

Autonomous Drone Navigation in Forest Environments Using Deep Learning

Guglielmo Del Col¹, Väinö Karjalainen¹, Teemu Hakala¹, Eija Honkavaara¹

¹Finnish Geospatial Research Institute in National Land Survey of Finland, Vuorimiehentie 5, 02150 Espoo, Finland
- (guglielmo.delcol, vaino.karjalainen, teemu.hakala, eija.honkavaara)@nls.fi

Keywords: Autonomous navigation, deep learning, UAV, forest environments, obstacle avoidance, trajectory planning

Abstract

Autonomous drone navigation in dense forests remains challenging due to unreliable GNSS signals, difficulty detecting thin branches, and cumulative drift in Visual-Inertial Odometry (VIO). This work investigates a deep learning-based navigation solution using a simulation-to-reality approach, focusing on boreal forests where fine obstacles and dense foliage are prevalent. A vision-based system was deployed, combining a semantically-enhanced depth autoencoder for small-branch detection and a Collision Prediction Network (CPN) based on motion primitive evaluation. The system, trained using RotorS and Aerial Gym simulations data, was implemented on a custom drone featuring a RealSense D435i and a RealSense T265 sensor suite and NVIDIA Orin NX for onboard processing. Real-world tests in open, lightly vegetated, and dense forests revealed robust performance against larger obstacles but highlighted limitations in thin-branch avoidance and odometry drift in highly cluttered environments. While simulation results were satisfactory, real-world trials achieved moderate success (60 m flights), demonstrating the potential of the framework for forestry applications. As future directions, integrating higher-resolution sensors, RGB-depth fusion, a y-velocity integration and possibly a small lidar, to address current gaps are proposed. The findings underscore the need for real-world validation beyond simulation to bridge the perception-action gap in complex environments.

1. Introduction

Autonomous drone navigation in cluttered environments, particularly under forest canopies, presents a complex set of challenges. Dense foliage, thin obstacles such as branches, and inconsistent lighting severely limit the effectiveness of traditional navigation pipelines. Moreover, the absence of GNSS signals in such environments further complicates localization and control, pushing the need for vision-based solutions that can interpret the scene in real time and make safe, robust decisions.

Conventional approaches often rely on Visual-Inertial Odometry (VIO), Simultaneous Localization and Mapping (SLAM), or dense 3D mapping using data structures such as octrees or voxel grids (Hornung et al., 2013; Oleynikova et al., 2017). However, these techniques are prone to failure in forests due to dynamic lighting, occlusions, and the high computational demands of dense mapping (Ebadi et al., 2022). LiDAR-based solutions, though accurate, are often unsuitable for small UAVs due to their size, weight, and power constraints (Zhang et al., 2024).

To address these limitations, recent work has increasingly shifted toward map-less, learning-based control methods. Instead of building an explicit world model, these methods learn direct mappings from sensor inputs to control commands or collision probabilities using deep neural networks. Examples include supervised learning approaches trained on pilot demonstrations or crash data (Gandhi et al., 2017; Loquercio et al., 2018), and reinforcement learning techniques that generalize across domains through simulation-to-reality transfer (Sadeghi and Levine, 2016). These techniques enable more agile and responsive navigation in cluttered environments by avoiding the bottlenecks of traditional mapping.

However, many of these models simplify critical factors such as field-of-view limitations, aleatoric uncertainty, and the dynamic state of the robot (Pfeiffer et al., 2018; Nguyen et al.,

2022). As a result, they may perform poorly when deployed in real-world settings with unpredictable visual degradations and complex obstacle geometries.

Two recent contributions have sought to overcome these limitations. ORACLE (Nguyen et al., 2024) combines deep ensembles and uncertainty modeling to predict collision probabilities with obstacles. Meanwhile, seVAE (Kulkarni et al., 2023b) introduces a semantically-aware depth autoencoder that enhances feature representation for fine-grained structures such as branches.

This paper builds upon both ORACLE and seVAE, integrating their key innovations into a single end-to-end navigation framework tailored for forest environments. In contrast to prior work evaluated in structured or semi-natural outdoor spaces, the system developed in this work is tested extensively in complex real-world forests in Finland, characterized by dense vegetation, uneven terrain, and variable lighting.

