Proceedings of the Sixth International Conference on
Railway Technology: Research, Development and Maintenance
Edited by: J. Pombo
Civil-Comp Conferences, Volume 7, Paper 21.2
Civil-Comp Press, Edinburgh, United Kingdom, 2024
ISSN: 2753-3239, doi: 10.4203/ccc.7.21.2

Tram Driving Path Identification Systems for Driverless Tram Operations

S. Hovanotayan and K. Nakano

**Institute of Industrial Science, The University of Tokyo
Japan**

Abstract

In this research, the tram driving path identification system for driverless trams is proposed. The primary purpose of the system is to identify the path in front of a tram where the tram will occupy. With this system, it is highly expected that it will substantially assist with preventing collisions between trams and frontal obstacles, and other relevant hazardous incidents. For the proposed system, it consists of two main modules: the rail detection module and the boundary generating module. In the study of rail detection module, the comparative analysis of different types of rail detectors, including traditional methods and learning-based methods, is performed to investigate the most suitable rail detectors. The comparative results reveal that the deep learning-based models are more accurate and reliable. Further, the boundary generating module is designed. Subsequently, the preliminary test of the proposed identification system based on the rail detection and designed boundary generating modules is carried out. According to the preliminary result, the proposed identification system successfully managed to perform its designated task very well; hence, it can be said that this proposed tram driving path identification system is promising for driverless tram operations.

Keywords: trams, driverless, driving path, rail detection, deep learning, boundary.

1 Introduction

In recent years, due to the rapid advancement of artificial intelligence as well as many scientific technologies and innovations, the widespread use of automation systems can be witnessed in diverse fields, including railway engineering domain. Moreover, the reduction in birth rate in Japan over the last decades seems to have an undesired impact on the traffic demands, especially trains, and also the number of train drivers. For this reason, the realization of automation in railways is significant. Based on the standard “IEC 62267:2009 – Railway applications – Automated Guided Transport (AUGT) – Safety Requirements” [1], for the main features of driverless train operation (DTO) and unattended train operation (UTO), there will be no driver on-board and the trains need to be able to detect obstacles and control themselves in order to avoid collisions. Current intelligent train detection systems leverage software and hardware available, for instance, advanced information transfer and communication, automatic train control as well as precise localization technologies to prevent crashes with on-track obstacles. This situation represents the high level of railway transportation modernization. The well-known examples of UTO trains in Japan is the Yurikamome line, meanwhile, those of DTO is the Tokyo Disney resort line.

While fully automated train operations (ATOs) are already in the advanced stage of development, limited research effort has been focused on driverless trams. When it comes to driverless systems, particularly trams, the safety of the operation is undeniably of great importance. For trams, differing from trains, they are typically operated not only on dedicated tracks identical to trains, but also roadways commonly shared with cars, which are relatively more complicated in comparison with trains. Simultaneously, as a well-known fact, tram cars also navigate through city streets in which there is a high risk of fatal crash accidents resulting from the enormous number of pedestrians and cyclists. This strongly confirms the requirement of a fast and accurate perception system for driverless tram systems.

To achieve the driverless tram operations, initially, the introduction of an ability to understand the frontal environment into the trams is crucial. Failure to grasp it can cause irreparable loss and damage to trams, their surrounding and trustworthy. With this safety concern, undoubtedly, one of the most critical things to first recognize is a tram driving path for enhancing safety against collisions between trams and obstacles appearing in the path to be occupied by trams. The definition of the tram driving path, in this paper, is the space between two rail tracks and the space between the rail tracks and the ends of a tram vehicle that will be occupied by the tram as it moves forward, as illustrated in Figure 1. Given this definition, it leads to this research study of the rail detection and boundary generating.

In this paper, thus, the key methods for the tram driving path identification system are studied. The paper is organized as follows, in the first section, the overview idea and main components of the proposed system - rail detection and boundary generating modules, the framework, and the experiment description are provided in details. Successively, the comparative result of the different rail detectors within the rail detection module along with the preliminary test result of the introduced tram driving identification system are presented and discussed. In the final section, the conclusion of this research study is drawn based on the insights from the results and the future vision towards the sustainable development of this system by emphasizing the feasibility of its applications in real-world driverless tram commercial services.

