Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2018) 26: 987 – 998

doi:10.3906/elk-1704-214

An adaptive fault diagnosis approach using pipeline implementation for railway inspection

Yunus SANTUR*, Mehmet KARAKÖSE, Erhan AKIN
Department of Computer Engineering, Faculty of Engineering, Fırat University, Elazığ, Turkey

Received: 18.04.2017 • **Accepted/Published Online:** 13.12.2017 • **Final Version:** 30.03.2018

Abstract: Railway tracks must be periodically inspected. This study proposes a new approach for eliminating two major disadvantages experienced during rail inspection applications performed via computer vision. The first is the blurring effect on images, resulting from physical vibration during movement on the rail lines. This effect significantly reduces the high accuracy rate expected from anomaly inspection algorithms. The second disadvantage is the need to operate in real time. This study presents a new three-stage computer vision method approach that eliminates both disadvantages. First, a three-stage pipeline architecture is implemented and IMU-assisted blur detection is performed on images taken from the left and right rail lines. Next, a convolutional neural network is used for learning. In the third test stage, anomaly detection and classification training are conducted. By performing the implementation with parallel programming on graphic processing units, a highly accurate, cost-effective computer vision rail inspection, based on image processing and capable of operating in real time, is successfully carried out.

Key words: Railway inspection, blur removal, convolutional neural network, pipeline

1. Introduction

To avoid accidents, railway track components must be periodically inspected. Today this inspection is commonly carried out using computer vision systems (CVSs) [1]. A typical CVS used for rail inspection consists of a light source, a camera system, an encoder, and computer system components on which the application software will run. This equipment can be mounted to the rail transportation vehicle. Specifically designed test equipment can also be used. A rail inspection process using a CVS is expected to be cost-effective, suitable for real-time operation, fast, and highly accurate [2]. In the rail inspection process, camera images constitute the input data of the system using the CVS. However, the physical structure of the rail lines causes vibrations that blur the images. This situation may decrease the general accuracy rate of the CVS [3]. Manual fixes for blurred images are time-consuming and cannot be used in real-time inspection applications.

In image processing, an unclear image is described as having motion blur. Motion blur is created because the object to be photographed is moving at a faster speed than the focus of the camera, e.g., when the hand holding the camera moves at the moment the photograph is taken and the camera is shaken. In industrial applications, lack of clarity due to motion blur is a general disadvantage that decreases the accuracy rate of the system. In order to prevent motion blur in images, traditional methods such as fixing the camera with a tripod or using Steadicam equipment are employed. However, these solutions are not sufficient for rail inspection applications [4].

*Correspondence: ysantur@firat.edu.tr

The elimination of motion blur in image processing applications is called deblurring, and it is done by modeling the blur as a convolution on the whole image. The success of the deblurring process depends on the parameters, either manually selected or speculated, in the convolution process. In the works of Ito et al. for a Sony camera, the motion blur created by the vibration of the camera movement is described in terms of empty values in the Fourier spectrum [5]. To remove motion blur from images acquired in a series, speculating blur kernel values has been proposed.

Wang et al. investigated the blurring effect on camera images for CVS rail inspection and proposed a method for the deblurring process based on machine learning [6]. According to the study, two types of motion blur can be created on rail lines. The blur type created by instant concussions from rail vibrations is described as Gaussian blur.

A similar use of inertial measurement unit (IMU) data was proposed in another study [6] using a CVS to perform the deconvolution process accordingly in real time. In the rail line inspection application, it was possible to automatically detect the motion blur caused by the physical rail line vibrations on the images to develop data to be input into the system. Thus, a preprocessing stage of automatic deblurring was conducted in order to increase the overall accuracy rate of the rail inspection system.

Another important point in rail inspection applications using CVSs is that the system operates at high speeds. Today, rail transport systems can move at speeds of 300 km/h. The fact that the inspection process works at high speeds is extremely important in long-distance inspection applications.

Rail inspection applications using high-resolution cameras were shown to be capable of achieving a speed of 75 km/h, depending on the frame sizes of the images taken [7].

To achieve higher operating speeds, 3D-laser cameras were used instead of high-resolution cameras. In their study, Le et al. used a 3D-laser camera to find rail surface faults and achieved a real-time operating speed close to 100 km/h. However, 3D-laser camera costs are very high.

