IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.
Cite as T-ASE paper.

# AI-Driven Landing Zone Detection Module for Vertical Take-Off and Landing Vehicles Using Projection-Based LiDAR-Navigation Pipelines

Nirasha Herath, *Graduate Student Member, IEEE*, Oscar De Silva, *Senior Member, IEEE*, George K. I. Mann, and Awantha Jayasiri, *Senior Member, IEEE*

***Abstract*—This paper introduces an artificial intelligence-based landing zone detection module (LZDM) for vertical take-off and landing (VTOL) navigation. It employs a projection-based point cloud semantic segmentation (PCSS) convolutional neural network model combined with point cloud accumulation and a range image generation module. The proposed method addresses the limitations of existing projection-based PCSS methods, which often struggle with low-resolution and non-repetitive scan raw light detection and ranging (LiDAR) data commonly found in aerial datasets. The proposed LZDM was developed using three sets of aerial datasets collected from a DJI M600 hexacopter drone, a DJI M300 RTK quadrotor, and a Bell412 helicopter. The results were evaluated using both qualitative and quantitative metrics, demonstrating its robustness and effectiveness. In terms of quantitative results, the proposed method achieved mean intersection over union and accuracy values greater than 0.93 and 98 percent, respectively, across all three datasets, highlighting its accuracy in identifying safe landing zones (LZs). To assess the real-time feasibility of the proposed LZDM, it was deployed on a reconfigurable hardware-accelerated module. This setup achieved processing rates higher than 10 Hz for all three datasets and a throughput of over 5 million pts/s on the Jetson AGX Xavier dedicated hardware combined with the PyTorch TensorRT optimization module. The supplementary materials, including the inference code, sample testing data, and instructions are available in our GitHub repository at *https://github.com/nira16herath/CENet-LZ-detection/tree/main***

***Note to Practitioners*—VTOLs, such as small drones and helicopters, are increasingly becoming popular in time-critical and safety-critical operations, including parcel delivery to remote areas, search and rescue, and other autonomous missions. While advanced trajectory planning systems are available to guide these vehicles to designated (or targeted) positions, the system also demands reliable and accurate landing of the vehicle at nominal and emergency situations. Therefore, ensuring reliable and rapid LZ detection to improve safe autonomous landing and expedite decision-making remains a significant challenge, particularly in unknown and unprepared terrains. Autonomous systems must identify safe LZs with precision and should avoid landing on unsafe areas, such as water bodies, inclined surfaces, or other unstable surfaces. Furthermore, VTOL-based aerial datasets typically generate low-resolution and non-repetitive scan LiDAR point clouds, making accurate LZ detection more challenging. The work proposed in the paper provides a fast, accurate, and semantic-aware LiDAR-based LZ detection that provides a low risk of misidentification, which ensures safe landings in complex environments. The proposed method has been trained and validated using real-world custom-labeled aerial datasets and deployed on reconfigurable hardware, ensuring real-time suitability for LZ detection in diverse environments. Since the method relies solely on LiDAR sensors, practitioners can implement the proposed method without additional sensor modalities or high-resolution LiDAR point clouds of environments.**

***Index Terms*—Artificial intelligence (AI), landing zone detection, projection-based point cloud semantic segmentation, vertical take-off and landing (VTOL) vehicles.**

## I. Introduction

VERTICAL take-off and landing (VTOL) vehicles, including drones and full-scale helicopters, are increasingly utilized across diverse aerial applications due to their increased mission flexibility in terms of take-off and landing [1], [2]. Identifying safe landing zones (landing zones) for VTOL vehicles is critical, specially in unknown terrains during emergency landings, where rapid and accurate LZ detection is necessary to ensure safety. Traditionally, identifying safe landing zones for VTOL vehicles has relied mainly on the visual assessments by pilots [3]. As aerial operations become more complex they demand autonomous systems, and there is a pressing need for reliable LZ detection to support automated decision-making. Such implementations reduce pilot workload, and enable fully autonomous VTOL operations in challenging environments.

To address the challenge of safe LZ detection for VTOL vehicles, studies have been carried out from traditional rule-based modeling and image processing-based methods to more advanced artificial intelligence (AI)-based methods. These methods typically rely on sensors such as cameras, light detection and ranging (LiDAR) sensors, or a combination of both [4].

Traditional heuristic methods [3], [4], [5], [6], which employ mathematical models and predefined heuristic rules,

Received 5 October 2024; revised 8 March 2025; accepted 8 April 2025. Date of publication 23 April 2025; date of current version 2 May 2025. This article was recommended for publication by Associate Editor J. Wang and Editor Z. Li upon evaluation of the reviewers' comments. This work was supported in part by the National Research Council of Canada's Artificial Intelligence for Logistics Program, in part by the Natural Sciences and Engineering Research Council of Canada, and in part by the Memorial University of Newfoundland. *(Corresponding author: Nirasha Herath.)*

Nirasha Herath, Oscar De Silva, and George K. I. Mann are with the Intelligent Systems Laboratory, Faculty of Engineering and Applied Science, Memorial University of Newfoundland, St. John's, NL A1C 5S7, Canada (e-mail: vnherath@mun.ca; oscar.desilva@mun.ca; gmann@mun.ca).

Awantha Jayasiri is with the Flight Research Laboratory, National Research Council of Canada, Ottawa, ON K1V 9B4, Canada (e-mail: awantha.jayasiri@nrc-cnrc.gc.ca).

Digital Object Identifier 10.1109/TASE.2025.3563822

IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.

are primarily dependent on geometric data from LiDAR sensors. These methods analyze terrain characteristics like slope, roughness, obstacle presence, and size of the landing areas to determine potential landing zones. These methods are simple to implement and have low computational complexity, making them suitable for straightforward environments. However, they lack the ability to incorporate semantic information of the terrains or objects, leading to inaccuracies, particularly in complex environments. For example, heuristic methods may struggle to correctly classify water bodies, frozen lakes, or unstable surfaces as non-landing zones [3], [5], [7], which poses a significant risk in real-world applications.

The image processing-based LZ detection methods [8], [9], [10] primarily rely on algorithms such as edge detection, pattern recognition, and texture analysis. These methods utilize visual data from cameras and are effective in environments within the operational envelope characterized by factors like good lighting, clear field of view, and desirable altitude levels. However, variations in lighting and altitude levels can degrade image quality, making it difficult to accurately detect landing zones, subsequently increasing the risk of false positives [8]. Additionally, these image processing-based methods often require manual parameter tuning and thresholds, which can be time-consuming and may not guarantee optimal performance as a general LZ detection method.

The AI-based methods are particularly advantageous compared to the rule-based and image processing-based methods as they leverage data-driven techniques that enable continuous learning and adaptation from new data. Additionally, these methods allow extensive code optimization and provide faster run times by utilizing graphical processing unit (GPU) computations [7]. The prevailing AI-based methods for LZ detection utilize various sensor modalities, including cameras, light detection and rangings, or a combination of both in a sensor fusion approach.

The camera-based AI LZ detection methods [11], [12], [13], [14] leverage the color and textural features in images to detect potential landing zones. The advantage of using vision-based methods lies in the high resolution and detailed imagery they can provide, making them effective in well-light conditions. However, these systems are susceptible to image blur, low visibility, glare, and reflections, specially in low-light conditions and higher altitudes [8], [15], which can significantly degrade their performance. In contrast, the LiDAR-based AI LZ detection methods [7], [16], [17] exploit the strengths of LiDAR sensors, which provide highly accurate three-dimensional (3D) spatial data. These sensors also provide accurate elevation measurement, canopy penetration [4], and dust cloud penetration using LiDAR multi-return capability, which are crucial for detecting potential landing zones, specially in challenging environments with varying terrains or lighting conditions [18]. The camera-LiDAR fused LZ detection methods [19], [20] combine data from cameras and LiDAR sensors to enhance LZ detection. These systems improve accuracy in various lighting and environmental conditions by integrating the detailed imagery from cameras with the precise spatial data from LiDAR. Although this fusion approach allows for a more comprehensive understanding of the environment, it increases the complexity of data processing.

Among the LiDAR-based AI methods, point cloud semantic segmentation (PCSS) is a recently used technique for LZ detection [7], [17] due to its ability to categorize individual points within a point cloud into meaningful semantic classes. However, point-based PCSS, a variant of LiDAR PCSS, face limitations when utilized in real-time applications due to their point-wise processing of input data [7], [17]. While accurate, these methods often suffer from higher inference times and lower processing and throughput rates [7] (e.g., throughput rates ranging from 8000 to 19000, pts/s on onboard inference hardware [7]). Such limitations hinder their suitability for real-time operations, where rapid decision-making is critical for safe and effective LZ detection in VTOL vehicles.

Projection-based PCSS, another variant of LiDAR PCSS, has gained attention due to its faster processing rate. Furthermore, this PCSS method strikes a balance between speed and accuracy [21], making it a potential candidate for real-time LZ detection in VTOL vehicles. LiDAR-based state-of-the-art aerial datasets [22], [23], [24] typically consist of low-resolution point cloud data due to a combination of factors, including the type of LiDAR sensors used in aerial applications, such as low-resolution sensors (e.g., VLP-16) [25], which generate low-resolution point clouds, and non-repetitive scan LiDAR sensors (e.g., Opal 3D, Livox Horizon, and Livox Avia) [26], [27], [28], which generate low-resolution point clouds with irregular, non-uniform point distributions (e.g., rosette patterns), as well as the variations of flight speed and altitude. However, existing projection-based methods have been designed and optimized for ground vehicle-based LiDAR data, which is typically high-resolution and consists of uniformly distributed point clouds. Therefore, the direct application of existing projection-based methods to aerial LiDAR data presents challenges in accuracy and reliability, as these methods were not originally designed to handle low-resolution point clouds. Addressing these challenges requires improving projection-based PCSS methods to suit aerial LiDAR perception while ensuring computational efficiency and segmentation reliability in real-time LZ detection for VTOL vehicles.

AI-based methods require large volumes of labeled datasets to ensure robustness and accuracy in real-world conditions. However, most of the existing camera-based [29] or LiDAR-based [30], [31], [32] annotated large databases with semantic labels are tailored for ground-based vehicle applications. Although the recent availability of LiDAR-based aerial semantic segmentation datasets [33], [34] has led to the adoption of PCSS methods for aerial applications [35], [36], there remains a significant scarcity of publicly available datasets specifically designed for LZ detection. These datasets primarily focus on urban and infrastructure scenes rather than specific LZ, non-LZ semantic labels. Similarly, while there are aerial datasets suitable for LZ detection [22], [23], they often lack the necessary semantic labels. This lack of labeled datasets poses a challenge to the development and validation of AI-based LZ detection methods, limiting their ability to generalize across diverse environments and scenarios.

IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.

To address the above-identified limitations, this paper proposes a LiDAR projection-based AI landing zone detection module (LZDM) that offers higher accuracy and faster execution. The proposed method is trained and validated on field datasets, ensuring robustness and reliability in real-world scenarios. The approach involves several key steps: first, generating a point cloud map from LiDAR data; second, acquiring ground truth semantic labels tailored to the specific terrain characteristics; third, implementing a customized region of interest (ROI) method to segment the generated point cloud map; fourth, projecting the cropped segments to range images; fifth, training the projection-based PCSS convolutional neural network (CNN) model on the resultant range images; and finally, validating the method on the LiDAR-navigation pipeline.

To the best of the author's knowledge, this is the first work that utilizes a projection-based PCSS for LZ detection for VTOL vehicles, which allows to overcome the challenges of computational complexity and general applicability of existing AI and non-AI-based safe LZ detection methods in literature.