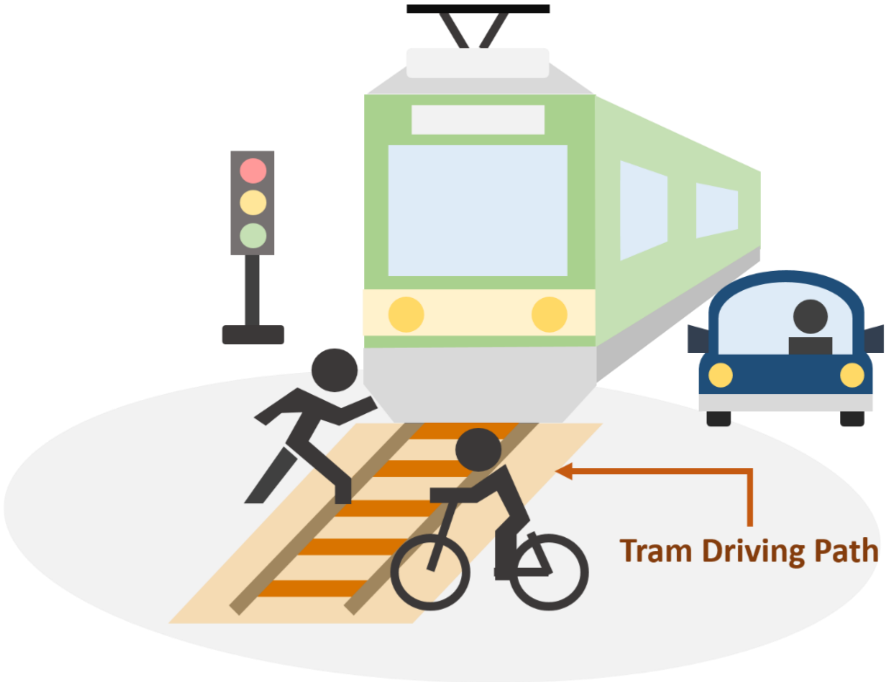

Figure 1: Tram driving path

2 Methods

In this section, the overview of the tram driving path identification system, as depicted in Figure 1, and the outline of the internal algorithm are given in each subsequent subsection. First and foremost, this system comprises two district modules working in sequence, namely rail detection module and boundary generating module.

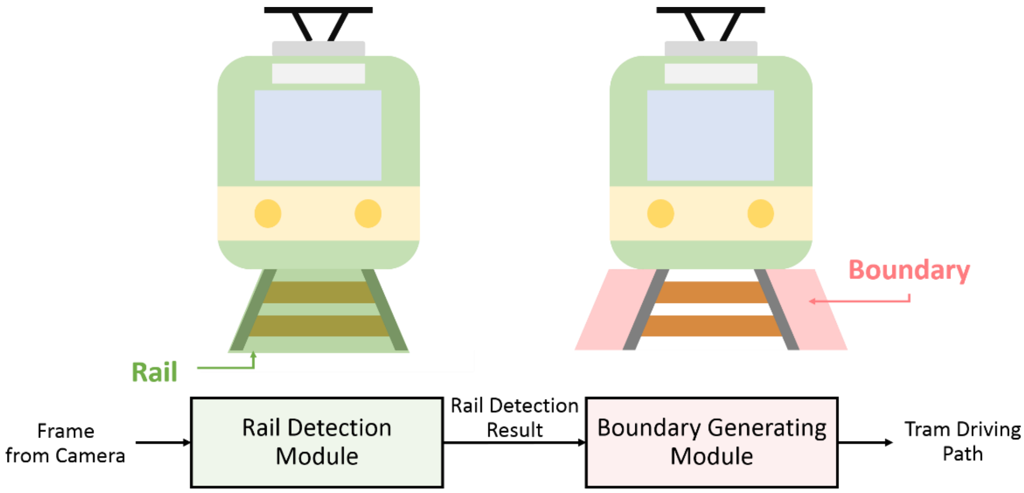

Figure 2: Tram driving path identification algorithm

2.1 Rail detection module

For this module, it is a binary classification task. Consider the tram driving path shown in Figure 1, which indicates that the detection of rails is the utmost priority without reservation. In general, there are two major categories of detection methods. The first one is a traditional image processing technique or known as a feature-based approach. This technique relies on features in an image. One of the features of rail tracks is the presence of edges, marked by its sharp contrast between the track and the surface on which it is placed. This sharp contrast can be represented by a large gradient in pixel intensities. By taking advantage of the edge feature, and applying edge detection techniques, the detection of railway tracks is assumed to be possible. One of the most widely-used edge detectors are Sobel edge detection and Canny edge detection, which are both selected for this study.

