Construction and Building Materials 384 (2023) 131472

Contents lists available at ScienceDirect

Construction and Building Materials

journal homepage: www.elsevier.com/locate/conbuildmat

Comparison of deep learning and analytic image processing methods for autonomous inspection of railway bolts and clips

E. Aldao *, L. Fernández-Pardo, L.M. González-deSantos, H. González-Jorge

Research Institute of Physics and Aerospace Science, University of Vigo, Campus Ourense, 32004 Ourense, Spain

ARTICLE INFO

Keywords:
Railway maintenance
Optical inspection
Image processing
Deep learning

ABSTRACT

In this work, different methods are proposed and compared for autonomous inspection of railway bolts and clips. A prototype of an autonomous data acquisition system was developed to automatically obtain information of the state of the railway track using LiDAR and camera sensors. This system was employed in a testing railway track installed in the facilities of the University of Vigo to obtain the images used in this work. Then, the images were further processed using analytic image segmentation algorithms as well as a neural network to detect the bolts and clips. Once these elements are detected, their relative position is computed to evaluate if there is any missing component. Finally, the orientation of the clips is computed to ensure that all the bolts are correctly placed. Four different methods were implemented, and their performance was evaluated using the segmentations provided by the analytical methods and the neural network.

1. Introduction

High-speed railway track maintenance is a crucial task to ensure the safety and a correct development of the operations [1]. This task was traditionally carried out by human operators who manually inspected all the elements of the track and fixed the pathologies they found. This is a process that requires a large amount of time and human resources and can lead to delays in railway services. In fact, in 2018, the EU countries spent approximately 20.8 billion euros in the maintenance of their railway tracks according to a report of the European Commission [2]. Apart from the direct maintenance costs, it is also necessary to consider the indirect costs associated with delays and accidents, which the transport companies must assume [3]. The correct allocation of resources and the scheduling of preventive maintenance tasks [4,5] play a fundamental role in this aspect [3,6].

The future perspectives of railway maintenance go through the automation of operations. The autonomous acquisition of data, as well as its subsequent verification, allow to considerably reduce the labour costs, that can represent around a 25% of the total maintenance expenses [7]. In addition, the inspection would be carried out in a systematic way, eliminating the subjectivity of human operators. In recent years, several companies have developed autonomous track inspection systems [8,9]. They are typically motorized land vehicles, locomotives, or UAVs (Unmanned Aerial Vehicles) that have sensors installed to acquire information on the state of the infrastructure [10–12]. This information is georeferenced using GNSS measurements, and if any type of anomaly is detected, a human operator is sent to the indicated location to carry out the necessary maintenance tasks.

This will allow infrastructure management companies to considerably reduce the costs of inspection and personnel required. Due to the large volume of resources needed, railway infrastructure operators, such as ADIF in Spain, often contract part of the construction and maintenance works to external companies [13]. The University of Vigo collaborates with one of these contracting companies, COPASA [13], in the RIIM (Railway Inspection and Information Model) project, financed by the Ministry of Science and Innovation. The project is devoted to the development of autonomous inspection systems, being the present work part of its results.

COPASA works on the maintenance of international and Iberian gauge tracks, among others. This last type of track, in accordance with ADIF technical regulations [14,15], uses SKL type clips, which must be inspected periodically to ensure that they are correctly positioned. These inspections are carried out manually by trained operators, which entails a high cost of time and human resources. Ferreño et al. conducted several experimental and finite element structural analyses of SKL-type clips in which they studied the resistance to static and fatigue loads of these fasteners [16]. According to their results, SKL-type clips have a very high safety factor, and it is very unlikely that they will suffer

* Corresponding author.
E-mail address: enrique.aldao.pensado@uvigo.es (E. Aldao).

https://doi.org/10.1016/j.conbuildmat.2023.131472
Received 23 January 2023; Received in revised form 13 April 2023; Accepted 16 April 2023
Available online 23 April 2023

structural failures during their useful life. However, clamp bolts are very susceptible to loosening due to vibrations and must be inspected periodically [17,18].

Jing et al. conducted a review of the main trends and technologies used in autonomous railway inspection systems [10]. According to the authors, the common approach for inspecting the condition of railway fasteners is the use of optical cameras, due to their high resolution, fast data acquisition speed, and relatively affordable information processing. There are other alternatives, such as structured light sensors, which, using a set of cameras and a projector of a light pattern, generate a high-resolution 3D point cloud of the geometry of the environment [19].

Cui et al. presented various methodologies to detect and verify the positioning of clips and bolts in 3D point clouds generated by structured light sensors [19]. They developed point cloud processing algorithms based on neural networks such as PointNet++, PointSIFT, and PointCNN from various databases. In this way, they segmented the point clouds and detected improperly positioned clips and bolts. Structured light sensors offer great precision and detail in the geometry of these elements, which facilitates the detection of structural pathologies. However, the main drawback of this technology is the high cost and complexity of the sensors, which makes their use less common in this type of inspections. Most works use RGB cameras since their cost is much lower and they are lightweight and versatile equipment that can be used simultaneously for other tasks, such as inspecting cracks in concrete sleepers.

Regarding inspection using RGB cameras, previous works have focused on detecting damages and deformations of clips [20–24]. Although this may be a common phenomenon in other types of tracks, it is a different problem than the one addressed in this work. The authors could not find an equivalent methodology to the one presented by Cui et al. based on RGB image processing to verify the positioning of these elements. For this reason, the main novelty of this work is the proposal and comparison of various image processing techniques for the detection and positioning of clips and bolts using RGB images. The proposal of this work is analogous to that presented by Cui et al. [19], but using inexpensive equipment instead of structured light sensors. For this purpose, a low-cost inspection system based on two Sony Alpha 6000 cameras was developed.

In order to study the positioning of clips and bolts, the first necessary step is to detect them within the images. Wei et al. conducted a survey on the different methods of fastener detection in images categorizing them into three main families [20]: those based on image processing, which use filters on the image to detect specific characteristics such as edges or determined shapes and colours [21–23]; those based on feature extraction, which use feature extraction algorithms that obtain a set of characteristics that are fed to a classifier such as a Support Vector Machine (SVM) [24,25]; and those based on Convolutional Neural Networks (CNN), which, based on training, detect different elements in the image without the need for an easily recognizable geometric feature [26–32]. There are also hybrid approaches, such as the one presented by Aydin et al. The authors used a CNN to extract a feature vector from the image, and then used a SVM to classify the state of the clips and bolts based on this vector [33].

In this work, two different image segmentation methods were used: one based on colour image processing and a Convolutional Neural Network DeepLab V3+ [34]. They provide a pixel-wise classification, which allows to detect the different elements of the image. This information is used to verify that the clips and bolts are placed properly. The first step is to verify, that each sleeper has its corresponding clip and bolt and that these elements are concentric. To check that the clips are not loose, their relative angle with respect to the rail is calculated. For this, four algorithms for calculating the orientation of objects in images are proposed.

Within the algorithms for calculating the orientation of objects in images, we can differentiate two categories of methods according to their approach. On the one hand, methods based on geometric relationships, such as the work done by Yi and Marshall [35], who calculated the principal directions of a set of points to detect the orientation of elements in an image. On the other hand, Deep Learning techniques, which, through a training process, allow detecting the orientation of certain elements in an image, such as the works presented by Shahin et al. and Itakura et al. [36,37].

In this study, four different methods are proposed: two based on geometric relations (analytic methods) and another two based on neural networks. The main objective of this work is to compare the effectiveness of these different methods in a specific case of inspection of railway clips. To do this, the performance of all of them was tested in terms of precision and calculation time, using data acquired from a test track installed at the University of Vigo. This process was carried out for all the proposed algorithms and the influence of the segmentation algorithm on the results was verified.

The rest of the manuscript is organized as follows: Section 2 presents the materials used in this work. In Section 3, the methods are introduced and described. In Section 4, the performance of the different methods is compared and analysed, and finally, in Section 5 the main conclusions of this work are presented.

2. Materials

The test railway track used for RIIM project at the University of Vigo facilities is shown in Fig. 1. It is an Iberian gauge track with a total length of five meters that was used for the development of prototypes and inspection software.

For data acquisition, the low-cost prototype represented in Fig. 1 a) was developed. The system incorporates two Sony Alpha 6000 cameras to verify the positioning of clips and bolts. This is a low-cost solution compared to the work of Cui et al, in which the authors used structured light sensors [19]. The main advantage of this approach, in addition to its lower economic cost, is that by using cameras, it allows to simultaneously inspect the condition of the concrete sleepers and check for the presence of cracks.

