

Detecting Rails and Obstacles Using a Train-Mounted Thermal Camera

Amanda Berg^{1,2}(✉), Kristoffer Öfjäll¹,
Jörgen Ahlberg^{1,2}, and Michael Felsberg¹

¹ Computer Vision Laboratory, Department of Electrical Engineering,
Linköping University, 581 83 Linköping, Sweden
{amanda.berg,kristoffer.ofjall,jorgen.ahlberg,michael.felsberg}@liu.se
<https://www.cvl.isy.liu.se>

² Termisk Systemteknik AB, Diskettgatan 11 B, 583 35 Linköping, Sweden
{amanda.,jorgen.ahl}berg@termisk.se
<https://www.termisk.se>

Abstract. We propose a method for detecting obstacles on the railway in front of a moving train using a monocular thermal camera. The problem is motivated by the large number of collisions between trains and various obstacles, resulting in reduced safety and high costs. The proposed method includes a novel way of detecting the rails in the imagery, as well as a way to detect anomalies on the railway. While the problem at a first glance looks similar to road and lane detection, which in the past has been a popular research topic, a closer look reveals that the problem at hand is previously unaddressed. As a consequence, relevant datasets are missing as well, and thus our contribution is two-fold: We propose an approach to the novel problem of obstacle detection on railways and we describe the acquisition of a novel data set.

Keywords: Thermal imaging · Computer vision · Train safety · Railway detection · Anomaly detection · Obstacle detection

1 Introduction

Every year, there is a large number of collisions between trains and objects inappropriately and unexpectedly located on or close to the railway. Such unexpected objects are, for example, animals (moose, deer, reindeer), humans, vehicles, and trees. Collisions with such objects affect the safety of the train passengers, most likely kills the animal or human being located on or near the rail track, cause delays in the train traffic, and also results in costs for repairing the train after the collision.

A train driving at normal speed has a stopping distance of at least one kilometre and, hence, it is often far too late to brake the train to a full stop when the engine-driver detects an undesirable object in front of the train. Under impaired visibility such as bad weather conditions (rain, fog, or snow fall), or

when driving in dusk/dawn or in the dark, the situation is even worse. The chances that the engine-driver detects an unexpected object in front of the train and manages to reduce the speed of the train before a collision, are virtually non-existent. Note also that even if a train needs one kilometre to reach a full stop, the capability to detect certain obstacles also at shorter distances is valuable. For example, a collision with a moose at 50 km/h instead of 200 km/h will still kill the moose, but the repair costs for the train will be significantly lower.

In this paper, we describe a system using a thermal camera to detect obstacles on or near the rails in front of the train. The reason to use a thermal camera is its independence of illumination and its ability to see in complete darkness. The images will, however, be of lower resolution than those of a modern visual camera, which also means that we will need to carefully assess the compromise between pixel footprint size and field of view.

In order to detect obstacles, we first need to localize the rails in the incoming stream of thermal images. Secondly, we need to find possible obstacles. Since we do not know the type of obstacles to find in advance, we have chosen to develop an anomaly detector, *i.e.*, we try to detect objects that do not look like rails where rails are expected to be.

1.1 Railway vs. Road Detection

At a first glance, the problem addressed here might look similar to that of road detection, which is a popular research topic since it is a prerequisite for applications in intelligent cars (safety systems as well autonomous driving). However, at a closer look, the two problems are quite different. Roads are typically structureless, but with defined borders such as lines, curbs, or railings. In urban areas and on highways, visual road detection is carried out by detection of lane markers. A common approach is to reproject the image onto the ground plane followed by line detection in the resampled image [1, 6, 7]. Using other sensors such as LIDAR, similar lane detection approaches are used [5]. Sometimes, the detection of driveable areas, *i.e.*, flat areas without 3D elevation, is the crucial component.

