Autonomous UAV Volcanic Plume Sampling Based on Machine Vision and Path Planning

Rolland, Edouard; Grøntved, Kasper Andreas Rømer; Christensen, Anders; Watson, Matthew; Richardson, Tom

Go to publication entry in University of Southern Denmark's Research Portal

Autonomous UAV Volcanic Plume Sampling Based on Machine Vision and Path Planning

Edouard G. A. Rolland[1] ⬡ Kasper A. R. Grøntved[1] ⬡ Anders Lyhne Christensen[1] ⬡
Matthew Watson[2] ⬡ Tom Richardson[3] ⬡

Abstract—Drones currently serve as a valuable tool for in-situ sampling of volcanic plumes, but they still involve manual piloting. In this paper, we enable autonomous dual plume sampling by using a machine vision model to detect eruptions. When an eruption is detected, a sampling trajectory is automatically generated to intercept the plume twice to collect comparative samples. The machine vision model is developed by training a YOLOv8 object detection model thanks to a database of 1505 images that feature labelled plumes. The obtained average precision value of the model's plume class, at 90.7%, is comparable to that of state-of-the-art models for wildfire smoke monitoring. The performance of this method is assessed using a *software-in-the-loop* simulation of the drone and a simulated plume model. Although the results confirm the efficacy of using a machine vision model for triggering an onboard path-planning algorithm, it also suggests the potential for a hybrid strategy that integrates visual servoing with our proposed path-planning approach.

◆

This is the pre-print version of the paper. The final version has been published in 2024 International Conference on Unmanned Aircraft Systems (ICUAS) and can be accessed at 10.1109/ICUAS60882.2024.10556912.

1 INTRODUCTION

The emission of ash, gas, and aerosols into the atmosphere by volcanoes can have severe consequences for people living nearby, including decreased air quality, contamination of drinking water systems, and impacts on air traffic [11]. Fine airborne ash of less than 63 microns, at high altitudes, can compromise aircraft reliability and potentially lead to engine failure [4, 10]. Precise anticipation of plume movements is essential to alert of potential risks and enable decision-makers to implement measures that reduce the socioeconomic impact of eruptions.

Ash dispersion can be predicted using advection-diffusion models based on meteorological and satellite data [38]. To use these models, it is necessary to know how the *particle size distribution* (PSD) evolves within a plume over time after the eruption. The PSD of the plume changes with the distance it travels, influenced by the wind, resulting in a faster descent of heavier ash particles to the ground [28, 36]. However, the responsiveness of remote sensing for PSD evaluation is limited due to the timing of satellite overpasses and varying weather conditions, which can delay or affect the realisation of these measurements [29, 43]. An alternative method using drones for in-situ and timely measurement of the PSD evolution can overcome the limitations of satellite-based techniques and thus increase the frequency of updating the prediction of the volcanic ash dispersion models [36]. Drones have emerged as a promising tool, offering new possibilities for the collection of volcanic ash samples, which are unbiased unlike manual collection of ash from the ground [42].

In this paper, we present a method for conducting autonomous PSD evolution assessment using fixed-wing drones. This method involves comparative sampling to monitor the PSD evolution of a specific plume over time, enabling the collection of multiple PSD measurements for analysis. In this context, Schellenberg et al. [35] introduced a path-planning strategy that enabled the sampling of the same volcanic plume at two separate intervals. It allows the plume to travel a distance of 5 to 15 kilometres between interceptions, thereby facilitating the collection
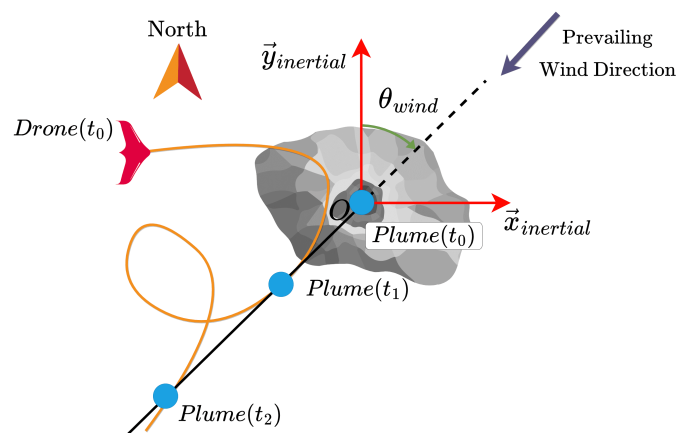

Fig. 1. At t_0 in the inertial frame, the machine vision model detects a plume. The drone autonomously generates a trajectory to intercept it at t_1 and t_2. The inertial frame is defined at the summit of the volcano.

of significant comparative data. The method, known as *coordinated plume interception* (CPI), enables the drone to generate a trajectory that takes into account anemometric conditions. The drone starts from the first sampling point at time t_1 and follows a trajectory to intercept the plume again at time t_2, as illustrated in Fig. 1. Nevertheless, the capability of the path-planning method introduced by Schellenberg et al. [35] to perform dual interceptions has not been confirmed through a simulation-based approach or real-world trials. The approach furthermore requires manually bringing the drone to the first sampling point $Plume(t_1)$. Work thus needs to be done to enable the realization of a fully autonomous dual plume sampling system. One step towards achieving full autonomy is to enable the drone to detect and respond to plumes in real time. Such a system would not only reduce the pilot's cognitive load but also enhance the robustness of comparative plume measurements.

In this paper, we explore autonomous dual sampling of plumes using a computer vision method. Our study makes three major contributions: firstly, the creation of a database containing labelled images of plumes and the summit of Volcano Fuego in Guatemala [33]. Secondly, this database is used to train a YOLOv8 plume detection model [32], capable of identifying volcanic plumes and the volcano's summit. Finally, we propose applying this computer vision model to an algorithm for autonomous path planning for the dual interception of volcanic ash plumes. This model enables the detection of an eruption at a specific moment (denoted as t_0 in Fig. 1) and then generates an ad-hoc

[1] *UAS Drone Center, MMMI, University of Southern Denmark (SDU), Campusvej 55, DK-5230 Odense M,* {`edr, kang, andc`}`@mmmi.sdu.dk`

[2] *School of Earth Sciences, Wills Memorial Building, University of Bristol, Queens Road, Clifton, Bristol BS81RJ,* `glimw@bristol.ac.uk`

[3] *Department of Aerospace Engineering, Queens Building, University of Bristol, University Walk, Bristol BS81TR,* `aetsr@bristol.ac.uk`

dual-sampling trajectory. The performance of these methods is evaluated through simulations that combine the use of PX4 *software in the loop* (SITL) [21] with a plume movement model. This study thus demonstrates the possibility of combining the plume vision model and the dual interception path-planning algorithm, offering innovative possibilities for onboard planning and sample collection.

