ABSTRACT

Title of Thesis:	Development of Firebrand Optical Characterization Device for Deployment on UAVs
	Jayce Baek, Bachelor of Science in Mechanical Engineering, 2025 Leonello Castro Cillis, Bachelor of Science in Aerospace Engineering and Bachelor of Science in Physics, 2025 Max Goldberg, Bachelor of Science in Computer Science and Bachelor of Science in Mathematics, 2025 Zachary Kiedrowski, Bachelor of Science in Environmental Science and Bachelor of Science in Mathematics, 2024 Evan Osborne, Bachelor of Science in Chemical Engineering, 2025 Imaad Ali Syed, Bachelor of Science in Electrical Engineering, 2025 Nolan Westlake, Bachelor of Science in Mechanical Engineering, 2025
Thesis Directed By:	Assistant Professor, Dr. Fernando Raffan-Montoya, Department of Fire Protection Engineering

Wildfires increasingly threaten ecosystems and communities, driven by climate change and expanding wildland-urban interfaces. A key but poorly understood factor in wildfire spread is firebrand transport—embers carried by wind that can ignite spot fires far from the main blaze. This project developed a stereo vision camera system for potential deployment on unmanned aerial vehicles (UAVs) to track firebrand release and movement. Using synchronized, high-resolution cameras, the system captures 3D trajectories, allowing for estimation of firebrand velocity, lifespan, and dispersal.

Unlike ground-based systems, a UAV-integrated approach enables rapid, flexible, and safer data collection in active fire zones. The future insights gained from this technology can support improved fire spread models and defensible space guidelines, contributing to more effective wildfire management.

DEVELOPMENT OF FIREBRAND OPTICAL CHARACTERIZATION DEVICE FOR DEPLOYMENT ON UAVs

by

Jayce Baek, Leonello Castro Cillis, Max Goldberg, Zachary Kiedrowski, Evan Osborne, Imaad Syed, Nolan Westlake

Thesis submitted in partial fulfillment of the requirements of the Gemstone Program
University of Maryland, College Park, 2025

Advisory Committee:
Dr. Fernando Raffan-Montoya
Dr. Nicolas Bouvet
Dr. Lina Castaño
Ms. Adetola Koiki
Dr. Peter Sunderland
Dr. Ali Tohidi

Acknowledgements

Team SUPA' HOT would first like to thank our mentor Dr. Fernando Raffan-Montoya for his continued support, mentorship, and technical expertise throughout our project. His guidance was essential to our team's success and direction. We want to thank Adetola Koiki for her support in the testing setup of our firebrand characterization device and collaborative efforts in understanding stereo vision applications with firebrands. We would also like to thank Dr. Arnaud Trouvé for his advice and guidance in the preliminary stages of our project. Our thanks also go to the members of the thesis defense panel for generously dedicating their time to delve into our research and providing constructive feedback. Lastly, we would like to thank the Gemstone Program staff for their constant support over our four years in the program as well as our LaunchUMD donors who aided our project financially.

Table of Contents

Chapter 1: Introduction

Wildfires are dangerous events that hurt lives and may increase in intensity and frequency. A major way that wildfires spread is through firebrands, otherwise known as embers. Current knowledge on how these firebrands spread is limited due to the difficulty of observing them during active wildfires. This research endeavor aims to develop a drone mountable device to address this gap in understanding.

1.1 Danger and Impacts of Wildfires

Wildfires have long been a major cause of concern for people across the world. Wildfires have received even more attention over the past few years due to the increasing and more extreme effects of climate change [1], [2]. In places such as California and Australia, a historic and unprecedented number of wildfires have uncontrollably devastated vast regions of land in the last few years. Additionally, the common practices of suppressing wildfires have led to a buildup of fuel on forest floors which results in higher intensity fires [3]. As high intensity wildfires cause massive destruction and threaten human life it is imperative that models can accurately predict fire spread so that firefighting efforts can be used most effectively. An important mechanism of wildfire spread is through lofted firebrands, alternatively known as embers. There is a limited understanding of how these firebrands form and spread in active wildfires [4]. By developing a tool to measure firebrands during wildfire events there is a potential to better models and better manage risk along wildland urban interfaces (WUI).

Wildfires are dangerous natural disasters that commonly occur in areas with extremely dry conditions and high winds. They are often caused by human activity or

natural events such as lightning strikes. The smoke and ash from fires contain particulate matter, carbon monoxide, nitrogen oxides, and various volatile organic compounds, which can significantly reduce air quality in nearby areas. Smoke exposure causes pulmonary inflammation, bronchitis, exacerbations of asthma, and heart failure. Wildfires also release mercury into the atmosphere which leads to health issues [5]. Climate change is amplifying drought frequency and severity [6], which leads to an increase in wildfires that can become more destructive. In fact, wildfires themselves contribute to greenhouse gas production, which can influence climate change. The destruction of homes, businesses, and properties is detrimental to the economy because of the expenses for health-related issues and rebuilding of property. These health impacts and loss of property, along with the direct burns and injuries caused by wildfires, present a considerable threat to the quality of life of millions of people annually.

Wildfires also have strong ecological impacts. Wildfires produce carbon dioxide and other greenhouse gases which contribute to global warming and can lead to positive feedback loops with more wildfires [7]. Climate change will be causing more intense wildfires worldwide [8]. Intense wildfires can destroy edible plant material and cause trophic cascades where entire food webs collapse as primary consumers do not consume enough energy [7].

1.2 Firebrands and The Wildland Urban Interface

Firebrands, commonly called embers, are burning pieces of organic material produced by wildfires that become lofted in the air during wildfire events [9]. Firebrands play a significant role in the unpredictable spread of wildfires, especially via spot fires. Spot fires are created when firebrands are lofted and dropped ahead of the

fire front in a density great enough to ignite another fire as shown in Figure 1 [10]. Ignition of fires ahead of the fire front is a concern for fighting fires as it makes fires complicated and hard to predict and can trap fire fighters between fire lines [1]. When fires reach what is known as the wildland urban interface, or the edge territory where human developments meet wildland, firebrand lofting contributes heavily to the ignition of structural fires. Firebrands are one of the primary causes of the devastating fires in Maui in 2023 which resulted in 100 deaths and billions of dollars in damages [11]. In experimental burns in the Pine Barrens in New Jersey in 2013-2015, firebrands were collected and determined to come from bark, and tree and shrub branches [12]. The firebrands were tracked visually from cameras on towers and firebrands had higher densities during the fire 13 meters from the fire front [12], a reasonable distance to be lofted to get from a forest line to a building to ignite the building.

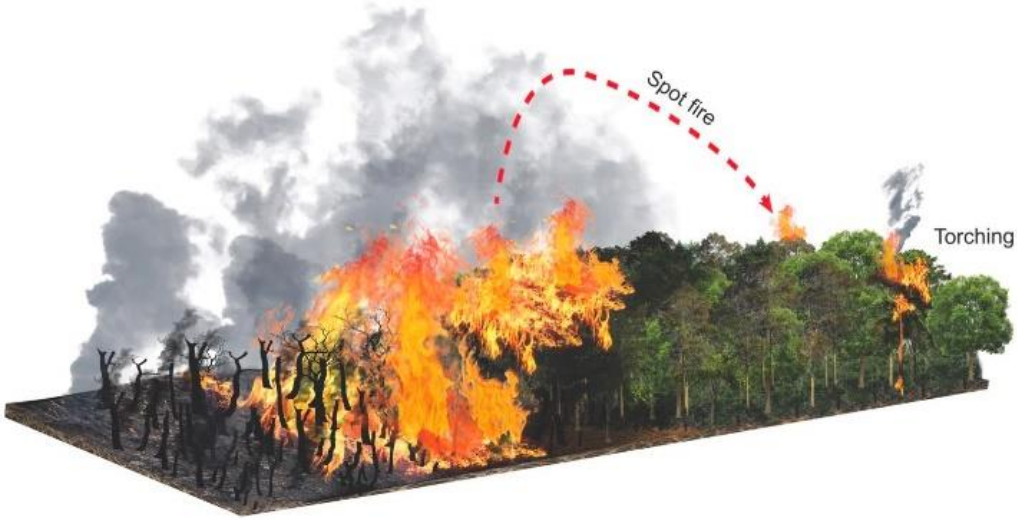


Figure 1. Illustration of spot fire torching ahead of the fire front [10].

1.3 Ignition from Firebrands

It is vital to understand the aerodynamic properties that facilitate the collection of firebrands on roofs since firebrands are the primary way of igniting structural fires

on the Wildland Urban Interface. An experiment was performed by blowing air at different speeds and angles at a model house with firebrands on the roof to examine how the firebrands would build up. The focus was not on the temperature of firebrands, so model firebrands that were not burning were used. These were produced by sorting and crushing pine straw, sticking it in a blender, and then sieving it to the desired size. It was determined that where low pressure vortices form, the firebrands would be removed from the roof with higher frequency. However, the firebrands tended to get caught in crevices where considerable amounts of combustible organic matter are already known to collect. On average, even in the higher wind scenarios, over 50% of the model roof remained covered with firebrands [13].

Examining metal particles, it was found that for smoldering to be observed, the particles would have to be at least 550 °C, and for flames they would have to be 650 °C [14]. Another interesting insight is that, for relatively large particles, the total size was less of a contributing factor towards ignition than temperature (though both play a role) [14]. However, in another study it was shown that increasing the number of firebrands present could increase the likelihood of ignition. The firebrands were 490–650 °C when they impacted the surface, and it was shown that 6-10 of these firebrands in a 50 mm × 50 mm area were needed to ignite a fire [15]. Of course, these results could vary based on material, moisture content, or other environmental factors. Another study has shown that the resulting ignition can be based on if a firebrand is glowing or flaming. Flaming firebrands may result in flaming ignition while glowing firebrands may result in smoldering reactions [10]. This relationship is easier to measure with visible light cameras while temperatures can be quantified with infrared cameras.

1.4 Conventional Methods of Measuring Firebrands

General convention when studying wildfires refers to three phases in the formation of firebrands: the generation phase, the transportation phase, and the landing phase. Of these phases, still there is relatively little understanding about the generation phase, regarding the distributions of size, shape, and mass of the firebrands that are generated. One such method of measuring these firebrand characteristics is examining imprints on surfaces not ignited. This measurement method was first done with a single trampoline following the Angora Fire in 2007 [9]. One group collected seven trampolines from the area impacted by the Bastrop Complex Fire in 2011 and performed an analysis on the holes left by the firebrands to collect data [16]. An example of such a trampoline is given in Figure 3. While some of the trampolines were affected by excessive damage from the fire, images were examined using image processing software to determine summary statistics about the area of the firebrand imprints. This simple method of studying the impacts of firebrands is innovative and provides useful insight. However, a major limitation of this method lies in the fact that measurements can only be made after the fire has already done its damage and moved on. It is difficult to relate patterns in the firebrand production to characteristics such as wind speed or fire intensity without the temporal aspect of taking measurements.

Figure 2. Map of fire perimeter with trampoline locations and firebrand count at each location [16].

Firebrands have also been physically collected after fires. For example, following the Beppu-City Fire in 2010 firebrands were collected and georeferenced with the fire line to determine distance traveled [9]. In this instance firebrand mass was also measured, though the collection method was not very controlled as the researchers could not have preplanned the fire coming through the city. This collection method was also used after the Itiogawa-City fire in 2016 which in contrast started as an urban fire. In the case of the Itiogawa-City fire, researchers randomly collected firebrands from both the ground and roof tops and oven dried the firebrands before weighing and using image processing to characterize size [17].

Despite the limited investigation into firebrands in active, uncontrolled wildfires [18], many lab investigations have been performed to better understand the interactions involving firebrands. A study was done by burning Douglas Fir trees and collecting the firebrands [15]. While not entirely representative of real-world situations,

studies such as this can help guide further testing involving concepts like fire resistance. Trees with moderate moisture levels were ignited using custom burners, resulting in firebrand showers. Water pans were placed on the ground spaced evenly in the surrounding area, resulting in firebrands landing in the pans and being quenched. A sample of quenched firebrands was then removed, allowed to dry, and then measured with a precision balance and calipers [15]. The process carried out here gives great initial understanding of firebrand characteristics, but it could also be used to calibrate the measurements of an instrument or artificial firebrand production device. In fact, the developers of the current stationary "emberometer" which is described later used a similar process in their testing [19].

While laboratory experiments and studies conducted in areas recently affected by wildfires provide different information regarding the fate of firebrands and theoretical behaviors, studies with controlled burns allow for close simulations of wildfires with intentional measurements being taken during the burn. The New Jersey Pine Barrens controlled burns in 2013-2015 had towers both above and below the tree canopy to measure wind speed, air temperature, and turbulence during the burn [9]. In these experiments the fuel was collected before the burn and firebrands collected during the burn to provide insights as to how the fuel available influences that firebrands produced. The firebrands were collected using plastic films overtop of pans though video footage showed that these films sometimes deflected smaller firebrands [9]. Using this information, we can deduce that a similar bias was likely present in using the trampoline method like in the Bastrop Complex Study. This plastic sheet method is an effective way to measure firebrands first implemented in building fire tests in the 1960's [20], and provides more evidence than just using water pans, however it still

has limitations and does not give information about the mechanisms that loft the firebrands. There was IR footage collected during the Pine Barrens controlled burns that was further analyzed using video processing software to determine particle count [21]. The combination of these methods together can produce a more sophisticated understanding of the firebrand dynamics.

Other work has been done towards understanding characteristics of firebrands, especially how they interact with potential fuel sources. In general, for a firebrand to ignite a fuel source, it must contain enough energy to heat and pyrolyze the fuel [16]. Additionally, to create a flame as opposed to a smolder, additional energy must be present to create a reaction in the surrounding fuel [14]. While the primary focus for forest fires should be on understanding conventional firebrands, it is also important to recognize that metal particles of a certain temperature can also ignite fires. Metal particle ignition was studied by experiments where hot particles were dropped one at a time onto different fuel beds and ignition patterns were observed [14]. While examination of ignition by single particles may work for the study of large particles but often would not be an effective way to study small particles. These unique metal particles are relevant to the study of wildfires as in WUI settings where the impacts of wildfires on people is the greatest, it is possible that there are nonconventional fuel sources that could lead to distinct types of firebrands.

1.5 Firebrand Generation in Laboratory Settings

There are multiple ways of producing simulated firebrands in laboratory settings. Laboratory standards for working with firebrand production were developed by using a heat drier and a burner to create a small-scale firebrand shower out of pine twigs and wood pellets that acted on a structure made of plywood boards [2]. The heat

drier blew 400 °C air at 5m/s, targeted at the wood structure as shown in Figure 3 to simulate conditions at the front of a wildfire [2]. An infrared (IR) camera was shown to be effective at reading the temperatures of this process, having a range from 200 °C to 800 °C [2]. While the IR camera capabilities and image processing associated with these procedures are useful, the most important insight provided by this data is the understanding of how many firebrands are needed for ignition.

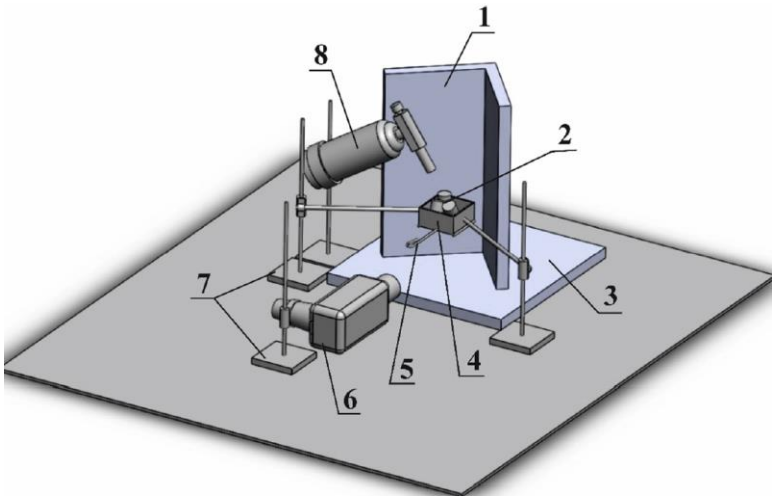


Figure 3. Laboratory burner setup: 1 – construction sample; 2 – particle samples; 3 – underlying surface; 4 – cell; 5 – stopper; 6 – heat gun; 7 – tripods; 8 – burner [2].