For railways, we have a different situation. The railway has a defined structure; two rails at a specified distance from each other and perpendicular sleepers. On the other hand, there are no defined borders or lane markers. We thus need a different strategy than for roads and lanes, and we have developed a new method for detecting railways. Nonetheless, some ideas from lane detection are transferable to this new domain; for example, resampling the image in a ground plane grid is advantageous compared to line detection directly in the camera image [4]. Unfortunately, this resampling destroys information close to the vehicle where the road is densely sampled in the original image, while much effort is spent representing far areas where little information is available. Our proposed method attains the advantages of a ground plane projection without resampling each frame. Thus, our contributions are:

- We have collected a dataset of thermal video recorded from a train.
- We propose a method for detecting the railway in such video.
- We propose a method for detecting possible obstacles on the railway.

1.2 Outline

In the remaining of the paper, we will first describe the acquisition of test data (Section 2) and then the proposed rail and obstacle detection algorithms (Section 3). Experimental results are presented in Section 4, and finally our conclusion and an outlook can be found in Section 5.

2 Data Collection

In order to collect relevant image data, we mounted a thermal camera and a mirror into a custom camera house – basically a metal box, see Fig. 1a. The purpose of the arrangement with the mirror is to lower the risk of the (expensive) thermal camera to break due to collision with small objects. In a final system, a smoother housing with a protective window will be used. The used camera is a FLIR SC655 acquiring images in the long-wave (thermal) infrared band, *i.e.*, in the wavelengths 8–12 μm . It has a resolution of 640×480 pixels and acquires 50 frames per second. Two different optics were used, with horizontal fields of view of 7 and 25 degrees respectively. The 7 degree optics give a good resolution at large distances (more than one kilometre), however, when the railway is curved, it often is outside the field of view. The 25 degree optics almost always keeps the railway within the field of view, but the detection distance is much lower. The prioritization will in the end be made by the customer, and might be different for different trains (the two extremes, 7 and 25 degrees, are both quite unlikely). In the following, for the sake of presentation, images acquired using the 25 degree optics are shown.

The camera house was mounted to a train as shown in Fig. 1b. Obviously, this is an exposed placement, not to be copied to the final system. A recording computer was installed inside the train, with a display visible to the driver (Fig. 1c). In addition, a forward-looking video camera was placed inside the train. Note that the thermal camera cannot be placed there, since the glass in the windshield is intransparent to long-wave infrared radiation.

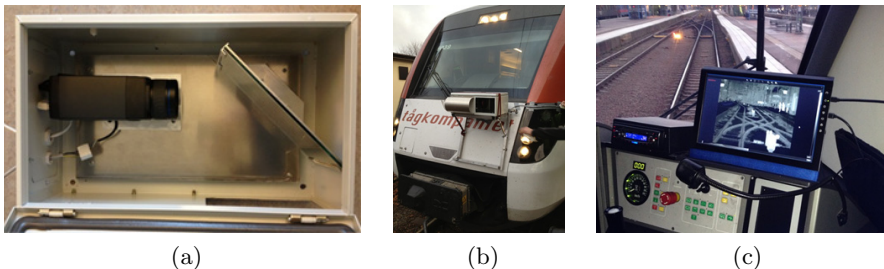


Fig. 1. The camera installation for data collection. (a) The camera house with a 45 degree mirror. (b) The camera house mounted on the front of a train. (c) The display at the driver's panel.

3 Proposed Method

3.1 System Overview

The system includes a thermal camera in a temperature-controlled housing (currently under development) connected to a computer and a display with a graphical user interface. The computer runs software for acquiring thermal images, computing the scene geometry, detecting the railway, detecting and tracking possible obstacles, and giving alarms to the driver conditioned on certain criteria. This process is illustrated in Fig. 2.

Since the extrinsic and intrinsic parameters of the camera can be assumed to be known, the scene geometry is computed first. This includes the homography from pixel to ground coordinates (relative to the train) and possible rail locations. The geometry is computed once, and used, in each frame, for a rough estimation of the rail location as described in Section 3.2. Next, this estimate is refined using an adaptive correlation filter, which is also used for anomaly detection, *i.e.*, finding obstacles on the rails, see Section 3.3.

Moreover, a foreground-background segmentation algorithm is used for finding moving objects near the railway, and detected foreground objects as well as anomalies are kept track of using a multi-target tracker. Finally, the output of the tracking is subject to a filtering, where only detected obstacles fulfilling certain alarm criteria will be reported to the operator. This paper focuses on the three first steps, *i.e.*, the blocks with solid borders in Fig. 2.

