Tracking Hazardous Aerial Plumes using IoT-Enabled Drone Swarms

Carl Seiber, David Nowlin, Bob Landowski and Matthew E. Tolentino
Intelligent Platforms & Architecture Lab
University of Washington, Tacoma, WA, USA
{*carls4, dbnowlin, ski0, metolent*} *@uw.edu*

***Abstract*—Emergency response teams are charged with ensuring citizen safety from life-threatening events such as structural fires, vehicle accidents, and hazardous material spills. While managing such events is dangerous, the release of hazardous materials, such as toxic chemicals, into the atmosphere is particularly challenging. Upon arrival at a scene, response teams must quickly identify the hazardous substance and the contaminated area to limit exposure to nearby population centers. For airborne toxins, this assessment is complicated by environmental conditions, such as changes in wind speed and direction that can cause hazardous, aerial plumes to move dynamically. Without a way to dynamically monitor and assess atmospheric conditions during these events, response teams must conservatively predict the extent of the contaminated area and then orchestrate evacuations and reroute traffic to ensure the safety of nearby populations.**

In this paper, we propose outfitting swarms of drones with Internet of Things (IoT) sensor platforms to enable dynamic tracking of hazardous aerial plumes. Augmenting drones with sensors enables emergency response teams to maintain safe distances during hazard identification, minimizing first response team exposure. Additionally, we integrate sensor-based particulate detection with autonomous drone flight control providing the capability to dynamically identify and track the boundaries of aerial plumes in real time. This enables first responders to visually identify plume movement and better predict and isolate the impact area. We describe the composition of our prototype IoT-enhanced drone system and describe our initial evaluations.

***Keywords*-Server, IoT, Swarm, Drones, Sensors**

I. Introduction

Hazardous material incidents constitute critical emergencies as toxic substance releases can threaten public health, radically impact the environment, and in the worst case cause fatalities. When spills occur, emergency response teams must quickly identify the agents involved and develop a real-time plan to limit exposure to first responders and surrounding communities. While all types of hazardous spills are difficult to manage, airborne releases of toxic chemical agents in gaseous form, commonly used within industrial processes, are particularly challenging [1]. The odorless, invisible nature of gaseous toxins make released plumes difficult to assess, track, and manage. Changing environmental conditions, such as temperature, humidity, wind speed and direction combined with obstructions such as buildings, vehicles, and natural land features further complicate the identification and tracking of moving toxic plumes. Despite these challenges, first responders must rapidly identify the spilled substances and mitigate the impact with limited information.

Emergency response organizations arriving on scene initially maintain a conservative distance to minimize first responder exposure. Team members immediately don protective gear and traverse to the spill site to identify the contaminant at the source. After identification, the safe distance threshold is re-calibrated based on the substance. Many response teams then leverage computational fluid dynamics (CFD) models using on-scene laptops to input observed environmental parameters to identify how the airborne plume may move and which areas are most likely to be affected [2]. Because these models may not take into account every effect that can impact plume movement, response teams typically add a buffer zone to the model to widen the predicted area affected. While this increases the safety margin, such measures can further exacerbate traffic disruptions, cause unnecessary evacuations, and temporarily halt operations critical to the economic health of the region.

Response operations face two significant challenges when handling airborne hazardous material spills. First, time is critical. Minimizing the time needed to safely identify spilled toxins and engaging all available response personnel is crucial to minimize the threat. Second, the prediction model used during these emergency events is critical to effectively managing these events over time. However, while these models are helpful in managing airborne plumes, validating on-scene generated models is difficult. Moreover, it is difficult to convey plume movement to on-scene responders actively working the scene. Finally, without a way to validate the model, tracking hazardous plume movement while the response team mitigates the threat remains unclear.

