Unmanned Aerial Vehicle Mapping with Semantic and Traversability Metrics for Forest Fire Mitigation

David Russell[1*], Tito Arevalo[1,2*], Chinmay Garg[1*], Winnie Kuang[1*], Francisco Yandun[1*], David Wettergreen[1], George Kantor[1]

***Abstract*— Climate change and other factors are causing increasingly frequent and intense forest fires worldwide. Robotics systems can improve the feasibility of prevention and mitigation efforts. In this work, we propose an unmanned aerial vehicle that can map a forest region to allow an unmanned ground vehicle to autonomously clear fuel to prevent the spread of fire. We developed a multi-sensor payload consisting of cameras, LiDAR, and GPS with onboard processing. We also implement a SLAM system to understand the 3D structure of the environment, a semantics system to identify fuel and other features in the environment, and a traversability system that predicts which region a UGV can navigate. This approach provides a 3D map of the environment and geo-registered maps describing the locations of fuel and traversable regions. We validate our method with preliminary field trials and show that this is a promising approach.**

I. INTRODUCTION

In recent years, the number of wildfires has been above the 10-year average in the United States and Europe [1], [2]. According to NIFC [3], the main factors that increase fire susceptibility are hot and dry conditions that can turn vegetation into highly combustible material. Wildfire researchers have studied the mitigation of these factors by mapping potential zones that contain fuels that provide favorable conditions for fire propagation [4], [5]. The capability of satellite imagery to cover large regions have made remote sensing the most popular method to this aim [6]. However, both low spatial resolution and infrequent visitations preclude precise mapping.

Unmanned aerial vehicles (UAVs) have been proposed as an alternative for preventing, monitoring, and fighting wildfires. The ability to cover large areas and create higher resolution maps with data available on-demand makes UAVs especially suitable for these tasks. UAVs can also carry different types of sensors (e.g., visual, thermal, or spectral cameras, laser scanners) that provide more information for decision making and sometimes actuated hoses to fight fires [6]. Given the typically unstructured nature of forests, there are several challenges in processing all the sensor measurements to robustly map or actuate in the environment.

Considering the danger that wildfires represent for human firefighters, significant effort has been oriented toward the use of robots in fire fighting [7]. However, robotic forestry maintenance can play a key role in prevention tasks, especially when teaming aerial and ground vehicles. For example, effectively cleaning combustible vegetation can be a suitable way to prevent wildfires [8].

We developed an aerial sensing system based on visual imagery, inertial, and LiDAR data to identify fuel clusters and characterize the terrain morphology. In that way, we can create a multilayer map that a ground robot can use to remove hazardous material and navigate the environment. We propose the use of a SLAM system that tightly couples the state estimations from each sensor and fuse them in a factor graph scheme. The resulting odometry, along with a deep learning semantic segmentation approach, is used to create a map that localizes fuel clusters in the scene. We also evaluate the traversability of the terrain using the 3D point cloud generated by the SLAM system. Finally, we create a two-layer map that fuses both fuel localization and traversability information for a ground robot to navigate. The contributions of our work are twofold: first the thorough evaluation of each component of our system and second the complete system itself, which provides a novel perception system for wildfire prevention and forestry management.

This work is organized as follows: Section II summarizes previous related work. In Section III we describe our aerial vehicle and the sensing payload. Section IV describes the localization and mapping algorithm. The image processing methods to detect and localize fuel clusters are presented in Section V. Section VI introduces our early efforts to create the traversability layer of the map. The results of the tests for each part of the system are summarized in Section VII-A. Finally Section VIII describes the conclusions of our work.

II. RELATED WORK

Unmanned aerial vehicles (UAVs) have proven to be useful and effective in forestry applications as they provide a targeted approach for monitoring and conducting multi-temporal surveys for canopy closure monitoring, tree height estimation or vegetation classification [9], [10]. Specifically, for fire risk management and mitigation, drones have proven to be useful by reducing the human intervention in such dangerous scenarios [6], [11]. For example, a drone equipped with visual and thermal cameras was used in [12] to detect hotspots in a controlled wildfire. A swarm of drones has also been proposed as a decentralized and more efficient way to

*Authors contributed equally

[1]The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA {davidrus, chinmayg, wkuang, fyandun, dw0s, gkantor}@andrew.cmu.edu

[2]Department of Electronics, Universidad Técnica Federico Santa María, Valparaíso, Chile tito.arevalo@sansano.usm.cl

monitor fires, as in [13] and [14]. However, only simulated results are presented in both works.

