

# Monitoring Tree Intrusion Along Railway Networks Using LiDAR Technology

Coral Kim Man Yip<sup>1</sup>, LI Shu Yung<sup>1</sup>, SM Nazim Uddin Shoikot<sup>2</sup>, Muhammad Saad Shahid Anwel<sup>2</sup>

<sup>1</sup> MTR Corporation Limited, Hong Kong, People’s Republic of China

<sup>2</sup> Kodify Limited, Hong Kong, People’s Republic of China

## ABSTRACT

MTR Corporation has a well-established tree management regime for the monitoring of tree intrusion along its railway networks to ensure operational safety and efficiency. Trees near tracks pose a risk which can cause service disruptions and potential accidents, especially in open sections with road traffic interfaces. Effective monitoring helps ensure smooth operations. This paper discusses MTR’s strategy in tree management by employing groundbreaking technology in the Light Rail network.

**KEYWORDS** Vegetation Management, Gauging, Object Detection, Railway Safety

**CONTACT** Coral KM Yip ✉ coralkmyip@mtr.com.hk

Received 30 October 2025

## 1. Introduction

Tree management involves monitoring of live biometric parameters like growth and health. There are over 5,000 trees in the Light Rail Line network. Without proper management, tree intrusions can cause power outages, affecting railway services and road traffic.

MTR conducts regular cab rides and visual checks, and appoints external arborists to perform tree surveys. Pruning is carried out for high-risk intrusions, but inspecting large areas is time consuming and challenging. To enhance safety and maintenance efficiency, MTR has introduced a LiDAR-based tree intrusion monitoring system installed on a passenger train to perform daily monitoring in situ with its services and with direct coverage of the concerned track areas.

## 2. Overview of technology

The Tree Intrusion Monitoring System (TrIMS) is an advanced, integrated solution to enhance railway safety by monitoring vegetation encroachments. The system utilises LiDAR sensors to capture high-resolution 3D point cloud data and local positioning information, which then undergoes Simultaneous Localisation and Mapping (SLAM) to provide a detailed 3D map of the surroundings. Complementing this, video cameras offer visual monitoring, while AI algorithms process the data for tree detection, risk assessment, and growth trend analysis. Data collected is transmitted via a 5G network to an off-board server for further analysis and reporting.

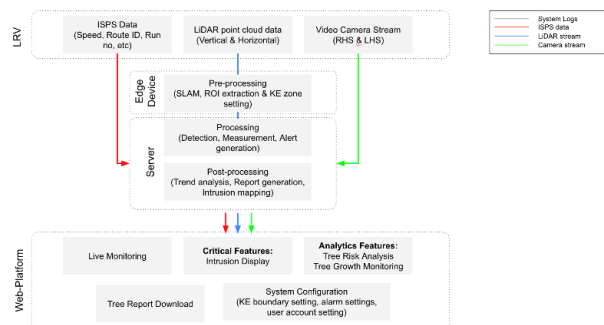


Figure 1. System Design of the TrIMS.

This section outlines the technologies adopted.

### 2.1. Light Detection and Ranging (LiDAR)

LiDAR is an advanced remote sensing technology that uses laser pulses to measure distances and generate precise 3D data of objects and environments (IBM, 2023).

Some notable applications of LiDAR include Autonomous Vehicles (AVs), Infrastructure Asset Management, and Forestry (Nature, 2021).

#### 2.1.1. LiDAR Data Output

There are two main outputs from the LiDAR sensors used in the project, namely point cloud data and High-Frequency Inertial Measurement Unit (IMU) Data. Combining these two types of data allows us to iteratively create a dense and precise 3D map of the surroundings.

#### 2.1.2. Advantage of using LiDAR compared to cameras

Traditional cameras are integrated into the system, which mainly serve to complement the results and report generation. AI analytics primarily depend on the 3D map

generated using LiDAR's data since cameras are highly dependent on the external weather conditions and lighting, like fog, rain, and typhoons (Photoniconline, 2024).

Without camera limitations, LiDAR is a more reliable choice for capturing high-precision data as LiDAR does not rely on ambient light or clear visibility to operate effectively (Hesai, 2024).

## 2.2. Global Positioning System (GPS)

The GPS antenna on the LRV captures georeferenced location signals and calculates the vehicle's precise latitude, longitude, and elevation. This real-time data is used to track the vehicle's movement along the track. It is particularly effective in the Light Rail network as it is a fully open section where the satellite signals are unobstructed.

## 2.3. Simultaneous Localisation and Mapping (SLAM)

SLAM involves simultaneously constructing or refining a map of an unknown environment while determining the location of an agent within it (MathWorks, 2024). This dual capability is vital for autonomous systems like TrIMS, which must navigate without preexisting maps. The challenge lies in accurately estimating the agent's position and building a reliable map from sensor data, all while mitigating the effects of noise and uncertainty inherent in real-world conditions.

