AUTOMATED CALIBRATION OF FEM MODELS USING LIDAR POINT CLOUDS

B. Riveiro [a*], G. Cubreiro [a], B. Conde [a], M. Cabaleiro [a], R. Lindenbergh [b], M. Soilán [a], JC. Caamaño [a]

[a]Dept. of Materials Engineering, Applied Mechanics and Construction, School of Mining Engineering, University of Vigo, 36310, Spain - (belenriveiro@uvigo.es, gelocubreiro@hotmail.es, bconde,mcabaleiro,msoilan,jccaam@uvigo.es)
[b]Dept. of Geoscience & Remote Sensing, Faculty of Civil Engineering and Geosciences, Delft University of Technology, 2628 CN Delft, The Netherlands – (R.C.Lindenbergh@tudelft.nl)

Commission II, WG II/10.

KEY WORDS: Point clouds, LiDAR, rasterization, FEM analysis, bending moment, torque

ABSTRACT:

In present work it is pretended to estimate elastic parameters of beams through the combined use of precision geomatic techniques (laser scanning) and structural behaviour simulation tools. The study has two aims, on the one hand, to develop an algorithm able to interpret automatically point clouds acquired by laser scanning systems of beams subjected to different load situations on experimental tests; and on the other hand, to minimize differences between deformation values given by simulation tools and those measured by laser scanning. In this way we will proceed to identify elastic parameters and boundary conditions of structural element so that surface stresses can be estimated more easily.

1. INTRODUCTION

In recent years several methods for the automatic processing of point clouds acquired with laser scanning have been developed. Some of them were focused on converting those point clouds into finite elements models in order to perform structural evaluation, as in Riveiro et al., 2016; Truong-Hong et al., 2013; Cabaleiro et al., 2014 or Castellazzi et al., 2015. This technique has also some disadvantages, as there is not a control over points that are being measured. This means that a certain laser scanner does not measure the same point at different moments so surface identification techniques are needed. Nowadays, inverse engineering processes (as Finite Element Model (FEM) calibration) are limited to laboratory tests applications, where structural elements are also subjected to a known load distribution. Other disadvantages are possible variation on local point cloud properties, the presence of temporary objects, etc. On non-rigid surface monitoring a large number of reliable reciprocations and a priori assumptions are needed (Tam et al., 2013); the establishment of reciprocations without presence of control points is still a problem that need to be investigated.

Studies related to tensional state estimation on structural elements from LiDAR data have been implemented at laboratory. In these first attempts some simplifications were adopted, e.g., the applied loads are known, they are just focused on bending effects, as presented by Gordon et al., 2003 or Gordon and Lichti, 2007. In their works, a beam deflection measuring model based on constrained least squares fitting to the beam mechanics equations was developed. Other example is implemented on Park et al., 2007, where a computational model for stress calculation on deformed beams by FEM techniques is proposed. Choi et al., 2013 uses laser scanning to model measurements of the serviceability responses of a building structures under dynamic loading. In Olsen et al., 2010, a beam is discretized into elements and volumetric change is obtained by analysing each cross section.

Finite elements method has acquired increasing importance within the field of structural engineering where applications can range from mechanical systems optimization (Costas et al., 2014) to structural health assessment on constructions during service life (Conde et al., 2016). Nevertheless, computer models do not always represent precisely real cases because unknown values for some parameters have to be adopted; moreover, it is usually desirable to make simplifications due to the model complexity. Parameter identification can be adopted to approximate the value of the unknowns. Influential input parameters are those whose value widely impacts the simulation results. This is followed by an optimization process in order to find the real value of the influential input parameters. This is achieved by comparing FEM results with field measurements.

Even though the current limitations described above, our work aims to explore the potential of LiDAR point clouds for the calibration of FEM models. We pretend to determine the most accurate values for those elastic parameters that mostly influence the structural behaviour of structural members. Moreover, the case studies evaluated with our methodology do not only focus on the effects of bending (as most of the previous works), but also combine the effects of torsion. Actually, this second effect is particularly challenging in open cross sections such as I, L or double T, which have a very low torsional rigidity. This paper presents an overview to the methodology for the fully automatic processing of the data collected in real case studies. Finally, the results obtained by our algorithm in terms of deformation, as well as values of the influential input parameters calibrated during the optimization process using FEM, are presented.