While fighting wildfires has received much attention, using robots for management tasks to prevent or reduce the risk of uncontrolled wildfires is still at an early stage. Recently, Couceiro et. al. [8] described the concept of a muti-robot system where a UAV explores the environment and creates a map of the fuel clusters, and a ground robot removes them. Nevertheless, to the best of our knowledge, there is no implementation of such a system either in simulated or real environments. Our work fits in this scheme, by using the drone and the sensing payload to create a map of the environment with fuel cluster localization and terrain traversability information.

Simultaneous Localization and Mapping (SLAM) is necessary to enable UAVs and generally mobile robots to have a level of autonomy so that they could precisely capture fine-level details from the environment. Over the years, research has explored equipping UAVs with a sensor suite such that each sensor modality exploits the strengths of another sensor to compensate for its weaknesses (i.e., repeating structures, insufficient lighting, lack of visual features) [15], [16]. For instance, a LiDAR inertial odometry via smoothing and mapping (LIO-SAM) [17] expands the well-known LOAM [18] framework by including a loop closure method and feedback from absolute measurements, namely GPS and compass heading.

Including semantics in a SLAM pipeline is especially useful for our application visual information can be used to detect the fuel clusters and then locate them in a global reference frame. Precisely, semantic SLAM allows such a joint 3D representation with classification information. It has gained attention in recent years especially for applications like autonomous driving or assistive robotics [19], [20]. Unfortunately, many of these advances are not well-suited to being directly applied to the forestry domain, to the best of our knowledge. The different visual appearance of vegetation and other natural structures, along with artifacts in the images produced by the movement of the drone make the use of available state of the art methods like SuMa++ [21] and Kimera [22] especially challenging in our application.

Terrain characteristics form an integral part of the navigation strategy of an autonomous ground vehicle, and therefore its estimation and mapping have been an important research topic. Many works in this area have been used for different types of robotic systems where parameters such as slope, roughness, and step height are commonly employed [23], [24]. Based on the type of sensor information used and their characteristics, approaches based on proprioceptive data, the geometry of the scene, or its visual appearance have been used [25], [26].

III. AERIAL PLATFORM AND SENSING PAYLOAD

The aerial platform we used in this work is the commercial drone DJI M600. The drone was equipped with a sensing payload that consists of two 1.4 Megapixel cameras in a stereo configuration with a baseline of 0.25m, an inertial measurement unit (XMTi-30-AHRS Development Kit, XSens), a three-dimensional LiDAR (Velodyne VLP-16), and a consumer-grade computer with an Intel Core i7 (5Ghz) processor. To synchronize the sensor data acquisition, we used a custom-built board that generated pulse per second (PPS) signals mimicking a GPS receiver [15]. In this way, we ensure the time synchronization is maintained even in absence of a GPS signal. The telemetry and geo-localization data from the drone computer were also retrieved by the payload computer using serial communication. Figure 1 shows the CAD rendering of the sensing kit and its mounting on the drone.

(a) (b)

Fig. 1: DJI M600 with sensor payload: (a) isometric view of DJI M600 with sensor payload; (b) Exploded view of sensor payload and key components.

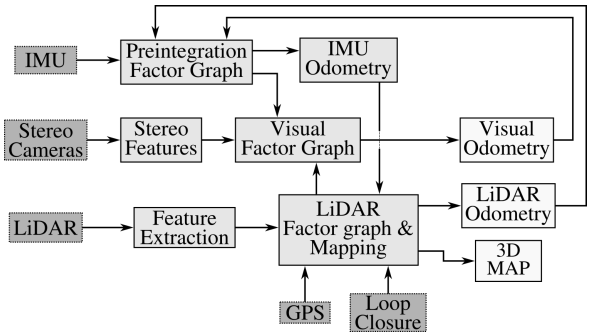


Fig. 2: General scheme of the multi-sensor SLAM system.

IV. SIMULTANEOUS LOCALIZATION AND MAPPING

We employed a multi-sensor system for simultaneous localization and mapping to detect the fuel clusters in the environment by obtaining its geometric and visual information. Our implementation is inspired by [15], [17], where the camera, LiDAR, inertial, and -optionally- GPS measurements are coupled in a factor graph scheme to obtain the 6 degrees of freedom pose of the platform and a three-dimensional map of the environment in the form of a point cloud. Thus, the estimated state of the robot is represented as a 6-dimensional vector corresponding to the position of the drone and its orientation in space. The SLAM system uses three tightly coupled factor graphs: inertial, visual, and LiDAR, as shown in Figure 2.