An alternate method that has been used for firebrand production is the NIST "Fire Dragon" [19]. The "Fire Dragon" uses a loading bay for fuel pellets and constant burner with ducts at an angle to that firebrands and produced and propelled by a blower through the duct and blown out the top in a way that the firebrand trajectories can be studied in a small volume of integration as shown in Figure 4 [19].

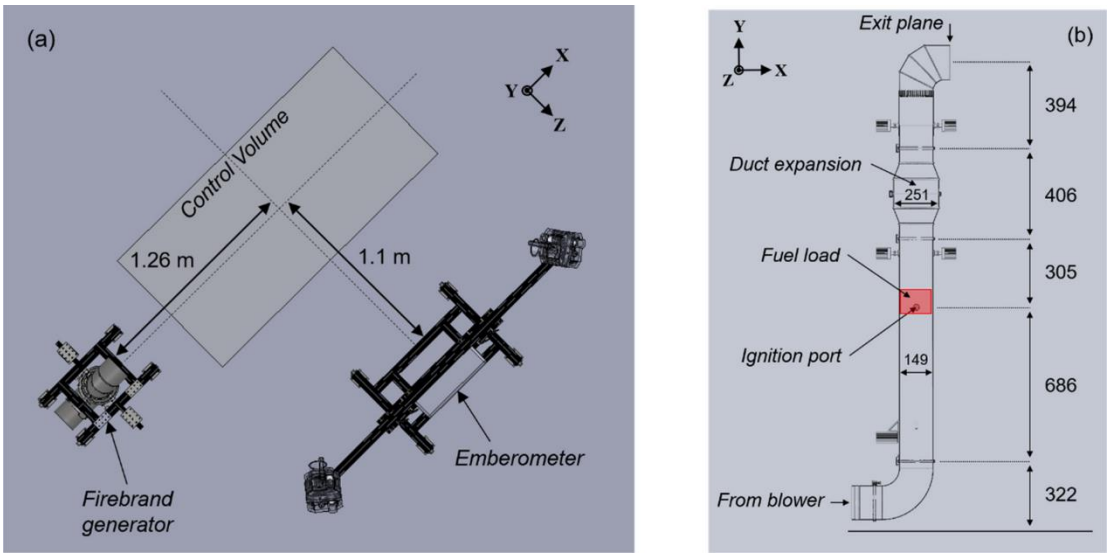


Figure 4. Experimental layout for NFRL tests: **a** Top view; **b** Structure details of the firebrand generator (all dimensions in mm) [19].

Wind tunnels offer another mechanism for propelling firebrands to be studied. This allows for the replication of wind gusts that may be present during a wildfire event [12], [20]. These methods for creating and observing firebrands are practical means of validation for technologies being created but are not necessarily representative of the reality of wildfires.

1.6 Firebrand Characteristics

Many studies have been conducted as efforts to characterize firebrands, including controlled burns, simulations, laboratory experiments, and indirect observation of wildfires' aftermath. While the information collected by each of these studies tells an incomplete story, we can put them together to gain a better understanding of the current knowledge base on firebrands. This information was employed to direct the focus of testing the device that the team developed.

The Bastrop Complex Fire study investigated ember (firebrand) production during the most destructive wildfire in Texas history, which burned over 32,000 acres and destroyed 1,700 structures in 2011. By analyzing burn marks on trampolines collected from the field and using image recognition software, the study provided real-world data on firebrand size and density. The study found that between 91% and 99% of the holes left by firebrands were less than 0.5 cm^2, and there were 9 and 68 holes per ft^2 on the different trampolines [16]. This provides a notional idea of the small size of the firebrands, though there is quite a large variation in firebrand sizes. Indeed, the density of firebrands is heavily dependent on the intensity of a fire [8]. In Manzello's experiment with the Douglas Fir trees, the firebrands examined were all cylindrical. The results showed that the average diameter of the firebrands was 3-4 mm, and the average length was 40-53 mm [15]. It is important to recognize, however, that while cylindrical firebrands were examined in this study, in real wildfire events around 70% of firebrands tend to be disk-shaped rather than cylindrical [13].

1.7 NIST Emberometer and PTV

Measuring airborne firebrands in situ remains the most ideal method of data collection, and a task attempted in the most influential project to the inspiration of our own intended methodology. Bouvet, Link, and Fink developed their "emberometer" to not only measure temperature and size of firebrands, but their trajectories and the quantity of fire brands in a "shower" as well [19]. Preliminary tests were conducted to track the trajectory of a marker on a stick, followed by tests where a large quantity of small spherical objects were released and tracked simultaneously [19]. To test this instrument further, a consistent firebrand source was needed. The NIST "Fire Dragon" was created for artificial firebrand generation, producing firebrands that were expelled

from a duct and subsequently tracked by the emberometer [19]. The tools at the Fire Research Laboratory of the National Institute of Standards and Technology, a water pan for ember collection, akin to the one discussed earlier, was used to gain an independent sample of firebrand size for comparison [19]. This provided a good framework for how to produce firebrands for testing purposes in our work, though any future testing with a drone would necessitate different constraints, as the calibration steps would be significantly altered.

The NIST "emberometer" [19] consisted of four cameras pointed at a region in space, called the interrogation volume, where artificial firebrands were expelled from the "Fire Dragon." Because of the differing viewpoints of the four cameras, stereoscopic analysis provides a better understanding of the volume taken up by a firebrand. As the authors describe, the four cameras can construct a boundary around a firebrand in space. Using 3D reconstruction, the volume of the firebrand can be determined [19]. The analysis of the trajectories of these firebrands is an ideal use case for 3D-PTV (Particle Tracking Velocimetry).

3D-PTV can be used to develop a velocity vector field for each firebrand "within a three-dimensional observation volume based on the time-resolved motion of particle tracers" [22]. Algorithms in support of this process have undergone continued development over recent years, but more research is required to increase the accuracy of the output data. This method provides a much more accurate depiction of firebrand size than others not considered volume.

OpenPTV is a particle tracking velocimetry software designed to detect and track the trajectories of particles in an input video, with support for multiple viewpoints (stereo vision) [23]. Being open-sourced, numerous users have contributed to its

development. It consists of a core library written in C, which can be interfaced with via Python, including a GUI-based implementation, PyPTV [23].

1.8 Approach to Characterizing Firebrands

The team has developed a device using stereoscopic cameras, or two cameras facing at a known angle, to take video footage of embers suitable for analysis to determine ember flux, the rate at which burning or glowing firebrands (embers) are transported through a given area. The device collects footage at specific tolerances to isolate bright firebrands against noisy backgrounds and track the movement and number of firebrand particles present using particle tracking velocimetry software. This device is at a size suitable for carrying on an Unmanned Aerial Vehicle (UAV), or drone, so that footage of firebrands being produced on the front of a fire can be collected to determine firebrand behavior. Information collected by the device can be compared between sites and referenced with information about fire spread to identify critical densities of firebrands that relate to probability of ignition for nearby spot fires or structural fires.

1.9 UAV Considerations

UAVs are an emerging technology in military and emergency rescue fields for on-demand monitoring and information collecting services. A UAV can surpass current methods such as human observation, manned aircraft, ground instrument surveying and satellite imaging while reducing human involvement risks [24]. Detecting and predicting wildfires at the early stage can be efficiently and safely executed with the versatility, extended range of vision and portability of UAVs. The capabilities to acquire real time information that can assist in mitigating wildfire disasters would significantly

benefit frontline firefighters, neighboring communities, resources, and ecosystems and can save lives [25].

Researchers have found that using a system with a single UAV to monitor an entire forest wildfire comes with limitations. When compared to systems with multiple UAVs, the single UAV system has a lesser combined payload, less flight time, and difficulties reaching an appropriate altitude. Researchers at Northern Arizona University and University of Alabama proposed a leader-follower UAV coalition formation that covers a wide area [26]. A heterogeneous fleet of drones and fixed wing aircraft is used to perform compound tasks. Multicopter rotary drones can hover over specific areas to collect high resolution data while fixed wing aircraft can fly at higher altitudes in circles with rapid scanning [26]. The fixed wing aircraft have higher computational abilities to serve as the leader of the coordination to relay information to other rotary drones and ground intel [26]. However, this system causes difficulties with coordination and calibration while adjusting to the dynamicity and unpredictability of wildfires. Also, UAVs were prone to malfunctions within the hostile environment around wildfires which would cause communication and coordination issues within the fleet [26]. The developed measurement device focusses on applications as a payload for a single rotary UAV. This allows for focused sustained measurements from stationary points.

The mobility of the drone can optimize firebrand data collection. The primary technologies for 3D mapping are light detection and ranging (LiDAR), synthetic aperture radar (SAR), structure from motion (SfM), and stereo vision [27]. Most of these 3D mapping methods require a fixed wing aircraft with forward motion [27]. Stereo vision is optimized with stability so a multicopter with hovering capabilities best suits our data collection needs. However, the possibility of incorporating a “swarm” of drones

to carry multiple cameras for the stereo vision setup as seen in the work done by Innocente and Grasso [28] would be an interesting avenue for further investigation. This system would have severe limitations though. To have successful stereo vision camera calibration the cameras must remain in a fixed position relative to each other. This is easily achieved by mounting the cameras to a single UAV; however, it would be exceedingly difficult with multiple UAVs in the turbulent environment of a wildfire. Additionally, a single UAV package provides a more affordable approach which can allow for more extensive applications. Stereo vision has been primarily applied to ground vehicles which provide stability and payload capacity UAVs cannot [27]. The aerial capabilities of UAVs have significantly improved over the past years. According to Brien who reviewed the capabilities of electric multi rotary civilian drones, the average endurance is from 10 minutes to 50 minutes, up to 15 kg of payload capacity, 0 to 75 kilometers per hour ground speed and up to 3000 meters in altitude [29]. UAV mobility is advantageous compared to ground vehicles, especially considering the spread and speed of the firebrands and wildfires discussed in previous sections. As inspired by work by the Gemstone team ARM IT [30], the UAV needs to be able to hold some kind of motor assembly to rotate the camera up and down. Other common UAV sensors like infrared or ultrasonic can be mounted without exceeding current rotary drones' capacity.

Another key consideration is wildfire influence regional weather, creating harsh winds from vacuums caused by the updraft of warm air [31]. The drones need to be able to maintain function and stability through the winds and immense heat. FAROS is a drone developed by Myeoung and Jung with fireproof and cooling capabilities without inhibiting the payload capacity or battery life of the drone [32]. They incorporated an

aramid fiber exterior skin that protects integral parts of the drone. The aramid skin has flame retardant and resistant properties with a melting point higher than 500 degrees Celsius and costs around $11.50 per pound making it a very cost-effective option [32]. Since aramid fibers are flexible and compatible with carbon fiber and lightweight metals such as aluminum, they can be easily incorporated onto an existing drone [32]. Inside, a thermoelectric cooler module was installed to regulate the temperature. Within the main frame for the drone, there is an air buffer for the cooling air to flow and cool all the inner components [32]. Ideally, a UAV outfitted with a firebrand measurement device needs to have similar considerations to adapt to wildfire purposes. Otherwise, the UAV must fly at a further distance from the wildfire and firebrands to avoid heat damage to the UAV or measurement device.

1.10 Motivation for Research

The understanding of firebrand dynamics in active wilfire settings is limited [4]. Factors like lofting distance, or firebrand flux can play a role in the probability of spot fires or structural fires being ignited. If firebrand flux, or the number of firebrand particles moving through a unit space, can be measured, then resources can be better allocated during emergency responses in wildfire events.

Perhaps the most technically advanced way of looking at the spread of firebrands is by constructing a 3D model of their spread over time. While this is one of the hardest challenges to overcome in terms of the rigorous computing and modeling required to do so, having this kind of tool would provide the most accurate measurements of the size of the firebrands both at a given instance and as a function of time. This method would also accurately estimate the speed of the firebrands and their direction of motion. By creating a sensor package to measure firebrand flux in situ data

can be acquired to begin constructing such a model. It is vital to understand the distributions of firebrands in field settings as there is not currently a good understanding [4], and constructing 3D models of firebrand spread using stereoscopic vision data would create this understanding.

Chapter 2: Methods

This chapter outlines the experimental and computational methods developed to support the design and testing of a stereoscopic ember-tracking system. It begins with a description of the surrogate firebrand generation process, followed by an explanation of the video capture setup and file processing workflow. Calibration procedures for 3D particle tracking using OpenPTV are detailed, culminating in the development of a static sensor package suitable for future UAV integration. Each section addresses the iterative improvements and tradeoffs encountered while pursuing a low-cost, open-source firebrand flux meter.

2.1 Surrogate Firebrand Testing Progression and Process

A progression of techniques was used to create particles to be tracked by stereoscopic cameras. A discussion of the camera setup is described in more detail later but to understand the cameras first the particles that the cameras were measuring must be understood. While many of these particle simulations were rudimentary, they provide the necessary benchmarks for successful particle tracking.

The primary goal of particle simulation is to create repeatable and quantifiable particle streams to train the PTV software on. First, marbles were used for tests as known objects that have a predictable path of motion. A drop tower was constructed such that when a plate was removed, marbles would fall vertically towards the ground. Footage was taken of these marbles against a black background for clear contrast. These tests enabled the camera shutter speed to be tested and adjusted so that there were sufficient frames of footage so that the marbles could be tracked from frame to frame. If shutter speed was too high, then files were too big to handle efficiently and

the recording software glitched. However, with lower shutter speed the marble moved too far in between frames to effectively be tracked. Time of exposure was also tested during these tests. With high time of exposure, the marbles became too blurry and would not be visible. These parameters continued to be modified for later tests, but these initial tests served as practice for operating the cameras and finding faults in the process of recording and converting files with low stakes. A ramp setup was also constructed for marble tests, though it lacked a high contrast background, a controlled mechanism for releasing the marbles, and a way to mount the cameras while facing the ramp, so this setup was quickly set aside in favor of the marble drop tower.

Moving past initial marble drop tests, to represent firebrands moving more accurately in an unpredictable way while being lifted by air currents a fan-duct mechanism was constructed to launch particles. The fan-duct mechanism was first used with Styrofoam pieces to test the operation of the fan and the area that particles would be expelled into. Styrofoam pieces were chosen to represent firebrands as they are light and could loft in the air in a similar fashion to firebrand lofting. After initial tests to determine the workings of the fan-duct mechanism and the visibility of Styrofoam pieces, tests moved towards using artificially created embers instead. While Styrofoam by size and wight could represent firebrands, the glow of a firebrand is important in spotting firebrand movement. While this contrast makes firebrands more visible, the light produced can easily become too bright and cause particles to appear to blend together. Artificially produced firebrands were loaded into the fan-duct device and the air movement of the fan caused them to glow and be expelled out of the duct opening so that the cameras could capture footage.

Wooden dowels were used to produce the firebrands in these tests. Previous protocols have called for oven drying the dowels to ensure even combustion when ignited. This step was skipped in producing embers for testing due to time and logistical constraints. These tests track a variety of firebrand particles being expelled from the duct, so having moisture levels not homogenized across the particles is acceptable for the tests. To produce firebrands, batches of approximately 100 grams of dowels were measured out into aluminum trays. After positioning the fan-duct apparatus underneath a vent hood with protective curtains in front of the duct opening and fixing the cameras in place the dowels were placed in a wire mesh basket and burned on a propane burner for 30 seconds as shown in figure 5. After another 30 seconds the burning dowels were stirred around to ensure even burning. When the flames extinguished the firebrands were placed back in an aluminum tray. From here the firebrands were dumped into the top of the duct where the firebrands settled approximately one foot below the duct opening on wire mesh inside the duct. The duct opening was adjusted to face parallel to the ground and once cameras were recording the fan was turned on. The fan propelled glowing firebrands out of the duct for up to 10 seconds. After each test, the footage was examined and parameters adjusted as needed.

Figure 5. Team member Leonello Castro Cillis preparing artificial firebrands from dowels.

Following the initial testing setup, two more iterations were used. The second iteration as shown in figure 6 used a different duct tubing as the first one used was damaged by the firebrands. The third iteration was a more compact duct design with a wider opening at a 45-degree angle as shown in figure 7. This apparatus allowed the firebrands to be launched at a higher density though for only approximately three seconds each time.

