

¹ DuoDIC: 3D Digital Image Correlation in MATLAB

² **Dana Solav¹¶ and Asaf Silverstein¹**

³ **1** Faculty of Mechanical Engineering, Technion, Haifa, Israel ¶ Corresponding author

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review ↗](#)
- [Repository ↗](#)
- [Archive ↗](#)

Editor: [↗](#)

Submitted: 01 March 2022

Published: unpublished

License

Authors of papers retain
copyright and release the work
under a Creative Commons
Attribution 4.0 International
License ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).

⁴ Summary

⁵ Three-dimensional Digital Image Correlation (3D-DIC) is a non-contact optical-numerical
⁶ technique for measuring the 3D shape and full-field displacement, deformation, and strain,
⁷ from stereo digital images of the surface of an object. 3D-DIC is useful in numerous applications,
⁸ such as characterizing the mechanical behavior of materials and structures, quantifying material
⁹ parameters, and validating numerical simulations.

¹⁰ DuoDIC is a freely available open source MATLAB toolbox for 2-camera stereo 3D-DIC, which
¹¹ can be used either as a standalone package or as functions in custom scripts. DuoDIC receives
¹² two series of synchronized images taken from two cameras: (1) images of a flat checkerboard
¹³ target, which are used to calibrate the stereo camera pair; and (2) images of a speckled test
¹⁴ object, which may undergo movement and deformation. The toolbox processes the image
¹⁵ series and integrates several camera calibration algorithms with the 2D subset-based DIC
¹⁶ software Ncorr ([Blaber et al., 2015](#)), to transform matching image points into 3D points, and
¹⁷ outputs a dynamic point cloud, meshed surfaces, rigid body motion, and full-field displacement,
¹⁸ deformation, and strain measures. Furthermore, DuoDIC offers advanced functions to visualize
¹⁹ various measures on the 3D meshes and overlaid on the original images.

²⁰ The simple user interface allows novice users to perform 3D-DIC analyses without interacting
²¹ with MATLAB syntax, while stand-alone functions can be integrated in custom scripts by
²² more proficient MATLAB users. As such, DuoDIC is suitable for students, researchers, and
²³ professionals in various fields.

²⁴ The package is composed of four main scripts: (1) stereo camera calibration; (2) image cross-
²⁵ correlation (2D-DIC); (3) 3D reconstruction; and (4) post-processing. This paper describes
²⁶ the algorithms implemented in each step and demonstrates its performance in two test cases,
²⁷ which are also included as sample data: rigid body translations of a cylindrical container and
²⁸ uniaxial tension of a rubber dog-bone specimen.

²⁹ Statement of need

³⁰ 3D-DIC is useful in numerous applications where the full-field displacements and strains are
³¹ required, primarily for measuring the mechanical response of materials and structures and for
³² characterizing mechanical properties, with widespread applications from nano- to macro-scale,
³³ and rapid progress and developments ([Pan, 2018](#); [Reu, 2015](#); [Sutton et al., 2017](#)). 3D-DIC is
³⁴ preferable over strain gages and extensometers due to its non-contact nature and the large
³⁵ amount of local information that can be collected.

³⁶ Commercial 3D-DIC software are typically expensive, proprietary and closed-source, which
³⁷ may pose a barrier, especially for students and researchers. Notable free and open-source
³⁸ 3D-DIC contributions include DICe ([Turner, 2015](#)), MultiDIC ([Solav et al., 2018](#)), and ADIC3D
³⁹ ([Atkinson & Becker, 2021](#)). Each of these packages uses different algorithms for correlation and
⁴⁰ stereo calibration and requires different calibration targets. Specifically, our previous toolbox
⁴¹ MultiDIC focused on multi-view applications, therefore required a non-planar calibration object,

42 which is relatively difficult to make accurate, thus preventing easy implementation. To this end,
 43 DuoDIC enables a simpler calibration procedure that requires only a flat checkerboard pattern.
 44 In addition, with respect to MultiDIC we improved the post-processing and visualization
 45 options. DuoDIC is written in MATLAB, providing the flexibility and simple implementation
 46 of a high-level language, which meets the needs of the experimental mechanics community.
 47 Moreover, a main advantage of open-source software is the ability to compare results and
 48 performance, and to cross-validate the implementation of different algorithms and approaches.
 49 To this end, DuoDIC may also comprise a complementary tool for researchers in the field.