Apart from the cameras, the system incorporates two solid-state Livox Avia LiDAR sensors to inspect the ballast surface. These devices generate 3D point clouds that can be used to detect changes in the ballast such as voids and deformations. LiDAR sensors are particularly effective at detecting these types of pathologies at relatively long distances. However, their precision and resolution at short distances are much more limited than structured light sensors, making them ineffective for inspecting the positioning of clips and bolts. For this reason, cameras were used to inspect the clips and bolts, contrary to the work presented by Cui et al. [19]. It is important to note that although the inspection prototype was developed to detect defects in the ballast shoulder and cracks in the concrete sleepers, these functions are not within the scope of this study, but rather the inspection of clips and bolts through RGB images.

In order to correct the LiDAR measurements and determine the pose of the camera in each photograph, the data sampling is synchronized

a) b)

Fig. 1. Inspection system prototype. a) Inspection trolley. b) Displacement measurement system.

with the motion of the trolley using two rotatory encoders Sick HTL DBS36. For this, the system shown in Fig. 1 b) was developed. As it can be appreciated, it consists of a wheel in contact with the rail that rotates with the movement of the trolley. This rotation is transmitted to the encoder through a pulley and a reduction gear. To ensure a good contact and friction with the track, the system has two springs that, through a lever effect, press the wheel against the surface of the rail.

For controlling and synchronizing data acquisition a Raspberry 4B was used, as depicted in the scheme of Fig. 2. This device receives in real time the pulses from the encoder through the GPIO ports and calculates the displacement of the trolley. At the same time, it also receives distance measurements from the LiDAR sensor, which are corrected considering the position of the trolley at each instant of time. Besides, it also controls the camera shutter of the Sony Alpha through a USB interface and captures a picture of the track per meter of displacement. All the information is stored in a USB 3.0 flash drive.

The described system was used in the test track of the University of Vigo for data acquisition. This manual pushing trolley is only a prototype for the development and testing of algorithms. It does not represent a final product meant for improving the efficiency of railway track inspections. In the future, to obtain completely autonomous inspections without an operator pushing the trolley, the sensors will be installed in a system embedded in a train and the algorithms will be adapted for this purpose accordingly.

3. Methodology

Using the acquisition system described in the previous section, a total of 128 pictures of the test track were obtained and processed following the methodology depicted in Fig. 3. The first step is Image Segmentation (Section 3.1) to detect the different elements of interest in the photographs. Two methods were used, one based on analytic image processing, and other based on a neural network. Once the different elements are separated, the next step (Section 3.2) consists in checking that each sleeper has a bolt and a clip, as well as ensuring that they are placed in their corresponding position. Finally, to verify that the clip is correctly fastened and aligned, the relative angle with respect to the rail is computed. For this purpose, four different methods were developed as it will be explained in the following sections.

3.1. Image segmentation

3.1.1. Image processing segmentation

For the segmentation of the image using analytic algorithms, firstly, a coarse segmentation by colour of the rail and clips is carried out. As can be seen in Fig. 4, both have characteristic reddish and greenish colours that are well differentiated from the rest of the image. Specifically, to obtain a more robust segmentation under lighting changes, the RGB image was transformed to the CIE Lab colour space [38]. This colour representation includes two hue components (a*, b*) and another component of the lighting level (L*).

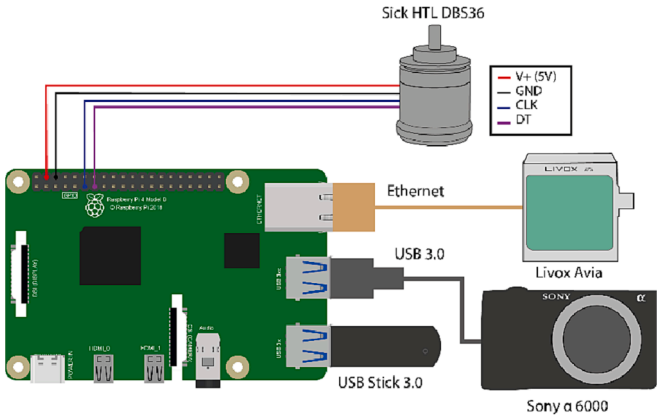


Fig. 2. Connection scheme.

This colour space allows us to filter the pixels of the clips and the rail by hue components, that do not depend on the illumination of the scene [26]. Fig. 5 presents a point cloud of the Lab coordinates of the pixels of a track image, along with the filtering criteria used in this work. Ideally, the hue components of the rail and clips follow a gaussian distribution with average $\vec{\mu} = (\vec{a}^*, \vec{b}^*)$ and covariance matrix $\overline{\overline{\Sigma}}$. From manually segmented images, the components a* and b* of the two categories were extracted and their corresponding values of $\vec{\mu}$ and $\overline{\overline{\Sigma}}$ were estimated. Graphically, $\vec{\mu}$ and $\overline{\overline{\Sigma}}$ represent an ellipse in the (a*, b*) plane of the Lab Colour space, centred in the point $\vec{\mu}$ and with semi-axis given by $\overline{\overline{\Sigma}}$. To filter the points of the clips and the rail, the elliptical cylinder of 95% confidence (semi-axes given by $2\overline{\overline{\Sigma}}$) was considered for the two categories, and the pixels whose hue components are within the surface were extracted, as represented in Fig. 5.

After performing the colour filtering, the result shown in Fig. 6 a) is obtained. As can be appreciated, the filter is not perfect, as there are certain areas of dirt and/or rust on the surface of the clips and rail that have a different hue than the computed average. Hence, to correct these errors, a circular morphological closing filter was used for the two categories. This filter fills, within a given radius, the empty pixels in the segmentation that are surrounded by pixels of a given category. In this way, the filled segmentation represented in Fig. 6 b) is obtained. Specifically, radii of 7 pixels were used for the clips and 12 pixels for the rail.

Once the pixels of the rail and the clips have been segmented, the sleepers are detected in the image. Firstly, an entropy filter [39] is applied to the image and the coordinates where the entropy is equal to zero are obtained (Fig. 7 a)). The entropy is related to the texture of the image, and as can be seen, the sleeper areas have a lower entropy than the ballast. However, this does not allow the sleepers to be segmented correctly as there are many spurious pixels. To obtain the position of the sleepers, as they are perpendicular to the rail, the mean entropy along the rail direction was evaluated.

To do this, the direction of the rail is calculated through a linear regression using the pixels of the colour segmentation (Fig. 7 b)). Then, a sliding rectangle of dimensions 1500×200 pixels is moved along the direction of the rail, and the number of zero-entropy pixels along the rail is evaluated (Fig. 7 c)). Several local maxima are obtained that represent the position of the sleepers. Finally, for each of these local maxima, two bounding boxes of dimensions 240×280 pixels and distanced 140 pixels from the centre of the rail are calculated. If these boxes do not go outside the limits of the image, a crop of the region is extracted for further analysis (Fig. 7 d)).

These cropped images such as Fig. 8 a) represent the areas in which the bolts and clips should be located. To detect the position of the bolts, a circular Hough transform [40] was used on the grayscale cropped images (Fig. 8 b)). This algorithm first computes the colour gradients in the images to search for edges, and then calculates the correlation of these edges with circumferences in an established range of radii. If a strong correlation is detected, above a threshold value, the position and centre of the circumference is estimated. Otherwise, no circumference is classified within the image. In this work, after the experimental adjustment of the parameters of the Hough transform, search radii between 30 and 50 pixels were used, as well as a threshold value of 0.85. Finally, the pixels inside the circumference are extracted, and this segmentation is combined with the colour-based segmentation of the clips to obtain the final segmentation such as the depicted in Fig. 8 c).

