

CNN-based classification of scale deposits and damages in drainage pipes using a mobile inspection rover

*Veronika Putz¹, Christina Hofer¹, Sabrina Fleischanderl¹, Leander B. Hörmann¹,
Johannes Schröck¹, Wolfgang Hofer¹, Harald Hölzl², Tobias Schachinger², Vesna Micic Batka²*
¹Linz Center of Mechatronics GmbH, Altenberger Straße 69, A-4040 Linz
²ÖBB-Infrastruktur AG, Praterstern 3, A-1020 Wien

Abstract

A convolutional neural networks (CNN)-based approach for automated classification of scale deposits and structural damage in drainage pipes of railway tunnels is presented and validated using data collected by a mobile inspection rover - a waterproof, battery-powered robot equipped with cameras that record video data for offline analysis. For evaluation, two deep learning approaches are considered: image classification using VGG16 and object detection using YOLOv11. Both models were trained and validated using data acquired during field tests in Austrian railway tunnels. While VGG16 enables simple frame-wise classification, YOLOv11 provides detailed spatial information and allows parallel detection of defects and various severity levels of scaling per frame. Results showed reliable detection of severe defects, with some confusion between adjacent severity levels. To support planning of drainage pipes maintenance, a post-processing method was introduced that transformed frame-wise detections into a time-series based indicator implying the condition of pipe segments. The approach demonstrated the potential of combining autonomous inspection with machine learning in the future for condition-based and predictive maintenance of tunnel drainage systems.

Keywords: Predictive Maintenance, Fault Detection, Autonomous Inspection, Machine Learning, Tunnel Drainage, Scale Deposits

Introduction

In pressure-relieved railway tunnels, drainage systems are installed to protect tunnel structures against hydrostatic pressure caused by groundwater. Drainage is typically realized using slotted pipes that run parallel to each side of the tunnel. The long-term functionality of these drainage systems can be significantly impaired by the formation of deposits within the pipes: Especially at inner pipe surfaces and within the drainage slots, carbonate minerals, hereafter referred to as "scale deposits", can accumulate. In the drainage system, calcium carbonate originates primarily from mineralized groundwater as well as from leaching of cement used in the tunnel construction [1,2]. The rate at which the scale deposits form depends among other factors, on precipitation and the resulting variations in groundwater inflow. Under favorable conditions, deposits can rapidly accumulate within a short period of time. As a result, the effective cross-section of the drainage system reduces, and the drainage slots can be clogged by the accumulated deposits. To restore the functionality of the drainage pipes and to avoid damage, regular drainage cleaning is required. Such main-

tenance work is costly and usually requires closing the railway tunnel for traffic.

We are therefore developing an automated inspection system capable of autonomously inspecting the drainage and operating without tunnel closures. Regular automated inspections, combined with systematic classification and spatial mapping of scale deposits and potential damages within the piping system would allow monitoring growth rates of scale and hence would support not only the condition-based maintenance, but also predictive maintenance strategies.

The development of such an inspection system, capable of autonomously traversing tunnel drainage pipes, is associated with several technical challenges [3,4]. The used drainage pipes typically are made of thermoplastics, have a circular or bottom-flattened cross-section and have narrow diameters between 160 mm and 300 mm. These diameters may already be constricted by scale deposits, reducing the available cross-section for an inspection system. Lack of reliable radio communication within the pipes requires the system to move autonomously through the drainage despite pipe bends and branching points. It must be able to orientate, identify obstacles, and to collect meaningful data that

enables an assessment of the conditions inside the pipe. Furthermore, the drive of the inspection system must be able to cope with difficult conditions: Scale deposits at the bottom of the pipe can exhibit different consistencies, ranging from hard, consolidated layers to soft, sludge-like material [1], and can hinder the motion of an inspection system.

In this work, we present a demonstrator capable of moving through the drainage pipes and collecting meaningful data. An overview of feasible AI methods for classification of scale and damages in video data is provided, and two different models are implemented and validated using data collected in real railway tunnels. Finally, a method to post-process raw results, providing easy-to-read visualization for the state of the drainage system, is shown.

