Machine Vision and Applications manuscript No.
(will be inserted by the editor)

Calibrating a profile measurement system for dimensional inspection in rail rolling mills

Álvaro F. Millara · Julio Molleda · Rubén Usamentiaga · Daniel F. García

Received: date / Accepted: date

Abstract Modern, high-speed, railway transportation requires rails to conform strictly to requirements specified in various standards. One key requirement is the conformance of the dimensions of the rail cross-section to those of the corresponding rail model, within tight tolerances. This paper deals with a system for dimensional quality inspection during the manufacture of railway rails. Optical triangulation is used to build a profile from laser lines projected on the rails from four different locations. Then, the profile is compared to that of the corresponding rail model. The differences between certain numerical values (the dimensions) for the profile and the model are compared to standard tolerances for each dimension in order to detect dimensional defects. As a prerequisite for this, the cameras used to capture the laser lines must be calibrated. Standard calibration plates are unsuitable for sheet-of-light calibration in a production environment, as determining the location of the laser emitters relative to the plates would be an issue. For this reason, a cylinder-based calibration target is used instead. Different calibration algorithms are discussed, and compared to said standard calibration. The results of accuracy and repeatability tests in the production environment are also shown. The accuracy of the system is found to be appropriate for the purpose of quality inspection under the requirements of applicable rail standards.

Keywords Camera calibration · Dimensions · Inspection · Machine vision · Optical triangulation · Profile measurement system · Quality control · Rail · Rolling mill

1 Introduction

Steel rails are the basis for railways. They enable the movement of trains to transport both cargo and passengers over short and long distances; as well as cranes, to manipulate cargo in heavy industry and warehousing. Rails are engineered, designed and produced to support high-speed, heavily loaded modern trains. In order to ensure the safety and quality of transportation, tight control must be exerted

A. F. Millara · J. Molleda · R. Usamentiaga · D. F. García
Department of Computer Science and Engineering, University of Oviedo
E-mail: fernandezmalvaro@uniovi.es (corresponding author)

over both the internal structure of the rails, which affects their fatigue behavior [Ahlström *et al. (2005)*], and their shape, which affects the stability of both the rails and the railroad vehicles they support, as well as the amount of vibration suffered by the vehicles.

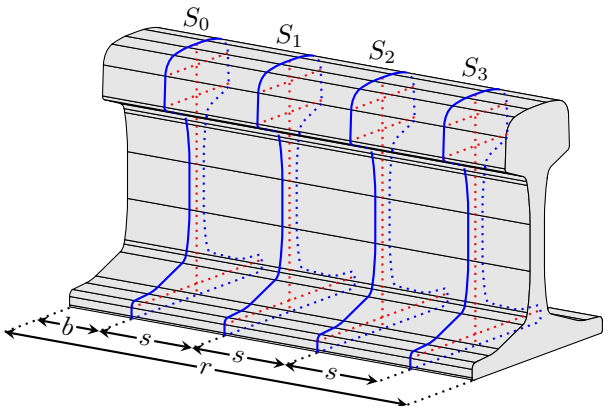

Fig. 1: Rail sections shown on a rail in blue. In red, some dimensions. r is the rail length, b is the distance between the beginning and the first section and s is the distance between sections. The distances depicted are not to scale.

These requirements have been codified in multiple standards that regulate the design and manufacturing of railway rails. Some examples of rail standards are EN-13674, by the European Committee for Standardization, [CEN (2011)]; the AREMA Manual for Railway Engineering, by the American Railway Engineering and Maintenance of way Association, [AREMA (2011)]; UIC Code 860, by the International Union of Railways, [UIC (2008)]; and GOST R 51685-2000, a Russian State Standard, [Gosstandart of Russia (2001)].

Rail standards include both destructive and non-destructive tests. The former include laboratory tests on chemical composition and material properties like tensile strength and hardness, whereas the latter are designed to detect surface and internal defects and measure rail dimensions and flatness. As described in [Papaelias *et al. (2008)*], non-destructive tests use a number of techniques such as laser triangulation, in order to determine the shape of the rail profile, for dimension and flatness measurement and surface defect detection; ultrasound, in order to detect internal defects; and eddy currents, in order to detect surface and near-surface defects. Most non-destructive tests are usually required to be automated and performed on all the rails, shortly after cooling.

Apart from quality inspection inside the factory, non-destructive tests are also periodically applied to the rails after they are installed, as the rail heads are subject to heavy wear and quickly degrade. Simple laser triangulation systems can be installed on rail maintenance vehicles to this end. Some examples of these systems are described in [Li *et al. (2007)*] [Liu *et al. (2011)*], [Zheng *et al. (2012)*], and [Wang *et al. (2017)*]. Whereas a number of works exist on the inspection of installed rails, academic papers on non-destructive testing (NDT) techniques for rail inspec-

tion in a factory setting are somewhat scarce. Traditional mechanical gauges and specialized portable coordinate-measuring machines, such as the Miniprof commercial profile measuring system, described in [Greenwood Engineering A/S (2010)], are an alternative for the inspection of rail samples, both on the tracks and in the factory. However, they would be too slow for the whole output of a rail factory.

This paper deals with a system for dimensional inspection of rails as they are being manufactured, using a laser triangulation setup. As stated before, the continuous use of mechanical gauges in factory conditions is not feasible because of speed concerns. Optical techniques for 3D measurement other than simple laser triangulation, such as fringe pattern projection, could potentially be used. However, such techniques require a more complex setup, and usually have a higher computational cost [Lohry *et al. (2014)*], which makes their use inadvisable for tasks where laser triangulation suffices.

This system is built on our previous work, first described in [Molleda *et al. (2016)*], and later in [Millara (2018) *et al.*], which is mostly concerned with the design of indicators for system health. The findings described in this paper increase the robustness of the rail profile measurement system presented in previous works by *i)* revising the calibration procedures, which have been vastly improved; and *ii)* designing a new feature to assess the accuracy of the measurement process.

Descriptions of sheet-of-light calibration methods suitable for a production environment are scarce in the literature, as are papers on non-destructive testing techniques for rail inspection in a factory setting. This paper contributes one such method, a procedure to calibrate the rail profile measurement system in industrial environments achieving similar accuracy to that the state-of-the-art methods can only achieve in laboratory environments.

2 Materials and Methods

This section describes the physical structure of the system, as well as the techniques, both well-known and novel, that were used for the implementation of the calibration features.

2.1 Physical structure

A theoretical model for a laser triangulation system is depicted in Figure 2, and can be described as follows:

Four laser emitters (designated L_1 to L_4, clockwise from top left to bottom left) are placed on the corners of a square. The laser emitters point towards the center of the square, forming a single *measurement plane* which contains the square and all the emitters.

Four cameras (designated C_1 to C_4, again clockwise from top left to bottom left) are placed on the corners of a second square. The sides of this second square are parallel to those of the first one, and the square is located on a plane, called the *camera plane*, which is parallel to the measurement plane. The cameras point towards the center of the first square.

