computer graphics & vision

Inffeldgasse 16/2, A-8010 Graz

www.icg.tu-graz.ac.at

TUG

Railroad Car Separation From Continuous Train Images

Computer Vision Project (710.014) - SS2010

Robert Hödl

Inst. for Computer Graphics and Vision
Graz University of Technology, Austria

Technical Report
ICG–TR–010
Graz, March 21, 2011

contact: Robert Hödl robert.hoedl@student.tugraz.at

Abstract

This work presents a novel approach in detecting the separation points of railroad cars in continuous train images, which were recorded using a line scan camera. First a segmentation of the train is obtained by a combination of an edge and a region based procedure. The edge based approach only detects vertical edges in order to meet special characteristics of the line scan camera. The region based procedure calculates a difference image with a predefined background. The combination then provides a very robust segmentation result. From this segmentation an 1D signal is retrieved and analysed in order to determine the position of the single separation points. Extensive experimental evaluations demonstrate the success of this approach.

Keywords: *continuous train image, line scan camera, railroad car separation, segmentation*

Contents

1 Introduction

Today more than ever, reliability, convenience and speed are important factors for the commercial success of modern railways. To meet these requirements more and more automated railway monitoring systems have been deployed in the recent years. The task described within this paper is part of such an automated railway monitoring system. Amongst other things such systems visually capture entire freight trains while passing a terminal in order to identify the single railroad cars.

Up to now within Europe this identification has to rely on visual information since the railroad car number according to the RIV (Regolamento Internazionale dei Veicoli) conventions [2] is the only way to identify railroad cars operated by 'foreign' carriers. As a preprocessing step for the vision-based identification it is necessary to divide the recorded continuous train image into the single railroad cars. Once a train and the corresponding railroad cars are identified the variety of applications is numerous. These include general fleet management tasks, search for certain railroad cars, necessary documentation tasks, recognition of loading gauge exceedance or the detection of dangerous goods plates.

There is only limited work on vision based railroad car separation publicly available. Existing systems for automated railway monitoring such as the Trackblaze - WIZ[1] (Australia) use RFID tags or as the VACIS[2] System from SAIC (USA) rely on the usage of hardware coupling sensors in order to accomplish the task of separating the railroad cars. Since these two systems explicitly apply techniques different from a vision based approach they are not further examined.

In [5], Quing-Jie *et al.* describe the segmentation of the foreground of containers in videos of a moving train. The goal is to reliable detect loading patterns of containers and the gaps in between them. The proposed method segments the containers by dividing the captured images into regions. First the periodicity of railroad cars is analysed in order to remove the regions above and below the containers. Then the gaps between the containers are identified by applying a histogram based background model. The obtained results are refined using colour information.

The environment of the approach described by Quing-Jie *et al.* is significantly different from one which will be discussed within this paper. First of all the recording technique used in [5], is based on a movie camera whereas the

[1] http://www.trakblaze.com/wiz.htm
[2] http://www.saic.com/products/security/rr-vacis/

line scan cameras used in this described setup shows very distinct characteristics as can be seen later on. Another important difference is that Quing-Jie *et al.* focus on the segmentation of railroad cars loaded with freight containers. The goal of this approach will be to generalise on all kinds of (freight) railroad cars.

1.1 Specific Project Environment / Constraints

This specific task originates in the environment of a railway monitoring system called WaggonID developed by the Siemens AG Österreich[3].

The goal of this work was in particular to integrate a vision based method for railroad car separation into the railway monitoring system. In order to capture the visual information of the freight trains, a setup consisting of a line scan camera, artificial lighting and inductive sensors was installed along the train line. So far the separation of the railroad cars is done using inductive sensors embedded in the railroad tracks which determine the axle patterns of the traversing railroad cars. These axle patterns are then queried and assigned to a specific car type which enables the system to estimate the bounds of a single railroad car.

Since the implementation of inductive sensors is rather expensive and the axle pattern estimation of the bounds of the railroad cars is sometimes imprecise a solution based only on the captured visual information was needed. Hence, the goal of this project is to perform:

Railroad car separation from continuous train images.

In contrast to existing approaches several issues have to be considered for the given task:

- **line scan camera**
  The main difference to other segmentation tasks is the usage of a line scan camera as an imaging device. The main characteristic of images captured by a line scan camera are miscellaneous jittering horizontal streaks induced in the background (see Figure 1).