2. METODOLOGY

The point cloud processing chain was focused on identifying the surface to be evaluated during the comparison of the real point cloud (measured by a terrestrial laser scanning) and the simulated point cloud (those points whose defection are obtained through the FEM analysis). The methodology starts defining the reference system for both point clouds and the method for the establishment

correspondences between measures of both real and simulated point clouds. After that, our approach was based on exploring the normals of points with the purpose of determining the weight of bending or torsion in the surface deformation. Thus, using the theories of beam deformation in order to be able to estimate the normal and shear stresses at the surface of analysis. This algorithm was implemented in Matlab.

On the other hand, a numerical model of the beam was built based on the geometry measured during the experiments. FEM analysis was performed with Abaqus and, thanks to its Python scripting interface, we can access to model characteristics from Matlab to change the values of the input parameters. We executed sensitivity analysis based on Pearson's correlation coefficients for several elastic variables. Young's modulus was found to be the only influential input parameter and it was later subjected to the optimization process. The optimization processes were based on a genetic algorithm, whose fitness is quantified by a weighting function. The algorithm searches for the value of the Young's modulus that represents a global minimum of the weighting function. The results show that value which best fits to data is 237.82 GPa (usual Young's modulus value is 210 GPa). The programming was developed in Matlab environment communicated with Abaqus kernel.

2.1 Automated segmentation of the surface of analysis

Before processing experimental point clouds within the algorithm we have to discard those point that are not part of the beam, this is made manually through CloudCompare software; then, the point cloud is imported in Matlab and it is rasterized with the purpose of using the raster structure as the link between the real point cloud and numerical model simulation results. Additionally, this raster image is used to map the estimated stresses on the beam flange surface.

From scanned beam only upper flange points are needed for programming, since they are the ones measured without occlusions. At this part of the work not only undesired points will be deleted but also it will be made a noise filtering, produced both by laser scanning measuring error and other phenomena, as mixed-pixel, produced when light beam hits a discontinuous surface so it is divided in two or more providing a virtual point whose coordinates are not real.

2.1.1 Horizontal surface filtering from normal vectors: We will keep points that are located on a horizontal surface, which can be done with a principal components analysis, for which a neighbourhood radio is defined and in each point it is got the position of its neighbours. The first step of a principal components analysis is to calculate covariance matrix:

$$\Sigma = \begin{pmatrix} \sigma_x{}^2 & \sigma_{xy} & \sigma_{xz} \\ \sigma_{xy} & \sigma_y{}^2 & \sigma_{yz} \\ \sigma_{xz} & \sigma_{yz} & \sigma_z{}^2 \end{pmatrix} \tag{1}$$

Where variance in each axis xi or covariance in each pair of axis $xixj$ are calculated as:

$$\sigma_{xi}{}^2 = \frac{1}{n}\sum_{k=1}^{n}\left(xi_k - \overline{xi}\right)^2 \tag{2}$$

$$\sigma_{xixj} = \frac{1}{n}\sum_{k=1}^{n}\left(xi_k - \overline{xi}\right)\left(xj_k - \overline{xj}\right) \tag{3}$$

Where n is the number of points, xi_k is the xi coordinate of k point and $\overline{xi}$ is the mean of all xi coordinates.

If we calculate matrix eigenvectors we get three orthogonal axis for which covariances are null, and diagonal elements get maximum or minimum values, what is the same, we get a Cartesian axis system that represents direction of maximum and minimum data deviation. The third eigenvector represents normal direction to plane that best fits the points and can be used for a thresholding operation.

2.1.2 Elimination of points that are not found in plane or edge: A method for point classifying based on their dimensionality have been purposed recently, form principal components analysis (Gressin et al., 2013). Points in neighbourhoods describe shapes that can be represented as an ellipsoid whose diameters are oriented according to eigenvectors and sizes are directly related with eigenvalues, thereby we can classify point distributions into three different shapes: linear predominance (a_{1D}), planar predominance (a_{2D}) and volumetric predominance (a_{3D}).

