A Multi-View Multi-LiDAR Synchronised Setup for Smoke Plume Data Acquisition: A Simulated Drone Swarm Observation Case Study

G. Pérez*[1], L. Denis[1], H. Chakraa[2], M. Bronz[2]

[1]LAAS-CNRS, Université de Toulouse, Toulouse, France.
[2]École Nationale de l'Aviation Civile (ENAC), Université de Toulouse, 31400 Toulouse, France.

ABSTRACT

Each year, hundreds of thousands of hectares are devastated by wildfires, causing severe impacts on biodiversity and major economic losses. In this context, better understanding of wildfires for rapid and effective response is essential. We aim to develop methods of wildfire analysis using a drone fleet to provide accurate information for optimal intervention. This work forms part of this project with a focus on smoke analysis. The chaotic behaviour of fires, and the high temperatures makes the safe deployment of drones in a wildfire context a challenging task, leading to a lack of data for smoke and fire analysis. To address this gap, we present an experimental setup leading a public dataset of temporally synchronised and spatially calibrated images and point clouds of smoke plumes in a controlled environment. Using three 3D LiDARs Livox-MID360 and five heterogeneous HD cameras, this setup is the first to make multi-view and multi-LiDAR data of smoke scenes available[1]. A proof-of-concept demonstration is presented, featuring a simulation of a swarm of drones interacting with a dynamic smoke plume represented by a 3D point cloud obtained with the described setup.

1 INTRODUCTION

1.1 Context

Due to climate change and rising summer temperatures, which are breaking new records every year, wildfires are becoming increasingly frequent. Whether caused by arson, accident or natural causes, they are extremely devastating and seem difficult to predict. These extreme fires have a direct impact on the natural environment and the health of living beings (including humans), as they are very difficult to control and release toxic fumes that severely affect air quality [1]. Given the increasing frequency of wildfires, it is essential to be prepared and to manage these fires as effectively as possible in order to minimise their socio-ecological impact [2]. In this context, a drone fleet equipped with LiDARs [2] and cameras could gather data of the smoke to analyse it and try to model in order to predict the short-term evolution of the smoke plume and the fire. This information could prove useful to firefighters on the ground in helping them devise a strategy to contain the fire. To our knowledge, we are the first to build a dataset combining LiDAR and images to capture smoke rising, allowing to explore diverse methods for smoke understanding. This opens the door to exploit new kind of data for drone control or machine learning tasks.

1.2 Related work

The presented setup tries to link computer graphics/computer vision works on smoke reconstruction, to potential robotics applications that can be done with drones in the context of wildfires.

LiDAR for environment observation. Since several years now, LiDARs have become a central component in several robotics fields and for different purposes. In autonomous vehicles fields, combining several LiDARs enables to build a precise map of the surrounding environment with minimised blind spots, making localisation, object tracking and decision-making more robust [3] [4] [5]. LiDARs were also impressively impactful in agriculture fields since it can help to monitor and characterise the state of crops and trees [6], using different types of vehicles [7][8][9][10].
As LiDARs are mainly used in robotics for object detection and navigation, data accuracy is critical, in this context some work show interest to the interaction with aerosols or in adverse weather, but usually considering it as noise that reduces performances [11] [12] and not to observe their properties.

Camera-LiDAR sensor fusion. Even if 3D LiDAR is now a widely used component in robotics they have, like all sensors, limitations in some cases. The discrete representation of the environment and the absence of colour information makes it hard to characterise the targets. It is then widely recognised that sensor fusion gives more promising information. Specifically in the case of LiDARs, Camera-LiDAR sensor fusion was the first association to rise in literature starting from the

*Email address: gaetan.perez@laas.fr
[1]Git repo available here : https://github.com/Typos91/LiCamFlow

[2]Please note that when speaking about LiDARs or 3D LiDARs, we refer to mid-range sensors (around 100m), unless otherwise stated.

pioneer KITTI dataset [13].
Even though this sensor combination is majorly used in the case of autonomous vehicles, and object detection, we found very few works referring to Camera-LiDAR sensor fusion in case of dynamic phenomenon observation, and more specifically fluid flows like in this study.