Rails enter the system through the center of the second square, their longitudinal axes perpendicular to the measurement plane. While a rail traverses the structure,

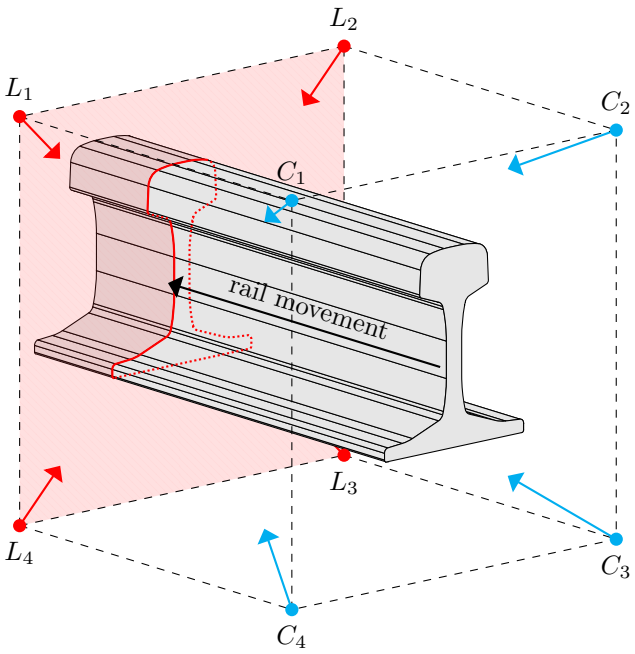

Fig. 2: Location of the cameras (C_1 to C_4) and laser emitters (L_1 to L_4) in the system, relative to the rails. The distances depicted in this figure are not to scale.

the cameras are simultaneously triggered every time the rail covers a given distance, and each of the cameras captures the laser line projected on the rail by its corresponding emitter. Here, the corresponding emitter of a given camera, C_n, is defined as the laser emitter L_n.

The cameras are connected to a computer, which applies the principles of laser triangulation, extracts the laser lines from the images and their location on the measurement plane, and computes the rail dimensions.

In practice, mechanical reasons prevent the emitters from actually being placed in such a way that their measurement planes match perfectly. In order for the cameras not to capture points contained in different measurement planes, which would greatly reduce the accuracy of the system, different wavelengths are used for contiguous laser emitters (so that L_1 and L_3 do not use the same wavelength as L_2 and L_4), and band-pass filters are attached to the cameras so that each camera can only capture a narrow range of wavelengths around that of the corresponding laser emitter.

Apart from this, the four emitters are assumed to form a single measurement plane, unless stated otherwise.

2.2 Camera calibration

Camera calibration is a process that computes a mapping between the *image coordinate system* and the *world coordinate system* (a coordinate system defined so

that the plane $Z = 0$ is the measurement plane). This mapping can then be used to translate pixels in the images into points in the measurement plane, and vice versa.

The simplest model for the relationship between a point in 3D space and its projection on an image plane (its coordinates in the image coordinate system) is the pinhole camera model. In this model, the camera has no lens, and its aperture is considered to be a point.

Let (u, v) be the coordinates in the image coordinate system, given in pixels, and (X, Y, Z) the coordinates in the world coordinate system, given in world units (usually meters or millimeters), where the measurement plane is defined as $Z = 0$. Then:

$$w \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \begin{bmatrix} R|T \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

The 3×4 matrix $P = K \begin{bmatrix} R|T \end{bmatrix}$ is usually called the *camera matrix*. Here, R is a 3×3 rotation matrix which represents the 3D rotation of the camera and T is a column matrix which expresses the translation of the camera relative to the world coordinate system. w is an arbitrary scale factor.

K, called the *camera intrinsic matrix*, can be defined as:

$$K = \begin{bmatrix} f_x & \gamma & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

where f_x and f_y designate the focal length in pixel widths and heights, respectively; c_x and c_y designate the pixel coordinates of the principal point of the image; and γ is the skew coefficient of the camera.

As stated before, the pinhole camera model lacks a lens, which means that it cannot account for lens distortion. Therefore, lens distortion must be modeled separately. The most prominent distortion patterns, as described in [Tsai (1987), Heikkilä and Silvén (1997)], and the ones usually modeled by computer vision frameworks, are radial distortions and tangential distortions.

On one hand, radially symmetric distortion is caused by the shape of the lens itself. Radial distortion can be modeled as follows. Let (x, y) be a pair of original, distorted coordinates in the camera coordinate system:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

and $r = \sqrt{x^2 + y^2}$. Then:

$$\delta_{r_x} = x(k_1 r^2 + k_2 r^4 + k_3 r^6 + ...)$$
$$\delta_{r_y} = y(k_1 r^2 + k_2 r^4 + k_3 r^6 + ...)$$

One or two coefficients are usually enough to accurately model radial distortion, as stated in [Heikkilä and Silvén (1997)], although three coefficients are often used.

On the other hand, tangential distortion appears because the lens and the image plane are not perfectly aligned. It can be modeled as follows. Again, let (x, y) be a

pair of distorted coordinates in the camera coordinate system, in the same way as before, and $r = \sqrt{x^2 + y^2}$:

$$\delta_{t_x} = 2p_1xy + p_2(r^2 + 2x^2)$$

$$\delta_{t_y} = 2p_2xy + p_1(r^2 + 2y^2)$$

Thus the relationship between distorted and undistorted camera coordinates can be given as:

$$x_{corrected} = x + \delta_{r_x} + \delta_{t_x}$$

$$y_{corrected} = y + \delta_{r_y} + \delta_{t_y}$$

with (x, y) defined as above.

There are more complete distortion models that take into account additional distortion patterns, like prism distortion, caused by imperfections in the lens. However, no popular computer vision framework takes into account such complex models. Alternatives to the polynomial approach described here also exist, such as [Fitzgibbon (2001)].

The calibration parameters described here are usually separated into two categories: intrinsic parameters and extrinsic parameters. Intrinsic parameters are the parameters that depend exclusively on the camera and its lens. These parameters do not vary for a given camera unless its lens is replaced or certain settings are changed. These parameters are the distorsion coefficients (usually k_1, k_2, k_3, p_1 and p_2), along with the values contained in the camera intrinsic matrix K: f_x, f_y, c_x and c_y. Extrinsic parameters are the parameters that describe the location and bearing of the camera relative to the world coordinate system. They are also known as the camera pose. Extrinsic parameters must be found again if the camera or the world coordinate system are moved in any way. The extrinsic parameters are the translation vector and the rotation matrix (or an alternative representation, such as a Rodrigues vector).

Calibration is performed by capturing certain features of a known calibration object, and building a mapping between the location of these features as they appear in the image, or images, with their location on the mentioned calibration object. This calibration object usually takes the form of a planar surface where a predetermined pattern of circles or squares is printed. The intrinsic calibration process implemented in modern computer vision frameworks is usually based on Zhang's camera calibration algorithm, described in [Zhang (2000)].

Intrinsic parameters generally need only be computed once, while extrinsic parameters may have to be computed periodically. When only extrinsic parameters are needed, a simpler calibration algorithm can be used that takes the intrinsic parameters as an input. This limited form of calibration can be called extrinsic calibration.

2.3 Camera calibration in the rail profile measurement system

In the case of the profile measurement system, the movement of rails through the structure causes the location of the cameras to suffer small, cumulative changes which quickly reduce the accuracy of the measurements, and eventually prevent the system from measuring altogether. Therefore, the system must include an extrinsic

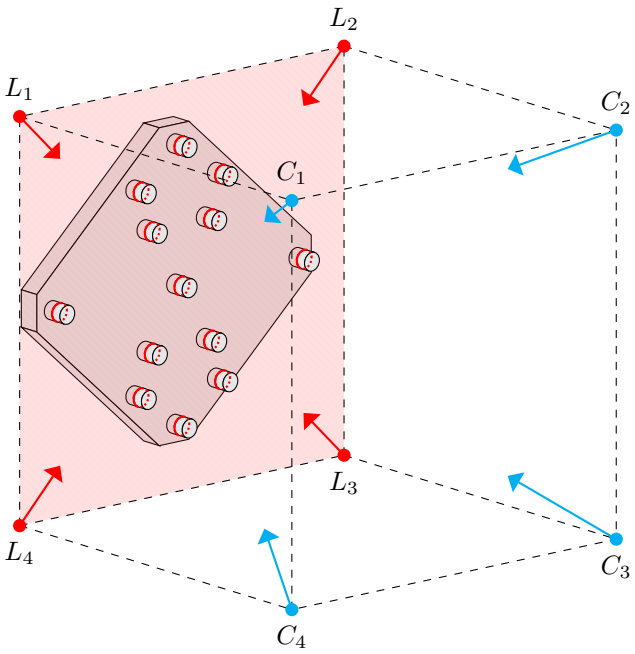

Fig. 3: The calibration target, shown in its position relative to the cameras (C_1 to C_4) and laser emitters (L_1 to L_4). The depicted distances are not to scale.

calibration procedure, in order to enable the users to periodically update the camera poses.