Let σ_1, σ_2 and σ_3 be typical deviations of points with respect to eigenvectors 1, 2 and 3, respectively, so that $\sigma_1 \geq \sigma_2 \geq \sigma_3$. Dimensional coefficients are calculated as:

$$a_{1D} = \frac{\sigma_1 - \sigma_2}{\sigma_1} \; ; \quad a_{2D} = \frac{\sigma_2 - \sigma_3}{\sigma_1} \; ; \quad a_{3D} = \frac{\sigma_3}{\sigma_1} \tag{4}$$

That is how we can identify planar points (flange ones) and linear points (flange edge ones) and discard volumetric shapes.

2.1.3 Connected components method (coarse noise filtering): Point cloud is turned to a three dimensional image and elements that are not in close contact with an enough number of other elements are removed; that can be done by a voxelization process followed by a connected components algorithm (Riveiro et al., 2016). In graph theory, connected components are any subgraph which is connected to other one by a path and is not connected to graph vertex. When working with Matlab binary images analysis (2D or 3D), connected components are sets of pixels/voxels marked as 1 and connected to each other, isolated from the remainder ones. We create a binary image from voxelization and mark as 1 those voxels that have any point inside, then we identify connected components in order to delete small groups that are not in close contact with big ones.

2.1.4 Image limits definition: Point cloud has undefined limits; in order to determine the boundary of beam flange we will focus on zones where point density increases and delete regions (in x and y) that has a few number of points. The aim of this part of the program is to get four data: minimum and maximum x coordinates and minimum and maximum y coordinates, creating a rectangle at xy plane that will be the boundary of the rasterized image.

2.1.5 Fine noise filtering: Starting from the assumption that deformed beam flange is a surface, a way for identifying those points that are placed in a wrong position is to fit them to a polynomial surface and to check which ones are further away than a certain distance from fitted surface.

Beam mechanics theories state that the deflection $y(x)$ of a pin or fixed-supported beam can be parameterized by two polynomials that are tangent at the point of application of load so they shape a continuous and derivable curve that cannot be expressed as a single function of x. In order to create the surface we can fit the

data to a high-degree polynomial; the higher the grade, the greater the fitting. In next Figure they are shown the fitting of three polynomial to the deflection of a fixed-supported beam subjected to a punctual load. It can be seen that fitting increases with polynomial degree.
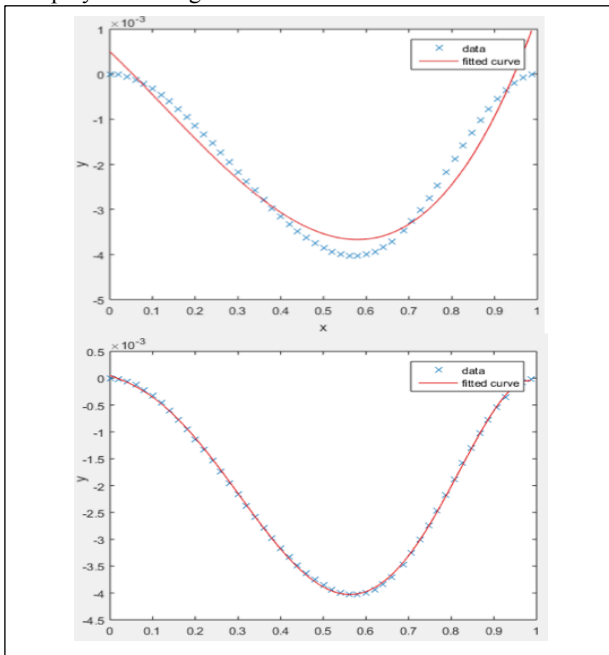

Figure 1. Comparison of point beam profile and fitted curves: Fitting to a third degree polynomial (top) and fitting to a fifth degree polynomial.