Another category is the deep learning-based segmentation method. Until presently, deep learning stands out as an excellent tool in this regard, and countless segmentation models have been released. More specifically, the models can be classified into two types with respect to model architectures, including Convolutional Neural Networks (CNN) and transformers. Among those, some have swiftly risen to prominence, such as U-Net, PSPnet, DeeplabV3plus and SegFormer. The first three, U-Net, PSPnet and DeeplabV3plus, are all based on CNN, conversely, the latter, SegFormer, exploits transformers; all are selected for this study. Consequently, it is greatly anticipated that rail tracks in an image frame can be detected by segmenting with deep learning-based methods. In summary, for the deep learning-based method, the rail detection is regarded as a segmentation task, and the above-mentioned segmentation models will be compared to each other to figure out the most appropriate model for the module.

2.2 Boundary generating module

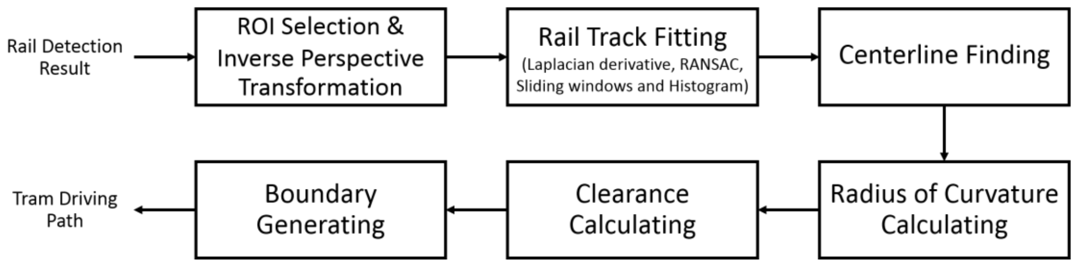

Figure 3: Boundary generating module framework

The proposed boundary generating module aims to identify the space between tracks and the vehicle's ends. The method involves the selection of region of interest (ROI), inverse perspective transformation, rail track fitting and centreline finding, calculating the radius of curvature and determining a clearance, as illustrated in Figure 3.

First of all, the inverse perspective transformation becomes part of this module because a certain feature can be correctly obtained if looked upon from a different perspective. A radius of curvature of rail tracks is such a feature. Actually, parallel rail tracks merge into one point, called vanishing point, in the distant area of the vision from a tram vehicle's cab. The distance between two adjacent tracks decrease towards the vanishing point, making it difficult to calculate a radius of curvature. For this reason, the original image of the detected rail tracks, produced by the rail detection module in a form of a binary image, is mapped into the top-view image using inverse perspective transformation to earn the parallel characteristic of the rail tracks.

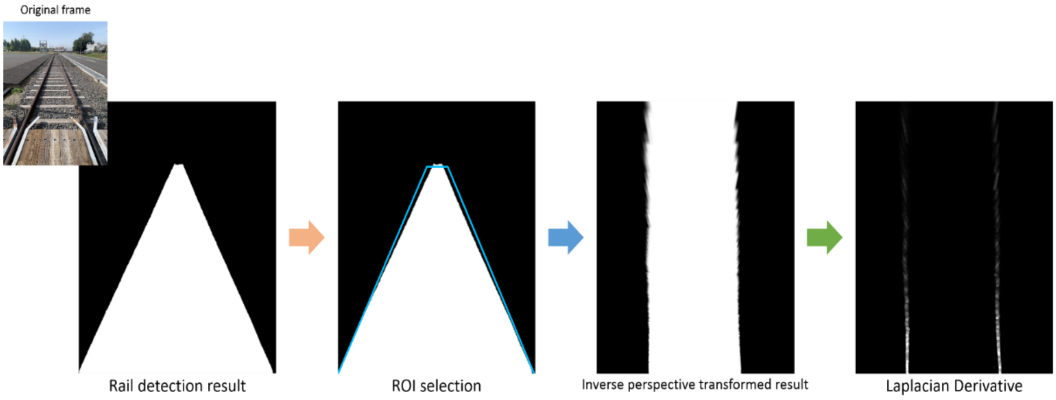

Figure 4: ROI selection, inverse perspective transformation and Laplacian derivative

For actual rail tracks' geometries, roughly, there are straight and curved tracks. In the light of this fact, a sufficiently-high order polynomial needs to be employed to handle the fitting of various tracks. Therefore, the third-order-polynomial function is adopted and its corresponding mathematical expression is written as below

$$u = A + Bv + Cv^2 + Dv^3 \tag{1}$$

Where, u is a horizontal pixel coordinate, v is a vertical pixel coordinate and A, B, C, D are coefficients of the third-order degree polynomial function