As for intrinsic calibration, it must be performed to find the intrinsic parameters of the cameras when first setting up the system and when replacing the cameras or their lenses. However, this can be done using existing tools, so a specific intrinsic calibration feature is not required.

General-purpose calibration patterns are unsuitable for calibrating a sheet-of-light setup in a production environment, as determining the location of the laser emitters relative to the plates would require the calibration plates to be placed precisely on the measurement plane, so that the computed camera poses, relative to the plates, would match the camera poses relative to said plane. Asking users to manually place the plates would not be practical, nor would it result in accurate poses. A robotic arm could be used in order to accurately place them and remove them when needed, but it would be prohibitively expensive, and itself require calibration.

Therefore, an alternative approach is required. As shown in Figure 3, a calibration target is used that is comprised of 13 cylinders that protrude from a surface. In order to perform extrinsic calibration, this surface must be placed roughly parallel to the measurement plane, and in such a way that the cylinders intersect said plane. In this way, the contour of each cylinder in the measurement plane is circular or near circular.

It must be noted that this specific pattern is rotationally symmetrical, as it remains the same when rotated by an angle of 180°. This causes an ambiguity that

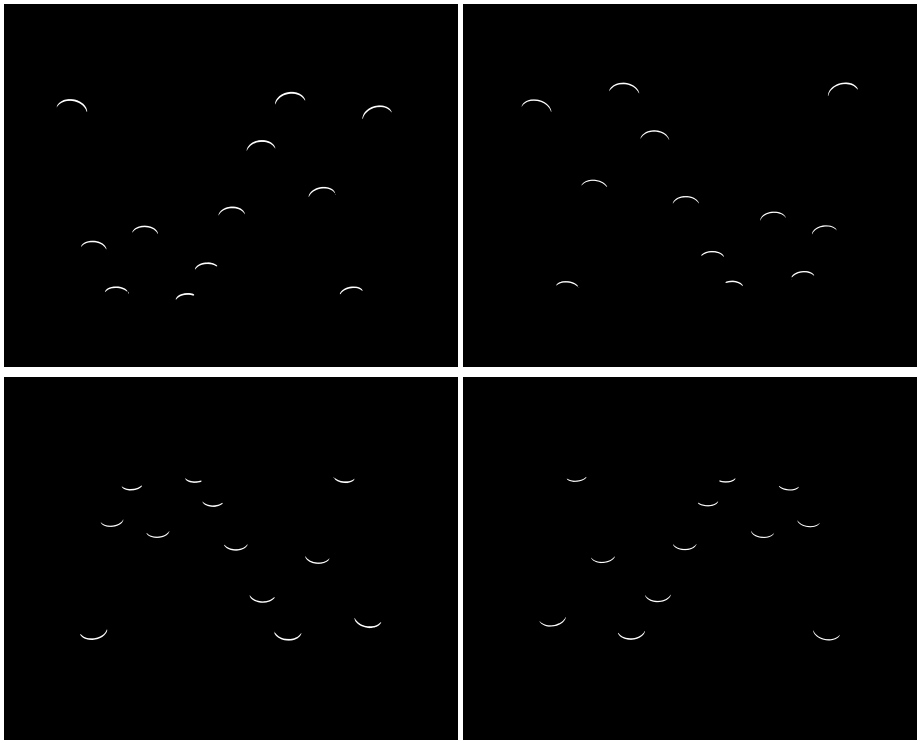

Fig. 4: The calibration target as seen from the four cameras. In each of the images, one of the cylinders is occluded, but the remaining twelve can be seen clearly.

must be taken into account when performing the extrinsic calibration. Although alternative cylinder configurations could be used that would avoid this issue, finding (and then implementing and testing) an alternate configuration that does not present rotational symmetry and where cylinders do not occlude the laser emitters, while ensuring accuracy in the relevant areas of the measurement plane, would be a significantly complex task. As such, and because this pattern has a specific purpose and its general position relative to the cameras will always be known, compensating for this issue later is more feasible.

Two algorithms have been devised to carry out extrinsic calibration for the profile measurement system. These algorithms require the following inputs: *i)* one image of the calibration target from each of the cameras (Figure 4 shows an example set of images); *ii)* the radius and center of every cylinder on the calibration target; *iii)* an approximation for the angle by which the calibration pattern is rotated, as seen from each of the cameras (this is required in order to eliminate the ambiguity caused by rotational symmetry in the pattern, as outlined before); and *iv)* the set of intrinsic parameters for all the cameras.

2.3.1 The FitEllipses algorithm

The first algorithm, called *FitEllipses*, generates a pose for each of the cameras as follows:

1. Laser lines are detected in the image and extracted using Steger's algorithm, as described in [Steger (1996)]. A predefined number of points at the end of the lines is discarded, as extraction of the line ends is usually inaccurate.
2. Ellipses are fit to the lines using the direct least-square method.
   All lines for which this is not possible, or that are unlikely to correspond to actual calibration cylinders, are discarded. Specifically, a line is discarded for all purposes if and only if its length is shorter than a third of the median length of all the lines, or an ellipse cannot be fit to it.
   Additionally, ellipses are marked as unusable for pose computation if their quality is low. These unusable ellipses cannot be discarded at this time, in order to prevent the cylinders with which they correspond from being associated with the wrong ellipse. An ellipse is marked as unusable for calibration if and only if its major radius is more than double its minor radius, or the sum of the absolute values of the extra angles exceeds a predefined value. The extra angles are defined as the angles between the horizontal line that goes through the center of the ellipse and each of the two lines that join the end of the laser line with the center of the ellipse.
3. In order to be able to compare them to the actual cylinder coordinates, the coordinates of the ellipse centers are made dimensionless. First, the ellipse that corresponds to the centermost cylinder is found. This is done by computing the centroid of the set of ellipse centers, and then using the ellipse center that is closest to it. Then, the coordinates of the center of the centermost ellipse are substracted from the coordinates of all ellipse centers, and all the ellipse centers are rotated by the predefined amount described before, roughly compensating for camera rotation. Finally, ellipse coordinates are constrained to the range $[-1, 1]$ by applying the following rules: Positive X coordinates are divided by the value of the highest positive X coordinate, negative X coordinates are divided by minus the value of the lowest (furthest from the center) negative X coordinate, positive Y coordinates are divided by the value of the highest positive Y coordinate and negative Y coordinates are divided by minus the value of the lowest (furthest from the center) negative Y coordinate. Cylinder coordinates are constrained in a similar way.
4. Each ellipse is associated with the cylinder that is closest in Euclidean distance to its adimensional coordinates, unless the distance to the cylinder exceeds a predefined threshold, or a different ellipse is closer to the same cylinder. All the ellipses that cannot be associated with a cylinder in this way are discarded. Ellipses that were marked as unusable in Step 2 are also discarded, and the associated cylinder is not taken into account.
5. The extrinsic parameters that minimize the distance between the original ellipse centers and their corresponding cylinders are found, as described in Section 2.2.
6. Optionally, the results may be improved by applying the known intrinsic parameters and the extrinsic parameters that were found in Step 5 to the original ellipse centers from Step 2, in order to translate them into the world coordinate system, then repeating Step 4 and Step 5 using the translated ellipse center coordinates

instead of the constrained coordinates (and the cylinder locations in the world coordinate system instead of their similarly constrained coordinates).
This may be an improvement over the initial pose because the described cylinder association algorithm is very conservative: in order to prevent incorrect correspondences from being used, only the most obvious correspondences are chosen. When the translated coordinates are used, more correspondences are obviously correct, and therefore available.
As an alternative approach to cylinder and ellipse mapping, all the possible sets of cylinder associations can be tried, in order to choose the one that minimizes calibration error. This is the approach used in [Usamentiaga *et al. (2016)*].