2 RELATED WORK

With the advent of drones, researchers now have access to a flexible and easily deployable platform to conduct experiments in previously hard-to-reach and unsafe locations [42]. The literature references the use of drones as early as 2007 in the context of volcano surveillance [20]. Experiments conducted with drones use both fixed-wing and multi-rotor UAVs [5, 18, 44], which have been progressively integrated into volcano monitoring protocols [39]. Scientific publications rarely, if ever, mention the existence of a fully autonomous drone system from take-off to landing that conducts in-situ sampling of volcanic plumes and has been tested in the field. Ash sample collection is carried out using two different strategies: the predominant method involves manual piloting in *first-person view* (FPV) [9, 18, 36, 44]. Another approach also aims to have the drone follow a fixed flight plan above the sampling area and then hope that it enters the plume [22, 44]. Recent research combines the above-mentioned approaches, with autonomous ascents and descents, while sample collection is performed manually [35, 36, 44]. Without the ability to identify plumes, autonomous plume interception by drone systems is simply not possible. Furthermore, to achieve autonomous plume sampling, it is imperative to detect ash plumes with high accuracy.

The visual detection of plumes is most commonly performed using thermal infrared imagery [3, 27, 34]. In this study, however, we explore volcanic plume detection using visible-wavelength cameras to avoid the cost and complexity of an onboard thermal camera. Limited research has delved into this approach, primarily utilizing images from volcano ground monitoring stations [13, 37]. This machine vision task bears a resemblance to the detection of smoke plumes from wildfires, an extensively studied problem.

To detect plumes, researchers have explored methods based on the analysis of images, relying on the examination of texture or colour changes across image sequences to detect volcanic plumes or smoke plumes in pictures [26,37,52]. However, these methods have limitations. As pointed out respectively by Celik et al. [52] and Simionato et al. [37], the colour of smoke, and its contrast with the background are not parameters robust enough for broad application. Additionally, most of these methods necessitate a fixed camera viewpoint incompatible with fixed-wing drone flights.

Deep learning, particularly *convolutional neural network* (CNN), offers an alternative to image analysis methods for plume detection. The efficacy of CNNs in object classification is well-documented [14] with studies like Frizzi et al. [8] achieving 97.9% accuracy in detecting fire-related elements using a database of flame and smoke images. Similarly, Guerrero et al. [13] trained two different CNN segmentation architectures (U-Net and SegNet) and achieved respective accuracies of 98.3% and 95.56% using a database containing pictures captured by ground-based cameras around Mount Etna in Italy. CNNs are typically composed of two main blocks: the first extracts features from images by applying multiple convolution operations, and the second performs the classification task. It can autonomously select discriminating features for object detection, offering versatility and adaptability to specific tasks [14, 24]. The layer arrangement in a CNN is crucial for optimal performance, necessitating architecture selection tailored to the problem discussed in this paper. Zheng et al. [49] compared architectures like EfficientDet, Faster R-CNN, YOLOv3, SSD, and advanced CNN models for wildfire smoke detection, considering speed and accuracy. EfficientDet excelled in accuracy (95.7%), while YOLOv3 offered the best speed-accuracy balance. YOLO's accuracy in smoke detection is also supported by other studies [47, 48]. Mukhiddinov et al. [23] improved prediction speed and small smoke detection using transfer learning and modified architecture based on a 6,000-image aerial footage database.

At the beginning of 2023, Ultralytics released version 8 of YOLO, which quickly found application in research related to wildfire and smoke detection through surveillance cameras [40, 46]. Yandouzi et al. [46] found YOLOv8 to offer the best accuracy-speed balance among new machine learning models. Given its proven accuracy and low detection speed in similar applications, we have decided to use the YOLOv8 architecture for volcanic plume detection, especially considering the computational limitations of drones where fast models are essential to minimize latency. Though YOLO models may confuse smoke plumes with meteorological phenomena like fog, haze, or clouds [2, 48]. To enhance accuracy, al-Omari et al. [1] and Mukhiddinov et al. [23] suggest incorporating cloud shape and size as key features. Despite these challenges, significant results have been achieved, with Mukhiddinov et al. [23] reporting an average precision of 73.6% for smoke plume detection.

3 MACHINE VISION FOR PLUME & SUMMIT DETECTION

Transfer learning of a CNN requires a labelled dataset of training and validation images. Since databases on volcanic plumes are currently not widely available, we created a new database of annotated volcanic plumes for this study. This database contains images of the Fuego volcano, enabling the trained model to be fine-tuned to the environment of Guatemala and the plumes from stratovolcanoes.

3.1 Data collection

We use data from the University of Bristol Flight Lab field study to Guatemala, which took place from March 22 to April 3, 2019. The drone used for these missions was a Skywalker X8, equipped with a Pixhawk onboard computer running ArduPlane 3.7.1 and a Raspberry Pi 3B+, for mission management and communication with the ground station. The drone was equipped with two high-resolution cameras: a GoPro Hero 9 for recording a ground view and another for capturing the front view [36]. In total, 832 minutes of footage were collected from each camera over 14 flights, averaging 59.4 minutes per flight. We converted the raw data into pictures with a sampling rate of 1 image every 10 seconds for the labelling.

3.2 Training and Validation Sets

In this study, we use the images from twelve flights to create the training and validation sets. However, it is necessary to remove the pictures that do not feature the volcano's summit or plumes. This is because the front-facing camera does not capture these elements during the drone's ascent and descent phases. The 1505 images extracted provide a diverse representation of the plumes and summit in varying environmental conditions, e.g. lighting, and weather. Fig. 2 presents examples of images from these two sets.

Fig. 2. Example of images from the training and validation sets

We separate the pictures between the training and validation sets using the common 80%–20% ratio.