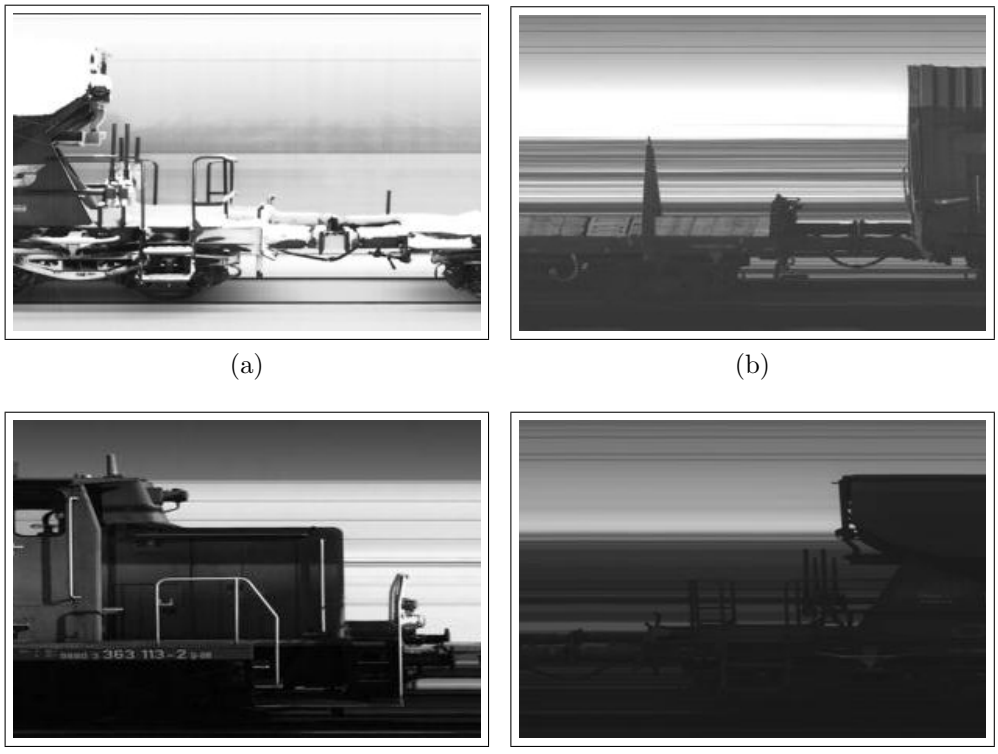
- **image quality**
  Although the images are captured at a very high resolution the image quality suffers from:

  - *distortions* - due to static sampling rate but changing train speeds

[3] http://www.mobility.siemens.com/

  - *noise* - mostly due to weather conditions like rain or snow fall.
  - *low contrast* - especially at twilight and during night because of insufficient artificial illumination (see Figure 1b and 1d).

- **railroad car appearances**
  It is not possible to make any suggestions about the appearance of railroad cars (except a minimal length). An overview of different basic types of railroad cars can be found in a "*Guide to Railcars*"[4].



(a) (b) (c) (d)

Figure 1: Examples of different horizontal streaks as a typical characteristic of the line scan cameras.

In addition certain constraints were defined:

- **grayscale images**
  Not all WaggonID setups are equipped with line scan cameras which are able to provide colour images. So the given task has to be performed on grayscale images only.

[4] http://www.worldtraderef.com/WTR_site/Rail_Cars/Guide_to_Rail_Cars.asp

- **real-time detection**
  Real-time detection of the separation points is a requirement, especially as it is just considered as a preprocessing step. Hence the detection has not to be performed online but should be possible within a reasonable amount of time. In particular, this allows the caching of the whole train image data beforehand, while analysing it afterwards.

Considering the described problem definition where basically a camera is capturing a static scene, the question arises how to separate the foreground from the background. The mentioned issues and given constraints strongly indicate that the simple application of well known background modelling techniques will not suffice to accomplish this task. Background subtraction methods like Approximated Median [4] or Running Average [3] are not able to handle the strong varying backgrounds as the ones addressed in this paper with the induced jittering horizontal streaks and the sudden changes in illumination caused by shadowing in between the railroad cars. Situations observed many times within the test-datasets. Statistical methods like Mixture of Gaussians [9] or Eigenbackgrounds [1] would face the same challenges but are computationally much more expensive and so their application would contradict the real-time processing constraint.

In the course of this paper it will be demonstrated that through a smart choice of methods these issues will be overcome.

The remainder of this paper is structured as follows. Section 2 gives an in-depth description of this approach. In Section 3 the experimental results are presented. Finally Section 4 concludes the paper.

