Detecting Railway Infrastructure using Point Cloud Data

Anthony Anazo
Point Cloud Data Railways 1
University of Twente
Enschede, The Netherlands
m.a.c.anazo@student.utwente.nl

Daniel Actor
Point Cloud Data Railways 1
University of Twente
Enschede, The Netherlands
d.j.actor@student.utwente.nl

Reinier Algra
Point Cloud Data Railways 1
University of Twente
Enschede, The Netherlands
r.a.algra@student.utwente.nl

Joris Jonkers
Point Cloud Data Railways 1
University of Twente
Enschede, The Netherlands
j.w.jonkers@student.utwente.nl

Table of Contents

I. Introduction

In 2016 the EU parliament created Directive (EU) 2016/919 which brought about a new European-wide railway system named the European Railways Traffic Management System (**ERTMS**), designed to enhance railway safety, security, ease and reduce travel times across Europe. As a result, organisations like ProRail were entrusted with the crucial task of changing the railways from analogue to digital. However, the sheer magnitude of this task, encompassing approximately 7,097 kilometres of rails, has proven to be a tedious challenge.

To alleviate this challenge and reduce the workload and overall project costs, a modern approach was sought. Namely the application of object detection to detect railway infrastructure which may need to be changed to meet the requirements laid out for the ERTMS. This concept gave rise to the Point Cloud Data Railways project, which aims to provide a practical solution.

The Point Cloud Data Railways project aims to detect railway infrastructure from data gathered by **LiDAR**[1] sensors from the perspective of a train. This new approach leverages point clouds—extensive datasets composed of millions of points captured by laser scanners. These point clouds, once clustered, give a detailed representation of the railways, which can then be analysed for object detection, segmentation, or classification. The primary goal of this project is to implement a Part-A^2 model[2] and configure it using the MMDetection3D[3] framework to allow for the detection of railway infrastructure in the dataset.

---

[1] Light Detection And Ranging
[2] See Methodology/Modelling for more information
[3] mmdetection3d.readthedocs.io/en/latest

II. Methodology

CRISP-DM Framework Application

This project applies the Cross-Industry Standard Process for Data Mining (**CRISP-DM**) framework to address the challenge of detecting railway infrastructure through object detection technology. The choice to use CRISP-DM was made as it is an industry standard for data science projects.

Business Understanding

The project aims to develop a system capable of identifying railway infrastructure components by utilising the Part-A^2 model applied to LiDAR data supplied by Strukton Rail. This initiative supports the digitalization of the Dutch railway network, through streamlining the railways mapping and inspection process.

Presently, rail mapping and inspection is a costly endeavour, requiring specialised equipment and trained personnel. This project seeks to improve upon the process by determining the feasibility of using LiDAR-equipped trains and object detection to perform the same task. This promises to reduce costs in mapping out the railways to standardise the railway network as required by the ERTMS.

Data Understanding

The dataset includes LiDAR point clouds labelled with various infrastructure elements. These point clouds offer a view of the rail network from the train’s perspective. To build a comprehensive understanding, an initial exploration of the data was necessary.

Data Preparation

For preparing the data, the MMDetection3D framework was used. It has a tight integration of the data preparation and modelling steps and is designed to work with LiDAR data. It also has an implementation of the model that will be tested and was previously used by the project supplier, Bram Ton. Thus, for compatibility with Bram Ton's and this project, this framework was used.

The dataset, being custom and not predefined in the MMDetection3D framework, presented two main challenges: adapting the dataset for compatibility and configuring the framework to suit the custom dataset. This entailed a restructuring of the dataset, including the creation of a suitable annotation file in pickle[4] format with bounding boxes for model predictions, adapted from code provided by Bram Ton.

Modelling

The MMDetection3D framework used is based on PyTorch[5] and MMDetection made as part of OpenMMLab. It supports all models and papers included in MMDetection. It also supports many popular indoor and outdoor point cloud datasets like Lyft and Kitti.

The **Part-A²** network consists of two stages, the part-aware stage and the part-aggregation stage.