3.2.1 End-to-End Test Flight Set

The data from the two remaining flights is used to create a set that shows two complete flights, called *test flight 1* and *test flight 2*, from takeoff to landing, with different weather conditions. This is done to fully evaluate the model's performance and prepare it for real-world use. *Test flight 1*'s 214 images show clear weather conditions with no clouds, representing optimal conditions where plumes are easily distinguishable. *Test flight 2*'s 242 images feature approximately 60% cloud cover, representing more challenging conditions for plume detection. This dataset allows assessment of the model's effectiveness in both ideal and adverse weather scenarios, akin to real-world flight conditions.

3.2.2 Labelling

We use Roboflow, a software for building object detection training databases, to label the images [7]. We manually label plumes in the images. However, it is sometimes difficult to distinguish a plume from a cloud. In cases where doubt remains about the nature of the cloud, it is left unlabelled.

As an additional class in the dataset, we also label the summit of the Fuego volcano. This enables the training of the detection model to identify this specific summit. This feature could be beneficial during flight operations, allowing operators to quickly distinguish Fuego from nearby peaks in an FPV video feed. To label an object, each side of the bounding box must be tangent to the object's contour. However, defining a closed contour for the summit of the volcano is not feasible. Thus, to ensure consistent labelling in the database, we establish reference points on the volcano summit that the bounding box should intersect with, as illustrated in Fig. 3.
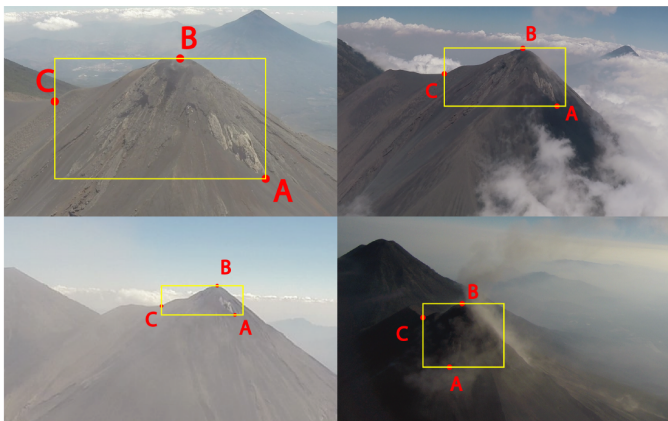

Fig. 3. Labelling of the volcano summit considering reference points

3.2.3 Pre-Processing

To reduce the training time, we follow the training recommendations provided by the YOLO developers [41]: the size of each image is reduced to the dimensions of 640×640 pixels.

Additionally, to increase the amount of training data [16, 45], we generate new images from the existing ones by modifying their hue variation, saturation, brightness, exposure, and introducing noise. As a result, the training set grows from 1211 images to 3633 images. We provide access to the database of the 1211 images through Hugging Face [33].

3.3 Model Evaluation

To evaluate the performance of the trained model, we use *intersection over union* (IOU), which is a common metric within object detection and localisation. The IOU quantifies how well a predicted bounding box matches the ground-truth box. A threshold value is then used to classify the predictions as correct or incorrect. It is now possible to quantify the number of *false negatives* (FN), *false positives* (FP), and *true positives* (TP). In the context of object detection, the number of *true negatives* is infinite, as any background element can be considered a true negative [25].

The *precision* and *recall* metrics are defined as follows:

$$\text{Precision} = \frac{TP}{TP+FP} \qquad \text{Recall} = \frac{TP}{TP+FN} \qquad (1)$$

Recall is thus the true positive rate, denoting the ability of our model to detect all plumes or the summit. Precision is the proportion of true positives among all the instances classified as positive, thus denoting the ability of the model to correctly detect the plumes and the summit of the volcano [51].

The *precision-recall* curve, which is generated by evaluating precision and recall at various confidence thresholds, assesses *average precision* (AP), and is defined as the area under the precision-recall curve. The AP evaluates a model's overall performance regarding detecting a single object class. Additionally, this curve aids in selecting a confidence threshold regarding the user expectations in terms of precision and recall [25].

The *mean average precision* (mAP) provides a comprehensive measure of a detection model's average accuracy in detecting various kinds of objects. In the context of this study, mAP corresponds to the mean AP value for both the 'plume' and 'summit' objects [25].

3.4 Training Setup

YOLOv8 provides with various pre-trained models, allowing us to benefit from transfer learning [23]. We use the YOLO-v8s model because it strikes the best balance between detection speed and mAP [15]. The default hyperparameters provided with YOLO-v8s are used to train the model. Furthermore, we train the model for a maximum of 300 epochs, with early stopping to avoid overfitting and stopping patience of 50 epochs.

The training is carried out on the Google Colab platform, which provides access to powerful CPUs and GPUs. Specifically, the setup used consists of an NVIDIA Tesla T4 GPU with, 4 CPUs and 25.5 GB of RAM. After the training, the model is assessed on the validation set to gain an overview of its general performance, thanks to the metrics detailed above. Additionally, as a complementary assessment, the model's performance is also evaluated on the end-to-end test flight dataset.

3.5 Results

The model converged after 2.7 hours of training and finished at epoch 133 though the YOLO model did not achieve significant performance improvements after epoch 83 as shown in Fig. 4 and 5. Figure 4 presents the bounding box regression loss, which describes how well predicted bounding boxes capture objects [50]. We provide access to the model weights via Hugging Face [32].
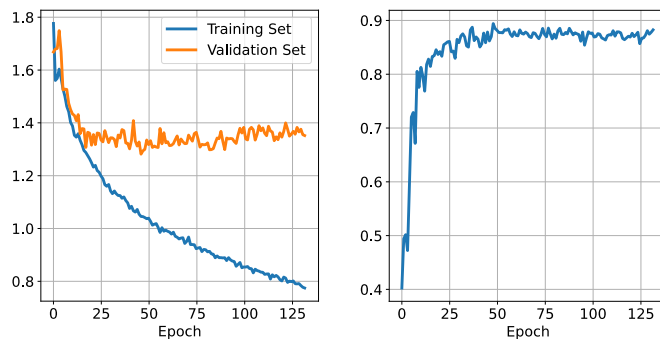

Fig. 4. Bounding box regression loss Fig. 5. mAP (IOU threshold − 50%)

