei Chen. Deliversense: Efficient delivery drone scheduling for crowdsensing with deep reinforcement learning. In *Adjunct Proceedings of the 2022 ACM International Joint Conference on Pervasive and Ubiquitous Computing and the 2022 ACM International Symposium on Wearable Computers*, pages 403–408, 2022.

[51] Xinlei Chen, Aveek Purohit, Carlos Ruiz Dominguez, Stefano Carpin, and Pei Zhang. Drunkwalk: Collaborative and adaptive planning for navigation of micro-aerial sensor swarms. In *Proceedings of the 13th ACM SenSys*, pages 295–308, 2015.

[52] Haoyang Wang, Xuecheng Chen, Yuhan Cheng, Chenye Wu, Fan Dang, and Xinlei Chen. H-swarmloc: Efficient scheduling for localization of heterogeneous mav swarm with deep reinforcement learning. In *Proceedings of the 20th ACM Sensys*, pages 1148–1154, 2022.

[53] Xinlei Chen, Susu Xu, Jun Han, Haohao Fu, Xidong Pi, Carlee Joe-Wong, Yong Li, Lin Zhang, Hae Young Noh, and Pei Zhang. Pas: Prediction-based actuation system for city-scale ridesharing vehicular mobile crowdsensing. *IEEE Internet of Things Journal*, 7(5):3719–3734, 2020.

[54] Susu Xu, Xinlei Chen, Xidong Pi, Carlee Joe-Wong, Pei Zhang, and Hae Young Noh. ilocus: Incentivizing vehicle mobility to optimize sensing distribution in crowd sensing. *IEEE Transactions on Mobile Computing*, 19(8):1831–1847, 2019.

[55] Yuming Hu, Xiubin Fan, Zhimeng Yin, Feng Qian, Zhe Ji, Yuanchao Shu, Yeqiang Han, Qiang Xu, Jie Liu, and Paramvir Bahl. The wisdom of 1,170 teams: Lessons and experiences from a large indoor localization competition. In *Proceedings of the 29th ACM MobiCom*, pages 1–15, 2023.

[56] Danyang Li, Jingao Xu, Zheng Yang, Qiang Ma, Li Zhang, and Pengpeng Chen. Leovr: Motion-inspired visual-lidar fusion for environment depth estimation. *IEEE Transactions on Mobile Computing*, 2023.

[57] Weiguo Wang, Yuan He, Meng Jin, Yimiao Sun, and Xiuzhen Guo. Metaspeaker: Acoustic source projection by exploiting air nonlinearity. In *Proceedings of the 29th ACM MobiCom*, pages 1–15, 2023.

[58] Weiguo Wang, Luca Mottola, Yuan He, Jinming Li, Yimiao Sun, Shuai Li, Hua Jing, and Yulei Wang. Micnest: Long-range instant acoustic localization of drones in precise landing. In *Proceedings of the 20th ACM SenSys*, pages 504–517, 2022.

[59] Jianzhong He, Shiliang Zhang, Ming Yang, Yanhu Shan, and Tiejun Huang. Bdcn: Bi-directional cascade network for perceptual edge detection. *IEEE transactions on pattern analysis and machine intelligence*, 44(1):100–113, 2020.

[60] Linxi Huan, Nan Xue, Xianwei Zheng, Wei He, Jianya Gong, and Gui-Song Xia. Unmixing convolutional features for crisp edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6602–6609, 2021.

[61] Xavier Soria, Angel Sappa, Patricio Humanante, and Arash Akbarinia. Dense extreme inception network for edge detection. *Pattern Recognition*, 139:109461, 2023.

[62] Jiang-Jiang Liu, Qibin Hou, and Ming-Ming Cheng. Dynamic feature integration for simultaneous detection of salient object, edge, and skeleton. *IEEE Transactions on Image Processing*, 29:8652–8667, 2020.

[63] Shuai Wang, Luoyu Mei, Ruofeng Liu, Wenchao Jiang, Zhimeng Yin, Xianjun Deng, and Tian He. Multi-modal fusion sensing: A comprehensive review of millimeter-wave radar and its integration with other modalities. *IEEE Communications Surveys & Tutorials*, 2024.

[64] Songnan Bai, Qingning He, and Pakpong Chirarattananon. A bioinspired revolving-wing drone with passive attitude stability and efficient hovering flight. *Science Robotics*, 7(66):eabg5913, 2022.

[65] Yunlong Song, Angel Romero, Matthias Müller, Vladlen Koltun, and Davide Scaramuzza. Reaching the limit in autonomous racing: Optimal control versus reinforcement learning. *Science Robotics*, 8(82):eadg1462, 2023.

[66] Michael O'Connell, Guanya Shi, Xichen Shi, Kamyar Azizzadenesheli, Anima Anandkumar, Yisong Yue, and Soon-Jo Chung. Neural-fly enables rapid learning for agile flight in strong winds. *Science Robotics*, 7(66):eabm6597, 2022.

[67] Enrico Ajanic, Mir Feroskhan, Stefano Mintchev, Flavio Noca, and Dario Floreano. Bioinspired wing and tail morphing extends drone flight capabilities. *Science Robotics*, 5(47):eabc2897, 2020.

[68] Kirill V Rozhdestvensky. Wing-in-ground effect vehicles. *Progress in aerospace sciences*, 42(3):211–283, 2006.

[69] Guanya Shi, Xichen Shi, Michael O'Connell, Rose Yu, Kamyar Azizzadenesheli, Animashree Anandkumar, Yisong Yue, and Soon-Jo Chung. Neural lander: Stable drone landing control using learned dynamics. In *2019 international conference on robotics and automation (ICRA)*, pages 9784–9790. IEEE, 2019.

[70] Guanya Shi, Wolfgang Hönig, Yisong Yue, and Soon-Jo Chung. Neural-swarm: Decentralized close-proximity multirotor control using learned interactions. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3241–3247. IEEE, 2020.

[71] Guanya Shi, Wolfgang Hönig, Xichen Shi, Yisong Yue, and Soon-Jo Chung. Neural-swarm2: Planning and control of heterogeneous multirotor swarms using learned interactions. *IEEE Transactions on Robotics*, 38(2):1063–1079, 2021.