7. Also optionally, the results can be improved further by applying the pose obtained in the last step to the laser lines corresponding to the ellipses that were used, translating them to the world coordinate system, fitting circles to them and projecting the circle centers back to image coordinates, then finally repeating Step 5, except that circle centers are to be used instead of the corresponding ellipse centers.

2.3.2 The Iterate algorithm

The second algorithm, called *Iterate*, requires an initial pose in order to function, and uses it to generate more accurate poses for each of the cameras as follows:

1. A model of the calibration cylinders is built from circle primitives, using the known location and radius information.
2. Laser lines are extracted from the image as in Step 1 of the *FitEllipses* algorithm.
3. The initial pose is applied to the laser lines, translating them to the world coordinate system.
4. Correspondences between the points that constitute the translated laser lines and the model are found. As in [Usamentiaga *et al. (2018)*], performance is improved by using an R-tree structure when searching for correspondences.
5. Based on the correspondences between translated laser points and model points, correspondences between the original laser points and the same model points are trivially computed.
6. The extrinsic parameters that minimize the distance between the original laser line points and their corresponding model points are found, as described in Section 2.2.
7. Step 3 and all the steps after it are repeated, replacing the initial pose with the pose computed in Step 6. This is repeated until predefined convergence criteria (maximum number of iterations, minimum difference between successive poses) are reached.

Due to the significantly higher number of correspondences it uses, *Iterate* generates more accurate poses than *FitEllipses*. For this reason, both are employed to calibrate the system. First, *FitEllipses* is applied to compute an initial pose, then *Iterate* refines this pose and generates the final pose that will be used in the system. Step 6 and Step 7 of *FitEllipses* are not needed in this case, as high accuracy of the initial pose is not required.

2.4 Estimating the calibration error

The calibration error can be estimated in multiple ways, by applying the computed pose to the laser lines and comparing the result with the known cylinder pattern. Some of these ways are the following.

2.4.1 Radius error

After fitting circles to the translated laser lines, the radius error (err_r) is computed as the absolute difference between the radius of each of the circles (r_{fit}) and the real radius of the corresponding cylinder (r_{model}). For each laser line:

$$err_r = |r_{fit} - r_{model}|$$

Failure to correctly fit circles to the translated lines may cause high error values to be returned, even if ellipses could be fit to the original lines. Therefore, this estimation cannot be relied upon by itself.

2.4.2 Center error

After fitting circles to the translated laser lines, the center error (err_c) is computed as the Euclidean distance between the center of each of the circles (c_{fit}) and the center of the corresponding cylinder (c_{model}). For each laser line:

$$err_c = \sqrt{(c_{fit_x} - c_{model_x})^2 + (c_{fit_y} - c_{model_y})^2}$$

This estimation presents the same problem as the radius error estimation, as detailed before.

2.4.3 Point error

For each cylinder, the point error (err_p) is computed as the average absolute difference between the real radius of the cylinder and the Euclidean distance (d) between each point (p) of the corresponding laser line (ps) and the real cylinder center. For each laser line:

$$err_p = \sum_{d \in ds} \frac{|d - r_{model}|}{|ds|}$$

where ds is the set of distances between each point and the real cylinder center:

$$ds = \{\sqrt{(p_x - c_{model_x})^2 + (p_y - c_{model_y})^2} \mid p \in ps\}$$

2.4.4 Global point error

The global point error (err_{gp}) is computed for each of the laser line points as the difference between the radius of the nearest real cylinder and the Euclidean distance between the line point and the center of said cylinder. Therefore, it is obtained in the same way as point error, except that ps is replaced with the set of all points in all laser lines, and the radius of the nearest cylinder to each given point is used instead of r_{model}.

Global point error takes into account all laser lines, including those that were discarded because ellipses could not be fit to them. Although this may include lines that were not actually part of the calibration target, most such lines are actually partially occluded cylinders that would not be taken into account in any other way. The radius, center and point error estimations operate on circles, whereas the global point error estimation operates on individual points. For this reason, all these estimations can be used with the results of the *FitEllipses* algorithm, which also operates on circles.

The radius, center and point error estimations are not readily available for *Iterate*, and they have the shortcomings detailed before, whereas global point error is robust and does not require correspondences to be computed. Thus, the latter is chosen as the main error estimation to be used when comparing poses.

2.5 Testing the calibration

In order to both ensure the quality of the computed poses and determine when the system should be re-calibrated, a simple testing procedure has been devised. This testing procedure complements the general calibration error computations described in Section 2.4 by approximating the measurement error for different dimensions.

The testing procedure is carried out in the following way. First, four images of the calibration target are captured, one from each of the cameras. Preferably, these should not be the same images that were used for extrinsic calibration. Then, laser lines are extracted from the images, translated into world coordinates and combined into a single point cloud. The points that make up the line are associated with their corresponding calibration target cylinder and a circle is fit to the set of points associated with each of the cylinders. Finally, the distances between circles are measured and compared with the known distances between the corresponding cylinders, and the computed error is reported to the user. Distances are measured between the farthest parts of the circles, so as not to rely on potentially noisy estimations for the circle centers.

Not all these distances can be displayed to the user, however, as the result would be unwieldy. Therefore, only a small number of distances are shown: those that are considered to be representative of actual measurement error, based on analysis of the calibration target and the rail profiles. Specifically, the distances between cylinders that are located in regions of the measurement plane where relevant rail dimensions are computed during measurement. Figure 5 shows the calibration target cylinders, and the distances that are considered when testing the camera poses.

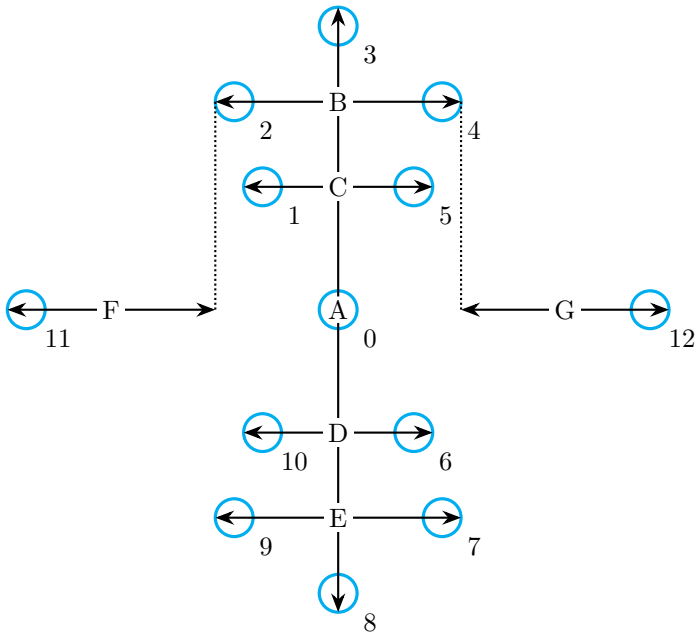

Fig. 5: Distances considered when testing the camera poses. Lines A to G attempt to simulate rail height, head width, upper web thickness, lower web thickness, foot width, left asymmetry and right asymmetry, respectively.

3 Results and discussion

The system described in this paper was implemented both in a laboratory environment and in a production environment in a rail rolling mill. The hardware that was employed is detailed in Appendices A and B, respectively.

3.1 Experiments on calibration procedures

The quality of the proposed calibration procedures, and their influence on the measurements made by the system, must be studied experimentally in order to be able to determine the optimal settings for them.