The inertial factor graph (Figure 3) uses the IMU mea-

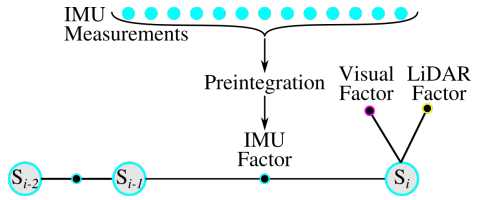


Fig. 3: IMU preintegration graph. Visual and LiDAR factors refer to the odometry data generated from these sensors. The S_i is the current optimized state (position and orientation).

surements and a preintegration process [27] to constrain the relative pose between states. When available, the latest visual or LiDAR odometry estimations are included as prior factors to obtain an optimized inertial state $\boldsymbol{S}$.

The second factor graph employs the approach described in [15]. It uses the images from the stereo pair, and obtains Oriented FAST and Rotated BRIEF [28] visual features that serve as the landmarks to estimate the constraints between states. The IMU preintegration is also included as another edge between states, while the LiDAR odometry is used as a prior factor, when available. Figure 4 depicts the overall graph, whose result is the optimized visual state $\boldsymbol{\zeta}$.

The LiDAR factor is created following the approach proposed in [18]. We first calculate the relative position and orientation between LiDAR scans using a scan to map registration algorithm. It is initialized using the odometry from the IMU $\boldsymbol{S}$, which improves the accuracy of the registration process. This results in the constraint edge between LiDAR states in the graph. Additionally, we included here loop closure and GPS factors, in a similar way to proposed in [17]. The first result of this graph is the optimized LiDAR state $\mathscr{S}$, which is used in the inertial and visual graphs. The second result comes from the scan to map registration, which provides a three-dimensional point cloud of the environment. Figure 5 shows a general scheme of the implemented LiDAR factor graph.

V. SEMANTIC SEGMENTATION FOR FUEL LOCALIZATION

The final objective of this work is to localize flammable regions which could be removed by the UGV. To this aim,

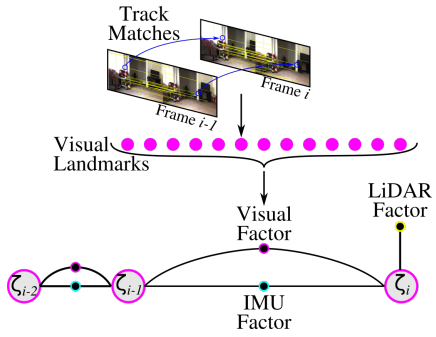


Fig. 4: Visual factor graph. LiDAR odometry is used as prior factor to constrain camera odometry.

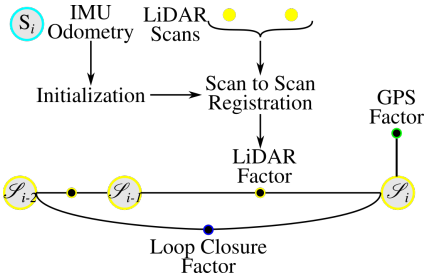


Fig. 5: LiDAR factor graph. A scan to scan algorithm computes raw LiDAR odometry using a initial estimation. The, lidar factor is completed with the GPS and loop closure restrictions.

we used a image-based semantic segmentation deep network to predict features of the scene which cannot be determined from the SLAM system's 3D point cloud. The main classes we segment are: trunks, canopies, background and fuel. Fuel is specially important and consists of any vegetation that is at ground level so a ground vehicle can remove it. Trunks and canopy will be employed in future work for improving our SLAM pipeline. Background basically contains miscellaneous objects like dirt or the sky.

A. Data

We explored several sources of data with imagery of forest scenes and chose two datasets that best matched our application domain. The first dataset used a graphics engine to render photorealistic viewpoints from a procedurally-generated forest scene [29]. Using a different texture, the same scene was used to render ground-truth classification images representing the class of the object at each pixel in the scene. From this dataset, which we will call the *synthetic* dataset, we used 3154 images with a consistent resolution of 848 x 480. The second dataset contains 151 real images from a forest located in the Setes Fontes region and described in [30]. These images were manually labeled with precise polygons and had a resolution of 1280 x 720. We will refer to this dataset as the *Setes Fontes* dataset.

B. Semantic SLAM

The goal of semantic SLAM is to fuse the semantic observations from different observations into a global map that captures both the shape and semantics. To accomplish this task, we used a loosely coupled approach to fuse the semantic segmentation (at image level), the LiDAR readings, and the drone odometry given by the SLAM system. We call it a loose coupling as there is no feedback from the semantic mapping into the SLAM pipeline.

In the image domain, we use a segmentation network based on a transformer architecture called SegFormer [31]. Given the relatively low amount of real-world images in our training dataset, this network was especially suitable since it showed strong performance on benchmark datasets and good generalization capabilities. We trained this model using the default parameters used in the MMSegmentation [32] implementation.