3.1.2. DeepLab V3 + network

A DeepLab V3 + neural network [34] was trained to segment the images and obtain cropped labelled images DeepLab V3 + is a pixel-wise segmentation neural network that associates each pixel of the image to

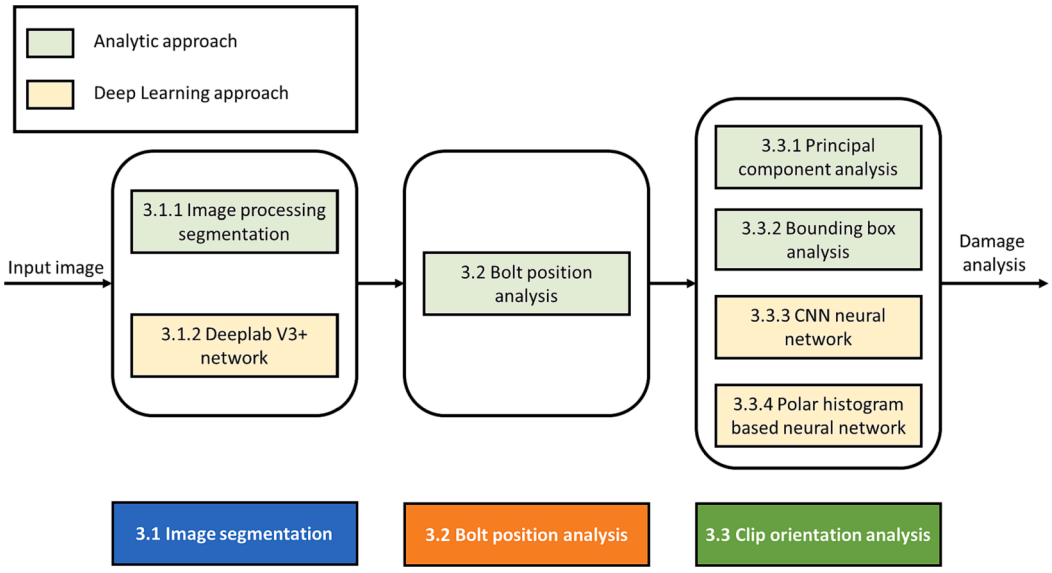


Fig. 3. Damage inspection process.

Fig. 4. Image of the railway track.

one of the different categories with which the network was trained. In this work, as depicted in Fig. 9, five classes were defined: sleepers, rail, bolts, clips and the background. This image represents one of the segmentations used for the training of the network. In this work, all the 128 images were manually labelled as the attached one.

The network was trained using the labelled images, splitting 100 images for training and the rest for validation. After performing the training, an accuracy of 97.14% was obtained for the validation dataset. Once the different elements are identified a similar procedure as the described Section 3.1.1 was followed. With the neural network a much cleaner segmentation of the sleepers is obtained, and a much simpler approach can be used to compute the intersection of the rail and sleepers. A DBSCAN clustering algorithm [41] was implemented to separate the points labelled as sleepers, and a linear regression was computed for all the clusters. Finally, the intersections of the lines of the rail and sleepers were computed as shown in Fig. 10 and the segmented cropped images are extracted. In this case, the bolts and clips are already labelled directly by the network and no further processing of the labels is needed.

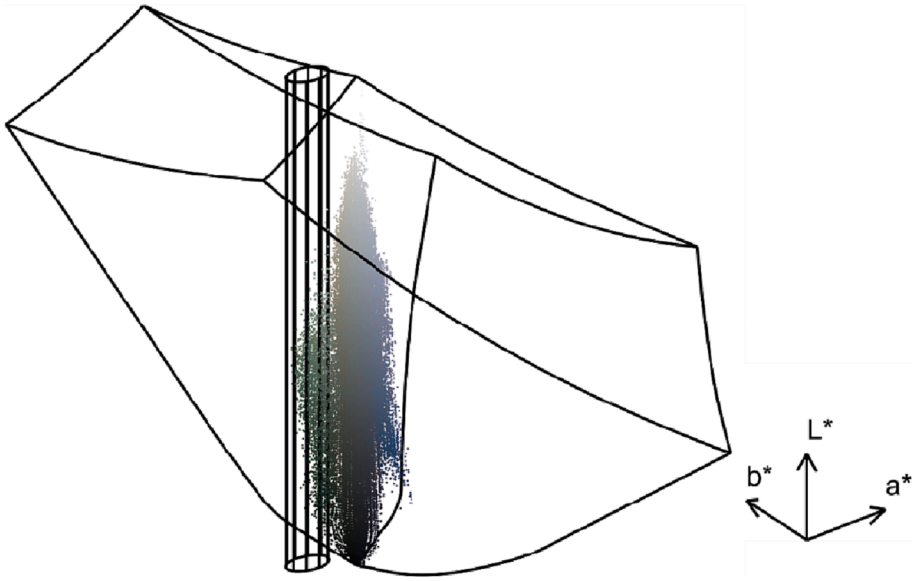


Fig. 5. CIE Lab Colour Space.

Fig. 6. Clip and rail segmentation. a) Colour segmentation. b) Morphologically closed segmentation.

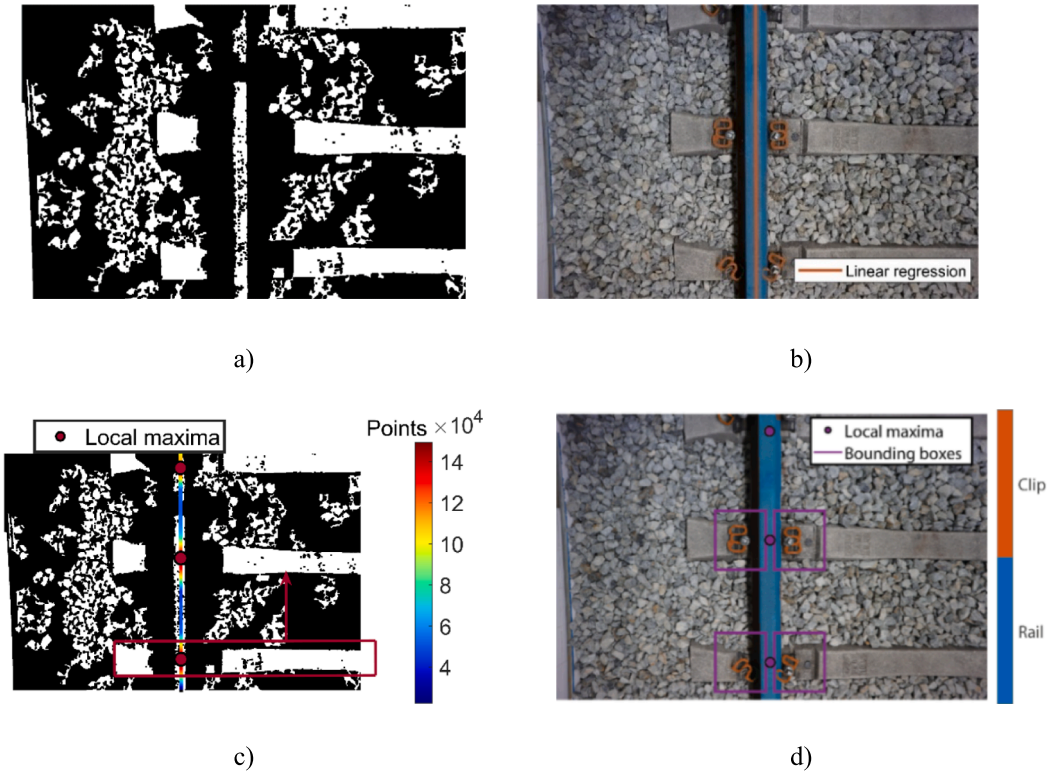


Fig. 7. Clip position detection. a) Entropy filter. b) Linear regression. c) Detection of areas with high concentration of low entropy points. d) Final position extraction.

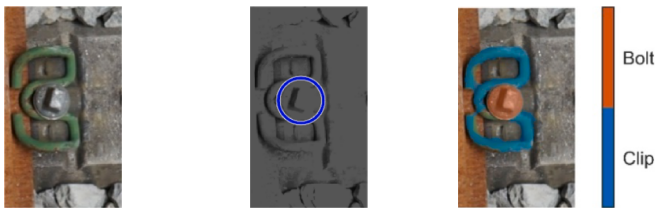


Fig. 8. Final segmentation. a) Initial image. b) Bolt detection. c) Segmented clip and bolt.

3.2. Bolt position analysis

Fig. 11 presents the methodology followed to determine whether the bolts and clips are correctly placed. Firstly, the images are analysed to check if both components are present and if so, compute their relative orientation. The number of pixels of each category in the labelled image are counted and if it is lower than a specified threshold of 100 pixels, a message is shown warning that one of the components is missing.

If both elements, clip and bolt are present, their relative position is computed using the coordinates of their centres. The pixels of the clips are enclosed in a horizontal rectangle, as shown in Fig. 12 and the mean point is taken as the centre of the clip. Regarding the bolts, as it is a circular geometry, the average position of the coordinates is taken as the centre.

To obtain an estimation of the distance among both centres in length

Fig. 9. Manual segmentation.