In this work, we propose employing swarms of unmanned aerial vehicles, or drones, to address these two challenges - reducing the time to identify toxic substance problem and dynamically tracking the movement of toxic airborne plumes. Drones have been increasingly deployed to identify changing environmental conditions [3][4][5][6][7][8][9][10][11][12]. Unlike previous work, we propose integrating IoT platforms onto drones to augment their sensing capabilities for detecting and managing air-

978-1-4673-9944-9/18/$31.00

borne hazardous plumes. By using a drone for toxin identification, we enable response teams to initially maintain safe distances but also minimize identification latency. We also couple IoT data collection with the drone flight control, enabling two key new capabilities. First, our drones provide a visual reference point for plume movement that enables on-scene first responders to simply *look up* to gauge their proximity to the plume as well as its trajectory over time. To do so, we configure each drone to continually monitor contaminant density to identify the boundaries of the plume and autonomously move with the plume. Second, we coordinate the movement of the drones so they continually move together. Using a swarm of drones, equipped with IoT platforms enables response teams to quickly visualize areas with the highest concentration of toxins moving in real time. Additionally, we transmit all drone-collected sensor data, including GPS positions, particulate concentration levels, and time stamps to record plume movement during response events. This provides the key missing piece to localize the generic CFD models used today - the data to validate the predicted path of a hazardous plume.

II. System Architecture

We built a complete system consisting of IoT platforms, sensors, and swarms of drones to locate, identify, and track hazardous plumes in real time. Once located, each drone transmits data about plume boundaries so it can be tracked in real time including data points such as wind direction, speed, location, and the chemical composition and density measured within the air. In addition, all drones dynamically coordinate as a swarm and adjust their position to dynamically track the outer perimeter of the plume. This enables ground-based responders to see a visual representation of the area occupied by the plume in real time. Over time, this also enables immediate identification of high risk areas that should be evacuated.

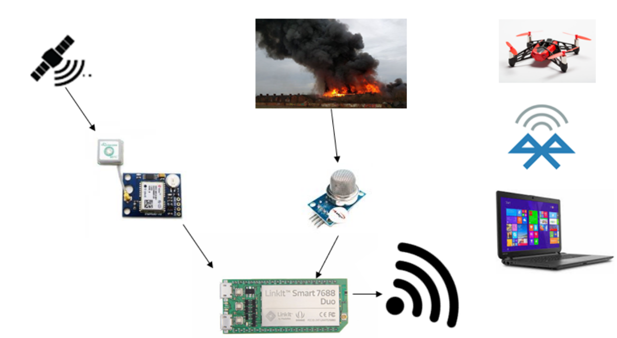

Figure 1: An overview of the system with a single drone

Our system consists of drones in the form of quad copters equipped with a microprocessor-based IoT board, onboard GPS receiver, and air quality sensors. These drones use air particulate and GPS data to locate hazardous plumes. Using this data, each drone dynamically adjusts its heading and position based on current environmental observations. This data is also transmitted over the wireless network to an on-scene server, which stores the data for later analysis as well as displays the data in real time via browser as shown in figure 1.

A. Drone & IoT Platform Integration

Our current prototype system uses the Parrot rolling spider drone. Each drone is equipped with a Linkit 7688 Duo Microprocessor board, MQ135 Air quality sensor, NEO-6M GPS module and 3.7v 250mAh battery. These components are coupled using a custom printed circuit board (PCB) that connects the air quality sensor, GPS, and battery to the microprocessor. Since the battery is only 3.7 volts there is a step-up boost circuit on the PCB that enables the 3.7 supply voltage to be converted to 5 volts for use by the board and sensors. Given the limited lift capacity of these drones, weight and form factor are critical constraining factors in our design. We choose to use these drones for our initial evaluation, although we have specifically designed all components to be used with larger, more capable models in future work.