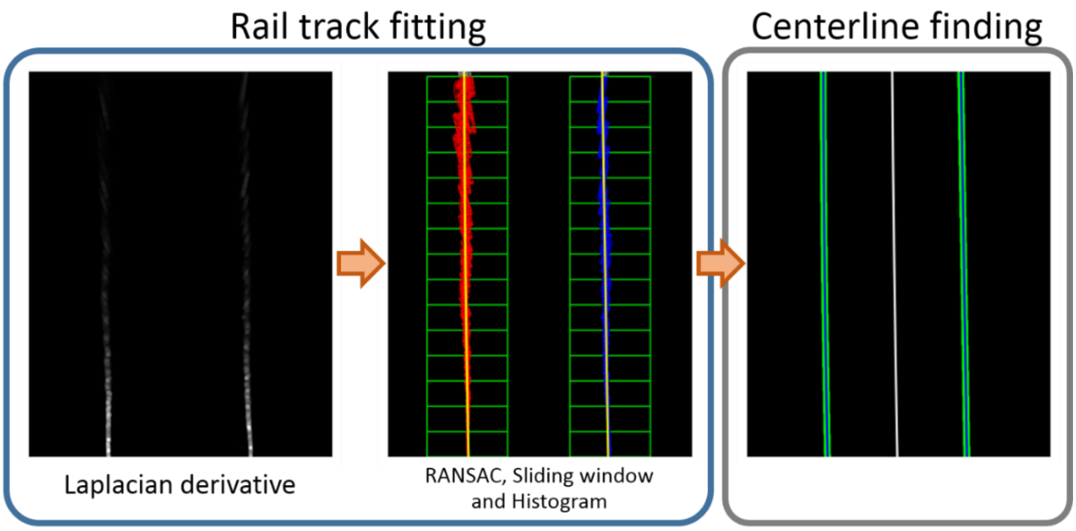

Figure 5: Rail track fitting and track centreline fining

Further, consider Figure 6, during negotiating a curved track, a rail vehicle's car body does not follow the path of the curve owing to being a rigid body. Hence, the vehicle will project towards the outer rail track near its end [2]. To prevent a collision, a clearance required at the end of the vehicle is tremendously vital. This clearance is associated with the rail vehicle's width, and end throw, which relies on the radius of curvature of the track centerline, the vehicle's length, and the center-to-center bogie distance. The corresponding mathematic equations are shown in Equation (2) and (3),

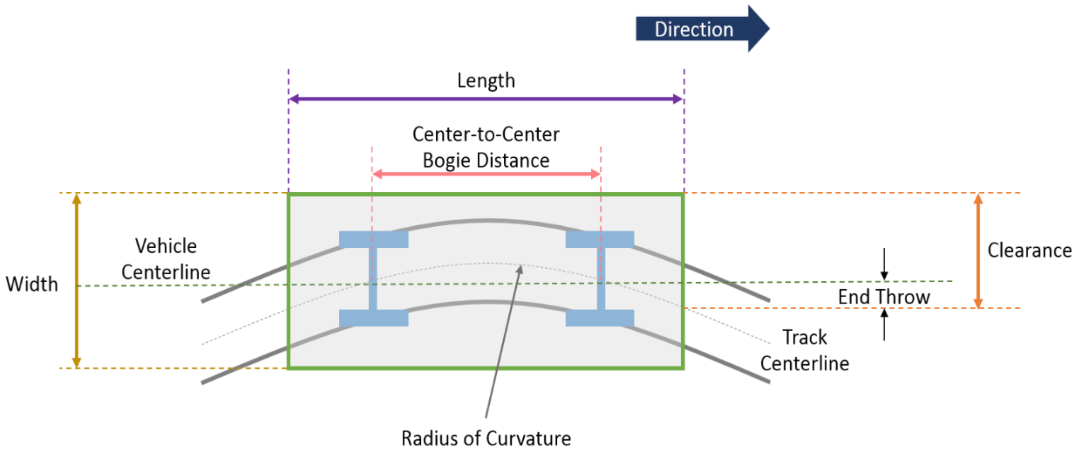

Figure 6: Clearance of a rail vehicle at a curved track

The equation of the radius of curvature is defined as

$$R = \frac{\left[1 + \left(\frac{du}{dv}\right)^2\right]^{3/2}}{\left|\frac{d^2u}{dv^2}\right|} \tag{2}$$

The formula for clearance is derived as follows

$$\text{Clearance} = \text{Half width} + \text{End throw} = \frac{W}{2} + \frac{L^2 - K^2}{8R} \tag{3}$$

Where, W is a rail vehicle's width, L is a rail vehicle's length, K is a center-to-center distance between two bogies, and R is a radius curvature of track centerline

Delving into the framework of this module, the initial step incorporates the ROI selection and inverse perspective transformation, which is applied to the segmented result from the rail detection module to acquire an aerial view of rail tracks. In this view, the rail tracks appear to be relatively parallel to each other, enabling the rail track fitting with a polynomial using Laplacian derivative, RANSAC algorithm, sliding windows and histogram to measure a radius of curvature of the track centerline later on. Following this, the clearance value will be computed to generate a boundary. Ultimately, the detected rail and the generated boundary are transformed back into the original image, and the visual illustration of the tram driving path is output.