50 Algorithms and workflow

51 The entire 3D-DIC procedure in DuoDIC is organized in four main scripts, which the user can
 52 run without having to interact with any MATLAB syntax. **Figure 1** outlines the workflow
 53 of the 3D-DIC procedure. The functionality and algorithms incorporated in each step are
 54 detailed in the sub-sections below. The figures in this section represent sample data, which are
 55 provided with the toolbox. The experimental setup used for taking these images consisted of
 56 two machine-vision cameras (BFS-U3-51S5M-C, FLIR, USA), each equipped with a 5.0 MP
 57 Sony IMX250 monochrome sensor and a 25 mm lens (Fujinon HF25SA-1, Fujifilm Corporation,
 58 Japan). The cameras were synchronously hardware-triggered using a MatchID Triggerbox
 59 (MatchID, Belgium). A rubber dog-bone specimen, cut out of a bicycle tube, was mounted on
 60 an optical table (Thorlabs, USA), such that its left end was fixed and its right end was moved
 61 to the right in 19 steps with 1 mm increments, such that a total of 20 images were taken from
 62 each camera (one reference state and 19 deformed states).

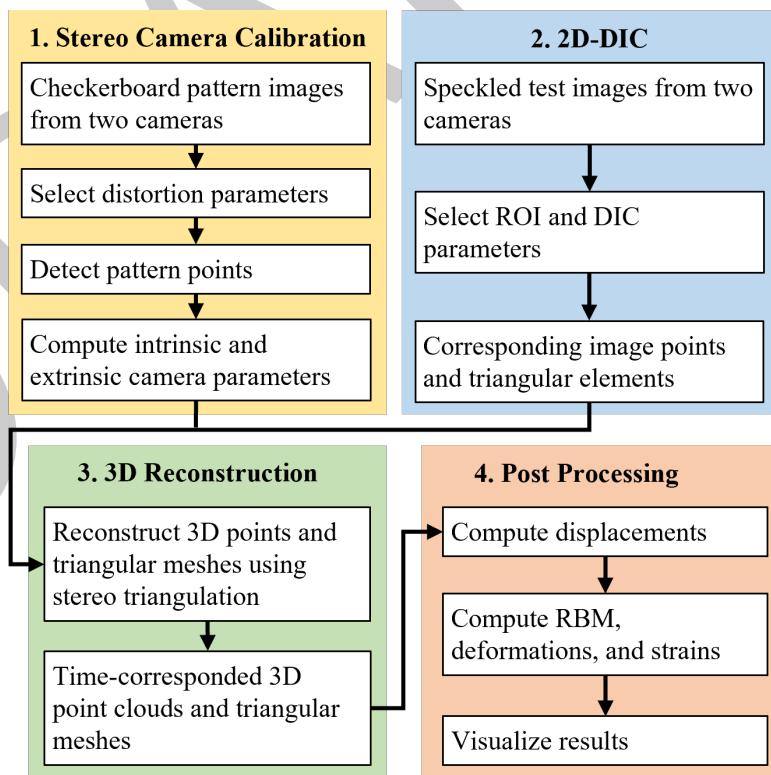


Figure 1: DuoDIC algorithm workflow. Each color represent one of the main steps of the 3D-DIC pipelines, which is executed in one of the four main scripts.

63 Step 1: Stereo camera calibration

64 In this step, the intrinsic and extrinsic parameters of both cameras are computed by integrating
 65 functions from MathWorks Computer Vision Toolbox, which implement algorithms by Zhang
 66 ([Zhang, 2000](#)), Heikkila and Silven ([Heikkila & Silven, 1997](#)), Bouguet ([Bouguet, 2013](#)),
 67 and Bradski and Kaehler ([Bradski & Kaehler, 2008](#)). This script takes as input multiple
 68 simultaneous images of a checkerboard target captured by two cameras in a stereo pair. The
 69 checkerboard pattern points in each image are automatically detected and used for calibrating
 70 the cameras' intrinsic and extrinsic parameters. The intrinsic parameters include focal lengths,
 71 principal point (optical center), and up to 6 distortion coefficients. The user can choose
 72 between [0,2,3] radial distortion coefficients, [0,2] tangential distortion coefficients, and [0,1]
 73 skew parameter. The extrinsic (stereo) camera parameters comprise the 3D position and
 74 orientation of the second camera with respect to the first (a total of 6 degrees of freedom).
 75 Furthermore, the computed camera parameters are used for computing the reprojection errors,
 76 which represent the distance between the reprojected and the detected pattern points, and
 77 comprise the accuracy of the estimated camera parameters. An example of the results, which
 78 are presented at the end of this step is shown in [Figure 2](#). This calibration was obtained using
 79 52 stereo images of a pattern with a square size of 10 mm, which are included in the sample
 80 data provided with the toolbox.