3.5.1 Validation Set

The results obtained are summarized in Fig. 6 and 7. We obtained an AP value of 90.7% (at an IOU threshold of 50%) for the plume class. These performance metrics match the work in similar domains, such as smoke detection [23, 47].
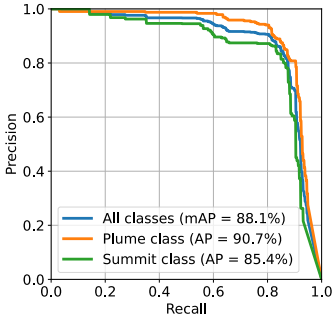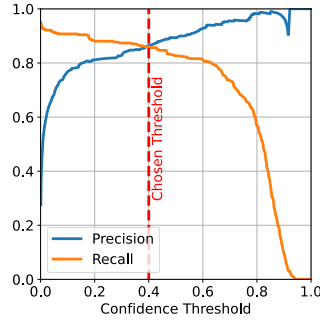
Fig. 6. Precision-recall curve

Fig. 7. Recall and precision for plume class by confidence threshold

3.5.2 End-to-End Test Flight Set

To carry on the evaluation of the performances of the model on the test flight datasets, we need to define the confidence threshold. The confidence threshold should be chosen based on user preference: either to maximize recall or precision. Based on the results in Fig. 7, we chose a confidence threshold of 41%, which yields a balance between recall and precision. Thus, the resulting model allows for predictions on this set, yielding the metric values presented in Table 1.

Table 1. Performance of the model on the two test flights

	Test flight 1		Test flight 2
Object	Summit	Plume	Plume
Precision	100.0%	99.1%	76.8%
Recall	92.2%	94.1%	80.3%

The summit class is not considered for test flight 2 as clouds covered the summit in that flight. Fig. 8 and 9 show snapshots of the predictions for the two flights.

When the sky is cloud-free, plume detection is nearly perfect (precision = 99.1% and recall = 94.1%). The values of recall and precision decline as the weather conditions change, and it is observed that the model considers clouds located at the periphery of the field of view of the camera as plumes. In test flight 1, it is worth noting that the model can detect the summit at a distance of 8 km. The video footage is available on YouTube [30, 31]. With the chosen confidence threshold, it shows 88.0% precision and 83.6% recall, effectively detecting plumes and summits in most instances. Despite the challenge of distinguishing clouds from plumes with a standard camera, the machine learning model's performance is satisfactory. Even in the challenging scenario (test flight 2), it achieves 76.8% precision and 80.3% recall. Additionally, for real-world applications, ground-station scientists could manually validate detections, thereby reducing the impact of false positives on the mission. Our data was collected in the dry season. Thus, we cannot assess the model's performance under the more challenging conditions of the wet season (fog, heavy clouds). Conducting a dedicated field trip to validate the model and gather additional data will enhance dataset diversity, leading to improved model generalization and robustness.

4 PATH PLANNING FOR AUTONOMOUS DUAL PLUME INTERCEPTION

Schellenberg et al.'s [35] path-planning method requires manual intervention to guide the drone to the initial sampling point. Leveraging the machine vision model, we assume that the system is now capable
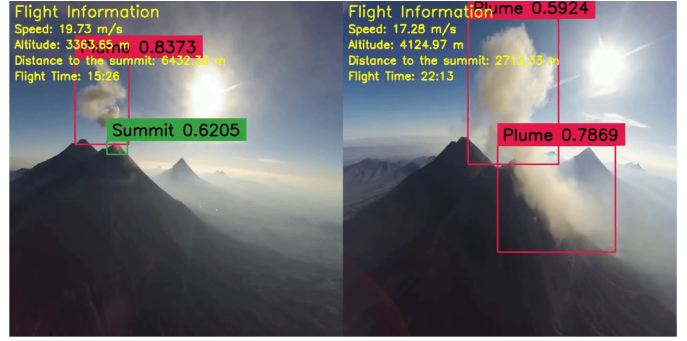
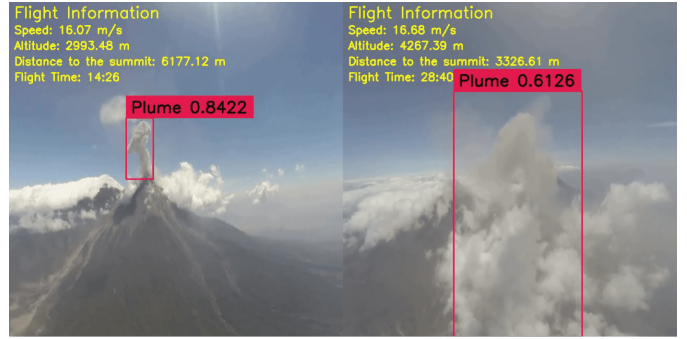Fig. 8. Plume and summit detection model output on test flight 1

Fig. 9. Plume detection model output on test flight 2

of detecting the emergence of a plume in the atmosphere. To enhance the autonomy of the system, the model can be employed as a means of detection to trigger the generation of an ad-hoc trajectory, as depicted in Fig. 1. The trajectory is divided into two parts: the *approach trajectory*, which enables the drone to reach the first sampling point after detection, and the *CPI trajectory* based on Schellenberg's approach [35]. This CPI trajectory allows the drone to sample the plume again after a few minutes, allowing the plume to travel several kilometres between intercepts.

4.1 Hypothesis and Path Planning Methodology

To generate the complete dual-plume interception trajectory, we base our method on Schellenberg's approach and assumptions [35]. The plume is considered as a point, and its movement is assumed to be rectilinear and uniform, with its characteristics defined by the wind vector. In order to generate the CPI trajectory, Schellenberg first calculates the trajectory in the wind frame: this consideration simplifies the calculations as, in this frame of reference, the plume is stationary. To derive the trajectory in the inertial frame, the trajectory is discretized into waypoints. The conversion is facilitated using equation 2, considering the time it takes for the drone to reach each waypoint. The time required to reach each waypoint, denoted as t_{wp}, is determined by considering the distance to the waypoint in the wind frame and the drone's speed in this frame (V_a, drone's airspeed), assumed to be constant.

$$\begin{bmatrix} X_{wp} \\ Y_{wp} \end{bmatrix} = \begin{bmatrix} X \\ Y \end{bmatrix} - t_{wp} W_{speed} \begin{bmatrix} \sin\theta_{wind} \\ \cos\theta_{wind} \end{bmatrix} \quad (2)$$

Where (X_{wp}, Y_{wp}) is the waypoint coordinates within the reference frame in meters, and corresponds to (X, Y) in the wind frame.