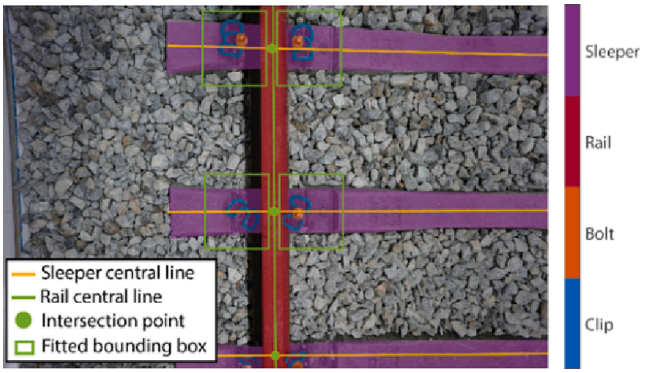


Fig. 10. Final segmentation.

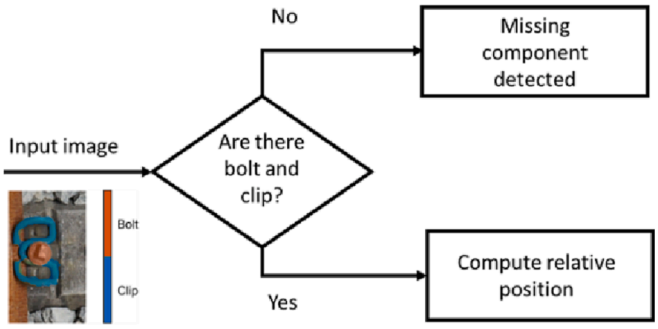


Fig. 11. Workflow diagram.

units, a PinHole Camera Model [42] was used to project the centre points in the camera reference frame. The camera is placed perpendicularly to the ground at height of 1.5 m. Both the clip and bolt are placed above the sleepers, so their distance with respect to the sensor plane can be approximated with this value of 1.5 m. Hence, with this simplification, the distance among both centres in length units can be estimated using Equation (1):

$$d = \frac{\Delta u w z}{f pix} \tag{1}$$

Where:

- Δu is the distance between the clip and bolt centres measured in pixels.
- w is the sensor width.
- z is the perpendicular distance to the ballast plane.
- f is the focal length of the camera.
- pix is the resolution of the sensor.

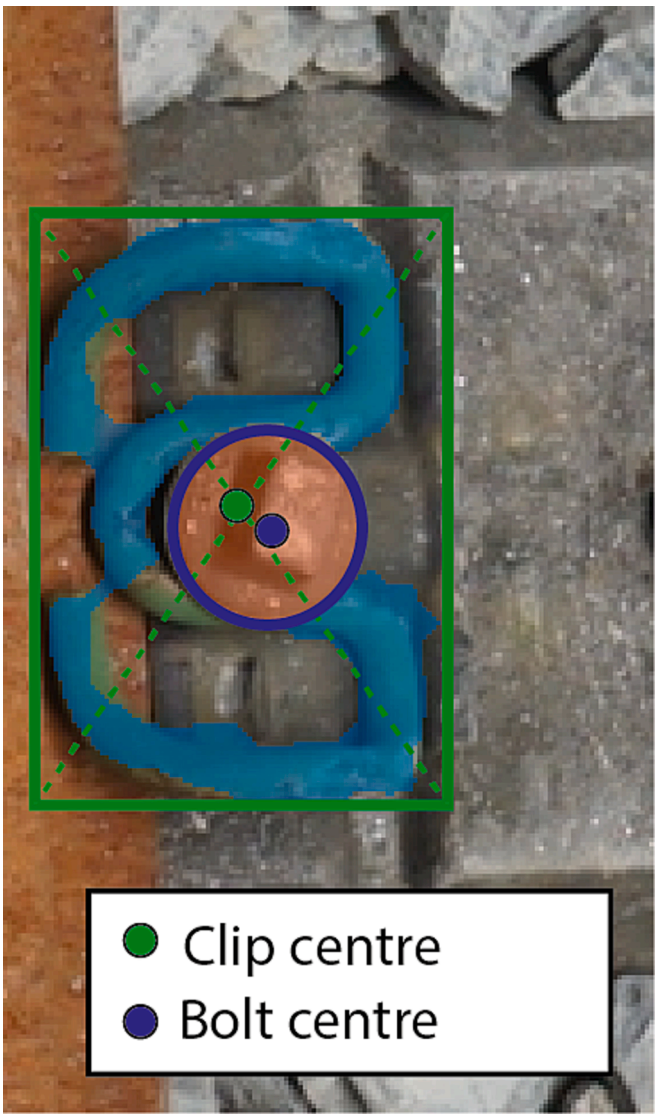


Fig. 12. Distance calculation.

3.3. Clip orientation analysis

3.3.1. Principal component analysis

The first method used for evaluating the orientation of the clip is the Principal Component Analysis. The clips have an elongated shape in the direction of the rail. Therefore, in this direction the dispersion and variance are maximum. To obtain this principal component, the components of the inertia tensor are computed using Equation (2):

$$I_{ij} = \vec{x}_i . \vec{x}_j \tag{2}$$

Where:

- I_{ij} is the component i, j of the inertia tensor.
- $|.|$ denotes the dot product.
- $\vec{x}_i$, $\vec{x}_j$ are the vectors of the components i, j of all the pixels of the clip.

Once the inertia tensor is computed, its eigenvalues and eigenvectors are calculated to obtain the principal directions of dispersion of the points of the clip (Fig. 13). This is an absolute orientation within the image, thus, to obtain the relative orientation with respect to the rail, the angle among them is computed using the direction of the rail computed with the linear regression (see section 3.1.1). This procedure will be

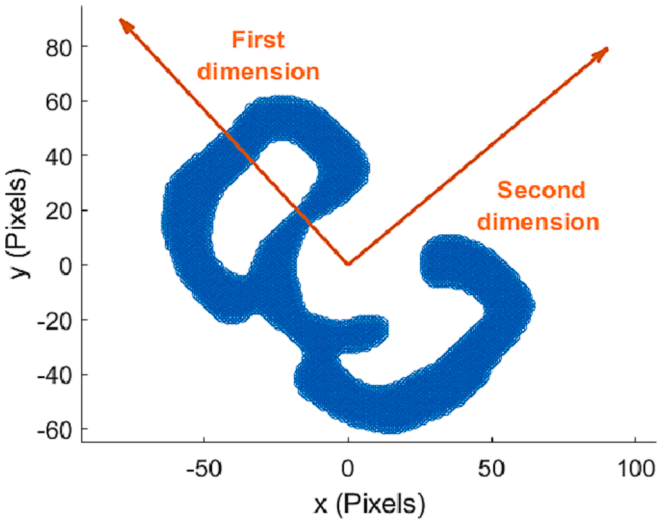


Fig. 13. Principal Component Analysis.

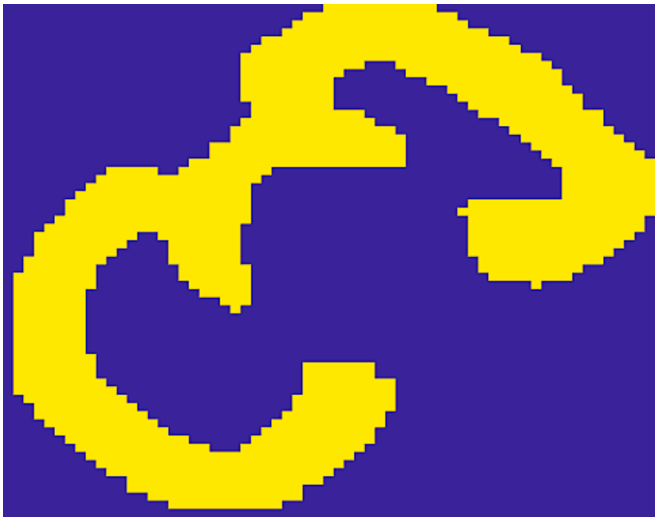


Fig. 15. Example of binary segmentation image.

followed in all of the following methods described below, as they firstly compute the absolute orientation as well.

3.3.2. Bounding box analysis

Another useful property for studying the absolute orientation of the clip is the aspect ratio of the sides of a horizontal bounding box. As can be seen in Fig. 14, when the clip is inclined, the enclosing box tends to be more squared, while when it is horizontal, the difference between both sides is maximum. To determine the orientation of the clips, the coordinates of the clip points are rotated synthetically and the relationship between sides R and L is evaluated. When this ratio is maximum, the clip will be aligned with the horizontal axis (Fig. 14) and this rotation angle will be equal to the orientation of the clip. To determine this angle a quasi-Newton BFGS algorithm [43] was employed.