SLAM algorithms typically operate through the following steps:

1. **Sensor Data Acquisition:** Collect data from various sensors, such as LiDAR, cameras, or radar, to perceive the environment.
2. **Feature Extraction:** Identify and extract distinct features or landmarks from the sensor data that can be reliably recognised and tracked over time.
3. **Data Association:** Match newly observed features with previously identified ones to maintain consistency in the map and the agent's location.
4. **State Estimation:** Use algorithms like the Extended Kalman Filter (EKF) or particle filters to estimate the agent's position and the map's features, updating these estimates as new data is acquired.
5. **Map Update:** Integrate the new information into the existing map, refining its accuracy and completeness.

### Implementation of SLAM in TrIMS

TrIMS employs a robust LiDAR-Inertial odometry method that fuses LiDAR and IMU data via a tightly coupled iterated Extended Kalman Filter (iEKF). By processing raw LiDAR point clouds directly, the system minimises intermediate processing steps, which leads to faster, more accurate mapping and better capture of subtle environmental details (Xu & Zhang, 2020).

## 2.4. Integration with the Robot Operating System (ROS)

The implementation of SLAM is facilitated through the Robot Operating System (ROS), open-source middleware which provides a flexible framework for integrating various sensors and algorithms, offering tools and libraries that support the development of complex robotic systems. Within TrIMS, ROS handles:

- **Sensor Data Acquisition:** Managing the collection and synchronisation of data from LiDAR and IMU sensors, ensuring coherent and timely data streams for processing.
- **Data Processing Pipelines:** Coordinating the flow of data through the FAST-LIO algorithm, from initial acquisition to the generation of localisation and mapping outputs.
- **System Modularity and Scalability:** Allowing for modular development, where components can be independently modified or upgraded, and facilitating scalability to accommodate additional sensors or processing capabilities as needed.

The use of ROS not only streamlines the development and integration process but also enhances the system's robustness and flexibility.

## 3. Application of technology in monitoring tree intrusion

The design of the system carefully considers the unique features of the railway environment and characteristics of the Light Rail Line. The setup of LiDAR and onboard cameras takes account of the extremely sharp lateral curve up to R20m in the Light Rail to ensure full capture of views under the challenging vehicle end throw effect. The data processing unit compensates for unwanted disturbances arising from vehicle vibrations as well as wind disturbances to ensure data accuracy.

### 3.1. Data Collection

TrIMS employs a combination of various sensors for the continuous collection of environmental data, including Point Cloud Data, videos, and positional information. Data is separated into sections, where one section refers to the data collected between two consecutive stations.

#### 3.1.1. LiDAR Arrangement

TrIMS uses **two LiDAR sensors** positioned strategically on top of the Light Rail Vehicle (LRV). These sensors are designed to maximise the field of view, allowing the system to capture detailed data on both the **tree crowns** and **tree trunks**. This is particularly important in the Light Rail network as the smallest curve can be up to R20m.

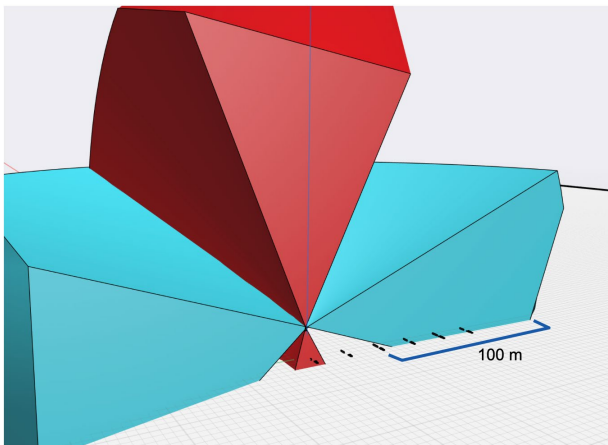


Figure 2. Field of View (FOV) of LiDAR (cyan: horizontal FOV; Magenta: vertical FOV).

1. **Top-Facing LiDAR Sensor:**  
The first LiDAR sensor is positioned facing upwards, focusing on capturing data of tree crowns and overhanging vegetation that allows the system to measure the height of trees and evaluate their proximity to the LRV, ensuring that branches or tall trees do not obstruct the train's path.
2. **Forward-Facing LiDAR Sensor:**  
The second LiDAR sensor is positioned facing forward, scanning ahead along the railway track. This sensor captures data about the tree trunks and other vegetation along the track, allowing the system to detect any intrusions that may occur at ground level or within the track's clearance zones.

**3.1.2. Video Data**

Two video cameras are installed at the front of the train, facing left and right, respectively. For every intrusion, corresponding videos are stored and reported for deeper analysis. Additionally, it is also used to stream live video from the train's operation for active monitoring.