3.1.1 Intrinsic calibration

In order to study the effect of multiple intrinsic calibration patterns and implementations on the accuracy of the resulting measurements, an experiment is performed in the following way. First, images of five calibration targets are taken in different positions within the camera field of view and at different angles. The patterns depicted on the calibration targets are similar to the ones shown in Figure 6. These include a circle pattern with a matte white background (similar to Figure 6.**a**), for use with most versions of the MVTec HALCON software (**WhiteCircle**), described in [MVTec Software GmbH (2016), Section 3.2.3]; a circle pattern with

a reflective background (otherwise similar to Figure 6.**a**), also for use with most versions of the MVTec HALCON software (**ReflectiveCircle**); a small circle pattern (similar to Figure 6.**b**), for use with MVTec HALCON 12 and newer, described in [MVTec Software GmbH (2016), Section 3.2.3] (**SmallCircle**); a concentric ring pattern (similar to Figure 6.**c**), for use with the technique proposed in [Vo *et al. (2011)*] (**Concentric**); and an 8×7 checkerboard pattern (similar to Figure 6.**d**); for use with the MATLAB Computer Vision System Toolbox, as detailed in [The MathWorks, Inc. (2018), p. 3-220] (**MATLAB**). A set of calibration images is taken for each of the patterns. Additionally, five additional images of the checkerboard pattern are taken, at a distance similar to the one between the camera and the calibration target in the production environment.

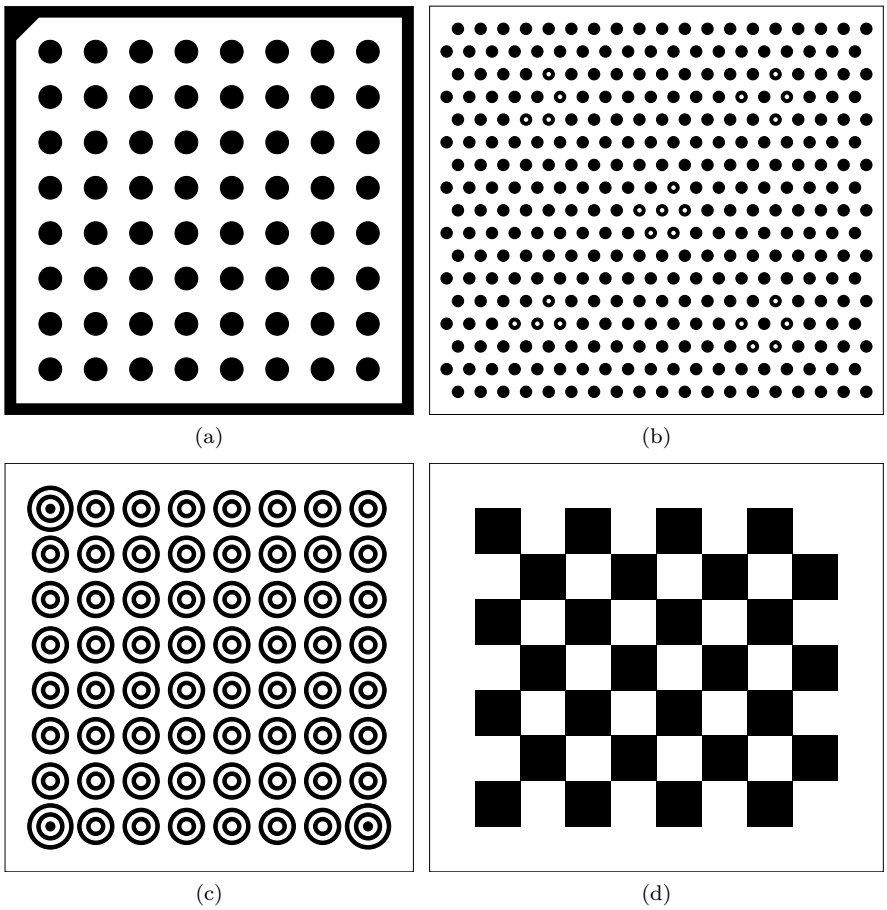

Fig. 6: Some examples of calibration targets: (**a**) A circle pattern used in most versions of the MVTec HALCON framework; (**b**) a circle pattern used in recent versions of MVTec HALCON (12 and newer); (**c**) a concentric ring pattern; and (**d**) the checkerboard, used in the MATLAB calibration toolbox and other frameworks.

Intrinsic parameters are generated for each of the image sets, using the implementation that corresponds to each calibration pattern. Then, each of the additional checkerboard pattern images is used to generate a pose using every set of intrinsic parameters. Each possible pair of a pose and a set of intrinsic parameters is used to translate each of the additional checkerboard pattern images to world coordinates, with the exception of the one that was used to generate the pose. The absolute differences between square sides in the translated checkerboards and known square sides are finally averaged for each pose and pattern image, as an estimation of the calibration error.

	WhiteCircle	*ReflectiveCircle*	*SmallCircle*	*Concentric*	*MATLAB*
Best	0.026	0.024	0.020	0.025	0.021
Median	0.045	0.039	0.031	0.035	0.036
Worst	0.072	0.058	0.046	0.055	0.085
Average	0.047	0.040	0.033	0.037	0.039
Std. dev.	0.014	0.008	0.008	0.009	0.018

Table 1: Comparison among sets of intrinsic parameters. Calibration error is given in mm. Each set of intrinsic parameters is labeled with the name of the calibration pattern that was used to generate it.

Table 1 compares the calibration error for each set of intrinsic parameters. It can be seen that the calibration error is very similar even with different algorithms and patterns. Error values are given here in mm, rather than computing the reprojection error (which is usually preferred when comparing intrinsic calibration techniques), because this gives a better idea of the influence of different intrinsic calibration techniques on the overall measurement accuracy at a similar distance.

3.1.2 Extrinsic calibration in a laboratory environment

In order to evaluate the extrinsic calibration procedures for the profile measurement system, as described in Section 2.3, they are compared with a well-known sheet-of-light calibration procedure in the laboratory environment.

This well-known calibration procedure, described in [MVTec Software GmbH (2016), Section 6.3.1] and depicted in Figure 7, entails capturing images of the calibration plates, such as those shown in Figure 6, roughly perpendicular to the measurement plane at two or more different heights. The poses of these calibration plates are computed as described before, the laser lines projected on them are extracted and translated into world coordinates relative to one of the plates and a plane is finally fit to these world coordinates. For the purposes of this work, this will be called the *SheetOfLight* algorithm.

Images for the experiment are captured in the following way:

1. The system is set up as depicted in Figure 3. The emitters are aligned so that their laser planes are near-coplanar to the naked eye.
2. Three sets of images of the calibration target are taken: with the target centered between the laser emitters and with the target displaced roughly 3 cm to the left and right.

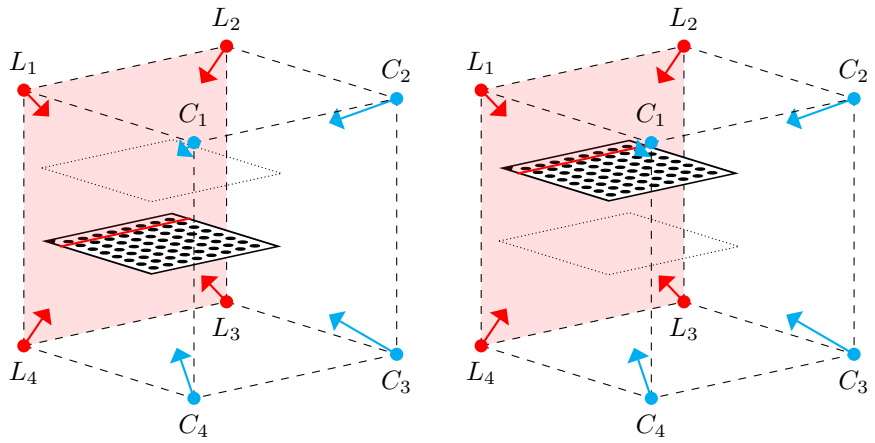

Fig. 7: Sheet of light calibration with the MVTec HALCON circle pattern. The pattern is shown at two different heights, relative to the cameras (C_1 to C_4) and laser emitters (L_1 to L_4). The red lines represent the intersection between the measurement plane and the pattern.