The key contributions of this study include the following:

- Development of a novel projection-based AI LZDM for fast and accurate LZ detection for VTOL vehicles.
- Development of a LiDAR navigation pipeline for LZ detection compatible with low-resolution and non-repetitive scan LiDAR data.
- Generation of LZ and non-LZ ground truth semantic labels for *Holyrood* and *Paradise* datasets [7] and selected sequences from *MUN-FRL* [22], and *MARS-LVIG* datasets [23], totaling approximately 52 *million* labeled points.
- Validation of the system accuracy and performance of the proposed LZDM on a VTOL LiDAR-navigation pipeline.

## II. Related Work

This section presents an overview of existing LZ detection methods, categorized into three major areas: rule-based, image processing-based, and AI-based methods. Each subsection will discuss the progress and limitations within these areas.

### A. Rule-Based LZ Detection Methods

Rule-based methods employ predefined rules and mathematical algorithms to detect potential landing zones based on criteria such as terrain flatness and obstacle absence. Scherer et al. [3] have worked on enabling a full-scale helicopter to detect and land in previously unseen terrains autonomously employing LiDAR point cloud maps. This method has been field-validated with varying ground clutter and approach direction. The study [4] has presented a method for selecting safe landing zones for unmanned aerial vehicles (unmanned aerial vehicles) based on LiDAR point clouds. The method employs principal component analysis and an improved region-growing algorithm to detect flat regions suitable for landing. Terrain complexity is assessed through an improved progressive sample consensus algorithm, which helps in selecting the most appropriate LZ based on terrain complexity measures. The study validates the effectiveness of this approach through simulation experiments using terrain point clouds from urban and natural scenes. Lorenzo et al. [5] have proposed a LZ detection method using LiDAR point clouds for helicopters. The algorithm uses principal component analysis to evaluate the terrain around a point, focusing on a planar surface analysis centered at each point. It assigns quality values based on terrain slope, vegetation, and other criteria like the presence of obstacles. If the quality values indicate unsuitability (e.g., steep slope, rough terrain, or obstacles), the point is immediately classified as unsuitable for landing. The process repeats until the largest area with good terrain parameters is found, making the method accurate but time-consuming. Garg et al. [6] have presented a potential landing site selection method for unmanned aerial vehicles using terrain information from digital elevation maps. The process incorporates quadtrees to determine flat and obstacle-free areas suitable for landing. The approach not only focuses on finding safe landing zones but also integrates path planning algorithms to ensure a feasible approach path for unmanned aerial vehicles.

### B. Image-Processing-Based LZ Detection Methods

The studies [8], [9], [10] have utilized image processing techniques to detect landing zones. The study [8] has presented an automatic LZ detection system for unmanned aerial vehicles, utilizing the Gabor transform and Markov chain codes to identify and verify suitable landing zones. The system exploits the flat surface properties in images, extracting them through the Gabor transform, and then uses histogram operations across Gabor-oriented images to pinpoint candidate pixels for landing zones. These candidate pixels are then grouped as one region using Markov chain codes. The approach also incorporates methods to mitigate false detections caused by complex backgrounds and variable contrast, ensuring the reliability of the identified landing zones. However, this method tends to misidentify water surfaces as false candidate regions due to their homogeneous appearance, which can pose challenges in practical applications. Desaraju et al. have [9] presented a vision-based LZ detection method for micro aerial vehicles, particularly focusing on rooftops. The methodology estimates the landing site confidence by incorporating a Gaussian process. Singh et al. [10] have proposed a LZ detection method for multi-rotor unmanned aerial vehicles using image processing techniques combined with real-time decision-making. This method utilizes a depth stereo camera to capture images, which are then processed through the Canny Edge detection algorithm to identify edges and a diameter-area estimation algorithm to form potential landing zones within $3\,m^2$ circles devoid of edges. Concurrently, the method detects moving objects within the image by analyzing changes in pixel colors.

### C. AI-Based LZ Detection Methods

Advancements in AI-based methods for LZ detection have progressed from machine learning to more complex deep learning (DL) techniques. Initially, Mukadam et al. [13] have demonstrated the use of support vector machines with different kernels (linear, polynomial, radial basis function) combined with image processing techniques. For training and

IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.

testing, this study has used satellite images from Google Earth, with ground truth derived by manually labeling each image with binary numbers to represent potential landing zones. Concurrently, other studies [11], [12] have explored the application of two-dimensional (2D) CNN-based methods using camera images due to their faster execution. Lee et al. [11] have described a system for object detection and UAV LZ localization that employs a combination of faster R-CNN and feature-matching algorithms. This study has used a custom-collected dataset duplicating a real environment labeled with bounding boxes for known landing zones. The study [12] has introduced a CNN-based YoLo v3 algorithm for obstacle detection for drones, aiming to enhance the safety of the identified landing zones (e.g., helipads), particularly in environments where global positioning system (GPS) signals are compromised. This method has utilized a custom-collected dataset with ground truth manually labeled to include bounding boxes for both landing zones and obstacles.

However, due to the lack of reliable depth information in camera images, the studies [16], [37] have shifted towards 3D CNN methods employing voxelized LiDAR data. Maturana and Scherer [16] have developed a 3D volumetric CNN method operates on density grid maps that enhances the detection of safe landing zones from LiDAR data, particularly in low vegetated areas. This study has utilized two types of custom point cloud datasets: synthetic and semi-synthetic. The synthetic dataset was entirely created through simulations, and the semi-synthetic dataset was created by combining real point cloud data for vegetation and ground with simulated point clouds for solid obstacles. The study [37] has proposed a 3D CNN model VoxNet, for real-time object detection. VoxNet utilizes voxelized input data to efficiently process and recognize objects in 3D space, demonstrating its potential in environments where rapid and accurate object detection is crucial. Although voxelization increases computational demands and can potentially lead to the loss of fine details present in the raw point cloud data [38], the move towards 3D CNN demonstrates a significant shift towards leveraging the comprehensive spatial information provided by LiDAR sensors.

Among the available AI-based methods for LZ detection, PCSS plays a major role. The PCSS can be performed mainly in three methods: point-based [38], [39], voxel-based [40], and projection-based [18], [41]. Point-based PCSS methods operate directly on raw LiDAR point clouds, while voxel-based methods convert raw LiDAR points into regular voxel grids. Both approaches utilize 3D convolutional neural networks for processing the data. On the other hand, existing projection-based methods transform the raw 3D point clouds into spherical, cylindrical, birds-eye view (top-down orthogonal) range images [42] or fusion of any of these projection techniques [43], [44]. These range images are then processed with standard 2D convolutional neural networks for semantic segmentation. Generally, point-based and voxel-based methods achieve state-of-the-art accuracy but come with a trade-off of higher computational complexity and slower real-time inference speeds [45]. Moreover, in voxel-based methods, the information will be lost due to voxelization [44]. In contrast, projection-based methods are known for their faster inference speeds and comparable accuracy for high resolution LiDAR data [18].

Initially, image-based semantic segmentation methods have been employed for LZ detection, leveraging visual data to classify terrain features and potential landing zones. The study [14] has proposed a deep regression method for LZ detection based on image semantic segmentation. The authors have modified an image semantic segmentation model to map the thematic classes to safety scores, thereby obtaining a continuous safety map through deep regression. This method is demonstrated on public image-based datasets, with ground truth generated by assigning a safety score to each location in the RGB images. The studies [19] and [20] have presented multi-sensor-based LZ detection methods using semantic segmentation for unmanned aerial vehicles, integrating data from cameras and LiDAR. In both the methods, semantic segmentation is only applied within the image pipeline. In study [19], the semantic labels derived from camera images are transferred to the projected LiDAR point clouds in camera coordinates. Then these semantic labels, along with the geometric features of the point cloud, are used as constraints to detect potential landing zones. This method was initially trained on publicly available image and video datasets, then validated within a simulation environment, and subsequently tested in real-world scenarios. Meanwhile, Lim et al. in [20] have combined the semantic information from images to the slope extraction data from LiDAR point clouds to detect potential landing zones. This method was trained on both publicly available and custom-collected aerial image datasets, with these images being annotated using an online labeling tool.

Recent studies [7], [17] have focused on LiDAR-based semantic segmentation for LZ detection, demonstrating its advantages over image-based semantic segmentation methods, particularly in terms of depth accuracy and sensitivity to varying lighting conditions. The authors Zha and Yuan [17] have developed an autonomous LZ detection method using PCSS for space exploration missions, focusing on lunar terrain. They have constructed a lunar terrain point cloud simulation dataset, LTS-PCD, and utilized a point-based PCSS model to predict potential landing zones. The detected landing zones were then further evaluated to find the largest and nearest LZ. In [7], Balasooriya et al. have developed an autonomous LZ detection method for VTOL vehicles, utilizing LiDAR point clouds. The method employs a point-based PCSS model. This system integrates into a visual LiDAR odometry and mapping (VLOAM) [46] pipeline, ensuring real-time operational capability with accurate LZ detection. Additionally, the system was tested on different custom-collected real-world aerial datasets to evaluate its accuracy and runtime tread-off.

## III. Methodology

Fig. 1 illustrates the training process of the proposed LZDM, which involves two key modules: input generation and ground truth generation. Firstly, a full-resolution LiDAR point cloud map is generated. The resultant full-resolution map is then pre-labeled employing a pre-labeling algorithm followed by manual adjustments using a modified version of point cloud labeling tool (PCLT), forming the ground truth semantic labels

IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.

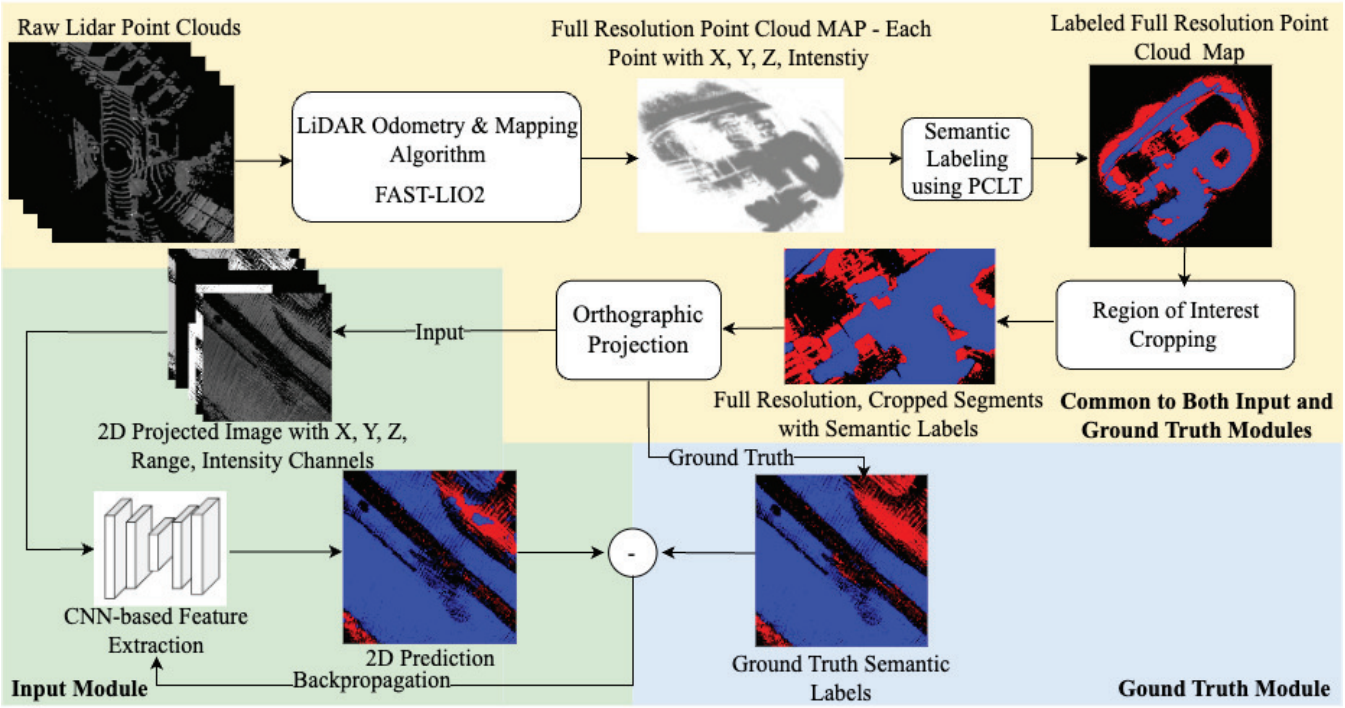


Fig. 1. Illustration of the training process of the proposed LZDM.