Figure 3. Dual camera Field of View (FOV) – Left and Right.

**3.1.3. GPS Data**

By integrating GPS data with LiDAR, TrIMS can accurately localise detected vegetation and provide a detailed map of intrusions of the railway environment.

**3.2. Data Pre-processing**

The Data Pre-processing pipeline in TrIMS transforms raw sensor data into actionable intelligence for vegetation management. Following established practices, it includes stages of data acquisition, filtering, feature extraction, and data fusion, ensuring high-quality information for georeferenced maps and LRV localisation.

Integrating data from LiDAR, video cameras, and GPS systems creates a robust dataset. The initial step filters out noise, and the 3D LiDAR data is processed using SLAM algorithms to update the environmental map and determine the LRV's real-time position.

SLAM is employed to process data from LiDAR sensors, enabling the system to construct detailed 3D maps, track vegetation growth over time, and enhance localisation accuracy.

**3.3. SLAM and GPS integration for precise positioning**

The integration of SLAM with GPS ensures the accurate positioning of detected trees along railway tracks and georeferencing. The process involves:

1. **Timestamped SLAM Data Acquisition:**  
During SLAM processing, each LiDAR reading is recorded with an absolute timestamp and its corresponding local position. This information forms the basis for constructing a local trajectory of the sensor platform (hku-mars, 2020).
2. **Local Trajectory Construction:**  
The sequence of local positions is used to create a trajectory representing the motion of the sensor (e.g., the train) through the environment. This trajectory captures the relative movement in the local coordinate system.
3. **GPS Data Synchronisation:**  
GPS sensors provide GPS positions along with timestamps. For every local position in the trajectory, we match the corresponding GPS reading based on the timestamp, thereby aligning the local data with the global coordinate system.
4. **Georeferencing the SLAM Map:**  
By associating each local position with its corresponding absolute GPS coordinate, we transform the entire SLAM map from the local frame to a global reference frame. This step enables the determination of the absolute GPS coordinates for any point within the map.

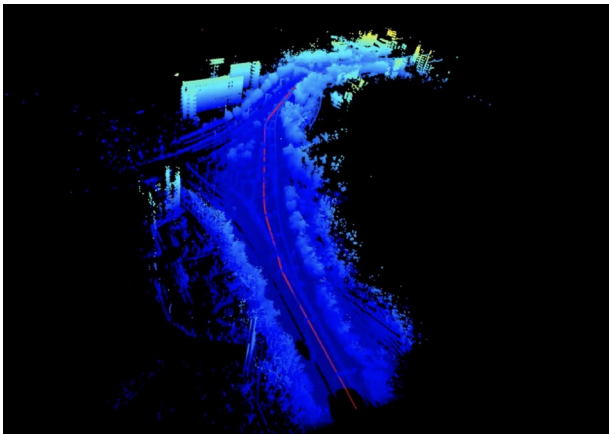


Figure 4. 3D Point Cloud Data output from the LiDAR sensors.

### 3.4. Establishing the ROI and KE Boundaries

#### 3.4.1. Defining the Region of Interest (ROI)

The Region of Interest (ROI) is a predefined area for the continuous monitoring of vegetation intrusion, typically covering the entire vegetation zone along the tracks and adjacent areas that could cause a risk to rail operations. The ROI can be adjusted dynamically to monitor vegetation near the LRV during operation.

The pre-defined monitoring zones are segmented based on the level of potential risk:

1. Area of Interest: Encompasses the full extent of the track and surrounding vegetation.
2. Area of Warning: Includes the KE-zones, which are critical zones around the vehicle and track where early warnings about potential vegetation intrusion are generated.
3. Area of Alert: Zones where any intrusion is immediately flagged for action.

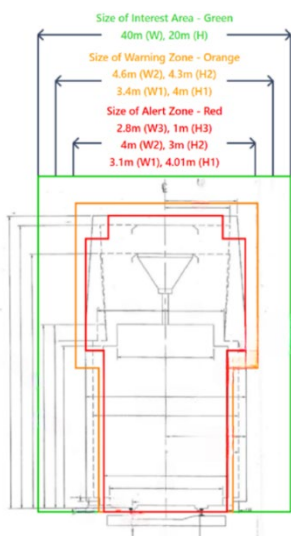


Figure 5. Pre-defined monitoring zone.

#### 3.4.2. Setting the Kinematic Envelope (KE) Boundaries

The Kinematic Envelope (KE) defines the spatial limits within which a Light Rail Vehicle (LRV) safely operates, accounting for train dynamics such as tilting, track irregularities, and vehicle sway. TrIMS uses parameters such as the LRV's dimensions, clearance needs, and train movement/vibration data from the IMU to set KE boundaries. These boundaries help accurately detect vegetation encroachment near the tracks, flagging any intrusion for further analysis.