Subsequently, we modified an approach for RGB-D semantic SLAM [33] to project the detections from the image to the LiDAR domain (i.e., three-dimensional). First, the image is passed through the semantic segmentation network to get a classification result for each pixel. Using the extrinsics of the LiDAR relative to the camera, we transformed the LiDAR measurements into the camera's coordinate frame. Then, using the calibrated camera intrinsic, we project each LiDAR point into the image plane. Points within the field of view of the camera are assigned a classification label from the corresponding pixel in the semantic map. This semantically-textured point cloud is transformed into the inertial reference frame using the current pose of the drone estimated by our SLAM system.

We use an octomap [34] representation to efficiently discretize the generated semantic point cloud into voxels. Each voxel has a resolution of 0.05m and contains information about the predicted classification. Each time a new semantic point cloud is created, it is used to update this global octomap. Since each voxel can contain multiple observations, we use two approaches to determine the aggregate classification. The first method assigns the class label using the highest-confidence prediction from the neural network that corresponds to that voxel. Alternatively, we use a Bayesian method which maintains a probability distribution over the classes. Each new observation is multiplied by the current distribution and then re-normalized. The voxel is then labeled with the most probable class.

VI. TERRAIN TRAVERSABILITY EVALUATION

Given that we have a 3D point cloud of the environment, we also evaluated the traversability of the terrain to give a ground robot a prior map for global planning and navigation. In this way, we can use the versatility of the drone to cover larger regions and get views of scenes that can be obstructed at ground level. The traversability and semantic maps are then fused forming a multi-layer map for the ground vehicle's operation.

As mentioned before, we used the 3D point cloud generated by the SLAM system as input. As the first step, the Cloth Simulation Filtering algorithm [35] was used to separate the cloud points into ground and non-ground points. This separation allowed us to evaluate the traversability of terrain patches based on the presence of non-ground points in a particular area and further use the data to get the terrain slope and roughness.

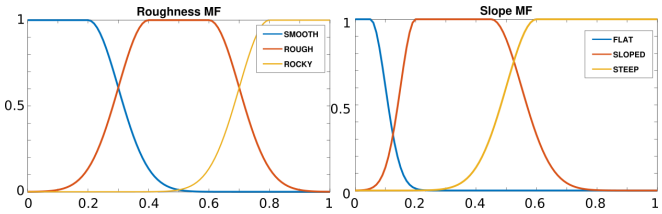
A. Heuristics-based Approach

The heuristics-based approach fundamentally uses the ratio of non-ground and ground points to determine the traversability of a region. First, the entire point cloud is divided in the XY plane into a grid of 1m resolution in each direction. This value was chosen to achieve a trade-off between computational efficiency and spatial resolution. Within each grid block, we then calculated the ratio between the number of non-ground and ground points. If it exceeded a threshold of 0.3, the grid block was assigned a traversability index of zero (i.e., non-traversable). Otherwise, the index was set to one (i.e., traversable). This threshold was selected by trial and error in different experiments and was found to be highly dependent on the scene.

This approach is computationally costly since it requires the division of the point cloud into a grid and calculations for each block, and as the map increases and the point cloud data updates, this becomes highly inefficient. The complexity depends on the resolution of the map, with higher resolutions growing exponentially in complexity. Moreover, this makes use of a simple heuristic that is not resilient to noise and imperfections in the point cloud data.

B. DEM-based Approach

In this approach, the point cloud data is translated into a Digital Elevation Map (DEM) representation. Using the topotoolbox from MATLAB [36], we calculated the roughness and slope of the terrain. The more efficient implementation of this toolbox allowed us to divide the point cloud in a grid of 0.3m resolution. Using these two indicators, we implemented a fuzzy logic system inspired by [25] [37] to obtain a single traversability index per terrain patch that ranges from 0 to 1 (low to high traversability).



(a) (b)

Fig. 6: Input Membership Function Plots (a) Roughness and (b) Slope.

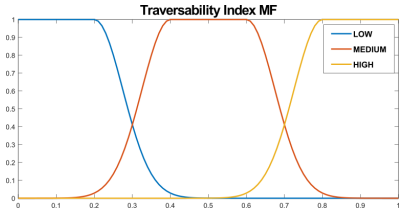


Fig. 7: Output Membership Function Plots.