2 System

The algorithm for the separation point detection consists of two main stages. First, a foreground/background segmentation is calculated by combining an edge and a region based approach. Secondly the separation point detection is realised by retrieving and analysing an 1D signal from the segmentation result. The separation point detection itself is then implemented by determining specific local maxima in the signal. The basic workflow of the algorithm is illustrated in Figure 2.

In the following, we give a more detailed description of the system modules.

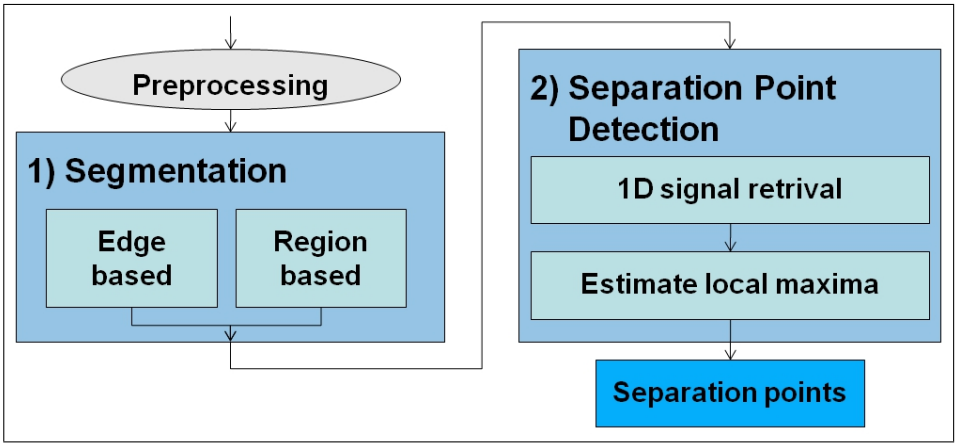


Figure 2: Basic workflow of the algorithm consisting of two main stages: Segmentation and Separation Point Detection

2.1 Preprocessing

As a preprocessing step the different input samples have to be normalised regarding size (10 % of the original), orientation and colour information (grayscale). Since the original input images consist of small image patches, these patches are put together to form panoramic train images. First a rank order median filter (3×3 mask) is applied to these panoramic images in order to remove noise while preserving edges. The following steps are applied to the panoramic train images.

2.2 Segmentation

The segmentation part is a combination of two approaches. An edge-based and a region-based approach. This combination provides a far more robust segmentation result (see Section 2.2.3).

2.2.1 Edge Based Segmentation

The edge-based segmentation approach makes use of the (as already discussed in Section 1.1) very particular characteristic of the line scan cameras to induce horizontal streaks in the background. All one can say about the background is that it either contains no information about edges (which is the case at night) or it contains horizontally aligned elements only (compare Figure 1). The main idea is now to detect all vertical edges in the image, since these strongly indicate the existence of foreground structures.

Well known first-derivative edge detection operators like Roberts [6], Prewitt [8] and Sobel [8] can be considered suitable for this kind of task because of their ability to filter edges in desired directions only. The latter one is chosen because of its popularity '*as a simple detector of horizontality and verticality of edges*' [8]. The sometimes considered superior Scharr operator [7] only improves the detection of edges which are not axially aligned but this is not the case right now.

So as a first step the input image I is convolved with a Sobel operator S_x to calculate vertical gradients in the image only:

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} * I = S_x * I. \tag{1}$$

As already mentioned this operation is based on the assumption that the background doesn't contain any vertical artefacts (*i.e.*, all vertical structures in the image belong to the foreground).

On the result of the Sobel operation G_x edges with a certain strength in their responses have to be identified. Therefore a threshold T is chosen to determine if, depending on the intensity value, a pixel of an input image I belongs to the foreground or to the background resulting in a binary output image I_b:

$$I_b(m,n) = \begin{cases} 1 & \text{for } I(m,n) \geq T, \\ 0 & \text{for } I(m,n) < T. \end{cases} \tag{2}$$

The constantly changing illumination conditions (day, night) and the different project setup conditions in which the line-scan cameras are operated make the usage of a fixed threshold as defined by equation (2) obsolete. Instead it will be necessary to apply an adaptive thresholding method for this specific task.

A suitable method is called P-tile (from 'percentile') thresholding [8]. It makes use of prior information such as the expected ratios of foreground and background in the whole train images which can be estimated beforehand. Based on the image histogram H_I it is possible to choose a threshold T such that $1/p$ of the image area has an intensity value less than T. It is assumed that the background of the train occupies at least 70% of the area in the panoramic train image ($1/p \geq 1/7$). P-tile thresholding is defined as

$$H_I(g) = \sum_{n=0}^{g} H(n) \tag{3}$$

The threshold T corresponds to the intensity value g where the cumulative sum of pixel intensities of the histogram H_I of image I is closest to the demanded percentile, so where $H_I(T) = 1/p$.

