FH University of Applied Sciences
TECHNIKUM WIEN

MASTER THESIS

Thesis submitted in fulfillment of the requirements for the degree of Master of Science in Engineering at the University of Applied Sciences Technikum Wien - Degree Program Data Science

Multi-sensor rail track detection in automatic train operations

By: Attila Kovacs

Student Number: 2110854031

Supervisors: Lukas Rohatsch
Daniele Capriotti

Wien, February 3, 2024

Declaration

“As author and creator of this work to hand, I confirm with my signature knowledge of the relevant copyright regulations governed by higher education acts (see Urheberrechtsgesetz /Austrian copyright law as amended as well as the Statute on Studies Act Provisions / Examination Regulations of the UAS Technikum Wien as amended).

I hereby declare that I completed the present work independently and that any ideas, whether written by others or by myself, have been fully sourced and referenced. I am aware of any consequences I may face on the part of the degree program director if there should be evidence of missing autonomy and independence or evidence of any intent to fraudulently achieve a pass mark for this work (see Statute on Studies Act Provisions / Examination Regulations of the UAS Technikum Wien as amended).

I further declare that up to this date I have not published the work to hand nor have I presented it to another examination board in the same or similar form. I affirm that the version submitted matches the version in the upload tool.“

Wien, February 3, 2024 Signature

Kurzfassung

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Schlagworte: Deep Learning, Computer Vision, Segmentation, Automatic Train Operations

Abstract

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Keywords: Deep Learning, Computer Vision, Segmentation, Automatic Train Operations

Contents

1 Introduction

According to the International Energy Agency (IEA), the global demand for passenger and freight transportation will more than double by 2050 compared to 2019 (International Energy Agency, 2019). However, a greater demand entails higher energy consumption as well as increased CO2 emissions and atmospheric pollutants. Given the fact that railway is one of the most efficient and reliable modes of transportation there seems to be consensus between politicians and researchers that a greater reliance of rail has the potential to counterbalance the negative impacts of transportation (Islam et al., 2016; Pagand et al., 2020). The IEA lists minimizing costs per passenger-kilometer or ton-kilometer moved as one of three pillars that are essential to increase the market share of rail transportation[1].

Automatic Train Operations (ATO) which refers to a system that automates different aspects of train operations is expected be one of the key drivers of a more efficient and competitive railway system (ALSTOM Transport SA, 2021; Europe's Rail Joint Undertaking, 2019). ATO is estimated to reduce energy consumption by up to 45%, increase the level of punctuality, increase operational flexibility, and allow for a 50% better utilization of the infrastructure when combined with other technologies.

ATO relies on advanced technologies that are used to perceive and interpret the railway environment in order to allow autonomous operations with minimal or no human intervention (Deutsche Bahn AG, 2022). One aspect of ATO is the precise identification and localization of railway tracks. The ability to detect and isolate tracks based on video images is essential for ensuring the safe navigation of trains through the railway network or in shunting yards. Accurate track detection ensures that the train can make informed decisions, such as adjusting speed, navigating turns, and responding to potential obstacles.

Traditional methods of track detection often rely on rule-based algorithms and image processing techniques, but these approaches may face challenges in diverse environmental conditions such as bad weather, complex background, lighting variations (e.g., day and night), and dirty cameras. This master's thesis addresses the task of multi-sensor rail track detection in the context of ATO. We explore deep learning techniques, particularly convolutional neural networks (CNNs), that have demonstrated great success in computer vision tasks, including image segmentation. The application of deep learning to track detection is expected to outperform conventional non-AI-based techniques and thereby improving the accuracy and robustness of the system. Our analysis is based on a multi-sensors dataset, including images of normal RGB

[1] The other pillars are maximizing revenues from rail systems, and ensuring that all forms of transport (especially road transportation) pay not only for the use of the infrastructure they need, but also for the adverse impacts they generate.

cameras, high-resolution cameras, and infrared cameras, with different orientations, respectively. This multi-sensor approach allows to compare the the effectiveness of different cameras and informs the deployment of those in order to improve the robustness of track detection in diverse conditions.

In the context of rail track detection, researchers have explored various areas. Yet, applying deep learning techniques to detect rail tracks is a relatively raw field. In particular, there is no research that is focusing on comparing different input images such as RGB and infrared cameras and images that are oriented to the left, center, and right of the locomotive. The contribution of this thesis to the literature is three-fold: First, we select and train a deep learning model capable of accurately detecting and segmenting railway tracks using data from RGB cameras, high-resolution cameras, and infrared cameras. In contrast to approaches that have been specifically tailored to the task, we apply a general framework that is easier to use by practitioners without elaborate software engineering skills. The results of the deep learning model are compared to a non-AI based method specialized in identifying lines in images. Second, we conduct a comprehensive performance evaluation to assess the accuracy and computational efficiency of the proposed track detection system on images generated by different cameras. Third, we explore the integration of the developed model into real-world applications by applying to identify tracks in video streams.

By achieving these objectives, this research provides valuable insights and advancements to the field of railway automation, with implications for improving the safety and efficiency of automatic train operations.

2 Literature review

Traditionally, rail track detection has been performed by first extracting features of an image (e.g., gradient-based thresholds) and then detecting rails. These approaches achieve good results in certain conditions. However, deep-learning based approaches are often more robust in real-world environments (Giben et al., 2015; Li and Peng, 2022; Wang et al., 2019). Deep learning techniques, particularly CNNs, have emerged as powerful tools for image segmentation tasks, demonstrating success in various computer vision applications. Recent surveys on image segmentation and object detection using deep-learning techniques is provided by Cheng et al. (2023) and Zaidi et al. (2022), respectively.

The following sections examine related research in track detection, considering both deep learning-based segmentation and traditional non-AI segmentation methods.

2.1 Traditional rail track detection

While deep learning has shown remarkable success in track detection, non-AI segmentation techniques continue to play a role in this field as they allow the integration of domain-specific knowledge and rules into the algorithm and require less data for training. These methods are often referred to as line segment detectors and involve traditional computer vision techniques such as thresholding, edge/contour detection, template matching, and region growing (Almazàn et al., 2017; Grompone von Gioi et al., 2010, 2012; Sahoo et al., 1988).

Kaleli and Akgul (2009) present a dynamic programming algorithm to extract the rail tracks in front of the train. The idea is to first identify the vanishing point which refers to the imaginary intersection of the tracks as the distance between the tracks decreases from the bottom of the image to the top. This step is based on computing the gradient and applying Hough transform to detect the straight lines that indicate the tracks. Next, dynamic programming is used to extract the space between the two tracks. Qi et al. (2013) apply a method based on histogram of oriented gradients (HOG) to identify tracks and switches. First, HOG features are computed; railway tracks are then identified by a region-growing algorithm. The proposed method is able to predict the patch the train will travel by detecting the setting of the switches. Nassu and Ukai (2011) introduce an approach that performs rail extraction by matching edge features to candidate templates.

While the previously mentioned approaches focus on images by on-board cameras, Purica et al. (2017) examines the detection of tracks in aerial images taken by drones. The solution approach is based on Hough transform.

(Arastounia, 2015) and (Yang and Fang, 2014) develop methods to recognize railroad infrastructure from 3D LIDAR data. In (Arastounia, 2015), railway components such as rail tracks, contact cables, catenary cables, masts, and cantilevers are classified based on local neighborhood structure, shape of objects, and topological relationships among objects. (Yang and Fang, 2014) focus on the detection of tracks. The authors utilize the geometry and reflection intensity of the tracks to extract features and identify tracks.