Using the DEM data and the ground point cloud data, we calculate a traversability index on a scale of 0 to 1, indicating low to high traversability respectively. Using fuzzy logic framework on the terrain data with inspirations from [25] [37], a fuzzy traversability index is calculated. The fuzzy traversability index quantifies the suitable terrain for the ground rover to operate on using the roughness and slope information of the terrain. The terrain roughness and slope are inputs represented by the linguistic fuzzy sets, {SMOOTH, ROUGH, ROCKY} and {FLAT, SLOPED, STEEP} respectively. The output is represented by the linguistic fuzzy set {HIGH, MEDIUM, LOW} and a numeric score is computed

from this set using the centroid method. The final cost is the complement of the fuzzy system output, calculated as $(1 - traversability)$. All the fuzzy sets are defined by the Gaussian Combination fuzzy membership function plots shown in figures 6 and 7. The following rules in the fuzzy inference system govern the final output traversability index :

1) If (Slope is FLAT) and (Roughness is SMOOTH) then (Index is HIGH)
2) If (Slope is STEEP) or (Roughness is ROCKY) then (Index is LOW)
3) If (Slope is FLAT) and (Roughness is ROUGH) then (Index is MEDIUM)
4) If (Slope is SLOPED) and (Roughness is SMOOTH) then (Index is MEDIUM)
5) If (Slope is SLOPED) and (Roughness is ROUGH) then (Index is MEDIUM)

VII. RESULTS

To test the full system we used data collected with our drone and sensing payload in a forestry region located at $41°13'00.9"$N $8°31'38.6"$W in Oporto, Portugal. All flights lasted around five minutes and were performed under the canopy with the sensing platform facing the drone's left side at an inclination of 62 degrees (pitch). The sensor data was captured and processed using the robot operating system (ROS), except for the traversability evaluation that was performed offline. The unlabeled data produced in these field trials is referred as the *Oporto* dataset.

We first evaluated if the odometry computed by the SLAM system was suitable to obtain reliable pose estimations. This part is especially important as the semantics and traversability evaluation rely on the drone positioning and the map generated. Thus, we compute the accumulative root mean squared error (RMSE) during the complete trajectory. The latitude and longitude GPS coordinates were transformed to the x, y, z UTM reference system and used as ground truth. Even though it is a suboptimal choice, a real-time kinematic (RTK) was not available at the time of the experiments.

As Table I shows, the LiDAR odometry has a lower RMSE compared to visual odometry in both aerial surveys. This is likely because the drone produced blurred images, which together with the presence of close or textureless objects (e.g., the sky on the horizon) made the computation of robust visual features highly prone to errors. In these scenarios, the tightly coupled approach we used proved to be useful as the LiDAR helped to keep the errors bounded.

TABLE I: Evaluation of the visual and lidar localization from the SLAM system.

	RMSE			
Coordinates	Survey 1		Survey 2	
	Visual	LiDAR	Visual	LiDAR
x	0.5630	0.3226	0.3136	0.3236
y	1.6503	0.3229	0.5461	0.5250
z	0.2957	0.2148	0.2293	0.1817
mean	0.8363	0.2867	0.3630	0.3434

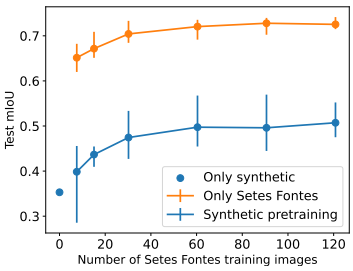


Fig. 8: Test mIoU for very few training images on the *Setes Fontes* dataset. Error bars represent minimum and maximum result across the five folds of *Setes Fontes*.

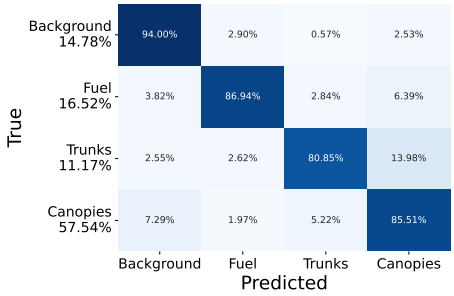


Fig. 9: Confusion matrix for the *Setes Fontes* test datasets normalized per class with the true fraction of each class reported on the y axis labels.

The LiDAR odometry showed a better performance mainly to the presence of the GPS and loop closure factors.

Considering the scale of the intended operation (the test area spans approximately $4684m^2$) we considered these errors acceptable to use the SLAM system for the semantic and traversability mapping in the early stage of the drone and ground robot integration.

A. Semantic Mapping

Given the limited availability of real data and the labor-intensive nature of labeling to obtain ground truth, we explored the utility of models trained with simulated (*synthetic*) data. We conducted three types of experiments: models trained solely with *synthetic* data, models trained with real data (*Setes Fontes*), and a mixture of both. For the last two cases, we trained with an increasing number of real images to evaluate the performance of the model with minimal real images. Thus, we conducted training experiments using (or adding) 7, 15, 21, 30, 60, 91, and 121 data points from the *Setes Fontes* dataset. We trained for 10000 iterations and evaluated each model in 30 *Setes Fontes* images not seen in the largest training split are used for evaluation and we replicate this experiment over five folds of the data. The three models are a fine-tuned implementation as the base networks were first trained with the *CityScapes* dataset [38].