As a last step of the edge based approach, the binary image obtained by the P-tile thresholding method is median filtered to remove noise. An example of the whole process is pictured in Figure 3.

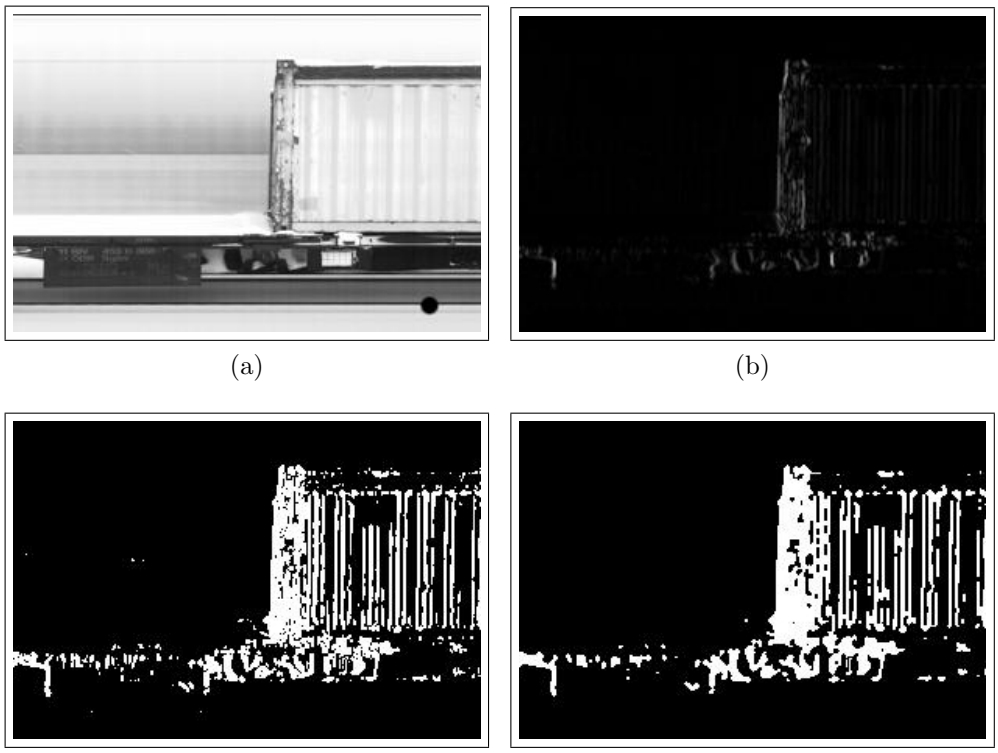


Figure 3: Edge-based segmentation: (a) input image (b) vertical gradients calculated using the sobel operator (c) p-tile threshold image (d) final result after median filtering.

2.2.2 Region Based Approach

For the region-based segmentation first a difference image is generated. This is accomplished by taking the absolute value of the subtraction between the original input image and a previously defined background which corresponds to that particular scene. A description of how exactly the background is defined can be found in Section 2.4.

Then again the adaptive p-tile thresholding method is applied resulting in a binary image. In order to gain a more robust result morphological operations (2× closing, 1× opening with a squared structure element) are applied. Figure 4 illustrates the workflow of this approach.