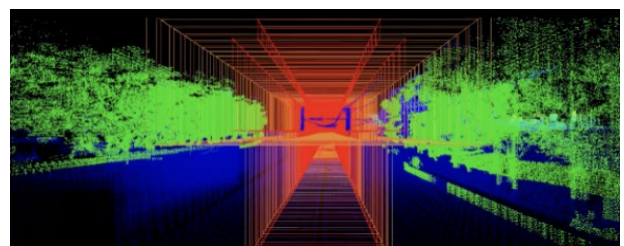
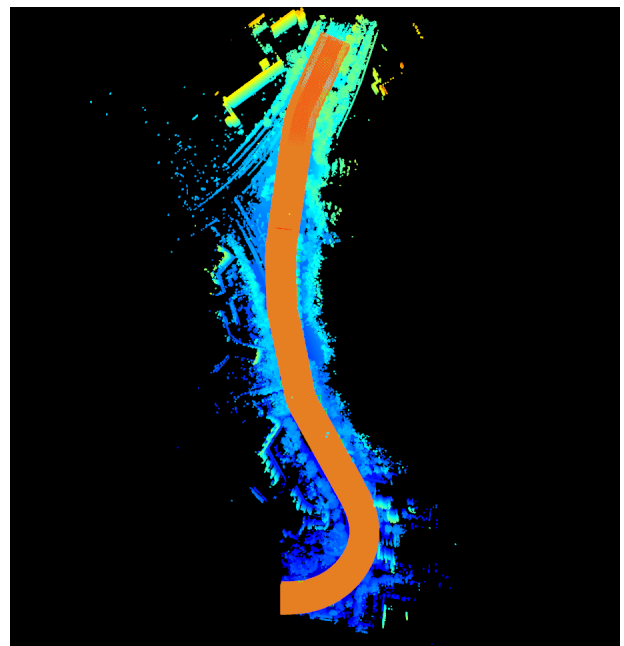


Figure 6. SLAM Map of a Light Rail section with applied KE zone in top view and front view.

The KE zones typically consist of multiple 3D prisms, which represent the vertical and horizontal areas around the LRV where vegetation must be monitored as illustrated in Figure 6 above.

#### 3.5. Tree Detection

The detection of trees from the 3D LiDAR data is achieved by applying advanced machine learning techniques, specifically leveraging PP-YOLOePlus-X, a state-of-the-art object detection model. The detection pipeline integrates several critical steps, starting with the

transformation of 3D point cloud data (PCD) into 2D raster images, followed by training a robust AI model to recognise and classify trees.

### 3.5.1. Approach to Tree Detection

The detection process begins by converting the 3D PCD collected from the LiDAR sensors into 2D rasters. This transformation simplifies the detection process, as 2D image-based object detection algorithms are well established and optimised for such tasks. The generated 2D images represent an aerial view of the environment, with treetops prominently visualised, allowing the model to focus on identifying vegetation within the defined Region of Interest (ROI).

1. Raster Generation:

The first step in the AI detection pipeline involves the generation of 2D rasters from the 3D point cloud data, representing bird's-eye-view (BEV) intensity maps of the trees. These projections are essential as they capture the spatial distribution of trees along the railway track, forming the basis for subsequent training and detection tasks (Forest3D, 2020).

2. Labelling and Augmentation:

The rasters are then manually annotated using LabelImg, an open-source image annotation tool. Each tree canopy is enclosed with a bounding box that captures the full spatial extent of the canopy in the BEV image. This manual labelling ensures precise delineation of individual trees and consistent ground truth across the dataset.

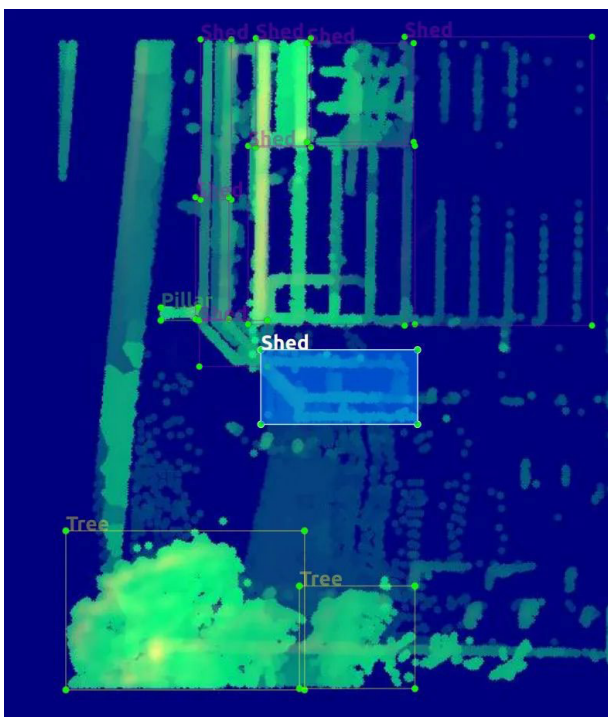


Figure 7. Annotation of a tree canopy on a 2D raster image.