B. System Software

Each of the drone-mounted IoT platforms collect air quality and GPS data. During flight the drones are oriented in a four quadrant, grid pattern with one drone serving as the master drone managing communication with the ground-based command and control system. The data from slave drones, such as the left, right, and rear drones in figure 2 are sent to the main drone using sockets over WiFi and routed to a central command and control computer. Air quality data is also routed to the main computer where it is used to control the swarm through Bluetooth control channels. Each drone's position is adjusted, based on a new heading, to modify the swarm configuration dynamically based on the air quality data sensed.

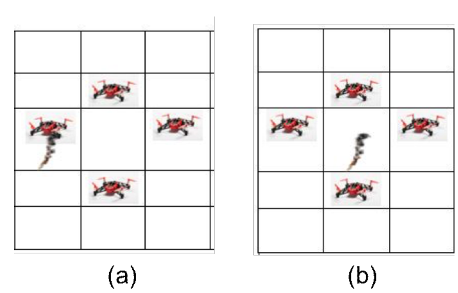

Figure 2: (a) Left drone finding plume. (b) Swarm surrounding plume

To illustrate how the drones react to changing conditions we use a swarm of four drones. Within a swarm, each drone is assigned a designated position on the horizontal plane -

front, right, rear, and left. When the left drone is in the highest concentration of smoke as shown in figure 2(a), this means the swarm should recenter itself such that the perimeter of the smoke can be identified. Consequently, the swarm will move left toward the smoke until all drones surround the smoke as in figure 2(b). The swarm will move toward the drone that is in smoke until all drones take up positions within the smoke. Once all the drones are located within the smoke, the swarm will begin to expand until all drones are on the edge of the smoke plume. Although we only use four drones in this example, our approach works with any number of drones. In fact, with additional drones, we expect to be able to visually identify aerial plumes in multiple dimensions.

III. Evaluation Methodology

To evaluate our proposed system we first had to identify which components to use. We started by experimenting to identify which quad-copters and IoT platforms to use. The evaluation criteria we used included lift capability, durability, battery life, positional accuracy, and data communication integrity. Our first challenge was to ensure our design did not exceed the lift capacity of the small drones. This turned out to be a key challenge that heavily influenced the design of the IoT platform we integrated. All components were tested individually as well as incrementally as we integrated components prior to full system testing to ensure that issues encountered during system evaluations could be easily identified and repaired. This proved crucial given the inclusion of multiple components on each drone. This testing is based on three key areas: physical parameters (size matters), system inter-operability and latency, as well as integration with each drone's autonomous control system.

A. IoT Platform and Drone Integration

Drones. For this work, we used the Parrot Rolling Spider drone. This drone was selected for several reasons. First, its size and weight made it an ideal candidate for an indoor evaluation. The small form factor enabled use in confined areas, while its weight allowed for nimble movement without requiring bulky, high output motors. This improved the stability of our autonomous systems with minimal weight, however, the small form factor also limited our payload capacity which impacted the IoT platforms we could use. The GPS, IoT board, air quality sensor and other sensors were chosen based on a compromise between weight, size, and functionality.

Before pairing each drone with sensors and IoT board, we tested each drone for mobility. Since our swarm system does not require high speed maneuvering capabilities we ensured basic flights movements - lift, descent, forward, reverse, left, right, rotate right and rotate left. We then tested the battery life of each drone under two scenarios: 1) motionless hovering and 2) minimum battery life with constant motion. We then identified each drone's lift capacity under different battery capacities; in other words, how much weight each drone could carry while maintaining stable flight characteristics. We found the combined payload could not exceed 40 grams without compromising maneuverability. This constraint proved to be a challenge for the GPS module as it will not operate without a 7 gram antenna.

Battery selection was also critical to power both the IoT platform, sensors, and drones. Ideally, drones should remain in flight for as long as possible to track aerial plumes, but this requires larger-capacity batteries. Given the small form factor drones we used in this prototype system, we had to strike a balance between weight and power capacity. We first characterized the power load of the drones as well as the IoT platform during flight. Based on the components data sheet, each IoT board draws between 200 - 300 milliampere (mA), the air quality sensors require 150 mA, and the GPS requires 45 mA for a total of 400 - 500 mA per drone. Using a battery load analyzer we found that a lithium polymer (LiPo) 3.7 volt battery would last for over 20 minutes before needing to be recharged.