In the literature, convolutional neural networks (CNNs) (different types of neural networks) have been successfully applied to many applications such as self-driving cars, object detection and classification, industrial inspection, and image recognition [8–11]. CNNs are very similar to feedforward neural networks. They are made up of neurons that have learnable weights and biases. Each neuron receives some inputs, performs a dot product, and optionally follows it with nonlinearity. However, CNNs are easier to train and have far fewer parameters than fully connected networks with the same number of hidden units. Therefore, a CNN is used as a machine-learning method in this study.

Consequently, an approach using a CNN-based three-stage pipeline architecture with parallel GPU programming for rail inspection applications via a CVS is proposed in this study. In the first step, motion blur is detected by analyzing the images taken from the rail lines along with IMU data. Then the deblurring process is adaptively performed. In the second step, system learning was carried out with CNNs using the learning data set. In the test phase, the performance values of the trained system were compared by obtaining the accuracy performance and operating speed data over a 5-km section of rail line.

2. Railway analysis

Information about rail line components and the types of failures occurring in these components, as provided by studies in the literature, will be analyzed in this section. The basic components constituting the rail lines are given in Figure 1.

The part over which the rail vehicle wheels move constitutes the most significant component of rail

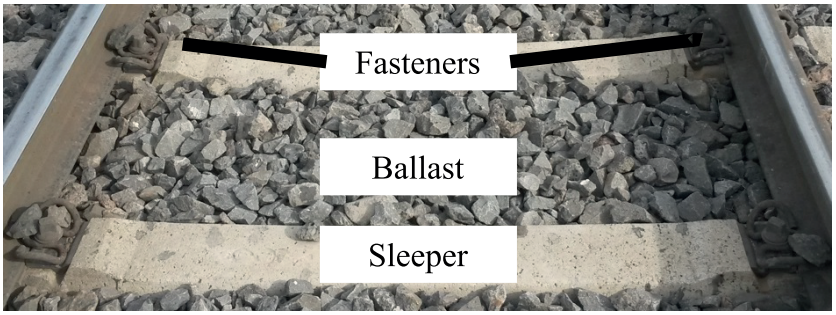


Figure 1. Rail line components.

analysis in terms of transportation security. The building material consists of wood, concrete, or iron and transmits the load on the rail line to the ballast.

The railroad building material consists of pebble stones, which distribute the load equally on the sleeper. The unit has different geometrical shapes (hexagonal or round) that connect the rail line to the sleeper.

Rail line inspection is traditionally performed by manual and visual examination using human labor. However, this type of inspection is slow and unsafe and, most of all, it remains limited to the knowledge of a competent person. Another method is based on inspecting the rails using mechanical devices. This method, called contact-based, diagnoses faults by using graphs obtained from the friction between the mechanical device and the rail. It is fast and provides accurate results; however, because it requires contact with the rail, it may damage the rails or increase existing damage [12].

In the works of Hackel et al., laser cameras were used for detecting anomalies such as missing screws and traverse and rail fractures [13]. Aytekin et al. [14] detected deficient bolts on rail lines using a laser camera. The main advantage of conducting rail inspection processes using 3D-laser cameras is the higher operating speed and measurement accuracy, whereas the very high cost constitutes the main disadvantage.

Huber-Mörk et al. carried out a study classifying rail surface faults based on image processing [15]. In their study, faults were classified as wear, puncture, and crack and were grouped by type. A 97% accuracy rate was achieved in this study.

Railway analysis studies in the literature usually involve the detection of these components based on image processing and identifying and/or classifying any deficiencies and failure conditions that occur in them [16–18]. Santur et al. proposed a method to detect rail faults using a laser camera [16]. Karaköse et al. carried out a study on rail diagnosis based on the fuzzy integral [17]. In the works of Li et al., morphological image processing was used to detect rail surface faults [18].

This study proposes a cost-effective approach that addresses all the disadvantages seen in the literature. This approach is based on CNNs that are suitable for pipeline architecture. It uses two cameras to monitor both sides of the rail lines, three GPUs, and one computer compatible with the Nvidia CUDA.

3. Proposed approach

A general block diagram of the proposed approach method is given in Figure 2. In the first step, the obtained left and right line images were processed simultaneously on GPU-1 and GPU-2, and the image from which the noise and motion blur were removed was transferred to GPU-3 as the data matrix. In the second pipeline stage, GPU-3 was used for the CNN learning model. In the third pipeline stage, anomaly detection classification was performed on both image frames in a CNN previously trained with data on the computer [9].