2.2 Deep-learning based rail track detection

Deep-learning based techniques such as semantic segmentation incorporate convolutional neural networks (CNNs) and other deep architectures to automatically learn features from raw image data. Semantic segmentation aims to assign a label to each pixel in the image, distinguishing between the pixels that belong to the rail tracks and those that represent the background and is therefore particularly well suited for rail track detection.

Giben et al. (2015) and Le Saux et al. (2018) were among the first authors who evaluated the performance of deep learning-based segmentation against traditional segmentation techniques in rail track detection. In Giben et al. (2015), the authors propose a CNN for localizing and inspecting the condition of railway component based on gray-scale images. The authors

report that the CNN model is better suited to capture complex patters compared to approaches that rely on traditional texture features (e.g., discrete Fourier transforms of local binary pattern histograms). Le Saux et al. (2018) detect rail tracks in aerial images by devising a CNN based approach and different traditional approaches such as thresholding.

Wang et al. (2019) propose the RailNet – a deep-learning based rail track segmentation algorithm that combines the ResNet50 backbone with a fully convolutional network. In order to train the model, the authors compile a non-public dataset consisting of 3000 images from forward-facing on-board cameras. Experiments show that RailNet is able to outperform general purpose models for segmentation. In Li and Peng (2022), the authors compile a real-world railway dataset based on which a rail detection method referred to as Rail-Net is devised. Rail-Net outperforms traditional methods by around 51% and other deep-learning methods by around 6% based on accuracy when applied on the newly developed dataset.

A machine-learning based approach is proposed by (Teng et al., 2016) where features are extracted from super-pixels (i.e., a group of adjacent pixels with similar characteristics) and classified by applying a previously trained support vector machine.

2.3 Lane detection

Lane detection for road vehicles is similar to rail track detection for locomotives in the sense that both tasks aim to identify and segment elongated shapes in complex environments that vary in lighting conditions, shadows, and occlusions. The field of lane detection has a rich body of literature which is among others attributed to the existence of well-established benchmark datasets such as TuSimple (2017) and CULane (Pan et al., 2018).

Early work on lane detection is based on traditional approaches such as Hough transform and clustering (Duda and Hart, 1972; Ma and Xie, 2010). Recently, the focus of researchers has shifted to deep-learning based approaches (Meyer et al., 2021; Wang et al., 2022; Zheng et al., 2022). Tang et al. (2021) and Yang (2023) provide comprehensive surveys on lane detection approaches. In (Yang, 2023), the authors propose a combined approach in which the advantages of traditional and deep-learning based methods are mixed.

3 Datasets

Labeled images are an essential prerequisite for training deep-learning algorithms to detect objects accurately. With the growing popularity of deep-learning, we have observed the creation of new datasets specifically designed for railway applications.

The rail semantics dataset 2019 (RailSem19) is the first publicly available dataset for detect-

ing objects (including rail tracks) in the railway domain (Zendel et al., 2019). The French railway signaling dataset (FRSign) is a dataset focusing only on traffic lights (Zendel et al., 2019), whereas the Railway Pedestrian Dataset (RAWPED) is focusing on pedestrian detection methods (Toprak et al., 2020). The dataset proposed by Wang et al. (2019) – railroad segmentation dataset – has been compiled for the development railroad segmentation algorithms but it is not available to the public. The Rail-DB dataset is available upon request (Li and Peng, 2022). The dataset comprises 7.432 annotated images, featuring different scenarios (e.g., weather conditions).

This thesis is based on the first freely available multi-sensor dataset "Open Sensor Data for Rail 2023" (OSDaR23) for the development of fully automated driving in the railway sector (Deutsche Bahn AG, 2023; Tagiew et al., 2023). Unlike the previously mentioned dataset that involve a limited number of sensors and perspectives, the system on the locomotive used to create the OSDaR23 dataset includes multiple infrared cameras, RGB cameras with different resolution, lidar, radar, positioning, and acceleration sensors.

Preliminary experiments indicated that our model fails to generalize when trained only on the OSDaR23 dataset due to reasons that will be described in the next section. Therefore, we also train our model on images from the RailSem19 dataset. In the following, we give a detailed description of the two datasets used in this thesis.

3.1 OSDaR23 dataset

3.1.1 Overview

The OSDaR23 contains 21 video sequences captured around Hamburg, Germany between 09.09.2021 and 15.09.2021 (a map of the exact locations is given in Figure 1). The sensor setup is very comprehensive including six RGB cameras, three IR cameras, six lidar sensors, a 2D radar sensor, and position and acceleration sensors. In this thesis, we focus on images by RGB high resolution, RGB low resolution, and infrared sensors with three orientation (left, right, and center), respectively. One example per sensor is given in Figure 2. A detailed description of the sensors can be found in Appendix **??**.

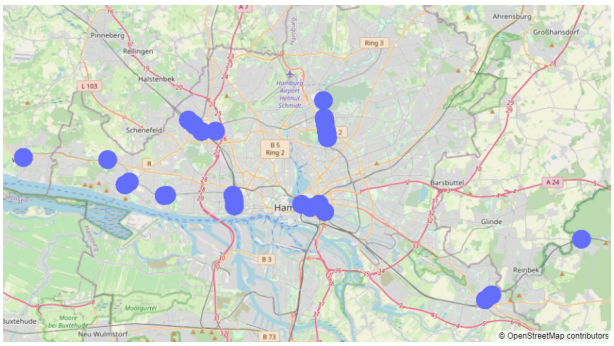


Figure 1: Locations where images were captured around Hamburg, Germany.

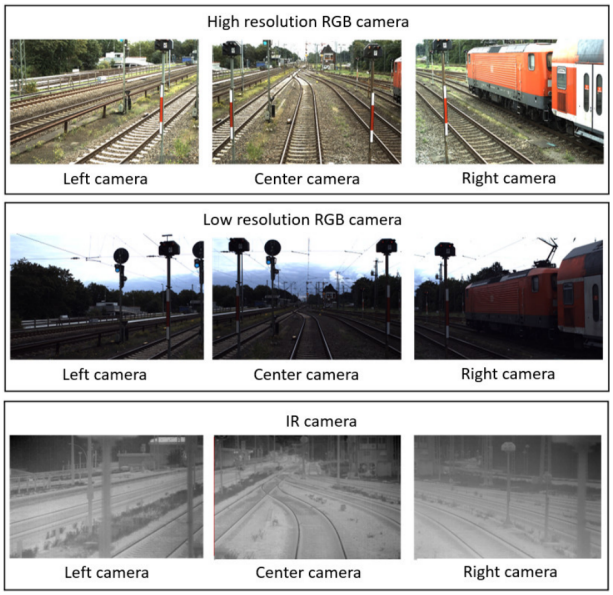
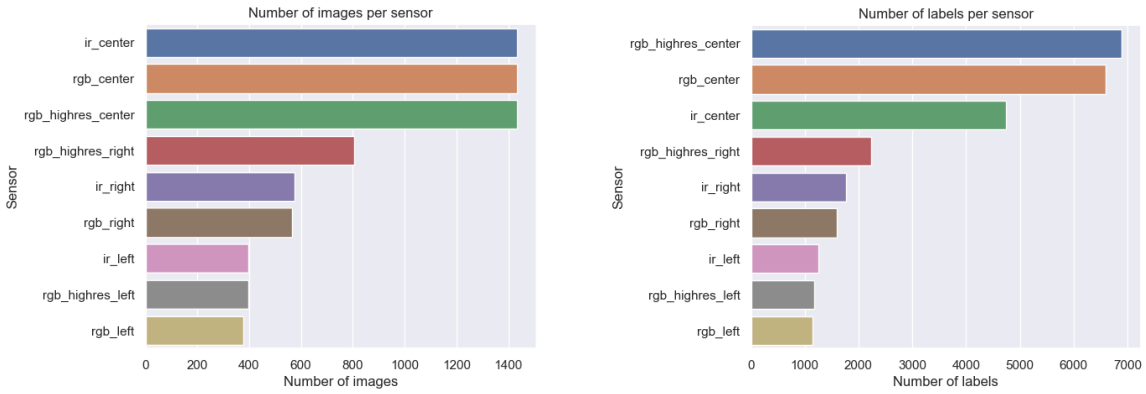


Figure 2: Example images of high resolution RGB, low resolution RGB, and infrared sensors (Tagiew et al., 2023).