Autonomous inspection rover

The centerpiece of our approach is a field-proven demonstrator - a waterproof robot, that moves through drainage pipes autonomously. It is wireless, battery-powered, and equipped with sensors for obstacle detection essential for autonomous movement and recording of video data for offline evaluation. The demonstrator builds upon an initial concept, which was introduced in [3], but has been significantly improved based on experience from field tests.

Figure 1 above shows a technical drawing of the demonstrator. It consists of the rover, equipped with a crawler-track drive capable of moving forward and backward, and a sensor lance attached to its front. Figure 1 below shows the demonstrator, being immersed in a maintenance hole of the drainage system at the beginning of an inspection drive.

The sensor lance carries a red ring laser and an integrated camera, which are used for obstacle detection. At its tip, the lance contains a second integrated camera, which is used for recording video data from within the drainage pipe. An Arducam OV5647 camera module with a wide-angle lens (160° diagonal field of view), connected to a Raspberry Pi Zero 2W board, is used. A white-light LED source placed at the tip of the sensor lance provides the necessary background illumination for video recordings in the drainage pipe with no ambient light. During inspection, videos from this camera are captured at a resolution of 1920 x 1080 pixel and a frame rate of 30 frames per second (fps). Videos are stored locally on the demonstrator for offline analysis.

To allow building a more generalized database for defect detection and classification, instead of the sensor lance, a high-quality external camera (GoPro 10) can

be clamped onto the demonstrator, recording videos with a resolution of 3840 x 2160 pixel at a frame rate of 30 fps. Data acquired using these two cameras was used as input for training AI models for detecting scale deposits and defects like damaged pipe walls.

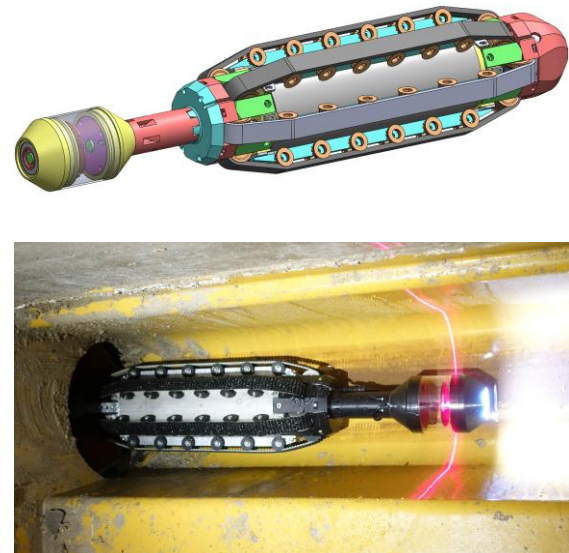


Figure 1: Technical drawing of the inspection rover (above), rover immersed in the drainage system (below)

During the inspection, the drive of the demonstrator is set to a constant velocity of approximately 0.3 m/s. However, its exact velocity may vary, since unknown drag, due to sludge-like sintering, low-traction sections and various obstacles, may reduce the set velocity. The true speed of the demonstrator is not yet measured, and the location, at which an individual video frame is recorded inside the drainage, can only roughly be reconstructed from its time stamp. In the next steps we plan to integrate detection of drainage branching points and maintenance holes as well as velocity estimation methods for improved location accuracy of the robot within the pipe.

Recorded data

The demonstrator was used to collect data in drainage systems of two operational railway tunnels in Austria during short maintenance closures:

- In the “Pummersdorfer Tunnel”, approx. 30 minutes of video data were collected using both cameras. Nine measurement runs were conducted with and against the flow of the water, starting at different entry points. A total length of approx. 550 m was covered. In this tunnel, only

slight scale and small isolated damages in the pipe walls occurred.

- In the “Sieberg Tunnel”, in four segments of the tunnel, 10 measurement runs were conducted with and against the flow of the water using both cameras, providing a total length of 37 minutes of video data. In the videos obtained, severe examples of both scaling and damage could be captured.

The structure and appearance of scale deposits can considerably vary, depending on their chemical composition [1]. Additionally, the color of the pipe material and the shape of the cross section of the pipe (circular, flattened) can differ in the field. The video material available from the experiments carried out covers various scenarios. Some examples are shown in Figures 2 to 5. Figure 2 shows a pipe made of blue thermoplastic material, partially filled with water, with scaling as well as a hole in the pipe wall visible on the right. Figure 3 shows a similar scene with more severe scaling in a pipe with reddish-brown walls.