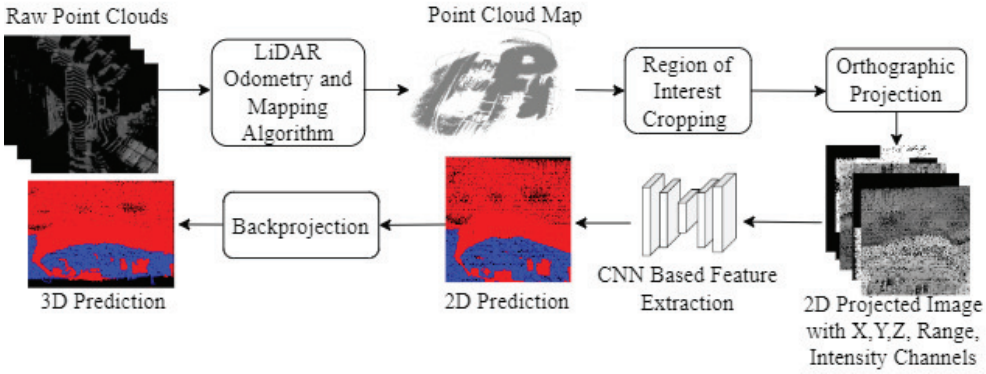


Fig. 2. Illustration of the inference process of the proposed LZDM.

(landing zones, Non-landing zones) for CNN training. Then, the full-resolution LiDAR point cloud map with semantic labels is divided into several segments in order to create the training dataset. Next, these cropped segments containing Cartesian coordinates $(X, Y, Z)$, intensity, and semantic labels are transformed into range images through a 2D projection technique, resulting in input images with five channels and the respective ground truth images for the CNN training.

The inference process of the proposed LZDM is illustrated in Fig. 2. In this process, the LiDAR point cloud map undergoes ROI cropping in real-time. Following this, the cropped 3D point clouds are projected into range images. Semantic labels for landing zones and non-landing zones are then generated for each of these projected segments.

The rest of the methodology details the dataset selection, LiDAR point cloud map generation, acquisition of ground truth semantic labels, customized ROI cropping method, and 2D projection procedure employed in this study. Following that, the CNN model, the implementation on onboard inference hardware, and the performance metrics used to assess the performance of the proposed LZDM are discussed.

### A. Data Selection

To train and validate the proposed LZDM, data from four different LiDAR-based aerial datasets were utilized. Namely, the *MUN-FRL* [22], *Holyrood* and *Paradise* [7], and the *MARS-LVIG* [23]. The selected aerial dataset sequences include *Bell412*-1, *Bell412-6*, and *Lighthouse* from *MUN-FRL* and the *HKairport02* and *AMtown03* sequences from the *MARS-LVIG* dataset. The sequences from *MUN-FRL* and *Holyrood*, *Paradise* datasets are captured using VLP-16, a low-resolution 16-channel Velodyne LiDAR sensor, whereas the

IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.

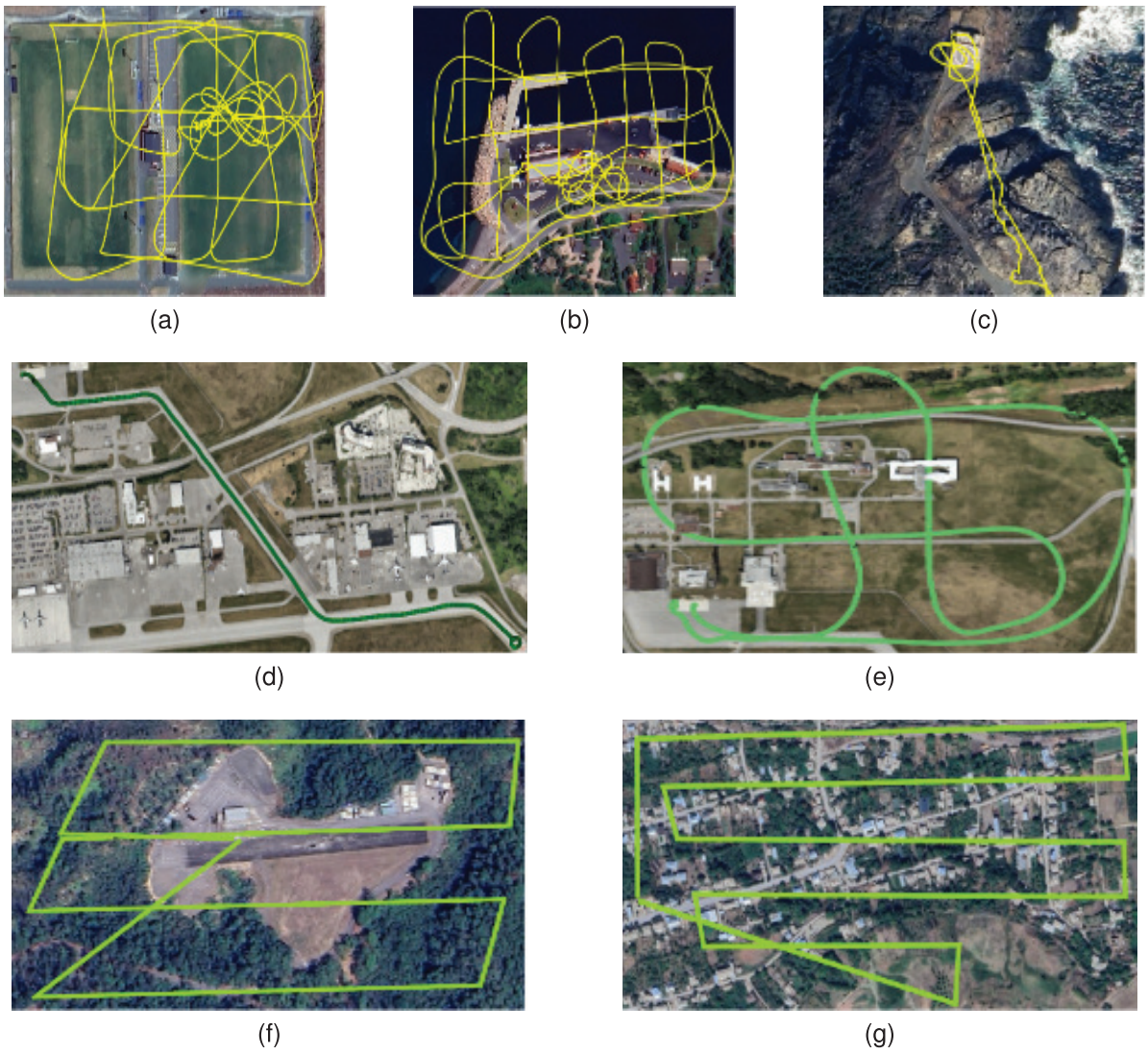


Fig. 3. Illustrations of the flight paths of the aerial datasets used in this study. (a) Paradise dataset, (b) Holyrood dataset, (c) Lighthouse sequence, (d) Bell412-1 sequence, (e) Bell412-6 sequence, (f) HKairport02 sequence, (g) AMtown03 sequence.

sequences from *MARS-LVIG* are captured using Livox Avia, a non-repetitive scan LiDAR sensor.

The selection of datasets was guided by the need to ensure diversity in environmental conditions and operational settings to improve generalization and mitigate dataset-induced biases. The included dataset sequences represent both rural and urban landscapes in different geographic locations, aerial platforms, altitudes, LiDAR sensor types, seasonal changes, and distinct noise characteristics.

In this study, *Holyrood* and *Paradise* datasets were grouped together for training and validation as they both involved pre-generated maps from [7]. Similarly, the *Lighthouse*, *Bell412-1*, and *Bell412-6* sequences were grouped together as they all consist of low-resolution LiDAR data. The *HKairport02* and *AMtown03* sequences were also grouped together as they both consist of non-repetitive scan LiDAR data. Henceforth, these dataset groupings will be referred to as *Holyrood-Paradise*, *Lighthouse-Bell412*, and *HKairport-AMtown*, respectively.

The next sections further discuss the aerial datasets used in this study.

1) **Aerial Dataset 01: *Holyrood-Paradise***
The *Paradise* dataset, shown in Fig. 3a, and the *Holyrood* dataset, shown in Fig. 3b were captured in Paradise and Holyrood, Newfoundland, Canada, respectively, by the Intelligent Systems Lab (ISL) at Memorial University of Newfoundland (MUN). Data collection for both datasets was performed using a DJI M600 hexacopter drone. The *Paradise* dataset features a football ground surrounded by vegetation, while the *Holyrood* dataset includes a marine base parking lot.
2) **Aerial Dataset 02: *Lighthouse-Bell412***
The *MUN-FRL* dataset is collected by ISL at MUN. This dataset includes nine sequences, including the *Lighthouse*, *Bell412-1*, and *Bell412-6*, featuring flight distances ranging from 300 m to 5 km. The selected sequences in this study were captured using a DJI-M600 hexacopter drone and the National Research Council Bell412 Advanced Systems Research Aircraft.
The *Lighthouse* sequence, captured by the hexacopter drone at an altitude of 28 meters, features diverse terrain

IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.

with a helipad, ocean waves, and snowy rocks in St. John's, Newfoundland, Canada, providing both landing and non-landing scenarios. The *Bell412-1* sequence, at an altitude of 25 m, includes a low-altitude flight along a flat taxiway. The *Bell412-6* sequence, flying at 168 m, includes a looping trajectory mostly over buildings, highways, and runways. Both the *Bell412* sequences were captured by the Bell412 helicopter in the vicinity of Ottawa International Airport, Ontario, Canada.

Fig. 3c, Fig. 3d, Fig. 3e depicts the flight paths of the *Lighthouse*, *Bell412-1*, and *Bell412-6* sequences from the *MUN-FRL* dataset on Google Satellite map based on the GPS data recorded by the payload set-up.

3) **Aerial Dataset 03: *HKairport-AMtown***

The *MARS-LVIG* dataset is collected by Mechatronics and Robotic Systems lab at the University of Hong Kong. This dataset, which was captured by a DJI M300 RTK quadrotor, consists of 21 sequences captured across four distinct environments that feature an aero-model airfield, an island, a rural town, and a valley. For this study, the *HKairport02* sequence from the aero-model airfield and *AMtown03* from the rural town were selected based on their distinct terrain features and the richness of their data sets, which were ideal for training and testing of the proposed LZDM.

The *HKairport02* sequence was captured at the Hong Kong Model Engineering Club in Yuen Long, featuring flights over a concrete runway at a consistent altitude of 80 m and a cruising speed of 6 m/s. This sequence, covering a path length of 2.040 km, provides detailed insights into UAV operations in semi-structured airport environments.

The *AMtown03* sequence captured in Urtsadzor in Armenia covers a comprehensive aerial survey of a rural town at a higher cruising speed of 12 m/s and covers an extensive path length of 5.109 km, the longest among all sequences surveyed.