The labelled data undergoes conventional augmentation such as rotations, scaling, and flips, by using Roboflow's data augmentation pipeline. This augmentation increases the diversity of the training data, allowing the model to generalise better and handle variations in tree shapes, sizes, and environmental conditions, helping the model generalise better to real-world conditions.

3. Training the Model:

The augmented rasters are used to train the detection model. The model selected for this task is PP-YOLOePlus-X, a variant of the popular YOLO (You Only Look Once) model. PP-YOLOePlus-X is well suited for detecting individual trees within complex scenes, as it can efficiently process large-scale datasets and achieve high accuracy in object localisation. The model is trained using the PaddlePaddle/PaddleDetection framework (PaddleDetection, 2021), which is equipped with cutting-edge capabilities for object detection and segmentation tasks.

4. Tree Detection:

After the model is trained, it is applied to detect trees in new 2D raster images generated by the system. The AI model identifies tree locations and classifies and distinguishes them from other environmental objects. These detections are then projected back into the 3D space, providing the system with the 3D coordinates of each detected tree along the railway track.

Intrusion zones are not annotated directly. Instead, they are constructed programmatically by combining continuous 3D bounding boxes aligned along the odometry path of the train, forming volumetric regions that represent potential clearance or intrusion areas relative to the rail trajectory.

5. Post-Processing:

Post-processing steps are applied to refine the detections. This includes filtering out false positives (e.g., non-vegetation objects such as poles or vehicles) and verifying the tree classifications. The post-processing stage ensures that only relevant vegetation is flagged for further analysis and is mapped accurately in the context of the overall railway environment.

### 3.5.2. Rationale for the Chosen Approach

This approach was selected due to several key advantages:

- 3D to 2D Transformation: Converting 3D point clouds to 2D rasters simplifies the tree detection task by leveraging existing, well-established 2D object detection techniques. These models are highly optimised for speed and accuracy, allowing for real-time detection.

- **Efficiency and Precision:** The use of PP-YOLOePlus-X ensures that the model delivers high performance in identifying individual trees, even in dense and cluttered environments. The model's architecture allows it to efficiently handle complex input data while maintaining high accuracy levels.

### Output of the AI Detection Process

The primary output of the AI detection process is a set of 3D coordinates representing the locations of the detected trees. These coordinates are mapped along the track, providing information on tree positions relative to the Light Rail Vehicle (LRV). Additionally, each detection is classified (e.g., tree, pole, shed), enabling the system to differentiate between vegetation and other objects. The output also includes confidence scores for each detection, which reflect the certainty with which the model identifies the trees.

### 3.6. Intrusion Detection

The intrusion detection process leverages the SLAM-generated 3D map and AI-driven tree detection to identify trees encroaching on predefined KE zones along the railway.

The main steps involved are related to tree detection and localisation in the SLAM map; checking for KE zone intersection, and extraction of the relevant tree metrics before the information is synthesised with the Global Navigation Satellite System (GNSS) data and uploaded to the MMI. An intrusion report will be generated capturing key information such as location, tree height, and distance from track to user for actionable insights.

### 3.7. Risk Analysis

The risk analysis module assesses the potential hazard posed by each tree near the railway tracks by assigning a risk level based on tree characteristics and spatial relationships. The steps are as follows:

1. **Reference Point Identification:**  
Determining the closest point on the track to the intrusion's geohash, which serves as the centre for the region of interest (ROI).
2. **Defining the Region of Interest (ROI):**  
A square ROI with a 40-metre diameter is established around the reference point to capture all potentially relevant trees.
3. **Tree Detection within ROI:**  
All trees located within the ROI are identified from the SLAM map. For each tree, the system measures its height and calculates its horizontal distance from the track.

4. **Risk Categorisation and Scoring:**  
Trees are evaluated and assigned a risk score as follows:
  - **Category 1 (Score: 3):** Tree height exceeds their lateral distance from track, posing a high risk.
  - **Category 2 (Score: 2):** Trees on a slope that do not meet the first condition, indicating a moderate risk.
  - **Category 3 (Score: 1):** All other trees, representing a lower risk level.
5. **Average Risk Calculation and Normalisation:**  
The total risk score from all trees in the ROI is divided by the number of trees to derive an average risk score. This average is then normalised by dividing by the maximum possible score (3), resulting in a value between 0 and 1. The final risk level is computed by scaling this normalised value to a predefined range using the following formula:

$$\begin{aligned} \text{Final Risk Level} \\ = (\text{Normalized Score}) \times (\text{MAX}_{\text{RISK\_LEVEL}} - \text{MIN}_{\text{RISK\_LEVEL}}) + \text{MIN}_{\text{RISK\_LEVEL}} \end{aligned} \quad (1)$$

This final risk level provides a quantifiable measure of the threat posed by the trees within the ROI.