We conducted experiments to determine how many real training images were needed, with and without synthetic pretraining, as seen in Figure 8. We used mean Intersection over Union (mIoU) to evaluate the quality of the predictions

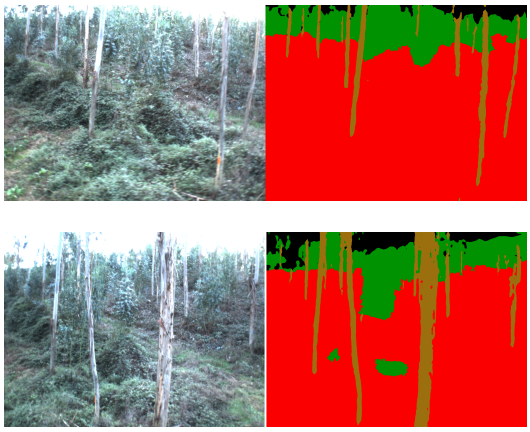


Fig. 10: Predictions on the *Oporto* dataset. Black is background, red is fuel, brown is trunks, and green is canopy.

on the test set. It is interesting to note the relatively high performance of a model that used only 7 real images. Also, the model trained solely on synthetic data fails to generalize to real data, even after properly accounting for differences in mean and variance of both datasets. We found that combined real and synthetic data performs worse than training the same model only using real data. This suggests that the synthetic data comes from a completely different distribution than the real one, making its contribution detrimental. In the future, we plan to keep researching the causes of this interesting outcome.

As the model trained in 121 images (80% of the *Setes Fontes* dataset) showed the best performance and was used for deployment in the *Oporto* dataset, which was never seen during training. The confusion matrix on the *Setes Fontes* test set can be found in Figure 9 and qualitative results are in Figure 10. As we are mainly interested in the fuel instances, we aggregated the background, trunks, and canopies in a single non-fuel class. In that case we obtained an IoU of 78.2% and 95.3% for `Fuel` and `Not Fuel`, respectively, which yields a mIoU of 86.7%. This shows that our system performs well at its primary task of identifying fuel.

Finally, we tested the whole semantic SLAM system on the *Oporto* dataset. The results can be seen in Figure 11. The system was able to distinguish the region of fuel on the ground. Accurate calibration and pose estimation allowed our system to correctly predict the class of the small trunks.

B. Traversability

The heuristic and DEM-based approaches were used to build the occupancy grids shown in Figures 12 a and b, respectively. With the default settings, the heuristic grid showed a more permissive layout, that better resembled the real traversability conditions observed in the field experiments. Conversely, the fuzzy approach is quite conservative, leading to fewer traversable zones (reducing the viability of obtaining a proper path). However, we believe the fuzzy system has better potential as it is faster and provides a better map resolution. Furthermore, we can tune the membership

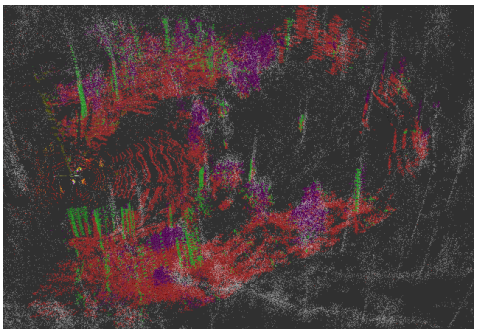
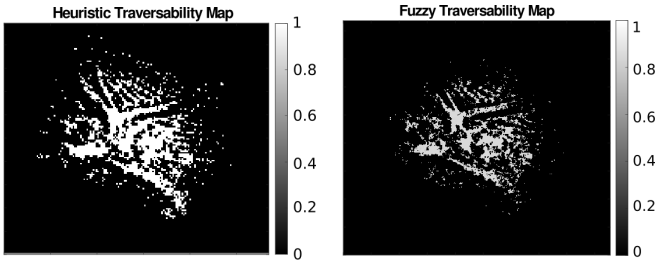


Fig. 11: Semantic SLAM results. Fuel is red, trunks are green, canopy is purple, and background is black. White points are unlabeled.



(a) (b)

Fig. 12: Traversability Map Results from (a) Heuristics-based Approach and (b) DEM-based Approach.

functions to reduce the sparsity of the traversable regions. A traversability map with reduced sparsity and a more generalized traversability estimate can be achieved with a deep learning-based approach, but it evidently needs a lot more forest data.