Fig. 3f and Fig. 3g depict the flight paths of the *HKairport02* and *AMtown03* sequences from the *MARS-LVIG* dataset on Google Earth, respectively.

### B. LiDAR Point Cloud Map Generation

This study introduces a modified LiDAR point cloud map generation process and a real-time ROI cropping method. These key steps are designed to achieve higher accuracy by improving the resolution of the point clouds used in the proposed LZDM while maintaining a faster runtime in the context of low-resolution and non-repetitive LiDAR data.

Cropped segments from the LiDAR point cloud map were utilized in this study because they provide more context-specific information about the environment than raw LiDAR point clouds. Moreover, this approach increases the resolution of the input point cloud segments to the CNN compared to using raw point cloud as inputs, offering better semantic segmentation capabilities [47]. Fig. 4a illustrates a part of the point cloud map for the *Lighthouse* dataset, including a raw point cloud shown in white. Fig. 4b and 4c illustrate enlarged views of the yellow-highlighted area in Fig. 4a. Specifically,

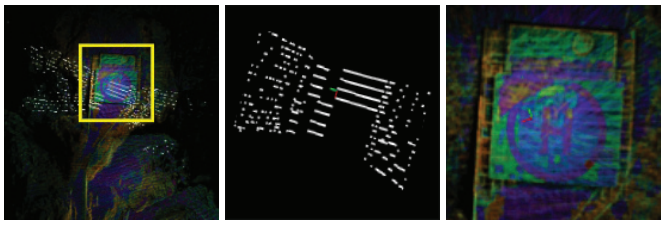


Fig. 4. Illustration of (a) part of the point cloud map for the Lighthouse dataset, (b) enlarged view of the raw point cloud falls within the yellow rectangular area in column 2, (c) enlarged view of the point cloud map segment falls within the yellow rectangular area in column 3.

Fig. 4b presents the enlarged view of the raw point cloud within the highlighted area, while Fig. 4c displays the map segment in the same area. The increased point density in Fig. 4c compared to Fig. 4b is evident, providing additional information that can enhance the performance of the CNN.

As the mapping algorithm, fast-direct LiDAR-inertial odometry (FAST-LIO2) [48] was chosen from the various available LiDAR odometry and mapping algorithms to generate LiDAR point cloud maps for all the aerial datasets used in this study. For the two Bell412 dataset sequences, additional visual features were incorporated into the mapping process using a similar method as in [22] due to the lack of sufficient features in the point clouds for FAST-LIO2. The decision was solidified after a thorough evaluation of the accuracy provided by the available mapping algorithms in generating the LiDAR point cloud maps. Additionally, FAST-LIO2 mitigates noise through its tightly coupled LiDAR-inertial measurement unit (IMU) fusion and removes outliers using robust correspondence. However, some residual noise may still be present due to inherent limitations of the LiDAR sensors and environmental disturbances. To account for this, the proposed method was evaluated across diverse datasets, as described in Section III-A, to assess its performance under various operational and environmental conditions.

While the selected mapping algorithm was effective, modifications were necessary to tailor the original algorithm to meet the requirements of our study, as outlined below.

1) **LiDAR Full-resolution point cloud map:**
A significant limitation of the original mapping algorithm was its reliance on down sampled LiDAR point clouds, for LiDAR map generation. In this study, the original FAST-LIO2 algorithm was modified to generate full-resolution maps. This modification was used in both the training and inference phases of this study.

2) **LiDAR point cloud map cropped segments:**
To ensure that LiDAR point cloud maps were of manageable sizes for processing by the projection-based PCSS model during training and validation, a customized cropping method was developed. Consequently, modifications were made to the original mapping algorithm to allow it to publish cropped segments from the full-resolution point cloud map. It is important to note that the cropped segments are published concurrently with the full-resolution map generation in real-time. This

IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.

modification in the original mapping algorithm was used only during the inference phase of this study to enable the real-time generation of LiDAR point cloud map cropped segments. In the training phase, an offline ROI cropping method was developed, which did not involve this modification to the LiDAR point cloud mapping algorithm. The ROI cropping methods developed for both phases are detailed in Section III-D.

### C. Acquisition of Ground Truth Semantic Labels

Initially, a pre-labeling method described in [7], which was adapted from [3], was employed to generate ground truth semantic labels for the full-resolution LiDAR point cloud map for all sequences used in this study. The employed pre-labeling algorithm first divided the point cloud map into $3 \times 3\,\mathrm{m}^3$ patches, considering the VTOL vehicle dimensions and $\pm 1$m navigation accuracy. Then, patches with a standard deviation of the z-axis less than 0.5 m were selected. Next, the selected patches were checked to ensure the maximum z-deviation was less than 6 m. Patches that satisfied both conditions were then checked for a point density greater than 15 points per patch. Finally, the resultant patches with a slope below 5° were labeled as landing zones. Although this pre-labeling method was helpful in detecting landing zones and non-landing zones using the considered criteria, it incorrectly classified some areas. Specifically, it did not perform well in detecting water bodies as non-landing zones. Therefore, after pre-labeling, the resultant semantic labels were manually adjusted using a modified version of PCLT which is used in [30].

### D. Customized ROI Cropping Method

The next step of the proposed LZDM is the ROI cropping. This cropping process is necessary for managing the size of the input data to ensure it remains within the computational capabilities of the projection-based PCSS model, thereby avoiding processing delays that could impede the system's performance. Furthermore, by cropping the area around the vehicle's immediate vicinity, the system focuses on the most relevant areas for real-time decision-making. This localized approach not only streamlines data processing but also enhances the timeliness of the information, which is essential for quick LZ decisions.

For the training data, $100 \times 100 \times z_{\text{range}}\,\mathrm{m}^3$ bounding boxes were randomly cropped from the LiDAR point cloud map with the condition that each cropped segment contained a minimum of $200,000$ points. $z_{\text{range}}$ stands for the range of the $z$ coordinates in the generated LiDAR point cloud map. The considered bounding box parameters were chosen as they provide a good representation of the environment across all the altitude levels of the data sequences used in this study. Moreover, the minimum point count was selected through simulation studies in order to filter out the sparse areas, providing sufficient data density for successful training and enhanced model performance. This strategy was crucial given the scarcity of comprehensively labeled aerial datasets for LZ detection and plays a significant role in enriching the dataset for more robust model training.

The area directly behind the vehicle's most recent mapping pose typically shows a higher density of data points compared to the front, which tends to be sparser. Therefore, to prioritize the denser, more contextually rich cropped segment, the ROI cropping method employed during the inference phase of the proposed LZDM slightly differs from that used in the training phase. Moreover, unlike in the training phase, the inference phase ROI cropping happens in real-time, concurrently with map generation.

Generally, a pose represents the position and orientation of the VTOL vehicle in 3D space. A mapping pose in LiDAR mapping represents the estimated position and orientation of the vehicle at a given time, which is used to register LiDAR scans into a map and update the environment representation.

**Algorithm 1** Customized ROI Cropping Algorithm (Inference Phase)

**Inputs:** $LPCM$, $P_k$
**Outputs:** $CPCS$
$x_{off}, y_{off} \leftarrow 50, 50$
$x_{\min}, x_{\max}, y_{\min}, y_{\max}, z_{\min}, z_{\max} \leftarrow \text{minMax}(LPCM)$
$x_k, y_k, z_k \leftarrow P_{k-10}$
$x_1, x_2 \leftarrow x_k - x_{off}, x_k + x_{off}$
$y_1, y_2 \leftarrow y_k - y_{off}, y_k + y_{off}$
$z_1, z_2 \leftarrow z_{\min}, z_{\max}$
$CPCS \leftarrow \text{crop}(LPCM, x_1, x_2, y_1, y_2, z_1, z_2)$

In inference, the cropping process begins by selecting the most recent mapping pose ($P_k$) of the LiDAR point cloud map ($LPCM$) generated at that moment as the reference point. From there, moving ten mapping poses backward along the trajectory, at pose ($P_{k-10}$), a $100 \times 100 \times z_{\text{range}}\,\mathrm{m}^3$ bounding box was defined. Where, $z_{\text{range}}$ is the range of the $z$ coordinates of the $LPCM$. Then, the point cloud data within this bounding box were extracted and used as the cropped point cloud segment ($CPCS$) for inference. The raw LiDAR point clouds of the datasets considered in this study are initially represented in the LiDAR coordinate frame. As the mapping algorithm FAST-LIO2 generates a globally registered $LPCM$, both the $LPCM$ and the extracted $CPCS$ are represented in the world coordinate frame. Before projecting the $CPCS$ into the image coordinate frame for training and inference, it is transformed into the local coordinate frame, as shown in (3). The specific positioning of the bounding box relative to the $P_k$, as well as its dimensions, were meticulously optimized through extensive parameter tuning and analysis to ensure accuracy in real-time operations. This process is followed during inference to enable real-time predictions concurrently with the generation of the LiDAR point cloud map while in flight, ensuring that the cropped segments align precisely with the dynamic environment.

As the selected datasets in this study feature low flight speeds and LZ detection is primarily conducted during the approach and landing phases of VTOL operations, higher-speed flight scenarios are not expected in the phases considered. In lower-speed scenarios (e.g., hover), the pose selection method remains effective as the $CPCS$ is generated by centering at the hovering location.

IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.

The customized ROI cropping method used to obtain one *CPCS* during the inference phase is summarized in Algorithm 1. This process is executed iteratively at each new mapping pose, generating a new *CPCS* at the LiDAR odometry update frequency, as determined by the mapping algorithm (FAST-LIO2) employed in this study. The mapping pose update frequency in Algorithm 1 is same as the LiDAR odometry update frequency, as determined by the mapping algorithm (FAST-LIO2).

### E. Orthographic Projection

In this study, the orthographic projection technique was specifically chosen to create a top-down view of the *XY* plane for the aerial datasets. This technique was selected for several key reasons:

1) **Clarity in Representation:**
   Orthographic projection provides a clear and undistorted view of the landscape from above, which is essential for aerial datasets where elevation changes are significant and need to be accurately represented without perspective distortion.
   Although alternative projection techniques, such as spherical and cylindrical projections, are commonly used for 3D to 2D transformations in ground-based applications, they introduce distortions and large empty sectors (sky) with empty returns when applied to aerial data, leading to unnecessary processing. This is because, in aerial applications, LiDAR point clouds are generated through a sweeping motion, capturing only a limited sector (swath) of the environment rather than a full sphere or cylinder. These challenges can negatively impact LZ detection by altering the perceived surface geometry.
2) **Enhanced Spatial Accuracy:**
   A top-down orthographic view is useful for tasks requiring precise spatial measurements, such as identifying landing zones or other specific geographical features in aerial datasets. It ensures that all points are equally scaled, providing a true-to-scale map that is crucial for accurate LZ detection.

In the training phase, following the ROI cropping process, each *CPCS* was transformed into two 2D images: an input image (*IMG*) and a ground truth image (*GT*), each with a resolution of $300 \times 300$ pixels using orthographic projection. This conversion involved projecting the *CPCS* onto the *XY* plane. The resolution of $300 \times 300$ pixels was selected after a preliminary ablation study, considering the trade-off between accuracy, resolution, and processing rate. Moreover, this resolution is sufficient for VTOL LZ detection as each pixel captures an area of $0.11\,\mathrm{m}^2$, which is smaller than the footprint of the smallest aerial platform and significantly smaller than that of the largest aerial platform considered in our study.