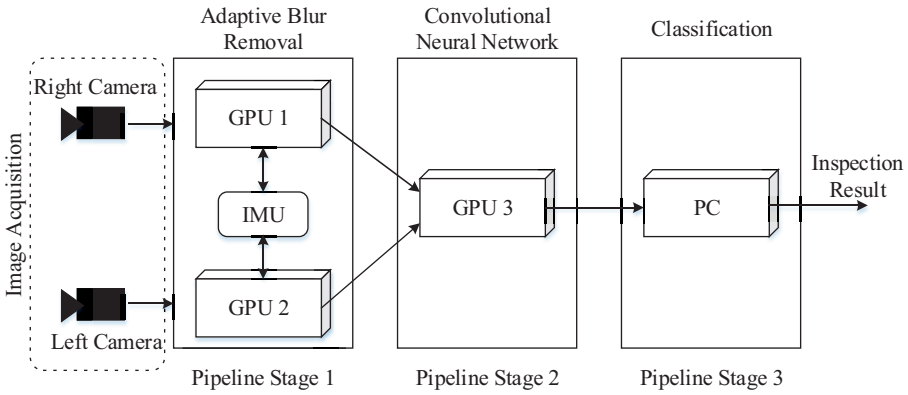


Figure 2. Proposed approach.

3.1. First pipeline stage (blur detection and deblurring process)

In this application, an IMU-based approach was proposed to detect Gaussian blur on the image and to implement the deblurring process in an adaptive manner. A block diagram of this stage is given in Figure 3. IMUs are used in autonomous mobile robots and in seafaring, aerial, and space vehicles. These systems have accelometers, gyroscopes, and magnetometers within them and are used to find positional information and establish balance in control applications [2]. IMUs are used in industrial applications and are selected for their desirable features, which include degrees of freedom (DOFs), good working speed, and precision measurement. The highly accurate Xsens 10 kHz 10-DOF MTi 100-series IMU was used in this application [19]. In Euler space (Figure 3), the angle of the object in the x-axis is called roll, in the y-axis pitch, and in the z-axis yaw. To find Euler angles, transformation matrices, Kalman filters, and the open-source attitude heading reference system (AHRS) algorithm, developed by Sebastian Madgwick, are commonly used [20].

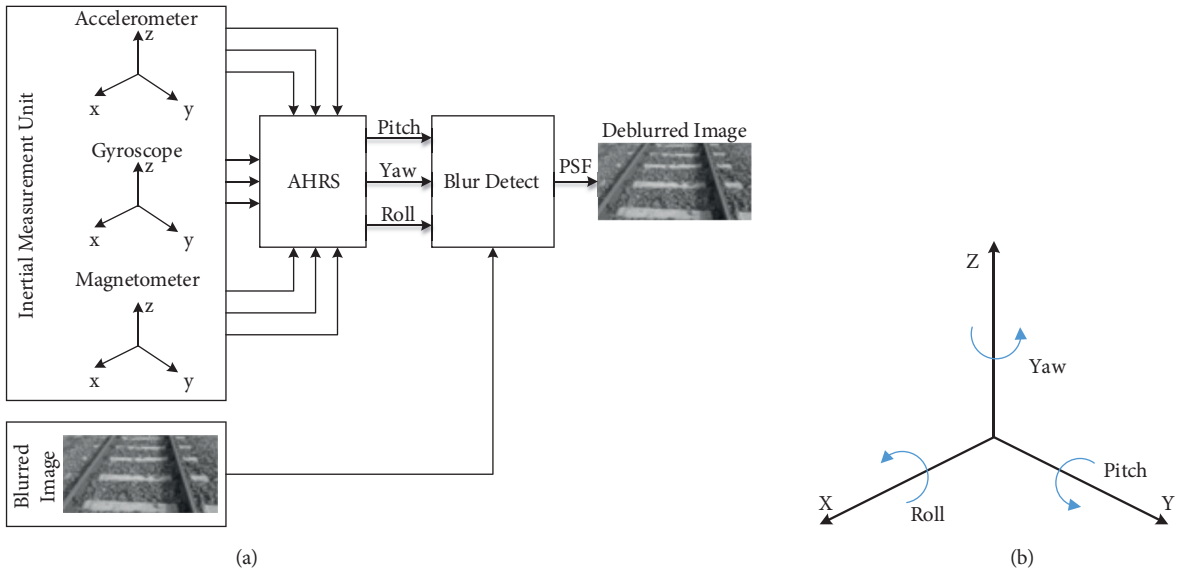


Figure 3. (a) IMU-assisted deblurring process, (b) Euler angles.