The final number of images and labels after filtering the dataset, i.e., removing images that do not contain annotated tracks, is 7.421 and 27.386, respectively. The distribution of images and labels per sensor is displayed in Figure 3. The size of the images is given in Table 1. Figure 4 illustrates the number of track labels per image. Most images contain track pairs. However, there are also images with odd number of tracks. The largest group are images containing one pair of tracks. Generally, the number of available images decreases as the rail network is getting more complicated.

All images were taken between 8AM and 17PM, so we cannot expect to test the effect of different sensors in the night. In particular, the RGB cameras fail to capture clear and detailed images in low-light conditions. Infrared cameras on the other hand, detect infrared radiation



(a) Number of images per sensor

(b) Number of labels per sensor

Figure 3: Number of images and labels per sensor, respectively.

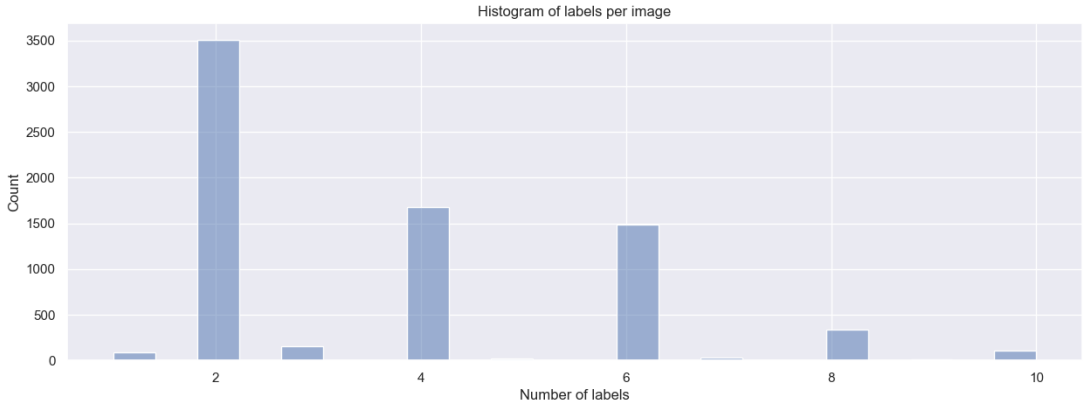


Figure 4: Track labels per image. Most images depict track pairs. However, there are also images with odd number of tracks.

Sensor	Widht [px]	Height [px]	Aspect ration
RGB low resolution	4112	2504	1.64
RGB high resolution	2464	1600	1.54
Infrared	640	480	1.33

Table 1: Size of images per sensor.

emitted by objects based on their temperature rather than visible light and are, therefore, used in low-light conditions or complete darkness[1]. The thermal radiation is converted into electrical signals which are then processed to a visual image that is visible to the human eye (Clark et al., 2002). Warmer areas are displayed as brighter, while cooler areas appear as darker shades of gray.

Emissivity, a material property that indicates how efficiently an object emits infrared radiation, plays a significant role in thermal imaging. Emissivity is measured on a scale from 0 to 1, where 0 indicates a perfect reflection of the radiation (no emission such as a mirror), and 1 indicates perfect emissivity (total emission in an object referred to as blackbody). Detecting rail tracks in infrared images is based on the principle that polished metallic surfaces such as tracks have a low emissivity, whereas organic materials that appear often in the background have a high emissivity.

3.1.2 Brightness of the images

In deep learning, the quality of the images can have a large impact on the efficiency. Image segmentation tasks, where the goal is to identify and classify each pixel in an image, are particularly sensitive to variations in pixel brightness and intensity. In this section, we analyze the

[1] All objects with a temperature greater than absolute zero emit infrared energy.

brightness of the images for each type of sensor. Brightness is defined as the average pixel intensity $\frac{1}{N}\sum_{i=1}^{N} I_i$; N is the number of pixels in the image, and I_i is the intensity of pixel i. In Figure 5 shows a series of box plots with brightness values per sensor.

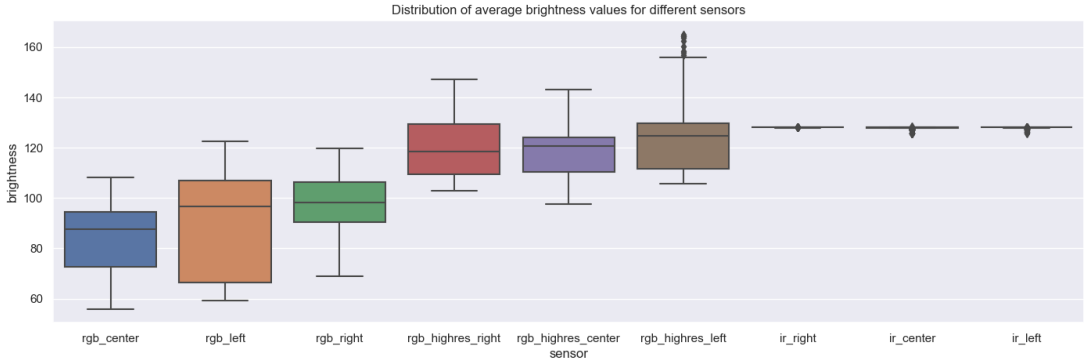


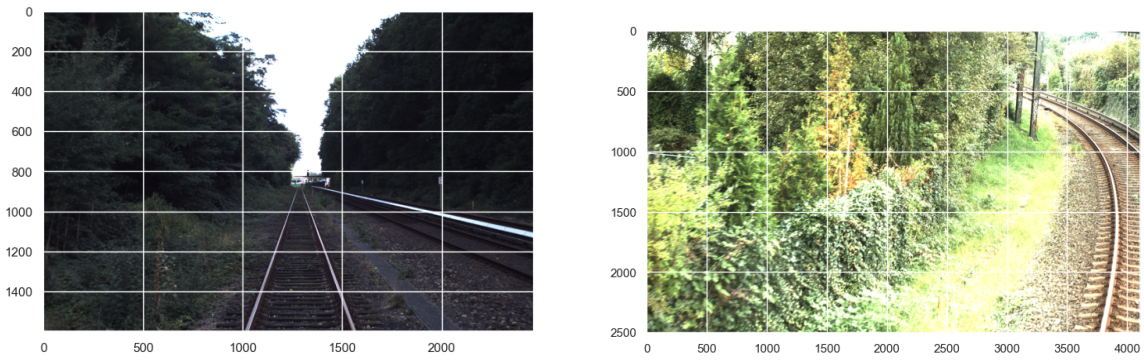
Figure 5: Brightness of images by sensor.