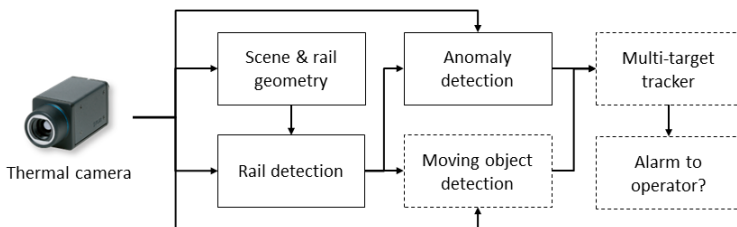


Fig. 2. The system for rail and obstacle detection. The components with solid contours are in the focus of this paper.

3.2 Scene Geometry and Rail Detection

Assuming a flat ground, which is appropriate in a railway setting with limited gradients, there is a one to one mapping from pixels to points in a ground coordinate system fixed to the train, commonly referred to as the inverse perspective mapping (IPM), a homography determined from the known camera parameters. Further, the train has a fixed position and orientation relative to the railway during all normal modes of operation. Assuming a locally constant curvature of the railway, the curvature is the only free parameter determining the position

of the rails in the image. This is exploited to obtain fast and reliable rail detections. These assumptions allow us to obtain the advantages of rail detection in the reprojected ground plane image without the need of explicitly reprojecting each frame, thus avoiding the drawbacks of sampling.

Rail Geometry. Given the design of a railway engine, it is apparent that the rails will be parallel to the engine at a point midway between the fore and aft bogies, see Fig. 3a, later referred to as the *parallel point*. The orthogonal offset at this point, λ in Fig. 3a, the signed distance between the center of the railway and the center of the engine, is determined by the local curvature, $Q = 1/R$, and the wheel base, c . It is given by the width of a circle segment

$$\lambda = \frac{1}{Q} - \sqrt{\frac{1}{Q^2} - \frac{c^2}{4}} \approx \frac{Qc^2}{8}, \tag{1}$$

where the approximation for small curvatures is linear in Q . Together with the length-wise camera mount offset t and camera parameters, this determines the position of the rails in the image for each possible curvature.

The deviation, d in Fig. 3b, of a rail from the parallel point can be derived from geometric relations and is given by

$$d = h \cot \left(\frac{\pi - \sin^{-1}(hQ)}{2} f \right) \tag{2}$$

with the inverse

$$Q = \frac{1}{R} = \frac{1}{h} \sin \left(\pi - 2 \cot^{-1} \frac{d}{h} \right). \tag{3}$$

The geometry and parameters are illustrated in Fig. 3b.

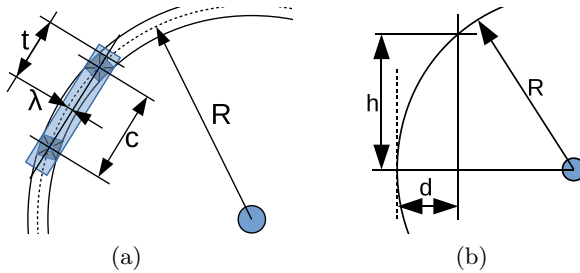


Fig. 3. Illustration of railway geometry. (a) Geometry of a railway engine on a railway with constant curvature $1/R$. (b) Geometry of a constant curvature rail.

Histogram Bin Mapping. Given (1), (3) and the camera parameters, the corresponding curvature of the left and right rail, if passing through a given pixel, can be determined for each pixel below the horizon. Placing bins in the one dimensional curvature space, look-up images are generated, mapping each pixel to a curvature bin for the left and right rail respectively. Such mappings are shown in Fig. 4, where a suitable range of curvature is discretized into 100 bins. In automotive applications, the lateral position of the car on the road is not fixed, thus requiring at least a two dimensional histogram. Pre-calculation of bin mappings is thus not plausible in an automotive setting.

Further, the expected orientation of each rail in the image, if passing through a given pixel, can be determined. This is shown in Fig. 5, where 0 is vertical and $\pm\pi/2$ is horizontal.