4.2 Approach Trajectory

To maximize the chance of a successful initial interception, a smooth transition between the trajectory phases, approach, and CPI trajectory

is essential. The challenge here is to find a method of generating a trajectory that connects the drone's position at the moment the plume appears to the first sampling point $P_{1,2}$ depicted in Fig. 10. This can be achieved by aligning the drone's heading vector (θ_{drone}) with the wind vector as the drone reaches the point $P_{1,2}$. With this condition, the trajectory is realized using 'Dubins Paths' which enables the creation of a tailored trajectory between two points in a two-dimensional space while also considering the initial and final orientation of the tangent [6].

4.3 CPI Trajectory

Schellenberg et al. [35] proposed five different types of CPI trajectories and ranked their relevance according to several parameters: energy consumption, flight zone, and resistance of the trajectory to wind variations [35]. Their simulations conducted using ArduPilot SITL revealed that a downwind circular trajectory in the wind frame yielded the best results. When converted into the inertial frame, it results in a loop, as shown in Fig. 1. The radius describing the circular trajectory in the wind frame, which allows sampling the plume at t_1 and t_2 for computing the drone's trajectory between $Plume(t_1)$ and $Plume(t_2)$, is detailed in equation 3. $d_{P_1 P_2}$ denotes the distance between the two sampling points and is a user-defined parameter.

$$R_{wind} = \frac{d_{P_1 P_2} V_a}{2\pi W_{speed}} \quad (3)$$

Figure 10 illustrates a complete dual-interception trajectory in the wind frame, incorporating the aforementioned considerations regarding the approach path. The parameters employed to generate this example include $\theta_{drone}(t_0) = 10°$, $d_{P_1,P_2} = 5\,\text{km}$, $\theta_{wind} = 20°$, $W_{speed} = 10\,\text{m s}^{-1}$, and $V_a = 22\,\text{m s}^{-1}$.
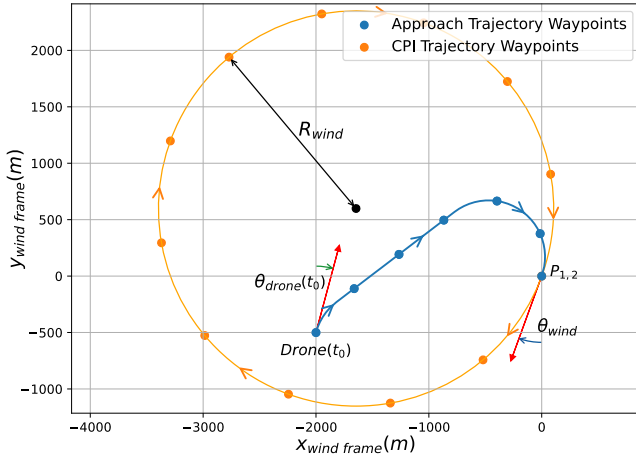

Fig. 10. Example of a dual interception trajectory in the wind frame, triggered by the machine vision model

It is now possible to generate a complete interception trajectory regardless of the drone's initial position, as illustrated in Fig. 11. The red trajectory represents a real ascent path based on Schellenberg's data [36] collected in Guatemala. The figure depicts the simulation of the appearance of a plume (blue trajectory) during this ascent, detected by the machine vision model, leading to the generation of a dual plume trajectory (in yellow).

4.4 Simulation and Experiments

To validate the efficacy of the path-planning method, it is necessary to assess the drone's position in conjunction with the plume's position using a simulated model of each element.

4.4.1 Plume Simulation

To obtain a numerical plume model, we use the hypotheses mentioned in Section 4.1 again. We implement the plume model using the equa-
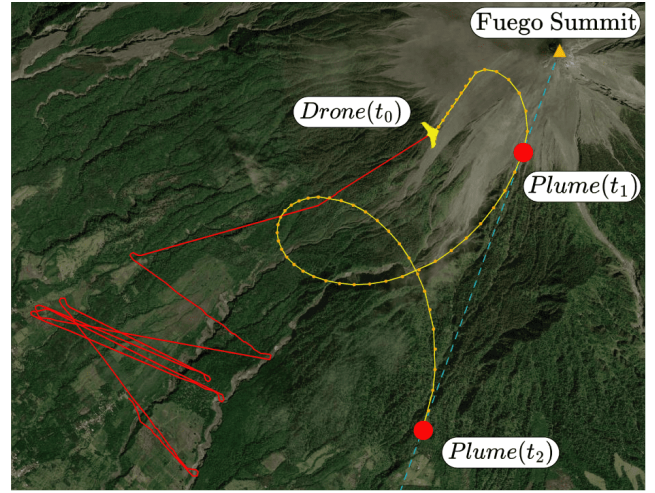

Fig. 11. Simulation of the generation of a complete interception trajectory - $W_{speed} = 10\,\text{m s}^{-1}$ - $\theta_{wind} = 20°$ - $d_{P_1,P_2} = 5\,\text{km}$ and $V_a = 22\,\text{m s}^{-1}$

tion 4, derived from Newton's third law, to describe the plume's movement after the instant of an eruption (t_e).

$$\begin{bmatrix} x_{plume}(t) \\ y_{plume}(t) \end{bmatrix} = (t_e - t) \cdot W_{speed} \begin{bmatrix} \sin(\theta_{wind}) \\ \cos(\theta_{wind}) \end{bmatrix} \quad (4)$$

4.4.2 Drone Simulation

The execution of the generated flight path is simulated using PX4 SITL [21], the ROS2 humble robotic middleware [19], and a simulation environment in Gazebo Ignition [17]. The PX4 model of a generic Vertical Take-Off and Landing (VTOL) drone is used for the experiments, with unlimited energy and a defined airspeed of $15\,\text{m s}^{-1}$. The mission plans' waypoints are generated using our Python-based software and then pushed to PX4 via QGround Control. To streamline the development process, we created a development environment based on a Docker container tailored for PX4, ROS2, and Gazebo development, which has been released on GitHub [12].