Fluid flow Observation. Fluid flow observation is an important concept in various fields like fluid mechanicals but also computer graphics or machine-vision. Several techniques to extract the volumetric fluid flow information were designed in the past. One of the most famous techniques was the Particle Imaging Velocimetry (PIV), it regroups a set of techniques reviewed first by [14] which uses lasers illuminating a fluid flow with tracers particles, and capturing the scattering effect of the laser on these tracers across time, to get an information of the local dynamics of the fluid and following it in time. A similar approach was to capture the scattering effect of a flow particles (without tracers) in participating media like smoke [15] [16]. However, the scattering effect depends on the relative size of the particles in relation with the laser wavelength making this technique dependent of the composition of the media studied. PIV technique is a pioneer in fluid flow volumetric observations as it enables to perform 3D reconstruction of the flow using multiple cameras [17]. However, the observation is usually made on small volumes and needs an important setup that must be installed in a laboratory which is therefore not adapted in our case.
A widely used alternative to these setups involving lasers, and usually made for small-scale studies, is the tomography. It consists of sets of 2D views of the subject, which are used to build a 3D volumetric reconstruction. Tomography proved to be very useful especially for gas and smoke observation, which is more complex to observe with the PIV method. [18] uses optical flow algorithm and a specific noisy background to record transparent non-stationary gas flow with 16 cameras, and builds a 3D reconstruction of refractive index of the gas enabling to determine physical properties of the gas. [19] takes low-resolution 2D views as an input to flow tracking by adding physical constraints through the Navier-Stokes equation and a pressure solver usually used in simulations. Recently, Saclarflow [20] became a huge milestone specifically in 3D smoke reconstruction, presenting physically-based simulated smoke voxel grids trained with real data. By releasing the dataset they used to train their model, it opened the door to new machine learning-based tomography techniques [21] for smoke flow observation. This work aims to follow on from fluid flow tomography works, and add LiDAR data, giving geometrical precision to the observations. We think that these data may be of interest outside the computer graphics community with the addition of LiDAR data. At larger scales, other types of sensors can be used for flow observation. More specifically with smoke, RADARs and Doppler LiDARs are used to observe smoke density and speed information at long range [22], enabling to study precisely the speed fields inside a wildfire smoke plume. However, these sensors are not made for mobile robotics and need to be static to be used.
Other methods involving images for smoke plume observation can be used, such as satellite imagery or using a fleet of drones. [23] deploys a fleet of drones, in which a leader drone observes a plume using a camera and controls four other drones to optimise observation. These drones use only cameras to observe the plume and then generate a 3D coloured point cloud from the images recorded using the COLMAP algorithm. However, our work and objectives focus on the use of a camera-LiDAR combination to directly get a 3D point cloud from the sensors and improve the quality of the generated point cloud obtained when using solely COLMAP.

Datasets. In both robotics and computer graphics, datasets are of great importance since they enable testing algorithms developed with real-world conditions, showing the limits but also opening new subjects. However in robotics, those algorithms are generally not used directly for control but for closely related topics like computer vision tasks or localisation.
As mentioned previously, the Scalarflow [20] dataset had major a impact on the computer graphics community in smoke observation; however, the lack of situation diversity in the sequences prevents proper generalisation.

2 MULTI-VIEW AND MULTI-LIDAR ACQUISITION

In order to build a coherent dataset, we want to be able to merge image and point cloud data. These can be used if sensors are spatially calibrated (meaning that the frame transformation between each sensor is known) and temporally synchronised.

2.1 Setup

The complete setup consists of 3 ”acquisition modules” plus 2 cameras placed all around a fog machine to minimise blind spots. An acquisition module corresponds to a fixed piece with 1 LiDAR Livox MID-360 and 1 camera attached. The two remaining cameras are not attached to any LiDAR. For the LiDARs to be able to capture the whole plume, they are rotated 90° so that the horizontal view becomes the vertical one, and therefore the vertical FOV is no longer a limitation.

Choice of LiDAR. The choice of the LiDAR is crucial, as it is a balance between what can be assembled on a drone and the quality of the output data. For this choice, 2 LiDARs were compared with respect to their mechanical properties (size of sensor and weight) and the data quality: the Livox MID-360 and the Ouster OS1. Figure 2 gives a visual of the comparison between the two sensors. Two launches of smoke were performed using the fog machine, one with a Livox MID-360 LiDAR’s recording (2a 2b 2c), and the other using an Ouster OS1 LiDAR (2d 2e 2f). In figures 2e 2f and 2b 2c, the plume was extracted from the rest of the point cloud to focus the comparison on it. The smoke extraction is explained in section 3.2.