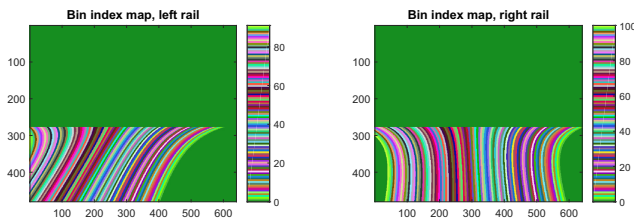


Fig. 4. Images illustrating the mapping from pixel positions to the corresponding curvature histogram bin indexes, for left and right rail respectively.

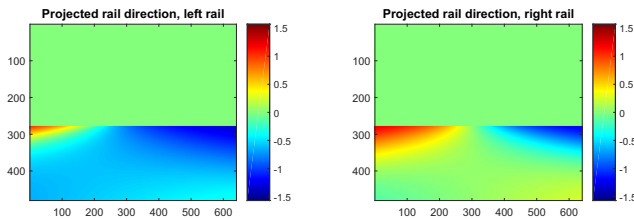


Fig. 5. Expected rail direction in image, left and right rail respectively

Curvature Estimation. The curvature histogram mappings and the expected orientation are calculated in advance for rails close to the camera where the flatness and constant curvature assumptions hold. For detecting rails further ahead, a different approach is used, where lines in the original image are traced, starting from the histogram based detection. The detected lines are projected onto the ground plane, whereafter a spline-based curvature rail model is fitted to the projected detections. By this, the model can be fitted in the ground plane while only the detected lines are projected, not the full image. However, the histogram based detection is the focus of this paper.

For each frame, Gaussian derivative filters are applied, estimating edge strength and orientation. For each pixel (x, y) and rail, the edge strength $A_m(x, y)$ is modulated depending on the difference between the estimated orientation $P_m(x, y)$ and the expected orientation $P_e(x, y)$, Fig. 5, according to

$$A_m(x, y) \exp\left(-\frac{(P_m(x, y) - P_e(x, y))^2}{\sigma_P^2}\right), \tag{4}$$

where the parameter σ_P determines the orientation error tolerance. The modulated value is added to the curvature bin determined by the bin look-up image, Fig. 4. Assuming limited rail curvature and camera view, the modular nature of the orientation does not require any special attention.

Finally, the peak of the curvature histogram is extracted. The result is illustrated in Fig. 6, where an image is shown with the areas corresponding to the histogram peak overlaid. The histogram is also shown, together with the corresponding histogram obtained without orientation modulation. Using orientation weighting, false curvature responses are significantly reduced, resulting in a stronger peak to noise ratio in the histogram.

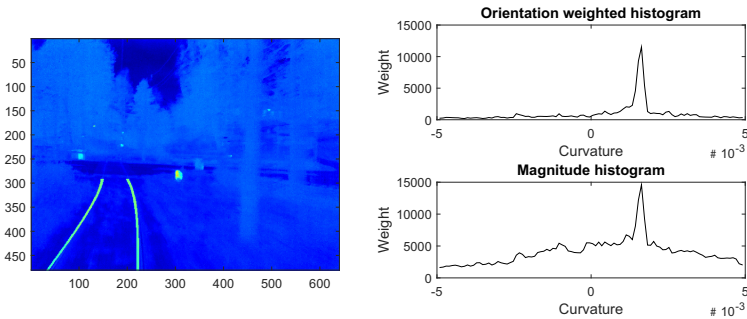


Fig. 6. Left: Camera image with areas mapping to the peak histogram bin overlaid. Top right: Curvature histogram generated from the image using orientation dependent weighting of edge magnitudes (4). Bottom right: Histogram generated from the same image without orientation weighting.

3.3 Combined Correction and Anomaly Detection

A rail mask which does not follow the rail properly will cause false detections to appear. This will happen, for example, when the assumption above about constant curvature does not hold. Therefore, the rail mask needs to be corrected before anomaly detection can be applied. Correction as well as anomaly detection is performed row-wise using an adaptive correlation filter, similar to the one used in the MOSSE tracker [2]. The original image as well as the binary mask from the rail detection serves as input. In each frame, for each row of the original

image, all pixels within the rail mask are rescaled. Rescaling is performed using cubic interpolation and an example of an image with rescaled masked rows can be seen in Fig. 7. As opposed to [2], the correlation filter is one-dimensional, similar to the scale filter in [3], and applied row-wise.