Among the three types of sensors, low resolution RGB cameras produce the darkest images (a black image has a value of 0, a white image has a value of 255). High resolution images are brighter on average, which can be explained by different exposure settings such as shutter speed and ISO sensitivity. Both low and high-resolution cameras feature pixels of equal size $(3.45\mu m)$, so the amount of light per pixel is the same. Infrared cameras produce images with almost constant brightness as the non-visible infrared image is mapped on a visible spectrum.

Figure 6 show one example of a bright and a dark image, respectively.

3.1.3 Entropy of the images

Shannon entropy is a measure from information theory that reflects the uncertainty or randomness associated with a set of data (Shannon, 1948). In the context of images, Shannon entropy can be used to quantify the complexity in the pixel values of an image. A high entropy value indicates higher complexity or randomness in the pixel values, while a low entropy value suggests



(a) Image with low brightness value.

(b) Image with high brightness value

Figure 6: Examples of very bright and very dark image, respectively.

more homogeneity. Rahane and Subramanian (2020) report a positive correlation between the entropy of the training data and the performance of semantic segmentation tasks, highlighting that more complex images are harder to learn by deep-learning networks. The Shannon entropy for a grayscale image is given by $H(X) = -\sum_{i=1}^{n} P(x_i) \cdot \log_2(P(x_i))$, where $P(x_i)$ is the probability of occurrence of pixel x_i (i.e., the number of pixels with intensity x_i divided by the total number of pixels). A box-plot with the entropy distribution is given in Figure 8 for each sensor. The maximum entropy value is $8 = \log_2(256)$ as we convert the images to grayscale with 256 different intensity levels.

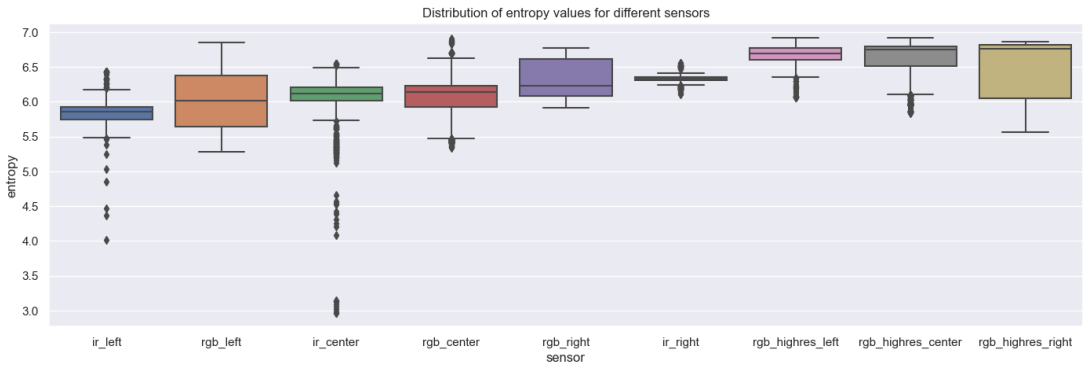


Figure 7: Entropy of images by sensor.

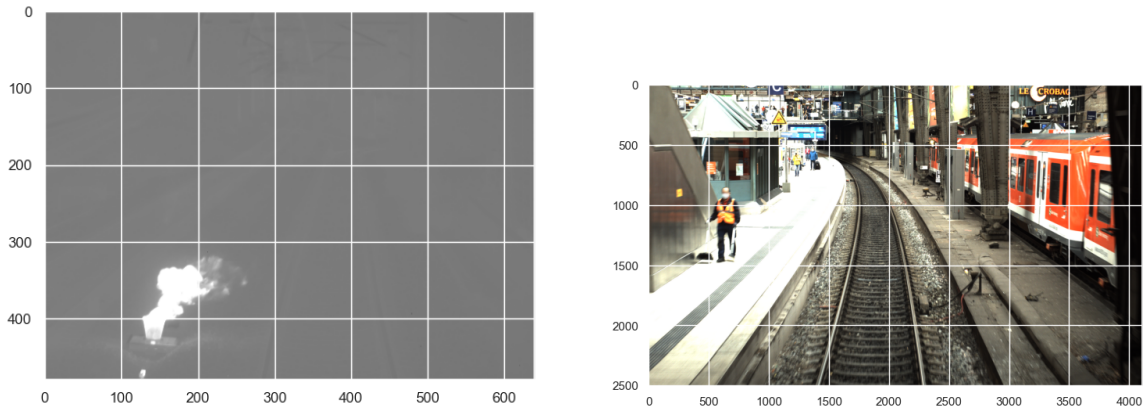
Overall, the images have a high entropy – the median values range from 5.9 to 6.8. High resolution images have the highest entropy. Infrared images do not seem to have lower entropy on average. However, certain images seem to have a very low randomness; and as it turns out also very low level of information when looking at Figure 8a. Figure 8 highlight the visual difference between a very low and a very high entropy image.

3.1.4 Occlusion

Certain track labels are hidden or occluded by other objects. One example is given in Figure 8b where the train on the right covers the tracks. In this section, we analyze the occlusion of the labels and examine those occlusions visually.

Figure 9 shows the number of labels with a given occlusion level. Most of the labels, 20.069, are not occluded at all or have only a slight occlusion. However, 320 labels in 196 images are marked with an occlusion level of 100%. Figure 10 shows two examples where the track labels are fully covered.

We keep all images in the dataset as the CNN might recognize that there has to be a track below a train and because most images with covered labels have visible labels as well.



(a) Image with low entropy value.

(b) Image with high entropy value

Figure 8: Examples of image with minimal and maximal entropy, respectively.

3.1.5 Images and video frames

The rail tracking systems aim to analyze video streams by treating each video frame as an independent image. Consequently, the images in the OSDaR23 dataset represent individual frames extracted from video sequences. Figure 11 shows three examples of video sequences with seven frames each. It is clear to see that there is only minor variation in the images.

Two major issues when training a model on a dataset that consists of similar images is a lack of generalization (i.e., inability to generalize well on diverse and unseen images) and reduced robustness (i.e., vulnerability to variations in lighting conditions and backgrounds). In order to mitigate the issues, we add images from the RailSem19 dataset to the training data.

3.2 RailSem19 dataset

The RailSem19 dataset (Zendel et al., 2019) is not the primary focus on of this thesis. However, the previous analysis reveals that among the 7,421 images within the OSDaR23 dataset, a significant number show high similarity. This similarity is due to the fact that the images are frames from a video sequence and the presence of three cameras for each orientation, respectively. The RailSem19 dataset is added to the training set in order to increase the performance of the segmentation approach. In the following, we will examine the RailSem19 dataset in more detail.

The dataset consists of 8500 rail images, taken in different countries, and weather and lighting conditions. The number of rail annotations is 58.483. All images have a size of 1920x1080 pixels. Figure 12 shows a histogram of images with the respective number of track labels. Most images contain 4 track labels (i.e., two pairs of tracks) but the dataset also contains complicated networks with 26 pairs of tracks. An example of a simple and an example of a complicated infrastructure is given in Figure 13.