IoT Platform. We used the Linkit Smart 7688 Duo IoT board given its compact size and weight. Each board weighed 10 grams and the form factor closely aligned with the available drone footprint making mounting on the drone straightforward. This board is a dual platform architecture supporting both Linux and Arduino as well as file system memory expansion capabilities via an on-board SD card slot. The MQ-135 air quality sensor was chosen based on its ability to provide real time data with minimal latency as well as its small form factor. After an initial burn-in period of 10 hours, each air quality sensor was required to run for at minimum of one hour prior to testing to ensure accurate readings. Hazardous air quality thresholds were identified by empirically testing the sensors with various gases and analyzing the results. As previously mentioned, we used a 250 milliampere, 3.7 volt LiPo battery. We also integrated a GPS module that we connected to the IoT board via UART connection.

We designed and developed a custom PCB, shown in figure 3, to combine all sensors with the IoT board that could be mounted on each drone. We used the Eagle design tool for wiring schematics and built a two layer board to include necessary components including the on-board step-up circuit to convert the 3.7 volt batteries to the 5 volts required for all components.

To mount the IoT board, PCB, and sensors on the drone, we used 3D modeling software and a 3D printer to create a custom case to use as a base mount as shown in figure 4. We mounted the base, with all components, to the drone by taking advantage of the wheel mounts to secure the platform.

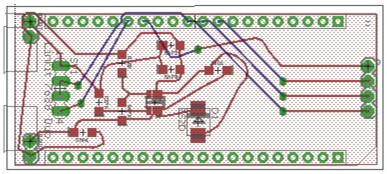

Figure 3: Board Mount Design

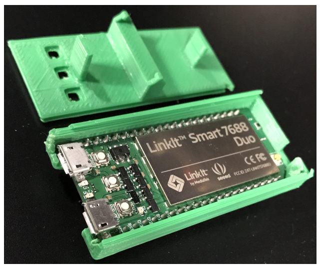

Figure 4: Board Mount Prototype

B. Software Configuration

To control the drones, we used the Parrot drone Software Development Kit (SDK) that is available for Parrot devices. This enabled us to leverage the built-in motor control algorithms for hovering and basic movement. We further developed and integrated our own control algorithms for the tracking system using the Node.js API to the Parrot SDK. For primitive functions, such as moving, launching, and landing the drone, we used standard SDK functions. To use GPS location data, we used a JavaScript GPS library to read and interpret NMEA sentences generated by the GPS. We then used the current position and heading to identify the speed and heading for drone movement.

For the air quality sensor, the analog-to-digital (ADC) pins on the IoT board were controlled by the Arduino-based MCU chip. Communication between the MPU (Linux side and main processing/communication) and the Arduino based MCU is obtained via an internal UART connection. This meant we needed to use a script on the Arduino side that captured data from the Analog to Digital Converter (ADC) pins. The benefit of this setup is that the MCU is better suited for real time software applications such as analog sensor inputs.

C. Swarm Communication

Communication between drones is necessary to exchange position information, distribute control commands, and log sensor readings. We used two levels of communication within the swarm: inter-drone links and swarm-to-ground links. All drones used three dedicated communication channels using distinct ports between drones. Two of the three channels were used strictly for transmitting air quality data and GPS positions. These channels used standard socket communication over the 2.4GHz WiFi network to the master drone using standard IP addressing. The third channel was a receive-only channel dedicated for drone control messages sent from the base station over the Bluetooth Low Energy (BLE) protocol. These three channels constituted the inter-drone links.