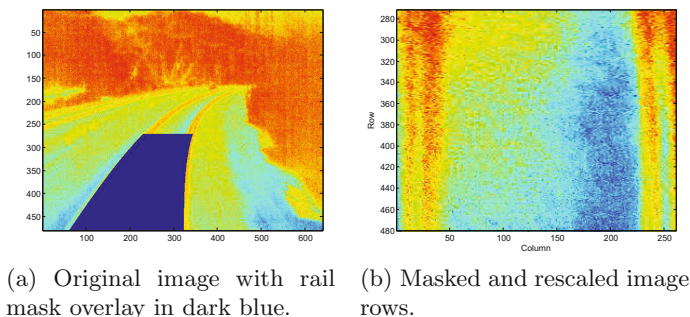


Fig. 7. (a) The original image and the binary rail mask serves as input to the rail mask correction and anomaly detector. (b) All pixels within the rail mask are rescaled row-wise in order for the masked rows to have the same width.

Rail Mask Correction. Rail mask correction is an iterative procedure. The correlation filter is trained to give a Gaussian distribution, $\mathcal{N}(w/2, w/2)$, where w is the filter width, in response when applied to one-row image patches of rails. If the mask does not follow the rail properly, the filter responses will have a similar offset since the filter is trained to give a Gaussian with its mean in the center of the rails. By adjusting the horizontal positions of the rail mask rows based on the filter response information, the rail mask can be corrected. In Fig. 8, an example of an erroneous rail mask and its filter responses before and after correction can be seen.

Correction is performed bottom-up using the filter responses, starting with the response at the bottom row. All filter response rows are correlated with different displacements to the bottom row, one by one, and the displacement with the highest correlation is considered the final displacement for that row. In order to enforce smoothness in the correction displacement, each row is only allowed to have a displacement of ± 1 pixel relative to the previous row. Detections from the previous frame are used during the correction phase in order not to introduce errors in the corrected mask. That is, rows which had a detection in the previous frame are not corrected and their displacements are set to the one of the previous row without a detection.

A correction like the one described above is performed in each iteration of the correction phase. In each iteration, a new rescaled image and new responses are calculated. If the mean correlation over all rows of the current iteration to the mean response of all previous non-detected rows is lower than the mean

correlation of the previous iteration, the iteration is stopped and the correction of the previous iteration is accepted as the final one. Approximately three iterations are needed to correct the rail mask if it has a moderate offset.

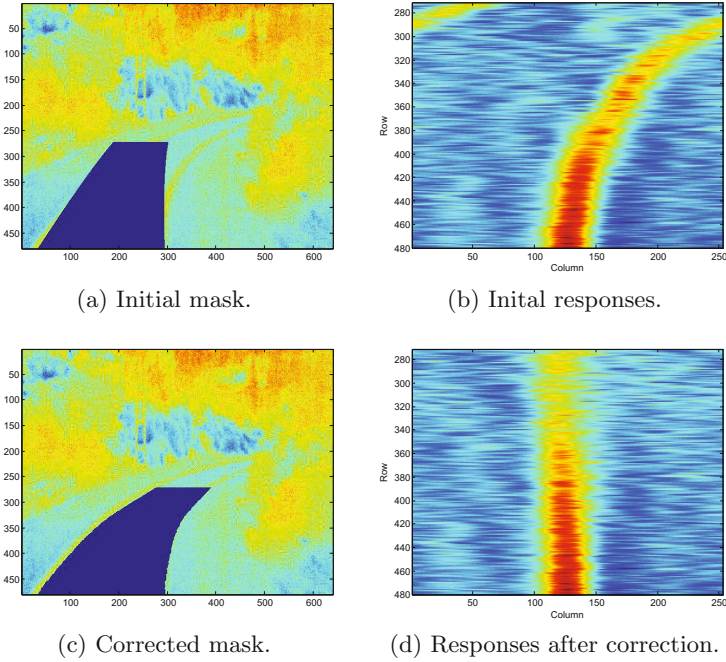


Fig. 8. Example of a rail mask correction. Here, 3 iterations were needed.