Figure 14a and Figure 14b show the distribution of the brightness values and the entropy values as defined in Section 3.1, respectively. The RailSem19 images are mostly brighter than

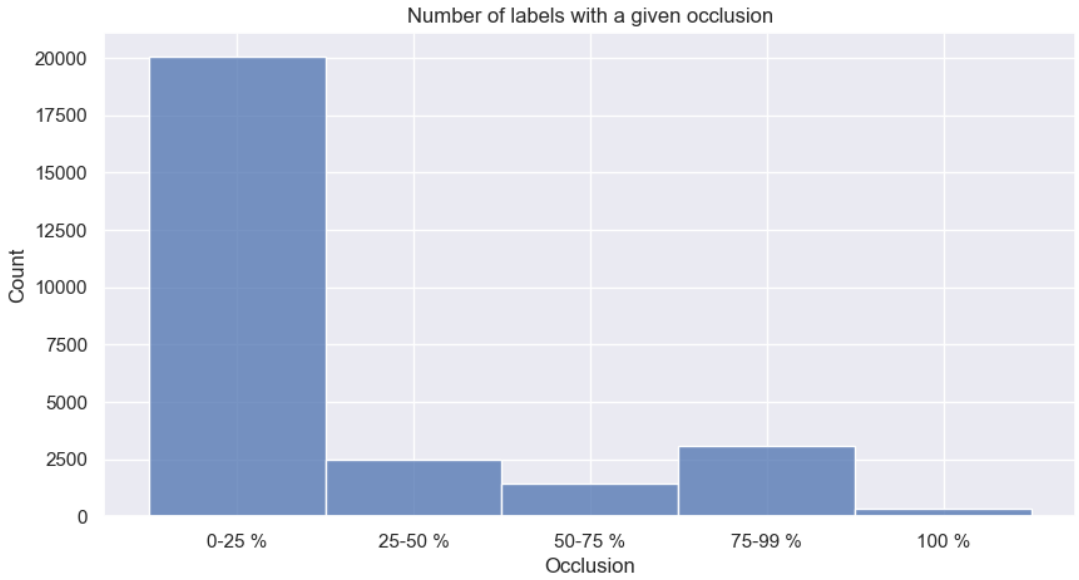
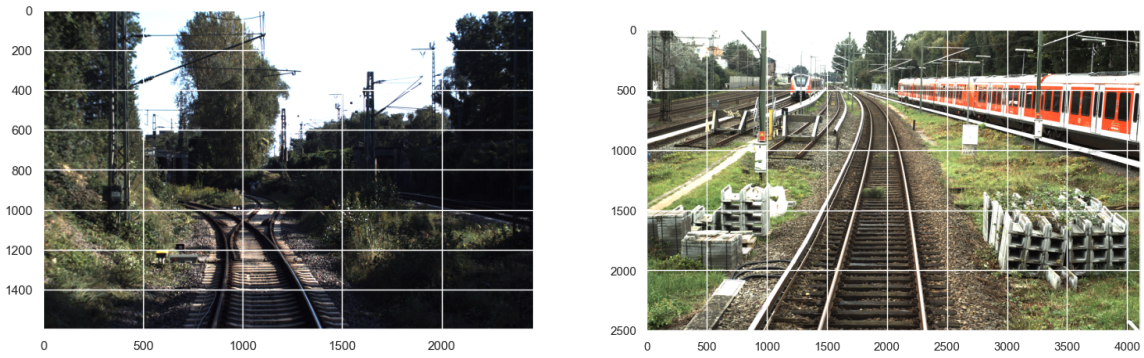


Figure 9: Histogram showing the occlusion level for track labels.



(a) Track on the right is hard to recognize due to poor lighting condition.

(b) Track is covered by train on the right.

Figure 10: Examples of track labels with 100% occlusion.

the OSDaR23; the distribution of the values ranges from almost black images to almost white images (see Figure 15a and Figure 15a). The entropy of the RailSem19 images is similar to the images in the OSDaR23 dataset. The outliers at the lower end of the entropy spectrum are associated with infrared images in the OSDaR23 dataset. Howerver, in RailSem19 low entropy images are often images that were generated in tunnels or at night. Figure **??** and Figure **??**) are images with a low entropy levels close to zero whereas Figure **??**) has the highest entropy level in the data set with a score of 6.97.

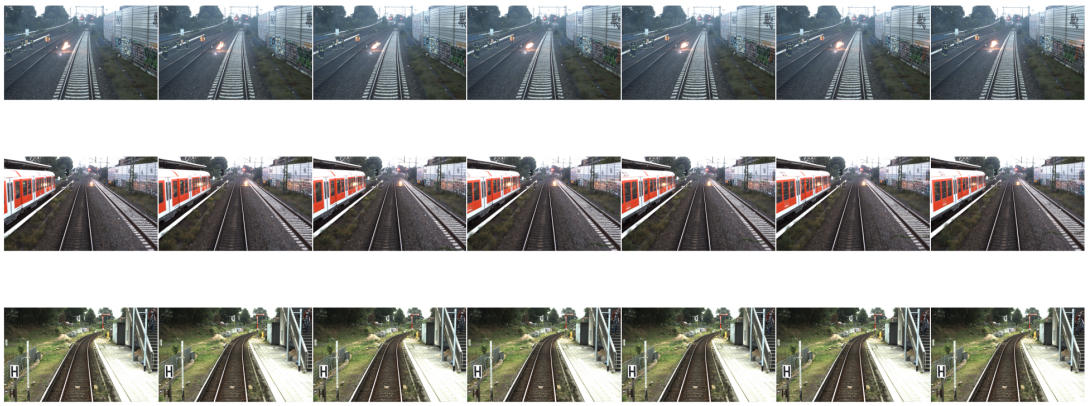


Figure 11: Three examples with seven video frames (i.e., images), respectively.

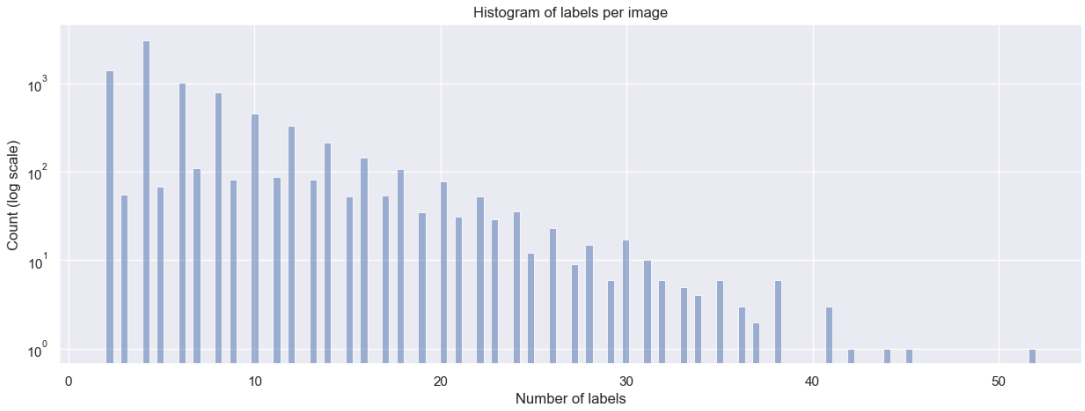


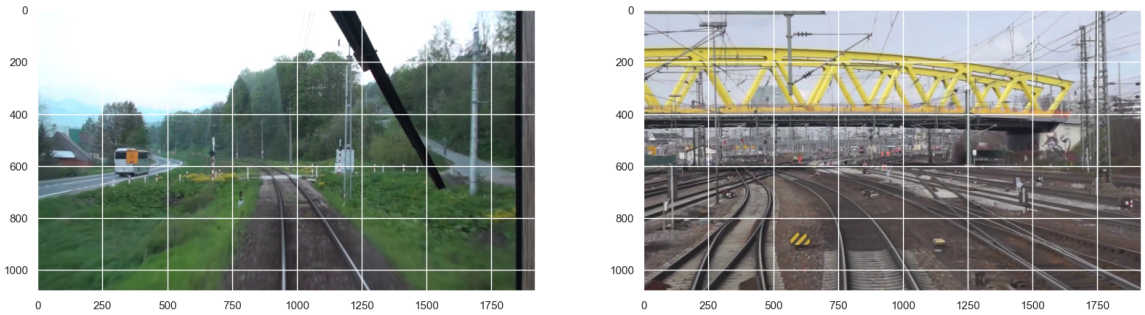
Figure 12: Track labels per image on a logarithmic scale. The images range from simple railroads with a single pair of tracks to complicated networks with 26 pairs of tracks.