The second level of communication consisted of swarm-to-ground channels. One drone was designated as the master, which served as an access point through which the other drones communicated with the base station server. To support this communication link from air to ground, the master used one additional WiFi channel for communication. Commands between the base station server and the master drone were then relayed to each drone based on the destination drone identifier. Similarly, the master drone was responsible for relaying sensor data from all nodes to the base station server. In our evaluation, we used a laptop as the main server for sending out control commands over BLE. The base station was also set up to receive all air quality data and GPS positions over WiFi links. All received data and commands were logged on the base station server. Once communication channels were established, the control algorithm was then evaluated to ensure all drones were properly receiving directional commands and all GPS and air quality data were being received.

D. Aerial Plume Testing

To test our system we needed to safely release airborne contaminants without compromising our health. Given our air quality sensor detects carbon monoxide (CO), we used cans of compressed air. Each release of compressed air emits small quantities of CO, which is part of the propellant. We tested our integrated gas sensor and found we were able to detect the increase in CO from each release of air from the can. This enabled us to selectively release a new *CO plume* during our swarm experiments.

IV. Evaluation Results

In our initial experiments, our primary goal was to characterize how well our prototype drone swarm detected and reacted to airborne plumes. As previously described, we used compressed air canisters to simulate the release of toxins in the air. Each experiment was conducted within a large classroom on the UW campus. During testing, we continually refined the detection and autonomous navigation logic after each experiment. We also logged the sensor values and movement commands from our control system during each flight and analyzed them after each experiment. We used these results to tune our drone navigation and control system. Although we have collected results from a significant number of experiments, figure 5 depicts results after refining our control system from 30 previous experiments.