4.4.3 Experimental Design

To evaluate the effectiveness of the proposed path-planning method, it is imperative to employ a pertinent metric for identifying interceptions. In this context, we utilize the Euclidean distance, denoted as $d(t)$, which measures the distance between the drone and the plume over time. This means that for the dual interception path, $d(t)$ will have two minima where the drone is supposed to intercept the plume as shown in Fig. 12. Perfect interceptions would imply $d_1 = d_2 = 0$. The tolerance for d_1 and d_2 is determined based on the actual diameter of plumes emitted by the volcano.

We assess the resilience of our path-planning method under various wind speeds ($1\,\text{m s}^{-1}, 2\,\text{m s}^{-1}, 3\,\text{m s}^{-1}, 4\,\text{m s}^{-1}$) and a fixed wind direction of $20°$. The initial drone position is fixed for all experiments, with $x_{drone} = y_{drone} = -2500\,\text{m}$, and the drone's initial yaw points toward the summit. The distance between points P_1 and P_2 is $d_{P_1,P_2} = 5\,\text{km}$.

4.5 Results

Figure 13 presents the results for the conducted simulations, and it is evident that none of the tested wind speeds allows for achieving a perfect interception of the plume. The values of d_1 and d_2 increase as the wind speed becomes higher. Consequently, with higher wind speeds, the size of the intercepted plumes should also be larger. Nevertheless, the proposed method is sensitive to the wind but provides performance levels compatible with achieving a dual interception. We hypothesize that this error is introduced by not considering the impact of drag in equation 2.
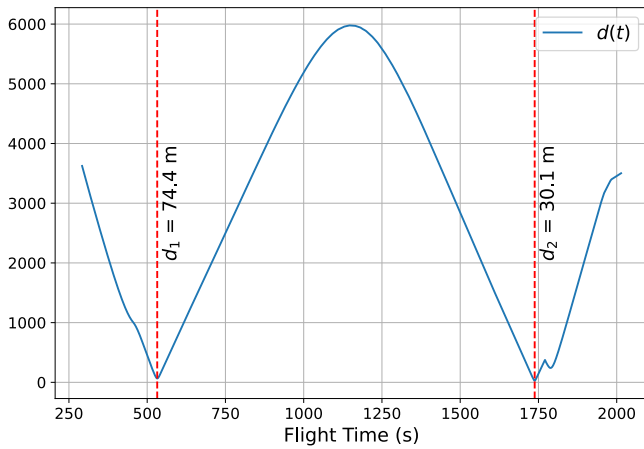
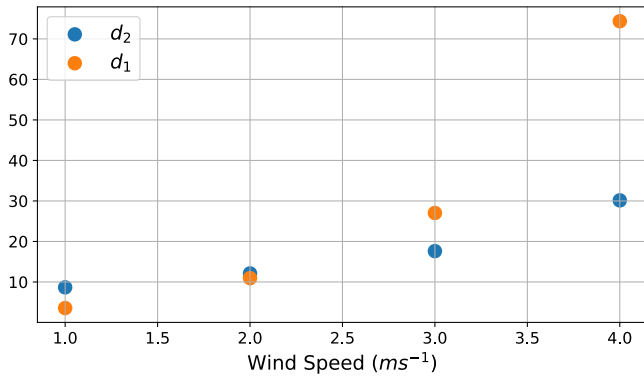Fig. 12. $d(t)$ for a $4\,\mathrm{m\,s^{-1}}$ wind speed

Fig. 13. Path planning resilience to wind variation

The simulation results thus show that the proposed path-planning method based on machine vision triggering has the potential to enable autonomous dual sampling of the plume. However, before deploying our system in the field, it is essential to enhance its robustness. The developed path-planning method relies on several strong hypotheses: the characteristics of plume movement, the assumption of constant wind conditions, and the capability of the machine vision model to detect the plume right after its emergence in the atmosphere. Thus, these assumptions might not hold in a real-world scenario, potentially increasing the values of d_1 and d_2. To bridge the gap between simulations and reality, and utilizing the methods developed in this paper, a mixed approach combining path planning and visual servoing could be pursued. In this way, the machine vision model could be used in the pre-sampling phases as an input for a closed control loop for guiding the drone to the plume. After sampling, the drone can follow the dual interception trajectory and then switch back to visual servoing mode several hundred meters ahead of the next interception.

5 Conclusion and Future Work

Drones are increasingly being incorporated into volcanic monitoring protocols and are expected to be used more frequently for in-situ sampling. In this paper, we presented a novel approach for autonomous dual-plume sampling based on machine vision triggering, enabling the generation of an ad-hoc trajectory. We created a database of 1505 labelled images, consisting of plumes and the summit of the Fuego Volcano in Guatemala [33]. A YOLOv8-type object detection model was trained to detect these elements [32]. The model's performance is promising enough to be later embedded in a real drone system. The obtained AP value for plume detection competes with state-of-the-art object-detection models for fire plume detection [23, 47]. Additionally, two test flights were conducted to assess the model's performance in real-world conditions, demonstrating its reliability. To potentially reduce false positive detections caused by clouds, labelling and incorporating them into the training process could enable the model to differentiate between clouds and plumes more effectively.

As a use case of this model, we implemented a path-planning method for dual plume interception triggered by machine vision. The method consistently demonstrated effective dual interception of the same plume. Nevertheless, to enhance the system's resilience to wind, additional efforts should be directed towards implementing a visual servoing algorithm based on our machine vision model. Moreover, from a broader perspective, the challenge of dual interception of a linearly moving object by a fixed-wing drone collecting comparative measurements can be extended to domains beyond volcanology. This approach could also be applied to track the evolution of other elements, such as oil spills, wildfires, floods, or wildlife. Combining a dedicated machine vision model with a predictive model of movement and a path-planning method would allow the acquisition of comparative data.

Our immediate objectives involve evaluating the effectiveness of the plume detection model in real flights above Fuego in Guatemala. Conducting additional experiments in different weather conditions will allow us to gather more diverse data, which will help improve the dataset's quality and subsequently train a better model. Our next objective is to develop the mixed approach in a simulated 3D environment, enabling the implementation of visual servoing, and followed by an end-to-end real-world experiment.

References

[1] Y. Al-Smadi, M. Alauthman, A. Al-Qerem, A. Aldweesh, R. Quaddoura, F. Aburub, K. Mansour, and T. Alhmiedat. Early wildfire smoke detection using different YOLO models. *Machines*, 11(2):246, 2023.