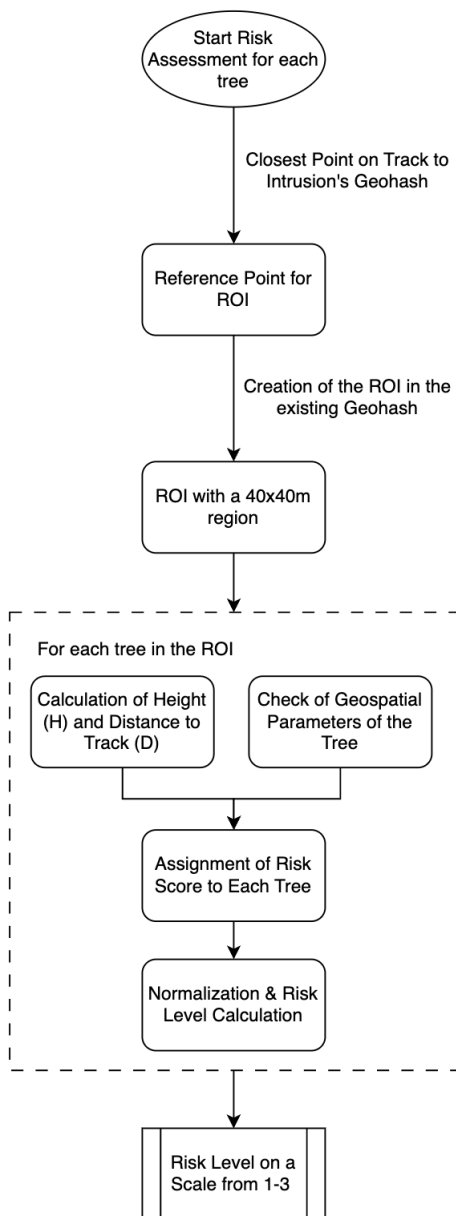


Figure 8. Process Flow for Risk Level Analysis.

### 3.8. Tree Growth Prediction

Monitoring vegetation growth between consecutive stations is vital for proactive maintenance. The growth trend prediction algorithm operates as follows:

1. **Data Sufficiency Check:**  
The system verifies that each section contains data for at least five trees to ensure statistically robust averages.
2. **Metric Calculation for Individual Trees:**  
For each tree in the section, the key metrics such as height and volume are calculated. The volume is approximated by voxelising the tree's point cloud data: the tree's 3D points are segmented into uniformly sized voxels, and the total volume

is estimated by multiplying the number of voxels occupied by the tree by the known voxel size. These individual measurements serve as the foundation for the section-wide analysis.

3. **Computation of Section Averages:**  
Using the individual metrics, the system calculates the average height and proximity of trees for the entire section, providing a snapshot of the current vegetation state.
4. **Monthly Averaging and Data Handling:**  
Monthly averages are computed for sections with sufficient data. If a section lacks adequate data, the system checks for historical values: if previous data exists, it assumes no growth for the current month; otherwise, the monthly values are recorded as null.
5. **Growth Determination:**  
The system then compares the current month's averages with those of the previous month to determine growth trends. This includes calculating the differences in average height and volume, which reflect how the vegetation is evolving over time.

The detected tree characteristic data is visualised to enable practicable maintenance monitoring and planning.

### 3.9. Visualisation and Growth Index Grouping

Tree Growth Visualisation integrates key parameters, including tree height, volume, and distance to the track, and further manipulated into two defined indexes.

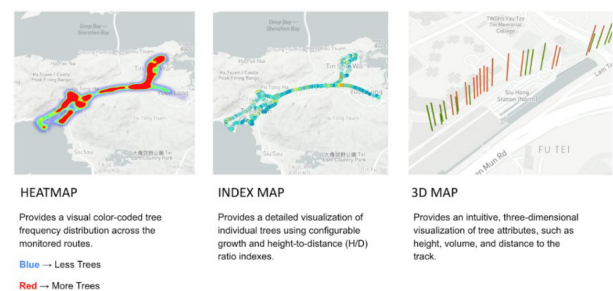


Figure 9. Overview of Visualisation Maps of Tree Growth in Various Forms.

A height-to-distance (H/D) ratio is defined as the ratio of tree height to the lateral distance from the track, which is an intuitive representative reflection of the potential hazard level.