2.3 Dataset and experiment details

To attain the objective of this research, datasets and experimental tests are necessary for evaluating the quality of the proposed method under complex conditions and dynamic environment of railway systems. Particularly, the experiments are divided into two stages. The first experiment is set up for testing the rail detection module, while the second experiment is the preliminary test for the tram driving path identification system to examine whether it is potent for the designated task.

The experimental data entirely comes from a public source dataset. This dataset offers a gigantic quantity of annotated frames related to rail transit scenes. The name of the dataset is Railsem19 [3]: the largest and most varied railway and tram scenario open dataset ever released containing 8500 images in total, which are captured from the ego-vehicle perspective along railway and tram lines with the variations in lighting and weather conditions.

In order to fairly compare the performance of each rail detector, each deep-learning segmentation model is trained by using 5950 images randomly split from Railsem19 Similarly, 1275 images of Railsem19 are randomly selected for creating a validation dataset. Then, the remaining serves as a test dataset to assess the effectiveness of each deep-learning model. On the other hand, since the conventional models, Sobel and Canny edge detectors, do not demand a training stage, all images from the dataset are used as a test dataset for them. About the training process, each deep learning model is trained for 40 epochs based on Jaccard loss function and Adam optimizer. Moreover, to increase the diversity of the dataset, the data augmentation is applied for preparing the data in the training stage with a variety of techniques, such as the horizontal flip, random rotation, random crop, random contrast, image blurring, adding Gaussian noise, colour transform, and changing hue and saturation values.

As a further matter, each trained model is evaluated on the test data in a quantitative manner. In addition to ubiquitous evaluation metrics for binary segmentation task, including accuracy, precision and recall, the Intersection-over-Union (IoU) and processing time (frame per second: fps) metrics are also considered to help assess the inference performance and inference speed of the system in real time. Eventually, the identification system, which is constructed based on the rail detection module and the designed generating boundary module, will undergo the preliminary test.

Last but not least, several images of rail tracks are further gathered from the test track located at Kashiwa campus, the University of Tokyo. This certain dataset is implemented for the preliminary test of the proposed tram driving path identification system. In this step, the developed system will be tested with the set of data from the test track at Kashiwa campus without determining any metrics.

3 Results

In this section, the experimental results of rail detectors and the preliminary test results of the proposed tram driving path identification systems are shown and discussed.

The comparative result of the classical and learning-based approaches for rail track detection is presented in Table 1. Unfortunately, it was found that either Sobel or Canny detectors is unsuccessful in detecting rail tracks. It is obvious that they failed in almost all scenarios, especially when shadow, road marks and objects like humans or cars exist in the scenes, as illustrated in Figure 7. This is attributed to the fact that the shadow, road marks and such objects also have their edge features similar to the railway tracks and the inability of each edge detector to distinguish those edges, undermining the reliability of the traditional methods.

On the contrary, for the deep learning-based approaches, they are proven to be effective in segmenting railway tracks from images with superior performances in comparison with the feature-based ones. In term of accuracy, precision, recall and also IoU, DeeplabV3plus outperforms the others. Nevertheless, it is the PSPNet model, which surpasses the others when considering the processing time as a major metric. As per the result, it can be noticed that the higher accuracy, precision, recall and IoU are obtained at the cost of worse processing time, and vice versa. Since this system is intended for real-time operation, it becomes imperative to maintain a balance between overall accuracy and processing time prior to the deployment of the system.

It is worth noting that all deep learning models are built with ResNext50 backbone and trained on the personal computer with RTX 3060 GPU (VRAM 12 GB) under the environment of CUDA 11.4, Python 3.10.13 and Ubuntu.