3.3 Data splitting

Splitting a dataset into training, validation, and test sets is a standard practice in machine learning to evaluate and improve the performance of a model. Separating data into distinct sets and using them at different stages of development, prevents unintended data leakage where information from the test or validation set influences the training process.

The validation set assesses performance during training, refines hyperparameters, and guards against overfitting. Adjustments made based on validation set evaluations ensure the model generalizes effectively to new, unseen data.

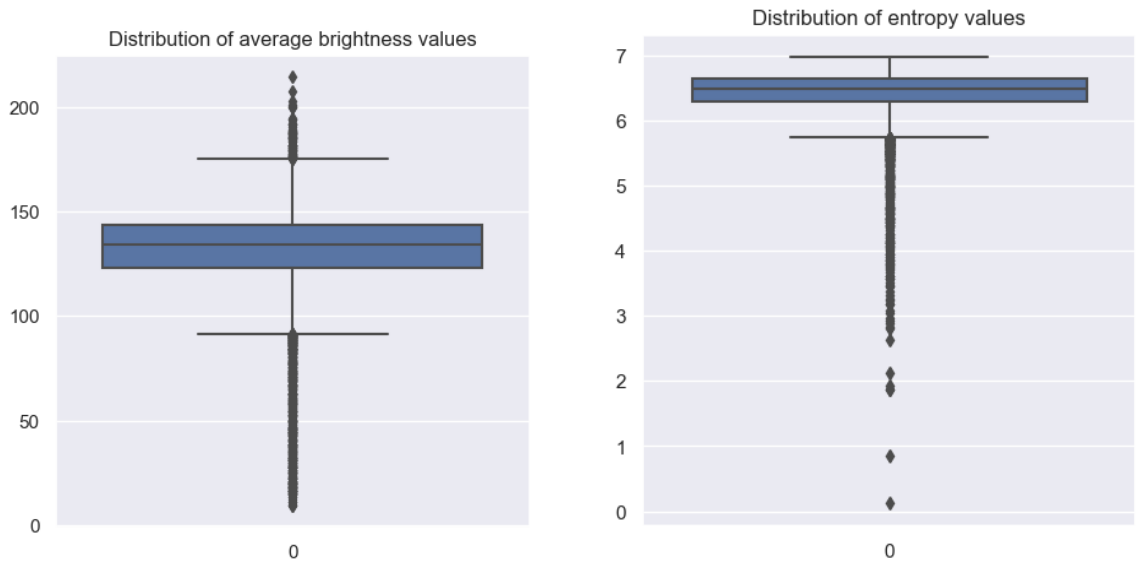
The training set is used to train the model – the model learns patterns and relationships within the training data. The validation set is used to evaluate the performance during the training process, fine-tune the hyperparameters of the model, and to prevent overfitting (i.e., a situation when a model performs well on the training data but fails to generalize to new data). We can make adjustments to the algorithm and optimize the performance on the validation set without compromising the model's ability to generalize to new data. The test set is reserved for the final evaluation of the model. The test set allows for an unbiased assessment of the model's



(a) Simple rail infrastructure.

(b) Complicated rail infrastructure.

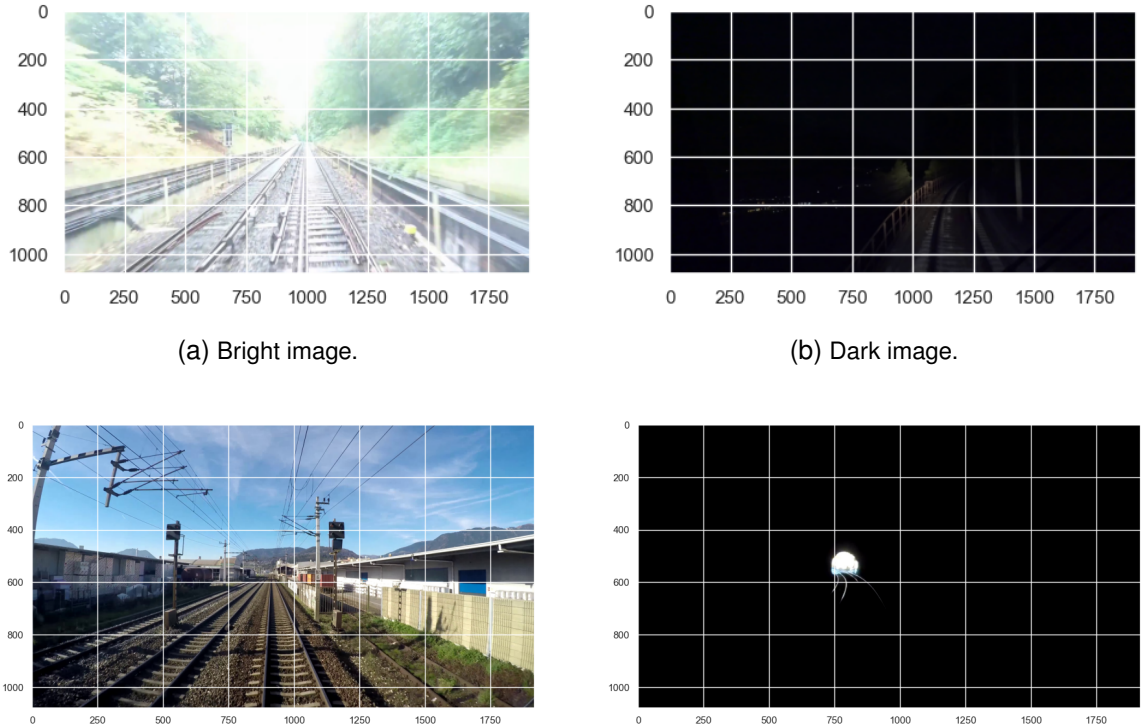
Figure 13: Examples of simple and complicated rail infrastructure.



(a) Boxplot of brightness values.

(b) Boxplot of entropy values.

Figure 14: Brightness and entropy of RailSem19 images.



(a) Bright image.

(b) Dark image.

(c) High entropy.

(d) Low entropy.

Figure 15: Extreme examples with regard to brightness and entropy.

performance on data it has never seen before and provides, therefore, an estimate of how well the model is likely to perform on new, real-world data.

Our approach is to perform a random split with a share of 70%/15%/15% for train, validation, and test set respectively. Splitting the RailSem19 set is straightforward; we randomly assign images to each set resulting in 5.950, 1.275, and 1.275 images per set, respectively. However, the OSDaR23 comprises many images with high similar (e.g., frames from a video that was take when the locomotive was standing still), so we perform a random split not on individual images but on videos sequences. The approach results in 5.506 images for training, 987 images for validation, and 928 images for testing. The assignment of video sequences to the respective set and the split on sensor level is given in Appendix A.1

4 Solution approach and experiments

4.1 Modeling and performance evaluation

4.2 Transforming labels

4.3 Non-AI based segmentation

4.4 Deep-learning based segmentation

5 Results

6 Conclusion

Etwas Text... Hier kommen noch einige Abkürzunge vor zum Beispiel Alphabet (ABC),world wide web (WWW) und Rolling on floor laughing (ROFL).