The *IMG* has dimensions $(H, W, C)$, where $H$ is the height, $W$ is the width, and $C$ is the number of channels. Each pixel within *IMG* comprises five channels $(x_n, y_n, z_n, d, r)$, where $x_n$, $y_n$, and $z_n$ are the cartesian coordinates of a LiDAR point in 3D space, and $r$ is the intensity. The depth $d$ of every point is derived as in (1).

$$d = \sqrt{x_n^2 + y_n^2 + z_n^2} \tag{1}$$

To mitigate the potential loss of 3D information caused by orthographic projection, this five-channel representation was used, specifically incorporating the z-coordinate. This ensures that elevation variations and terrain features are preserved, allowing the model to accurately interpret spatial relationships despite the 2D transformation.

Furthermore, in cases where multiple points map to a single pixel, only the point with the "lowest $z$-value" is considered, discarding the others. This method ensures that the projection prioritizes the highest physical obstacles, as seen from the sensor's viewpoint. Here, the points with "lowest $z$-value" refers to the points that are closest to the sensor, typically representing the highest points, such as treetops, buildings, or other structures, when the sensor is positioned above the scanned area.

On the other hand, the *GT* image was generated by projecting the *CPCS* onto the *XY* plane to produce a single-channel image where each pixel represents the semantic label ($label_n$) of each projected LiDAR point.

The $x_n$ and $y_n$ coordinates of each LiDAR point in *CPCS* were converted into image coordinates $u_n$ and $v_n$ as in (2) and (3), where $x_{\min}, x_{\max}, y_{\min}$, and $y_{\max}$ stands for the minimum and maximum values of $x$ and $y$ coordinates in *CPCS*. The subtraction of 1 in (2) accounts for the zero-based indexing convention used in image processing, where pixel indices range from 0 to H−1 for height and 0 to W−1 for width.

$$\begin{pmatrix} u_n \\ v_n \end{pmatrix} = \begin{pmatrix} (x_{\text{normalized}} \cdot (\mathrm{W} - 1)) \\ (y_{\text{normalized}} \cdot (\mathrm{H} - 1)) \end{pmatrix} \tag{2}$$

$$\begin{pmatrix} x_{\text{normalized}} \\ y_{\text{normalized}} \end{pmatrix} = \begin{pmatrix} \frac{x_n - x_{\min}}{x_{\max} - x_{\min}} \\ \frac{y_n - y_{\min}}{y_{\max} - y_{\min}} \end{pmatrix} \tag{3}$$

The orthographic projection algorithm used in the training phase of the proposed study for training data generation is summarized in Algorithm 2. In the inference phase, the same algorithm is employed without the generation of the *GT*.

### F. Projection-Based PCSS Model

The CENet, a projection-based PCSS model developed by Cheng et al. [18], was chosen as the baseline model for the proposed LZDM due to its outstanding performance on LiDAR-based semantic segmentation benchmark datasets [30], [31] and its open-source availability at the time of writing.

To tailor the CENet model to the requirements of this study, several key modifications to the original CENet model were implemented. These adjustments were aimed at optimizing the model's performance for LZ detection. In the training phase, the CENet model was modified to input orthographic images directly into it, eliminating the original model's need for the internal spherical projection step. This modification streamlined the data input process, enabling a more efficient handling of input image data. In the inference phase, the CENet model was modified to incorporate orthographic projection instead of spherical projection for the input point cloud map segments.

IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.

**Algorithm 2** Orthographic Projection Algorithm

**Inputs:** $CPCS$
**Outputs:** $IMG, GT$ (Matrices to store input range images and respective ground truth images, respectively)
**Initialize:**
$H \leftarrow 300$
$W \leftarrow 300$
$C \leftarrow 5$
$x_{\min}, x_{\max}, y_{\min}, y_{\max} \leftarrow \text{minMax}(CPCS)$
$N \leftarrow \text{length}(CPCS)$(Number of LiDAR points in $CPCS$)
$IMG \leftarrow [0]_{H\times W\times C}$
$GT \leftarrow [0]_{H\times W}$
$n \leftarrow 0$
**while** $n < N$ **do**
  $x_n, y_n, z_n, r_n \text{or } label_n \leftarrow CPCS_n$
  $d_n \leftarrow \sqrt{x_n^2 + y_n^2 + z_n^2}$
  $x_{\text{normalized}} \leftarrow (x_n - x_{\min})/(x_{\max} - x_{\min})$
  $y_{\text{normalized}} \leftarrow (y_n - y_{\min})/(y_{\max} - y_{\min})$
  $u_n \leftarrow (x_{\text{normalized}}) \times (W - 1)$
  $v_n \leftarrow (y_{\text{normalized}}) \times (H - 1)$
  $IMG_{u_n,v_n,0} \leftarrow x_n$
  $IMG_{u_n,v_n,1} \leftarrow y_n$
  $IMG_{u_n,v_n,2} \leftarrow z_n$
  $IMG_{u_n,v_n,3} \leftarrow d_n$
  $IMG_{u_n,v_n,4} \leftarrow r_n$
  $GT_{u_n,v_n} \leftarrow label_n$
  $n \leftarrow n + 1$
**end while**

TABLE I
NUMBER OF CROPPED POINT CLOUD SEGMENTS USED FOR TRAINING AND VALIDATION OF THE PROPOSED METHOD

| Dataset | Training Data | Validation Data | Number of Epochs |
|---|---|---|---|
| Holyrood-Paradise | 480 | 120 | 100 |
| Lighthouse-Bell412 | 480 | 120 | 100 |
| HKairport-AMtown | 480 | 120 | 100 |

Furthermore, the output of the model was modified in both the training and inference phases to focus exclusively on two categories: landing zones and non-landing zones.

### G. Implementation and Inference Speed Evaluation on Onboard Inference Hardware

The proposed method was trained and validated using a V100 GPU in Compute Canada cloud computer [49]. The number of training and validation data used from each set of datasets and the number of training epochs are shown in Table I.

To evaluate the inference speed and real-time feasibility on onboard inference hardware, the Nvidia Jetson AGX Xavier (Jetson AGX) developer kit with 512-core volta GPU with 64 Tensor cores was utilized [50]. This device was selected in this study to replicate the onboard inference capabilities of a VTOL vehicle, ensuring the model's suitability for real-time applications. Testing on the Jetson AGX aimed to validate that the proposed method can meet the required speed and performance for effective LZ detection in real-time scenarios. Ensuring compatibility with this onboard inference hardware was essential to ensure the operational success and safety of VTOL missions where autonomous landing is required. For further optimization of the model, PyTorch TensorRT optimization [51] was used.

### H. Performance Metrics

The performance of the proposed LZDM was assessed using mean intersection over union (mIoU), accuracy, average runtime, average throughput, and average processing rate as defined follows.

1) **Accuracy:** Evaluates the proportion of correctly predicted instances (both positive and negative) out of the total number of predicted instances.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

2) **mIoU:** A commonly used performance metric in evaluating DL-based PCSS models, and it assesses the ratio of the intersection (the common area between predictions and ground truth) to the union (the combined area covered by either predictions or ground truth) averaged over the number of classes.

$$\text{mIoU} = \frac{1}{C}\sum_{i=1}^{N}\frac{\text{Intersection}_i}{\text{Union}_i} \tag{5}$$

3) **Throughput:** Measures the number of points processed during a given time, reflecting the efficiency of the model.

$$\text{Average Throughput} = \frac{1}{N}\sum_{i=1}^{N}\frac{\text{pts}_i}{\text{t}_i} \tag{6}$$

4) **Average runtime:** Sum of the inference time (the sum of the time taken for 2D prediction by the CNN and the re-projection) and the cropping time (the time taken to pre-process the input data by cropping the point cloud map into segments in real-time before feeding them into the projection-based PCSS model), indicating the time efficiency.

$$\text{Average Runtime} = \frac{\sum_{i=1}^{N}(t_i + ct_i)}{\text{N}} \tag{7}$$

5) **Average processing rate:** Refers to the prediction frequency per input point cloud segment, highlighting the prediction frequency of the model.

$$\text{Average Processing Rate} = \frac{1}{\text{Average Runtime}} \tag{8}$$

Where $N$ denotes the number of cropped point cloud segments, $C$ denotes the number of classes (landing zones and non-landing zones), $TP$ stands for true positives, $FP$ for false positives, $TN$ for true negatives, and $FN$ for false negatives. $TP$ and $TN$ represent correctly classified positive and negative cases, respectively. In contrast, $FP$ and $FN$ are the misclassified positive and negative cases. Additionally, $t_i$ and $ct_i$ are the inference time and the cropping time for the $i^{\text{th}}$ scan. $pts_i$ is the total number of points in $i^{\text{th}}$ scan.

IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.

TABLE II
TUNED PARAMETERS FOR EACH RULE IN THE HEURISTIC METHOD FOR EACH VALIDATION DATASET FOR LZ DETECTION

| Dataset | Rule 01: No. of Points for a Plane Fit | Rule 02: Vertical Standard Deviation (spread of the points) (cm) | Rule 03: Plane Fit Successful | Rule 04: Residual Threshold | Rule 05: Slope Threshold |
|---|---|---|---|---|---|
| Original Parameters [3] | | | | | |
| N/A | >15 | <50 | True | <4 | <5 |
| Tuned Parameters | | | | | |
| Holyrood-Paradise | >20 | <60 | True | <0.9 | <4 |
| Lighthouse-Bell412 | >20 | <90 | True | <2 | <6 |
| HKairport-AMtown | >20 | <50 | True | <2 | <8 |

## IV. RESULTS AND DISCUSSION

Firstly, this section presents a performance comparison of the proposed LZDM with several state-of-the-art methods, including a traditional heuristic method [3], point-based PCSS method designed for LZ detection [7], and two projection-based PCSS models [18], [52]. Secondly, this section presents the key results of the proposed LZDM in this study, featuring both quantitative and qualitative assessments. These evaluations assess the effectiveness of the proposed LZDM when applied to aerial datasets comprised of low-resolution LiDAR and non-repetitive scan LiDAR data.

Table I presents a summary of the number of data used in both the training and validation phases of our method. The *Holyrood-Paradise*, *Lighthouse-Bell412*, and *HKairport-AMtown* datasets each provided 480 segments for training and 120 segments for validation. The training was conducted for 100 epochs per dataset.

To ensure a fair comparison, the same number of epochs was used for all methods in Section IV-A. Further, the projection-based PCSS models were trained and validated using raw point clouds, whereas the point-based PCSS method, the traditional heuristic method, and the proposed method were trained and validated using *CPCSs*, maintaining consistency in the number of training and validation data across all methods.

### A. The Performance Comparison on Validation Datasets

In this study, the traditional heuristic LZ detection method introduced by Scherer et al. [3] is used for comparison. This method has been widely recognized as a landmark contribution to LiDAR-based LZ detection, demonstrating significant influence on subsequent research in the literature. Its extensive adoption and impact on LiDAR-based LZ detection for VTOL vehicles make it an appropriate baseline for evaluating the proposed LZDM. For comparison with the proposed LZDM, the coarse evaluation of landing zones from [3] was implemented. In addition, all of its parameters were fine-tuned to better suit the data sets considered in this study, ensuring a fair and accurate evaluation. The tuned parameters for each rule in the heuristic method for each dataset used in this study are shown in Table II.

It is important to note that this heuristic method was also initially utilized for pre-labeling in this study, as described in Section III-C. However, the pre-labeling process was implemented solely to accelerate annotation and does not serve as a ground truth measurement. The resultant labels underwent manual adjustments to ensure accurate LZ and non-LZ labeling, making the ground truth labeling process primarily manual rather than directly derived from the heuristic method.