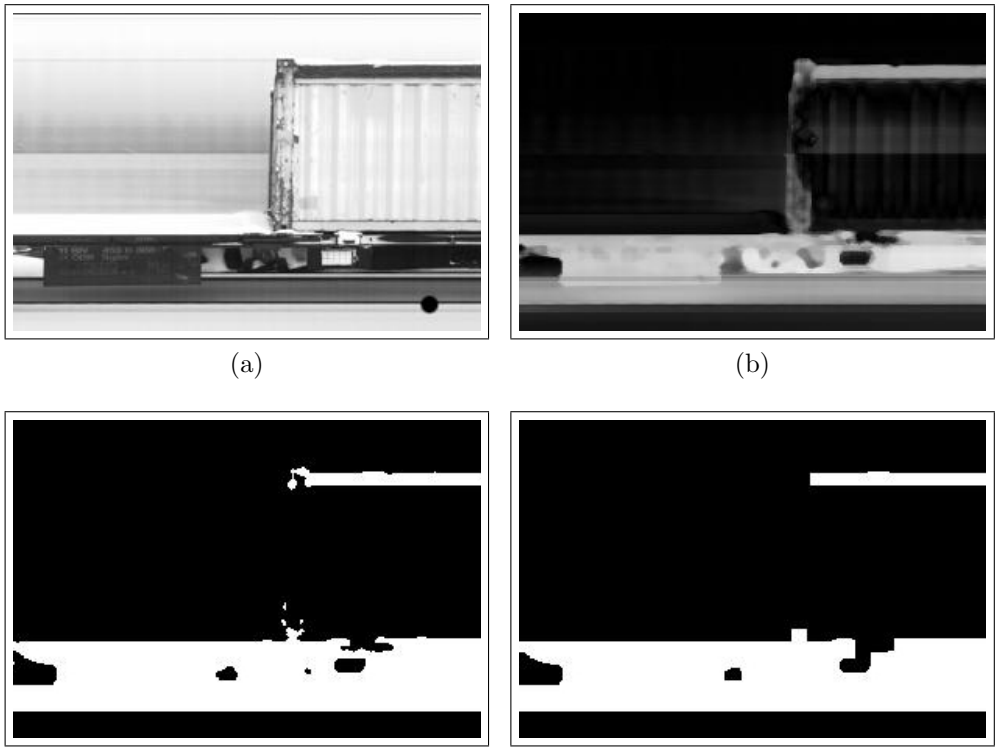


Figure 4: Region-based segmentation: (a) input image (b) difference image generated by subtraction of a predefined background (c) result after p-tile thresholding (d) final result after morphological operations.

The segmentation results of this region-based approach might be considered insufficient. For example as shown in Figure 4d at the location of the container no foreground is detected. On the other hand there is an over-segmentation beneath the railroad car because of the shadow. Nevertheless these results are sufficient, especially since both segmentation results are combined later on.

2.2.3 Combination

As already mentioned above, the overlay of the outcomes of the two approaches is realised by simple disjunction (OR) provides a robust segmentation result (see Figure 5). This operation compensates for deficiencies of the individual approaches: The edge approach cannot recognize large areas of homogeneous foreground (*e.g.*, long Tankers); the region based approach fails when background and foreground are strongly related (see Figure 4).

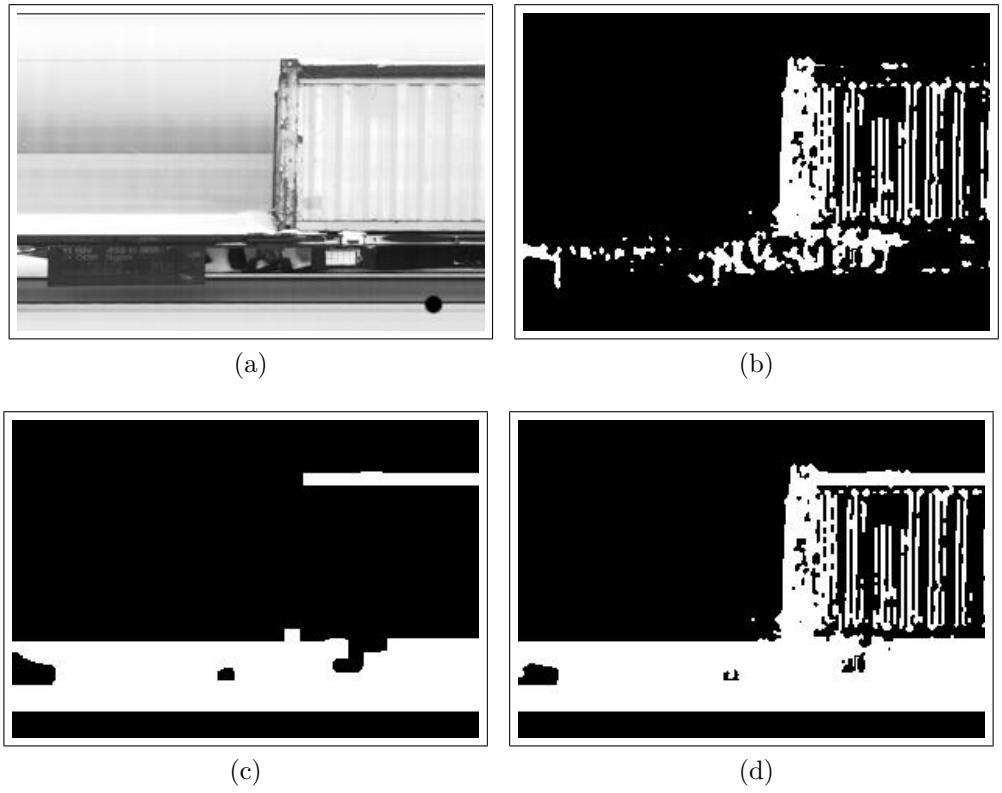


Figure 5: Segmentation: Combination (a) input image (b) result of edge-based segmentation (c) result of region-based segmentation (d) final segmentation result as a combination by disjunction.