In view of he above, we can say that fitted polynomial surface needs a high degree in longitudinal direction (x). On the other hand, based on the assumptions of beam mechanis that state the straight sections keep straight after deformation when the beam is subjected to torque, the polynomial degree in the transversal direction (y) is 1. Finally, point cloud is fitted to a polynomial surface $f(x^5, y^1)$ just with the purspose of remove noisy points whose residuals exceed a given threshold.

2.1.6 Cloud smoothing: Despite having filtered the fine noise, remainder points are not at a smooth surface due to measuring uncertainty, so a final processing has to be done. In this part of the work, each point high (z coordinate) is replaced by the mean high of its neighbours, as follows:

$$\overline{z}_i = \frac{1}{n_i} \cdot \sum_{j=1}^{n_i} z_{i,j} \qquad (5)$$

$\overline{z}_i$ is i point new z coordinate, n_i is the i neighbourhood population and $z_{i,j}$ is the z coordinate of point j. Method is comparable to a moving average filter.
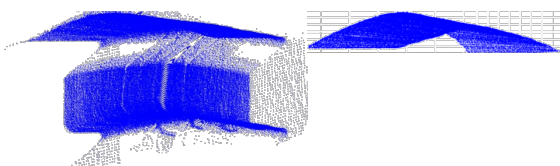

Figure 2. Visualization of point cloud before and after process.

2.2 Creation of real point cloud images

2.2.1 Test conditions: Tested beam is a European HEB100 profile with 587 cm length, made of S235 steel (yield strength 235 MPa). Beam has been equipped with three arms where the load is applied, resulting in bending and torsion simultaneously. At load application points, beam was reinforced using stiffeners in order to prevent sectional warping. Beam ends are fixed to ground with thin metal plates. Different load cases has been evaluated, in all of them beam is subjected to a punctual off centred forces that causes both bending and torsion.

During experiment a FARO Focus 3D (FARO Technologies Inc., Lake Mary, Florida, USA) was used. This tool measures distances in a range of 0.6 to 120 m with a nominal precision of ±2 mm, 25 m distance in normal illumination and reflectivity conditions (each at 90% and 10% reflectivity). Field of view is 305º vertically and 360º horizontally. Maximum angular resolution is 0.009º. Laser scanner has been placed at a distance of 2.6 m away the beam with an incidence angle of 63º. In horizontal plane, incidence angle was up to 50º. Minimum spatial resolution on the beam was 5 mm.

2.2.2 Creation of raster images: the link between experimental data and simulation results from Abaqus software is a raster structure that serve as geometric reference in order to compare both data. Laser scanner disadvantage is that there is no control over registered points location; they are distributed along beam surface randomly so there is a need of control over certain points if we want to compare different clouds. Rasterization consists in choosing a reference plane and dividing it in cells (pixels), then points are projected orthogonally. Each cell is assigned as a data the height of upper beam flange point located in the vertical line that contains pixel centre, this data is calculated as the mean z coordinate of points projected in the cell and mean heights are used to recreate three-dimensional object.
For the rasterization both the minimum x and y coordinates are calculated and resolution in each axis are defined (p_x and p_y). Each point is assigned an index as a function of the pixel row number in each axis and each cell data is got as mean height of points located above the pixel.

2.3 Analysis of the images created from the real point clouds.

Due to superposition principle, we can analyse separately bending and torsion.

2.3.1 Normal stresses: Bending effects are calculated from beam deflection, which is the curve formed by each centroid of the straight sections. However, in the experimental case we can only measures the upper flange points. Within the elastic regime, deflections are so small and consequently fibers can be supposed to have the same shape. This means that those points of the flange lying in the straight line defined by the normal to the flange surface passing through the centroid, are parallel to the deformed axis of the beam (the aforementioned beam deflection).
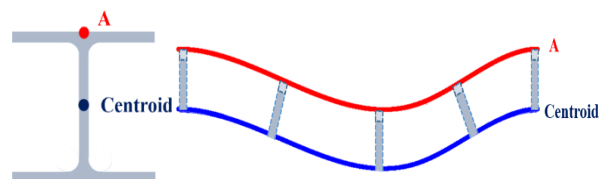

Figure 3. Height profile of a bended beam; blue lines if formed by centroids, and red one is formed by A points of each section.