Figure 6. An ember shower produced by the second iteration testing setup.

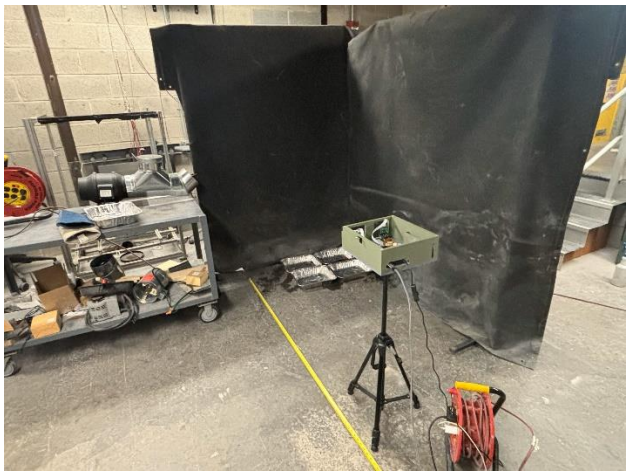


Figure 7. The third iteration testing setup with the blower duct on the left.

The final method of firebrand production for testing was with Christmas tree burns as shown in figure 8. The Christmas tree burns simulate high density firebrands with a background more akin to wildfires and for longer durations. Christmas trees

were purchased and stored for three weeks preceding the burns. Trees were stored with different treatments, watered, and not watered, to represent possible drought conditions. The burns produced firebrands adjacent to bright flames testing the device's ability to capture footage in lower-contrast environments.

Figure 8. A Christmas tree burn test being filmed by the stereo vision cameras.

2.2 Video Capture and File Storage Integration

To develop an accessible and affordable firebrand flux meter, it is critical to stick to open-source components. Both hardware and software must be open-source, so that the instrument has a reasonably low barrier of entry for usage for a wide variety of audiences. A Raspberry Pi computer was the first choice for footage capture due to its low cost and design for open-source robotics/electronics projects. The team initially attempted using a Raspberry Pi computer, but the frame rate was too low for the target application. The next step up is the Jetson Nano computer. The 4 GB model can record 2560x800 videos at 50 frames per second (using the OV9281 cameras), which is

sufficient for this project's applications. To use this computer, one must work within NVIDIA's Ubuntu-Linux style file system and NVIDIA's packages for the Jetson Nano. This presents a small barrier of entry to the open-source platform, as UNIX/Linux background is a prerequisite for fast, relatively painless operation of such a system. We deem that this is a preferable barrier of entry for an equitable instrument, as learning Linux/UNIX is likely less expensive (and more valuable) eventually, compared to spending more on high end cameras, such as IDS cameras.

In preserving the nature of this being an open-source ember flux meter, "open-source" cameras need to be used. This translates to inexpensive cameras that easily integrate with the Raspberry Pi/Jetson Nano. The OV9281 stereo camera module by ArduCam fits this qualification. These cameras have parameters that can be tuned with v4l2 controls via command line, which is not possible with COTS webcams. When tested, 4K Logitech BRIO webcams had low framerates when recording, confirming the value of the OV9281 stereo camera modules.

Figure 9. Setup exploring the use of brio webcams.

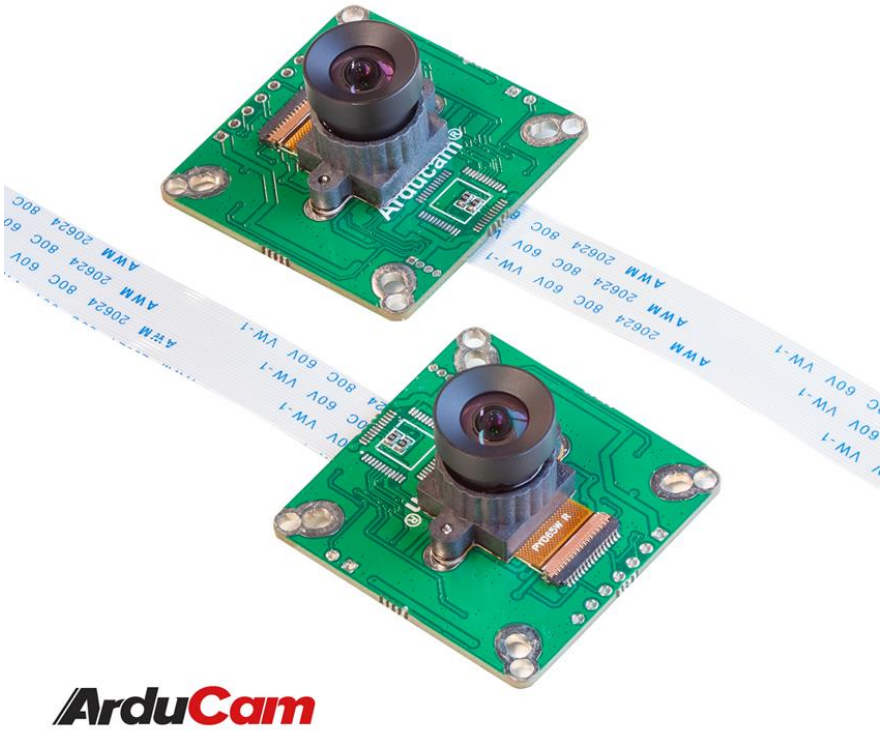


Figure 10. OV9281 camera modules by ArduCam.

By trying a variety of options, the team determined that a given GStreamer pipeline was the only way to record the video from the Arducam OV9281s at sufficiently high framerate (50 fps). Using only OpenCV or FFMPEG, video/frames can be recorded at low frame rates. Other scripts for recording videos (such as V4L scripts) were investigated, but these were unsuccessful for many reasons. The only long-term working video recording option was the GStreamer pipeline. However, an issue with the GStreamer pipeline is that recorded video gets bleached, meaning images appear to have lower contrast and higher exposure when displayed. This is possibly due to video encoding problems with the Jetson Nano and may be fixable with a patch from Jetson Nano forums.

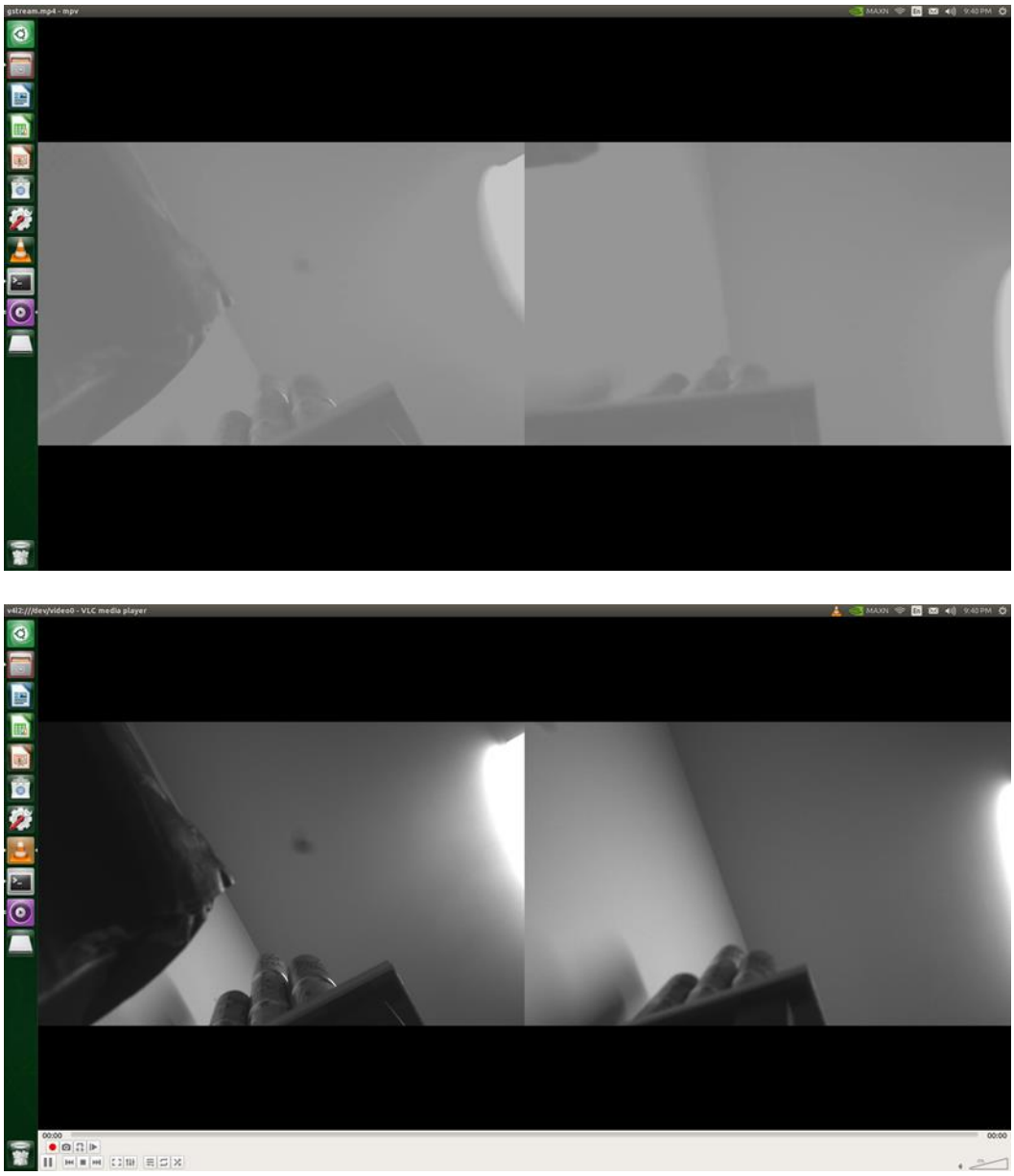


Figure 11. Bleaching from GStreamer (top) compared to typical images (bottom).

It is readily apparent that open-source software/hardware has complications associated with troubleshooting. The “tech support” for these issues are online forums, instead of dedicated staff. As a result, working with the open-source hardware/software presents a tradeoff between financial cost and the support available to the research team member required to work with the open-source component. The team accepted the

bleaching cost as an image background subtraction resolves this issue for video processing. In future iterations, it would be desirable to apply this patch, so that the dynamic range of the cameras is expanded.

This project utilized OpenPTV [23] for particle tracking analysis. This software must first be run on a test cavity, a provided set of test files which ensure that the packages are installed correctly and that the software functions as intended, once it is installed. A GStreamer pipeline alongside additional camera setting parameters (if applicable) are run in the terminal, capturing a video file from the cameras. This video is then transferred via USB drive from the Jetson Nano to a full-size computer, for ease of use. A MATLAB script preprocesses the frames, applying background subtraction. The background image is computed as the average of the video frames, with this then being subtracted from the current frame to extract moving features. Such a technique is effective at removing stationary bright spots from the background which would otherwise be at risk of being identified as particles. This script exports the frames as .tiff image files. Following Bauer [33], the image files are then stripped of their file extension, using the Bulk Rename Utility. The images are transferred into an environment for running PyPTV (GUI for OpenPTV). Again, following recommendations by Bauer [33], we opted to run OpenPTV in a Linux system, specifically an Ubuntu Virtual Machine (VM) on a Windows laptop (using Oracle's VirtualBox), for convenience. Note the following potential issues users may encounter when running on Linux:

- Editing OpenPTV's .ori files in Ubuntu required opening the files in a text editor and saving, as the PyPTV GUI's "save" button was inconsistent in its functionality.

- If processing fails, ensure a “res” subdirectory exists within the main project folder. This is likely a more general problem outside of just Ubuntu.

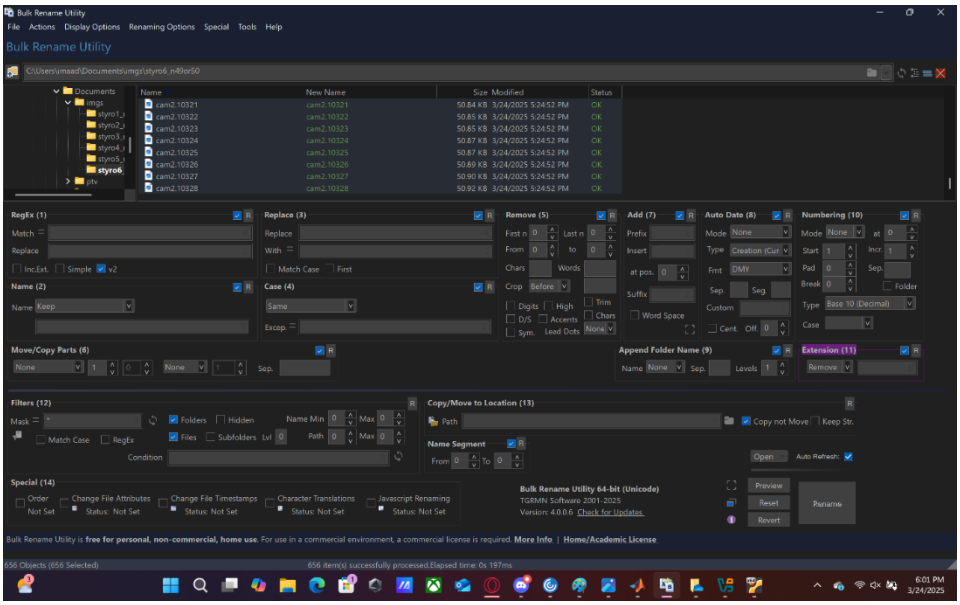


Figure 12. Bulk Rename Utility used to remove TIFF file extension.

These issues may arise from the VM setup or installation of python packages. This further illustrates the tradeoffs discussed earlier. While Linux is, in many ways, less user-friendly than Windows or Mac, it is preferred for open-source applications because it allows unrestricted access to the operating system’s source code, supports a wide range of programming tools and libraries natively, and avoids licensing fees or proprietary restrictions that can limit customization and redistribution. Convenience (dedicated user support and professional documentation in particular) comes at a financial cost as these features require ongoing investment in development and personnel that is ultimately passed on to the user. There is merit in the recommendation to use PyPTV on a Linux machine, as issues arose when processing runs on Windows-

native PyPTV, notably frequent crashes during run processing. The cause of the crashes is unknown. The Windows file system is also less convenient for running PyPTV (one cannot type "pyptv <directory>," one must reference the find and reference the actual pyptv script with the run directory), which was mitigated by running on the Ubuntu VM. PyPTV is installed on the VM using the standard installation process on the website. It is run using the same Anaconda python environment as specified by the OpenPTV website.

Table 1. System parameters of ember flux meter. Data specifics on lenses, camera parameters taken from [34].

Processor	Jetson Nano 4 GB
Hard Drive	512 GB microSD card (later switched to 64 GB due to boot problems)
Cameras	2x OV9281 with CamArray Hat
Lens	2x M12 2.8 mm focal length lenses
Pixel Size (micrometers)	3x3
Camera Array Size	1280x800
Camera FPS	50
OS	NVIDIA Ubuntu

Necessary camera parameters as seen in Table 1 (pixel size, back focal distance, sensor resolution) for the OV9281s were fed into OpenPTV so that the software can properly account for the distortions unique to any given camera when they capture images. OpenPTV must first be calibrated to create a mapping from any given 2D point in each camera's view into 3D space. One method of doing this is providing calibration images. These images must contain a "calibration target," an object with clearly visible points at known relative coordinates. These coordinates are stored in a text file, where

each line represents the X, Y, and Z coordinates of a single point in 3D space, relative to the object's origin. Several iterations of calibration targets were attempted, though ultimately the most effective was a wooden target consisting of 7 vertical columns with black dots along the front, mainly due to it being large enough to fill a sizable portion of the interrogation volume, leading to an accurate calibration across the image. After taking images of the target, Microsoft Paint was used to mark each point with white dots, while the rest of the image was filled with black. This enables OpenPTV to detect all of the points with no possibility of false detections. These black and white images are named in accordance with OpenPTV's conventions [33] and placed into the project's "cal" directory.

Test footage was captured of both falling Styrofoam beads and artificially produced firebrands. This footage was preprocessed in MATLAB as described earlier before being processed in PyPTV.