3.2. AHRS algorithm

AHRS is an algorithm used to find the Euler angles of an object in a three-dimensional space, as shown in Figure 3. The AHRS algorithm computes the Euler angles using the IMU data as input data. The camera moves instantly due to the vibration that occurs on the rail lines and a blurring effect occurs on the frames. In this study, the AHRS algorithm is used to detect camera vibrations. The adaptive deblurring process has two steps:

- Step 1 (learning): The blurred frames collected on the rail line are manually deblurred, the best threshold value obtained in the deblurring process is recorded with the time stamp, and the Euler angles are obtained from the AHRS algorithm. Thus, Euler angles and deblurring threshold value pairs are obtained.
- Step 2 (test): The new images obtained in the test phase and the simultaneously measured Euler angles are compared. The images above the threshold value are accepted as blurred, and the deblur parameters are adaptively selected on these images with the closest class obtained in the first step.
- The AHRS algorithm updates the quaternion vector given in Eq. (1) by computing with the IMU data, and the new quaternion vector provides the rotation matrix and Euler angles. IMU and AHRS units are given in Table 1. The initial rotation matrix is given in Eq. (2). The Euler angles (pitch, yaw, and roll) are calculated as in Eqs. (3–5) [20].

Table 1. IMU and AHRS units.

IMU units	Data	Max. DOF	Unit
Accelerometer	Acceleration speed	x, y, z	m/s^2
Gyroscope	Change in angular speed	x, y, z	rad/s
Magnetometer	Magnetic zone	x, y, z	au
AHRS units	Output value (degrees)	Fault sensitivity	
Pitch	–90 . . . +90	0.2 . . . 0.4	
Yaw	–180 . . . +180	0.2 . . . 0.4	
Roll	–180 . . . +180	0.2 . . . 0.4	

$$q = [q_0\, q_1\, q_2\, q_3] \tag{1}$$

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

$$p = arctan?\frac{2(q_0q_1 + q_2q_3)}{q_0^2 - q_1^2 - q_2^2 + q_3^2} \tag{3}$$

$$y = -arcsin?(2(q_1q_3 - q_0q_2)) \tag{4}$$

$$r = arctan\frac{2(q_0q_3 + q_1q_2)}{q_0^2 + q_1^2 - q_2^2 - q_3^2} \tag{5}$$

This vibration value will be deemed a concussion if it is higher than the threshold value, as shown in Eq. (6), and it will be regarded as having created a blurring effect on the image. The blur kernel amount estimated for the deblurring process will be the amount of change in these angles. The blurring process can be defined mathematically as in Eq. (7). The motion blur is created after the convolution of the input image (I) and blur kernel (K) matrices, with the addition of noise (N). The aim here is to find the net image after the convolution process. Therefore, the blurred image needs to be deblurred with the blur kernel K value. In this study, the point spread function (PSF) selects the nearest value obtained from Step 1. The simplified mathematical model of the system is given in Eq. (8).

$$D = \begin{cases} 1, if\, p\, |\, y\, |\, r > trh \\ 0, otherwise \end{cases} \tag{6}$$

$$B = I \otimes K + N \tag{7}$$

$$PSF = (D\alpha). \begin{cases} |p_t - p_{t-1}| \\ |y_t - y_{t-1}| \\ |r_t - r_{t-1}| \end{cases} \tag{8}$$

3.3. Second pipeline stage (learning stage CNN)

In the learning and test stages, healthy and faulty rail images are collected for use in learning algorithms. The data of these two classes are trained in compliance with the CNN algorithm. A typical CNN has three main layers, as shown in Figure 4.

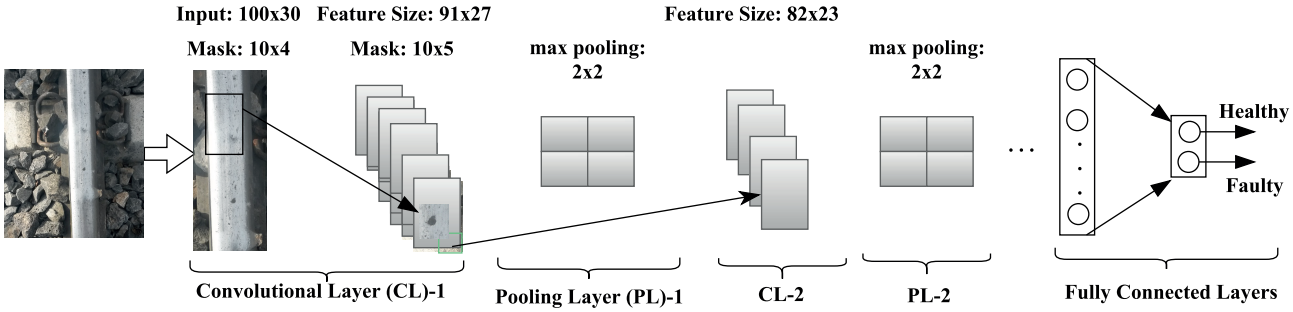


Figure 4. CNN model for the proposed approach.