Figure 2: Medium to severe scaling, visible hole in the pipe wall



Figure 3: Medium to severe scaling at the bottom, slight scaling at the side wall

A different scenario with severe scaling as well as formation of algae is shown in Figure 4, while Figure 5 contains isolated scaling occurring only at one side wall of the pipe. Note that Figures 4 and 5 were recorded using the high-quality external camera, while Figures 2 and 3 were recorded using the low-quality camera integrated in the sensor lance.



Figure 4: Medium to strong scaling at the bottom, formation of algae



Figure 5: Strong scaling with reddish color at the side wall of the pipe

Relevant approaches for detection and classification of scale and damage

Considering the strong variability in the appearance of scale and damages as well as possible variations in color and shape of the drainage pipes, convolutional neural networks (CNNs) provide a more robust and adaptive approach for classifying the state of the drainage system, compared to classical image analysis methods. Three fundamental approaches can be distinguished:

- **CNNs for image classification** like VGG16 [5], InceptionNet [6] and ResNet [7] assign a single class label to an entire image. Each video frame is treated as an independent image and receives one class prediction. Models only provide information about the presence of a class in the video frame but do not localize objects or indicate their position or size within the image.
- **CNNs for object detection and classification** like R-CNN [8], SSD [9,10], or YOLO [11], detect and classify multiple

objects within a single video frame. These models return bounding boxes that mark the position and size of each detected object along with the corresponding class label and a confidence score. Several instances from multiple classes can be detected within a single video frame. Additional information regarding the position and size of a detected object is available.

- **CNNs for image segmentation** provide pixel-level classification and therefore allow a more detailed analysis of objects. In *semantic segmentation* models such as U-Net [12, 13], each pixel in an image frame is assigned to a class, resulting in a segmentation map that distinguishes different object categories but does not separate individual objects of the same class. *Instance segmentation* models such as Mask R-CNN [14] detect individual objects and generate a separate segmentation mask for each instance, enabling precise localization of object boundaries.

Each of the three fundamental approaches provides a different level of information and is related to different requirements regarding training data volume, annotation effort and computational complexity: Image classification models require comparatively simple annotations - only a single label per video frame needs to be assigned. Object detection approaches require bounding box annotations for each object instance, while segmentation methods demand pixel-level labeling, which significantly increases the manual labeling effort as well as the amount of required training data. For the use-case in our study, defects such as severe damage or scale only rarely occur along the length of the pipes, leading to low availability of relevant training data for this class. Therefore, only two of the three fundamental approaches are considered for further use at this stage: VGG16 (due to its architectural simplicity, stable behavior, and ease of use) and YOLOv11 [15] (due to its ability for fast and accurate object detection).

Annotation of training data

Defect classes for scale deposits and damage were defined in consultation with field experts responsible for inspecting and maintaining tunnel drainage systems. According to their recommendation, the acquired dataset was annotated manually, using different severity levels for both damages and scale deposits. Due to the different working principles of

VGG16 and YOLOv11, annotation was handled individually for the two models.

For VGG16, each video frame can only have a single class label. It was decided to label frames according to the most severe visible error in the frame (e.g., if the same video frame contains instances of moderate and severe scaling, the label “severe scaling” is assigned). Since instances of scaling and damage may occur in the same video frame, two individual VGG16 models were used for classifying scaling and damages. Each model needed to be trained independently. Consequently, also two independent label sets were created for the training data. The following class labels were assigned:

- Label set 1 (damage): slight | moderate | severe
- Label set 2 (scaling): slight | moderate | severe

For object detection using YOLO, the release YOLOv11-OBb¹ was selected. It allows defining oriented bounding boxes, which - due to the given camera viewing angle - is highly beneficial in our use-case. For this model, if applicable, multiple defects were annotated in each individual frame by drawing bounding boxes around every visible defect. Size and orientation of the bounding boxes were adapted to best fit the visible size of the defect or scaling. The following labels were assigned to a single label set:

- Scaling: slight | moderate | severe
- Damage: slight | moderate | severe | very severe

Note that for object detection using YOLO, an additional class for ‘very severe’ damage was introduced. This class is only applied to rare cases in which e.g., a larger fragment of the pipe wall is damaged or even missing. Due to the lack of the ability to represent multiple classes within a single frame, this was omitted for VGG16; corresponding damages were instead included in the ‘severe’ class.