According to this, the deflection is measured as the height of the middle pixel corresponding to each cross section. The deflection curve is computed as the polynomial function fitted to the measured heights. For a concentrated load, the deflection curve is parameterized by two third-degree polynomial whose generic expression is:

$$\delta_1(x) = a_1 x^3 + b_1 x^2 + c_1 x + d_1 \quad (6)$$

$$\delta_2(x) = a_2 x^3 + b_2 x^2 + c_2 x + d_2 \quad (7)$$

Where δ_1 and δ_2 are polynomial at each side of load application point. According to Euler-Bernoulli beam theory, the second derivative of the deflection curve can be used to determine the bending moment at each section:

$$6a_1 x + 2b_1 = -\frac{M_{F1}(x)}{EI} \quad (8)$$

$$6a_2 x + 2b_2 = -\frac{M_{F2}(x)}{EI} \quad (9)$$

So, according to Navier's flexure formula so:

$$\sigma_{n1}(x) = -(6a_1 x + 2b_1)EI\frac{z_{MAX}}{I} = A_1 x + B_1 \quad (10)$$

$$\sigma_{n2}(x) = -(6a_2 x + 2b_2)EI\frac{z_{MAX}}{I} \, A_2 x + B_2 \quad (11)$$

Where σ are normal stresses, E is Young's Modulus, I is section moment of inertia and z_{MAX} is maximum distance between centroid and any upper flange point. Taking this relation in count, final results are parameters A_1, B_1, A_2 and B_2 that define surface stresses field.

2.3.2 Shear stresses: Twist angle on each section can be calculated from height difference between two points. The further the points are, the more accurate torque calculation will be. Twist angle on real beam can be obtained from height differences between parallel pixel rows.
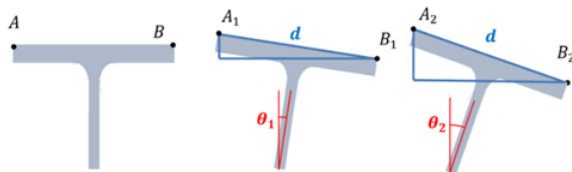

Figure 4. Twist angle thought as height differences between two upper flange points.

Twist angle diagram on a pin or fixed-supported beam subjected to a punctual torque is made of two straight lines crossed at the cross section where the load is applied. The general expression of any fiber height profile is given by:

$$\Delta h_1(x) = a_1 \cdot x + b_1 \quad (12)$$

$$\Delta h_2(x) = a_2 \cdot x + b_2 \quad (13)$$

As we know the relation between twisted angle θ and Δh:

$$\Delta \theta_1(x) = a_1 / d \cdot x + b_1 / d \quad (14)$$

$$\Delta \theta_2(x) = a_2 / d \cdot x + b_2 / d \quad (15)$$

Torque diagram $T(x)$ depends linearly on twist angle diagram's first derivative, so:

$$T_1(x) = \frac{a_1 GJ}{d} \quad (16)$$

$$T_2(x) = \frac{a_2 GJ}{d} \quad (17)$$

Which means that torque diagrams are got from fitting coefficients of straight lines to height profiles (a_i), from material properties (shear modulus G) and section (torsion section constant J and radius of gyration d). From Saint-Venant's principle, maximum shear stress on a HEB100 section takes place at flange middle point and its value is calculated with equation 17:

$$\tau = \frac{T}{J} \cdot t \quad (17)$$

Where T is sectional torque, J the torsional constant and t is flange thickness. From previous expression, it is deducted that surface shear stress field is defined from torque diagrams.

2.4 Parameter identification and generation of synthetic point clouds

This part of the work refers to the numeric modeling of structural element with the purpose of finding a model that minimizes differences between deflections got from FEM analysis, and those measured with the laser scanner. The synthetic point cloud comprises the nodes of the FEM mesh corresponding to the upper flange after simulation. This optimization problem is designed with the aim of identifying accurately the elastic parameters of the material and boundary conditions. The final aim is to estimate the stresses at the surfaces registered by the scanner in a more accurate manner.