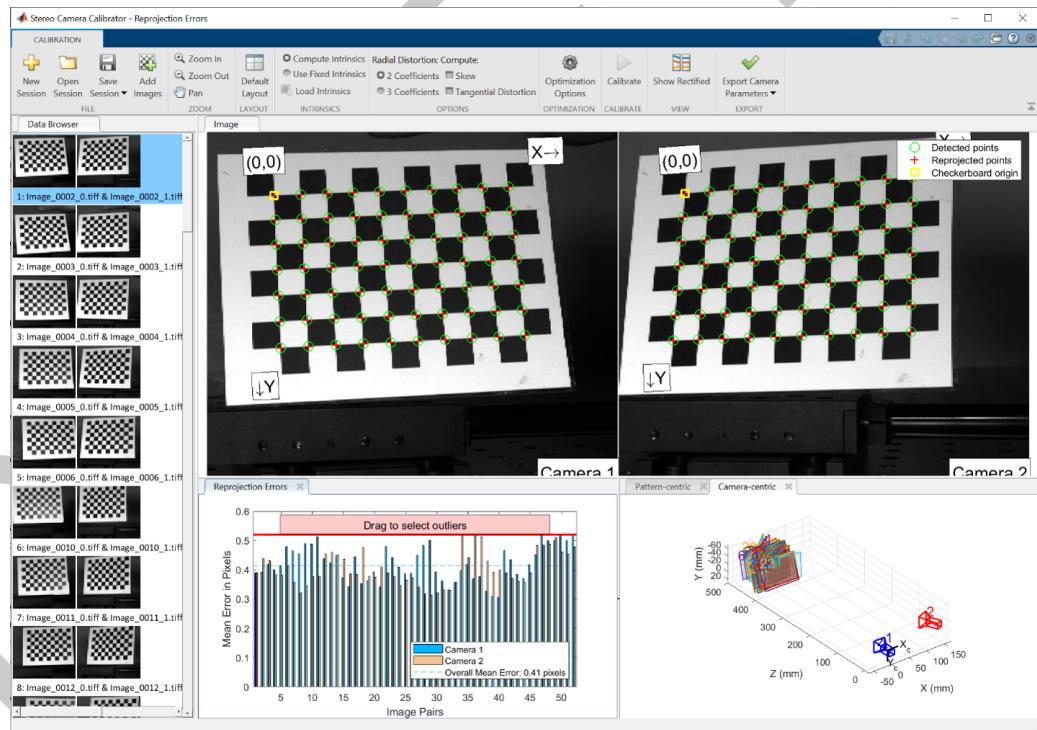


Figure 2: Stereo Calibration results. The top panels show the detected and reprojected points for a pair of stereo images. The bottom left panel shows the mean reprojection error for all pairs of images, and the bottom right panel shows the estimated positions and orientations of the cameras with respect to the checkerboard pattern images.

81 Step 2: Image cross-correlation (2D-DIC)

82 In this step, the script receives multiple images of a speckled test object captured simultaneously
 83 by the stereo camera pair. The first image pair defines the reference configuration, to which the
 84 rest of the image are compared. Typically, the reference configuration represents an unloaded,
 85 or undeformed, state and the rest of the images represent deformed states. Ncorr toolbox
 86 ([Blaber et al., 2015](#)) is utilized in this step to detect a dense grid of matching points on all

87 images. Although Ncorr was created as a 2D-DIC toolbox, typically receiving images from a
 88 single camera, DuoDIC utilizes it to detect matching points on images taken from two different
 89 views. The user can select the region of interest (ROI), the subset size and subset spacing, and
 90 then the images from both cameras in all states are cross-correlated with the reference image
 91 (first camera, first state), to detect a grid of corresponding points in the ROI, as demonstrated
 92 in [Figure 3](#). For interpreting the values of the correlation coefficients, refer to ([Blaber et al., 2015](#)) and ([Solav et al., 2018](#)). Furthermore, the point grid is meshed with triangular elements,
 93 which are then used in Step 4 for calculating the deformation and strain measures.
 94