1) Convolutional layer: The convolutional layer is the core building block of a CNN. This layer's parameters consist of a set of kernels that have a small receptive field but extend through the full depth of the input volume. During the forward pass, each filter is convolved across the width and height of the input volume, computing between the entries of the filter and the input and producing a two-dimensional activation map of that filter.

$$C_{ij}^{l} = \sum_{a=0}^{m-1} \sum_{b=0}^{m-1} w_{ab} f_{(i+a)(j+b)}^{l-1} \tag{9}$$

The convolutional layer's mathematical model is given in Eq. (9). In the equation, f represents the input image value, W represents an $m \times n$ dimension kernel matrix, and C represents the count of subsampling of this layer.

2) Pooling layer (feature selection): The pooling layer is a form of nonlinear downsampling. There are several methods for implementing pooling, among which max pooling is the most common. The most common form is a pooling layer with applied filters of 2×2 in size (Figure 4).

3) Connected layers: With several convolutional and max pooling layers, the high-level reasoning in the neural network is performed via fully connected layers. Neurons in a fully connected layer are fully connected to all activations in the previous layer, as seen in regular neural networks.

3.4. Third pipeline stage (test stage CNN)

In the test stage, a rail that contains faulty and healthy images not previously used in the system is selected. Then frames from this railway are acquired again and the results are compared with the learned CNN model.

3.5. Test analysis

In machine-learning applications, the confusion matrix given in Table 2 is used to obtain the accuracy (Acc), precision, and recall values of the system. The confusion matrix for the proposed method is as follows. The description of true positive (TP), false negative (FN), true negative (TN), and false positive (FP) values are given below.

Table 2. Confusion matrix.

	Prediction		
Actual		P (Healthy)	N (Faulty)
	P (Healthy)	TP	FN
	N (Faulty)	FP	TN

TP: Rail frames that are classified as "healthy" by the system and are actually "healthy" (correctly classified as "healthy" by the system).

TN: Rail frames that are classified as "faulty" by the system and are actually "faulty" (correctly classified as "faulty" by the system).

FP: Rail frames that are classified as "healthy" by the system and are actually "faulty" (wrongly classified as "healthy" by the system).

FN: Rail frames that are classified as "faulty" by the system and are actually "healthy" (wrongly classified as "faulty" by the system).

$$TPR = \frac{TP}{TP + FN},\ FPR = \frac{FP}{FP + TN},\ Acc = \frac{TP + TN}{N} \tag{10}$$

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

$$F_1\ score = \frac{2TP}{2TP + FP + FN} \tag{12}$$

The confusion matrix in Table 2 gives the sensitivity value of the true positive rate (TPR) of the system, given in Eq. (10). This value is also known as the recall value. An ideal system is expected to have a TPR value close

to 1 in a range of 0 to 1 and a false positive rate (FPR) value close to 0. Acc gives the overall accuracy of the system. A positive is obtained by dividing the graded images (rail images classified as "healthy" for this study) by the number of samples, as shown in the precision TP sample number via Eq. (11). Finally, the F_1 value is given by Eq. (12). The software and hardware components used in the proposed system are given in Table 3.

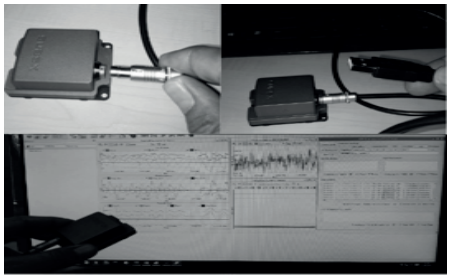
Table 3. Hardware and software components used in the proposed method.

Component	Explanation
Operating system	Ubuntu 16.04
GPU	Nvidia Geforce GTX 750
Camera	Mako G032B/C [21]
Programming language	Python tensorflow [22]
Image processing/deblurring	Python numpy, scipy, scikit libraries
AHRS	Programmed in the Python

4. Experimental results

This study aimed to detect anomalies on rail lines. Experimental studies were carried out using a three-stage pipeline architecture in the training and testing phases. For the training phase, a data set consisting of rail images was collected on the rail line. For this purpose, a special transportation device was designed to carry equipment (i.e. GPUs, computer, cameras, and an IMU) and be manually driven on the rail line (Figure 5).

a) Experimental test tool



b) Xsens mti-100 imu

c) Mako G032B/C Camera

Figure 5. (a) Experimental test tool, (b) Xsens MTi-100 IMU, (c) Mako G032B/C camera.