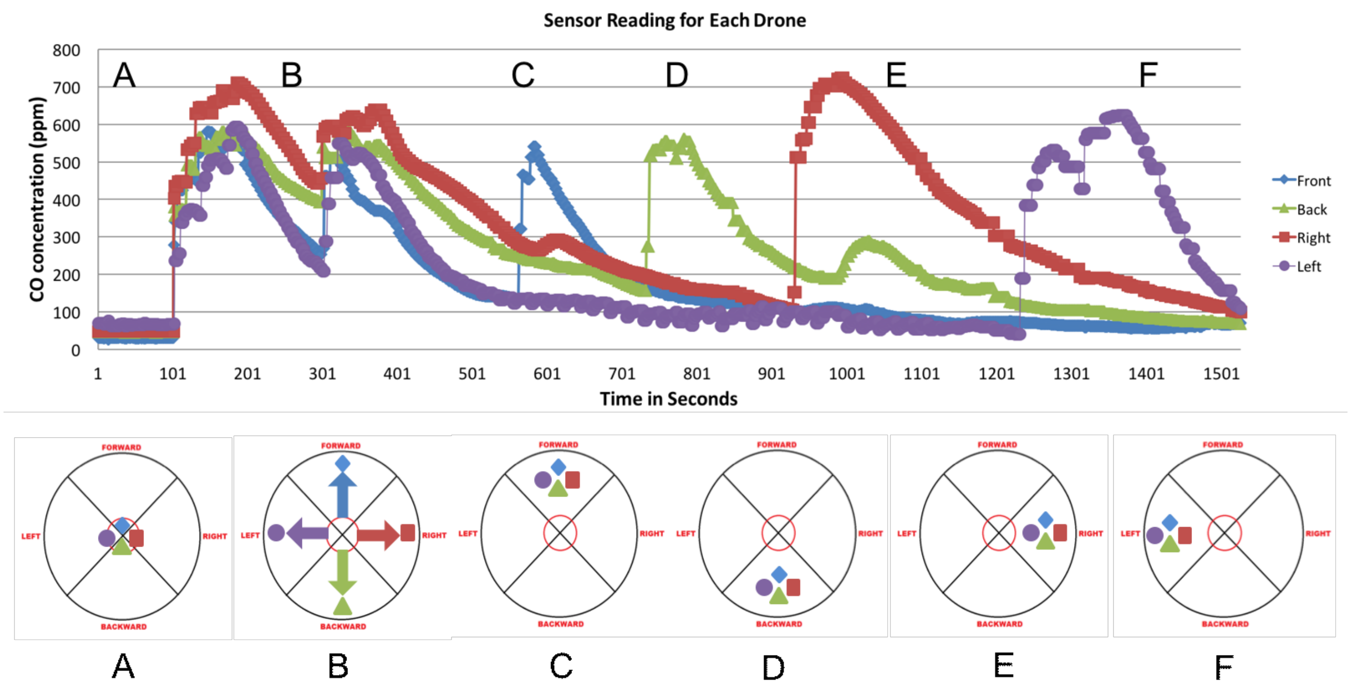

Figure 5: drone movement with sensor input

The upper half of figure 5 shows the the gas concentration level recorded by the air quality sensors of each drone during a 25 minute experiment. The lower half of figure 5 shows how the drone swarm moved when we introduced additional plumes of simulated toxins within the environment. We used the same shapes and colors within the upper and lower graphs to correlate each drone with its respective sensor readings over time. We have labeled figure 5 with letters to identify important changes. When the particulate concentration increases, our system should be able to detect the release, find the boundaries of the released plume, and then continue to track that plume over time.

During the first two minutes of each experiment, identified by label A in figure 5, we launched the drone swarm in a contaminant-free environment. This was done to ensure each drone successfully achieved flight and to validate all sensors were operational. Once the swarm was verified to be functional, our first test was to inject an aerial plume directly under the swarm. This contaminant injection is identified in figure 5 by the label B. As shown, the air particulate sensors of all drones immediately detect the increase in gas concentration. However, given the sudden contaminant release, it's unclear where the perimeter of the plume is located. Thus each drone autonomously moves outward from each other in their respective direction as shown by label B in the lower half of figure 5. After the initial injection the gas concentration level decreases after the initial spike, but then increases in what appears to be a second spike. This second spike reflects the movement of the drones back towards each other after finding the plume boundaries. Although the graphs do not reflect the slow movement over time, we observed the plume drifted within the room as the contaminant dispersed.

During real emergencies, additional airborne releases of toxins often occur while first responders are mitigating the threat. In these situations, the drone swarm should immediately detect the new release and once again identify the *new* plume boundaries. Consequently, once we validated the swarm accurately determined the boundaries and could track it over time, we injected additional contaminants in select locations near the in-flight swarm such that only a single drone would sense the injection. In this case, the entire swarm should dynamically and autonomously readjust their positions to account for the new boundaries. In the first test, we injected a new plume near the drone identified as covering the forward position. This is shown as label C in figure 5. The air quality sensor for the forward drone spikes, triggering the swarm to change positions and recenter around the the newly discovered area with higher CO concentration. Although not reflected in figure 5, after the initial movement of the swarm toward the area closest to the drone that detected the increase, all drones once again searched for the plume boundaries. This boundary search is reflected by the secondary increase in CO concentration after the initial spikes.

We then proceeded to further inject additional CO using the compressed air canisters near each drone incrementally as shown by labels D, E, and F. In each case, the nearest drone accurately detected the increased level of CO within the air, and the swarm was moved incrementally reflecting how the plume was moving over time when further releases occurred.

Limitations. Although these results show that we were able to accurately move the drone swarm in the horizontal plane, we intentionally restricted drone flight at the same elevation during these experiments. During real scenarios we would also want to know how the plume is moving vertically along the z-axis. Additionally, we would want to know how the drones are affecting the plume, particularly how much the swarm is moving contaminated air downwards towards first responders. We plan to address these limitations in future work by increasing the degrees of freedom in flight movements as well as increasing the size of the swarm to provide additional cross-sections along the z-axis.