Type	**Model**	**Accuracy**	**Precision**	**Recall**	**IoU**	**Processing time [fps]**
Traditional method	Sobel	*Failed to detect rail tracks accurately*				
	Canny	*Failed to detect rail tracks accurately*				
Deep learning-based method	U-Net	0.9940	0.9211	0.9531	0.8835	5.68
	PSPNet	0.9863	0.8663	0.8578	0.7624	**11.6**
	DeepLabV3+	**0.9949**	**0.9373**	**0.9550**	**0.8995**	6.17
	SegFormer	0.9930	0.9117	0.9411	0.8656	5.00

Table 1: The performance of each rail detection model

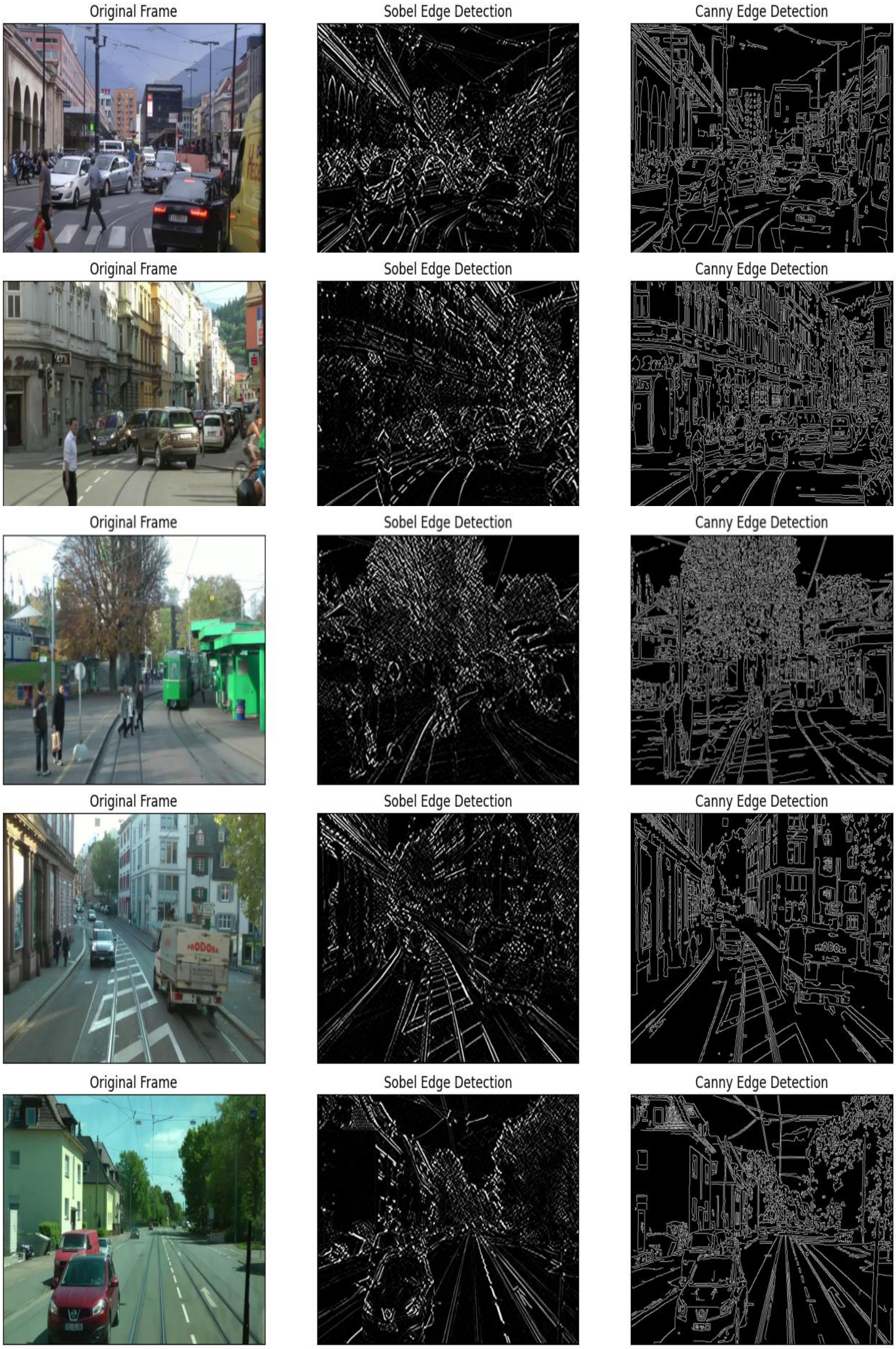

Figure 7: Examples of rail detection result from Sobel and Canny edge detections

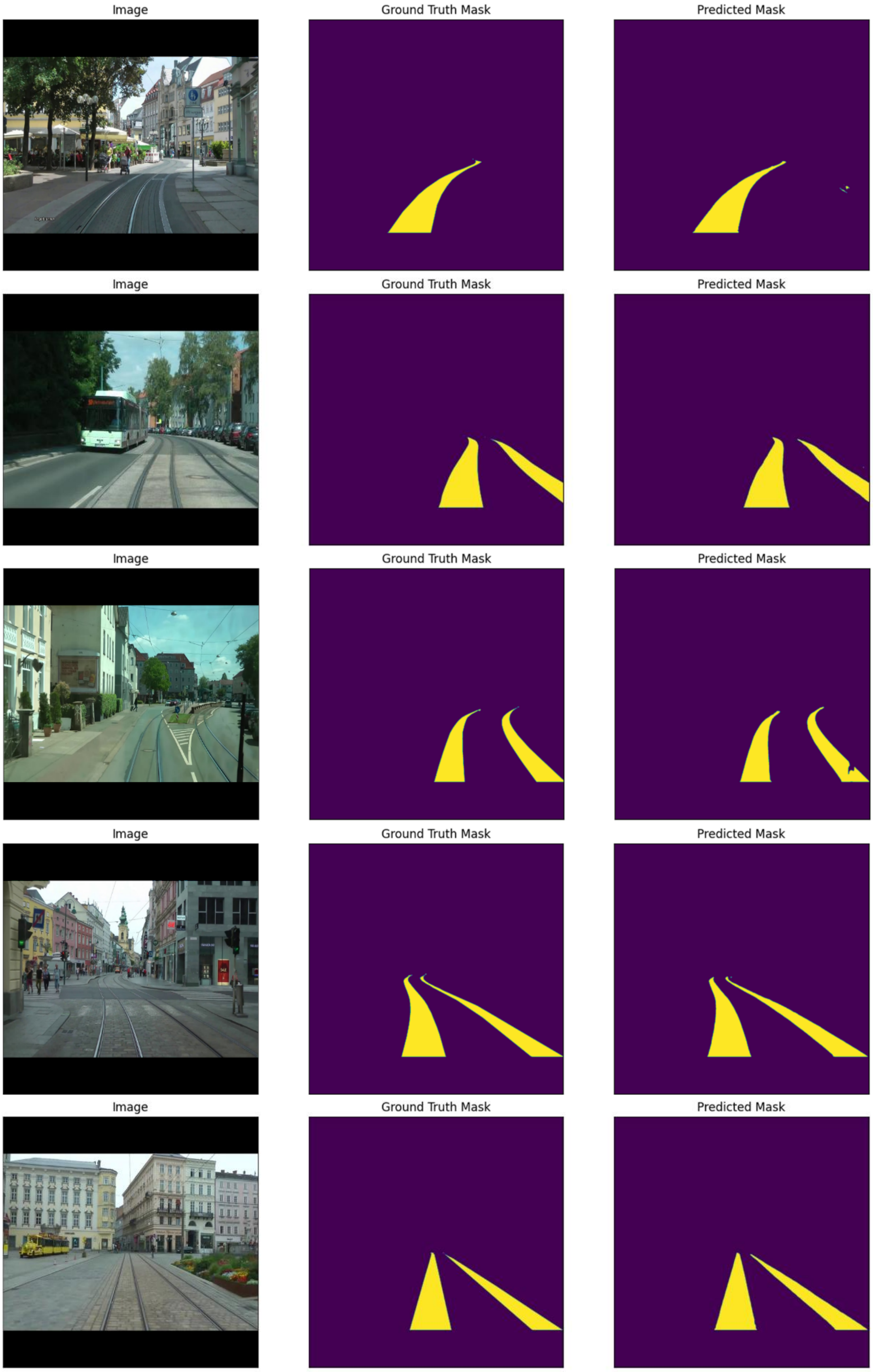

Figure 8: Examples of rail detection result from U-Net model

Next, the preliminary test of the developed tram driving path identification system is performed. In this preliminary test, DeepLabV3plus model is selected for the rail detection module due to its high inference and speed performance.

As previously mentioned, the images accumulated from the test track at Kashiwa campus of the University of Tokyo are utilized in this stage. Figure 9-10 demonstrate the original frame, rail detection results and preliminary test results of the system with both straight and curved tracks being tested. The rail track space and the boundary are indicated as green and red areas, respectively.