In the results, as expected, greater changes in IMU data caused the blur rate to increase. In the case of too much concussion, although blur detection was performed successfully, the deblurring process could not be performed. This is a common problem with image processing applications when the blur rates are higher. The blur detection and deblurring results are shown in Figure 6, along with the changes in pitch, yaw, and roll Euler angles obtained by the AHRS algorithm.

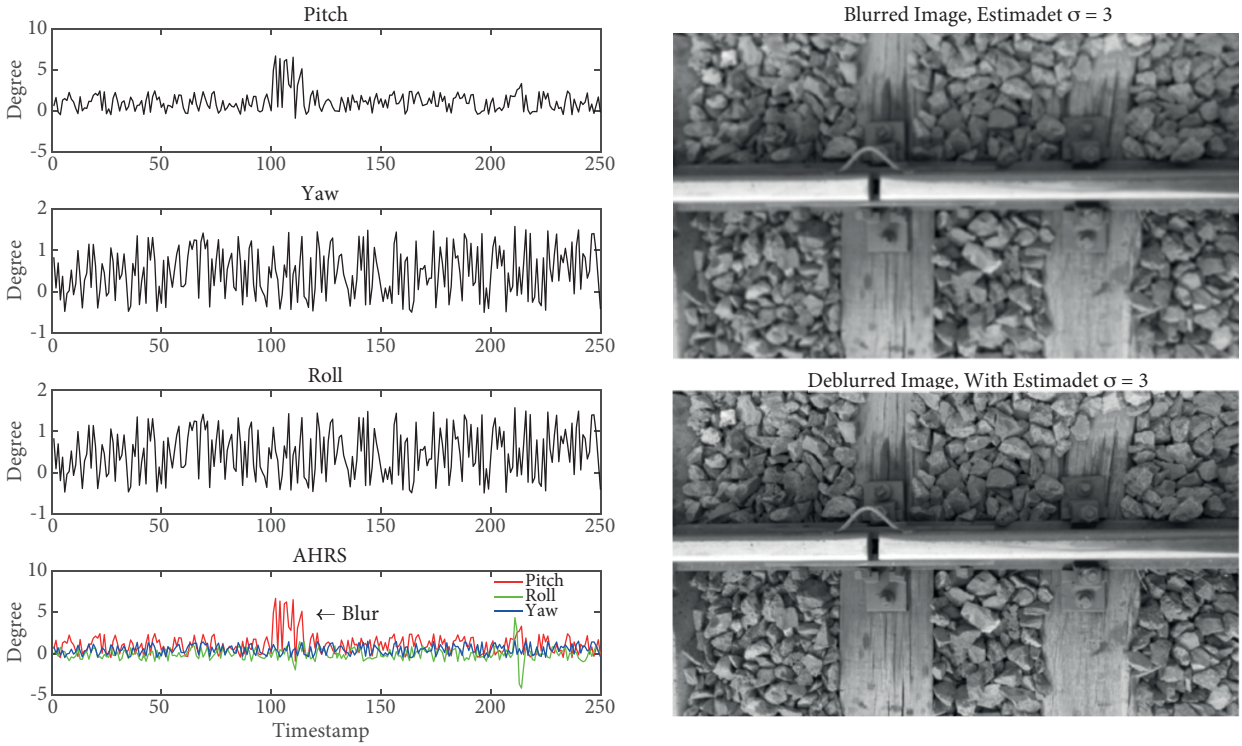


Figure 6. AHRS output data, blur detection, and deblurred image.

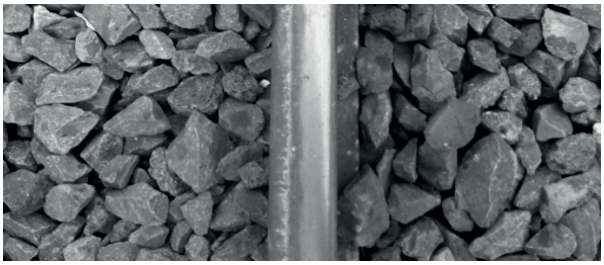
In the experimental studies, blur detection, deblurring, and the classification process with the CNN could be performed at 40 frames per second. This value indicated that the system could run on data sets collected from a testing device moving at the speed of 144 km/h based on image processing. Real-time operating speed was calculated by taking the *fov* value as 50 cm and was determined to be 72 km/h.