3.3.3. CNN neural network

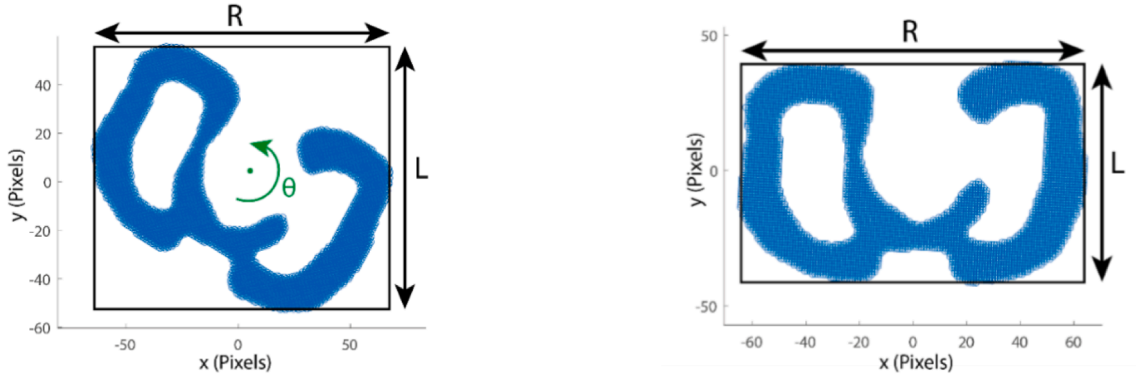
The third method is a Convolutional Neural Network (CNN) that processes low-resolution binary images of the segmentation like the one represented in Fig. 15. These images were obtained using the segmentations presented in Section 3.1. In order to reduce the dimensionality and obtain computation times of the same order of the other methods, a rescaling factor was applied to obtain 64x64 pixel images.

CNNs are tools capable of extracting high-levels abstractions from images by a combination of Convolutional, Max Pooling and Fully Connected layers. With this combination of layers, they can extract characteristic features from images such as angles, textures or shapes. This type of network is commonly used in image segmentation and recognition applications, as well as for detecting certain characteristics on images. Zhou et al. [44] carried out a survey analyzing the use of CNN for estimating the orientation of different objects in images. They employed different datasets (including grayscale and RGB images) and established performance metrics for three architectures of CNN.

In this work, instead of using grayscale or RGB images, a binary segmentation is provided as the input of the network. The output is the rotation angle of the clips, which was determined manually as it will explained in Section 4.1. Fig. 16 presents the architecture used in this work, in which Batch Normalization layers were included to improve the convergence of the training method. Regarding the training of the network, two datasets were generated using the segmentations generated by DeepLab V3 + and the colour segmentation. To increase the size of the datasets, five small rotations were added to each one of the exported clip labels, to finally obtain two databases of 2880 images, of which, 80% were used for training and the rest for validation.

3.3.4. Polar histogram based neural network

The CNN presented in Section 3.3.3 is a generic Deep Learning method, which can be used to solve much more complex problems. For



a) b)

Fig. 14. Bounding box angle analysis. a) Rotated clip. b) Best fitting angle.

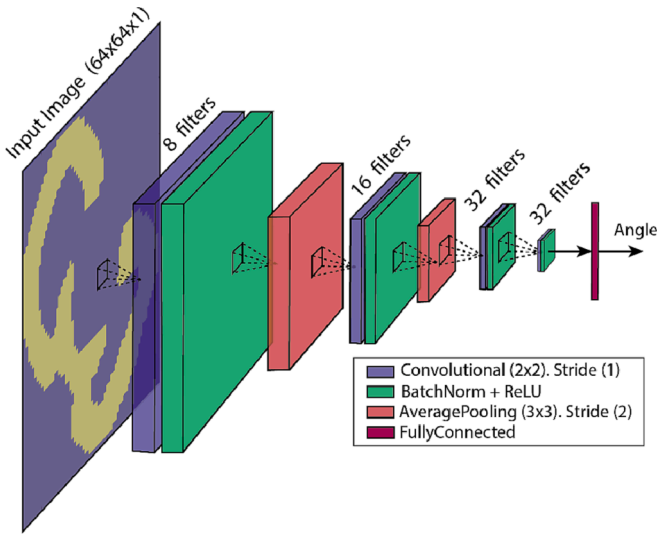


Fig. 16. Scheme of the convolutional neural network.

instance, Zhou et al. [44], used CNNs to estimate the orientation of hand-written digits from the MNIST dataset [45]. Those images are not segmented, and they include numbers with different geometries.

In this work, segmented SKL type clip images are used. Ideally, without considering possible distortion in the photographs or segmentation errors, they have the same shape in binary images. This allows us to establish relationships between their rotation angle and the spatial distribution of pixels. Therefore, an alternative method is proposed consisting of a fully connected neural network that processes the spatial distribution of pixels of the clips to determine the rotation angle. This is a much simpler method since the dimensions of the input data are considerably smaller. Firstly, the coordinates of the clips are transformed into polar coordinates with origin in the centre of the clip. Then, a histogram of 10x10 cells is computed as represented in Fig. 17. The histogram represents the point concentration for different radials and angles. The domain is divided azimuthally and radially from the origin to the outermost point of the clip.

These 10x10 matrices are used as the input of the neural network represented in Fig. 18. The network is composed by three layers of fully connected perceptrons followed by a batch normalization as well as a ReLU activation function. As in the previous section, small rotations were added to synthetically augment the dataset. The network was

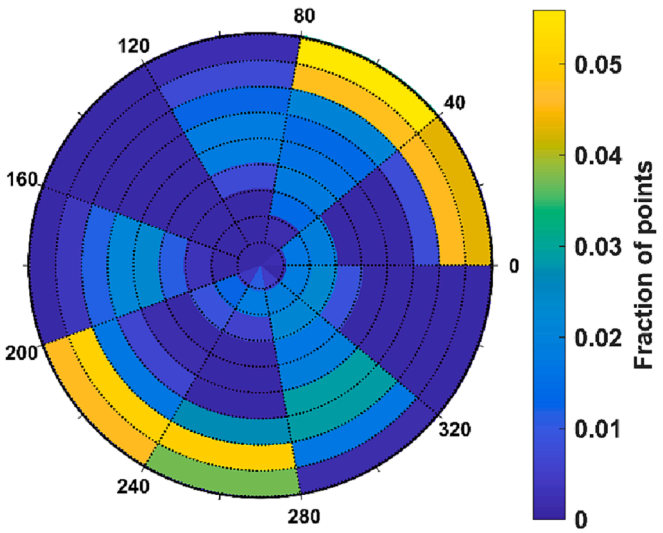


Fig. 17. Histogram of point density in polar coordinates.

trained using the augmented datasets described in Section 3.3.3 with the same ratios for training and validation.

3.4. Evaluation procedures

In order to define error metrics that quantify the accuracy of the results, it is first necessary to establish ground-truth values that allow for the calculation of the errors for the different proposed methods.

As introduced in Section 3.1.2, for the training of the neural networks, the dataset of track images was manually labelled pixel-wise. This manual labelling will be considered as the ground-truth, and the errors of the segmentation methods will be computed with respect to it. To quantify the segmentation errors for each category, the Intersection over Union (*IoU*) metric will be used. As represented in Equation (3), for each category k, the associated Intersection over Union (IoU_k) is equal to the ratio of True Positives (TP_k) divided by the union set of True Positives (TP_k), False Positives (FP_k) and False Negatives (FN_k) of the category. It is important to remark that for the computation of this metric only the images of the validation dataset are used.

$$IoU_k = \frac{TP_k}{TP_k + FP_k + FN_k} \tag{3}$$

Finally, the mean Intersection over Union (*mIoU*) of all categories is calculated using Equation (4):

$$mIoU = \frac{1}{N}\sum_{k=1}^{N} IoU_k \tag{4}$$

In order to quantify the errors of the position calculation methods for clips and bolts, all images were manually processed as shown in Fig. 19. To determine the separation distance between the clips and bolts, bounding rectangles were manually drawn for the clips and circles for the bolts (Fig. 19 a)). The centres of these geometries were taken as the geometric centre of the two elements, and the distance between them was calculated following the procedures described in Section 3.2. Regarding the orientation of the clips, as depicted in Fig. 19 b) a line was manually drawn on the images and the relative orientation with respect to the rail was computed processing the annotated pictures.