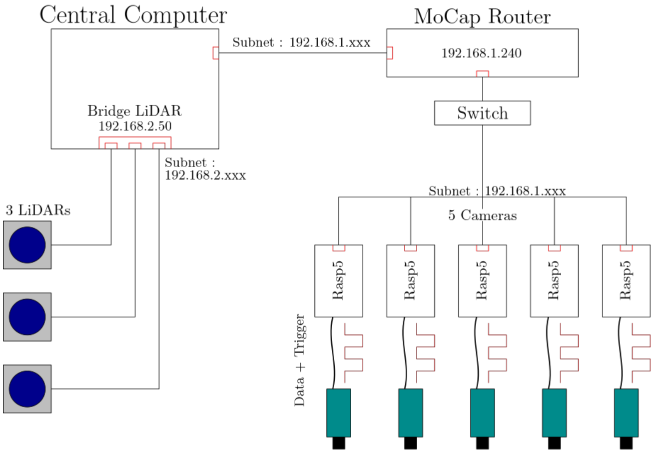
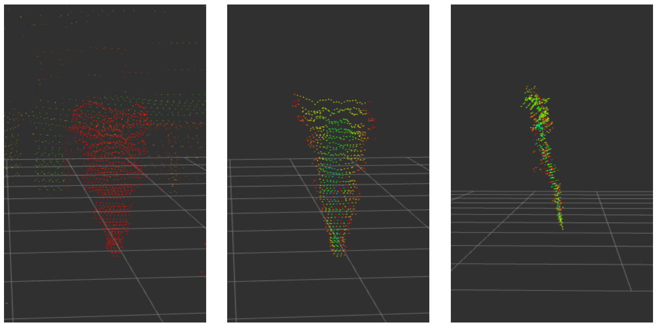


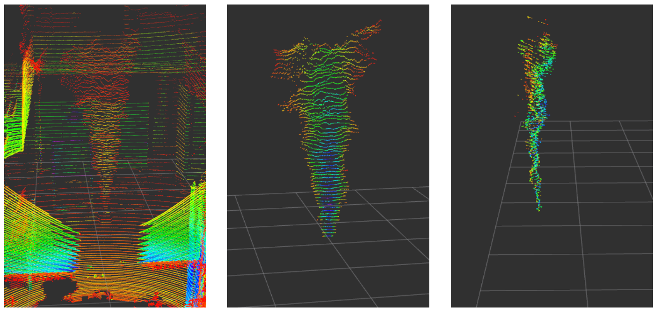
Figure 1: Schematic diagram of the final setup for multi-view/multi-LiDAR acquisition

Unlike the OS1 sensors, the MID-360 LiDAR employs a different, non-repeatable scanning method, leading to some holes in the 3D point cloud of the smoke plume, as it can be seen in figure 2a. However this pattern allows for more random distribution of points within the environment, providing an accurate representation with fewer scans (a maximum of 200,000 points per second for the Livox MID-360 compared to a maximum of 5.8 million points per second for the OS1). This difference in point number is clear when comparing figure 2a with 2d. This is also reflected in the number of points representing the smoke plume, which is around 2000 points for the Ouster LiDAR, whereas the Livox returns around 1000 points for the same smoke plume. It is also worth noting that the reflectivity given by the Livox LiDAR is harder to use because all points have very low value. On a scale from 0 to 255, with 0-150 values corresponding to diffuse scattering objects and 151-255 corresponding to total reflection objects, the returns of the smoke have their intensity value usually between only 0 and 10 (please note that the computation of the reflectivity varies depending on the LiDAR manufacturer, but it is always computed using the ratio of the amount of energy received relative to that emitted by the sensor).
However, the strength of the Livox LiDAR resides in its size ($65 \times 65 \times 60mm$) and its small weight (285 g with integrated mounting system versus 925 g with mounting system for the Ouster OS1), making it very appropriate for small drones easily deployable, which is an important characteristic in our context. Therefore, even if the Livox MID-360 shows some limitations in comparison with the OS1, its small size and adaptability to drone applications make it a better choice for our setup.

Network. This setup aims to be adaptable to different conditions, in particular, indoor and outdoor conditions in order to compare data in controlled and uncontrolled environments. Outdoor conditions also imply longer distances, which means



(a) Livox plume, full point cloud (b) Livox plume, front view (c) Livox plume, side view



(d) Ouster plume, full point cloud (e) Ouster plume, front view (f) Ouster plume, side view

(g) Rainbow colorbar from low to high intensity/reflectivity

Figure 2: Comparison of a smoke plume detected with 3 Livox MID-360 LiDARs (up) and a smoke plume detected with one Ouster OS1 LiDAR (bottom). The colour bar represents the evolution of the point colours with respect to their intensity (or reflectivity). This information is provided directly by the sensor, however, each sensor can have a different way to compute this value, so they cannot be compared directly.

longer cables for data transmission. Therefore, an Ethernet connection must be used. Since USB3.0 cameras are used, a Raspberry Pi 5 is added to each camera. The cameras are directly connected to their Raspberry Pi, and the latter are connected via Ethernet to a switch connected to a router. Furthermore, the Raspberry Pi will be used for temporal synchronisation using the GPIOs to trigger the camera. Figure 1 presents a schematic diagram of the components and the networking done for this setup. As shown in the figure, all elements (except for the LiDARs) are connected to a router, which assigns to them an IP address. This subnet is then used to access the Raspberry Pis using an SSH connection from the main computer.