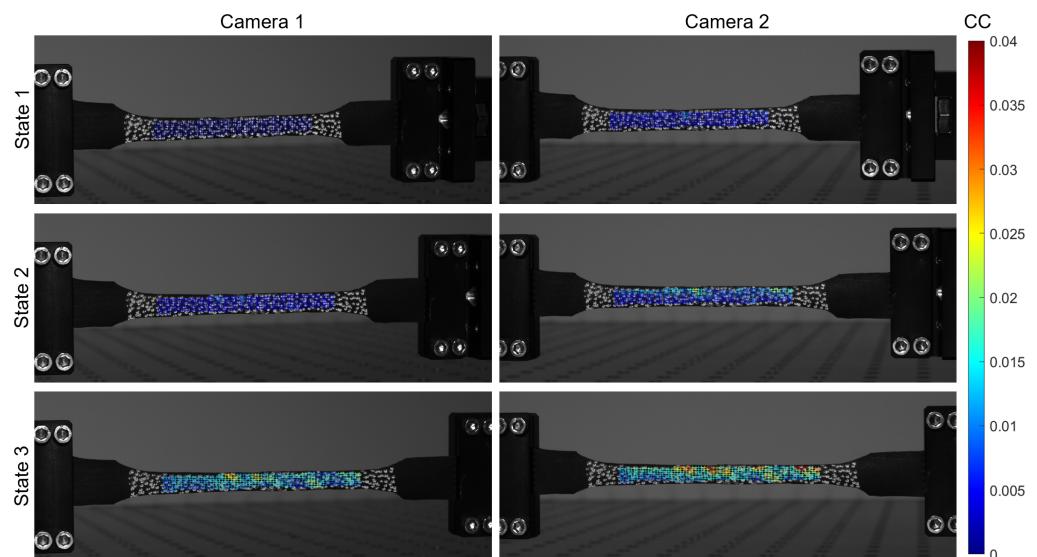


Figure 3: Image point matching computed using Ncorr and DuoDIC. The grid of corresponding matched points are plotted as crosses overlaid on each image, with the colors representing their correlation coefficient (CC) with the reference image (camera 1, state 1, top left panel).

95 Step 3: 3D reconstruction

96 The results from Step 1 and Step 2 are combined in this step to obtain the 3D position of
 97 each image point using stereo triangulation. The set of matching points on each pair of stereo
 98 images (as computed in Step 2) are first being undistorted using the intrinsic camera parameters
 99 of each camera, according to the distortion model selected in Step 1. Consequently, the image
 100 points are transformed into 3D world points using the stereo camera parameters computed
 101 in Step 1. In addition, the reprojection errors are obtained for each point, by calculating the
 102 distance (in pixels) between the detected and the reprojected points. At the end of this step,
 103 the 3D point cloud and triangulated surface are plotted and animated, utilizing function from
 104 the GIBBON toolbox ([Moerman, 2018](#)). Two animated figures plot the values of the matching
 105 correlation coefficient and the reprojection errors, as shown in [Figure 4](#) for the last image in
 106 the series, which represents the configuration with the largest stretch.