The first convolution layer takes a normalized image and filters it with kernels of 10 × 4 pixels. The second convolution layer takes the pooled feature map of the first layer and filters it with kernels of 10 × 5 pixels. The kernel size of the third convolution layer is 5 × 2 pixels. In this model, max-pooling units of 2 × 2 pixels are used. The activation function is hyperbolic tangent function. After three convolutional and max-pooling layers, the high-level reasoning in the CNN is performed via fully connected layers. In the learning stage 3500 railway images and in the test stage 2000 railway images were used.

Some of the classification results obtained in the test phase from the network and trained to use the CNN for anomaly detection are seen in Figure 7. According to the classification of the images, (a) had no detected anomalies and is classified as TP, (b) is classified as FN, (c) is classified as TN due to abrasion-related rail faults, and (d) is classified as FP. The performance of the proposed system is given in Table 4. The values shown in the table are the output values of the method. The output values of the CNN were manually controlled and the confusion matrix was created.

For the proposed method, the run times obtained after pipeline implementation and the total time spent for each image are given. In the training and testing stages, each image obtained with the high-resolution cameras from the rail line has a high resolution of 658 × 492. The proposed method, when running on a GPU with Nvidia CUDA, blur detection, deblurring operation, and classification, can be performed for 40 images in 1 s.

a) Healthy frame as TP (Correctly classified as "healthy")



b) Faulty frame as FN (Wrongly classified as "faulty")

b) Faulty frame as TN (Correctly classified as "faulty")

c) Healthy frame as FP (Wrongly classified as "healthy")

Figure 7. Test stage experimental inspection results.

Table 4. Confusion matrix for the proposed method.

	Prediction		
Actual		P	N
	P	1856	43
	N	3	98
	TPR	Precision	$F_1\ score$
Test performance	0.977	0.99	0.98

5. Conclusion

In general, there are three criteria expected from a rail inspection application. These are highly accurate results, high operating speed, and cost-effective implementation of the system. The images constitute the input data of the system in the application using a CVS. These rail images must be clear and noiseless in order to obtain a high rate of accuracy. However, the vibrations caused by physical conditions on the rail lines lead to motion blur on the images that are to be the input data of the system. For the real-time operation of the application, the motion blur must be dynamically and quickly detected and eliminated. Therefore, this study proposed the use of an IMU along with cameras in the rail inspection system. In the study, pitch, yaw, and roll values in Euler space were obtained by combining sensor data obtained from the IMU with the AHRS algorithm. The PSF value required for blur detection and the deblurring process could be dynamically found through the changes in these values. In the blur detection process, the TPR was 98% and the FPR was 2%. The TPR value in rail line anomaly detection was obtained at the high accuracy rate of 97%. A precision of 99% and an F_1 score of 98% were obtained.

The second criterion of success expected from rail inspection applications is high operating speed. For

this purpose, some rail inspection systems use 3D-laser cameras that ensure high accuracy and high operating speed due to their measurement precision. However, their high cost is their main disadvantage.

The third criterion of success expected from rail inspection applications is the cost of implementing the system, which should be feasible and economical. High speed and accuracy can be obtained in applications using 3D-laser cameras; however, their cost is quite high. In this study, an IMU-supported, cost-effective inspection method using a high-speed industrial camera was presented.

A benefit of the study is that learning and testing are done with CUDA programming and the pipeline architecture is on Nvidia GPUs. This approach is 10–20 times faster than traditional programming and that depends on the GPU properties used.

The recommended system can perform blur detection, deblurring operations, and classification of 40 images at a resolution of 658 $\times$ 492 in 1 s. This value can be regarded as the worst running speed; however, since blurring does not occur on every image in real-time applications, the deblurring process will not need to be performed for every image. The main disadvantage of the proposed method is that, although the blur can be detected when the vibration is excessive, the deblurring operation cannot be successfully performed in those cases. As a result, a cost-effective, real-time rail inspection approach that uses a three-stage pipeline architecture was presented.

Acknowledgment

This work was supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK, Grant No. 114E202).

References

[1] Koch C, Georgieva K, Kasireddy V, Akinci B, Fieguth P. A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure. Adv Eng Inform 2015; 29: 196-210.