Lighting. The dataset targets image-LiDAR connection, and handling semi-transparent interactions with the back-

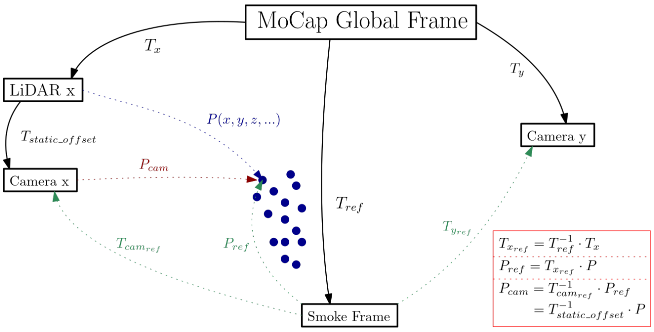


Figure 3: Scheme of the transformations and known positions given by MoCap. The black lines refer to the position given by the MoCap. The blue lines refer to positions in the LiDAR frame. The green lines are the position given in the reference frame, here the fog machine. The red lines refer to the position given in the camera frame from where the line starts. Transformations T are 4×4 homogeneous matrices containing translation and rotation between two frames. And Ps are the coordinates of a point given by the LiDAR, in a given frame.

ground is out of scope. For this reason, we focus on images where the light makes our understanding easier, and the background must have as little interaction as possible. This approach is inspired by Scalarflow [20] but differentiates by the size of the plume (ours elevates at five metres high, compared to approximately one metre in the Scalarflow dataset). As the sensors are disposed in a circle all around the smoke in order to capture not only one face of the smoke but also the whole plume, a black cloth cannot be set easily, and therefore the background is visible. However, to tackle this problem, the lights of the room are turned off, and 4 LED spots are posted directly towards the smoke plume in such a way that they do not appear behind the smoke for any camera. With this configuration, we obtain a nice contrast between the useless background and the subject of the images, which will be useful for the post-processing part.

2.2 *Sensor calibration*

Many pipelines in computer vision require precise camera calibration and knowledge of extrinsic and intrinsic parameters. Intrinsic parameters refers to internal characteristics of the sensor. In the case of a camera the parameters are the focal lengths, the central point of the sensor and the distortion effect of the lens. As for the extrinsic parameters, they refer to the position of the sensor in a given frame (translation and orientation). Therefore, the sensors were carefully calibrated using classical chessboard calibration for the intrinsic parameters of the cameras. As for the extrinsic parameters (cameras and LiDARs), motion capture was used.

Motion Capture. When working with cameras and LiDARs, especially when the objective is to use them for data fusion, it is important to have a precise estimation of the sensors positions, within the same frame. Using a chessboard detectable by LiDARs and the cameras, it is possible to find every position relative to one arbitrarily chosen sensor. However, this technique is heavy to set up and must be redone each time a sensor is moved intentionally or accidentally. When doing a setup with dozens of sequences, a dynamic way to have the sensors positions is needed. The motion capture is perfectly suited for this task, as it is able to give real-time positions of the sensors using markers like those presented in figure 4. The alignment between the motion capture rigid body origin and the sensors internal frame was done using the dimensions of the sensors given in their data-sheet meaning that a static offset between the motion capture rigid body and the actual sensor frame exists. Data is provided to do a more precise calibration using a reflective chessboard with motion capture markers.

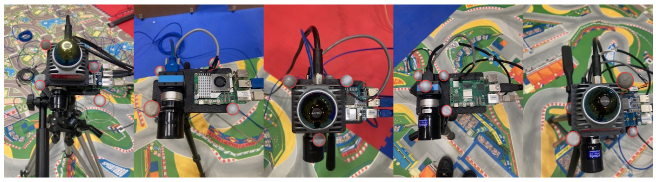


Figure 4: Motion Capture markers (circled in red) on each module

2.3 *Temporal synchronisation*

To observe a physical phenomenon, it is necessary to observe the evolution of the system over time. If several sensors are used, they must be synchronised so that the data can be merged with certainty that the same phenomenon is being detected by each sensor. This is particularly important for a plume of smoke, which is highly sensitive to the slightest disturbance.

Since perfect synchronisation is impossible, we estimate an offset acceptance. Using the first LiDAR frames of a sequence, when the smoke is expelled from the fog machine and rises, we measure the approximate speed of the rising smoke thanks to the known timestamps of the point clouds, which will give an approximation of the smoke speed at the source. With this technique, the estimated smoke speed is approximately $v_{smoke} = 3\text{m/s}$. According to [24], which carried out a comprehensive study that verified the sensor specification, the Livox MID-360 has centimetre accuracy. Therefore, with this precision limitation we can compute a temporal offset acceptance: $\delta t = \delta x / v_{smoke}$. With the LiDAR precision $\delta x \approx 0.01m$.