VIII. CONCLUSIONS

We presented an aerial system that, equipped with visual, inertial, and LiDAR sensors, allowed us to identify fuel zones in a forest environment and evaluate the traversability of the terrain. A thorough evaluation of the SLAM, semantic, and traversability subsystems were also presented. Special attention was taken to evaluate the sim to real gap and find insights about the use of a combination of simulated data for specific applications like segmenting fuel clusters in forestry. The overall system produced a geo-referenced two-layer map containing fuel localization and an occupancy grid that a ground robot can use to find the objectives to clear and navigate in the environment.

ACKNOWLEDGMENT

We would like to thank Rui Nunes, Maria Eduarda Andrada, and Professors Paulo Peixoto and João Filipe Ferreira for providing some of the data that was used in this work. This project was supported by SafeForest under the reference CENTRO-01-0247-FEDER-045931 and Agencia Nacional de Investigación y Desarrollo (ANID) /PFCHA/DOCTORADO NACIONAL CHILE/2019-21190471. Views expressed are those of the authors.

REFERENCES

[1] N. I. F. Center, "National fire news," National Interagency Fire Center, 2022, https://www.nifc.gov/fire-information/nfn(Accessed: March 2022). [Online]. Available: https://www.nifc.gov/fire-information/nfn

[2] J. San-Miguel-Ayanz, T. Durrant, R. Boca, P. Maianti, G. Libertá, T. Artés-Vivancos, D. Oom, A. Branco, D. de Rigo, D. Ferrari, H. Pfeiffer, R. Grecchi, D. Nuijten, and M. Onida, "Advance effis report on forest fires in europe, middle east and north africa 2020," Joint Research Centre, 2021. [Online]. Available: https://effis-gwis-cms.s3-eu-west-1.amazonaws.com/effis/reports-and-publications/effis-related-publications/Advance_EFFIS_Report+on+Forest+Fires+in+Europe_2020_210401xv_finalv2.pdf

[3] NIFC, "National interagency fire center," https://www.nifc.gov/fire-information/nfn, 2022, accessed: 2022-03-22.

[4] H. Mahmoud and A. Chulahwat, "Unraveling the complexity of wildland urban interface fires," *Scientific Reports*, vol. 8, no. 1, pp. 1–12, 2018.

[5] A. C. Fernandez-Pello, "Wildland fire spot ignition by sparks and firebrands," *Fire Safety Journal*, vol. 91, pp. 2–10, 2017.

[6] C. Yuan, Y. Zhang, and Z. Liu, "A survey on technologies for automatic forest fire monitoring, detection, and fighting using unmanned aerial vehicles and remote sensing techniques," *Canadian journal of forest research*, vol. 45, no. 7, pp. 783–792, 2015.

[7] J. J. Roldán-Gómez, E. González-Gironda, and A. Barrientos, "A survey on robotic technologies for forest firefighting: Applying drone swarms to improve firefighters' efficiency and safety," *Applied Sciences*, vol. 11, no. 1, p. 363, 2021.

[8] M. S. Couceiro, D. Portugal, J. F. Ferreira, and R. P. Rocha, "Semfire: Towards a new generation of forestry maintenance multi-robot systems," in *2019 IEEE/SICE International Symposium on System Integration (SII)*. IEEE, 2019, pp. 270–276.

[9] O. Nevalainen, E. Honkavaara, S. Tuominen, N. Viljanen, T. Hakala, X. Yu, J. Hyyppä, H. Saari, I. Pölönen, N. N. Imai, and A. M. G. Tommaselli, "Individual tree detection and classification with uav-based photogrammetric point clouds and hyperspectral imaging," *Remote Sensing*, vol. 9, no. 3, 2017. [Online]. Available: https://www.mdpi.com/2072-4292/9/3/185

[10] T. Sankey, J. Donager, J. McVay, and J. B. Sankey, "Uav lidar and hyperspectral fusion for forest monitoring in the southwestern usa," *Remote Sensing of Environment*, vol. 195, pp. 30–43, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425717301578

[11] C. Viegas, B. Chehreh, J. Andrade, and J. Lourenço, "Tethered uav with combined multi-rotor and water jet propulsion for forest fire fighting," *Journal of Intelligent & Robotic Systems*, vol. 104, no. 2, pp. 1–13, 2022.

[12] A. Viseras, J. Marchal, M. Schaab, J. Pages, and L. Estivill, "Wildfire monitoring and hotspots detection with aerial robots: Measurement campaign and first results," in *2019 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2019, pp. 102–103.

[13] A. Viseras, M. Meissner, and J. Marchal, "Wildfire front monitoring with multiple uavs using deep q-learning," *IEEE Access*, 2021.