Classification of data using VGG16

Two individual VGG16 models for detecting damage and scale deposits were trained, using 2073 images for training, 445 images for validation, and 445 images (not part of the training data) were used to test the performance of the obtained models. Table 2 lists the total occurrence of each error for damage and scaling for the various error classes in the test dataset.

¹ <https://github.com/ultralytics/ultralytics>

Tab. 1: Total occurrence of each class in the test dataset used for VGG16

Error classes	Total occurrences	
	Damage	Scaling
none	86	86
slight	214	214
moderate	120	120
severe	25	25

The respective performance of the models for detection of damage and scaling is shown using normalized confusion matrices (see Figures 6 and 7). The classification result obtained from the models is good: For each class, more than 90% occurrences were detected correctly. Confusion between classes rarely occurs and is limited to confusion between neighboring severity levels.

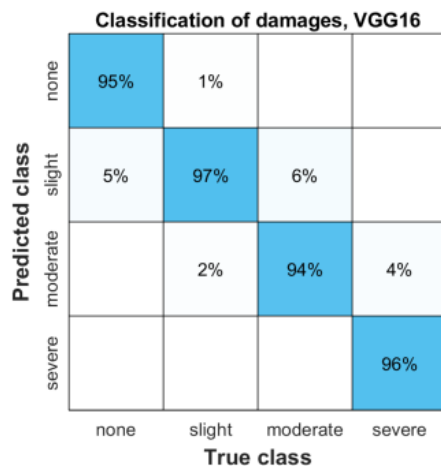


Figure 6: Normalized confusion matrix for classification of damages

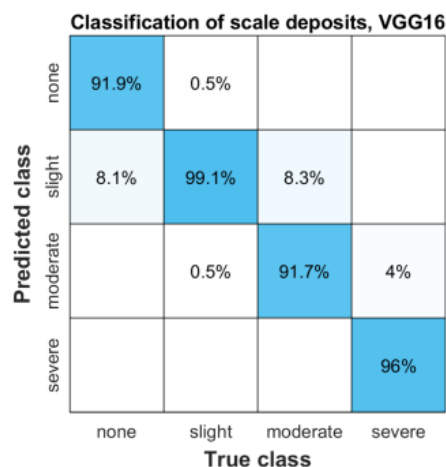


Figure 7: Normalized confusion matrix for classification of scaling

The obtained accuracy, which is calculated considering absolute occurrences of correct and false classifications, is 95.73% for damages and 95.51% for scaling.

Classification of data using YOLOv11

A single YOLOv11 model was trained using 609 video frames for training and 122 video frames for test. The datasets for training and test were selected to be mutually exclusive. Table 2 lists the total occurrence of each error class in the validation dataset in terms of video frames and in terms of instances. It is obvious that more than one instance of each class occurs in a single video frame.

Tab. 2: Total occurrence of each class in the test dataset used for YOLOv11

Error classes		Total occurrences	
		Frames	Instances
damage	slight	22	44
	medium	23	46
	severe	6	11
	very severe	3	6
scaling	slight	61	325
	medium	98	399
	severe	59	189

Figure 8 shows the obtained confusion matrix. Note that in Fig. 8, the abbreviation of row and column labels 'mod.' and 'v.s.' stand for 'moderate' and 'very severe', respectively. Confusion occurs between slight and moderate damages, as well as between slight, moderate and severe scaling. Severe and very severe damages were perfectly recognized (however, the total occurrence of these error classes in the test dataset is low). There is no confusion between damage and scaling.

The obtained accuracy, which is calculated considering absolute occurrences of correct and false classifications, is 70.3% and comparably low (mainly due to confusion between the slight and moderate classes). Combining slight and moderate damages and slight and moderate scaling would increase accuracy to 91.9%.

Figure 9 shows a sample video frame with the defects detected by the CNN marked as overlay. The benefit of the method, compared to VGG16, is obvious: The ability to detect multiple errors per video frame leads to an improved assessment of the overall area of the drainage pipe affected by scaling or damage within the image, which could not have been provided using the simple classification obtained from VGG16.