The main contributions of this work are:

- Designing and implementing a complete end-to-end autonomous navigation system that combines semantically-enhanced depth encoding with uncertainty-aware collision prediction.
- Refining model architecture and training strategies to improve obstacle awareness and navigational safety in dense forest scenes.
- Conducting comprehensive validation in both simulated and real-world forest environments, demonstrating reliable autonomous flight in GNSS-denied, cluttered conditions.

This article is based on the Master's thesis of the first author Del Col Guglielmo (2024).

2. Material and Methods

A comprehensive representation of the workflow of this project and the main algorithm pipeline for the drone navigation, are provided in Fig 2. The following sections will describe in detail the hardware and software components of this project.

2.1 System Overview

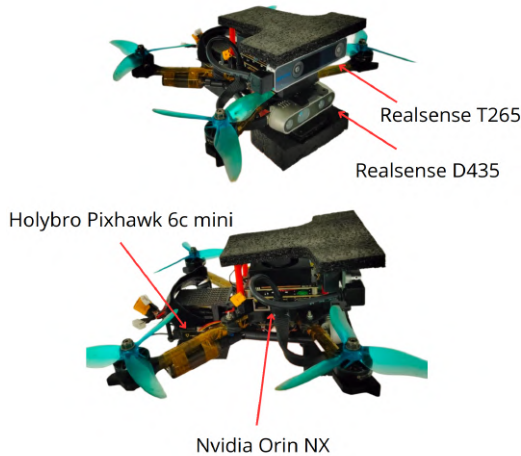


Figure 1. System Components

The proposed system in Figure 1 comprises three tightly integrated components: a perception module built around a convolutional autoencoder (Kulkarni et al., 2023b), a CPN for trajectory evaluation (Nguyen et al., 2024), and a real-time control interface. All modules communicate via Robot Operating System (ROS) and operate synchronously at 30 Hz. The Perception module includes a RealSense D435i stereo camera that captures depth images at 270×480 resolution, filtered to discard invalid pixels. These depth maps are passed through a 7-layer convolutional autoencoder, producing a 128-dimensional latent representation that emphasizes semantically critical features such as tree trunks and branches. The autoencoder is presented in Figure 2.

The Planning module is mainly composed by the collision prediction network, visible in Figure 2. At each planning step, the system evaluates 256 motion primitives, defined by combinations of x-velocity, z-velocity, and yaw rate. These candidate trajectories are scored by the CPN, which uses the latent vector and partial drone state to predict the likelihood of collision for each option. The safest primitive is selected for execution. The chosen velocity command is transmitted to the PX4 compliant flight controller of the drone, a Holybro Pixhawk 6C Mini, using MAVROS. Safety checks ensure that if all predicted trajectories pose excessive risk, an emergency stop is triggered. The controller maintains flight stability even during rapid trajectory changes or evasive maneuvers.

2.2 Semantically Enhanced Autoencoder

To enable compact and semantically meaningful representations of visual data, the system leverages a convolutional autoencoder architecture derived from the work of Kulkarni et al. (Kulkarni et al., 2023b). The primary function of the autoencoder is to compress depth images into a latent space that encodes information critical for safe path planning, especially

the presence of thin branches and small obstacles often missed by conventional depth-based features. The encoder comprises seven convolutional layers with progressively decreasing spatial resolution and increasing feature depth. These are followed by a bottleneck layer with 128 latent dimensions. The decoder, used only in training and not during inference, mirrors this structure using seven deconvolutional layers to reconstruct the original input. This symmetric architecture balances information preservation with computational efficiency. Training of the autoencoder was performed to fine-tune the model in forest-specific scenes generated using the Aerial Gym simulator (Kulkarni et al., 2023a). This simulator supports randomized obstacle placements resembling trees and branches, allowing the autoencoder to learn semantically salient features. The training objective combined a pixel-wise mean squared error (MSE) loss with a Kullback-Leibler Divergence (KLD) term, as shown in Equation 1. Semantic segmentation masks were used to weight the MSE loss, giving higher importance to visually sparse but critical classes such as branches. Invalid pixels were masked during training to prevent their corruption of the latent space.

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{MSE}}^{\text{sem}} + \lambda \cdot \mathcal{L}_{\text{KLD}} \quad (1)$$

where $\mathcal{L}_{\text{MSE}}^{\text{sem}}$ is the mean squared error weighted by semantic importance and $\mathcal{L}_{\text{KLD}}$ regularizes the latent space via a Kullback-Leibler divergence term. This combination helps preserve small yet crucial features in the latent representation.