2.3 Camera Calibration

Calibration is a critical step in reconstructing 3D paths. Calibration of the system determines orientation parameters, which allows for translating 2D stereo image data into an accurate 3D reconstruction. From the information obtained in the calibration setup, the OpenPTV software can reconstruct the original 3D positions of particles detected by the cameras from their representations in 2D image space. Various techniques were used to create known, physical 3D points for calibration with varying degrees of success. First, checkered grids were printed and hung up on flat surfaces in front of the cameras, though these were flat and as such provided an insufficient calibration of depth. Later, tape pieces were adhered to protective tarps during testing. While tape pieces allowed for a larger calibration target than printed grids, the tape did

not have high contrast with the protective tarps and were difficult to see. There was also an insufficient quantity of points present for an effective calibration. Using the NIST calibration system as design inspiration, smaller prototype calibration systems were constructed out of boxes. These prototypes had good contrast and depth but were small and it was difficult to get precise measurements of point coordinates. A metal triangle frame tower was purchased, though its shape made it difficult to orient so that all points were visible and covered enough of the interrogation volume. It was also not as sturdy as desired, and each point did not lie at a clear location, but rather at the place where two cylindrical beams made contact with each other, which is hard to distinguish from most vantage points. Using standing metal frames, grids were drawn with markers on plastic sheets, allowing for increased depth and coverage of the interrogation volume by the calibration points. These points were functional when used in OpenPTV for calibration, though inaccuracies were present due to movement of the plastic sheets and flaws in the lines drawn. A pegboard was mounted on each of two frames, angled in front of the cameras, and leveled by using a laser alignment tool. The pegboard setup was initially adequate; however, it lacked the necessary contrast to identify all particles and was thus scrapped after several tests. Finally, the last calibration technique involved constructing a triangular prism like the one used in the NIST calibration setup. The structure was constructed with 2x4 wooden frames in which black dots of 1 inch diameter were placed at known locations, later to be inputted into OpenPTV during calibration. These dots were leveled and aligned using the laser alignment tool and precisely measured out to the predetermined distances. The most significant reason these black dots were chosen was due to their contrast with the background, allowing the cameras to easily distinguish them, in turn facilitating the calibration procedure.

After taking still images of the calibration target with the cameras, the images were first processed in MS Paint to mark spots, before being transferred to OpenPTV for calibration. Each point was assigned coordinates in 3D space by providing a text file containing the X, Y, and Z coordinates of each point in 3D space relative to an arbitrary, fixed point in the calibration target's reference frame, called the origin. The origin was chosen to be the back of the calibration target at ground level. Four predetermined calibration points, one in each corner of the target, were identified by manually clicking on their locations in each image's representation within the PyPTV GUI. Calibration is then run, which provides the 3D coordinates of all the black dots on the calibration target in world space and each camera's orientation (position and rotation in world space). Once calibration is complete, OpenPTV automatically generates an internal calibration file, which informs it of how to map any 2D point on each image into 3D space for any video that is passed to it henceforth.

Figure 13. Left camera image of calibration target.

Figure 14. Right camera image of calibration target.

It is wiser to physically measure the (x,y) coordinate of each post of the calibration target, as opposed to measuring angles of the target sides to obtain (x,y) coordinates of the posts. Angle error likely has a larger impact on the generated coordinates, as opposed to actual cartesian coordinate measurements. After measuring (x,y), these values can be put into a MATLAB array and then the target file can be generated. Each calibration point requires an arbitrary integer index; we decided to designate the top left point as "1", proceeding sequentially rightwards across the row (to "2", then "3", and so on), wrapping around to the leftmost point in the next row down once the current row is complete. This results in the bottom-right-most calibration point having index "n," where "n" is the total number of calibration points.

The key indication of a successful calibration is that each point in each image is correctly matched to its corresponding index in the text file. The cameras themselves also have uncertainty in their orientation, i.e., their physical position and/or rotation may be imprecisely known. When attempting to achieve an accurate calibration, one must manually adjust the inputs for camera position, angle, and back focal distance parameters until the points are all properly matched. A general procedure for doing so is as follows:

1. Load images.
2. Orient with file (after having done manual orientation).
3. Show Initial Guess.
4. Update ori file. Modify the camera location parameters (first row), camera angle parameters (second row) and back focal length, as necessary.
5. Either Show Initial Guess or do 1-3 (this sequence reduces clutter on the screen).

6. Repeat 4-5 until a reasonable match is reached between the calibration targets in the image and points generated by the initial guess.
7. Sortgrid
8. Raw Orientation

To determine the interrogation volume, or the space within which particles can be reliably tracked, we used the field of view and depth of the cameras, as well as the geometry of the triangular calibration target. Based on this, a usable interrogation volume was defined and marked as a taped rectangle on the floor. Within this volume, we tested the system using two types of particles: Styrofoam and embers. For the Styrofoam particles, which are slightly larger than embers, we conducted two trials: 10 and 50 particles of Styrofoam. We dropped all the particles at once from outside the camera frame into the interrogation volume. We repeated this process using 20g and 60g of wooden dowels which were then burned until only embers remained. These embers were dropped into the same interrogation volume, with a large water pan at the bottom for safety. The cameras took both videos of both at an exposure of 30 for the Styrofoam and 15 for the embers. The exposure was adjusted to optimize the visibility of the respective particle. The next step is to process these images for particle tracking and analysis.

Once the calibration is complete, the cameras are ready to be tested by observing ember particles. The accuracy to which OpenPTV works can be verified during this process. The first step is to have the cameras record the flight of the embers as they come into their field of view. The number of embers flying in needs to be known beforehand. The software can then process the data and be able to obtain how many of the ember particles it can identify. Ideally, OpenPTV will detect the same number of

particles as were used in the test while also being able to differentiate one particle from another.

The next step is to analyze the ember trajectories. As these particles fly, they experience significant perturbations to their motion and are expected to constantly change speed and direction even through small time intervals. The velocity trajectories can be obtained by observing the time steps between each frame from the video of the ember motion. The velocity of any given particle at a given time is the change in position between that time and the next divided by the elapsed time. For a given frame i in the video, the velocity of a given particle is found using $\overrightarrow{v_{i,p}} \approx \frac{\Delta\overrightarrow{r_{i,p}}}{\Delta t_{i,p}}$, where $\overrightarrow{v_{i,p}}$ is the velocity, $\Delta\overrightarrow{r_{i,p}}$ is the change in position, and $\Delta t_{i,p}$ is the elapsed time, all at the frame i for particle p from the video.

While this algorithm could provide accurate results without introducing much complexity to the problem, it is only useful if OpenPTV can independently distinguish each particle from all other particles throughout the full firebrand trajectories. Therefore, the framerate needs to be high enough so that each frame is similar enough to the previous frame. This allows PTV to have a better chance of tracking each of the particle's motion throughout the trajectory.

It is important to note that PTV and Computer Vision work simultaneously to obtain accurate particle detection. Epipolar geometry is the mathematical tool that PTV uses in the calibration of the cameras as well as in tracking the particle trajectories. Epipolar geometry first assigns a reference frame to both the right and left cameras. The distance between the two and their relative rotation is known, and this information is passed onto the calibration. When a calibration target, such as the one described above, is used, the points on the target are registered by both cameras. However, the

same point in space has different coordinates in the right and left camera reference frames. The key step to successful calibration is to have multiple of these points that span across the interrogation volume, as determined by the camera setup. Since the coordinate of each of these points relative to the cameras is known, the calibration is complete. The cameras can then be used to obtain the location of any point in space just from the calibration information, even if the location of the point is not known a priori.

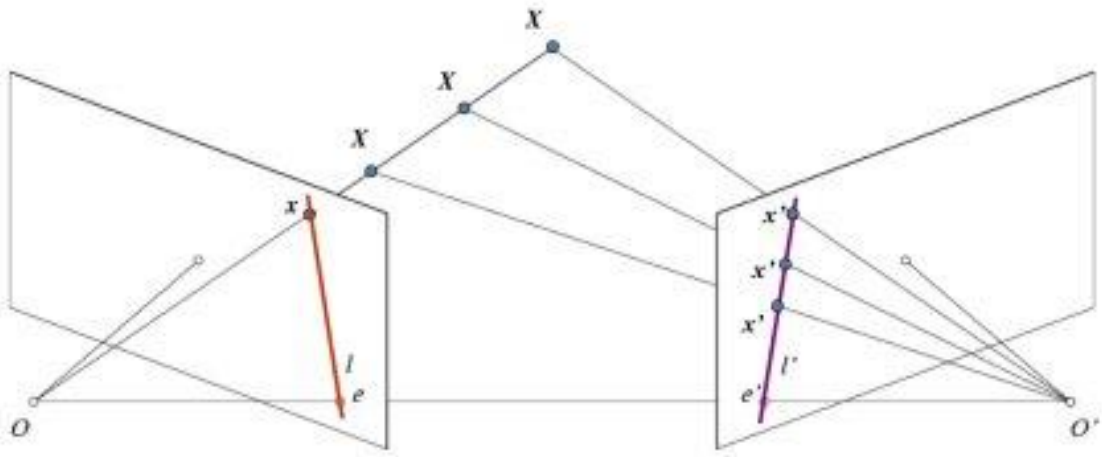


Figure 15. Epipolar Geometry Diagram representing epilines and epipolar planes [35].

The figure above shows a diagram of the epipolar geometry configuration of a two-camera system. The two origins represent the optical centers of their respective cameras, right and left, and the two planes are the focal planes for the cameras, whose distance from their optical center is determined by their camera focal length. The epipoles are also shown. The epipole for each camera is the point on the focal plane that is colinear with both optical centers. This parameter is consequential for understanding how to map a point in space detected by one of the cameras onto the other camera. As the diagram shows, if an object is located at point *X* in 3D space, the

left camera would only be able to retain its position to anywhere along the OX line. The l line, the epiline, on the left focal plane connects the left epipole to the point x on the left focal plane, which is the projection of point X onto the plane. As far as the left camera can tell, the object is not moving unless the l line is changing, which is what would happen if the object moved along the OX line only.

However, each point on the OX line can be precisely mapped to a unique point on the focal plane of the right camera. The mapping of OX onto the right focal plane conveniently forms another line, denoted by l' on the diagram above. Unlike for the left camera, where l does not change along the OX line, l' epiline for the right camera is changing. Having such a two-camera system in such a configuration would allow for motion to be captured by both cameras. Using the mathematics behind epipolar geometry, one can retain information about the position and velocity of objects captured by both cameras simultaneously.

The calibration results for the open-source ember meter using the wooden calibration setup are shown below.

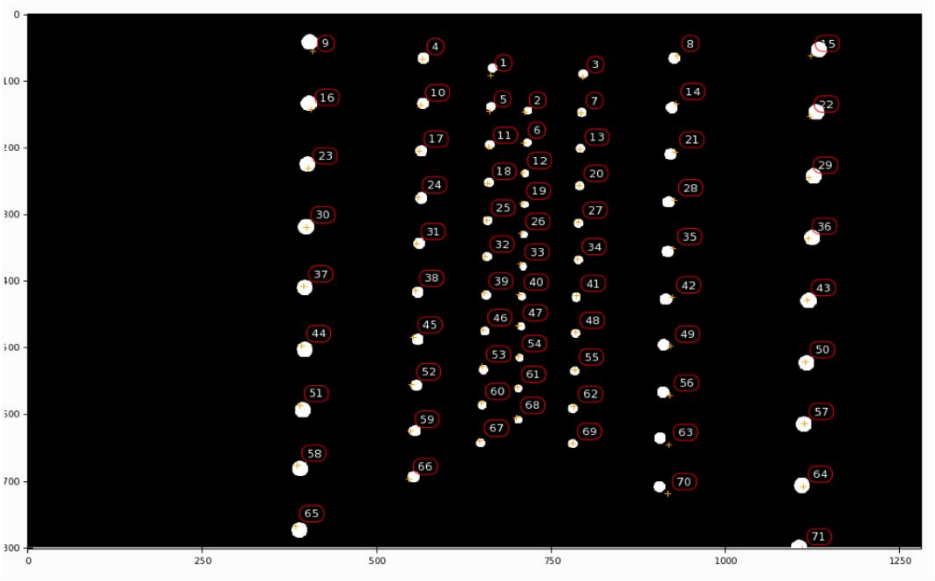


Figure 16. Camera 1, calibration image with calibration results in PyPTV. There is decent agreement between the orange crosshairs and white dots, however it is noticeable that this is not perfect, particularly near the side edges of the picture.

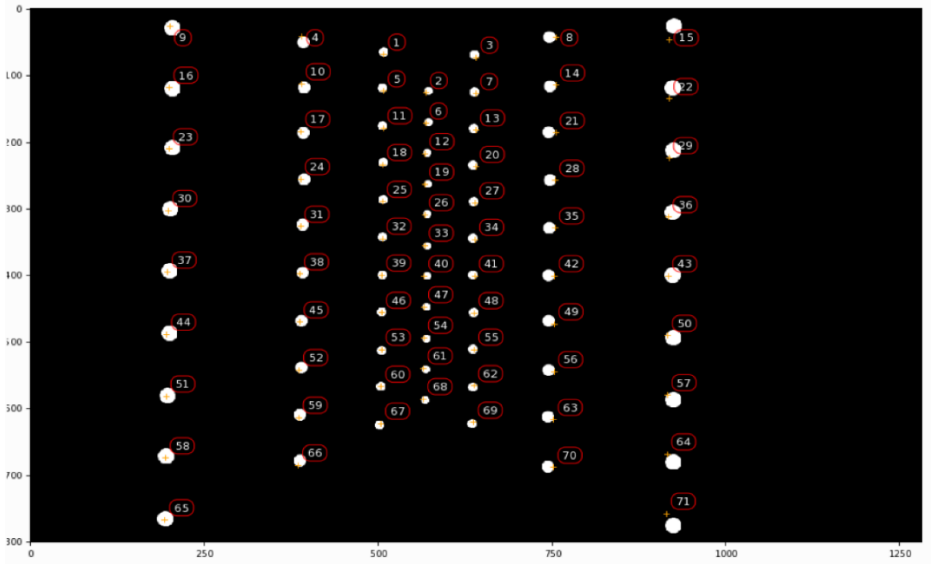


Figure 17. Camera 2, Calibration image and results in PyPTV.

2.4 Sensor Package Development

Before the static sensor package was developed, an adjustable testing device was used to hold the cameras when capturing footage. This adjustable device needed to ensure that the cameras were aligned on the same axis and angle and remained level. The device also needed to allow for lateral adjustments to test various distances required for optimizing stereoscopic performance. The device was used to capture footage only in a laboratory setting and needed to hold just the Arducam cameras and ribbon cables connected to the Jetson Nano, which was used for file storage and camera actuation. A 40mmx40mm aluminum T-slot rail was used as the base for the device, with 3D printed mount bases on the rail. One base was stationary while the other slid along the rail's length, providing flexibility in camera separation distance. Rigid mounts were designed to fit around the rail and were secured using foam rubber tape, providing a compressed, friction-fit connection. The mount bases had MG995 servo motors on top, with the camera holders attached to the servo shaft. The effective range of motion for the camera distances and Arduino actuation of the servos provided an angular margin of error of 1.5 degrees. The camera mounts were designed to fit the lens holder and be secured using 4 m^3 bolts on each corner. These were then fixed to the servo arm with a cutout on the bottom of the base of the camera mounts. The mounts were 3D printed as well for convenience and cost. Later, the servos were substituted for Thorlabs RP005 precise manual rotation stages which allowed for precise angles and locked orientations. This setup allowed for many preliminary tests to be carried out, though ultimately the dynamic nature of the adjustable rail setup led to issues with the ribbon connections from the cameras to the computer.

After settling on a distance of 12.5 cm between the cameras using the dynamic setup that gave a large enough field of view, a static sensor package enclosure was 3D printed to fit all necessary components. This package serves as both a permanent testing setup and as a prototype for a sensor package to be placed on a drone. The package contains two OV9281 cameras mounted in holders on top of Thorlabs precise angle mounts, the Jetson nano computer and CamArray Hat PCB, and a battery pack to power the Jetson nano. Included in the design are cutout access to the ports of the Jetson nano and windows for the camera lenses. There is also a mount cutout for use on a tripod in lab settings, or on a drone if the package is inverted. Extra space in the enclosure can be used to expend extra wires or memory devices. The Thorlabs precise angle mounts are fully adjustable by opening an Allen key-lock so different camera angle offsets can be used in different scenarios. These mounts use four bolts and nuts at the base of the sensor package box to provide a secure and stable camera setup. The camera mounts were now redesigned to have 4 holes at the base so it can be screwed onto the Thorlabs angle mounts itself which already came with threads for this purpose. The horizontal flat design was chosen to enhance stability when mounted on a tripod and to accommodate potential future drone applications. The top of the setup can be fully enclosed with a screw-in cover for added protection. Camera covers can be inserted into slits on the front of the camera windows when not in use, preventing dust buildup and potential damage. All components can be securely fastened using nuts on the underside, aligning with custom pegs and holes. Detailed diagrams illustrating the design and assembled package are provided below.