Predicted class	damages	v.s.	True class			
			damages			
			slight	mod.	severe	v. s.
	slight	25%	17%			
	mod.	75%	83%			
	severe			100%		
scale deposits	v.s.	True class				
		scale deposits				
		slight	mod.	severe	v. s.	
slight				67%	18%	
mod.				33%	75%	29%
severe					7%	71%

Figure 8: Normalized confusion matrix for classification of damages and scaling using YOLOv11.



Figure 9: Video frame with detected artifacts from various classes

Metric for visualizing scaling and damage

For the planning of maintenance measures based on recorded videos, raw detections of scaling and damage (as visible in Figure 9 for a single video frame) need to be further processed to provide an easy-to-read indicator for maintenance planning. A metric that visualizes the condition of an inspected section of the drainage pipe is required.

For this purpose, the total number of detected error events (scale deposits and damages of different severity levels) per frame is treated as time-series signal, which can be post-processed to obtain a meaningful indicator for the condition of the drainage pipe. Figure 10 shows the total number of detections per frame of a selected video obtained using the trained YOLOv11-OBb model described above. The video has a total duration of approx. 25 seconds (~750 video frames). It was recorded in a pipe segment with significant

scaling and several damages. A smoothed yet representative result is obtained by block-wise calculation of the median value over 15 consecutive frames, resembling a time segment of 0.5 s (Fig. 10, green line). Blocks are built with 7 frames overlapping. To avoid flattening the visible peaks in the detections per frame – especially in segments with a high density of defects – instead of the median value, the 90% percentile value can be used (Fig. 10, green line).

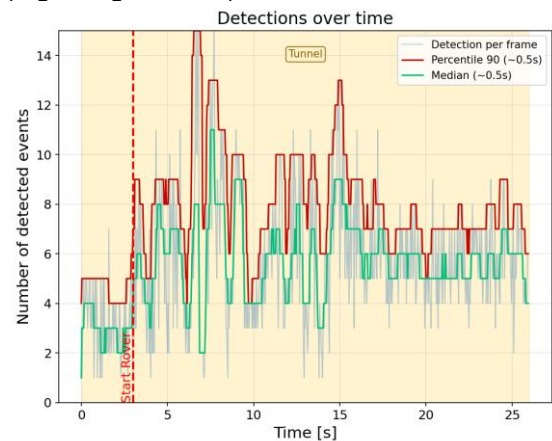


Figure 10: Total number of detections per frame vs. time stamp (raw detections and two indicators).

The obtained indicators are validated using an example image: In Figure 10, between 7 and 8 seconds after starting the experiment, a larger density of detected errors can be seen. Figure 11 shows a video frame recorded between these timestamps with the obtained detections of the CNN marked using color-coded bounding boxes, with many visible defects.



Figure 11: Sample video frame with many detections

Figure 12 shows a similar evaluation for a video recorded in a pipe segment, in which almost no errors (either scaling or damages) are visible. A sample video frame recorded approximately 37 seconds after starting the experiment is shown in Figure 13.

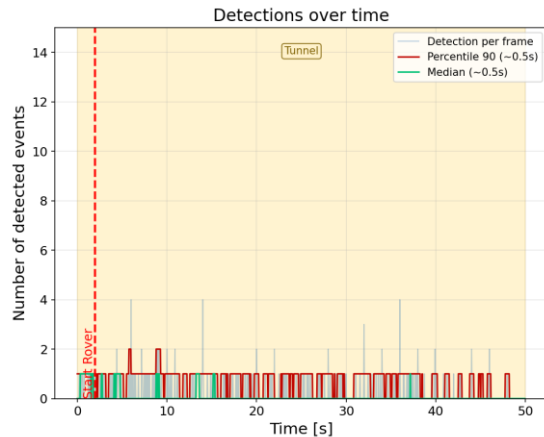


Figure 12: Total number of detections per frame vs. time stamp (raw detections and two indicators).

Comparing the evaluations in Figure 10 and Figure 12 as well as video frames shown in Figure 11 and Figure 13, the difference is obvious. By evaluating the frame-by-frame detections as time-series signal, an easy-to-read indicator for the state of a drainage segment can be obtained. Similar evaluations are feasible for each individual error class, which would allow different treatment, e.g., of severe damage and scaling and less severe error classes.