Detection of Anomalous Rows. When the rail mask has been corrected, the resulting resized image is analyzed for anomalous rows. In this step, the filter responses from the final, corrected rail mask are used. Given a filter response $y_{t,j}$ at time t and row j , a correlation coefficient $c_{t,j}$ is calculated between $y_{t,j}$ and the median of $y_{t,j}$, $j \in S_{t-1}$. S_{t-1} is the set of rows without a detection in the previous frame. The median is used in order to reduce the influence of outliers.

The ℓ^1 -norm $b_{t,j} = |c_{t,j} - m_{t-1,j}|$ is then thresholded to find detections.

$$d_{t,j} = \begin{cases} 0 & b_{t,j} < \gamma \\ 1 & b_{t,j} \geq \gamma \end{cases} \quad (5)$$

where $d_{t,j}$ indicates whether the row is anomalous or not and $\gamma \in [0, 1]$ is a constant. $m_{t-1,j}$ is the weighted mean of all previous correlation coefficients of row j , updated as $m_{t,j} = \beta c_{t,j} + (1 - \beta)m_{t-1,j}$, with an update factor $\beta \in [0, 1]$.

Instead of using the median filter response, the filter responses could be correlated to the Gaussian used for training. However, due to contrast variations and other distortions, a normalisation would be needed. Using the median response vector has proven to be the best method for this application.

Correlation Filter Update. All filter operations are performed in the Fourier domain using the Fast Fourier Transform in order to reduce computation time. The correlation filter is trained as

$$H = \frac{\sum_{j \in S} \overline{G} F_j}{\sum_{j \in S} \overline{F_j} F_j} \quad (6)$$

where S is the set of image rows of the rescaled image that are to be used for training. F_j is a Fourier transformed pixel row and G is the Fourier transform of the ideal filter response. In this case the ideal filter response is a Gaussian with its peak centered between the two rails. The bar denotes complex conjugation and the product $\overline{F_j} F_j$ is point-wise. The derivation of (6) is given in [2]. The filter response $y_{t,j}$ at time t and row j of image row z , is found as

$$y_{t,j} = \mathcal{F}^{-1}\{\overline{H_t} Z\}. \quad (7)$$

In each new frame, the filter is adaptively updated using a weighted average as

$$H_t = \alpha \left(\frac{\sum_{j \in S_t} \overline{G} F_j}{\sum_{j \in S_t} \overline{F_j} F_j} \right) + (1 - \alpha) H_{t-1} \quad (8)$$

where S_t is the set of rows at time t that have qualified for filter update and $\alpha \in [0, 1]$ is the update factor. The rows used for filter update are randomly chosen among the rows that did not have a detection.

4 Experimental Results

The test sequence that was used to evaluate the performance of the system consists of 386 frames. When the sequence was recorded, the train was driving at about 200 km/h on a railway which slightly bends to the right.

Three simulated objects are introduced on the rail during the sequence. Objects are assumed not to suddenly appear and to increase in size as the train approaches. The first object is a square and the second object is a rectangle simulating a pipe lying over the rails. The third object is an upright rectangle simulating a standing object, human or animal. The temperature of the objects are set to $T_b + \Delta T$ where T_b is the surrounding background temperature and ΔT is a temperature difference. Furthermore, the edges of the objects are smoothed using an averaging filter. Two example frames from the test sequence where the simulated objects are present can be seen in Fig. 9.

In each frame, 100 randomly selected rows are used to train the adaptive correlation filter. The update factor of the filter, α , was set to 0.025 and the update factor of the correlation coefficients, β , was set to 0.1.

The detection and performance evaluation is performed row-wise. Two parameters were varied during the evaluation: The detection threshold γ and the temperature difference ΔT . In each frame, the height of the rail mask was

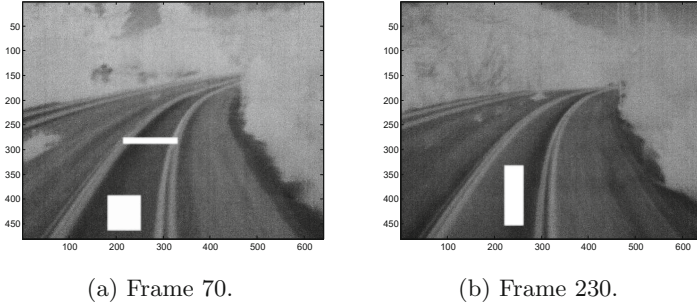


Fig. 9. The simulated objects (a) 1, 2, and (b) 3 in the test sequence.