2.3 Separation Point Detection

The separation point detection is based on the retrieval of a signal from the previously achieved segmentation. The following sections explain the procedure.

2.3.1 Signal Retrieval and Processing

The final 1D signal retrieved from a segmentation image is composed by two other signals. The first signal can be described as the envelope (ENV) enclosing the segmented train from above. So for every column in the image the sum of background pixels '*on top*' of the last foreground pixel (tfp) is determined. The second signal represents the sum of all foreground pixels (NPX) per image column (see Figure 6).

Figure 6: Illustration of the calculation of the ENV and NPX signals. Red bars correspond to the ENV signal and indicate the sum of pixels '*on top*' of the last foreground pixel. Blue bars represent the NPX signal indicating the sum of foreground pixels per image column.

So for an image $I(m, n)$ the ENV and NPX signals are defined as

$$\mathsf{ENV}(k) = \sum(I(1 : \mathsf{tfp}, k)) \qquad k = 1 \ldots n \tag{4}$$

$$\mathsf{NPX}(k) = \sum(I(1 : m, k) > 0) \qquad k = 1 \ldots n \tag{5}$$

with $\mathsf{tfp} \ldots$ topmost foreground pixel.

Then the superposition of the two signals results in the RAW 1D signal (example in Figure 7b). Both signals are weighted equally since they seem equally important for further calculations (see Table 3). Therefore it is reasonable to calculate the superposition of the two signals by a simple mean:

$$\mathsf{RAW}(k) = (\mathsf{ENV}(k) + \mathsf{NPX}(k))/2 \tag{6}$$

The RAW signal is then processed in order to emphasize possible separation points. First the signal is inverted (to obtain local maxima at the separation points), then smoothed by convolution with a Gaussian kernel g and finally weighted (squared). For an illustration see Figure 7c.
So the final signal SIG is calculated as follows:

$$\mathsf{SIG}(k) = (g * (\mathsf{RAW}(k)^{-1}))^2 \tag{7}$$

with $g \ldots$ 1D gaussian kernel.

2.3.2 Separation Point Assignment

The final step in the separation point detection is to locate all local maxima subjected to the following requirements:

- **minimum peak height (MPH)** - find only those peaks of the signal that are greater than MPH. A value of 130 for the mph was determined through extensive testing. See Figure 9 for more details.
- **minimum peak distance (MPD)** - find peaks that are at least separated by MPD (refers to the minimal expected railroad car length).