Therefore, we obtain a maximum wanted time offset of $\delta t \approx 3.3ms$ between sensors.

Camera triggering. In order to synchronise the shots, [20] uses the same hardware trigger for all cameras. In our case, as we want a setup that can be adapted to long distances, a single cable for all the cameras seems more complicated to manage.

We choose to parallelise the trigger signals, meaning that each Raspberry Pi associated with a camera will be responsible for sending the signal. In order for all signals to be sent at the same time, all Raspberry Pi units must be synchronised (i.e., hey must have the same time reference) and receive an instruction indicating when the signal should be sent. All this must be done while keeping in mind that the shots must also be synchronised with the LiDAR scans. The latter operate at 10Hz and have a scan time of just under 0.1s. We want to be able to take 2 photos for each scan, which means a frequency of 20Hz for the cameras. Based on these constraints, we construct the algorithm 1. We use the actual difference between two LiDAR scans to estimate when to send a signal to trigger the cameras. Algorithm 2 describes how to obtain the average value of the timestamps of the three LiDARs and compute the mean difference between two scans of a LiDAR.
For this algorithm to work properly, all devices must be synchronised so that they all have the same time reference. A central clock is chosen, to which all other devices (Raspberry Pi and LiDAR) can synchronise.

Synchronisation protocol. A synchronisation protocol is a protocol used to synchronise a local clock with that of a server connected to a physical clock (usually an atomic clock). For example, the NTP (Network Time Protocol) protocol allows synchronisation with a server that is accessible worldwide. These synchronisation protocols generally work with UDP communication protocols, hence the need for an Ethernet connection. For synchronisation with reference to a local physical clock, the PTP (Precision Time Protocol) synchronisation protocol is used [25].

2.4 Acquisition

Image Writing. One of the challenges when storing data received by a Raspberry Pi is its write speed. Indeed, the data must be stored faster than the rate at which it is received. This depends on several factors, such as the image size, its encoding, and the frame rate. In our case we work with images measuring a maximum of 1920 × 1200 px, with GRAY8 encoding (greyscale, 8-bit) and at a frame rate of 20 frames per second, which corresponds to a writing speed of 46 MB/s. This speed is below most of the SD card's maximum capacity, but if the frame rate were to be increased, a more specific storage system should be used.

ROS2 integration. ROS2 is a middleware platform used for the development of robotic systems. It enables the exchange of information between multiple programmes, known as nodes, via various standardised methods (topics, services and actions). Thus, for multi-sensory experimentation, it is a very adapted tool, especially when the various sensors are connected in some way (for example, with synchronisation or spatial calibration). Furthermore, ROS2 offers visualisation tools such as Rviz2 or RQT, which facilitate a comprehensive understanding of the system.
Therefore, ROS2 was used to record the LiDARs as well as

Algorithm 1: LiDARs - Cameras synchronisation using PTP and Raspberry Pi

Inputs :
$\Delta T_{\text{camera}_j}$: Time Exposure of the camera i
N : number of LiDAR
K : number of cameras
Temporal Variables:
$T_{\text{ts\_LiDAR}_i}$: LiDAR timestamp i
T_{capt} : Mean Timestamp between each LiDAR's timestamp
ΔT_{LiDAR} : Mean duration between 2 scans
t : Current Time of the computer
Outputs : Trigger signal sent from each raspberry Pi to their corresponding camera

Initialisation :
Start PTP protocol from the computer (master) to synchronise the devices clock (LiDARs and Raspberry Pi)
Wait until time is synchronised (indicated in LiDAR message)
For each LiDAR read $T_{\text{ts\_LiDAR}_i}$
for $t \leftarrow t_0$ **to** $t_0 + 1s$ **do**
  $T_{\text{capt}}, \Delta T_{\text{LiDAR}} \leftarrow$ `get_mean_timestamp`(t, N)
end
Starting Acquisition :
while *Aquisition active* **do**
  for $j \leftarrow 1$ **to** K **do**
    Compute $T_{\text{trigger}_j} \leftarrow T_{\text{capt}} - \frac{\Delta T_{\text{camera}_j}}{2} + \Delta T_{\text{LiDAR}}$ `// Estimate next time to Trigger each camera corresponding to its time of exposure`
  end
  $T_{\text{capt}}, \Delta T_{\text{LiDAR}} \leftarrow$ `get_mean_timestamp`(t, N)
  for $j \leftarrow 1$ **to** K **do**
    while `PTP_synced_clock()` $< T_{\mathit{trigger}_j}$ **do**
      Wait
    end
    Send Trigger `// This must be done in parallel thanks to the PTP calibration between the Raspberries`
  end
end