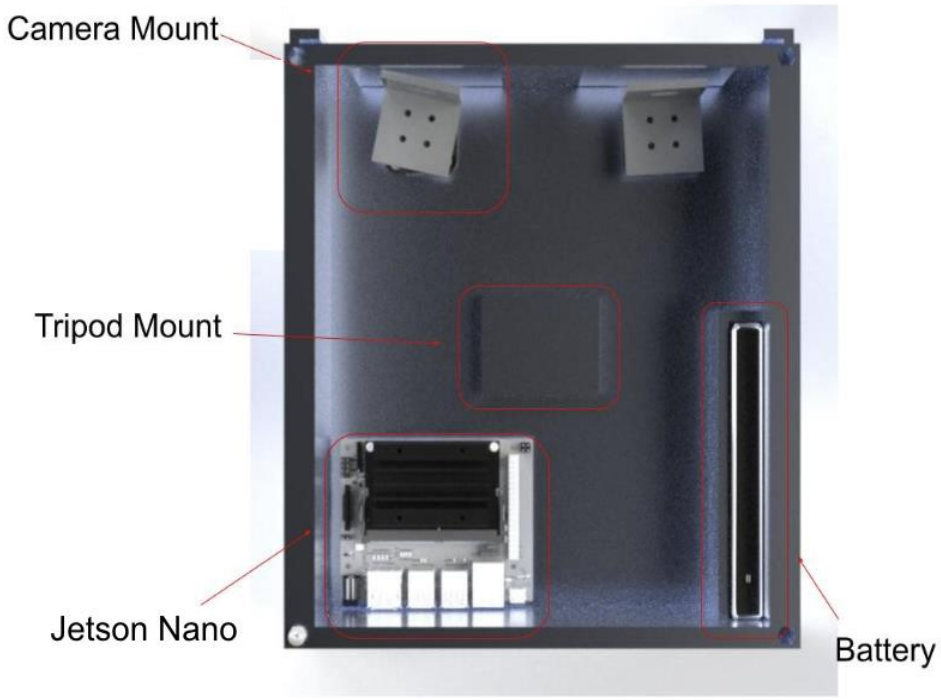


Figure 18. Sensor Package CAD top view

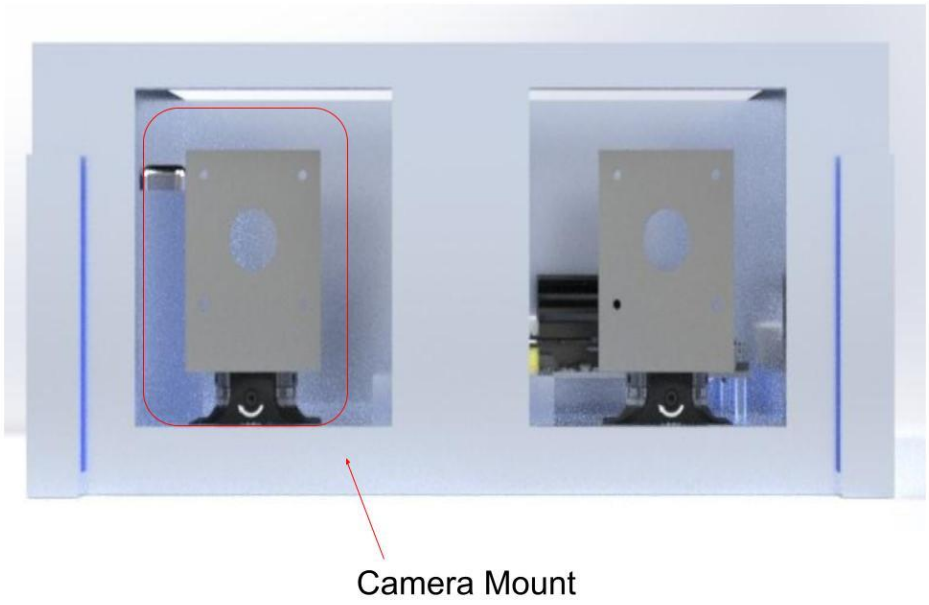


Figure 19. Sensor Package CAD front view.

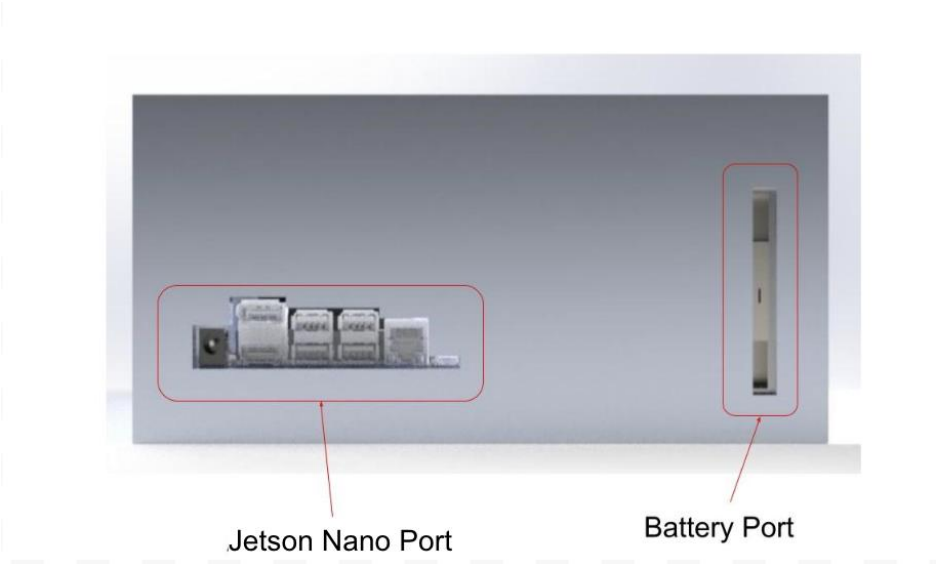


Figure 20. Sensor Package CAD side view.

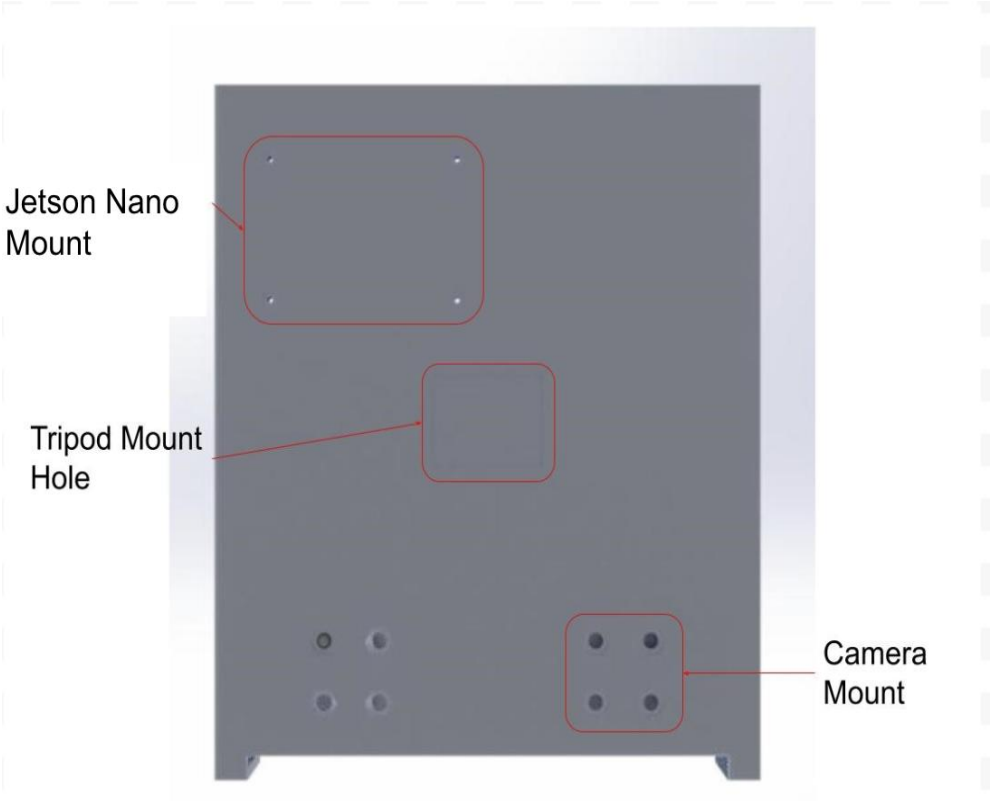


Figure 21. Sensor Package CAD Bottom View

Figure 22. Stereo camera setup

Figure 23. Alternative view of the ember flux meter.

Figure 24. ThorLabs RP005 Rotation Stage (Camera angle base mount)

While drone testing has been removed from the scope of this project, the prototype sensor package is ready for adaptation as a drone payload. After conversations with the UMD UAS Research and Operations Center (UROC), a Harris Aerial H6 drone with 50 x 50 cm clearance landing gear would be used for this package and capable of carrying it. For the integrity of the sensor package, the 3D printed material on the drone would have to be kept 6 m away from open flames. This ability to successfully capture footage at a 6 m distance without sustaining damage to the sensor package was tested during the Christmas tree burns.

Since mounting onto a standard gimbal is not feasible, a custom gimbal would be required, which can be quite expensive when sourced from specialized manufacturers. Instead, the sensor package can be directly mounted onto a drone using an M8 screw attachment at its base. The bottom plate can be modified to accommodate different drones and compatible mounting options as needed, allowing for a seamless.

transition from a static setup to a drone payload. The current sensor package weighs 1.98 kg, while the H6 drone has a maximum payload capacity of 8 kg. Given that many industrial drones support at least 5 kg of payload, this setup should be widely compatible.

The main objective of this project was to develop a versatile and cost-effective sensor package that can adapt to many applications. This system is designed for easy calibration and testing by mounting it onto a tripod for controlled experiments or attaching it to a drone for in-situ measurements of embers in dynamic environments. The sensor package demonstrates flexibility in deployment, making it suitable for diverse research scenarios, including wildfire monitoring, environmental assessment, and atmospheric data collection. After sourcing and assembling the components, the total cost of the package amounted to $536.69, making it an affordable solution for both field and laboratory use. The full final bill of materials for the sensor package is listed in the appendix. Future improvements could include enhancing data transmission capabilities, integrating additional sensor types, and improving the durability of the system to withstand extreme conditions.

2.5 Instrument Benchmarking

To demonstrate system functionality, two sets of tests were conducted with dropped particles for tracking. The first set used n = 10 and n = 50 particles for demonstrating general particle tracking. The particles were dropped from ~ 2 meters into the valid interrogation volume covered by the cameras. Video of the particles falling was recorded and processed in OpenPTV/PyPTV, before trajectories were generated in Jupyter notebook, based on Liberzon's original "postptv" program.

Pictures of the final resting positions of the Styrofoam were taken as well for qualitative analysis of trajectories generated by OpenPTV/PyPTV.

Figure 25. View of testing setup for Styrofoam drop tests.

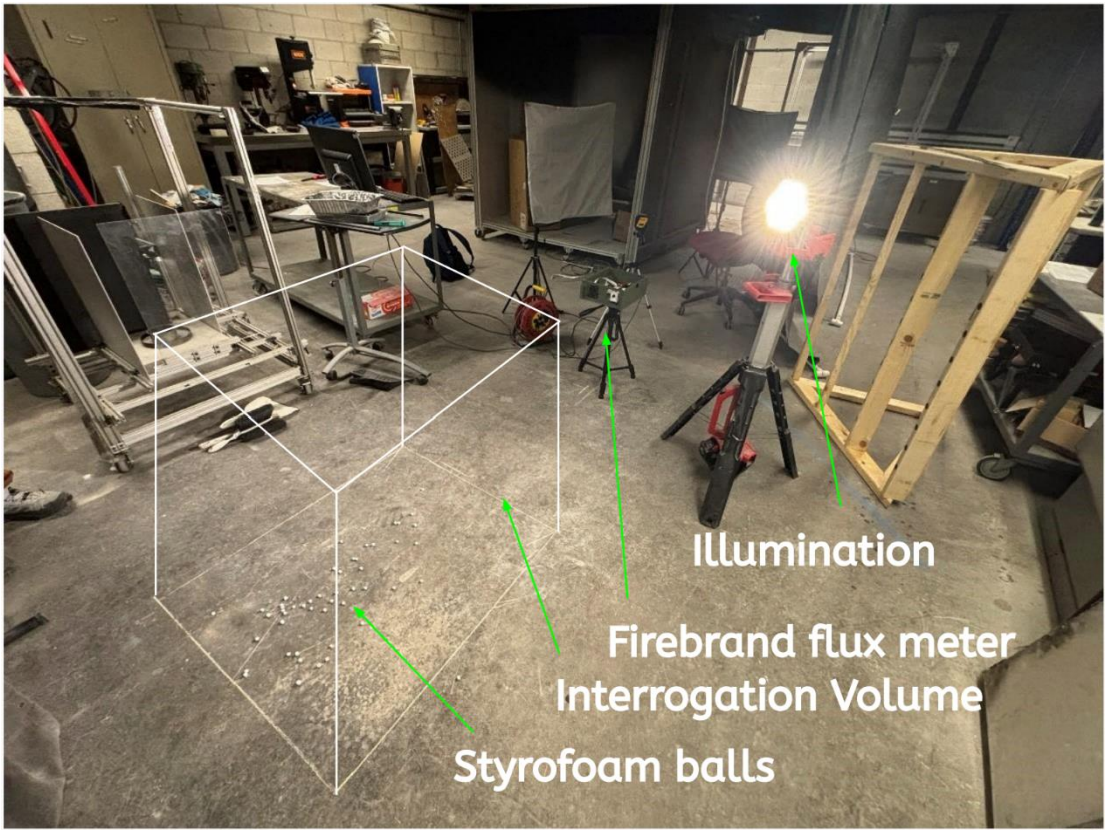


Figure 26. Labeled view of the testing setup for the Styrofoam drop tests.

In these tests, background subtraction is critical for isolating the moving particles from a stationary background. This makes particle tracking feasible as the particles pass through this “filter.” The background image is the time averaged image from a video. That is, each pixel intensity in the background image is that pixel's average intensity over all the frames. The additional criteria of multiplying the image elementwise by a logical array of pixel intensities over 5 (0-255 scale) was needed to remove artifacts generated by background subtraction alone. Some images for background subtraction are presented below.

Figure 27. Example of background image for left camera.

Figure 28. Example of background image for right camera.

The effect of background subtraction is shown in the following figures, isolating the particles for tracking. It is also shown that bleaching does not matter, as this effectively shifts the intensity of the background image.

Figure 29. Image before background subtraction and additional processing (leg blacked out).

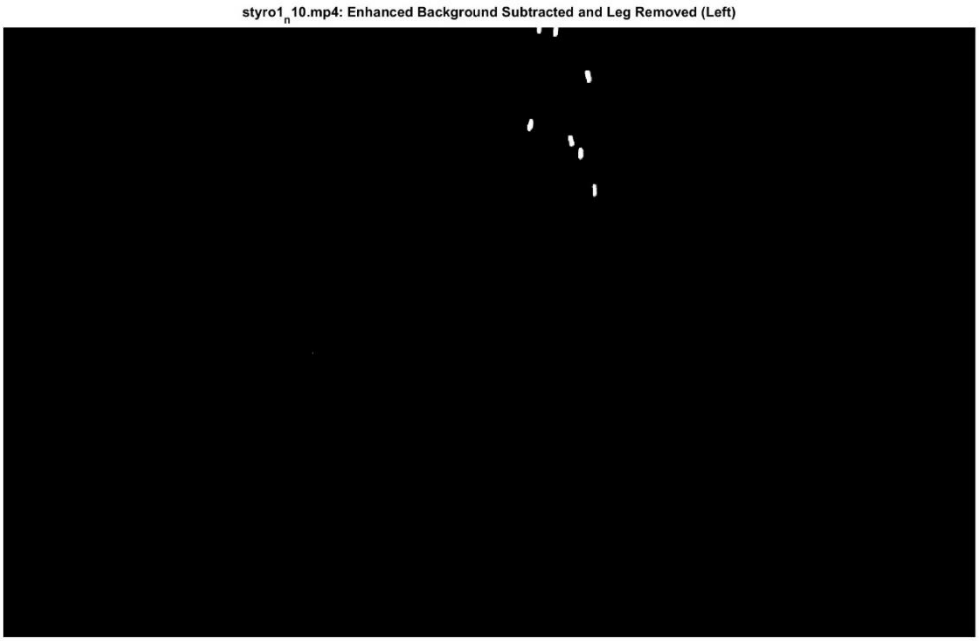


Figure 30. Background subtracted image, with enhanced brightness for visibility (brightness enhancement not applied to images).