3. The calibration target is removed from the system, and a circle pattern for MVTec HALCON is placed perpendicular to the measurement plane.
4. Images of the calibration target are taken from the upper cameras (1 and 2). A first set of images is captured with the laser emitters turned off and a high exposure time (between 400 ms and 1000 ms, depending on the camera, as they employ different wavelength filters), so that the pattern is clearly visible. Three other sets are captured with the laser emitters turned on and three different low exposure times (specifically, 5 ms, 10 ms and 15 ms), so that the laser line can be seen. All parts of the laser line that appear in regions of the image other than the plate are manually removed with an image editor for the purposes of the experiment.
5. The plate with the circle pattern is elevated roughly 3.5 cm and 8 cm from its initial location, and Step 4 is carried out again for both new positions.
6. The plate is removed and a gray polyethylene test piece, similar to the one shown in Figure 8, is placed under the laser emitters.
7. Images of the laser lines projected on the test piece are taken from the upper cameras with different exposure times (2.5 ms to 15 ms, in increments of 2.5 ms). Again, all parts of the laser lines other than the upper region are manually removed with an image editor.

After these images have been captured, 21 sets of poses are generated, 12 for the proposed calibration algorithms (three for each of the four algorithms) and 9 for the *SheetOfLight* algorithm. This is done in the following ways:

The *FitEllipses* and *Iterate* calibration algorithms are applied to the calibration target images. In the case of *FitEllipses*, both the initial pose and the poses that result from Step 6 and Step 7 above are considered, called *FitEllipses1*, *FitEllipses2* and *FitEllipses3*, respectively. The pose that results from *FitEllipses3* is used as the initial pose for *Iterate*. As three sets of calibration target images are used, 3 poses

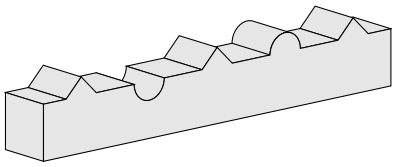

Fig. 8: Calibration test piece..

result for each camera and algorithm or algorithm stage. In this experiment, two points are clipped from each end of the laser lines when calibrating with *FitEllipses*, and 10 points are clipped in the case of *Iterate*. The influence of different numbers of clipped points on calibration results will be explored in Section 3.1.3.

The well-known *SheetOfLight* algorithm is applied to the plate and laser images. As three sets of plate images are used (each at a different level), three combinations of two images are possible. For each plate image, there are nine laser line images with different exposure times. In total, nine poses are generated for each camera using this algorithm.

Then, for each of the six sets of two test piece images (one image from each of the two upper cameras), lines are extracted using Steger's algorithm, as described before. 20 points are clipped from each line end. Then, each of the 21 sets of poses is applied to the extracted lines, and the translated lines for the two cameras are combined into a single test piece profile for each set of poses. Each profile is then aligned with the upper region of the model shown in Figure 8. Finally, for each profile, average distances from each point to the model are computed.

	FitEllipses1	*FitEllipses2*	*FitEllipses3*	*Iterate*	*SheetOfLight*
Best	0.052	0.049	0.047	0.041	0.037
Median	0.079	0.077	0.096	0.052	0.055
Worst	0.131	0.084	0.102	0.070	0.087
Average	0.087	0.071	0.082	0.056	0.064
Std. dev.	0.030	0.013	0.023	0.010	0.017

Table 2: Comparison between a well-known algorithm and the calibration algorithms for the profile measurement system. Best, median and worst profile errors (each computed as the average distance between profile points and the model) are shown (in mm) for each of the considered calibration algorithms, along with the average error and the standard deviation.

The best, median and worst of these average distances are computed for each of the algorithms employed. The results are depicted in Table 2. It can be seen that the results of *FitEllipses2* are a small improvement over those of *FitEllipses1*, while the performance of *FitEllipses3* appears to be worse. *Iterate* supposes an important improvement over all of them, and its results are similar to, and slightly better than, those of *SheetOfLight*. As such, and taking into account that using *SheetOfLight* in the production environment is not feasible, it is clear that, among the available algorithms, *Iterate* should be used in the system.

3.1.3 Extrinsic calibration in a production environment

The extrinsic calibration procedures for the system are also evaluated by studying their performance on a set of real calibration plate images captured in the production environment. To this end, calibration plate images that were taken in a roughly two-year period (between September 2015 and October 2017) are retrieved. When multiple sets of calibration images (one for each of the four cameras) were taken on the same day, or on contiguous days, only the last such set is considered. After applying this criterion, 35 sets of images can be used.

Clipped points	*Failure* (%)	*Error* ≥ 1 mm (%)
0	0.00	0.48
1	12.14	0.48
2	0.00	0.24
3	0.00	0.24
4	0.00	0.71
5	0.71	0.95
6	2.14	1.19
7	2.86	0.95
8	3.57	1.67
9	7.14	1.67
10	11.43	0.95
11	17.14	0.24
12	24.29	0.24

Table 3: Percentages of invalid calibrations for the *FitEllipses* algorithm. Both the percentage of images for which the algorithm failed (left) and the percentage of images for which the global point error was 1 mm or greater (right) are shown, grouped by the number of clipped points to each end of the laser lines.

FitEllipses clipped points	*FitEllipses1+Iterate*	*FitEllipses2+Iterate*	*FitEllipses3+Iterate*
0-8	0.052	0.052	0.052
9-12	0.051	0.051	0.051

Table 4: Comparison between the results of the *Iterate* algorithm using different initial poses. Global point error (in mm) is shown for the results of *Iterate* with 10 clipped points, using initial poses originating from different *FitEllipses* stages with different numbers of clipped points.

It must be noted that cases where the calibration algorithms failed to return a valid result, or where the operators decided to discard the result without saving, are not considered here. Therefore, this experiment is not suitable for the goal of studying the reliability of the algorithms in conditions other than optimal. Only their best-case error values are studied in this way.

For each set of calibration images, and for each possible number of clipped line points from 0 to 12, poses for all three *FitEllipses* stages (*FitEllipses1*, *FitEllipses2* and *FitEllipses3*, as described in Section 3.1.2) are computed when possible.

The *Iterate* algorithm is also applied to every set of calibration images, using the *FitEllipses3* pose with two clipped points for a given set as an initial pose, as clipping two points from the line ends appears to be optimal or close to optimal when applying *FitEllipses* to these images. *FitEllipses3* also appears to have better performance than the other *FitEllipses* stages on this set of images, as will be explained below. *Iterate* is also applied for all possible numbers of clipped line points from 0 to 12.

Error is measured as described in Section 2.4. The original (unclipped) laser lines are used in the case of the global point error, while the other error estimations use the same (clipped) lines as the calibration algorithms.

Figure 9 shows that apart from the radius error, which is clearly a poor error estimation, all the other estimations show the error values for *FitEllipses2* as being slightly lower than *FitEllipses1*, and those of *FitEllipses3* as being consistently better than both of them, despite the opposite being true for the laboratory environment, as seen in Section 3.1.2. It is also shown that the performance of *Iterate* is significantly better than those of the *FitEllipses* stages.

As for the number of clipped points, in the case of *FitEllipses* the optimal value appears to be two, when considering center and point error. For the global point error, the optimal value seems to be zero, according to the graph. However, taking into account the fact that no points were clipped when computing it for this experiment, and that the same images were used for calibration and for error estimation (as only one set of images is available for each calibration), it follows that the slightly lower error values when no clipped points were used for calibration might be caused by overfitting of the poses. Therefore, a value of two is finally chosen for *FitEllipses* (although different image resolutions may require different numbers of clipped points).