The geometric model was created from the point cloud measured by the scanner using Solidworks software. For the simulation, the materials and boundary conditions parameters defined in section 2.2.1. have been refined iteratively, as explained in below.

The work is developed in two stages: first, a sensitivity analysis of all considered input parameters; and second, the optimization of sensitive variables in order to identify model parameters as accurate as possible. For sensitivity analysis, using Pearson's method a coefficient for each variable is obtained; those variables that have a high value are considered as influential parameters for simulation; while, non-important ones will be assigned with the standard values as defined in section 2.2.1.

2.4.1 Numerical model: FEM analysis was performed with Abaqus software. Python scripting interface was used to access to model characteristics defined in Matlab in order to change the value of the input parameters.

Simulation results must be compared with rasterized image. A way to do it is to use output data from Abaqus that allows us to obtain final position of mesh nodes, so we get the point cloud of the upper flange beam that has to be rasterized and converted into same coordinate system as those coming from LiDAR data.

2.4.2 Sensitivity analysis: We executed sensitivity analysis taking in count the following variables:

- Support rigidity: the ends of the beam have been fixed to the ground by two end-plates that have a large width but small thickness so beam behaves as fixed-supported in the torsional model and as pin-supported in the bending model with some rigidity against end surfaces twist.
- Elastic parameters: Young's modulus and Poisson's ratio.
- Steel density.

The sensitivity analysis is based on Pearson's correlation coefficients, which assess influence of each input parameter X on a function result Y.

A weighting function is defined in order to measure fitting goodness of an input parameter set to real point clouds; function calculates a scalar that measures fitting as an error. Chosen scalar is root mean square error (RMSE), which follow this expression:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(z_{Refi} - z_{Abi}\right)^2} \qquad (25)$$

Z_{Refi} is i pixel height of scanned image, z_{Abi} is i pixel height from Abaqus image and n is number of pixels.

Sensitivity analysis programming was developed in Matlab environment communicated with Abaqus kernel. Obtained results show that only Young's Modulus is an influential parameter on simulation results, so it will be the only one introduced on optimization process.

2.4.3 Optimization process: It is based on a genetic algorithm, which evolves a population (each individual is a list of variables) by random actions that simulate biologic action through genetic changes and natural selection, based on individual adaptation to environment. Adaptation (or fitness) is quantified by a weighting function. It is especially useful when solving not-derivable optimization problems. In this work we try to search Young's modulus value that represents a global minimum of the weighting function. Programming was developed in Matlab environment communicated with Abaqus kernel, and results show that value which best fits to experimental data is 237.82 GPa (usual Young's modulus value given in standards is 210 GPa).

3. RESULTS

Using the methodology presented in section 2 it was possible to estimate stresses due to bending and due to torsion in a beam subjected to a concentrated loading.

The beam deflection curve was obtained from the height profiles of middle pixel of each cross-section. The distribution of normal stress for each cross-section of the beam was quantified through the bending moment, whose diagram was obtained by derivation of the deflection curve (Fig. 5).

Torque diagram is got from derivation of a fitted straight line (constrained least squares) to height difference (Δz) at each cross-section profiles (Fig. 6).
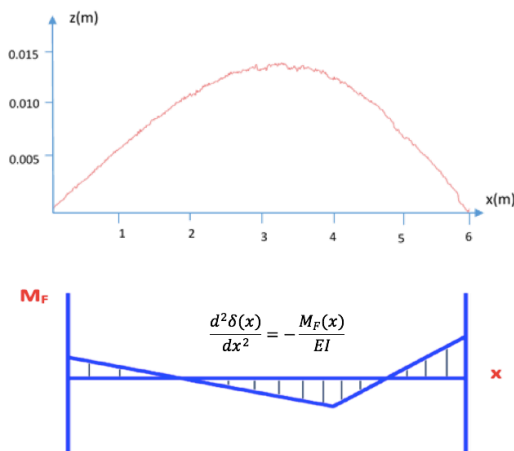