[2] C. Bahhar, A. Ksibi, M. Ayadi, M. M. Jamjoom, Z. Ullah, B. O. Soufiene, and H. Sakli. Wildfire and smoke detection using staged YOLO model and ensemble CNN. *Electronics*, 12(1):228, 2023.

[3] M. Bombrun, D. Jessop, A. Harris, and V. Barra. An algorithm for the detection and characterisation of volcanic plumes using thermal camera imagery. *Journal of Volcanology and Geothermal Research*, 352:26–37, 2018.

[4] W. Chen and L. Zhao. Review–volcanic ash and its influence on aircraft engine components. *Procedia Engineering*, 99:795–803, 2015.

[5] G. Di Stefano, G. Romeo, A. Mazzini, A. Iarocci, S. Hadi, and S. Pelphrey. The Lusi drone: A multidisciplinary tool to access extreme environments. *Marine and Petroleum Geology*, 90:26–37, 2018. doi: 10.1016/j.marpetgeo.2017.07.006

[6] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79(3):497–516, 1957.

[7] B. Dwyer, J. Nelson, J. Solawetz, and et al. Roboflow, 2022. Computer vision software.

[8] S. Frizzi, R. Kaabi, M. Bouchouicha, J.-M. Ginoux, E. Moreau, and F. Fnaiech. Convolutional neural network for video fire and smoke detection. In *IECON 2016–42nd Annual Conference of the IEEE Industrial Electronics Society*, pp. 877–882. IEEE, 2016.

[9] B. Galle, S. Arellano, N. Bobrowski, V. Conde, T. P. Fischer, G. Gerdes, A. Gutmann, T. Hoffmann, I. Itikarai, T. Krejci, E. J. Liu, K. Mulina, S. Nowicki, T. Richardson, J. Rüdiger, K. Wood, and J. Xu. A multipurpose, multi-rotor drone system for long-range and high-altitude volcanic gas plume measurements. *Atmospheric Measurement Techniques*, 14(6):4255–4277, 2021. doi: 10.5194/amt-14-4255-2021

[10] E. Gordeev and O. Girina. Volcanoes and their hazard to aviation. *Herald of the Russian Academy of Sciences*, 84:134–142, 02 2014. doi: 10.7868/S0869587314020121

[11] M. Gouhier, J. Eychenne, N. Azzaoui, A. Guillin, M. Deslandes, M. Poret, A. Costa, and P. Husson. Low efficiency of large volcanic eruptions in transporting very fine ash into the atmosphere. *Scientific reports*, 9(1):1449, 2019.

[12] K. A. R. Grøntved and E. G. A. Rolland. ROS2 & PX4 VSCode Dev Environment, 2021. [Online; accessed 31-January-2024].

[13] J. F. Guerrero Tello, M. Coltelli, M. Marsella, A. Celauro, and J. A. Palenzuela Baena. Convolutional neural network algorithms for semantic segmentation of volcanic ash plumes using visible camera imagery. *Remote Sensing*, 14(18):4477, 2022.

[14] B. Hakim and S. Brahim. *Classification des images avec les réseaux de neurones convolutionnels*. PhD thesis, Université Mammeri, 2018.

[15] G. Jocher, A. Chaurasia, and J. Qiu. Ultralytics YOLOv8, 2023.

[16] P. Kaur, B. S. Khehra, and E. B. S. Mavi. Data augmentation for object detection: A review. In *2021 IEEE International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 537–543. IEEE, 2021. doi: 10.1109/MWSCAS47672.2021.9531849

[17] N. Koenig and A. Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 3, pp. 2149–2154. IEEE, 2004.

[18] E. J. Liu, K. Wood, E. Mason, M. Edmonds, A. Aiuppa, G. Giudice, M. Bitetto, V. Francofonte, S. Burrow, T. Richardson, M. Watson, T. D. Pering, T. C. Wilkes, A. J. S. McGonigle, G. Velasquez, C. Melgarejo, and C. Bucarey. Dynamics of outgassing and plume transport revealed by proximal unmanned aerial system (UAS) measurements at Volcán Villarrica, Chile. *Geochemistry, Geophysics, Geosystems*, 20(2):730–750, 2019. doi: 10.1029/2018gc007692

[19] S. Macenski, T. Foote, B. Gerkey, C. Lalancette, and W. Woodall. Robot operating system 2: Design, architecture, and uses in the wild. *Science Robotics*, 7(66):eabm6074, 2022. doi: 10.1126/scirobotics.abm6074

[20] A. McGonigle, A. Aiuppa, G. Giudice, G. Tamburello, A. Hodson, and S. Gurrieri. Unmanned aerial vehicle measurements of volcanic carbon dioxide fluxes. *Geophysical research letters*, 35(6), 2008.

[21] L. Meier, D. Honegger, and M. Pollefeys. PX4: A node-based multi-threaded open source robotics framework for deeply embedded platforms. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6235–6240. IEEE, 2015.

[22] T. Mori, T. Hashimoto, A. Terada, M. Yoshimoto, R. Kazahaya, H. Shinohara, and R. Tanaka. Volcanic plume measurements using a UAV for the 2014 Mt. Ontake eruption. *Earth, Planets and Space*, 68:1–18, 2016.

[23] M. Mukhiddinov, A. B. Abdusalomov, and J. Cho. A wildfire smoke detection system using unmanned aerial vehicle images based on the optimized YOLOv5. *Sensors*, 22(23):9384, 2022.

[24] K. O'Shea and R. Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.

[25] R. Padilla, S. L. Netto, and E. A. Da Silva. A survey on performance metrics for object-detection algorithms. In *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pp. 237–242. IEEE, 2020.

[26] K.-M. Park and C.-O. Bae. Smoke detection in ship engine rooms based on video images. *IET Image Processing*, 14(6):1141–1149, 2020.

[27] M. R. Patrick, A. J. Harris, M. Ripepe, J. Dehn, D. A. Rothery, and S. Calvari. Strombolian explosive styles and source conditions: insights from thermal (FLIR) video. *Bulletin of volcanology*, 69:769–784, 2007.