The use of forest-specific data significantly improved the ability of the model to encode fine structural details. Visualizations of latent reconstructions showed clearer representation of branches and trunks, particularly in partially occluded views or under low-contrast conditions.

2.3 Collision Prediction Network

The decision-making core of the navigation system is a Collision Prediction Network (CPN), originally developed by Nguyen et al. (Nguyen et al., 2022, 2024), and retained here with minor adjustments. The CPN estimates the probability of collision for a set of predefined motion primitives, using both the latent encoding of the current depth image and the partial state of the drone. Each motion primitive defines a trajectory over a fixed time horizon and is characterized by a combination of three components: constant x-axis forward velocity, one of eight discrete z-axis velocities (vertical motion), and one of 32 yaw rates (angular turns). In total, 256 candidate trajectories are evaluated at each decision step. The architecture of the CPN includes a Long-Short Term Memory network (LSTM) layer that processes the trajectory sequence, followed by a set of fully connected layers outputting a sigmoid-scaled probability of collision for each trajectory. Ensemble learning is employed to reduce model variance. Three instances of the CPN are trained with different random seeds and weight initializations. Their predictions are averaged during inference to produce a more stable and robust estimate. To propagate sensor and model uncertainty, the Unscented Transform (UT) is used. UT generates sigma points around the current state estimate, capturing non-linear transformations in the prediction process more accurately than linearization techniques. This probabilistic propagation improves collision forecasting in ambiguous or noisy regions of the observation space. The overall decision-making frequency was maintained at 30 Hz, matching the perception pipeline, and allowing the system to respond in real time to dynamic changes or near-miss events.

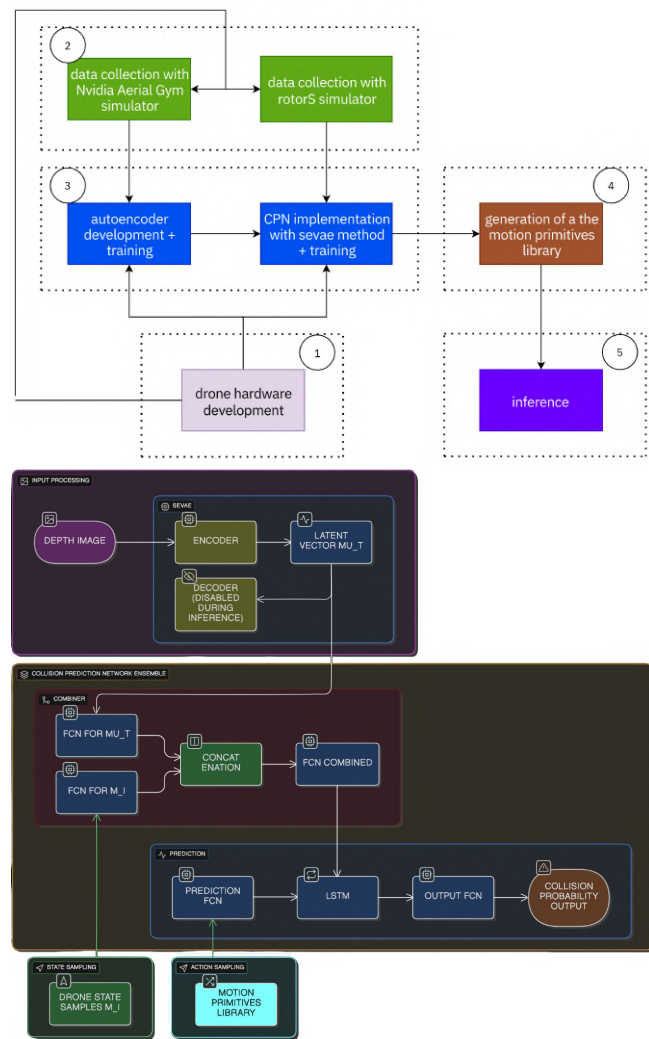


Figure 2. (Top) Work pipeline of the system. (Bottom) Navigation framework and planner integration (sevae-ORACLE model).

2.4 Experimental Setup

To evaluate the performance of the proposed navigation system, extensive experiments were conducted in both simulation and real-world forest environments. The aim was to assess the ability of the drone to autonomously reach a target destination while avoiding obstacles, particularly in complex and cluttered settings.