The authors opted for this approach instead of using a binary classification of well-positioned and loose clips because the latter is a criterion that is subject to some subjectivity if the rotation angle of the clips is small. Additionally, with binary classification, the results depend heavily on the image dataset used. If images of clips with small rotations relative to their optimal operating position are introduced, it is possible that detection methods will classify these clips as being in good condition. Conversely, if all the loose clips always present large rotation angles, this facilitates the classification of these elements, improving the accuracy of the results. For these reasons, this study opted for the measurement of the relative position and orientation of the clips instead of a binary classification.

4. Results and discussion

4.1. Accuracy

Table 1 displays the Intersection over Union (*IoU*) obtained on the validation dataset for different categories using the colour based segmentation method and the DeepLab v3 + neural network. As it can be observed, the neural network exhibits better performance overall. The predictions of the neural network for rail, background, and sleeper categories almost always coincide with the manually labelled values. As shown in Table 1, the lowest *IoU* values for the neural network occur in clips and bolts. This is because these are small and narrow parts, and any difference in labelling at the edges of these elements compared to the manual segmentation significantly decreases *IoU*.

As described in Section 3.1.1, for the colour segmentation, a pixel-

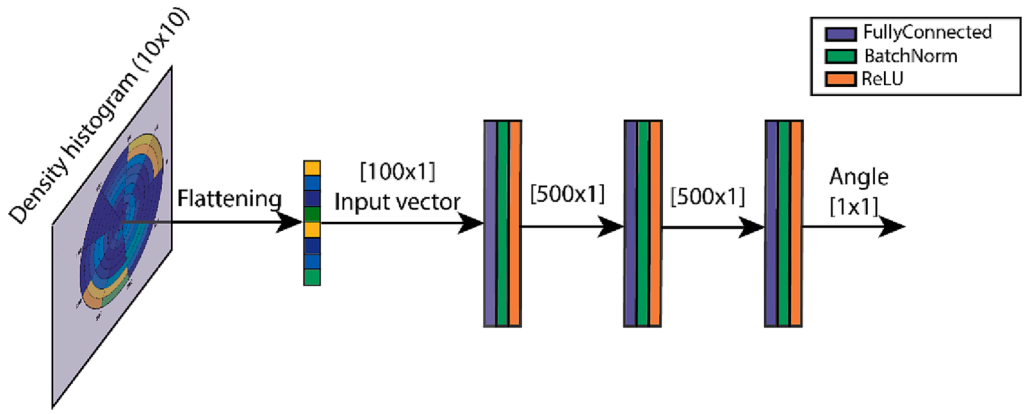


Fig. 18. Fully connected neural network.

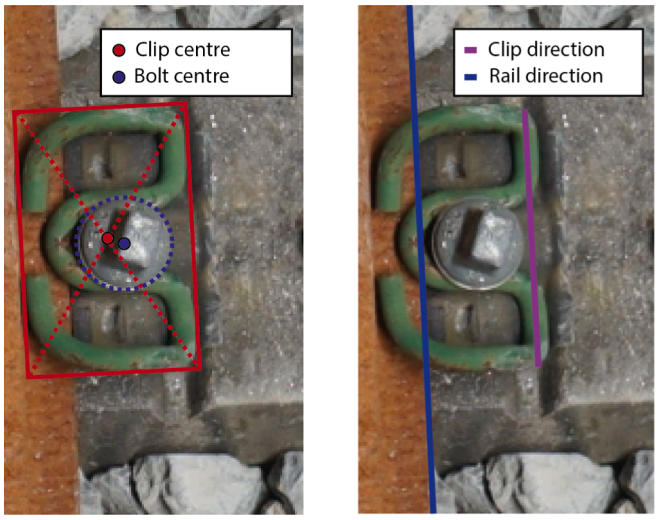


Fig. 19. Ground truth. a) Bolt and clip centre determination. b) Clip direction determination.

Table 1
Intersection over Union (***IoU***) results.

Category	Method Colour based segmentation	DeepLab V3+
Background	–	0.9845
Rail	0.7156	0.9574
Sleepers	–	0.9623
Clips	0.7193	0.8696
Bolts	0.8471	0.9124
mIoU	–	0.9372

wise segmentation of sleepers was not performed, as it is not necessary for the study of the position of clips and bolts. As a result, the *IoU* for this category and the image background could not be obtained. The *IoU* values are significantly lower for clips and rails. This is due to the shadows that are formed on the left side of the track (Fig. 6). Fig. 20 shows a comparison between the segmentation performed by the DeepLab V3 + neural network and colour segmentation. The neural network is able to identify the clips in the shadow areas, while colour segmentation struggles to detect any pixels in that area. Despite using the CIE-lab colour space, which is more resilient to changes in lighting, in these areas the camera cannot capture the correct colour gamut of the

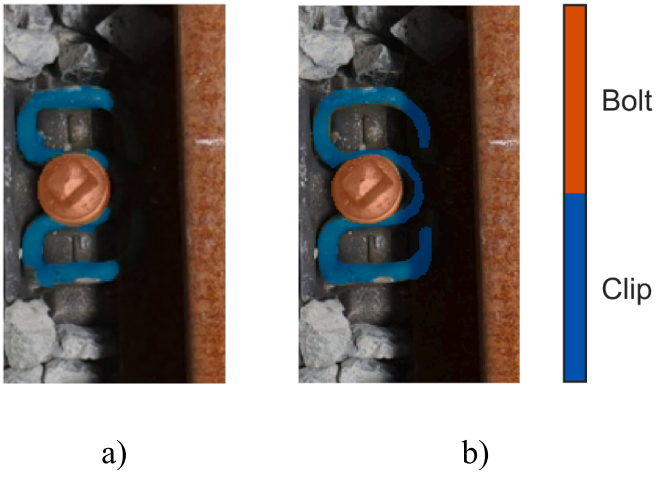


Fig. 20. Faulty segmentation in the shadow. a) Colour-based segmentation. b) DeepLab V3 +.

clips, which increases the *IoU* of colour segmentation.

For the calculation of the relative position between bolts and clips, a method based on geometric relationships was proposed. Therefore, to calculate the errors, all the images used in this work were employed (576 for each type of segmentation). Regarding the clip orientation calculation methods, it is important to note that some of them use Deep Learning techniques. In order to correctly evaluate these methods, only the images from the validation set were used to calculate the errors (20% of the 2880 images obtained through synthetic rotations). Finally, the results obtained in this work are presented in Table 2.

In general terms, for all the proposed methods, it can be seen that the accuracy is higher when using the segmentation provided by DeepLab V3 +. The segmentation of the neural network facilitates the calculation of the relative position and orientation of clips, with average errors lower than 2 degrees for the histogram-based neural network. The accuracy of the methods when using colour segmentation was lower due to segmentation errors caused by error sources such as shadow areas. The methods most affected by these segmentation errors were Principal Component Analysis and Bounding Box, since both depend on geometric properties calculated from the pixels of the segmentation.

On the other hand, neural networks were trained with segmentations that included these errors and obtain a more accurate estimation of the orientation of the clips, especially for the network based on a histogram in polar coordinates. The Convolutional Neural Network also performs well, but as it is a generic method for pattern detection, its errors are slightly higher than those of the histogram-based network.

Table 2
Accuracy results.

Process	Method	Used segmentation	Mean Error	Standard Deviation
3.2 Position estimation	3.2 Centre calculation	3.1.1 Colour based segmentation	10.1 mm	6.3 mm
		3.1.2 DeepLab V3+	5.0 mm	2.2 mm
3.3 Orientation determination	3.3.1 Principal Component Analysis	3.1.1 Colour based segmentation	6.89 deg	4.02 deg
		3.1.2 DeepLab V3+	3.15 deg	3.14 deg
	3.3.2 Bounding box	3.1.1 Colour based segmentation	5.94 deg	3.94 deg
		3.1.2 DeepLab V3+	3.92 deg	2.56 deg
	3.3.3 Convolutional Neural Network	3.1.1 Colour based segmentation	3.41 deg	1.85 deg
		3.1.2 DeepLab V3+	2.42 deg	1.49 deg
	3.3.4 Histogram-based Neural Network	3.1.1 Colour based segmentation	2.53 deg	1.69 deg
		3.1.2 DeepLab V3+	1.96 deg	1.35 deg

4.2. Computation time

Table 3 presents the average computation time for all the different processes. For the calculation, a laptop with an Intel Core i7 10750H processor along with a Nvidia GTX 1650 Ti was used. As indicated in the aforementioned table, the neural networks were executed using GPU acceleration, whereas the image processing algorithms used the CPU.