[28] F. Prata. Radiative transfer calculations for volcanic ash clouds. *Geophysical Research Letters*, 16:1293–1296, 11 1989. doi: 10.1029/GL016i011p01293

[29] V. H. R. Prudente, V. S. Martins, D. C. Vieira, N. R. de França e Silva, M. Adami, and I. D. Sanches. Limitations of cloud cover for optical remote sensing of agricultural areas across South America. *Remote Sensing Applications: Society and Environment*, 20:100414, 2020. doi: 10.1016/j.rsase.2020.100414

[30] E. G. A. Rolland. Volcán de Fuego - Computer Vision Plume Detection - YOLOv8 - Test Flight 2, Sept. 2023.

[31] E. G. A. Rolland. Volcán de Fuego - Computer Vision Summit and Plume Detections - YOLOv8 - Test Flight 1, Sept. 2023.

[32] E. G. A. Rolland, K. A. R. Grøntved, A. L. Christensen, M. Watson, and T. Richardson. YOLOv8-volcanic_plume-fuego_summit. Distributed by Hugging Face, 2024. doi: 10.57967/hf/1743.

[33] E. G. A. Rolland, M. Watson, T. Richardson, B. Schellenberg, M. Purvis, J. Freer, R. Clarke, M. Ferraro, M. Erbil, K. A. R. Grøntved, and A. L. Christensen. volcanic-plumes. Distributed by Hugging Face, 2024. doi: 10.57967/hf/1728.

[34] S. T. Sahetapy-Engel and A. J. Harris. Thermal-image-derived dynamics of vertical ash plumes at Santiaguito volcano, Guatemala. *Bulletin of Volcanology*, 71:827–830, 2009.

[35] B. Schellenberg, T. S. Richardson, A. Richards, and M. Watson. Automated real-time volcanic plume interception for UAVs. In *AIAA Scitech 2021 Forum*, p. 0811, 2021.

[36] B. J. Schellenberg. *Long range UAS operations for volcanic monitoring*. PhD thesis, University of Bristol, 2020.

[37] R. Simionato, P. A. Jarvis, E. Rossi, and C. Bonadonna. PlumeTraP: A new MATLAB-based algorithm to detect and parametrize volcanic plumes from visible-wavelength images. *Remote Sensing*, 14(7):1766, 2022.

[38] A. Stohl, A. J. Prata, S. Eckhardt, L. Clarisse, A. Durant, S. Henne, N. I. Kristiansen, A. Minikin, U. Schumann, P. Seibert, K. Stebel, H. E. Thomas, T. Thorsteinsson, K. Tørseth, and B. Weinzierl. Determination of time and height resolved volcanic ash emissions and their use for quantitative ash dispersion modelling: the 2010 Eyjafjallajökull eruption. *Atmospheric Chemistry and Physics*, 11(9):4333–4351, 2011. doi: 10.5194/acp-11-4333-2011

[39] D. K. Syahbana, K. Kasbani, G. Suantika, O. Prambada, A. S. Andreas, U. B. Saing, S. L. Kunrat, S. Andreastuti, M. Martanto, E. Kriswati, Y. Suparman, H. Humaida, S. Ogburn, P. J. Kelly, J. Wellik, H. M. N. Wright, J. D. Pesicek, R. Wessels, C. Kern, M. Lisowski, A. Diefenbach, M. Poland, F. Beauducel, J. Pallister, R. G. Vaughan, and J. B. Lowenstern. The 2017–19 activity at Mount Agung in Bali (Indonesia): Intense unrest, monitoring, crisis response, evacuation, and eruption. *Scientific Reports*, 9(1), 2019. doi: 10.1038/s41598-019-45295-9

[40] F. M. Talaat and H. ZainEldin. An improved fire detection approach based on YOLO-v8 for smart cities. *Neural Computing and Applications*, pp. 1–16, 2023.

[41] Ultralytics. Tips for best training results - Ultralytics YOLOv8 docs, 2024. Accessed: 2024-01-16.

[42] M. Watson. Spying on volcanoes. *Physics World*, 30(7):40, 2017.

[43] L. M. Western, M. I. Watson, and P. N. Francis. Uncertainty in two-channel infrared remote sensing retrievals of a well-characterised volcanic ash cloud. *Bulletin of Volcanology*, 77:1–12, 2015.

[44] K. Wood, E. J. Liu, T. Richardson, R. Clarke, J. Freer, A. Aiuppa, G. Giudice, M. Bitetto, K. Mulina, and I. Itikarai. BVLOS UAS operations in highly-turbulent volcanic plumes. *Frontiers in Robotics and AI*, 7:549716, 2020. doi: 10.3389/frobt.2020.549716

[45] M. Xu, S. Yoon, A. Fuentes, and D. S. Park. A comprehensive survey of image augmentation techniques for deep learning. *Pattern Recognition*, 137:109347, 2023. doi: 10.1016/j.patcog.2023.109347

[46] M. Yandouzi, M. Grari, B. Mohammed, I. Idrissi, O. Moussaoui, M. Azizi, K. Ghoumid, and K. E. Aissa. Investigation of combining deep learning object recognition with drones for forest fire detection and monitoring. *International Journal of Advanced Computer Science and Applications*, 14:377–384, 03 2023. doi: 10.14569/IJACSA.2023.0140342

[47] H. Yin, M. Chen, W. Fan, Y. Jin, S. G. Hassan, and S. Liu. Efficient smoke detection based on YOLO v5s. *Mathematics*, 10(19):3493, 2022.

[48] L. Zhao, L. Zhi, C. Zhao, and W. Zheng. Fire-YOLO: a small target object detection method for fire inspection. *Sustainability*, 14(9):4930, 2022.

[49] X. Zheng, F. Chen, L. Lou, P. Cheng, and Y. Huang. Real-time detection of full-scale forest fire smoke based on deep convolution neural network. *Remote Sensing*, 14(3):536, 2022.

[50] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren. Distance-IoU loss: Faster and better learning for bounding box regression. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(07):12993–13000, Apr. 2020. doi: 10.1609/aaai.v34i07.6999

[51] H. Zhu, H. Wei, B. Li, X. Yuan, and N. Kehtarnavaz. A review of video object detection: Datasets, metrics and methods. *Applied Sciences*, 10(21):7834, 2020.

[52] T. Çelik, H. Özkaramanli, and H. Demirel. Fire and smoke detection without sensors: Image processing based approach. *2007 15th European Signal Processing Conference*, pp. 1794–1798, 2007.