The ConvPoint-based LZ detection method, recently introduced in [7] for VTOL vehicles, is employed as the point-based PCSS method for comparison. Since this method builds upon ConvPoint [38], a state-of-the-art point-based PCSS model originally designed for ground vehicle-based applications, it is referred to as the 'ConvPoint model' throughout this paper. The presented results of the ConvPoint model in Table III were obtained using a batch size of 24 with a sampling size (the number of points taken from the input point cloud in each batch processing) of 3000. These parameters were chosen based on the findings in [7], which indicate that a lower sampling size combined with a high batch size provides a suitable hyperparameter setting to achieve a better trade-off between average accuracy and processing rate.

As the projection-based PCSS models for comparison the FiDNet [52] and the original CENet [18] are employed in this study. The primary reason for selecting these models is their open-source availability, and demonstrated effectiveness in ground-vehicle-based semantic segmentation applications. The original models were modified and retrained to enable their application for aerial LZ detection. The modifications included modifying the image size, the class point ratios, and the number of classes. The results presented in Table III for FiDNet and the original CENet were obtained for a spherical image resolution of $16 \times 2028$ pixels, following its original implementation, representing the maximum attainable resolution.

The results presented in Table III and Fig. 5 compare a traditional heuristic method, a point-based PCSS method designed for LZ detection, and two projection-based PCSS models with the proposed LZDM across three datasets used in this study. The metrics, including mIoU, accuracy, and average throughput, were obtained using a Compute Canada cloud computer [49]. Additionally, Fig. 5 visually contrasts the performance of all evaluated methods in terms of mIoU, accuracy, and throughput, highlighting the significant improvements achieved by the proposed method. Each radar chart in Fig. 5 represents different dataset comparisons, where the axes correspond to mIoU, accuracy, and throughput. The closer a point is to the outer perimeter, the higher the performance in that specific metric, allowing for a clear

IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.

TABLE III
THE PERFORMANCE COMPARISON ON VALIDATION DATASETS

| Category | Method | Dataset | Correct Predictions (TP+TN) | mIoU | Accuracy | Throughput (pts/s) |
|---|---|---|---|---|---|---|
| Traditional | Heuristic Method [3] | Holyrood-Paradise | 462118.4 | 0.750 | 0.913 | 10550478.5 |
| | | Lighthouse-Bell412 | 839627.2 | 0.671 | 0.869 | 12368886.4 |
| | | HKairport-AMtown | 1178698.8 | 0.791 | 0.917 | 5700935.2 |
| Point-Based | ConvPoint [7] | Holyrood-Paradise | 479327.6 | 0.879 | 0.947 | 242353.7 |
| | | Lighthouse-Bell412 | 891802.0 | 0.568 | 0.923 | 270343.3 |
| | | HKairport-AMtown | 1091292.6 | 0.719 | 0.849 | 373134.3 |
| Projection-Based | FiDNet [52] | Holyrood-Paradise | 21612.3 | 0.747 | 0.854 | 3012759.9 |
| | | Lighthouse-Bell412 | 26508.0 | 0.628 | 0.758 | 3568479.6 |
| | | HKairport-AMtown | 42481.8 | 0.641 | 0.794 | 4147567.8 |
| | Original CENet [18] | Holyrood-Paradise | 23282.6 | 0.859 | 0.920 | 1902795.7 |
| | | Lighthouse-Bell412 | 33782.0 | 0.908 | 0.966 | 2629406.0 |
| | | HKairport-AMtown | 50453.9 | 0.790 | 0.943 | 3992807.8 |
| | **Our Proposed Method** | Holyrood-Paradise | **501598.4** | **0.981** | **0.991** | **22199728.8** |
| | | Lighthouse-Bell412 | **947841.5** | **0.934** | **0.981** | **31370109.3** |
| | | HKairport-AMtown | **1264819.6** | **0.955** | **0.984** | **36725310.5** |

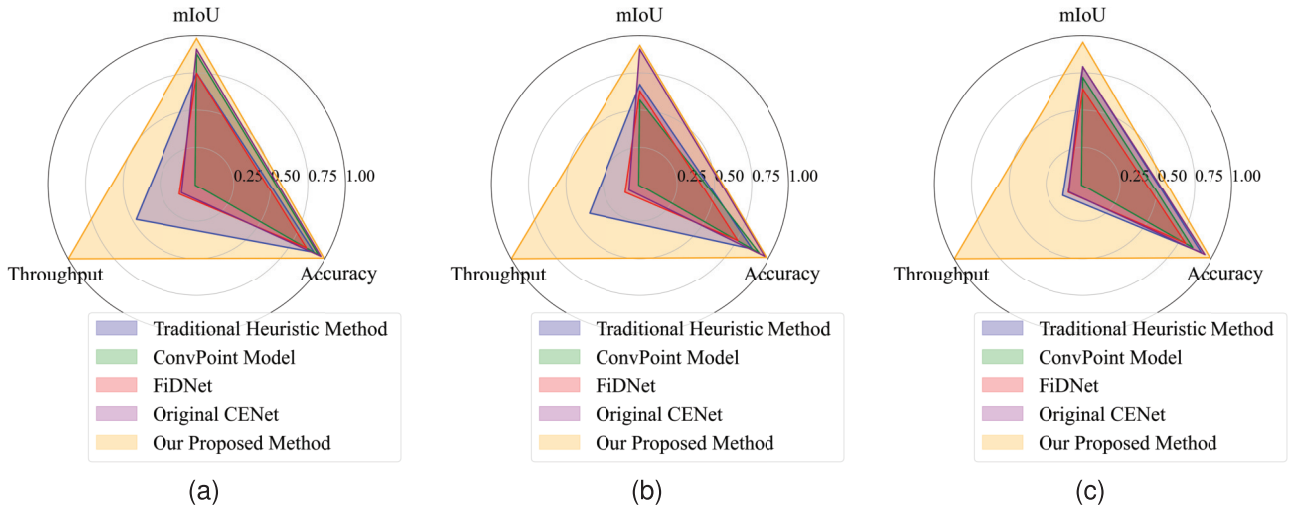


Fig. 5. Illustration of the performance comparison results for (a) Holyrood-Paradise, (b) Lighthouse-Bell412, (c) HKairport-AMtown datasets.

visual comparison of the strengths of each method across the datasets.

The results in Table III and Fig. 5 indicate that the proposed LZDM consistently outperforms all the evaluated methods across all the datasets considered in this study in terms of mIoU, accuracy and throughput. Among the compared projection-based methods, the proposed method achieves superior segmentation performance while significantly improving throughput, demonstrating its effectiveness for LZ detection. Although the mIoU and accuracy values appear closer on the *Lighthouse-Bell412* dataset for the original CENet, the proposed method significantly outperforms it in terms of the number of correct predictions. As expected, the proposed method offers significantly higher throughput compared to the ConvPoint model, which is a point-based PCSS method, making the proposed method more suitable for real-time LZ detection. Furthermore, while the heuristic method achieves higher throughput than other AI-based methods, its performance remains significantly lower than the proposed method. Unlike the traditional heuristic method, where parameters must be manually fine-tuned for each dataset to achieve optimal performance, the proposed method does not require dataset-specific parameter adjustments, making it more adaptable and scalable. These improvements suggest that the proposed method is better at handling the low-resolution and non-repetitive scan LiDAR data used in LZ detection for VTOL vehicles while also addressing the need for extensive manual parameter tuning.

TABLE IV
VALIDATION MIOU AND CLASS WISE IOU RESULTS FOR THE PROPOSED LZDM

| Aerial Datasets | mIoU | LZ IoU | non-LZ IoU |
|---|---|---|---|
| Holyrood-Paradise | 0.981 | 0.985 | 0.978 |
| Lighthouse-Bell412 | 0.934 | 0.978 | 0.890 |
| HKairport-AMtown | 0.955 | 0.930 | 0.979 |

### *B. Quantitative Results of the Proposed Method on Low-Resolution LiDAR Data*

The quantitative analysis of the proposed LZDM on low-resolution LiDAR data is presented in Table IV and Table V. The mIoU values for the *Holyrood-Paradise* and *Lighthouse-Bell412* datasets are 0.981 and 0.934, respectively, showcasing

IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.

TABLE V
VALIDATION ACCURACY RESULTS FOR THE PROPOSED LZDM

| Aerial Datasets | Accuracy |
|---|---|
| Holyrood-Paradise | 0.991 |
| Lighthouse-Bell412 | 0.981 |
| HKairport-AMtown | 0.984 |

the robustness of the proposed method in diverse environmental conditions. Specifically, for the *Lighthouse-Bell412* dataset, the *Bell412* sequences comprised a flat taxiway and areas with buildings, highways, and runways, while the *Lighthouse* sequence, which was collected during winter, included ocean waves and snowy rocks. This variation demonstrates the proposed method's ability to handle different environmental conditions, including seasonal changes, effectively.

Furthermore, the LZ intersection of union (IoU) scores of *Holyrood-Paradise* and *Lighthouse-Bell412* datasets are 0.985 and 0.978 respectively. The non-LZ IoU values are 0.978 for *Holyrood-Paradise* and 0.890 for *Lighthouse-Bell412* datasets, demonstrating class-balanced results for both the landing zones and Non-landing zones.

Additionally, the accuracy of the proposed method is also noteworthy. As presented in Table V, the accuracy for *Holyrood-Paradise* and *Lighthouse-Bell412* datasets are 0.991 and 0.981, respectively. These results further confirm the observations given by the mIoU values, highlighting the effectiveness of this method for data from low-resolution LiDAR sensors.

### C. Quantitative Results of the Proposed Method on Non-Repetitive Scan LiDAR Data

The evaluation of the proposed LZDM on non-repetitive scan LiDAR data is detailed in Table IV and Table V. For the *HKairport-AMtown* dataset, the mIoU achieved is 0.955. The LZ IoU value for the *HKairport-AMtown* dataset is 0.930, which reflects the method's ability to identify potential landing zones even in the challenging conditions presented by non-repetitive scan patterns. This high LZ IoU score signifies that the proposed method can effectively handle variations and inconsistencies in the LiDAR data, maintaining accuracy in LZ detection. Furthermore, the non-LZ IoU value for the *HKairport-AMtown* dataset is 0.979, which indicates the method's ability to accurately distinguish non-landing zones. This high score minimizes the occurrence of false positives, ensuring that the LZ detection process remains reliable.

The accuracy of the proposed method for the *HKairport-AMtown* dataset is 0.984, demonstrating its effectiveness and efficiency in the context of non-repetitive scan LiDAR data.

### D. Qualitative Results of the Proposed Method on Low-Resolution LiDAR Data

Fig. 6 and 7 present a visual evaluation of the proposed LZDM applied to low-resolution LiDAR data. Each figure contains sample results for the *Holyrood-Paradise* and *Bell412-Lighthouse* datasets. Each figure is organized to show the aerial view, ground truth, and prediction results sequentially from left to right for each aerial data sequence. In

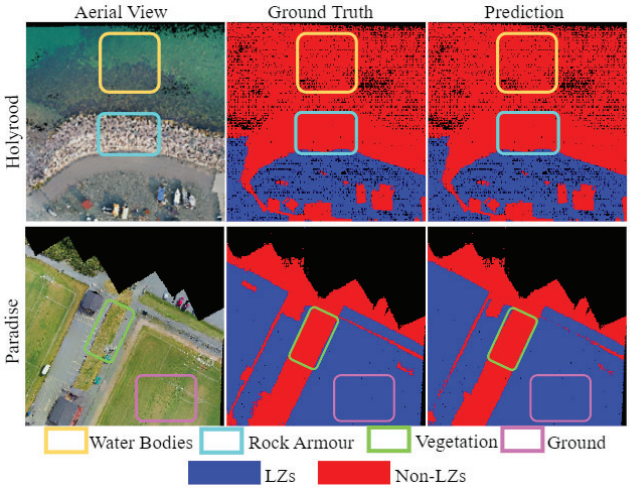


Fig. 6. Illustration of the qualitative results on Holyrood and Paradise data sequences.