6.1 Algorithms

Use a defined environment for algorithms.

Algorithm 1 is an example from the gallery (https://www.overleaf.com/latex/examples/euclids-algorithm-an-example-of-how-to-write-algorithms-in-latex/mbysznrmktqf) .

Algorithm 1 Euclid's algorithm

```
1: procedure EUCLID(a, b)                          ▷ The g.c.d. of a and b
2:     r ← a mod b
3:     while r ≠ 0 do                              ▷ We have the answer if r is 0
4:         a ← b
5:         b ← r
6:         r ← a mod b
7:     return b                                    ▷ The gcd is b
```

Bibliography

Emilio J. Almazàn, Ron Tal, Yiming Qian, and James H. Elder. Mcmlsd: A dynamic programming approach to line segment detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5854–5862, 2017. doi: 10.1109/CVPR.2017.620.

ALSTOM Transport SA. Autonomous mobility: The future of rail is automated, 2021. URL https://www.alstom.com/autonomous-mobility-future-rail-automated.

Mostafa Arastounia. Automated recognition of railroad infrastructure in rural areas from lidar data. *Remote Sensing*, 7(11):14916–14938, 2015. ISSN 2072-4292. doi: 10.3390/rs71114916. URL https://www.mdpi.com/2072-4292/7/11/14916.

Jieren Cheng, Hua Li, Dengbo Li, Shuai Hua, and Victor S. Sheng. A survey on image semantic segmentation using deep learning techniques. *Computers, Materials & Continua*, 74(1):1941–1957, 2023. ISSN 1546-2226. doi: 10.32604/cmc.2023.032757. URL http://www.techscience.com/cmc/v74n1/49879.

M Clark, D.M McCann, and M.C Forde. Infrared thermographic investigation of railway track ballast. *NDT & E International*, 35(2):83–94, 2002. ISSN 0963-8695. doi: https://doi.org/10.1016/S0963-8695(01)00032-9. URL https://www.sciencedirect.com/science/article/pii/S0963869501000329.

Deutsche Bahn AG. Automatic train operation (ato), 2022. URL https://digitale-schiene-deutschland.de/en/Automatic-Train-Operation.

Deutsche Bahn AG. First freely available multi-sensor data set for machine learning for the development of fully automated driving: Osdar23, 2023. URL https://digitale-schiene-deutschland.de/en/news/2023/OSDaR23-multi-sensor-data-set-for-machine-learning.

Richard O. Duda and Peter E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Commun. ACM*, 15(1):11–15, jan 1972. ISSN 0001-0782. doi: 10.1145/361237.361242. URL https://doi.org/10.1145/361237.361242.

Europe's Rail Joint Undertaking. Innovation in the spotlight: Towards unattended mainline train operations (ato goa 4), 2019. URL https://rail-research.europa.eu/highlight/innovation-in-the-spotlight-towards-unattended-mainline-train-operations-ato-goa4/.

Xavier Giben, Vishal M. Patel, and Rama Chellappa. Material classification and semantic segmentation of railway track images with deep convolutional neural networks. In *2015*

IEEE International Conference on Image Processing (ICIP), pages 621–625, 2015. doi: 10.1109/ICIP.2015.7350873.

Rafael Grompone von Gioi, Jeremie Jakubowicz, Jean-Michel Morel, and Gregory Randall. Lsd: A fast line segment detector with a false detection control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(4):722–732, 2010. doi: 10.1109/TPAMI.2008.300.

Rafael Grompone von Gioi, Jérémie Jakubowicz, Jean-Michel Morel, and Gregory Randall. LSD: a Line Segment Detector. *Image Processing On Line*, 2:35–55, 2012. https://doi.org/10.5201/ipol.2012.gjmr-lsd.

International Energy Agency. The future of rail, 2019. URL https://www.iea.org/reports/the-future-of-rail.

Dewan Md Zahurul Islam, Stefano Ricci, and Bo-Lennart Nelldal. How to make modal shift from road to rail possible in the european transport market, as aspired to in the eu transport white paper 2011. *European transport research review*, 8(3):1–14, 2016.

Fatih Kaleli and Yusuf Sinan Akgul. Vision-based railroad track extraction using dynamic programming. In *2009 12th International IEEE Conference on Intelligent Transportation Systems*, pages 1–6, 2009. doi: 10.1109/ITSC.2009.5309526.

B. Le Saux, A. Beaupère, A. Boulch, J. Brossard, A. Manier, and G. Villemin. Railway detection: From filtering to segmentation networks. In *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, pages 4819–4822, 2018. doi: 10.1109/IGARSS.2018.8517865.

Xinpeng Li and Xiaojiang Peng. Rail detection: An efficient row-based network and a new benchmark. In *Proceedings of the 30th ACM International Conference on Multimedia*, MM '22, page 6455–6463, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450392037. doi: 10.1145/3503161.3548050. URL https://doi.org/10.1145/3503161.3548050.

Chao Ma and Mei Xie. A method for lane detection based on color clustering. In *2010 Third International Conference on Knowledge Discovery and Data Mining*, pages 200–203, 2010. doi: 10.1109/WKDD.2010.118.

Annika Meyer, Philipp Skudlik, Jan-Hendrik Pauls, and Christoph Stiller. Yolino: Generic single shot polyline detection in real time. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2916–2925, 2021.

Bogdan Tomoyuki Nassu and Masato Ukai. Rail extraction for driver support in railways. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 83–88, 2011. doi: 10.1109/IVS.2011.5940410.

I Pagand, C Carr, and J Doppelbauer. Fostering the railway sector through the european green deal. *European Union Agency For Railways*, 2020.

Xingang Pan, Jianping Shi, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Spatial as deep: Spatial cnn for traffic scene understanding. In *AAAI Conference on Artificial Intelligence (AAAI)*, February 2018.

Andrei I. Purica, Beatrice Pesquet-Popescu, and Frederic Dufaux. A railroad detection algorithm for infrastructure surveillance using enduring airborne systems. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2187–2191, 2017. doi: 10.1109/ICASSP.2017.7952544.

Zhiquan Qi, Yingjie Tian, and Yong Shi. Efficient railway tracks detection and turnouts recognition method using hog features. *Neural Computing and Applications*, 23:245–254, 2013.

Ameet Annasaheb Rahane and Anbumani Subramanian. Measures of complexity for large scale image datasets. In *2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pages 282–287. IEEE, 2020.

P.K Sahoo, S Soltani, and A.K.C Wong. A survey of thresholding techniques. *Computer Vision, Graphics, and Image Processing*, 41(2):233–260, 1988. ISSN 0734-189X. doi: https://doi.org/10.1016/0734-189X(88)90022-9. URL https://www.sciencedirect.com/science/article/pii/0734189X88900229.

C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948. doi: 10.1002/j.1538-7305.1948.tb01338.x.

Rustam Tagiew, Martin Köppel, Karsten Schwalbe, Patrick Denzler, Philipp Neumaier, Tobias Klockau, Martin Boekhoff, Pavel Klasek, and Roman Tilly. Osdar23: Open sensor data for rail 2023. *arXiv preprint arXiv:2305.03001*, 2023.

Jigang Tang, Songbin Li, and Peng Liu. A review of lane detection methods based on deep learning. *Pattern Recognition*, 111:107623, 2021.

Zhu Teng, Feng Liu, and Baopeng Zhang. Visual railway detection by superpixel based intracellular decisions. *Multimedia Tools and Applications*, 75:2473–2486, 2016.