[14] M. S. Innocente and P. Grasso, "Self-organising swarms of firefighting drones: Harnessing the power of collective intelligence in decentralised multi-robot systems," *Journal of Computational Science*, vol. 34, pp. 80–101, 2019.

[15] W. Shao, S. Vijayarangan, C. Li, and G. Kantor, "Stereo visual inertial lidar simultaneous localization and mapping," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 370–377.

[16] S. Zhao, H. Zhang, P. Wang, L. Nogueira, and S. Scherer, "Super odometry: Imu-centric lidar-visual-inertial estimator for challenging environments," *arXiv.org*, Aug 2021. [Online]. Available: https://arxiv.org/abs/2104.14938

[17] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and R. Daniela, "Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 5135–5142.

[18] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time." in *Robotics: Science and Systems*, vol. 2, no. 9. Berkeley, CA, 2014, pp. 1–9.

[19] I. Kostavelis and A. Gasteratos, "Semantic mapping for mobile robotics tasks: A survey," *Robotics and Autonomous Systems*, vol. 66, pp. 86–103, 2015.

[20] M. Siam, M. Gamal, M. Abdel-Razek, S. Yogamani, M. Jagersand, and H. Zhang, "A comparative study of real-time semantic segmentation for autonomous driving," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 587–597.

[21] X. Chen, A. Milioto, E. Palazzolo, P. Giguere, J. Behley, and C. Stachniss, "SuMa++: Efficient LiDAR-based Semantic SLAM," *IEEE International Conference on Intelligent Robots and Systems*, pp. 4530–4537, 2019.

[22] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, "Kimera: An Open-Source Library for Real-Time Metric-Semantic Localization and Mapping," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 1689–1696, 2020.

[23] S. Goldberg, M. Maimone, and L. Matthies, "Stereo vision and rover navigation software for planetary exploration," in *Proceedings, IEEE Aerospace Conference*, vol. 5, 2002, pp. 5–5.

[24] P. Papadakis, "Terrain traversability analysis methods for unmanned ground vehicles: A survey," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 4, pp. 1373–1385, 2013.

[25] A. Howard and H. Seraji, "Real-time assessment of terrain traversability for autonomous rover navigation," in *Proceedings. 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000) (Cat. No.00CH37113)*, vol. 1, 2000, pp. 58–63 vol.1.

[26] M. Visca, S. Kuutti, R. Powell, Y. Gao, and S. Fallah, "Deep learning traversability estimator for mobile robots in unstructured environments," *CoRR*, vol. abs/2105.10937, 2021. [Online]. Available: https://arxiv.org/abs/2105.10937

[27] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "IMU preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation," *Robotics: Science and Systems*, vol. 11, 2015.

[28] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International Conference on Computer Vision*, 2011, pp. 2564–2571.

[29] R. J. S. O. Nunes, "Procedural generation of synthetic forest environments to train machine learning algorithms," Ph.D. dissertation, Universidade de Coimbra, 2021.

[30] M. E. Andrada, J. F. Ferreira, D. Portugal, and M. Couceiro, "Testing Different CNN Architectures for Semantic Segmentation for Landscaping with Forestry Robotics," *In: IROS 2020 Workshop on Perception, Planning and Mobility in Forestry Robotics (WPPMFR 2020)*, 2020.

[31] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers," pp. 1–18, 2021. [Online]. Available: http://arxiv.org/abs/2105.15203

[32] Open MMLab, "Mmsegmentation." [Online]. Available: https://github.com/open-mmlab/mmsegmentation

[33] Z. Xuan and F. David, "Real-time voxel based 3d semantic mapping with a hand held rgb-d camera," https://github.com/floatlazer/semantic_slam, 2018.

[34] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: An efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, 2013, software available at https://octomap.github.io. [Online]. Available: https://octomap.github.io

[35] W. Zhang, J. Qi, P. Wan, H. Wang, D. Xie, X. Wang, and G. Yan, "An easy-to-use airborne lidar data filtering method based on cloth simulation," *Remote Sensing*, vol. 8, no. 6, p. 501, 2016. [Online]. Available: http://www.mdpi.com/2072-4292/8/6/501

[36] W. Schwanghart and D. Scherler, "TopoToolbox 2 – MATLAB-based software for topographic analysis and modeling in Earth surface sciences," *Earth Surface Dynamics*, vol. 2, pp. 1–7, 2014.

[37] A. Howard, H. Seraji, and E. Tunstel, "A rule-based fuzzy traversability index for mobile robot navigation," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, vol. 3, 2001, pp. 3067–3071 vol.3.

[38] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The Cityscapes Dataset for Semantic Urban Scene Understanding," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-Decem, pp. 3213–3223, 2016.