Algorithm 2: get_mean_timestamp

Inputs : t : Current Time of the computer
N : number of LiDAR

Outputs: T_{capt} : Mean Timestamps between all LiDARs
ΔT_{LiDAR} : Mean duration between 2 scans

for $i \leftarrow 1$ **to** N **do**
  Read $T_{\text{ts\_LiDAR}_i}$ from ROS2 message of LiDAR i
  `// recall that LiDARs and computer's clocks are synchronised`
end
Compute $T_{\text{capt}} \leftarrow mean(T_{\text{ts\_LiDAR}_i})$ `// Compute mean of timestamps`
Compute $\Delta T_{\text{LiDAR}} = T_{\text{capt}}(t) - T_{\text{capt}}(t - \Delta T_{\text{LiDAR}})$
Return :
$T_{\text{capt}}, \Delta T_{\text{LiDAR}}$

the motion capture positions, enabling the computation of the position of each sensor in the same frame with the smoke centred in (0,0). It was also used to send the information of when to trigger the cameras to each Raspberry Pi.

3 DATA POST PROCESSING

Different types of setups were tested, all respecting the imposed constraints (sensors all around the smoke, temporal synchronisation, spatial calibration, ...). So, the first step of the post-processing step was to qualitatively evaluate the data in order to strike a balance between data quality and ease with which the experimental setup can be reassembled in the same or different conditions.

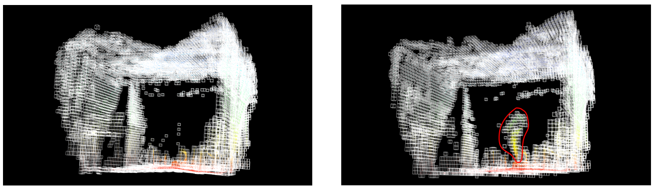
3.1 Images

As smoke is a semi-transparent substance, it is difficult to isolate, especially when the background is not plain. Therefore, we apply simple filters to try to remove the background without degrading too much the quality of the smoke in the image. The first step is to compute the median value of the background using images without smoke and then to subtract this from the images containing smoke. We finally apply a threshold filter to complete the background de-noising.

3.2 Point clouds

Merging clouds. To visualise the smoke as a single point cloud, the three point clouds sent by the LiDARs must be merged. To do so, the position given by the motion capture system is used to project all the points in the same global frame. Figure 3 gives the transformation matrices to apply to project a point on the global frame ($ref\ frame$ in the figure). Another possibility which was not used here is to use the ICP algorithm with all LiDARs 2 by 2 to find all the needed transformations.

Labelling and Smoke extraction. The first challenge in



(a) Octree reference map without the smoke

(b) Octree map with new smoke voxels

Figure 5: Visualisation of the octree map created to label and extract the points corresponding to the smoke plume.

processing LiDAR data once all point clouds are merged is the segmentation. More specifically, our aim is to identify which points correspond to the smoke plume so that we can extract them from the rest, as shown in figure 2, or just label them.

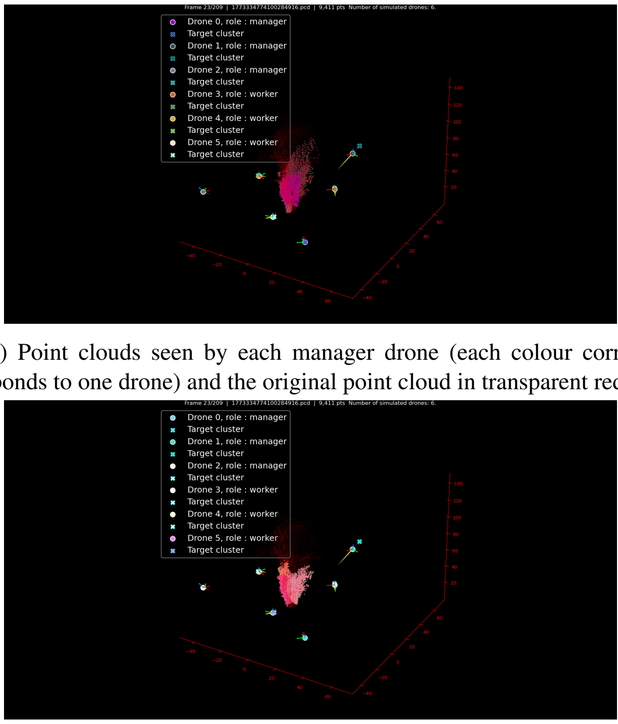
There are several segmentation algorithms, such as those based on the RANSAC algorithm, which attempts to identify specific shapes within the point cloud [26]. Other machine learning-based methods improve segmentation performance by also detecting dynamic objects [27]. In our case, we are seeking to extract a dynamic object with a potentially highly variable shape, rendering RANSAC-based algorithms ineffective. Furthermore, given the limited (or even non-existent) availability of LiDAR smoke data, the use of machine learning-based algorithms appears ill-suited.