[2] Santur Y, Karaköse M, Akın E. IMU based adaptive blur removal approach using image processing for railway inspection. In: IEEE 2016 International Conference on Systems, Signals and Image Processing; 23–25 May 2016; Bratislava, Slovakia. New York, NY, USA: IEEE. pp. 1-4.

[3] Gibert X, Patel VM, Chellappa R. Deep multitask learning for railway track inspection. IEEE T Intell Transp 2017; 18:153-164.

[4] Flusser J, Suk T, Boldys J, Zitova B. Projection operators and moment invariants to image blurring. IEEE T Pattern Anal 2015; 37: 786-802.

[5] Ito A, Sankaranarayanan AC, Veeraraghavan A, Baraniuk RG. Blurburst: removing blur due to camera shake using multiple images. ACM T Graph 2014; 3: 1-15.

[6] Wang L, Hang Y, Luo S, Luo X, Jiang X. Deblurring Gaussian-blur images: a preprocessing for rail head surface defect detection. In: IEEE 2011 International Conference on Service Operations, Logistics and Informatics; 10–12 July 2011; Beijing, China. New York, NY, USA: IEEE. pp. 451-456.

[7] Li Q, Ren S. A real-time visual inspection system for discrete surface defects of rail heads. IEEE T Instrum Meas 2012; 61: 2189-2199.

[8] Nvidia. Programming Guide: CUDA Toolkit Documentation. Santa Clara, CA, USA: Nvidia, 2008.

[9] Roohi S, Hajizadeh S, Núñez A, Babuska R, De Schutter B. Deep convolutional neural networks for detection of rail surface defects. In: IEEE 2016 International Joint Conference on Neural Networks; 24–29 July 2016; Vancouver, Canada. New York, NY, USA: IEEE. pp. 2584-2589.

[10] Weimer D, Scholz-Reiter B, Shpitalni M. Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection. CIRP Ann-Manuf Tech 2016; 65: 417-420.

[11] Liu Z, Sun J, Wang H, Zhang G. Simple and fast rail wear measurement method based on structured light. Opt Laser Eng 2011; 49:1343-1351.

[12] Li Q, Zhong Z, Liang Z, Liang Y. Rail inspection meets big data: methods and trends. In: IEEE 2015 International Conference on Network-Based Information Systems; 2–4 September 2015; Taipei, Taiwan. New York, NY, USA: IEEE. pp. 302-308.

[13] Hackel D, Stein I, Maindorfer ML, Reiterer A. Track detection in 3D laser scanning data of railway infrastructure. In: IEEE 2015 Instrumentation and Measurement Technology Conference; 11–14 May 2015; Pisa, Italy. New York, NY, USA: IEEE. pp. 693-698.

[14] Aytekin C, Rezaeitabar Y, Dogru S, Ulusoy I. Railway fastener inspection by real-time machine vision. IEEE T Syst Man Cy-S 2015; 45: 1101-1107.

[15] Huber-Mörk R, Nölle M, Oberhauser A, Fischmeister E. Statistical rail surface classification based on 2D and 21/2D image analysis. In: 2010 Conference on Advanced Concepts for Intelligent Vision Systems Conference; 13–16 December 2010; Sydney, Australia. pp. 50-61.

[16] Santur Y, Karaköse M, Akın E. Learning based experimental approach for condition monitoring using laser cameras in railway tracks. Int J Appl Math, Electron 2016; 4: 1-5.

[17] Karaköse M, Aydin, Akin E. The intelligent fault diagnosis frameworks based on fuzzy integral. In: IEEE 2010 Power Electronics Electrical Drives Automation and Motion International Conference; 14–16 June 2010; Pisa, Italy. New York, NY, USA: IEEE. pp. 1634-1639.

[18] Li Y, Zuo MJ, Lin J, Liu J. Fault detection method for railway wheel flat using an adaptive multiscale morphological filter. Mech Syst Signal Pr 2017; 84: 642-658.

[19] Xsens Technologies. MTi User Manual. Enschede, the Netherlands: Xsens Technologies, 2014.

[20] Madgwick SO. An efficient orientation filter for inertial and inertial/magnetic sensor arrays. Bristol, UK: University of Bristol, 2010.

[21] Mako G-032. User Manual. Stadtroda, Germany: Allied Visions Technology, 2008.

[22] Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Kudlur M. TensorFlow: a system for large-scale machine learning. In: USENIX 2016 Conference on Operating Systems Design and Implementation; 2–4 November 2016; Savannah, GA, USA. Berkeley, CA, USA: USENIX Association. pp. 265-283.