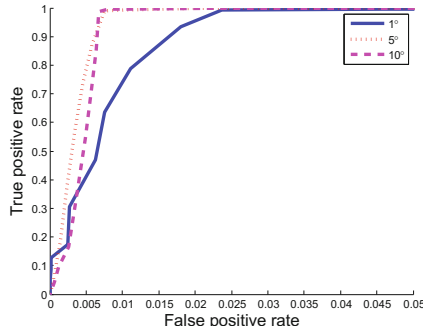


Fig. 10. ROC-curves of the true positive and false positive rates for three different values of ΔT .

set to 209 rows which yields 80674 rows for evaluation. A plot containing ROC-curves for $\Delta T = 1^\circ C$, $5^\circ C$ and $10^\circ C$ can be found in Fig. 10.

As expected, the results improve when the temperature difference increases. The results for $\Delta T = 5^\circ$ and $\Delta T = 10^\circ$ can be considered to be equal. A significant improvement can, however, be noted between $\Delta T = 1^\circ$ and $\Delta T = 5^\circ$. In order for the method to be useful, a true positive rate of at least 90% and a false positive rate of at maximum 1 – 2% are needed. Occasional false detections can be handled and as long as the true positive rate of an object is larger than the true positive rate per row, missed detections can be handled as well. In the case of $\Delta T \geq 5^\circ$, these criteria are fulfilled.

5 Conclusion

The conclusion is that the rail detection method works satisfactory, but, as expected, can only be used at a limited range due to its assumption of constant curvature. The combined correction and anomaly detection method successfully

compensates for model errors, and also gives detection results useful in the practical application. Future and ongoing work will include further development of both these algorithms. There are also several special cases that will need to be addressed, such as heated railroad switches and connecting railroads, that currently result in detections. In addition to that, detection of foreground objects and/or moving objects near (not on) the rails will be added to the system.

The next step is to install the system on one or more test trains and perform extensive testing and data collection at different speeds, weather conditions, and environments. These tests will be performed during 2015, and will most likely result in the discovery of additional special cases and circumstances that will need to be addressed.

Acknowledgments. This work was financed by Rindi Solutions AB. The research was funded by the The Swedish Research Council through framework grants for projects Energy Minimization for Computational Cameras (2014-6227) and Extended Target Tracking. The development of the multi-target tracker was supported by the European Community Framework Programme 7, Privacy Preserving Perimeter Protection Project (P5), grant agreement no. 312784. We also gratefully acknowledge the train company Tågkompaniet for using their train.

References

1. Aly, M.: Real time detection of lane markers in urban streets. In: IEEE Intelligent Vehicles Symp. (2008)
2. Bolme, D.S., Beveridge, J., Ross, D., Bruce, A., Lui, Y.M.: Visual object tracking using adaptive correlation filters. In: Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (2010)
3. Danelljan, M., Häger, G., Khan, F., Felsberg, M.: Accurate scale estimation for robust visual tracking. In: Proc. of the British Machine Vision Conf. (2014)
4. Borkar, A., Hayes, M., Smith, M.: Robust lane detection and tracking with ransac and kalman filter. In: IEEE International Conf. on Image Processing (2009)
5. Kammel, S., Pitzer, B.: Lidar-based lane marker detection and mapping. In: IEEE Intelligent Vehicles Symp. (2008)
6. Kreucher, C., Lakshmanan, S.: LANA: A Lane Extraction Algorithm that Uses Frequency Domain Features. IEEE Trans. on Robotics and Automation **15**(2) (1999)
7. Otsuka, Y., Muramatsu, S., Takenaga, H., Kobayashi, Y., Monji, T.: Multitype lane markers recognition using local edge direction. In: IEEE Intelligent Vehicles Symp. (2002)