As can be appreciated the most computationally demanding process is the image segmentation, as images with a quite high resolution ($2048 \times 1362 pix$) were used, that considerably increases the computation time. On the other hand, the positioning and orientation algorithms work with the image crops of the clip and bolt areas, whose resolution is considerably lower ($240 \times 280 pix$). The slowest algorithm was the traditional colour segmentation, mainly due to the multiple sequential operations involved, as described in Section 3.1.1. The neural network DeepLab V3+, thanks to the GPU acceleration manages to segment directly all the elements of the image in less than one second. Regarding the calculation of the position of bolts and clips, as well as the orientation of the latter, all the proposed methods are computationally inexpensive. The most demanding one was 3.3.2 Bounding Box, as it employs an iterative method quasi-Newton BFGS, that carries out several iterations until finding the local optimum with a given tolerance (A tolerance of 10^{-6} was considered in this work). This parameter could be increased to reduce the number of necessary iterations as well as the computation time. Nevertheless, given that the calculation time is less than a tenth of a second, it is not very relevant. The methods based on neural networks were also inexpensive, as the size of the input data as well as the number of layers and operations were smaller than for DeepLab V3 +.

5. Conclusions

This article presents a comparison of methods for autonomous inspection of the state of clips and bolts in railway tracks. Different approaches were proposed, both for the segmentation of the images and for the calculation of the position and orientation of the clips. An autonomous data acquisition system was developed to obtain the images used in this study and it was employed in a test track installed in the facilities of the University of Vigo. The proposed algorithms were tested, measuring their performance in terms of accuracy and computation time. The following results were achieved:

- Image segmentation was the most computationally demanding process due to the high resolution of the input images. However, reasonable computation times were achieved for the two proposed alternatives.
- DeepLab V3 + presented a better segmentation performance in general terms than the colour-based segmentation, being especially robust in shaded areas.
- Accuracies of the order of a few degrees of error were obtained for all proposed orientation computation algorithms. In the case of the calculation of relative position between bolts and clips, the error was of the order of millimetres. It was also possible to appreciate the effect on the accuracy of these algorithms of the errors in the segmentation. The methods based on neural networks were more resilient as they were trained using images with segmentation errors.
- For the case of DeepLab V3+, the processing time of a complete image and the verification of all the bolts and clips is below one second for all the proposed methods. This would enable to inspect several kilometres of track in around one hour. In this study, an inexpensive laptop GPU was used (Nvidia GTX 1650 Ti). For a high-end desktop card, the performance would be much higher.

In future works, the payload of the current acquisition system will be installed in a motorized railway inspection vehicle to achieve higher inspection speeds. The performance of the developed algorithms will be assessed and a parametric study of the influence of the motion blur on the results will be carried out. In that sense, calculations will be performed to determine the optimal camera configuration for different lightning conditions. The neural networks will be retrained using images obtained outdoors and the robustness of the algorithms will be studied in these new conditions, in which there may be reflections and shadows generated by sunlight. Furthermore, new neural network architectures will be explored, and the resolution of the input images will be reduced as an attempt to improve the calculation times. A trade-off will be carried out between accuracy and computation time for different configurations.

Table 3
Computation time.

Process	Method	Computation time (s)	Execution environment
3.1 Segmentation	3.1.1 Image processing	3.4107	CPU
	3.1.2 DeepLab V3+	0.6339	GPU
3.2 Position estimation	3.2 Centre calculation	0.0024	CPU
3.3 Orientation determination	3.3.1 Principal Component Analysis	0.0017	CPU
	3.3.2 Bounding box	0.0864	CPU
	3.3.3 Convolutional Neural Network	0.0415	GPU
	3.3.4 Histogram-based Neural Network	0.0043	GPU

Funding

E.A. is funded by the Spanish Ministry of Universities (grant ref: FPU21/01176).

L.M.G-d is funded by the Recovery, Transformation and Resilience Plan of the European Union – NextGenerationEU (University of Vigo grant ref. 585507).

Grant PLEC2021-007940 funded by MCIN/AEI/ https://doi.org/10.13039/501100011033 and by the European Union NextGenerationEU/ PRTR.

7. Institutional Review Board Statement

Not Applicable.

8. Informed Consent Statement

Not Applicable.

9. Data Availability Statement

The data used in this study are not publicly available due to the conditions of the ethics approval for the study. Contact the corresponding author for further information.

CRediT authorship contribution statement

E. Aldao: Conceptualization, Investigation, Methodology, Software, Writing – original draft. **L. Fernández-Pardo:** Conceptualization, Investigation, Methodology, Software. **L.M. González-deSantos:** Conceptualization, Methodology, Supervision, Writing – review & editing. **H. González-Jorge:** Conceptualization, Methodology, Supervision, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

Acknowledgments

COPASA participated in the development of this work by providing support and resources to obtain the experimental data. They also collaborated in the definition of inspection system requirements and their experience was essential for the development of the proposed algorithms.

References

[1] M. Sedghi, O. Kauppila, B. Bergquist, E. Vanhatalo, M. Kulahci, A taxonomy of railway track maintenance planning and scheduling: A review and research trends, Reliab. Eng. Syst. Saf. 215 (2021), 107827, https://doi.org/10.1016/J.RESS.2021.107827.

[2] European Commission, Seventh monitoring report on the development of the rail market under Article 15, 2019, (n.d.). https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A52021DC0005 (accessed March 8, 2023).

[3] A. Consilvio, A. di Febbraro, R. Meo, N. Sacco, Risk-based optimal scheduling of maintenance activities in a railway network, EURO J. Transp. Logist. 8 (2019), https://doi.org/10.1007/s13676-018-0117-z.

[4] K. Popov, R. De Bold, H.K. Chai, M.C. Forde, C.L. Ho, J.P. Hyslip, H.F. Kashani, P. Long, S.S. Hsu, Big-data driven assessment of railway track and maintenance efficiency using Artificial Neural Networks, Constr. Build. Mater. 349 (2022), https://doi.org/10.1016/J.CONBUILDMAT.2022.128786.

[5] Q. He, H. Sun, M. Dobhal, C. Li, R. Mohammadi, Railway tie deterioration interval estimation with Bayesian deep learning and data-driven maintenance strategy, Constr. Build. Mater. 342 (2022), https://doi.org/10.1016/J.CONBUILDMAT.2022.128040.

[6] R. Macedo, R. Benmansour, A. Artiba, N. Mladenović, D. Urošević, Scheduling preventive railway maintenance activities with resource constraints, Electron Notes Discrete Math. 58 (2017), https://doi.org/10.1016/j.endm.2017.03.028.

[7] D. Gattuso, A. Restuccia, ScienceDirect A tool for railway transport cost evaluation, Procedia Soc. Behav. Sci. 111 (2014) 549–558, https://doi.org/10.1016/j.sbspro.2014.01.088.

[8] Railway trolley for rail inspection and diagnosis, (n.d.). https://www.vaiacar.com/p/en/rail-inspection-trolley.html (accessed November 17, 2022).

[9] Manned & Autonomous Track Inspection Vehicles | Locomotives, Freight and Passenger Cars | ENSCO Rail, (n.d.). https://www.ensco.com/rail/manned-and-autonomous-vehicles (accessed November 17, 2022).

[10] G. Jing, X. Qin, H. Wang, C. Deng, Developments, challenges, and perspectives of railway inspection robots, Autom. Constr. 138 (2022), 104242, https://doi.org/10.1016/J.AUTCON.2022.104242.

[11] R. Montero, J.G. Victores, S. Martínez, A. Jardón, C. Balaguer, Past, present and future of robotic tunnel inspection, Autom. Constr. 59 (2015) 99–112, https://doi.org/10.1016/J.AUTCON.2015.02.003.

[12] DIGAV (Dispositivo de Inspección Gráfica de Aparatos y Vía) | Ineco, (n.d.). https://www.ineco.com/webineco/digav-dispositivo-inspecci%C3%B3n-gr%C3%A1fica-aparatos (accessed March 8, 2023).

[13] Ferrocarriles - COPASA, (n.d.). https://www.copasagroup.com/es/ferrocarriles/ (accessed November 18, 2022).

[14] ADIF, Technical Specification ET 03.360.566.8 (Clips), 2020. http://descargas.adif.es/ade/u18/GCN/NormativaTecnica.nsf/v0/5A66AAA7D5DFFA44C12585AE0058FC67/$FILE/ET%2003.360.566.8+M1%20Clips.pdf?OpenElement (accessed November 18, 2022).