[4] Annotations including bounding boxes of labels created from the dataset
[5] https://pytorch.org/docs/stable/index.html

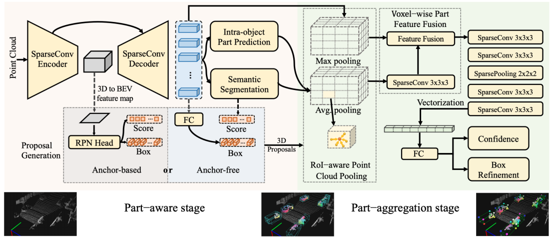


Figure 1: The framework of the Part-A² network for object detection

The part-aware stage uses part information derived from 3D ground-truth boxes to make 3D predictions and find the location of different parts of an object. It then groups those locations using the RoI-aware[6] point cloud pooling module to create a representation that encodes the specific features of each proposal. Then the part-aggregation stage learns to re-score the box and refine the box location by considering how the different parts are positioned relative to each other.

Unlike most other models in the MMDetection3D framework, the Part-A² model only needs point cloud data to work, as compared to using an image reference.

The Part-A² Model integration of MMDetection3D allows for but also requires, the configuration of many hyper-parameters.

Evaluation and Deployment

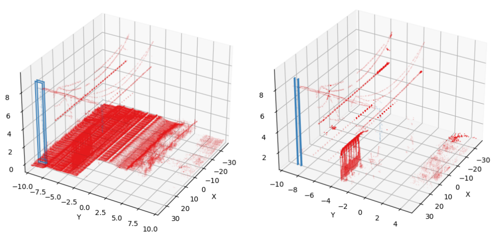
The methodology section does not detail the evaluation and deployment stages as they pertain to the subsequent phases of the CRISP-DM process, which will be described later on in the results section.

III. Results

[6] Region of Interest aware

Data Understanding Phase

In the product's developmental phase, data understanding posed a significant challenge. The absence of documentation for the provided dataset initially obscured its relevance to the problem at hand. To overcome this, data visualisation techniques were employed, specifically utilising Matplotlib to create scatter plots. These visualisations revealed extraneous points within the point clouds, with object occurrences confined to narrow ranges along train tracks. Subsequently, a decision was made to restrict the model's point ranges to the vicinity of the track. As can be seen in the image below.



(left) original (right) restricted

Additionally, the chosen object detection model hinged on recognising distinct shapes, necessitating an analysis of point clouds to determine the most representative dimensions and point ranges for bounding boxes across various object categories.

Data Preparation

The dataset initially lacked annotations and was therefore incompatible with the object detection task. To remedy this, a script provided by Bram Ton was employed, calculating bounding boxes for different object types. This script was enhanced to determine object sizes and ranges, partition the dataset into training, validation, and testing subsets, and confine coordinates for bounding boxes to the specified ranges as illustrated in the data understanding phase.

To augment the model's performance, point cloud data was subjected to several augmentation techniques, including horizontal rotations, translations, random point filtering, point shuffling, rotations along three axes, and scaling.

The provided data was not compatible with the framework initially and was converted from the LAS file type to BIN. To accomplish this a three-step process was used in which the LAS files were converted to PCD which were then converted to BIN files. These steps were accomplished through the PDAL[7] docker image and PYPCD[8] package.

Modelling

The Part-A^2 model, provided by the MMDetection3D framework, featured multiple tunable hyper-parameters and specific parameters that could be inferred from the data insights gathered. A critical aspect of the model was the generation of anchors, representing presumptive object locations and bounding boxes. As outlined in the data preparation section, the computation of annotations allowed for the stipulation of object ranges and sizes, integral for anchor generation. Other hyper-parameters were initially kept at default settings to facilitate preliminary explorations of model performance.

After the point cloud data and annotations have been prepared, the next step in the process is to configure the base model. This lengthy process included understanding most of the framework and configuring it to work with the custom railway dataset. The majority of the configuration was adjusting many of the config[9] functions.