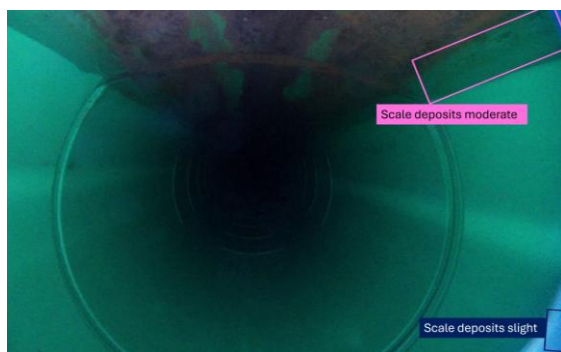


Figure 13: Sample video frame with a small number of detections

Summary and Outlook

Our next steps will focus on increasing data volumes by using further test runs in relevant scenarios, to increase reliability of the obtained results. To avoid confusion between slight and moderate scaling as well as between slight and moderate damage in the evaluation using

YOLOv11, we will furthermore consider including information obtained from the red line laser, used for obstacle detection, into the evaluation. Additionally, we will improve self-locating of the demonstrator within the drainage system using methods for velocity estimation and detection of maintenance holes and branching points.

Acknowledgement

This work has been supported by the COMET K2 Center of the Linz Center of Mechatronics (LCM) funded by the Austrian federal government and the federal state of Upper Austria.

References

- [1] Eichinger, S. et al. (2020). „Assessment and formation mechanisms of scale deposits in tunnels of the ÖBB-Infrastruktur AG – A subproject of the Task Force Drainage”. *Geomechanics and Tunnelling*, 13(3), 273-285.
- [2] Lin, Z., et al. (2023). „Optimizations for tunnel drainage water system based on locations of crystallizations”. *Desalination and Water Treatment*, 299, 232–238.
- [3] Hofer, W. et al. (2023). „A robotic tunnel drainage inspection rover”. *Transportation Research Procedia*, 72, 2724-2731.
- [4] Micic Batka V. et al (2026). “Shaping the future of rail – R&D at ÖBB-Infrastruktur AG / Die Zukunft der Bahn mitgestalten – F&E bei der ÖBB-Infrastruktur AG”. *Railway Proceedings, Band 1 "Koralmittel – Technik im Fokus" / Verlag der TU Graz*
- [5] Simonyan, K. et al (2014). “Very deep convolutional networks for large-scale image recognition.” *arXiv preprint arXiv:1409.1556*.
- [6] Szegedy, Ch. et al. (2015). “Going deeper with convolutions.” *Proc. of the IEEE conference on computer vision and pattern recognition*.
- [7] He, K. et al (2016). “Deep residual learning for image recognition” In *Proc. of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).
- [8] Girshick, R. (2014). „Rich feature hierarchies for accurate object detection and semantic segmentation”. In *Proc. of the IEEE conference on computer vision and pattern recognition* (pp. 580-587).
- [9] Liu, W. et al (2016). “SSD: Single shot multibox detector”. In *European conference on computer vision* (pp. 21-37).
- [10] Rudić, B. et al. (2021): “Robust Hierarchical Data Association for Multiple Object Tracking Under Abrupt Appearance Changes”. In *Proc. of the 2021 Int. Conf. on Electrical, Computer and Energy Technologies*
- [11] Redmon, J. et al. (2016). “You only look once: Unified, real-time object detection”. In *Proc. of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).

- [12] Nguyễn, Đ. (2015). "U-Net: Convolutional Networks for Biomedical Image Segmentation". *Lecture Notes in Computer Science*.
- [13] Fleischanderl, S. et al. (2022). „CNN–based crack detection in oxide layers of hot rolled steel sheet samples for the validation of a pickling process model". In *Proc. of the 2022 European Symposium on Software Engineering*.
- [14] He, K. (2017). "Mask R-CNN". In *Proc. of the IEEE international conference on computer vision* (pp. 2961-2969).
- [15] Khanam, R. et al (2024). "Yolov11: An overview of the key architectural enhancements". *arXiv preprint arXiv:2410.17725*.