Consequently, the decision was made to use a simpler approach, albeit one that is less adaptable to real-world situations. Given that the plume of smoke is the only dynamic object in the scene, each point cloud can be compared to a reference map created beforehand using data of the surrounding environment that excludes the smoke.

The reference map is the sum of the first point clouds of a sequence, without the smoke, and then converted into an octree grid. Then, for each new point cloud of the sequence, we generate a new octree which is compared to the reference map. All the voxels of the octree that were not here in the reference map are considered to contain points of smoke. To avoid outliers, a statistical outlier removal filter is applied to the remaining point cloud. Figure 5 shows two octree grids, one with and the other without the smoke. The voxels corresponding to the smoke are inside the red shape in figure 5b. The extraction of smoke can be applied in real time during the experimentations or afterwards using the C++ PCL library [28].

4 CASE STUDY : SWARM DRONE GUIDANCE

To demonstrate the usefulness of this data in the context of a drone swarm, we present an implementation of drone swarm guidance based on 3D point clouds recorded using the setup described in this paper. We aim to simulate a case of a drone swarm observing a smoke plume with a LiDAR sensor and reacting to its evolution.



(a) Point clouds seen by each manager drone (each colour corresponds to one drone) and the original point cloud in transparent red.

(b) Point clouds seen by each worker drone (each colour corresponds to one drone) following a circle around the centroid point computed by managers and the original point cloud in transparent red

Figure 6: Manager and worker drones organised to observe the smoke plume. Each circle represents a drone.

Simulated LiDAR. In order to replicate a real-world scenario as closely as possible, in which each drone has its own incomplete perception of the smoke, a basic LiDAR simulation is integrated into each drone. To create this simulated LiDAR, the real specifications of the Livox MID-360 LiDAR are used to create a volume where the drone can perceive the point cloud. We arbitrarily decide to keep only the 30% closest points in order to not detect the points behind the surface, as it is the case with a real LiDAR. *Guidance*. Following the idea of [23], we organise the fleet with one manager group and one worker group, each one composed of 3 drones. The first group aims to give a global description of the smoke plume, computing its centroid and main directions of propagation based on their LiDAR observation. To do so, they are set to a static position facing towards the smoke (assuming that the approximate plume position is already known) and far from the plume (but still in the LiDAR range) to have a global view of the scene. Each drone computes the centroid position of their perceived point cloud with their simulated LiDAR, as described previously. Then, the mean of computed centroids is calculated and sent to the worker drones group. In figure 6a, the manager drones are displayed joining their positions to observe the smoke plume and compute the centroid. When receiving their target, the workers will join a circular formation and rotate around the smoke, facing towards the centroid at a closer distance than the managers. In figure 6, the worker drones are added; they circle around the centre of the smoke plume calculated by the manager drones. Each colour displayed on the smoke plume corresponds to a worker drone's perception.

5 CONCLUSION AND FUTURE WORKS

We presented in this paper an experimental setup to record synchronised and spatially calibrated smoke images and point clouds in a controlled environment, with an application to drone swarm control. We believe that this data can be useful for smoke recognition and drone control around smoke plumes, and a simple example of application is described enabling to better understand our view. The objective of this dataset is also to provide data that can be used for smoke modelling and behaviour learning.
Current works are looking to complete this dataset with a larger range of situations, from a controlled environment with adjusted lighting to an outside environment with natural conditions (wind, sunlight, dynamic background, ...), and developing maximum swarm drone coverage of a smoke plume using the presented data.

ACKNOWLEDGMENTS

The authors acknowledge the support of the French National Research Agency (Agence Nationale de la Recherche, ANR) under reference ANR-23-IAS2-0002-01 for the FireFlies TSIA ANR Project.

REFERENCES

[1] Brauer M. Johnston F. H. Jerrett M. Balmes J. R. Elliott Reid, C. E. Critical review of health impacts of wildfire smoke exposure. *Environmental health perspectives*, 2016.

[2] François Pimont Julien Ruffault Eric Rigolot, Jean-Luc Dupuy. Les incendies de forêt catastrophiques. *Annales des mines - Série Responsabilité et environnement*, 2020.

[3] Sharad Chitlangia Akhil Agnihotri Ding Zhao Hanjiang Hu, Zuxin Liu. Investigating the impact of multi-lidar placement on object detection for autonomous driving, 2022.

[4] Jianhao Jiao, Haoyang Ye, Yilong Zhu, and Ming Liu. Robust odometry and mapping for multi-lidar systems with online extrinsic calibration. *IEEE Transactions on Robotics*, 38(1):351–371, 2022.

[5] Kim Gon-Woo Sualeh Muhammad. Dynamic multi-lidar based multiple object detection and tracking. *Sensors*, 19(6), 2019.