[7] https://hub.docker.com/r/pdal/pdal
[8] https://github.com/dimatura/pypcd
[9] File including variables used for configuration

The MMDetection3D has a strict middle encoder shape and since there is a unique point cloud range. The point cloud has four attributes which are the x,y and z-axis and intensity. The voxel sizes are calculated such that the middle encoder shape could be used in the framework. The calculation of the voxel sizes per attribute was the variance of the point cloud divided by the middle encoder shape.

Evaluation

The efficacy of the model was assessed using the mean Average Precision (mAP) metric, which gauges the overlap between the predicted and ground truth bounding boxes and evaluates the extent of background inclusion in predictions. Unfortunately, this metric indicated consistently that the model's performance was poor.

This led to the realisation that the model or data might not yet be attuned to the problem. Further investigation into the data understanding phase identified a class imbalance, with the background points considerably outnumbering those composing the objects of interest.

Deployment

The model will not be deployed, as it is not within the project's scope to do so. The results produced are not sufficient to yield the investment needed to do so, instead, a suggestion is made to re-evaluate the business understanding to determine whether improving the quality of the provided data is feasible.

IV. Conclusion

The goal of the project was to create and train the Part-A^2 model to detect infrastructure surrounding railways. By making adjustments to default implementations of the model and allowing the provided dataset to be used. Further, to augment and filter the data to improve its quality for the stated application.

Unfortunately, the complexity of the used model made it difficult to fit it into our data. Further, the analysis of the data showed that the target classes were of low enough quantity that it proved insufficient for the model to learn meaningful representations.

Even though the model's configuration was not sufficiently fitted to the data to detect railway infrastructure. A large amount of pre-processing and evaluation was implemented allowing future projects to continue fine-tuning the model using the developed tools.
However, as we were not able to visualise made predictions it is yet unclear to us exactly where the model makes its mistakes.

V. Discussion

The project aimed to detect the infrastructure surrounding the railways. First, an attempt at segmentation was made which was interrupted and replaced with object detection. This object detection did not yield results with high performance due to poor data quality, namely including a class imbalance and a low number of examples. This quality of the data was further reduced due to the unavailability of information on its contents, and the lack of comments concerning the bounding box generation script.

The findings show that there are a lot of irrelevant points in the point clouds and very few labelled points which leads the model to be unable to learn what points are part of the desired object to be detected.

We are not familiar with the formulae used to generate the bounding boxes, this may reflect the inaccuracies of our model. The model may be able to recognize the poles correctly during the segmentation stage, however, if the boxes are incorrect its learning goals may be invalid.

The significance of our findings is negligible and may not be applicable to other applications of the same model. The produced configuration for the Part-A^2 model does not make accurate predictions due to the many reasons given concerning poor finetuning and data quality and possibly more.

The project's original outline was to examine the efficacy of the Part-A^2 model. However, the PointPillars model implementation made by Bram Ton was not provided for this project. This caused much of the time available for the project to be spent preparing the data and the framework. The delay led us to not be able to do an extensive configuration of the Part-A^2 model during the project. In the future, the provision of past implementations may serve to allow for progress to be made more readily, by reducing the burdens of re-implementation.

Technical depth

The implementation of the Part-A^2 model on a point cloud dataset in the MMDetection3D framework requires in-depth knowledge of the framework. It has a steep learning curve with regards to working with custom datasets as the documentation was lacking in many aspects, and at the time of writing had not been updated for the latest version of the framework. This was a major challenge as it required a deep understanding of not just Machine Learning models but of this specific framework. Every configuration could have resulted, and many did, in an exception internal to the framework. This error-chasing made up the bulk of the project.

Reflection

Reflecting on the project's process, the steep learning curve of the framework made us spend a lot of time creating a functioning model. This could have been largely prevented with a similar project example, such as Bram Ton's project or more guidance on the framework. It would have allowed us to focus more on improving the effectiveness and efficiency of the model.

VI. Module application

To verify the parts of the module that were applied, the courses are listed and the contents that were applied will be discussed.