V. Related Work

There has been significant work done recently to use drones for detecting hazardous gasses [3][4][5][6][7][8][9][10][11][12]. In [5], a quad copter was armed with sensors to find and monitor hazardous gasses in the atmosphere. In [11] a drone was used to monitor and track the movement of a hazardous cloud and send the readings to a base computer using the flight controller for later analysis. A drone was used to monitor gas concentrations in large underground pipes and caves while streaming data to a mobile view for real time reporting in [12]. Each of these studies focused on using a single drone rather than a swarm. Additionally, while our swarm system also collects sensor readings for later assessment and analysis, it also provides a visual indicator useful for first responders to visually track plume movement over time.

VI. Conclusion

In this paper, we have described the hazardous plume identification and tracking system we built by combining sensors on a swarm of autonomous drones. Our early results reveal that we are able to accurately identify and track contaminant plumes over time to provide a visual indicator to on-scene first responders as well as collect data that can be used to validate and improve plume movement models over time.

VII. Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. CNS-#1742899.

References

[1] N. F. P. Association *et al.*, *NFPA 704, Standard System for the Identification of the Hazards of Materials for Emergency Response*. National Fire Protection Association, 2017.

[2] M. Pontiggia, M. Derudi, M. Alba, M. Scaioni, and R. Rota, "Hazardous gas releases in urban areas: Assessment of consequences through cfd modelling," *Journal of Hazardous Materials*, vol. 176, no. 1, pp. 589–596, 2010.

[3] R. R. Murphy, J. Peschel, C. Arnett, and D. Martin, "Projected needs for robot-assisted chemical, biological, radiological, or nuclear (cbrn) incidents," in *Safety, Security, and Rescue Robotics (SSRR), 2012 IEEE International Symposium on*. IEEE, 2012, pp. 1–4.

[4] A. Birk, B. Wiggerich, H. Bülow, M. Pfingsthorn, and S. Schwertfeger, "Safety, security, and rescue missions with an unmanned aerial vehicle (uav)," *Journal of Intelligent & Robotic Systems*, vol. 64, no. 1, pp. 57–76, 2011.

[5] N. Gerhardt, R. Clothier, and G. Wild, "Investigating the practicality of hazardous material detection using unmanned aerial systems," in *Metrology for Aerospace (MetroAeroSpace), 2014 IEEE*. IEEE, 2014, pp. 133–137.

[6] E. D. McCormack and T. Trepanier, "The use of small unmanned aircraft by the washington state department of transportation," Washington State Department of Transportation, Tech. Rep., 2008.

[7] P. Vincent and I. Rubin, "A framework and analysis for cooperative search using uav swarms," in *Proceedings of the 2004 ACM symposium on Applied computing*. ACM, 2004, pp. 79–86.

[8] T. Kersnovski, F. Gonzalez, and K. Morton, "A uav system for autonomous target detection and gas sensing," in *Aerospace Conference, 2017 IEEE*. IEEE, 2017, pp. 1–12.

[9] H. Menouar, I. Guvenc, K. Akkaya, A. S. Uluagac, A. Kadri, and A. Tuncer, "Uav-enabled intelligent transportation systems for the smart city: Applications and challenges," *IEEE Communications Magazine*, vol. 55, no. 3, pp. 22–28, 2017.

[10] T. Mac Thi, C. Copot, T. T. Duc, and R. De Keyser, "Ar. drone uav control parameters tuning based on particle swarm optimization algorithm," in *IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, 2016, pp. 475–480.

[11] F. Mochammad, A. R. Putra, and B. R. Trilaksono, "Implementation of hazardous chemical gas monitoring system using unmanned aerial vehicle (uav)," in *Frontiers of Information Technology (FIT), 2016 International Conference on*. IEEE, 2016, pp. 163–168.

[12] L. Shi, X. Wang, T. Zhang, C. Hu, K. Luo, and B. Bai, "Hazardous gas detection four-rotor uav system development," in *Mechatronics and Automation (ICMA), 2016 IEEE International Conference on*. IEEE, 2016, pp. 2461–2465.