[15] ADIF, NAV 7-3-8.2 Inspección de aparatos de vía, (n.d). http://descargas.adif.es/ade/u18/GCN/NormativaTecnica.nsf/v0/039993C49C56FEBAC12587DF0034EB78/$FILE/NAV7382_ED1.pdf?OpenElement (accessed March 8, 2023).

[16] D. Ferreño, J.A. Casado, I.A. Carrascal, S. Diego, E. Ruiz, M. Saiz, J.A. Sainz-Aja, A. I. Cimentada, Experimental and finite element fatigue assessment of the spring clip of the SKL-1 railway fastening system, Eng. Struct. 188 (2019) 553–563, https://doi.org/10.1016/J.ENGSTRUCT.2019.03.053.

[17] W. Chen, Z. Wu, Z. Zhu, W. Chen, W. Wang, Z. Zeng, Pull out and pre-tightening force tests for plastic dowel of the railway sleeper considering the influence of installing torque and frost force, Constr. Build. Mater. 267 (2021), https://doi.org/10.1016/J.CONBUILDMAT.2020.120948.

[18] A. Colaço, A. Castanheira-Pinto, P. Alves Costa, J. Fernández Ruiz, Combination of experimental measurements and numerical modelling for prediction of ground-borne vibrations induced by railway traffic, Constr. Build. Mater. 343 (2022), https://doi.org/10.1016/J.CONBUILDMAT.2022.127928.

[19] H. Cui, J. Li, Q. Hu, Q. Mao, Real-Time Inspection System for Ballast Railway Fasteners Based on Point Cloud Deep Learning, IEEE Access 8 (2020) 61604–61614, https://doi.org/10.1109/ACCESS.2019.2961686.

[20] X. Wei, Z. Yang, Y. Liu, D. Wei, L. Jia, Y. Li, Railway track fastener defect detection based on image processing and deep learning techniques: A comparative study, Eng. Appl. Artif. Intel. 80 (2019), https://doi.org/10.1016/j.engappai.2019.01.008.

[21] P. Babenko, Visual inspection of railroad tracks, (2010), (n.d.). http://crcv-web.eecs.ucf.edu/papers/theses/Babenko_Pavel.pdf (accessed March 8, 2023).

[22] P. Chandran, J. Asber, F. Thiery, J. Odelius, M. Rantatalo, An investigation of railway fastener detection using image processing and augmented deep learning, Sustainability (Switzerland) 13 (2021), https://doi.org/10.3390/su132112051.

[23] S. Liu, Q. Wang, Y. Luo, A review of applications of visual inspection technology based on image processing in the railway industry, Transp. Saf. Environ. 1 (2019) 185–204, https://doi.org/10.1093/TSE/TDZ007.

[24] X. Dai, E. Yang, S. Ding, C. Wang, Y. Qiu, Automatic Defect Inspection Algorithm of Railway Fasteners Based on 3D Images, Tiedao Xuebao/J. China Railway Soc. 39 (2017), https://doi.org/10.3969/j.issn.1001-8360.2017.10.012.

[25] C. Taştimur, M. Karaköse, E. Akın, A Vision Based Condition Monitoring Approach for Rail Switch and Level Crossing using Hierarchical SVM in Railways, Int. J. Appl. Math. Electron. Comput. (2016). https://doi.org/10.18100/ijamec.270634.

[26] X. Gibert, V.M. Patel, R. Chellappa, Deep Multitask Learning for Railway Track Inspection, IEEE Trans. Intell. Transp. Syst. 18 (2017) 153–164, https://doi.org/10.1109/TITS.2016.2568758.

[27] Y. Wu, Y. Qin, Y. Qian, F. Guo, Automatic detection of arbitrarily oriented fastener defect in high-speed railway, Autom. Constr. 131 (2021), https://doi.org/10.1016/J.AUTCON.2021.103913.

[28] Y. Zhan, X. Dai, E. Yang, K.C.P. Wang, Convolutional neural network for detecting railway fastener defects using a developed 3D laser system, Int. J. Rail Transp. 9 (2021), https://doi.org/10.1080/23248378.2020.1825128.

[29] Y.W. Lin, C.C. Hsieh, W.H. Huang, S.L. Hsieh, W.H. Hung, Railway Track Fasteners Fault Detection using Deep Learning, in: 2019 IEEE Eurasia Conference on IOT, Communication and Engineering, ECICE 2019, 2019. https://doi.org/10.1109/ECICE47484.2019.8942769.

[30] C.-C. Hsieh, T.-Y. Hsu, W.-H. Huang, An Online Rail Track Fastener Classification System Based on YOLO Models, Sensors 22 (2022) 9970, https://doi.org/10.3390/s22249970.

[31] X. Wei, Z. Yang, Y. Liu, D. Wei, L. Jia, Y. Li, Railway track fastener defect detection based on image processing and deep learning techniques: A comparative study, Eng. Appl. Artif. Intel. 80 (2019) 66–81, https://doi.org/10.1016/J.ENGAPPAI.2019.01.008.

[32] A. Ji, W.L. Woo, E.W.L. Wong, Y.T. Quek, Rail track condition monitoring: a review on deep learning approaches, Intellig. Robot. 1 (2021) 151–175. https://doi.org/10.20517/IR.2021.14.

[33] I. Aydin, M. Sevi, M. Salur, E. Akin, Defect classification of railway fasteners using image preprocessing and a lightweight convolutional neural network, Turk. J. Electr. Eng. Comput. Sci. (2021), https://doi.org/10.3906/elk-2106-42.

[34] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation, (n.d.). https://github.com/tensorflow/models/tree/master/ (accessed November 7, 2022).

[35] W. Yi, S. Marshall, Principal component analysis in application to object orientation, Http://Www.Tandfonline.Com/Action/JournalInformation?Show=aimsScope&journalCode=tgsi20#.VsXpLiCLRhE. 3 (2012) 76–78. https://doi.org/10.1007/BF02826615.

[36] A.I. Shahin, S. Almotairi, DCRN: An Optimized Deep Convolutional Regression Network for Building Orientation Angle Estimation in High-Resolution Satellite

Images, Electronics 2021, Vol. 10, Page 2970. 10 (2021) 2970. https://doi.org/10.3390/ELECTRONICS10232970.
[37] K. Itakura, F. Hosoi, Estimation of Leaf Inclination Angle in Three-Dimensional Plant Images Obtained from Lidar, Remote Sensing 2019, Vol. 11, Page 344. 11 (2019) 344. https://doi.org/10.3390/RS11030344.
[38] H.L.C. Scale, Insight on Color, Hunter Labs, Reston, 1996, (n.d.). https://support.hunterlab.com/hc/en-us/article_attachments/201440625/an08_96a2.pdf (accessed March 8, 2023).
[39] R.C. Gonzalez, R.E. Woods, S.L. Eddins, Digital image processing using MATLAB, (n.d.) 1009. https://es.mathworks.com/academia/books/digital-image-processing-using-matlab-gonzalez.html (accessed December 1, 2022).
[40] P.V.C. Hough, A Method and Means for Recognizing Complex Pattern, 1962, (n.d.). https://people.csail.mit.edu/bkph/courses/EE693B/Patents/US003069654_Hough.pdf (accessed March 8, 2023).
[41] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise, (1996). www.aaai.org (accessed December 1, 2022).
[42] Multiple View Geometry in Computer Vision - Richard Hartley, Andrew Zisserman - Google Books, (n.d.). https://books.google.es/books?id=si3R3Pfa98QC&pg=PA153&dq=pinhole+intitle:%22Multiple+View+Geometry+in+computer+vision%22&redir_esc=y#v=onepage&q=pinhole%20intitle%3A%22Multiple%20View%20Geometry%20in%20computer%20vision%22&f=false (accessed December 1, 2022).
[43] C.G. Broyden, The Convergence of a Class of Double-rank Minimization Algorithms 1. General Considerations, IMA J. Appl. Math. 6 (1970) 76–90, https://doi.org/10.1093/imamat/6.1.76.
[44] Y. Zhou, J. Shi, X. Yang, C. Wang, S. Wei, X. Zhang, Rotational objects recognition and angle estimation via kernel-mapping cnn, IEEE Access 7 (2019), https://doi.org/10.1109/ACCESS.2019.2933673.
[45] MNIST handwritten digit database, Yann LeCun, Corinna Cortes and Chris Burges, (n.d.). http://yann.lecun.com/exdb/mnist/ (accessed December 4, 2022).