For *Iterate*, no value appears clearly better for these sets of images. However, lower quality images may benefit from potentially noisy laser line ends being clipped. As this does not appear to affect pose quality in any significant way, a value of 10 clipped points for each line end is chosen.

Table 3 shows the percentage of invalid calibrations depending on the number of clipped points for the *FitEllipses* algorithm. The *Iterate* algorithm succeeded for all the images for which *FitEllipses* succeeded. It must be noted that two points were clipped from each end in the production environment, and calibration failures were discarded, so the failure percentage for two clipped points could not be anything other than 0 %. However, it can also be seen that no value improves the percentage of calibrations with error higher than 1 mm over that seen in the case of two clipped points, which reinforces the above conclusions.

Additionally, Table 4 shows error values for *Iterate* (with 10 clipped points) depending on the initial pose. It can be seen that the quality of the original poses is largely (but not completely) irrelevant when employing the *Iterate* algorithm.

3.2 Experiments on measurement accuracy and repeatability

The accuracy and repeatability of the measurement results of the system are studied in the following way.

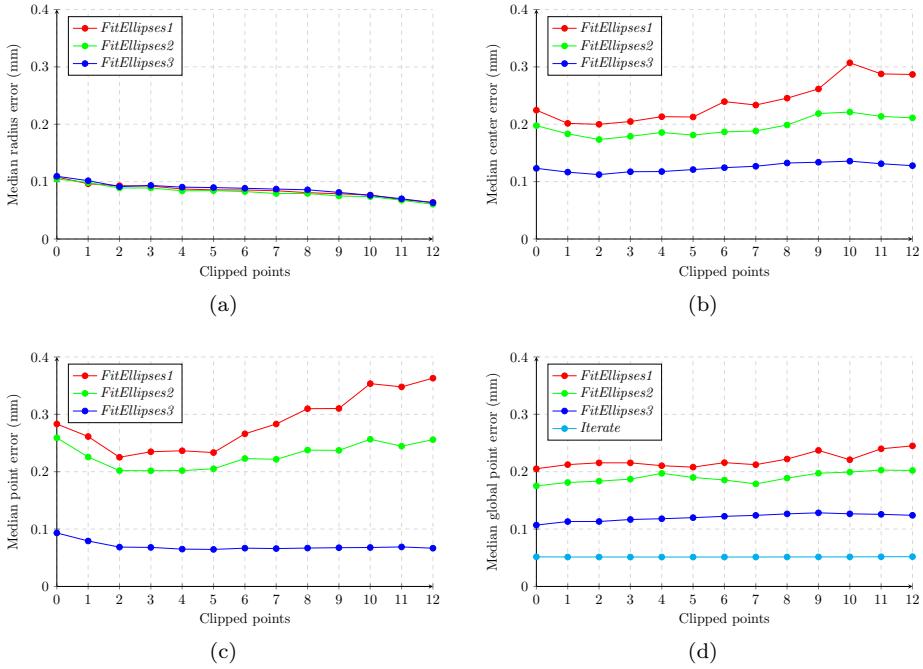

Fig. 9: Median error for different numbers of clipped points: (**a**) radius error, (**b**) center error, (**c**) point error and (**d**) global point error.

A randomly selected rail (60 E1) is inspected five consecutive times with the rail profile measurement system under the same working conditions in the rail rolling mill. It is also inspected with a Greenwood Miniprof Rail gauge [Greenwood Engineering A/S (2010)], which features an accuracy better than $\pm 11\,\mu$m and a repeatability of $\pm 2.5\,\mu$m.

Once the rail is inspected in the mill, it is cut into pieces, and several specimens are measured in a metrology laboratory using a Trimos Temp 1282 coordinate-measuring machine featuring an accuracy better than $\pm 1\,\mu$m in FW and better than $\pm 20\,\mu$m in HW and RH. Dimensions provided by the MiniProf gauge and the CMM are used to assess the dimensions computed by the PMS.

Table 5 shows differences among the measurements computed by the dimensional inspection system described in this paper (five repetitions of the experiment), and the measurements carried out by the technicians in the metrology lab, considered the ground truth. Also, differences with the dimension provided by the MiniProf Rail gauge are shown. As can be seen, errors provided by the system are between -0.05 mm and 0.20 mm. This means that dimensions provided by the system for HW, FW and RH have an error of less than 0.12 %.

In addition to accuracy, in order to evaluate the performance of the system, a repeatability evaluation is carried out to assess the consistency of the system. The repeatability evaluation expresses the range of measurements provided for the same dimension of the same rail specimen under consistent conditions. This test is carried out using the same method, on the same item, in the same installation, by the same

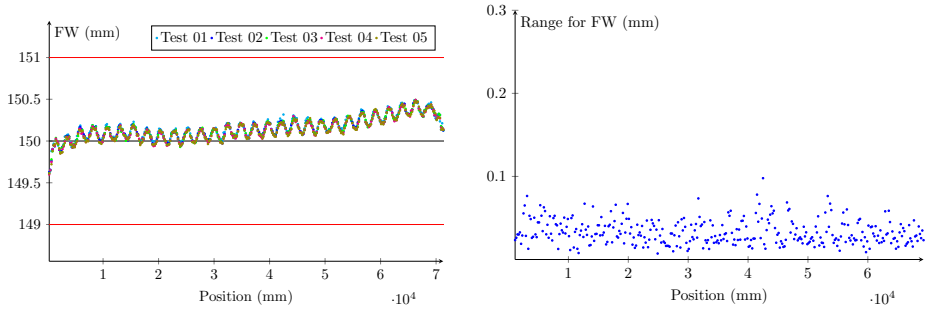

Fig. 10: Results of the repeatability evaluation for *Foot Width* (FW): (**a**) Computed values along the rail for the five test measurements (the horizontal black line is the expected value, whereas the horizontal red lines are the maximum and minimum tolerated values; the sinusoidal pattern in the measured values can be attributed to small irregularities in the rail caused by the rolling mills); (**b**) range of the values obtained from the five test measurements along the rail.

operator and using the same system within short intervals of time, as specified in ISO 5725-1:1994.

Figure 10 shows an example of the repeatability evaluation. The computed values for the five consecutive inspections of the rail for FW as well as the range of the measurements for this dimension in the five runs. Red lines indicate the lower and upper limit imposed by the specification to which the rail is manufactured; the black line is the expected value according to the specification. As can be seen, the system is highly repeatable.

Table 5: Accuracy of dimensional inspection (in mm). Dimensions computed using the profile measurement system (PMS) and a MiniProf Rail gauge (MP) are compared with the metrology lab results (GT).

Dim.	Length	PMS-GT					MP-GT
		#1	#2	#3	#4	#5	
HW	10000	-0.02	-0.04	-0.06	-0.04	-0.04	0.15
FW	10000	0.01	-0.01	-0.03	-0.02	-0.03	-
RH	10000	-0.08	-0.10	-0.09	-0.11	-0.11	-
HW	20000	0.05	0.04	0.04	0.04	0.03	0.15
FW	20000	0.03	0.02	0.03	0.03	0.03	-
RH	20000	-0.16	-0.16	-0.12	-0.12	-0.12	-
HW	30000	0.03	0.03	0.03	0.04	0.01	0.07
FW	30000	-0.05	-0.04	-0.06	-0.05	-0.05	-
RH	30000	-0.15	-0.19	-0.14	-0.14	-0.17	-
HW	40000	0.01	0.01	0.01	0.00	0.01	0.05
FW	40000	0.00	0.04	0.00	0.02	0.01	-
RH	40000	-0.19	-0.19	-0.20	-0.19	-0.19	-
HW	50000	-0.01	0.00	0.00	0.01	0.00	0.08
FW	50000	0.03	0.03	0.02	-0.01	0.00	-
RH	50000	-0.19	-0.19	-0.20	-0.20	-0.19	-
HW	60000	0.00	0.00	0.00	-0.01	0.00	0.13
FW	60000	0.01	-0.01	-0.02	-0.01	-0.03	-
RH	60000	-0.13	-0.12	-0.11	-0.12	-0.07	-

4 Conclusions

Different methods to calibrate a system for dimensional quality inspection of rails are described. Using the best such method, an accuracy better than ± 0.1 mm is achieved for most of the considered dimensions.