Chapter 3: Results

3.1 Processed Images

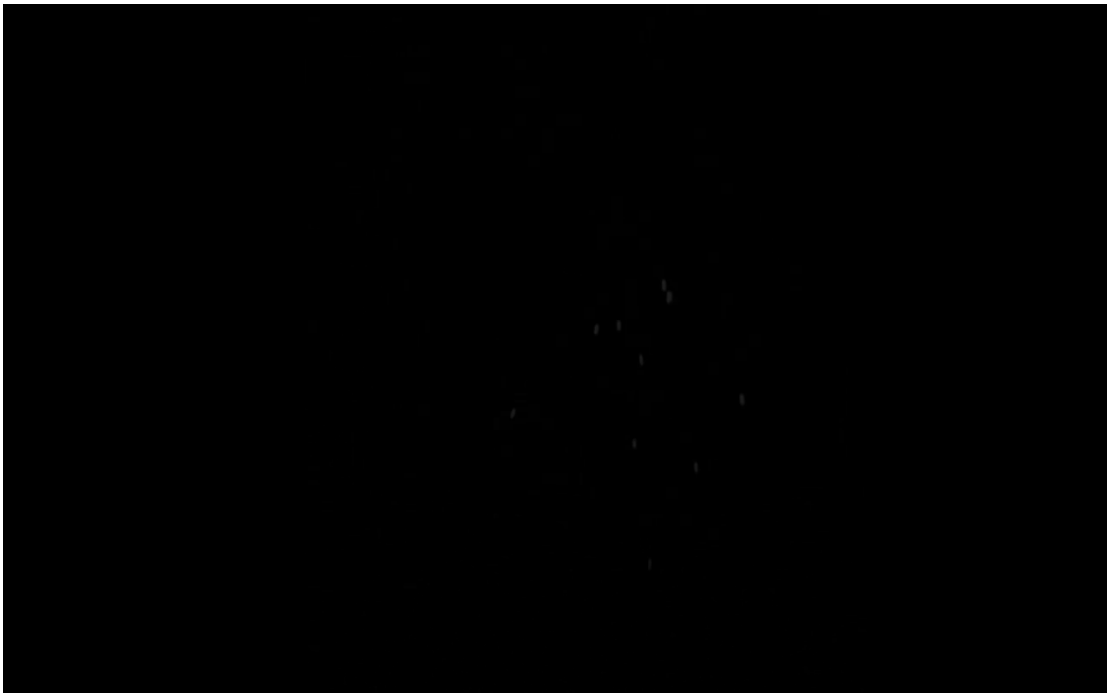


Figure 31. Camera 1 (left), frame 375 of Styrofoam video 1

Figure 32. Video 2, Camera 1 (left) frame 266

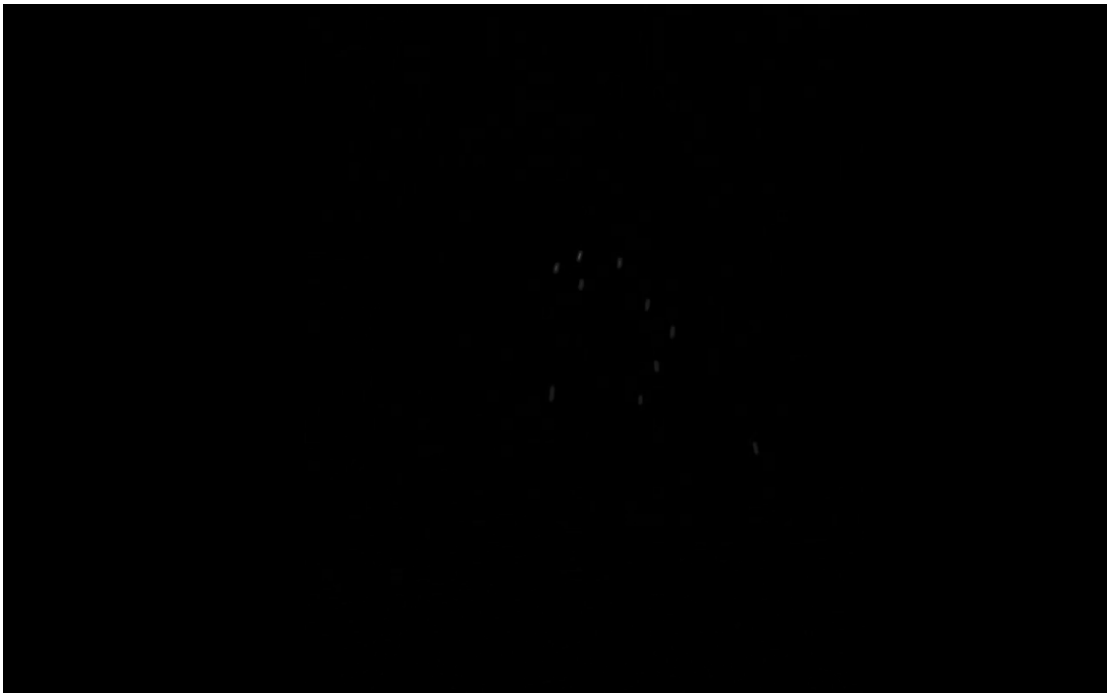


Figure 33. Video 3, Camera 1 (left) frame 259

3.2 Comparison of Measurements to Estimated Parameters

We present the ending trajectories of the Styrofoam bead test results, using the ending positions of the beads as a qualitative verification metric. While we initially use the last instance of particles in individual directories, we switch to using particles detected and matched in the last frame. The last particle instances vary in time for various trajectories, making verification a more challenging task. Using the particles detected and matched in the last frame, we see that OpenPTV/PyPTV's prediction is quite close to what was physically seen in the lab.

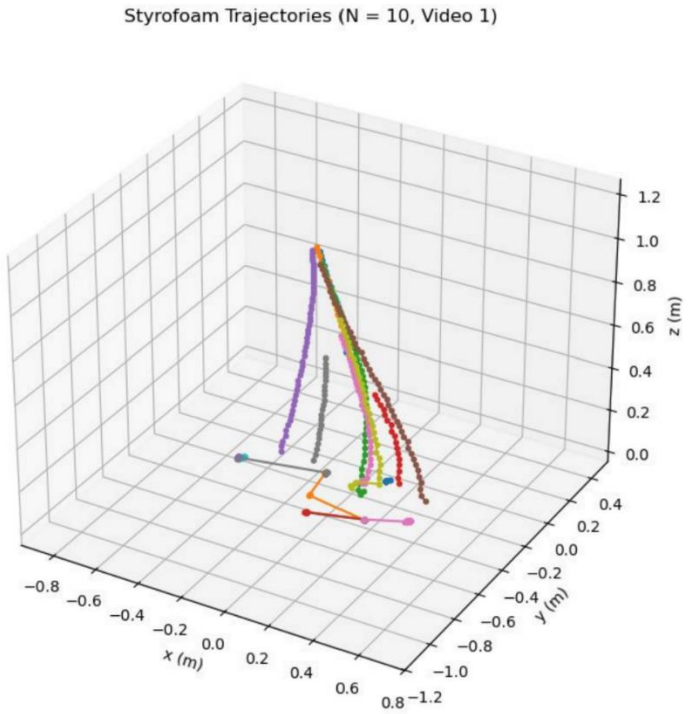


Figure 34. Trajectory of first Styrofoam drop video.

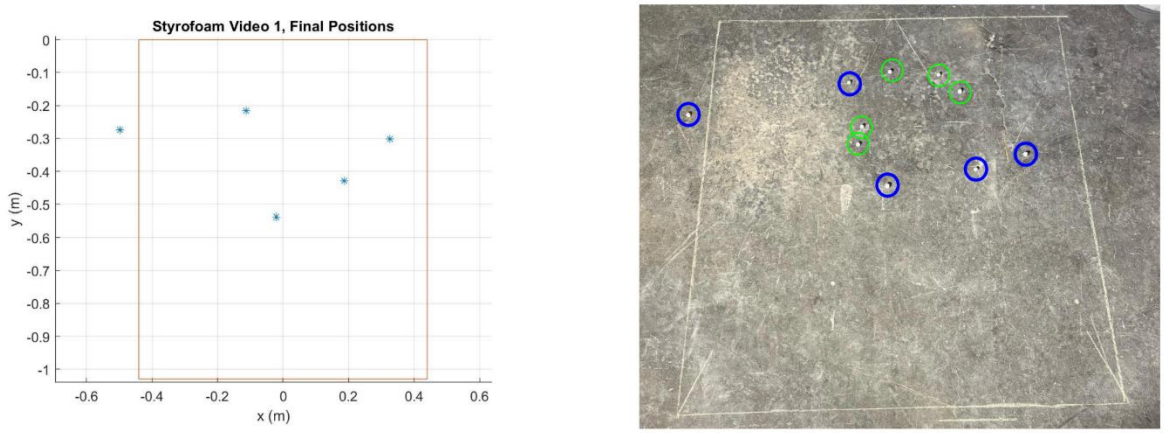


Figure 35. Ending positions of Styrofoam beads from first video. The prediction by software is shown on the left, while actual positions are shown on the right. The particles matched in both camera views in the last frame are highlighted in blue. A qualitatively good match is seen.

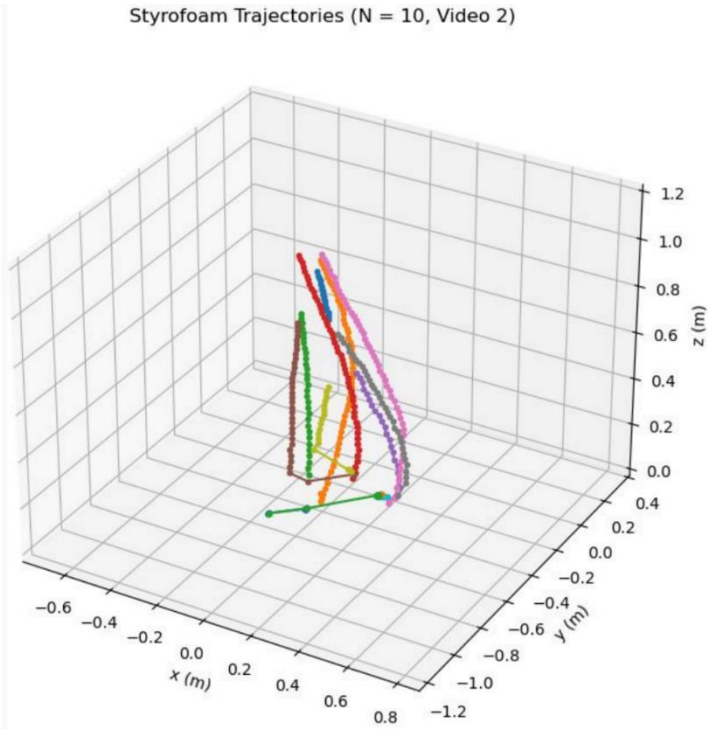


Figure 36. Second Styrofoam video's trajectories.

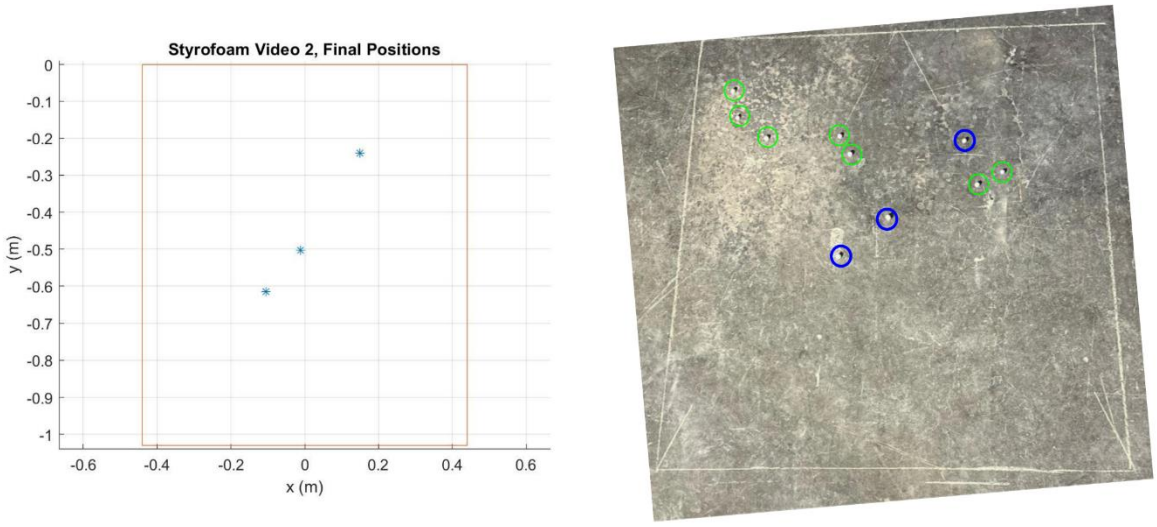


Figure 37. Ending positions for second Styrofoam test, n = 10 particles. The prediction by software is shown on the left, while actual positions are shown on the right. The particles matched in both camera views in the last frame are highlighted in blue. A qualitatively good match is seen.

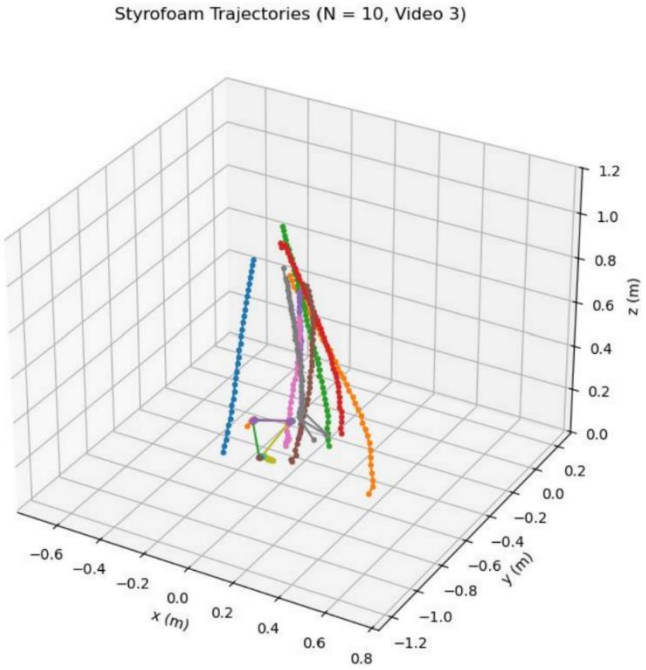


Figure 38. Third Styrofoam video's trajectories.