Env. ID	Type	Density	Description	Purpose
Sim-1	Dense Forest (sim)	High	Flat terrain with few scattered trunks	Baseline evaluation
Sim-2	Dense Forest (sim)	High	Branch-dense scene with randomized obstacle layout	Autoencoder and CPN stress test
RW-1	Open Field	Low	Grass clearing with distant trees	Real-world baseline test
RW-2	Sparse Forest	Medium	Mixed hardwood area with moderate vegetation	Basic obstacle avoidance
RW-3	Dense Spruce A	High	Tight conifer spacing, many thin low branches	Complex navigation and recovery test

Table 1. Environmental descriptions of the test environments and corresponding purposes.

2.4.1 Simulated Experiments Simulation experiments were carried out using the RotorS simulator (Gao et al., 2019) with Gazebo (Koenig and Howard, 2004) as the physics engine.

The environments were designed to emulate a variety of forest scenarios, including open clearings, sparse mixed woods, and dense spruce forests, with varying obstacle densities and structural complexity.

The drone was initialized at a fixed starting location and commanded to reach a goal approximately 30 meters away while navigating through synthetic trees and branches. These scenarios were constructed to evaluate basic performance in cluttered navigation tasks. Collision outcomes, trajectory stability, and mission success rate were measured across multiple flights.

Furthermore, these experiments were designed to allow a direct comparison with the simulated experiments in the previous work presented by Karjalainen et al. (2023), which highlighted limitations in detecting small, thin branches common in Finnish forests. To assess improvements, a set of twelve simulated forest environments, originally developed for the work of Karjalainen et al. (2023) were reused. These environments follow three difficulty levels ("easy", "medium", and "hard") categorized by the tree density. Specifically, three densities of 0.1, 0.15, and 0.2 trees per square meter were used, with four different forest layouts per density. High-definition tree models containing dense fine-branch structures (Globe Plants Team, 2022) were employed to test the perception limitations observed

in previous methods.

In total, two experiments were conducted with 180 simulated flights each, allowing for robust statistical comparison of navigational performance, obstacle avoidance, and generalization across varying environmental complexities.

In the first experiment the baseline presented in the ORACLE work (Nguyen et al., 2024) was tested, while the second experiment included tests with the sevae-ORACLE planner (Kulkarni et al., 2023b) and all the refinements explained in Figure 2.

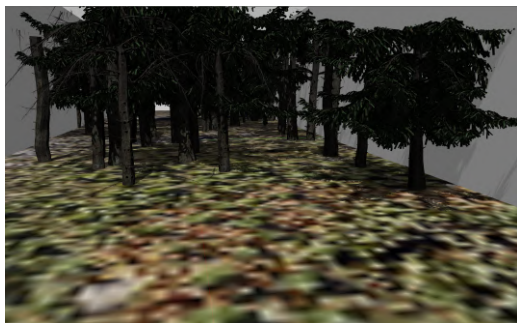


Figure 3. rotorS forest

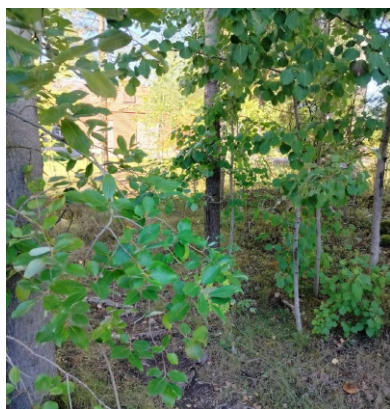


Figure 4. Example of the vegetation present in RW-2



Figure 5. Example of the vegetation present in RW-3

2.4.2 Real-World Experiments Three main experiments were conducted in the forests of southern Finland, each comprising four or five separate flights. All tests were carried out without GNSS to simulate the typical localization constraints of

under-canopy flight. The drone was manually armed, and from takeoff onward, all motion decisions were made autonomously by the neural-network-based planner.

Before conducting real-world tests, extensive validation of the onboard odometry system was performed. Early attempts using open-sourced VIO algorithm, VINS-Fusion (Qin et al., 2018), with the IMU from the autopilot or from the D435i resulted in drift and unstable pose estimates. These issues were resolved by switching to off-the-shelf T265 camera with built-in VIO capability, which provided sufficiently stable real-time odometry for flight control.

The experiments were designed to progressively challenge the system in increasingly complex environments. The first experiment was conducted in a small open field ($4 \times 4 \text{ m}^2$) to verify the basic functionality of the navigation system in the absence of obstacles.