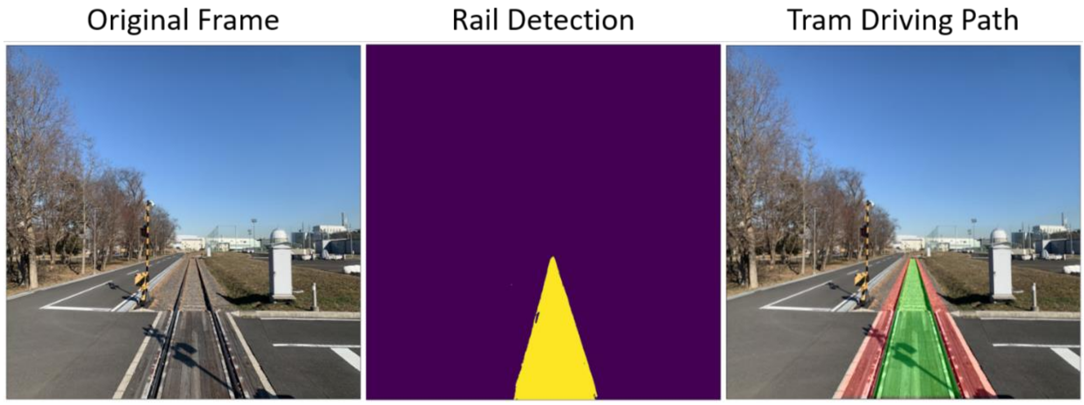

Figure 9: Preliminary test result of the system: straight track

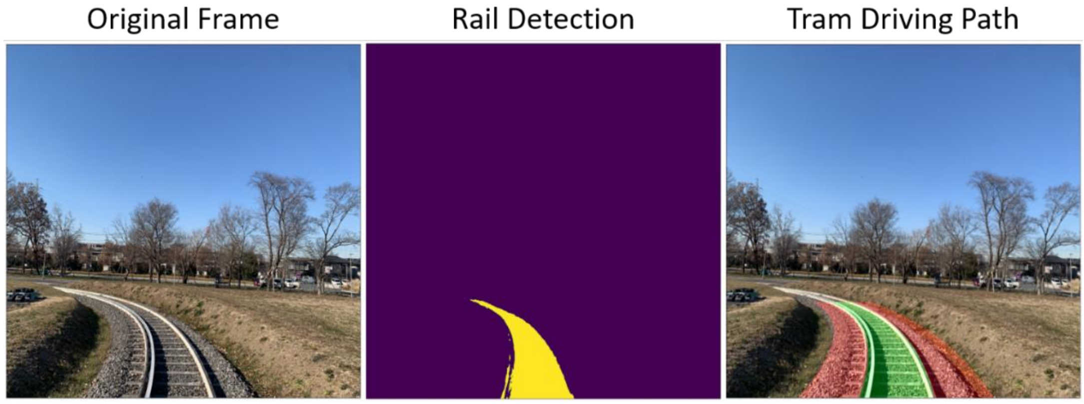

Figure 10: Preliminary test result of the system: curved track with R = 33.3m

From the preliminary test results as displayed in Figure 9-10, it is evident that the proposed tram driving path identification system can perform its assigned role very well, and the final results are genuinely impressive.

4 Conclusions and Contributions

To safely operate driverless trams, having the vision to perceive the surrounding is indispensable for the trams. Accordingly, the objective of this research is to introduce an algorithm for identifying the tram driving path. In particular, the algorithm is built on the rail detection and the configured boundary generating modules. Successively, the comparison of different rail detection methods is conducted, and then the proposed tram driving path identification is preliminarily verified via the specific test dataset. Both comparative results and preliminary test are summarized as subsequent

- The conventional methods, Sobel and Canny, did poorly work in rail detection, whereas the deep-learning models had incredible outcomes
- The results clearly highlight a speed-accuracy trade-off of the deep learning based-rail detection models
- The preliminary test results of the tram driving path identification system showcase the promising effectiveness of itself

It could be concluded that the proposed system has the potential to facilitate the actual driverless tram services. Additionally, the future works of this research are

- Other types of backbones, such as MobileNet and EfficientNet, are planned to be tested to help enhance the processing time of the rail detection module
- Full evaluation process for the developed tram driving path identification system is important. It is planned to be accomplished with more railway scenes images. Meanwhile, accuracy and root mean square error (RMSE) will be included as significant metrics as well in this process
- Testing the proposed identification system with real tram running experiments at test tracks is the next focus. This will allow us to gain a deep comprehension on how the system works in real time
- A transition track, the radius of curvature of which is varied, will be considered

References

[1] IEC 62267:2009, "Railway Applications. Automated Urban Guided Transport (AUGT). Safety Requirement", British Standards Institution, London, UK, 2009.

[2] S. Chandra and M. M. Aggrawal, "Railway Engineering", Second edition, Oxford University Press, India, 2013.

[3] O. Zendel, M. Murschitz, M. Zeilinger, D. Steininger, S. Abbasi and C. Beleznai, "Railsem19: A Dataset for Semantic Rail Scene Understanding", "2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)", Long Beach, CA, USA, 2019, pp. 1221-1229. doi:10.1109/CVPRW.2019.00161.