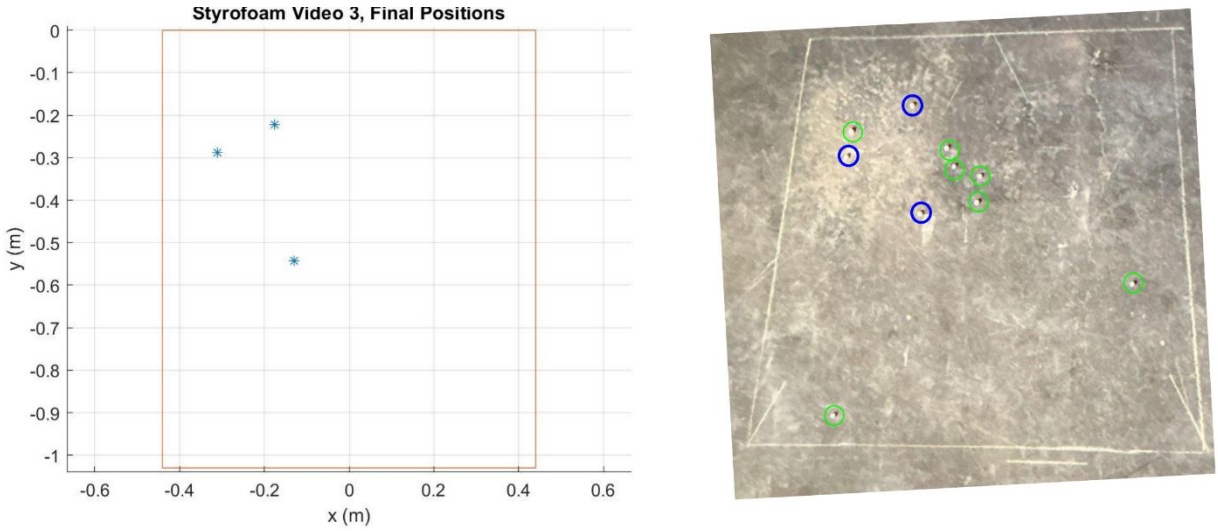


Figure 39. Ending bead positions of third Styrofoam particles. The prediction by software is shown on the left, while actual positions are shown on the right. The particles matched in both camera views in the last frame are highlighted in blue. A qualitatively good match is seen.

For tests with higher density, OpenPTV/PyPTV wasn't able to capture a significant number of trajectories. For 50 Styrofoam balls dropped, ~18 trajectories max had correspondences, meaning that many trajectories were missed. We present the end trajectories, but do not do any verification for the n=50 particle tests.

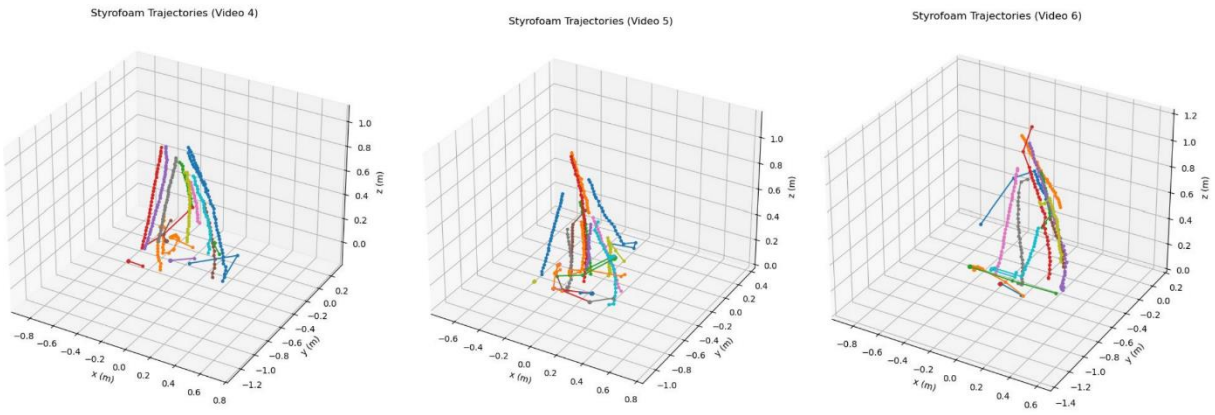


Figure 40. Styrofoam trajectories with n = 50 particles. It is evident that 50 trajectories are not seen, indicating that the sensor package and PTV are overwhelmed.

Firebrand trajectories for cumulative firebrand masses of 20 g and 60 g are presented. The trajectories for the 20 g cumulative mass firebrand tests are well connected. Some of the trajectories seem to end midair, which may be due to the firebrands physically burning out before reaching the ground (as seen in some video 1 frames).

We now present the firebrand results. We did 2 sets of tests, one set with 20 g of fuel (wooden dowels) and one set with 60 g of fuel.

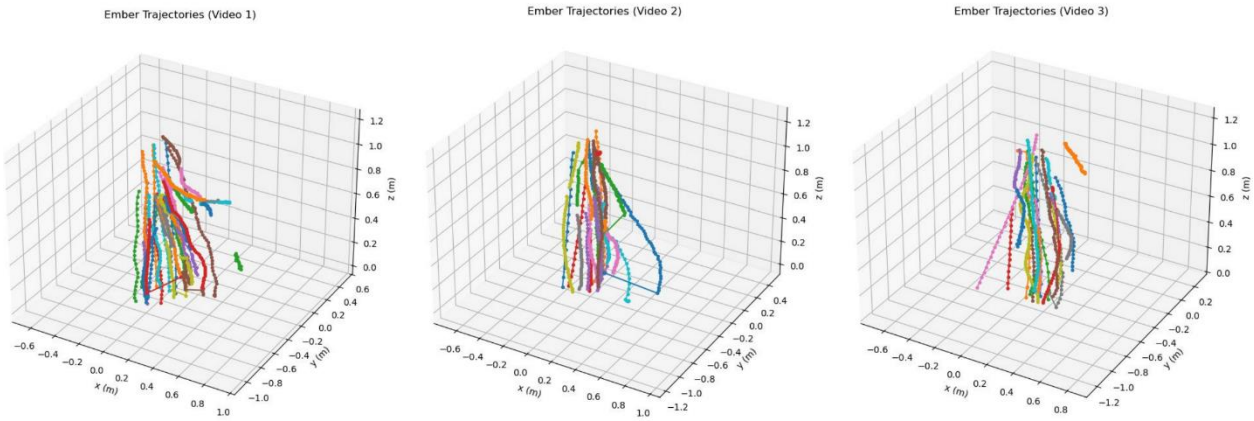


Figure 41. Trajectory plots of firebrands from a test with 20 grams of firebrands.

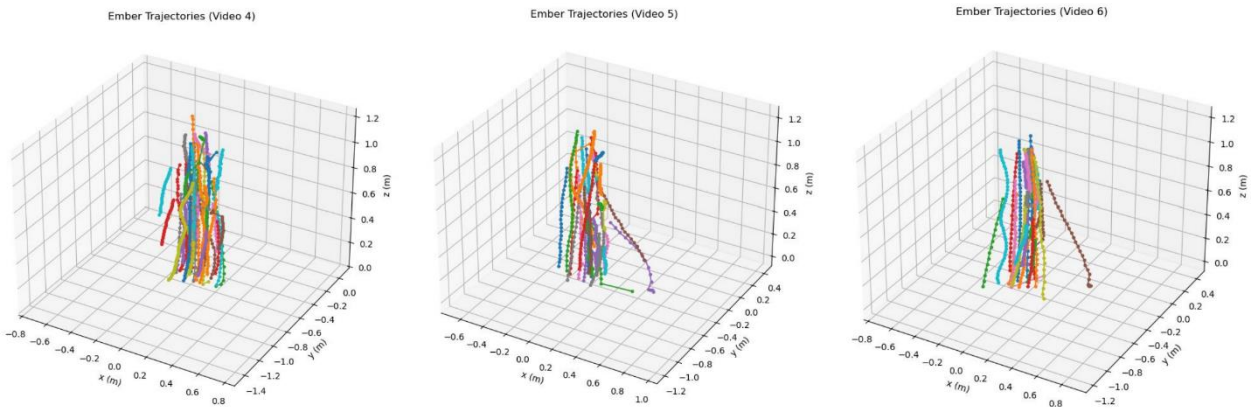


Figure 42. Trajectory plots of firebrands from a second set of tests with 60 grams of firebrands.

We see reasonably continuous and dense trajectories of the embers in all the plots shown in Figure 41 and 42.

3.3 Sensor Package Performance

During testing, the sensor package showed strong overall performance and proved to be highly versatile across different applications. The 3D-printed enclosure was lightweight, durable, and securely housed the components, making it straightforward to mount onto a drone. To avoid heat damage and maintain accurate calibration, we conducted all ember tests from approximately 1.5 meters away. While our current 3D-printed material held up sufficiently during these initial tests, future iterations should explore heat-resistant materials or additional protective measures, especially for use in wildfire environments where higher temperatures could pose a risk to the structure's integrity.

Although the stereo vision system successfully detected small particles and tracked their trajectories effectively, the current setup relies on an HDMI cable connecting the Jetson Nano to an external monitor. This configuration limits portability and transitioning to a wireless communication system would significantly enhance field usability and deployment flexibility.

Overall, this sensor package prototype provides an affordable, user-friendly option for rapid deployment in environmental monitoring scenarios. With improvements to thermal resilience and wireless connectivity, it can become an even more reliable and practical solution, particularly suitable for challenging conditions like wildfires.

3.4 Testing Anecdotes

Running open-source software can be a painful process. There have been times where the Jetson Nano behaves in strange ways or freezes. The easiest solution to solving this problem is usually cutting the power to the Nano, as one does not always have enough time to wait and see if the Nano unfreezes itself. If the user is particularly unlucky (which we were once), power cycling the Nano like this causes drive mounting issues. From the team's understanding, the microSD card is no longer mountable, meaning data must be recovered by professionals or by online software (which often require payment for useful features). It is a good idea to have several cards around with NVIDIA's Jetson Nano Ubuntu operating system ready to go, as one may run into these problems as well.

Chapter 4: Discussion

The team was ultimately able to produce a functional stereo vision device map trajectories of particles representative of embers. The scope of this project has been to design a package that could be equipped onto a UAV, and the team was successful in providing a suitable package to be flow with a UAV. Due to a lack of resources, this setup has not been tested in flight, which is why the final product of this thesis is a payload that collects ember data in a laboratory setting. We have considered the dimensional and weight requirements of this package and concluded that drone compatibility is possible if researchers with more funding wish to continue working on this concept.

4.1 System Performance Remarks

The calibration setup has several potential issues. An ideal setup follows NIST's setup, using rigid metal instead of bulk wood. Precise machining of the calibration target also makes coordinate measurements easy. The bulk wood has some variations and warping, which led to challenges with construction and system calibration. The wooden frame is prone to swaying under force, making it a relatively delicate calibration target.

4.2 Future In Situ and Ex Situ Testing of Firebrand Characteristics

It is evident that for PTV to be more effective, the particles need to be slow moving (with respect to frame rate) and well-spaced. The current system may be suitable for slow embers with low ember fluxes. Telescopic lenses can help with spacing problems by magnifying an image, and a higher frame rate camera can

address the speed problem. Since the telescopic lens also increases the mean distance moved by particles, a higher frame rate is more desirable. Using n = 50 particles overwhelms PTV, while n = 10 particles appears to work better.

Recall that the particle spacing displacement ratio is given as

$$p = \frac{\Delta_0}{u'\Delta t},$$

and is used to rate the difficulty for particle tracking with PTV [34]. If

$$p \ll 1,$$

particle tracking is extremely difficult [34]. The particle spacing is relatively small, given the lenses used (compared to telescopic lenses), and the imaging time step is relatively large, when compared to the framerates on the order of thousands of frames per second in fluid dynamics experiments. The mean distance moved by the particles is also relatively large, due to the low frame rate, and is made worse with faster particles (in our case, embers). Thus, the system performs as expected (that is, there are fewer than desirable correspondences in the Styrofoam tests with n = 50 particles), as high particle concentrations reduce the particle spacing displacement ratio.

Future work can include optimization of interrogation volume and telescopic lens magnification, as both limit one another. The “accessible” ember flux meter was

designed to have an interrogation volume like NIST's emberometer. We have experimentally seen that this design favors a large interrogation volume, at the expense of particle density and flux for tracking. The meter is designed for low flux measurements over a large volume.

4.3 Applications of Sensor Package

The sensor package should be drone mountable, allowing for potential applications in remote sensing and characterization of ember sources. For integration with a drone, software modifications need to be made, along with a logistics plan for data acquisition. For remote data acquisition with such a system, secure shell or VNC server can be used to get a remote connection to the airborne instrument. If the instrument parameters are already set, it is wiser to run secure shell, to avoid receiving unnecessary graphics data (that is user desktop GUI and other unnecessary graphics operations for remote operation). A mobile hotspot is needed to ensure a wireless connection. Below are schematics and logistics plans for field operation.

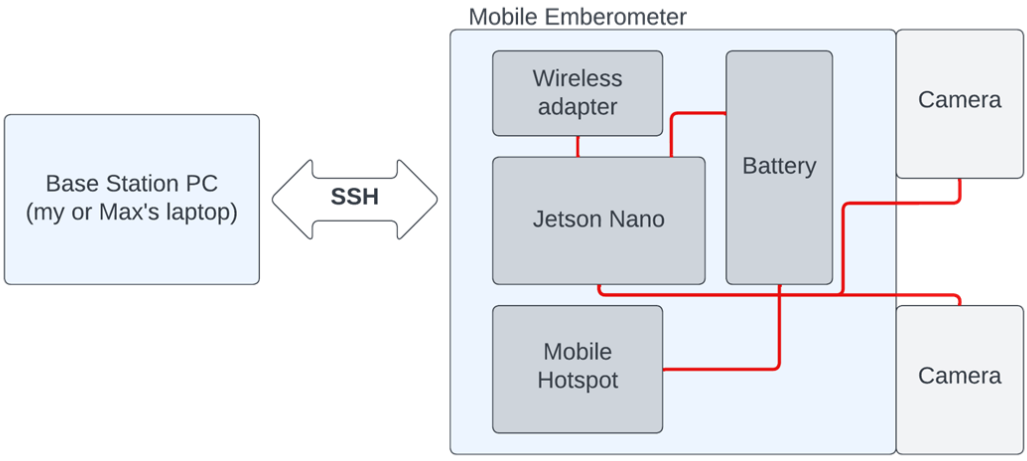


Figure 43. Block diagram of base station control of mobile ember flux meter.

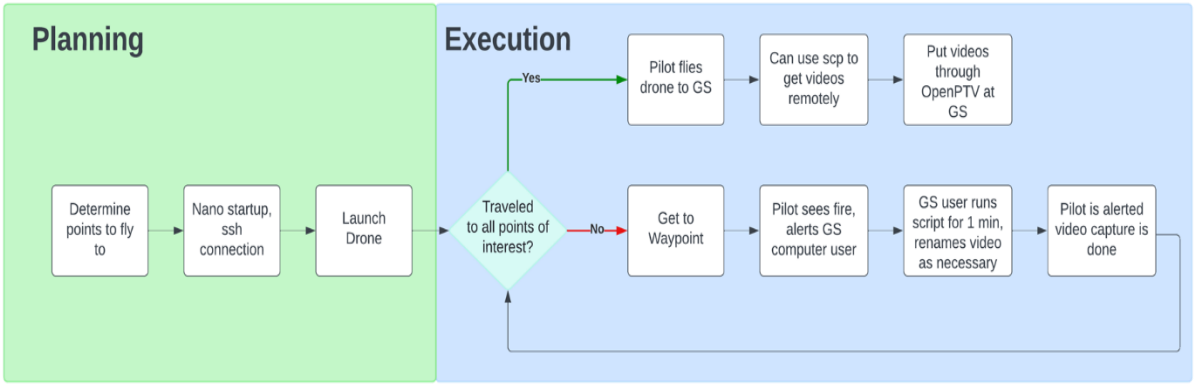


Figure 44. Logistics plan for operation in the field.