The second experiment took place in a small mixed forest ($15 \times 15 \text{ m}^2$) with varied vegetation, including trunks, branches, and dense lower foliage.

The third and most complex experiment was conducted in a dense spruce forest with limited visibility and high-contrast lighting conditions caused by sunlight and shadows. Four flights were conducted with increasing trajectory lengths (15, 30, 60, and 80 meters).

A summary of the experimental environments is provided in Table 1, and the corresponding performance results are presented in Section 3.

3. Results

3.1 Simulation Results

In RotorS simulations, the proposed sevae-ORACLE model demonstrated, in the second experiment (Sim-2), robust navigation in sparse and moderately dense forests as visible from Table 2. At a tree density of 0.1 trees/m^2 , the drone achieved a 100% success rate across all tested speeds (1.0, 1.5, 2.0 m/s), with no collisions observed. In moderately dense forests (0.15 trees/m^2), performance remained strong at lower speeds, while three out of 20 flights at 2.0 m/s resulted in collisions, yielding an 85% success rate. Minor contacts with vegetation were tolerated if the drone was able to continue the mission and complete the trajectory.

In the most challenging dense forest (0.2 trees/m^2), the drone surpassed expectations, achieving 95% success at 1.0 m/s and 90% at 1.5 m/s, despite the network not being trained on such complex scenes. However, at 2.0 m/s, performance dropped to 65%, with the drone frequently failing to react in time to avoid obstacles.

The comparison between Table 2 and Table 3 clearly demonstrates that the system developed for this work achieves overall better results than those reported in the work presented by Karjalainen et al. (2023). Specifically, seVAE-ORACLE consistently achieves a higher success rate across all tested conditions. Its performance is particularly notable in scenarios involving medium velocity and very dense forests, where it reaches a 90 % success rate compared to 65% in the work from Karjalainen et al. (2023). In other scenarios, the difference in success rates

is less pronounced, but seVAE-ORACLE still shows an advantage.

In contrast, the first experiment (Sim-1) demonstrated that the baseline ORACLE (Nguyen et al., 2024) model achieved a 0% success rate across all environments, highlighting the critical importance of the improved perception module in the sevae-ORACLE variant.

Density (tree/m ²)	[1 m/s]	[1.5 m/s]	[2 m/s]
0.10	20/20	20/20	20/20
0.15	20/20	20/20	17/20
0.20	19/20	18/20	13/20

Table 2. Success rates with refined sevae-ORACLE for different tree densities and speeds in RotorS.

Density (tree/m ²)	[1 m/s]	[1.5 m/s]	[2 m/s]
0.10	20/20	19/20	19/20
0.15	20/20	18/20	17/20
0.20	18/20	13/20	12/20

Table 3. Success rates with the work from Karjalainen (2023) and Karjalainen et al. (2023) for different tree densities and speeds in the same simulated forest environments that were used in this study.

3.2 Real-World Results

The first test, done in the open field (RW-1), revealed initial issues with grass misinterpretation and sunlight glare, which were mitigated by integrating a perception model trained on open terrain and refining the angular velocity controller. After these adjustments, the drone successfully completed four out of five autonomous flights.

In the mixed forest (RW-2), the drone completed all five flights successfully, demonstrating precise obstacle avoidance and smooth trajectory following. The planner occasionally guided the drone close to tree trunks, which was interpreted as a learned behavior from training in cluttered environments, where proximity to obstacles could increase available maneuvering space. Minor contact with leaves was observed in some flights but did not lead to failure.

In the cluttered spruce forest (RW-3), results were mixed. The first two flights were completed successfully. However, in the third and fourth flights, the drone lost control after approximately 45 meters, resulting in collisions with tree trunks. These failures were attributed to odometry drift and the reduced reaction time during abrupt transitions from open to cluttered areas. Despite speed reductions and threshold adjustments, the system showed limitations in handling extreme complexity, highlighting the need for incorporating real-world data into training for better generalization.

Flight No.	Goal (xyz)	Max. Vel.	Flight Outcome
1	[15.0, 0.0, 1.0]	1.0 m/s	Yes
2	[30.0, 0.0, 1.0]	1.0 m/s	Yes
3	[60.0, 0.0, 1.0]	1.0 m/s	Success up to 45 meters; lost control after sudden acceleration in an open area.
4	[80.0, 0.0, 1.0]	0.8 m/s	Success up to 45 meters; similar behavior to Flight 3.