Figure 5. Height images at the centroid (top) and derived bending moment diagram (bottom) for obtained from a certain load.
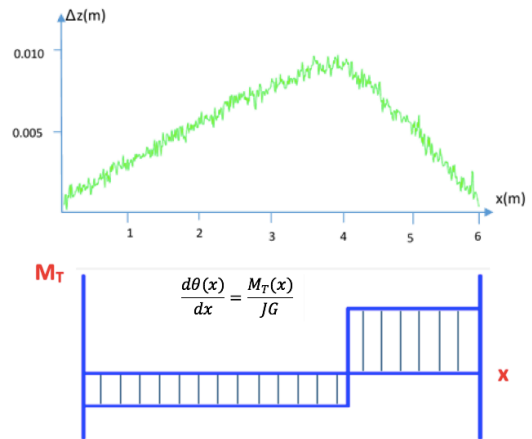

Figure 6. Height difference (Δz) profiles (top) and derived torque diagram (bottom) obtained for a certain load.

From the point clouds it was possible to detect beam defects, as residual plastic deflections from previous tests, which provoke that the height profiles do not to fit with the theoretical ones. Moreover, beams are subjected to gravity action (self weight) and this must be taken into account when comparing results with the numerical model. The solution is to evaluate differences between images from several cases so, using the superposition principle, resultant images represent load steps between tests. Results obtained for bending are shown in Figure 7:
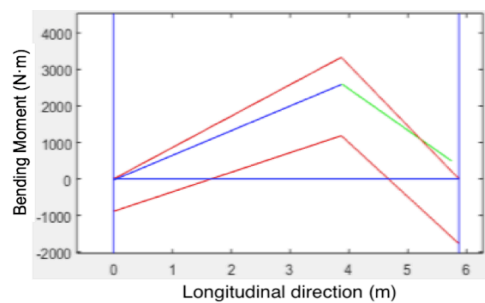

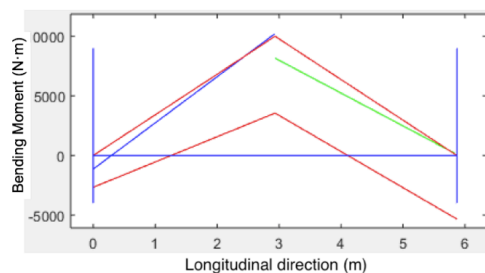Figure 7. Bending moments for a concentrated load of 2000N, 2 m away from right end.

Figure 8. Bending moments for a concentrated load of 6000N at mid span.

Blue and green lines correspond to bending moments, upper red line represents diagram for a pin-supported beam, lower red line represents diagram for a fixed-supported beam, and both cases are obtained from beam mechanics theory, solving problem assuming exact values for loads. It is difficult to obtain the error because supports rigidity were not assessed precisely (as it was considered a not influential input parameter) and therefore theoretic value is unknown. However, we can predict that bending moment at each section will be covered between pin and fixed-supported cases.

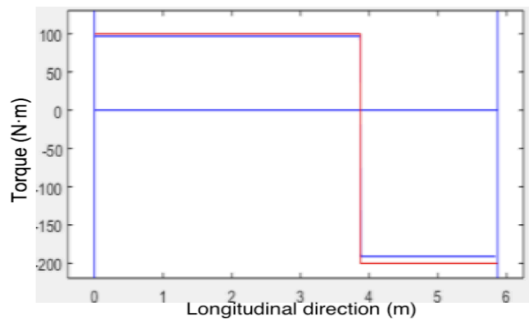Results obtained for torsion are shown below:

Figure 8. Torque obtained from the deformed point clouds for a concentrated load of 2000N applied 2 meters away from right end.
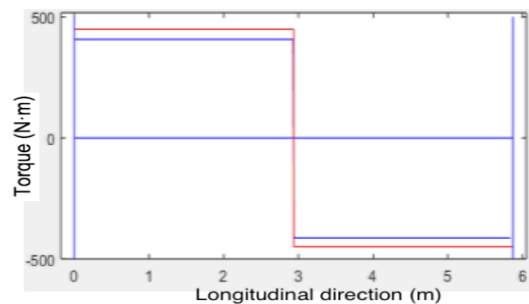

Figure 8. Torque obtained from the deformed point clouds for a concentrated load of 6000N applied at mid span.