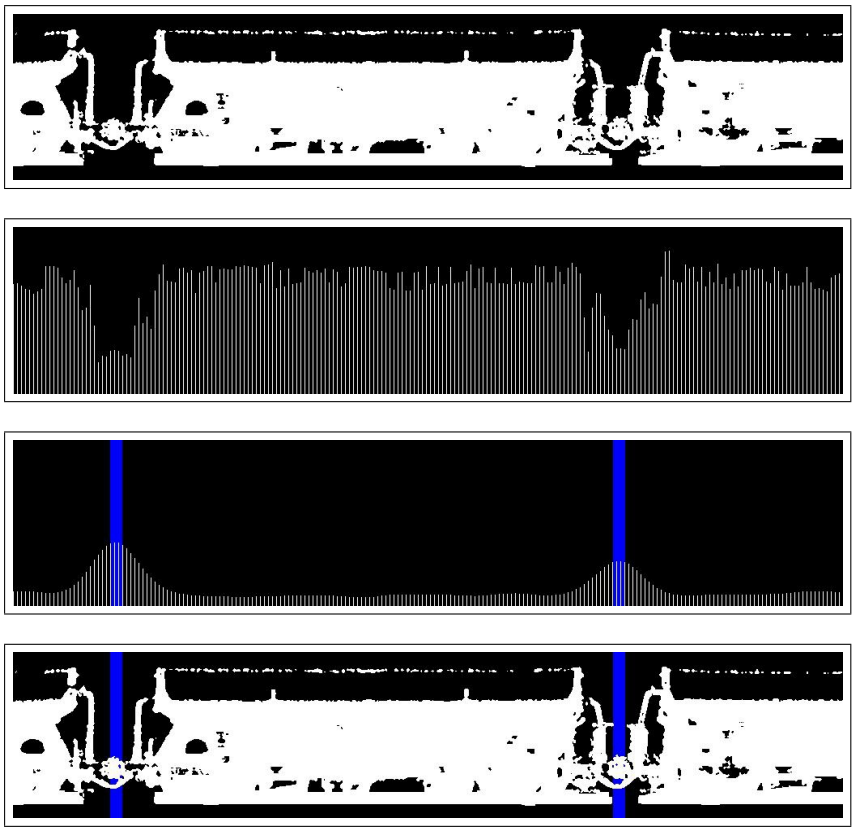


Figure 7: Separation point detection: (a) segmentation result as input image (b) RAW signal as the superimposed ENV and NPX signals (c) SIG signal as the final result of the processing of the RAW signal with detected local maxima (d) local maxima concurring exactly with the original separation points.

In order to detect the maxima subjected to the postulated requirements each element of the signal is compared to its neighbouring values. An element

is classified as maxima if its value exceeds the MPH and is larger than the value of its neighbours residing within the specified MPD.

2.4 Train Start / End Detection

When porting this prototype and integrating it as a module into an existing railway monitoring system it is suggested that the train start/end-detection is implemented the following way:

1. Sample small patches (as depicted in Figure 8) from input stream and normalise them (as described in Section 2.1).
2. Analyse the patches for occurrences of vertical gradients.
3. The patch and corresponding location where vertical artefacts are found for the first time denotes the start of the train.
4. Repeat the search for vertical structures in subsequent patches.
5. If no more vertical artefacts are found for at least the expected minimal railroad car length the end of the train is detected.

Figure 8: Start / End detection: Grey rectangle illustrates an example patch sampled from the input stream

The train start/end detection component of the algorithm can also be used to define the background needed for the difference image used for the region based approach discussed in Section 2.2.2. It is just necessary to 'remember' the patches before and after the train. A simple interpolation between the two patches results in the predefined background needed.

3 Experimental Results

The evaluation of the performance of the algorithm was carried out on five different datasets containing four train panoramas each:

DATASET	DAY / NIGHT	NO. TRAINS	NO. RAILCARS
01_Graz_Nacht	NIGHT	4	52
02_Graz_Tag	DAY	4	55
03_Karlsruhe_Nacht	NIGHT	4	69
04_Karlsruhe_Tag	DAY	4	36
05_Deutschwagram_Tag	DAY	4	98

Table 1: Overview of the different datasets.

Totally, the 20 trains contain 315 separation points (including start- and endpoints) which were manually labelled. A detection was counted as correct if the calculated point was within 120 pixels from the labelled groundtruth. The value of 120 was chosen heuristically and represents an estimated 1-1,5 meters in the real-world scene. The achieved results are presented in Table 2.

DATASET	FP	FN	TP	Precision	Recall
01_Graz_Nacht	5	0	53	**91 %**	**100 %**
02_Graz_Tag	9	6	50	**85 %**	**84 %**
03_Karlsruhe_Nacht	0	0	70	**100 %**	**100 %**
04_Karlsruhe_Tag	1	0	37	**97 %**	**100 %**
05_Deutschwagram_Tag	12	5	94	**95 %**	**94 %**
TOTAL	**20**	**17**	**298**	**94 %**	**95 %**

Table 2: Results obtained by the proposed approach for different datasets.

The results show that the presented approach has a very good overall performance, especially regarding 'night' datasets. On the first view this result might be surprising since the quality of these datasets suffers from very low contrast. But this behaviour is reasonable due to the fact that for night datasets it is rather easy to separate the foreground from the background because the background is monotone black. However the algorithm shows weaknesses when applied to 'day' datasets due to the much more challenging segmentation task to be performed on scenes captured during daylight.

Overall the problematic cases seem to be limited to flatcars or unusual structures like satellite dishes, mounted right above the junctions. For example the rather weak performance of the algorithm on the 02_Graz_Nacht dataset can be explained with the facts that it is a 'day' dataset and there is a high number of flatcars compared to the other datasets.

Table 3 provides a qualitative justification for the superposition of the ENV and NPX signals. It shows the performance of the algorithm when applied to either one or the other of the two signals. The performance when both are combined is listed again.

	Precision	**Recall**
TOTAL (signal ENV)	**93 %**	**70 %**
TOTAL (signal NPX)	**82 %**	**97 %**
TOTAL (signal SIG)	**94 %**	**95 %**

Table 3: Comparison of results when using just one of the two calculated 1D signals.