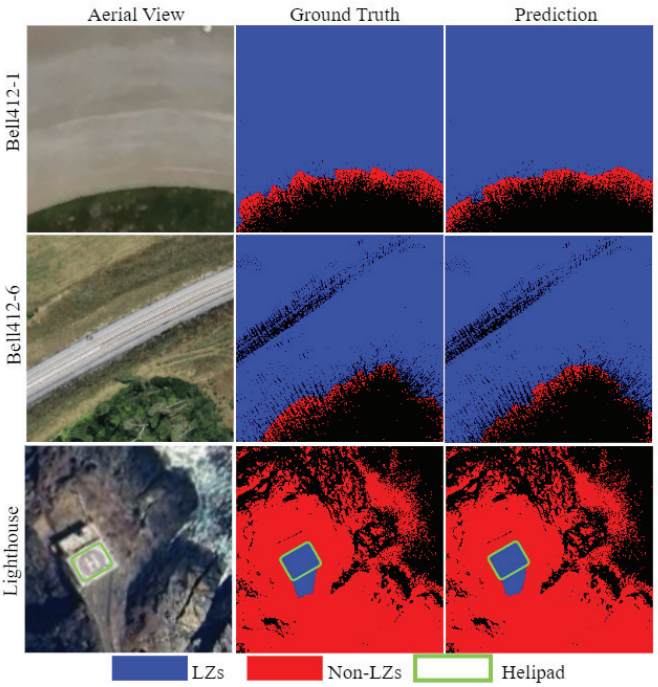


Fig. 7. Illustration of the qualitative results on Bell412-1, Bell412-6 and Lighthouse data sequences.

the figures, non-landing zones are indicated in red, whereas landing zones are shown in blue, and black areas represent regions with no points facilitating a clear comparison between the ground truth and the predicted landing zones.

As shown in the first row of Fig. 6, the proposed method is effective in identifying potential landing zones and non-landing zones. Specifically, the water bodies, denoted by the yellow bounding box, are correctly detected as non-landing zones. Additionally, the rock armour denoted in the aqua color bounding box is accurately detected as a non-landing zones, similar to the ground truth. For the *Paradise* dataset, vegetation areas, marked with light green bounding boxes, are correctly identified as non-landing zones. Furthermore, the

IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.

TABLE VI
VALIDATION RESULTS FOR THE PROPOSED LZDM

| Aerial Datasets | Avg. Cropping Time (ms) | Optimized Avg. Inference Time (ms) | Optimized Avg. Runtime (ms) | Optimized Throughput (pts/s) | Un-optimized Processing Rate (Hz) | Optimized Processing Rate (Hz) |
|---|---|---|---|---|---|---|
| Holyrood-Paradise | N/A | 93.3 | 93.3 | 5425014.1 | 4.3 | 10.718 |
| Lighthouse-Bell412 | 0.1929 | 95.1 | 95.2929 | 10159825.1 | 3.0 | 10.494 |
| HKairport-AMtown | 0.1925 | 95.7 | 95.8925 | 13431409.3 | 2.3 | 10.428 |

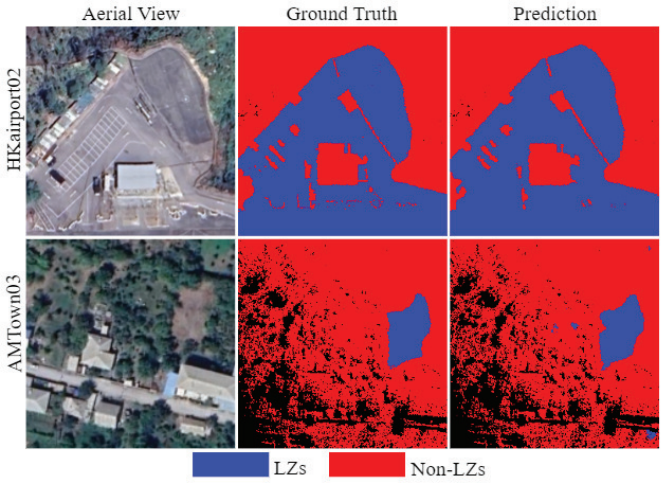


Fig. 8. Illustration of the qualitative results on HKairport02 and AMTown03 data sequences.

football ground, indicated by pink-colored bounding boxes, is accurately identified as an LZ, with the prediction results closely matching the ground truth.

Fig. 7 demonstrates the qualitative results of the proposed LZDM on the *Bell412-1*, *Bell412-6*, and *Lighthouse* dataset sequences. For the *Lighthouse* sequence, the proposed method accurately detects the helipad, denoted by the green bounding box, as a LZ. The precise classification of these areas is essential for the safe operation of VTOL vehicles, particularly in environments with clearly defined landing zones such as helipads. Moreover, the predictions closely align with the ground truths in the *Bell412* data sequences as well.

These qualitative results on low-resolution data demonstrate the proposed method's ability to accurately classify different types of terrain, ensuring that unsuitable areas for landing are correctly marked as non-landing zones, while suitable areas are identified as landing zones.

### *E. Qualitative Results of the Proposed Method on Non-Repetitive Scan LiDAR Data*

Fig. 8 presents the qualitative results of the proposed LZDM on non-repetitive scan LiDAR data for the *HKairport02* and *AMTown03* aerial data sequences. The figure displays aerial view, ground truth, and prediction results sequentially from left to right for each sequence. Moreover, non-landing zones are depicted in red, landing zones are depicted in blue, and areas with no points are shown in black.

The proposed method accurately differentiates between landing zones and non-landing zones in both sequences similar to the ground truths. These results demonstrate that the proposed method is not only suitable for low-resolution LiDAR data but also performs well with non-repetitive scan LiDAR data. These qualitative results further reinforce the quantitative findings.

### *F. Inference Speeds and Processing Rates of the Proposed Method for Low-Resolution and Non-Repetitive Scan LiDAR Data on Onboard Inference Hardware*

Table VI summarizes the performance metrics, including the average cropping time, optimized average inference time, optimized average runtime, optimized throughput, and un-optimized and optimized processing rates, for the three aerial datasets used in this study.

As shown in Table VI, the average inference time for *Holyrood-Paradise* dataset is 93.1 ms, while for the *Lighthouse-Bell412* dataset is 95.1 ms, and for the *HKairport-AMtown* dataset, it is 95.7 ms. As it is evident from Table VI, the average runtime is closer to the inference time for each dataset because the cropping times are extremely small, making their impact on the overall runtime negligible. It is also important to note that for the *Holyrood-Paradise* dataset, the cropping time is not applicable since an existing point cloud map from [7] was used for the analysis.

The *Holyrood-Paradise* dataset has a throughput of 5425014.1 pts/s, the *Lighthouse-Bell412* dataset has 10159825.1 pts/s, and the *HKairport-AMtown* dataset has 13431409.3 pts/s. These high throughput values are advantageous as they signify the system's capability to manage and process large volumes of data efficiently, which is crucial for real-time applications.

Further, the proposed method demonstrates high optimized processing rates, achieving an average of 10.718 Hz on the *Holyrood-Paradise* dataset, 10.494 Hz on the *Lighthouse-Bell412* dataset, and 10.428 Hz on the *HKairport-AMtown* dataset. Table VI also includes the un-optimized processing rates before applying PyTorch TensorRT optimization.

These results demonstrate that the proposed method is suitable for real-time, accurate LZ detection for VTOL vehicles. Moreover, it is evident that the proposed LZDM addresses the challenge of balancing speed and accuracy, overcoming the limitations of existing projection-based PCSS methods that often trade off accuracy for faster processing times, particularly in the context of low-resolution and non-repetitive scan LiDAR sensors. These results further indicate that the

**IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.**

proposed method is not only faster but also more accurate and suitable for diverse LiDAR data types.

## V. Conclusion

This study presented an AI-based LZDM for VTOL vehicles, employing a projection-based PCSS method combined with LiDAR point cloud accumulation. The proposed method was evaluated using three distinct groups of seven aerial data sequences collected from a DJI M600 hexacopter drone, an RTK 300 quadcopter drone, and a Bell412 helicopter, encompassing both low-resolution and non-repetitive scan LiDAR data. The method successfully addressed the accuracy limitations of existing projection-based methods for LiDAR data from low-resolution LiDAR sensors and non-repetitive scan LiDAR sensors without compromising faster runtime. Both qualitative and quantitative results reinforce the success of the proposed method. Quantitative results show that the method achieves a mIoU of 0.981 for the *Holyrood-Paradise* dataset, 0.934 for the *Lighthouse-Bell412* dataset, and 0.955 for the *HKairport-AMtown* dataset, with LZ IoU values ranging from 0.930 to 0.985 across all three datasets. Additionally, the optimized average processing rate per cropped point cloud segment is higher than 10 Hz across all three datasets, ensuring real-time feasibility and making it suitable for accurate LZ detection in VTOL. Moreover, the high throughput values, ranging from 5 *million* to 13 *million* pts/s, ensure that the proposed LZDM can handle large volumes of data efficiently. This further validates the proposed method's applicability in real-world. As future work, LiDAR-camera sensor fusion will be explored for LZ detection to leverage the complementary strengths of both sensors. Additionally, the point cloud map generation method will be selected in real-time based on its quality, determining the most suitable algorithm from existing mapping algorithms, as these algorithms may not consistently perform well throughout the entire flight. This optimized mapping algorithm will then be employed for detecting potential landing zones, aiming to further enhance the accuracy and efficiency of the LZ detection process.

## References


[1] Y. Zhou, H. Zhao, and Y. Liu, "An evaluative review of the VTOL technologies for unmanned and manned aerial vehicles," *Comput. Commun.*, vol. 149, pp. 356–369, Jan. 2020.

[2] J. Lin, Y. Wang, Z. Miao, H. Wang, and R. Fierro, "Robust image-based landing control of a quadrotor on an unpredictable moving vehicle using circle features," *IEEE Trans. Autom. Sci. Eng.*, vol. 20, no. 2, pp. 1429–1440, Apr. 2023.

[3] S. Scherer, L. Chamberlain, and S. Singh, "Autonomous landing at unprepared sites by a full-scale helicopter," *Robot. Auto. Syst.*, vol. 60, no. 12, pp. 1545–1562, Dec. 2012.

[4] L. Yan, J. Qi, M. Wang, C. Wu, and J. Xin, "A safe landing site selection method of UAVs based on LiDAR point clouds," in *Proc. 39th Chin. Control Conf. (CCC)*, Jul. 2020, pp. 6497–6502.

[5] O. G. Lorenzo, J. Martínez, D. L. Vilariño, T. F. Pena, J. C. Cabaleiro, and F. F. Rivera, "Landing sites detection using LiDAR data on many-core systems," *J. Supercomput.*, vol. 73, no. 1, pp. 557–575, Jan. 2017.

[6] M. Garg, A. Kumar, and P. B. Sujit, "Terrain-based landing site selection and path planning for fixed-wing UAVs," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Jun. 2015, pp. 246–251.

[7] N. M. Balasooriya, O. De Silva, A. Jayasiri, and G. K. I. Mann, "AI-based landing zone detection for vertical takeoff and land LiDAR localization and mapping pipelines," *Drone Syst. Appl.*, vol. 12, pp. 1–19, Jan. 2024.

[8] M. A. Kaljahi et al., "An automatic zone detection system for safe landing of UAVs," *Expert Syst. Appl.*, vol. 122, pp. 319–333, May 2019.

[9] V. R. Desaraju et al., "Vision-based landing site evaluation and informed optimal trajectory generation toward autonomous rooftop landing," *Auto. Robots*, vol. 39, no. 3, pp. 445–463, Oct. 2015.