4.4 Exploration of Affordability Tradeoffs

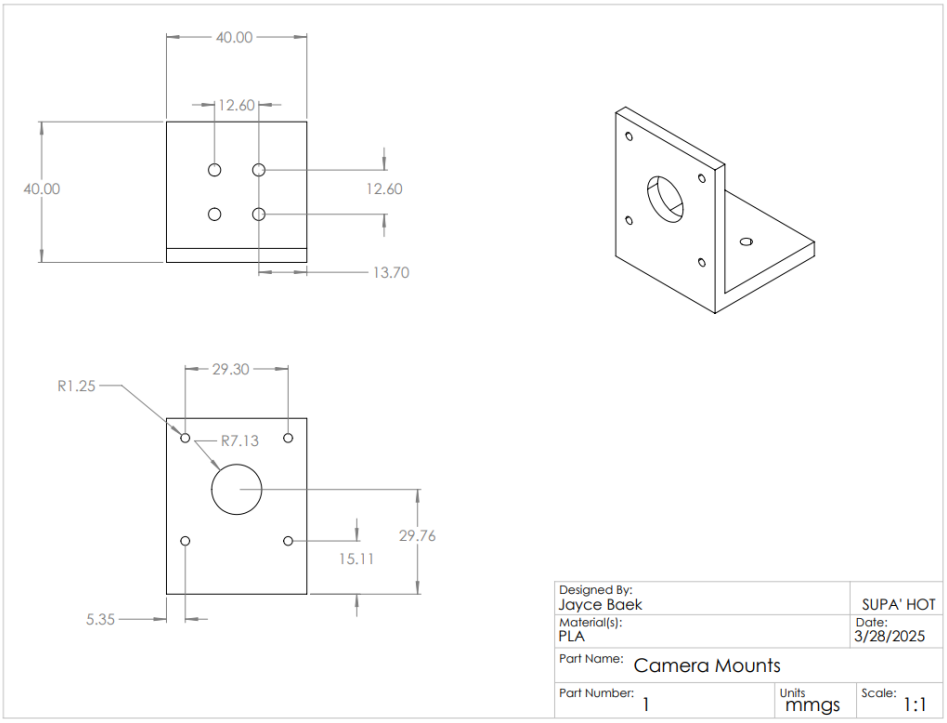
There are several tradeoffs made for the sake of affordability. Servicing of the accessible ember meter is less supported, as one must rely on NVIDIA forums and electronics/robotics/software enthusiasts, as opposed to dedicated support from a company. One must also become somewhat familiar with NVIDIA Ubuntu and the command line for painless operation of the ember meter. Affordability comes at the price of convenience.

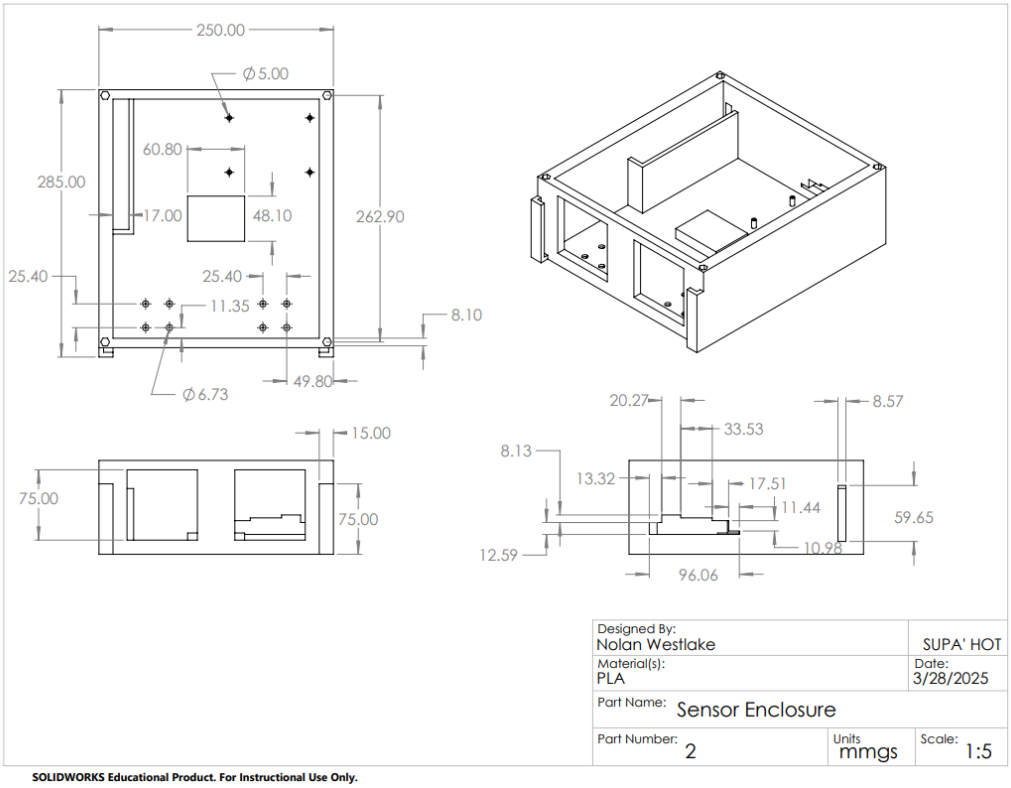
Appendices

1. Final Bill of Materials for Sensor Package

1	MECCANIXITY FFC FPC Flat Flexible Cable, 22 Pin 0.5mm Pitch 300mm Long (A Type) for LCD 3D Printer Camera DVD TV Laptop Audio Pack of 5	$6.29	
2	NVIDIA Jetson Nano Developer Kit	$149.00	
3	Thorlabs RP005 - Ø1" Manual Rotation Stage (Camera angle mounts) x2	$138.24	
4	Lithium Ion Battery Pack - 10Ah (3A/1A USB Ports)	21.50$	
5	Sensor box nuts and bolts -18-8 Stainless Steel Button Head Hex Drive Screw 2-56 Thread Size, 3/4" Long -Low-Strength Steel Hex Nut Zinc-Plated, 2-56 Thread Size -Zinc-Plated Steel Coupling Nut Low-Strength, 8-32 Thread Size -18-8 Stainless Steel Button Head Hex Drive Screw 8-32 Thread Size -Low-Strength Steel Hex Nut Zinc-Plated, 10-24 Thread Size -18-8 Stainless Steel Button Head Hex Drive Screw 10-24 Thread Size, 3/8" Long	$38.37	
6	Blackfly S (30mm)/Blackfly/Flea3/Chameleon3 ¼-20 Tripod Adapter	$21.79	
7	newlng M3 M4 M5 M6 Cross Countersunk 304 Stainless Steel Machine Screws Bolts and Nuts Fastener Repair Tools Assortment Kit	$18.58	
8	uxcell 0.5mm Pitch 22 Pin to 22 Pin Extension Connector Aapter for FFC FPC Cable Extend Zip HDD	$17.97	
9	Arducam 8MP IMX219 175 Degree Ultra Wide Angle for Raspberry Pi Camera Module	$59.98	
10	Arducam Multi Camera Adapter Module V2.2 for Raspberry Pi 4B	$49.99	
11	PLA 1kg spool	$14.98	
12		**$536.69**	Total

2. Sensor Package 3D Printed Components CAD and Engineering Drawings





Bibliography

[1] D. Viegas, “Overview of forest fire propagation research,” *Fire Safety Science*, vol. 10, pp. 95–108, 2011, doi: 10.3801/IAFSS.FSS.10-95.

[2] D. P. Kasymov *et al.*, “Effect of wood structure geometry during firebrand generation in laboratory scale and semi-field experiments,” *Journal of Physics: Conference Series*, vol. 1867, no. 1, p. 012020, Apr. 2021, doi: 10.1088/1742-6596/1867/1/012020.

[3] S.E. Caton, R.S.P. Hakes, M.J. Gollner, D.J. Gorham, A. Zhou “Review of pathways for building fire spread in the wildland urban interface. Part I: exposure conditions,” Fire Technol. (2017) 53:429–473.

[4] M. E. L. Houssami, E. Mueller, A. Filkov, J. C Thomas, N. S Skowronski, M. R Gallagher, K. Clark, R. Kremens, A. Simeoni, *Fire Technology* vol. 52, no. 3, p. 731-751, 2015, doi: http://dx.doi.org/10.1007/s10694-015-0492-z

[5] S. Solak and E. D. Bolat, "Distance estimation using stereo vision for indoor mobile robot applications," 2015 9th International Conference on Electrical and Electronics Engineering (ELECO), 2015, pp. 685-688, doi: 10.1109/ELECO.2015.7394442.

[6] B. I. Cook, J. S. Mankin, and K. J. Anchukaitis, “Climate Change and Drought: from past to Future,” *Current Climate Change Reports*, vol. 4, no. 2, pp. 164–179, May 2018, doi: https://doi.org/10.1007/s40641-018-0093-2.

[7] K. Gajendiran, S. Kandasamy, and M. Narayanan, “Influences of wildfire on the forest ecosystem and climate change: A comprehensive study,” *Environmental Research*, vol. 240, no. 2, p. 117537, Jan. 2024, doi: https://doi.org/10.1016/j.envres.2023.117537.

[8] R. Wadhwani, C. Sullivan, A. Wickramasinghe, M. Kyng, N. Khan, and K. Moinuddin, “A review of firebrand studies on generation and transport,” *Fire Safety Journal*, p. 103674, Sep. 2022, doi: https://doi.org/10.1016/j.firesaf.2022.103674.

[9] S. L. Manzello, S. Suzuki, M. J. Gollner, and A. C. Fernandez-Pello, “Role of firebrand combustion in large outdoor fire spread,” *Progress in Energy and Combustion Science*, vol. 76, p. 100801, Jan. 2020, doi: https://doi.org/10.1016/j.pecs.2019.100801.

[10] “Hazard - fire behavior: Wildfire,” *Fire behavior: Wildfire - WDF | NFCC CPO*. [Online Image]. Available: https://www.ukfrs.com/guidance/search/fire-behaviour-wildfire. (accessed Dec. 01, 2022)

[11] D. P. Roy, H. De Lemos, H. Huang, L. Giglio, R. Houborg, and T. Miura, “Multi-resolution monitoring of the 2023 maui wildfires, implications and needs for satellite-based wildfire disaster monitoring,” *Science of Remote Sensing*, vol. 10, p. 100142, Dec. 2024, doi: https://doi.org/10.1016/j.srs.2024.100142.

[12] A. Filkov *et al.*, “Investigation of firebrand production during prescribed fires conducted in a pine forest,” *Proceedings of the Combustion Institute*, vol. 36, no. 2, pp. 3263–3270, Jan. 2017, doi: https://doi.org/10.1016/j.proci.2016.06.125.

[13] D. Nguyen and N. B. Kaye, “Experimental investigation of rooftop hotspots during wildfire ember storms,” *Fire Safety Journal*, vol. 125, no. 103445, pp. 1–16, Oct. 2021, doi: 10.1016/j.firesaf.2021.103445.

[14] A. C. Fernandez-Pello *et al.*, "Spot Fire Ignition of Natural Fuel Beds by Hot Metal Particles, Embers, and Sparks," *Combustion Science and Technology*, vol. 187, no. 1–2, pp. 269–295, Dec. 2014, doi: 10.1080/00102202.2014.973953.

[15] S. L. Manzello, A. Maranghides, and W. E. Mell, "Firebrand generation from burning vegetation," *International Journal of Wildland Fire*, vol. 16, no. 4, pp. 458–462, 2007, doi: 10.1071/wf06079.

[16] S. Rissel and K. Ridenour, "Ember Production During the Bastrop Complex Fire," *Fire management Today*, vol. 72, no. 4, pp. 7–13, 2013, Accessed: Sep. 27, 2022. [Online]. Available: www.frames.gov/documents/usfs/fmt/fmt_72-4.pdf#page=7.

[17] S. Suzuki and S. L. Manzello, "Characteristics of Firebrands Collected from Actual Urban Fires," *Fire Technology*, vol. 54, no. 6, pp. 1533–1546, Jul. 2018, doi: https://doi.org/10.1007/s10694-018-0751-x.

[18] S. Zen, J.C. Thomas, E.V. Mueller, et al. Development of a Field Deployable Firebrand Flux and Condition Measurement System. Fire Technol 57, 1401–1424 (2021). https://doi.org/10.1007/s10694-020-01074-x.

[19] N. Bouvet, E. D. Link, and S. A. Fink, "A new approach to characterize firebrand showers using advanced 3D imaging techniques," *Experiments in Fluids*, vol. 62, no. 9, pp. 1–14, Aug. 2021, doi: 10.1007/s00348-021-03277-6.

[20] A. Filkov and Sergey Prohanov, "Particle Tracking and Detection Software for Firebrands Characterization in Wildland Fires," *Fire technology*, vol. 55,

no. 3, pp. 817–836, Dec. 2018, doi: https://doi.org/10.1007/s10694-018-0805-0.

[21] F. Hedayati, B. Bahrani, A. Zhou, S. L. Quarles, and D. J. Gorham, “A Framework to Facilitate Firebrand Characterization,” *Frontiers in Mechanical Engineering*, vol. 5, Jul. 2019, doi: https://doi.org/10.3389/fmech.2019.00043.

[22] N. Bouvet, E. Link, S. Fink, (2020), Development of a New Approach to Characterize Firebrand Showers During Wildland-Urban Interface (WUI) Fires: a Step Towards High-Fidelity Measurements in Three Dimensions, Technical Note (NIST TN), National Institute of Standards and Technology, Gaithersburg, MD, [online], https://doi.org/10.6028/NIST.TN.2093.

[23] OpenPTV consortium, “OpenPTV 0.0.07 documentation,” *Readthedocs.io*, 2019. https://OpenPTV-python.readthedocs.io/en/latest/index.html (accessed Nov. 29, 2024).

[24] R. Ghali, M. A. Akhloufi, and W. S. Mseddi, “Deep Learning and Transformer Approaches for UAV-Based Wildfire Detection and Segmentation,” *Sensors*, vol. 22, no. 5, p. 1977, Mar. 2022, doi: 10.3390/s22051977.

[25] V. G. Ambrosia and E. Hinkley, "NASA Science Serving Society: Improving Capabilities for Fire Characterization to Effect Reduction in Disaster Losses," *IGARSS 2008 - 2008 IEEE International Geoscience and Remote Sensing Symposium*, 2008, pp. IV - 628-IV - 631, doi: 10.1109/IGARSS.2008.4779800.

[26] F. Afghah, A. Razi, J. Chakareski and J. Ashdown, "Wildfire Monitoring in Remote Areas using Autonomous Unmanned Aerial Vehicles," *IEEE*

INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2019, pp. 835-840, doi: 10.1109/INFCOMW.2019.8845309.

[27] K. V. Stefanik, J. C. Gassaway, K. Kochersberger, and A. L. Abbott, "UAV-based stereo vision for rapid aerial terrain mapping," *GIScience & Remote Sensing*, vol. 48, no. 1, pp. 24–49, 2011. doi:10.2747/1548-1603.48.1.24

[28] M. S. Innocente and P. Grasso, "Self-organizing swarms of firefighting drones: Harnessing the power of collective intelligence in decentralized multi-robot systems," *Journal of Computational Science*, vol. 34, pp. 80–101, May 2019, doi: 10.1016/j.jocs.2019.04.009.

[29] T. O. Brien, H. B. Abrahamsen, A. Zamarro, M. Valls, O. Badia, J. Guasch, D. Ioannidis, K. Votis, C. Palaskas, S. Rogotis, and E. Mangina, "Remote Piloted Airborne Systems (RPAS) and the Emergency Services," 2015.

[30] R. Collins *et al.*, "Stereoscopic vision in unmanned aerial vehicle search and rescue," 2018.

[31] A. Schmidt. "How Destructive Wildfires Create Their Own Weather." *AccuWeather*, AccuWeather, 1 July 2019, https://www.accuweather.com/en/weather-news/how-destructive-wildfires-create-their-own-weather/346337.

[32] W. C. Myeong, K. Y. Jung and H. Myung, "Development of FAROS (fire-proof drone) using an aramid fiber armor and air buffer layer," *2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, 2017, pp. 204-207, doi: 10.1109/URAI.2017.7992713.

[33] D. Bauer, “Particle Tracking Velocimetry with OpenPTV,” *University of Kentucky College of Engineering*, 2016, https://openptv-python.readthedocs.io/en/latest/_downloads/b95ac75da8055488e8a60e0faacbe752/PTV_report-Dominik%20Bauer.pdf.

[34] “1MP*2 OV9281 Global Shutter Monochrome Stereoscopic Camera Bundle Kit for Raspberry Pi, NVIDIA® Jetson AGX Orin/Orin Nano/Orin NX,” *Arducam.com*, 2025. https://www.arducam.com/product/arducam-1mp2-stereoscopic-camera-bundle-kit-for-raspberry-pi-nvidia-jetson-nano-xavier-nx-two-ov9281-global-shutter-monochrome-camera-modules-and-camarray-stereo-camera-hat (accessed Apr. 30, 2025).

[35] “OpenCV: Epipolar Geometry,” *docs.opencv.org*. https://docs.opencv.org/4.x/da/de9/tutorial_py_epipolar_geometry.html (accessed Apr. 25, 2024).