Table 4. RW-3 Flight Test Results

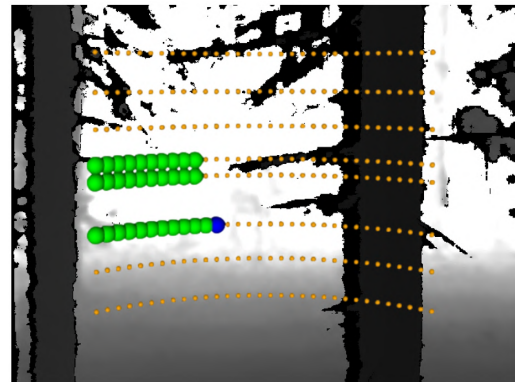


Figure 6. Planner real-time Representation Example (the green spheres represent the safe actions, and the blue one represent the chosen action that optimize the trajectory)

These results validate the neural network-based planner and show promising sim-to-real transfer. Failures in dense forests highlight the need for real-world data in training and improvements in odometry and control near transition zones.

4. Discussion

The experimental evaluation demonstrates that the fine-tuned navigation system performs reliably across both simulated and real-world forest environments. The results highlight the value of combining domain-specific perception enhancements with robust planning strategies to address the unique challenges of navigating in cluttered spaces denied by GPS, such as forests.

A central contribution of this work is the use of a fine-tuned autoencoder trained on synthetic forest-like data, and the validation of it via multiple test flights in simulated and real forest environments. This model successfully captured small and semantically relevant details in depth images, which are often lost when training on more generic datasets. The inclusion of a semantic weighting scheme during training, highlighting pixels associated with narrow structures such as branches, proved to be effective in improving obstacle perception, particularly in dense environments.

Beyond improvements to the perception module, the system also benefited from architectural and behavioral refinements. The conservative risk-avoidance policy, together with frequent trajectory replanning, enabled emergent recovery behaviors. Although not explicitly trained for recovery, the drone was able to stop, reorient, and resume navigation after near-collisions or contact with small obstacles. This robustness is particularly important when operating in highly variable unstructured environments.

Nevertheless, several limitations remain. The most persistent challenge involves detecting and avoiding thin obstacles such as branches, which often go unnoticed due to the sparsity and noise in stereo depth data. These issues were especially pronounced in the most difficult test (RW-3), where poor lighting and visual aliasing further compromised depth estimation and visual-inertial odometry. Failures in odometry were also a critical factor, as the system relies on the Intel RealSense T265 for pose estimation. In environments with repetitive textures or low visual features, the T265 suffered from significant drift and, in some cases, complete tracking failure—posing a major threat to safe and reliable navigation.

Despite these challenges, the system showed that vision-based planners trained entirely in simulation can transfer effectively to the real world, particularly in sparse forest environments. These findings support the viability of deep learning-based navigation systems in complex natural environments, while also pointing toward areas—such as sensor fusion and depth quality enhancement—where future improvements could yield substantial gains.

5. Challenges and Future Works

Despite demonstrating promising performance in forest navigation, the current system is subject to several limitations that constrain its robustness and generalizability. The most critical of these lies in the accuracy and reliability of the odometry estimation. The current setup relies primarily on the Intel RealSense T265 visual-inertial odometry (VIO) sensor. While this solution offers lightweight, real-time pose estimation without GPS, it exhibits substantial drift and even complete failure under conditions of low luminosity or rapid motion (in particular rapid curves and change of direction), scenarios frequently encountered in natural forest environments. These failures can lead to incorrect state estimates, causing unsafe behavior during navigation, particularly when the drone attempts to execute precise maneuvers in cluttered spaces. Addressing this core limitation is essential for deploying the system in more complex and unstructured real-world scenarios. Future work should explore odometry fusion strategies that combine VIO with additional sensing modalities such as LiDAR sensors to enhance pose robustness. Moreover, a replacement of the Realsense T265 sensor with a VIO open-source algorithm like the ones presented from Campos et al. (2021) or Pritchard et al. (2025), is considered fundamental.