Tugce Toprak, Burak Belenlioglu, Burak Aydın, Cuneyt Guzelis, and M. Alper Selver. Conditional weighted ensemble of transferred models for camera based onboard pedestrian detection in railway driver support systems. *IEEE Transactions on Vehicular Technology*, 69(5):5041–5054, 2020. doi: 10.1109/TVT.2020.2983825.

TuSimple. Tusimple, 2017. URL https://github.com/TuSimple/tusimple-benchmark/tree/master/doc/lane_detection.

Jinsheng Wang, Yinchao Ma, Shaofei Huang, Tianrui Hui, Fei Wang, Chen Qian, and Tianzhu Zhang. A keypoint-based global association network for lane detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1392–1401, 2022.

Yin Wang, Lide Wang, Yu Hen Hu, and Ji Qiu. Railnet: A segmentation network for railroad detection. *IEEE Access*, 7:143772–143779, 2019. doi: 10.1109/ACCESS.2019.2945633.

Bisheng Yang and Lina Fang. Automated extraction of 3-d railway tracks from mobile laser scanning point clouds. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(12):4750–4761, 2014. doi: 10.1109/JSTARS.2014.2312378.

Mingyue Yang. Lane detection methods survey for automatic driving. In *Journal of Physics: Conference Series*, volume 2547, page 012015. IOP Publishing, 2023.

Syed Sahil Abbas Zaidi, Mohammad Samar Ansari, Asra Aslam, Nadia Kanwal, Mamoona Asghar, and Brian Lee. A survey of modern deep learning based object detection models. *Digital Signal Processing*, 126:103514, 2022. ISSN 1051-2004. doi: https://doi.org/10.1016/j.dsp.2022.103514. URL https://www.sciencedirect.com/science/article/pii/S1051200422001312.

Oliver Zendel, Markus Murschitz, Marcel Zeilinger, Daniel Steininger, Sara Abbasi, and Csaba Beleznai. Railsem19: A dataset for semantic rail scene understanding. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1221–1229, 2019. doi: 10.1109/CVPRW.2019.00161.

Tu Zheng, Yifei Huang, Yang Liu, Wenjian Tang, Zheng Yang, Deng Cai, and Xiaofei He. Clrnet: Cross layer refinement network for lane detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 898–907, 2022.

List of Figures

List of Tables

List of source codes

List of Abbreviations

ABC Alphabet

WWW world wide web

ROFL Rolling on floor laughing

ATO Automatic Train Operations

CNN Convolutional Neural Network

A Appendix A

A.1 Dataset split

Splitting the OSDaR23 dataset in to train, validation, and test subsets is performed based on video sequences. Table 2 lists the assignment of sequences to the respective subset. Figure 16 show the split on an image level in total and for each sensor type.

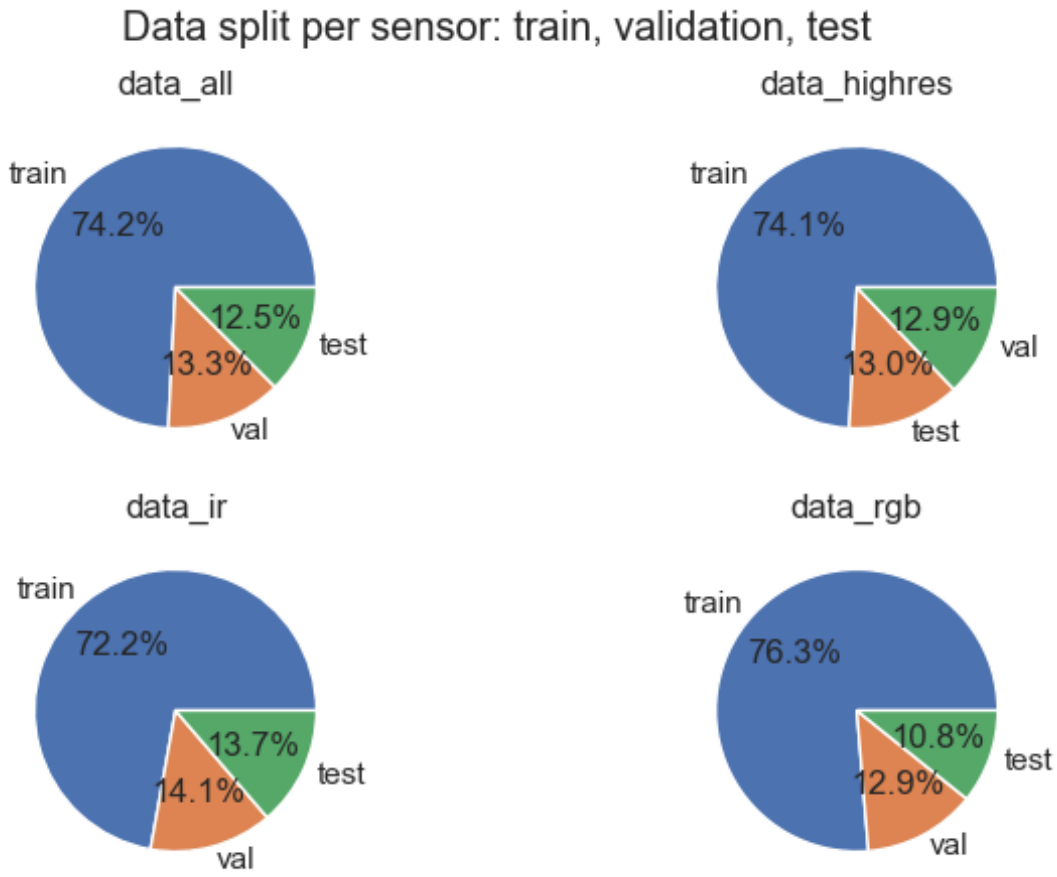


Figure 16: OSDaR23 data split on image level in total and on sensor level.

Train	Validation	Test
10_station_suelldorf_10.1	16_under_bridge_16.1	14_signals_station_14.1
11_main_station_11.1	18_vegetation_switch_18.1	21_station_wedel_21.1
12_vegetation_steady_12.1	1_calibration_1.2	4_station_pedestrian_bridge_4.2
13_station_ohlsdorf_13.1	3_fire_site_3.3	4_station_pedestrian_bridge_4.5
14_signals_station_14.2	9_station_ruebenkamp_9.1	7_approach_underground_station_7.2
14_signals_station_14.3	9_station_ruebenkamp_9.2	8_station_altona_8.3
15_construction_vehicle_15.1		
17_signal_bridge_17.1		
19_vegetation_curve_19.1		
1_calibration_1.1		
20_vegetation_squirrel_20.1		
21_station_wedel_21.2		
21_station_wedel_21.3		
2_station_berliner_tor_2.1		
3_fire_site_3.1		
3_fire_site_3.2		
3_fire_site_3.4		
4_station_pedestrian_bridge_4.1		
4_station_pedestrian_bridge_4.3		
4_station_pedestrian_bridge_4.4		
5_station_bergedorf_5.1		
5_station_bergedorf_5.2		
6_station_klein_flottbek_6.1		
6_station_klein_flottbek_6.2		
7_approach_underground_station_7.1		
7_approach_underground_station_7.3		
8_station_altona_8.1		
8_station_altona_8.2		
9_station_ruebenkamp_9.3		
9_station_ruebenkamp_9.4		
9_station_ruebenkamp_9.5		
9_station_ruebenkamp_9.6		
9_station_ruebenkamp_9.7		

Table 2: Dataset split: OSDaR23

B Appendix B