[10] J. Singh, N. Adwani, H. Kandath, and K. M. Krishna, "RHFSafeUAV: Real-time heuristic framework for safe landing of UAVs in dynamic scenarios," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Jun. 2023, pp. 863–870.

[11] M. R. Lee, A. Nugroho, T.-T. Le, Bahrudin, and S. N. Bastida, "Landing area recognition using deep learning for unammaned aerial vehicles," in *Proc. Int. Conf. Adv. Robot. Intell. Syst. (ARIS)*, Aug. 2020, pp. 1–6.

[12] S. Lee and Y. Kwon, "Safe landing of drone using AI-based obstacle avoidance," *Int. J. Mech. Eng. Robot. Res.*, vol. 9, no. 11, pp. 1495–1501, 2020.

[13] K. Mukadam, A. Sinh, and R. Karani, "Detection of landing areas for unmanned aerial vehicles," in *Proc. Int. Conf. Comput. Commun. Control Autom. (ICCUBEA)*, Aug. 2016, pp. 1–5.

[14] S. Abdollahzadeh, P.-L. Proulx, M. S. Allili, and J.-F. Lapointe, "Safe landing zones detection for UAVs using deep regression," in *Proc. 19th Conf. Robots Vis. (CRV)*, May 2022, pp. 213–218.

[15] D. Dissanayaka, T. R. Wanasinghe, O. De Silva, A. Jayasiri, and G. K. I. Mann, "Review of navigation methods for UAV-based parcel delivery," *IEEE Trans. Autom. Sci. Eng.*, vol. 21, no. 1, pp. 1068–1082, Jan. 2024.

[16] D. Maturana and S. Scherer, "3D convolutional neural networks for landing zone detection from LiDAR," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 3471–3478.

[17] K. Zha and J. Yuan, "Landing site selection of detector based on 3D point cloud segmentation," in *Proc. Int. Conf. Mech., Aerosp. Automot. Eng.*, Dec. 2021, pp. 104–112.

[18] H. Cheng, X. Han, and G. Xiao, "CENet: Toward concise and efficient LiDAR semantic segmentation for autonomous driving," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Taipei, Taiwan, Jul. 2022, pp. 1–6.

[19] F. Liu, J. Shan, B. Xiong, and Z. Fang, "A real-time and multi-sensor-based landing area recognition system for UAVs," *Drones*, vol. 6, no. 5, p. 118, May 2022.

[20] J. Lim, M. Kim, H. Yoo, and J. Lee, "Autonomous multirotor UAV search and landing on safe spots based on combined semantic and depth information from an onboard camera and LiDAR," *IEEE/ASME Trans. Mechatronics*, vol. 29, no. 5, pp. 3960–3970, Oct. 2024.

[21] R. Li, S. Li, X. Chen, T. Ma, J. Gall, and J. Liang, "TFNet: Exploiting temporal cues for fast and accurate LiDAR semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2024, pp. 4547–4556.

[22] R. G. Thalagala, O. De Silva, A. Jayasiri, A. Gubbels, G. K. I. Mann, and R. G. Gosine, "MUN-FRL: A visual-inertial-LiDAR dataset for aerial autonomous navigation and mapping," *Int. J. Robot. Res.*, vol. 43, no. 12, pp. 1853–1866, 2024.

[23] H. Li et al., "MARS-LVIG dataset: A multi-sensor aerial robots SLAM dataset for LiDAR-visual-inertial-GNSS fusion," *Int. J. Robot. Res.*, vol. 43, no. 8, pp. 1114–1127, Jul. 2024.

[24] T.-M. Nguyen, S. Yuan, M. Cao, Y. Lyu, T. H. Nguyen, and L. Xie, "NTU VIRAL: A visual-inertial-ranging-LiDAR dataset, from an aerial vehicle viewpoint," *Int. J. Robot. Res.*, vol. 41, no. 3, pp. 270–280, Mar. 2022.

[25] B. Li, "On enhancing ground surface detection from sparse LiDAR point cloud," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 4524–4529.

[26] V. Valseca, J. Paneque, J. R. Martínez-de Dios, and A. Ollero, "Real-time LiDAR-based semantic classification for powerline inspection," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Jun. 2022, pp. 478–486.

[27] L. Chen, Y. Xiao, X. Yuan, Y. Zhang, and J. Zhu, "Robust autonomous landing of UAVs in non-cooperative environments based on comprehensive terrain understanding," *Sci. China Inf. Sci.*, vol. 65, no. 11, Nov. 2022, Art. no. 212202.

[28] D. V. Nam and K. Gon-Woo, "Solid-state LiDAR based-SLAM: A concise review and application," in *Proc. IEEE Int. Conf. Big Data Smart Comput. (BigComp)*, Jeju Island, South Korea, Jan. 2021, pp. 302–305.

[29] M. Cordts et al., "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3213–3223.

[30] J. Behley et al., "SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9297–9307.

**IEEE Transactions on Automation Science and Engineering (T-ASE) paper, presented at ICRA 2026, Vienna, Austria.**


[31] Y. Pan, B. Gao, J. Mei, S. Geng, C. Li, and H. Zhao, "SemanticPOSS: A point cloud dataset with large quantity of dynamic instances," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Oct. 2020, pp. 687–693.

[32] T. Hackel, N. Savinov, L. Ladicky, J. D. Wegner, K. Schindler, and M. Pollefeys, "Semantic3D.Net: A new large-scale point cloud classification benchmark," 2017, *arXiv:1704.03847*.

[33] N. Varney, V. K. Asari, and Q. Graehling, "DALES: A large-scale aerial LiDAR data set for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 186–187.

[34] I. Melekhov, A. Umashankar, H.-J. Kim, V. Serkov, and D. Argyle, "ECLAIR: A high-fidelity aerial LiDAR dataset for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2024, pp. 7627–7637.

[35] Y. Sun, X. Zhang, Q. Xin, and J. Huang, "Developing a multi-filter convolutional neural network for semantic segmentation using high-resolution aerial imagery and LiDAR data," *ISPRS J. Photogramm. Remote Sens.*, vol. 143, pp. 3–14, Sep. 2018.

[36] K. Zhang et al., "A dual attention neural network for airborne LiDAR point cloud semantic segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5704617.

[37] D. Maturana and S. Scherer, "VoxNet: A 3D convolutional neural network for real-time object recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 922–928.

[38] A. Boulch, "ConvPoint: Continuous convolutions for point cloud processing," *Comput. Graph.*, vol. 88, pp. 24–34, May 2020.

[39] H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, and L. J. Guibas, "KPConv: Flexible and deformable convolution for point clouds," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6411–6420.

[40] H. Zhou et al., "Cylinder3D: An effective 3D framework for driving-scene LiDAR semantic segmentation," 2020, *arXiv:2008.01550*.

[41] T. Cortinhal, G. Tzelepis, and E. E. Aksoy, "SalsaNext: Fast, uncertainty-aware semantic segmentation of LiDAR point clouds," in *Proc. Int. Symp. Visual Comput.* San Diego, CA, USA: Springer, Oct. 2020, pp. 207–222.

[42] Y. Zhang et al., "PolarNet: An improved grid representation for online LiDAR point clouds semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9601–9610.

[43] M. Kellner, B. Stahl, and A. Reiterer, "Fused projection-based point cloud segmentation," *Sensors*, vol. 22, no. 3, p. 1139, Feb. 2022.

[44] K. Chen et al., "MVLidarNet: Real-time multi-class scene understanding for autonomous driving using multiple views," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Mar. 2020, pp. 2288–2294.

[45] A. Jhaldiyal and N. Chaudhary, "Semantic segmentation of 3D LiDAR data using deep learning: A review of projection-based methods," *Appl. Intell.*, vol. 53, no. 6, pp. 6844–6855, Jul. 2022.

[46] J. Zhang and S. Singh, "Visual-LiDAR odometry and mapping: Low-drift, robust, and fast," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 2174–2181.

[47] J. Zhang and S. Singh, "LOAM: LiDAR odometry and mapping in real-time," in *Proc. Robot., Sci. Syst. Conf.*, Jul. 2014, pp. 1–9.

[48] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO2: Fast direct LiDAR-inertial odometry," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, Aug. 2022.

[49] Alliance Canada.(2024). *Advanced Research Computing—National Services Clusters*. Accessed: Aug. 12, 2024. [Online]. Available: https://alliancecan.ca/en/services/advanced-research-computing/national-services/clusters

[50] NVIDIA.(2024). *NVIDIA Jetson AGX Xavier*. Accessed: Jun. 14, 2024. [Online]. Available: https://www.nvidia.com/en-sg/autonomous-machines/embedded-systems/jetson-agx-xavier/s

[51] PyTorch.(2024). *Pytorch Tensorrt*. Accessed: Jun. 14, 2024. [Online]. Available: https://pytorch.org/TensorRT/

[52] Y. Zhao, L. Bai, and X. Huang, "FIDNet: LiDAR point cloud semantic segmentation with fully interpolation decoding," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2021, pp. 4453–4458.


**Nirasha Herath** (Graduate Student Member, IEEE) received the B.Sc. (Hons.) degree in mechanical engineering (mechatronics specialization) from the Sri Lanka Institute of Information Technology, Malabe, Sri Lanka, in 2019. She is currently pursuing the Ph.D. degree in mechanical engineering with the Department of Mechanical and Mechatronics Engineering, Memorial University of Newfoundland, St. John's, NL, Canada. She is with the Intelligent Systems Laboratory, Memorial University of Newfoundland. Her research interests include robotics, with a focus on deep learning and computer vision techniques for autonomous navigation applications.

**Oscar De Silva** (Senior Member, IEEE) received the B.Sc. degree (Hons.) in mechanical engineering from the University of Moratuwa, Sri Lanka, in 2009, and the Ph.D. degree from the Memorial University of Newfoundland, St. John's, NL, Canada, in 2015. Following post-doctoral work with the ABS Harsh Environment Technology Center, St. John's, he joined the Memorial University of Newfoundland, as a Faculty Member, in 2016. He is currently an Associate Professor at the Faculty of Engineering and Applied Science, Memorial University of Newfoundland. His current research interests include the design of sensors and state estimators to support industrial surveying, inspection, and autonomous navigation applications.

**George K. I. Mann** received the B.Sc. degree (Hons.) in engineering from the University of Moratuwa, Sri Lanka, in 1984, the M.Sc. degree in computer integrated manufacture from Loughborough University, U.K., in 1989, and the Ph.D. degree from the Memorial University of Newfoundland, St. John's, NL, Canada, in 1999. From 1999 to 2000, he was a Research Engineer at C-CORE, Memorial University of Newfoundland. In 2001, he joined the Mechanical Engineering Department, Queen's University, Kingston, ON, Canada, as a Post-Doctoral Fellow. In 2002, he joined the Memorial University of Newfoundland, as a Faculty Member. From 2002 to 2007, he was the C-CORE Junior Chair of the Intelligent Systems Laboratory, Memorial University of Newfoundland, where he is currently a Professor of Mechanical Engineering with the Faculty of Engineering and Applied Science. His research interests include intelligent control, multi-robotic systems, autonomous navigation, and robotic control.

**Awantha Jayasiri** (Senior Member, IEEE) received the Ph.D. degree in robotics and control from the Memorial University of Newfoundland, St. John's, NL, Canada, in 2012. He was a Post-Doctoral Researcher at the Memorial University of Newfoundland and an Electrical Engineer in a local company. He joined as a Research Officer with the Flight Research Laboratory, National Research Council, Canada, in 2018. His research interests include real-time optimum control, deep learning, and computer vision in aerospace applications.