The Tree Growth Index is defined and utilises an algorithm that employs configurable weights to specific tree characteristics, and the tree parameters are classified into groups for calculation:

Table 1. Tree characteristic parameter groupings.

Value assigned to Group	Height (m)	Volume (m <sup>3</sup> )	Distance to Track (m)
1	<3	<10	>9
2	3-6	10-20	7-9
3	6-9	20-30	5-7
4	9-12	30-40	4-5
5	12-15	40-50	3-4
6	15-18	50-60	2-3
7	18-21	60-70	1.5-2
8	21-24	70-80	1-1.5
9	24-27	80-90	0.5-1
10	>27	>90	0-0.5

The growth index formula is shown below:

$$Growth\ Index = (W_{height} \times height) + (W_{volume} \times volume) + (W_{distance} \times distance) \quad (2)$$

By aggregating and analysing historical data, TrIMS generates a time-series visualisation of vegetation encroachment, which facilitates a data-driven decision-making process and allows the user to pinpoint trees with a higher index to prioritise tree maintenance activities.

### 3.10. Accuracy and Results

In this project, the accuracy of the AI model's intrusion detection capability is evaluated. Two key metrics, namely detection rate and false detection rate, were considered.

To assess these metrics, five representative sections were selected—each corresponding to the map between two consecutive stations—with a balance of trees and non-tree objects (e.g., poles, sheds). For each section, the process was as follows:

1. Tree Detection within the Region of Interest (ROI):  
Trees were first detected across the entire SLAM map, and then only those located within the predefined ROI were considered for further evaluation.

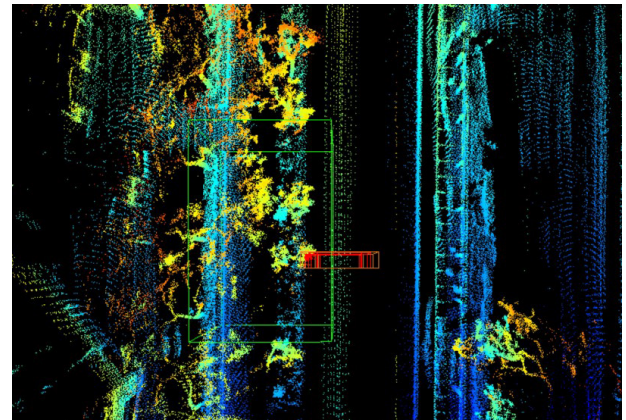
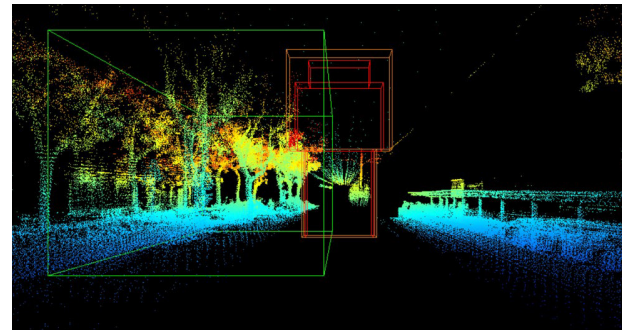


Figure 10. Illustration of intrusion recorded on MMI.

2. Ground Truth Comparison:  
True Positives (TP), False Positives (FP), and False Negatives (FN) were manually identified by comparing the detection results with the ground truth.
3. Relevant metrics were calculated as follows:
  - Recall (Detection Rate):  $Recall = \frac{45}{45 + 7 + 89} \quad (3)$
  - Precision:  $Precision = \frac{45}{45 + 7 + 85} \quad (4)$
  - False Detection Rate:  
 $False\ Detection\ Rate = 1 - Precision \quad (5)$
  - F1 Score:  $F1\ Score = \frac{2 \times (5 + 3'0' + 27 + 31,)}{5 + 3'0' + 27 + 31,} \quad (6)$

For the five sections evaluated, the following totals were recorded:

Table 2. AI Model Statistics.

Parameter	Value
True Positive	691
False Positive	18
False Negative	26
Precision	0.97
Recall	0.96

This resulted in a precision of 0.97 and a recall (detection rate) of 0.96 (and F1 score of 0.96), corresponding to a 96% detection rate and a 3% false detection rate—both within the desired thresholds, as illustrated in Table 2.

## 4. Challenges and recognition

### 4.1. Technical Challenges

Despite the significant benefits, challenges remain in vegetation monitoring and intrusion detection along railway tracks. Distinguishing trees from other objects in point cloud data is difficult in cluttered environments, requiring the ongoing development of advanced AI models. Additionally, processing large datasets in real time is challenging, as algorithms must balance efficiency with accuracy in detecting tree intrusions. Integrating LiDAR with GNSS improves georeferencing, but achieving precise real-time synchronisation can introduce errors that affect the detection accuracy.