Data quality

First, a look is taken at some of the dimensions of the data to assert the quality of the data starting with the data values. The data values (point clouds) were acquired through a Lidar sensor which made them easy to attain and very precise, thus the dimensions of precision and attainability were high. The appropriateness also stood out because the point clouds were very suitable for representing the objects that needed to be detected.

The data model includes the format of the data points and the bounding boxes relating to the point clouds. In the discussions of the dimensions of data quality, flexibility in particular stands out as the flexibility of the given model was quite low. It is not very usable for different purposes. The obtainability of the data for this project is very good, as it was already gathered by the LiDAR sensors on trains and labelled beforehand.

Data exploration-driven approaches such as visualisation using CloudCompare and Matplotlib were made to understand the data values more so the right adjustments could be made to the data such that it would be of higher data quality for this application.

KRR

KRR was not present in the project, because the concepts discussed in the lectures did not apply, as the data did not contain actions or scheduling to perform. There also was no need to use either first-order logic or propositional logic.

ML4DT

Many of the techniques and knowledge gained in the configuration and training of models from ML4DT helped us to determine the important parameters in the model configuration. The voxelization of the point clouds served a similar purpose as the kernels of convolutional networks used in computer vision. If implementing a custom network, many more of the taught concepts would likely be applied when determining a beneficial model architecture.

XAI

As the Part-A^2 model contains several different models, including a CNN, due to the compound nature of the different architectures, it is difficult to trace why the model makes certain predictions, thus the explainability is low. Furthermore, the produced model's predictions are of such low quality that we were not able to apply techniques such as DeepShap to identify important ranges within our points clouds. This is due to our model rarely making the predictions for bounding boxes, which we would then be able to trace. In the future, if the model accuracy were higher, it may be possible to distort certain regions or features to better determine the interpretability of the model.

Ethical perspectives

Most of the ethical considerations to be made are on the deployment or continuation of the developed model. Therefore most of the ethical perspectives taken into account during the planning and development of this project involve its deployment.

The model, if deployed in its current state, will make mistakes in misidentifying objects or missing them altogether. This can in turn result in some infrastructures not being mapped, and therefore not being replaced or digitalized as required for the ERTMS. These safety concerns caused by a lack of precision of the model lead us to recommend against its usage in its current state.

The public train system at present is open to all citizens, ensuring an equitable distribution of benefits. This project thus follows ethical behaviour from a justice and fairness perspective. A utilitarian might even say that even if the model is not super accurate, its deployment would still be worthwhile. The benefits for safety and efficiency outweigh any slight errors from this perspective. The same can be said for the common good perspective from which it could be argued that any positive contribution to public transportation safety and efficiency is good. On the other hand, virtue ethics might feel that the safety of the railroad could be under pressure. Careful research and consideration should be done to ensure this responsible deployment.

Furthermore, the usage and advancement of object detection in public spaces, like railways, can be used for malicious practices invading citizens' privacy. Thus, in the case of deployment, the usage of this data and its result must be done with purpose, law, and morality in mind. On top of that, it is recommended to write it down in a detailed manner for the public to read. Using object detection in public spaces by the government in this way might also be a first step that could allow them to use it in other situations as well, positively or negatively.

VII. Sources

EUR-Lex - 32016L0798 - EN - EUR-Lex. (2016). Retrieved October 25, 2023, from https://eur-lex.europa.eu/legal-content/NL/ALL/?uri=CELEX%3A32016L0798

180 jaar spoor: Het begin van de spoorwegen. (2019, December 3). ProRail. https://www.prorail.nl/nieuws/180-jaar-spoor-het-begin-van-de-spoorwegen#:~:text=Het%20aantal%20kilometers%20spoor%20is,treinen%20neemt%20alleen%20maar%20toe.

OpenAI. (n.d.). *Conversational Advice on Crafting Effective Introductions.* CHATGPT. Retrieved October 26, 2023, from https://chat.openai.com/share/d902f814-065c-4870-908c-13ab05d83c35

Open-Mmlab. (ca. 2022). *Figure 1*. GitHub. https://user-images.githubusercontent.com/79644370/143882774-6fc5f736-10d1-499a-8929-ca0768419049.png