A second limitation concerns the dependence of the perception module on stereo depth cameras. While these sensors are effective in general, they often struggle to capture fine-scale structures like thin branches, especially under poor lighting, occlusions, or in the presence of motion blur. Although the autoencoder is capable of learning latent representations that abstract some of this missing detail, its performance remains

fundamentally constrained by the quality of the raw depth input. Augmenting the perception pipeline with RGB information could help mitigate these limitations. A multimodal encoder that jointly processes depth and RGB could provide a more semantically rich and structurally complete understanding of the environment. Such an encoder could be trained using self-supervised or cross-modal consistency objectives to ensure complementary use of the two modalities.

Additionally, the current system relies on a discrete motion primitive library to generate trajectories. While this approach is computationally efficient, it lacks the adaptability required for complex or highly dynamic environments, where obstacles are irregularly spaced or unexpected behaviors are needed. Future improvements could involve adopting learning-based or sampling-based planners that operate in continuous action spaces and are guided by neural predictions derived from latent representations.

From a hardware perspective, the inclusion of alternative depth sensing modalities, such as lightweight time-of-flight (ToF) sensors or solid-state LiDAR, could enhance obstacle perception. However, these additions introduce trade-offs in terms of payload and power consumption. A complementary direction is to improve the quality of the stereo depth input via lightweight depth completion methods. For instance, integrating a sparse-to-dense completion network such as GuideNet-Lite (Tang et al., 2021) could significantly improve depth map density and accuracy, thereby strengthening the input to the autoencoder and subsequent navigation pipeline.

Finally, an important direction for future work involves scaling the system to handle cooperative multi-agent navigation and exploration under forest canopies. In such scenarios, the ability to share pose information between agents could mitigate individual localization errors and improve global trajectory planning, especially in GNSS-denied environments.

6. Conclusion

This paper evaluated a fully vision-based navigation system for autonomous drones operating in cluttered forest environments without reliance on GNSS. The approach combines a semantically-informed autoencoder with a collision prediction network to enable safe and efficient trajectory planning based solely on depth perception.

The system was validated in both simulated and real-world conditions, demonstrating reliable performance in open and moderately cluttered forest scenarios, and showing resilience under more challenging conditions despite occasional failures. In particular, the semantically-enhanced latent representations improved the ability of the drone to perceive and react to small obstacles such as branches, which are typically difficult to detect using stereo depth sensors alone.

Real-world experiments confirmed that the system could handle uncertainty in sensor input by stopping or rerouting when necessary, even without explicit recovery mechanisms. These results support the potential of simulation-trained, perception-driven navigation strategies to generalize to real environments when appropriately tailored to the domain.

While limitations remain, particularly in odometry reliability and the detection of fine structures, the outcomes demonstrate

that a lightweight, learning-based navigation stack can offer a viable foundation for autonomous drone operation in complex natural environments.

7. Acknowledgement

This research was funded by the Academy of Finland within project “Learning techniques for autonomous drone based hyperspectral analysis of forest vegetation” (decision no. 357380). This study has been performed with affiliation to the Academy of Finland Flagship Forest–Human–Machine Interplay—Building Resilience, Redefining Value Networks and Enabling Meaningful Experiences (UNITE) (decision no. 357908).