Rail standards usually allow for tolerance ranges of approximately 1 mm. Therefore, the stated accuracy can be considered sufficient for the purpose of dimensional quality inspection for rails. The usage of higher-resolution cameras in the rail profile measurement system would improve this accuracy, if needed.

Findings of this work show that although specific intrinsic calibration algorithms might make more of a difference for other cameras that produce more distorted images, the differences are negligible in the case of the cameras used in the laboratory environment for the purposes of this work. We attribute this to the high quality of the employed cameras and lenses.

Once we have a clear understanding about the implications of calibrating a profile measurement system for dimensional inspection of rails in rolling mills, we can state that the main contribution of this work is the proposed calibration technique which allows us to compute extrinsic parameters in industrial environments with an accuracy similar to that of well-known approaches that cannot be used effectively in an industrial context.

Acknowledgements This work was supported by the Spanish National Plan for Research, Development and Innovation [TIN2014-56047-P], by project FUO-EM-372-14 and by the "Severo Ochoa" program of the Asturian regional government (Administración del Principado de As-

turias) [PA-17-PF-BP16009]. None of these funding sources had any involvement in study design, collection, analysis or interpretation of data, the writing of this paper or the decision to submit it for publication.

The authors would also like to thank the technicians of ArcelorMittal Asturias for their helpful assistance during tests.

Conflict of interest

The authors declare that they have no conflict of interest.

A Hardware in the laboratory environment

Cameras: Genie Teledyne Dalsa HM1400.
Lenses: Schneider-Kreuznach APO-XENOPLAN 2.0/24-0005.
Laser filters: Coherent 635 CW - 20 BP and Coherent 685 CW - 20 BP.
Laser emitters: Coherent StingRay-640 (640 nm) and Coherent Stingray-685 (685 nm).

B Hardware in the production environment

Cameras: Genie Teledyne Dalsa TS-M2560.
Lenses: Goyo 16mm HR 1" F1.4 C.
Laser filters: MidOpt Bi632 and MidOpt BP695.
Laser emitters: Coherent StingRay-640 (640 nm) and Coherent Stingray-685 (685 nm).

References

[Ahlström *et al. (2005)]* Ahlström, J.; Karlsson, B. Fatigue behaviour of rail steel—a comparison between strain and stress controlled loading. *Wear* **2005**, *258(7)*, 1187–1193.

[CEN (2011)] European Committee for Standardization. EN-13674 — Railway applications - Track - Rail, 2011.

[AREMA (2011)] AREMA. AREMA Manual for Railway Engineering — Chapter 4: Rails, 2011.

[UIC (2008)] International Union of Railways. UIC Code 860 — Technical specification for the supply of rails, 2008.

[Gosstandart of Russia (2001)] Gosstandart of Russia. GOST R 51685-2000 — Railway rails - General Specifications, 2001.

[Papaelias *et al. (2008)]* Papaelias, M.; Roberts, C.; Davis, C.L. A review on non-destructive evaluation of rails: State-of-the-art and future development. *Proceedings of The Institution of Mechanical Engineers Part F-journal of Rail and Rapid Transit - PROC INST MECH ENG F-J RAIL R* **2008**, *222*, 367–384.

[Li *et al. (2007)]* Li, G.; Wang, C.; Liu, J.; Jin, W. Dynamic rail-wear inspecting system based on machine vision. *IEEE Conference on Industrial Electronics and Applications* **2007**, pp. 1–4.

[Liu *et al. (2011)]* Liu, Z.; Sun, J.; Wang, H.; Zhang, G. Simple and fast rail wear measurement method based on structured light. *Optics and Lasers in Engineering* **2011**, *49*, 1343–1351.

[Zheng *et al. (2012)]* Zheng, S.; Chai, X.; An, X.; Li, L. Railway track gauge inspection method based on computer vision. *IEEE International Conference on Mechatronics and Automation* **2012**, pp. 1292–1296.

[Wang *et al. (2017)]* Wang, C.; Ma, Z.; Li, Y.; Zeng, J.; Jin, T.; Liu, H. Distortion calibrating method of measuring rail profile based on local affine invariant feature descriptor. *Measurement* **2017**, *110*, 11–21.

[Greenwood Engineering A/S (2010)] Greenwood Engineering A/S. MiniProf Digital Profile Measuring: Your way to valid and reliable data. Commercial brochure, 2010. Retrieved on 2017-07-26.

[Lohry *et al. (2014)]* Lohry, W.; Chen, V.; Zhang, S. Absolute three-dimensional shape measurement using coded fringe patterns without phase unwrapping or projector calibration. *Opt. Express* **2014**, *22*, 1287–1301.

[Molleda *et al. (2016)]* Molleda, J.; Usamentiaga, R.; Millara, A.F.; García, D.F.; Manso, P.; Suárez, C.M.; García, I. A Profile Measurement System for Rail Quality Assessment During Manufacturing. *IEEE Transactions on Industry Applications* **2016**, *52*, 2684–2692.

[Millara (2018) *et al.]* Millara, A.F.; Molleda, J.; Usamentiaga, R.; Garcia, D.F. Profile Measurement of rails in rolling mills: integrating autonomic computing capabilities. IEEE Industry Applications Society Conference 2018.

[Tsai (1987)] Tsai, R.Y. A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-shelf TV Cameras and Lenses. *IEEE Journal of Robotics and Automation* **1987**, *3*, 323–344.

[Heikkilä and Silvén (1997)] Heikkilä, J.; Silvén, O. A Four-step Camera Calibration Procedure with Implicit Image Correction. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* **1997**, pp. 1106–1112.

[Fitzgibbon (2001)] Fitzgibbon, A.W. Simultaneous linear estimation of multiple view geometry and lens distortion. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* **2001**, *1*, I–125–I–132 vol.1.

[Zhang (2000)] Zhang, Z. A Flexible New Technique for Camera Calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2000**, *22*, 1330–1334.

[Steger (1996)] Steger, C. Extracting Lines Using Differential Geometry And Gaussian Smoothing. *International Archives of Photogrammetry and Remote Sensing* **1996**, *XXXI, part B3*, 821–826.

[Usamentiaga *et al. (2016)]* Usamentiaga, R.; García, D.F.; delaCalle, F.J. Real-time inspection of long steel products using 3D sensors: calibration and registration. *IEEE Transactions on Industry Applications* **2016**, pp. 1–8.

[Usamentiaga *et al. (2018)]* Usamentiaga, R.; García, D.F.; Molleda, J. Efficient registration of 2D points to CAD models for real-time applications. *Journal of Real-Time Image Processing* **2018**, Volume 15, issue 2, pp. 329–347.

[MVTec Software GmbH (2016)] MVTec Software GmbH. HALCON 12.0.2 Solution Guide III-C 3D Vision, 2016. Retrieved on 2018-11-13.

[Vo *et al. (2011)]* Vo, M.; Wang, Z.; Luu, L.; Ma, J. Advanced geometric camera calibration for machine vision. *Optical Engineering* **2011**, *50*, 110503.

[The MathWorks, Inc. (2018)] The MathWorks, Inc.. Computer Vision System Toolbox(TM) Reference, Rev. September 2018, for R2018b. Retrieved on 2018-11-13.