### 4.2. Patent and Novelty of the system

The use of LiDAR equipped with AI and machine learning technology in vegetation management is unique, and this method is patented under the Hong Kong Patents Registry (Publication No. 30104716). The project customised the equipment setup and data-processing algorithm to fit the Light Rail's unique characteristics, such as extremely sharp curves, mix of urban infrastructure in situ with trees, and meticulous design of indexes to intuitively reflect the tree conditions in the network.

## 5. Conclusion

Integrating LiDAR with AI and machine learning enhances tree management along railway networks, strengthening safety and operational efficiency while supporting sustainable development goals. This project overcomes the complex railway environment and geographical challenges in the Light Rail network, achieving a 96% intrusion detection rate with only 3% false detection, thereby optimising tree monitoring efficiency and effectiveness.

This technology is highly scalable to other MTR networks or similar railway systems, balancing infrastructure maintenance with ecosystem conservation.

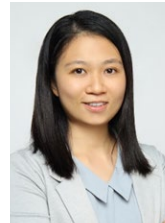
## Acknowledgements

Gratitude is expressed to the anonymous reviewers for their valuable comments.

## Financial interest disclosure statement

This work did not accept any funding or support from any funding agency or grant-awarding bodies.

## Notes on contributors



**Ms Coral KM Yip** received her BEng in Mechanical Engineering and BBA in General Business Management at the Hong Kong University of Science and Technology. She is a Chartered Engineer of the Institute of Mechanical Engineers. Currently she is Senior Design Engineer – Smart Maintenance in MTR, assisting the

development of various smart systems to uplift maintenance efficiency.



**Mr Li Shu Yung** received his BEng in Electrical Engineering and MSc in Building Service Engineering at the Polytechnic University. He is a Chartered Engineer of the Institute of Engineering and Technology. Currently he is Senior Design Engineer – Rolling Stock Engineering in MTR, assisting in the

design management in rolling stock and innovation projects with an aim for uplifting the performances of rolling stocks and interface systems.



**Mr SM Nazim Uddin Shoikot** holds a BEng in Computer Science from The University of Hong Kong. With over 5 years of experience in Artificial Intelligence and Software Development, he works as an AI Engineer specialising in the intersection of Computer Vision and Generative AI. His professional focus

is on leveraging the power of Machine Learning to develop practical AI solutions.



**Mr Muhammad Saad Shahid Anwel** holds a BEng in Computer Engineering from The University of Hong Kong and is Co-Founder and Chief Technology Officer at Kodifly Limited. With over 5 years of a diverse range of experience in Machine Learning, Robotics, and LiDAR Technology, Saad specialises in

leveraging AI and geospatial data to transform transport infrastructure monitoring.

## Reference

- [1] Nature. (2021). *Nanophotonics for light detection and ranging technology*. *Nature Photonics*, 15(5), 291–302. doi: 10.1038/s41565-021-00895-3
- [2] W. Xu and F. Zhang, “FAST-LIO: A fast, robust LiDAR-inertial odometry package by tightly-coupled iterated Kalman filter,” *arXiv preprint arXiv:2010.08196*, Oct. 2020. [Online] Available: <https://arxiv.org/abs/2010.08196> (Accessed: Jan 1, 2025).
- [3] hku-mars, “FAST\_LIO,” GitHub repository, 2020. [Online] Available: [https://github.com/hku-mars/FAST\\_LIO](https://github.com/hku-mars/FAST_LIO) (Accessed: Jan 1, 2025).
- [4] Forest3D Github repository, 2020. [Online] Available: [https://github.com/lloydwindrim/forest\\_3d\\_app](https://github.com/lloydwindrim/forest_3d_app) (Accessed: Jan 10, 2025)
- [5] PaddleDetection GitHub repository, 2021. [Online] Available: <https://github.com/PaddlePaddle/PaddleDetection> (Accessed: Jan 10, 2025).
- [6] IBM. (2023). *What is LiDAR?* [Online] Available at: <https://www.ibm.com/think/topics/lidar> (Accessed: Jan 10, 2025).
- [7] Photonicsonline. (2024). *The Great Debate LiDAR Or Camera-Based Systems For AVs*. [Online] Available at: <https://www.photonicsonline.com/doc/the-great-debate-lidar-or-camera-based-systems-for-avs-0001> (Accessed: Jan 10, 2025).
- [8] Hesai. (2024). *Benefits of Lidar vs. Cameras in Self-Driving Cars*. [Online] Available at: <https://www.hesaitech.com/benefits-of-lidar-vs-cameras-in-self-driving-cars/> (Accessed: Jan 10, 2025).
- [9] MathWorks. (2024). *What Is SLAM (Simultaneous Localization and Mapping)?* [Online] Available at: <https://www.mathworks.com/discovery/slam.html> (Accessed: Jan 10, 2025).