References

- Campos, C., Elvira, R., Rodriguez, J. J., Montiel, J. M., Tardos, J. D., 2021. ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial, and Multimap SLAM. *IEEE Transactions on Robotics*, 37(6), 1874–1890.
- Del Col Guglielmo, 2024. Refining Forest Mapping Drones for Swift Movement and Reliable Precision via Deep Learning.
- Ebadi, K., Bernreiter, L., Biggie, H., Catt, G., Chang, Y., Chatterjee, A., Denniston, C. E., Deschênes, S.-P., Harlow, K., Khatkhat, S., Nogueira, L., Palieri, M., Petráček, P., Petrlik, M., Reinke, A., Krátký, V., Zhao, S., Agha-mohammadi, A.-a., Alexis, K., Heckman, C., Khosoussi, K., Kottege, N., Morrell, B., Hutter, M., Pauling, F., Pomerleau, F., Saska, M., Scherer, S., Siegwart, R., Williams, J. L., Carlone, L., 2022. Present and Future of SLAM in Extreme Underground Environments. <http://arxiv.org/abs/2208.01787>.
- Gandhi, D., Pinto, L., Gupta, A., 2017. Learning to fly by crashing. *IEEE International Conference on Intelligent Robots and Systems*, 2017-September, 3948–3955.
- Gao, F., Wu, W., Gao, W., Shen, S., 2019. Flying on point clouds: Online trajectory generation and autonomous navigation for quadrotors in cluttered environments. *Journal of Field Robotics*, 36(4), 710–733. <https://onlinelibrary.wiley.com/doi/10.1002/rob.21842>.
- Globe Plants Team, 2022. Picea Abies - Norway Spruce (3D Model). Globe Plants. <https://globeplants.com/products/picea-abies-norway-spruce-3d-model> (22 December 2022).
- Hornung, A., Wurm, K. M., Bennewitz, M., Stachniss, C., Burgard, W., 2013. OctoMap: an efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots*, 34(3), 189–206. <http://link.springer.com/10.1007/s10514-012-9321-0>.
- Karjalainen, V., 2023. Vision-based navigation for autonomous drone flights inside a forest.
- Karjalainen, V., Hakala, T., George, A., Koivumäki, N., Suomalainen, J., Honkavaara, E., 2023. A DRONE SYSTEM FOR AUTONOMOUS MAPPING FLIGHTS INSIDE A FOREST – A FEASIBILITY STUDY AND FIRST RESULTS. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLVIII-1-W2-2023(1/W2-2023), 597–603. <https://doi.org/10.5194/isprs-archives-XLVIII-1-W2-2023-597-2023>.
- Koenig, N., Howard, A., 2004. Design and use paradigms for gazebo, an open-source multi-robot simulator. *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, 3, IEEE, 2149–2154.
- Kulkarni, M., Forgaard, T. J. L., Alexis, K., 2023a. Aerial Gym–Isaac Gym Simulator for Aerial Robots. *arXiv preprint arXiv:2305.16510*. <https://arxiv.org/abs/2305.16510>.
- Kulkarni, M., Nguyen, H., Alexis, K., 2023b. Semantically-enhanced Deep Collision Prediction for Autonomous Navigation using Aerial Robots. *IEEE International Conference on Intelligent Robots and Systems*, 3056–3063. <https://arxiv.org/abs/2307.11522v1>.
- Loquercio, A., Maqueda, A. I., Del-Blanco, C. R., Scaramuzza, D., 2018. DroNet: Learning to Fly by Driving. *IEEE Robotics and Automation Letters*, 3(2), 1088–1095.
- Nguyen, H., Andersen, R., Boukas, E., Alexis, K., 2024. Uncertainty-aware visually-attentive navigation using deep neural networks. *The International Journal of Robotics Research*, 43(6), 840–872. <https://journals.sagepub.com/doi/10.1177/02783649231218720>.
- Nguyen, H., Fyhn, S. H., De Petris, P., Alexis, K., 2022. Motion Primitives-based Navigation Planning using Deep Collision Prediction. *Proceedings - IEEE International Conference on Robotics and Automation*, 9660–9667. <https://arxiv.org/abs/2201.03254v3>.
- Oleynikova, H., Taylor, Z., Fehr, M., Siegwart, R., Nieto, J., 2017. Voxblox: Incremental 3D Euclidean Signed Distance Fields for on-board MAV planning. *IEEE International Conference on Intelligent Robots and Systems*, 2017-September, 1366–1373.
- Pfeiffer, M., Shukla, S., Turchetta, M., Cadena, C., Krause, A., Siegwart, R., Nieto, J., 2018. Reinforced Imitation: Sample Efficient Deep Reinforcement Learning for Mapless Navigation by Leveraging Prior Demonstrations. *IEEE Robotics and Automation Letters*, 3(4), 4423–4430.
- Pritchard, T., Ijaz, S., Clark, R., Kocer, B. B., 2025. ForestVO: Enhancing Visual Odometry in Forest Environments through ForestGlue. <https://arxiv.org/abs/2504.01261v1>.
- Qin, T., Li, P., Shen, S., 2018. VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator. *IEEE Transactions on Robotics*, 34(4), 1004–1020.
- Sadeghi, F., Levine, S., 2016. CAD2RL: Real Single-Image Flight without a Single Real Image. *Robotics: Science and Systems*, 13. <https://arxiv.org/abs/1611.04201v4>.
- Tang, J., Tian, F. P., Feng, W., Li, J., Tan, P., 2021. Learning Guided Convolutional Network for Depth Completion. *IEEE Transactions on Image Processing*, 30, 1116–1129.
- Zhang, J., Wolek, A., Willis, A. R., 2024. Uav-borne mapping algorithms for low-altitude and high-speed drone applications. *Sensors*, 24(7), 2204.