Red lines correspond to the theoretic values (solving elastic problem) of torque, and blue lines are obtained values for the beam tested. We can observe that the difference is in any case under 6%.

ACKNOWLEDGEMENTS

4. CONCLUSIONS

This work presents an algorithm able to measure automatically deflection due to both bending and torque based on 3D point clouds, without direct measuring at beam surface. FEM analysis where performed in order to find out influential input parameters, calculate accurately beam properties and thereby minimize errors at stress evaluation. It was found out that only influential variable is Young's modulus and its optimum value is 237.82 GPa. Shear stress measures differs less than 6% from theoretical values and normal stress measures cannot be known accurately but they are inside theoretical tolerances. It was verified that for metallic structures is not only important to control deflections due to bending but also those produced by torsion, especially in open cross sections which have a very low torsional rigidity.

REFERENCES

Cabaleiro, M., Riveiro, B., Arias, P., Caamaño, J. C., & Vilán, J. A. (2014). Automatic 3D modelling of metal frame connections from LiDAR data for structural engineering purposes. ISPRS Journal of Photogrammetry and Remote Sensing, 96, pp. 47-56.

Castellazzi, G., D'Altri, A. M., Bitelli, G., Selvaggi, I., & Lambertini, A. (2015). From laser scanning to finite element analysis of complex buildings by using a semi-automatic procedure. Sensors, 15(8), pp. 18360-18380.

Choi SW, Kim BR, Lee HM, Kim Y, Park HS. (2013). A deformed shape monitoring model for building structures based on a 2D laser scanner. Sensors, 13(5), pp. 6746-6758.

Conde, B., Díaz-Vilariño, L., Lagüela, S., & Arias, P. (2016). Structural analysis of Monforte de Lemos masonry arch bridge considering the influence of the geometry of the arches and fill material on the collapse load estimation. Construction and Building Materials, 120, pp. 630-642.

Costas, M., Díaz, J., Romera, L., & Hernández, S. (2014). A multi-objective surrogate-based optimization of the crashworthiness of a hybrid impact absorber. International Journal of Mechanical Sciences, 88, pp. 46-54.

Gordon SJ, Lichti D, Stewart M, Franke J (2003). Structural deformation measurement using terrestrial laser scanners. Proceedings of the 11th FIG Symposium on Deformation Measurements, Santorini, Greece, pp. 25-28.

Gordon SJ, Lichti DD. (2007). Modeling terrestrial laser scanner data for precise structural deformation measurement. Journal of Surveying Engineering, 133(2), pp. 72-80.

Olsen MJ, Kuester F, Chang BJ, Hutchinson TC. (2010). Terrestrial laser scanning-based structural damage assessment. Journal of Computing in Civil Engineering, 24(3), pp. 264-272.

Park HS, Lee HM, Adeli H, Lee I. (2007). Anew approach for health monitoring of structures: terrestrial laser scanning. Computer Aided Civil and Infrastructure Engineering, 22(1), pp. 1930.

Riveiro, B., DeJong, M. J., & Conde, B. (2016). Automated processing of large point clouds for structural health monitoring of masonry arch bridges. Automation in Construction, 72, 258-268.

Tam, G. K., Cheng, Z. Q., Lai, Y. K., Langbein, F. C., Liu, Y., Marshall, D., ... & Rosin, P. L. (2013). Registration of 3D point clouds and meshes: a survey from rigid to nonrigid. IEEE transactions on visualization and computer graphics, 19(7), pp. 1199-1217.

Truong-Hong, L., Laefer, D. F., Hinks, T., & Carr, H. (2013). Combining an angle criterion with voxelization and the flying voxel method in reconstructing building models from LiDAR data. Computer-Aided Civil and Infrastructure Engineering, 28(2), pp. 112-129.