In Figure 9 precision and recall are listed for different values of MPH. It shows how the MPH can be used in order to tweak precision and recall. A value of 130 for the MPH turns out to provide the most balanced scores for precision with 94% and recall with 95%. It should be noticed that with a higher MPH the precision increases as a result of the particular properties of the chosen signal. A higher value of the signal represents a higher confidence that a separation point was found.

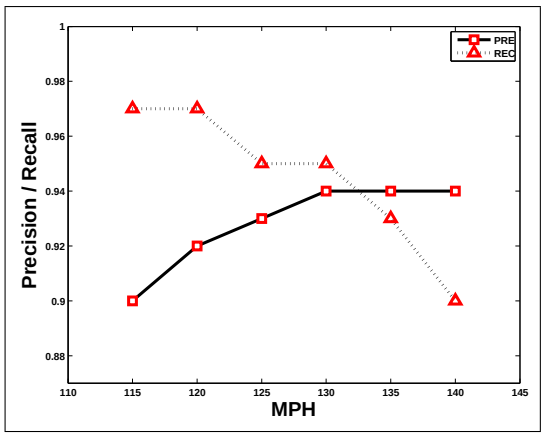


Figure 9: Performance of precision and recall at different minimum peak heights (MPH).

The mean runtime for the separation of one train on a 1,8 GHz Dual-Core processor using MATLAB was approximately 20 seconds. This strongly indicates the real-time capabilities of the algorithm.

4 Conclusion

This paper presents a novel approach for detecting separation points of railroad cars in continuous train images. The algorithm consists of two main stages. First a segmentation phase where the train itself is identified as foreground. Then based on the segmentation results a robust 1D signal is retrieved and analysed to identify the separation points of the single railroad cars.

The algorithm provides very good results especially regarding the poor image quality. Remarkably, for achieving these results only standard image processing operations are used (*i.e.*, no explicit background model is computed). This is for sure one of the main advantages of the algorithm because the runtime constraints are clearly met.

Nevertheless the algorithm shows deficiencies when separating flatcars. In order to further improve the results it will be necessary to investigate the implementation of an additional flatcar detection unit. This detection unit could make use of the general low profile of flatcars which should be easily detectable within the ENV signal. Once the location of one or more flatcars is known the separation point detection can follow different requirements.

It should also be mentioned that the installation of a plate simulating a monotonic or uniform structured background at the different setup locations would simplify the segmentation task within daylight images and therefore could dramatically improve results. Hence the results of the 'day' datasets would correlate with the ones made when processing 'night' datasets (which already have a monotonic background).

Another suggested improvement would be the implementation of a feature detector as a verification for already obtained results. This feature detector would just be applied to already detected separation points and their surrounding neighbourhood in order to verify the occurrence of a junction within this region. This procedure would only be applicable to 'day' datasets only since the low contrast within the 'night' dataset does not permit the detection of junctions.

Acknowledgement

This work has been funded by Siemens AG Österreich, CT T CEE AT.

List of Figures

List of Tables

References

[1] F. De la Torre and M. J. Black. Robust principal component analysis for computer vision. In *Proceedings IEEE International Conference on Computer Vision*, volume 1, pages 362–369, 2001. 6

[2] Union Internationale des Chemins de Fer (UIC). *Agreement governing the exchange and use of wagons between Railway Undertakings*. ETF - Editions Techniques Ferroviaires, 2000. 3

[3] D. Ha, J.-M. Lee, and Y.-D. Kim. Neural-edge-based vehicle detection and traffic parameter extraction. *Image and Vision Computing*, 22(11):899–907, 2004. 6

[4] N. J. B. McFarlane and C. P. Schofield. Segmentation and tracking of piglets in images. *Machine Vision and Applications*, 8:187–193, 1995. 6

[5] Kong Qing-Jie, Kumar Avinash, Ahuja Narendra, and Yuncai Liu. Robust segmentation of containers for intelligent train monitoring systems. In *IEEE Workshop on Applications of Computer Vision*, pages 1–6, 2009. 3

[6] Lawrence G. Roberts. *Machine Perception of Three-Dimensional Solids*. Outstanding Dissertations in the Computer Sciences. Garland Publishing, New York, 1963. 8

[7] Hanno Scharr. Optimale Operatoren in der Digitalen Bildverarbeitung, 2000. 8

[8] Milan Sonka, Vaclav Hlavac, and Roger Boyle. *Image Processing, Analysis, and Machine Vision*. Thomson-Engineering, third edition, 2007. 8, 9

[9] C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. volume 2, pages 246–252. IEEE Computer Society, 1999. 6