[6] Murcia Moreno Harold. *Characterisation of Trees in Orchards using 3D LiDAR Data Collected by Mobile Robots*. Theses, Université de Toulouse, December 2025.

[7] Jimenez-Berni & al. High throughput determination of plant height, ground cover, and above-ground biomass in wheat with lidar. *Frontiers in Plant Science*, Volume 9 - 2018, 2018.

[8] Escolà Alexandre & al. Mobile terrestrial laser scanner applications in precision fruticulture/horticulture and tools to extract information from canopy point clouds. *Precision Agriculture*, 18(1):111–132, Feb 2017.

[9] Lin Cao Guibin Wang Fuliang Cao Kun Liu, Xin Shen. Estimating forest structural attributes using uav-lidar data in ginkgo plantations. *ISPRS Journal of Photogrammetry and Remote Sensing*, 146:465–482, 2018.

[10] Xiaoxia Zhao & al. Analysis of uav lidar information loss and its influence on the estimation accuracy of structural and functional traits in a meadow steppe. *Ecological Indicators*, 135:108515, 2022.

[11] Cheng Zhao Dexin Yao Guanyin Chen Li Sun Tomas Krajnik Zhi Yan Tao Yang, You Li. 3d tof lidar in mobile robotics: A review, 2022.

[12] Mcaree Peter Phillips Tyson, Guenther Nicky. When the dust settles: The four behaviors of lidar in the presence of fine airborne particulates. *Journal of Field Robotics*, 34, 02 2017.

[13] C Stiller R Urtasun A Geiger, P Lenz. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[14] I Grant. Particle image velocimetry: A review. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 211(1):55–76, 1997.

[15] Tim Hawkins, Per Einarsson, and Paul Debevec. Acquisition of time-varying participating media. In *ACM SIGGRAPH 2005 Papers*, SIGGRAPH '05, page 812–815, New York, NY, USA, 2005. Association for Computing Machinery.

[16] Christian Fuchs, Tongbo Chen, Michael Goesele, Holger Theisel, and Hans-Peter Seidel. Volumetric Density Capture From a Single Image. In Raghu Machiraju and Torsten Moeller, editors, *Volume Graphics*. The Eurographics Association, 2006.

[17] Wieneke Bernhard Oudheusden Bas Elsinga Gerrit, Scarano Fulvio. Tomographic particle image velocimetry. *Experiments in Fluids*, 41:933–947, 12 2006.

[18] Wolfgang Heidrich Art Tevs Derek Bradley Marcus Magnor Hans-Peter Seidel Bradley Atcheson, Ivo Ihrke. Time-resolved 3d capture of non-stationary gas flows. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 27(5):132, 2008.

[19] James Gregson, Ivo Ihrke, Nils Thuerey, and Wolfgang Heidrich. From capture to simulation: connecting forward and inverse problems in fluids. *ACM Trans. Graph.*, 33(4), July 2014.

[20] Marie-Lena Eckert, Kiwon Um, and Nils Thuerey. Scalarflow: a large-scale volumetric data set of real-world scalar transport flows for computer animation and machine learning. *ACM Trans. Graph.*, 38(6), November 2019.

[21] Yue Gao, Hong-Xing Yu, Bo Zhu, and Jiajun Wu. Fluidnexus: 3d fluid reconstruction and prediction from a single video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025.

[22] N. P. Lareau and C. B. Clements. Environmental controls on pyrocumulus and pyrocumulonimbus initiation and development. *Atmospheric Chemistry and Physics*, 16(6):4005–4022, 2016.

[23] Nikil Krishnakumar, Shashank Sharma, Srijan Kumar Pal, and Jiarong Hong. 3d characterization of smoke plume dispersion using multi-view drone swarm, 2025.

[24] Klapa Przemysław Mitka Bartosz. Procedure and accuracy assessment results of livox mid-360 scanners. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLVIII-1/W6-2025:147–154, 12 2025.

[25] Ieee standard for a precision clock synchronization protocol for networked measurement and control systems. *IEEE Std 1588-2008 (Revision of IEEE Std 1588-2002)*, pages 29–34, 2008.

[26] Klein Reinhard Schnabel Ruwen, Wahl Roland. Efficient ransac for point-cloud shape detection. In *Computer graphics forum*, volume 26, pages 214–226. Wiley Online Library, 2007.

[27] Mersch Benedikt Wiesmann Louis Gall Jürgen Behley Jens Stachniss Cyrill Chen Xieyuanli, Li Shijie. Moving object segmentation in 3d lidar data: A learning-based approach exploiting sequential data. *IEEE Robotics and Automation Letters*, 6(4):6529–6536, 2021.

[28] Radu Bogdan Rusu and Steve Cousins. 3D is here: Point Cloud Library (PCL). In *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 9-13 2011. IEEE.