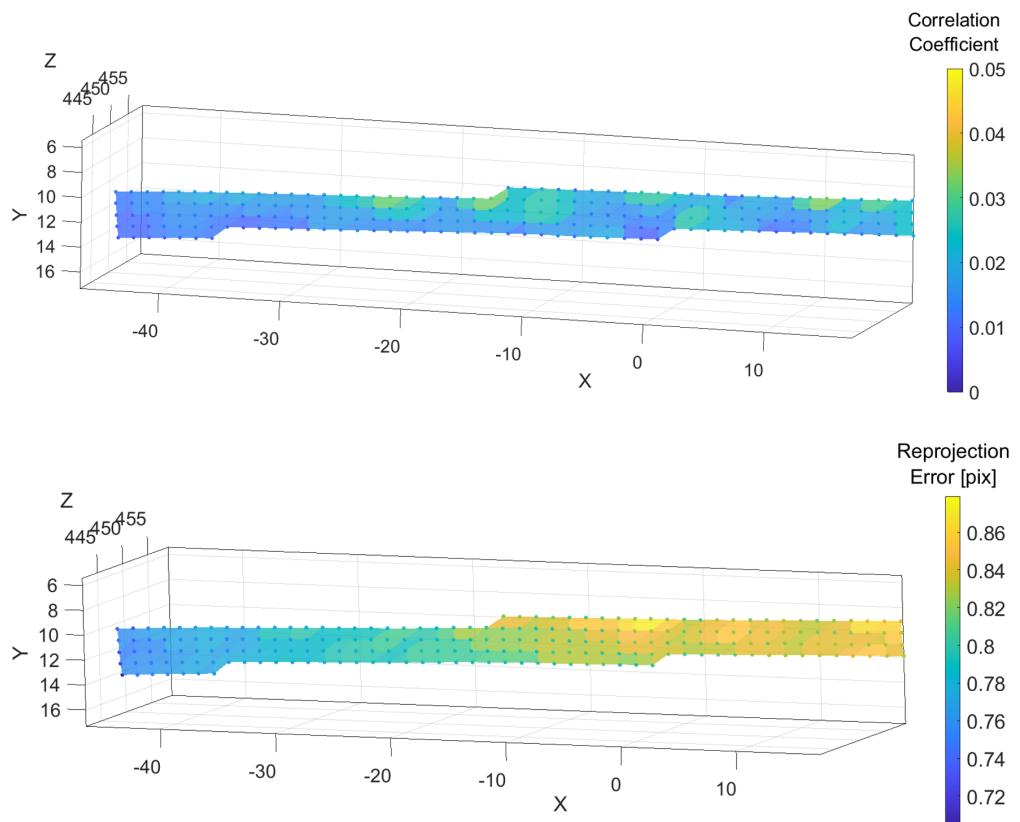


Figure 4: Step 3 results: 3D reconstruction and animated plots of the correlation coefficients and the reprojection errors.

107 **Step 4: Post processing**

108 The 3D coordinates calculated in Step 3 are used in this step to derive the full-field displacement,
 109 deformation, and strain maps. The displacements are calculated for each point, and the
 110 deformations and strains are calculated for each triangular element, using the Cosserat point
 111 element method (Solav et al., 2014, 2016; Solav, Camomilla, et al., 2017; Solav, Meric, et al.,
 112 2017). Detailed information on the post processing methods can be found in the MultiDIC
 113 paper (Solav et al., 2018). In short, for each triangular element, the position vectors of its
 114 vertices are used to calculate the deformation gradient tensor \mathbf{F} , from which the right and
 115 left Cauchy-Green deformation tensors ($\mathbf{C} = \mathbf{F}^T \mathbf{F}$ and $\mathbf{B} = \mathbf{F} \mathbf{F}^T$, respectively) are derived,
 116 as well as the Green-Lagrangian and Eulerian-Almansi strain tensors ($\mathbf{E} = 0.5(\mathbf{C} - \mathbf{I})$ and
 117 $\mathbf{e} = 0.5(\mathbf{I} - \mathbf{B}^{-1})$, respectively). The principal components and directions of these tensors are
 118 computed for deriving the principal stretches λ_i and strains E_i and e_i , as well as the equivalent
 119 (Von-Mises) strain, maximal shear strain, and area change.

120 **Figure 5** and **Figure 6** show two example figures, plotting the 3D displacement magnitudes
 121 overlaid on the images and first and second principal Lagrangian strains (magnitude and
 122 directions) on the 3D triangular mesh, respectively. Examples of animated GIF files exported
 123 using DuoDIC can be found in the GitHub repository.

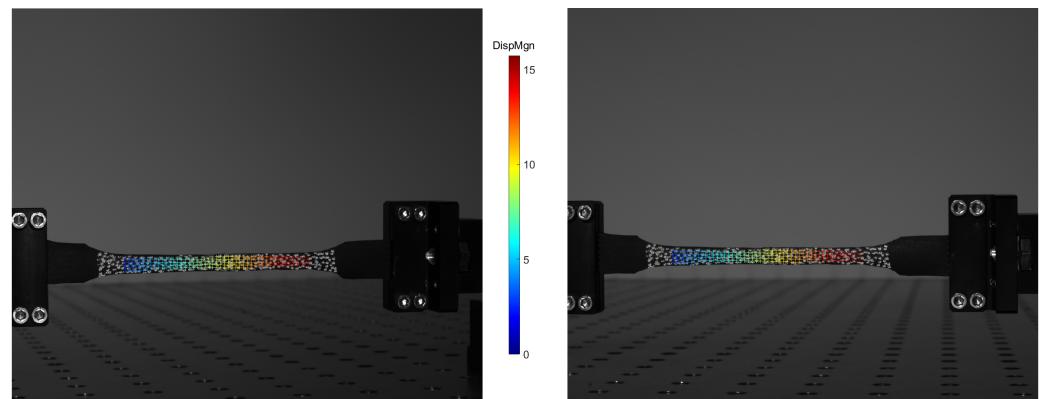


Figure 5: Full-field displacement magnitudes (in mm) overlaid on the original images.

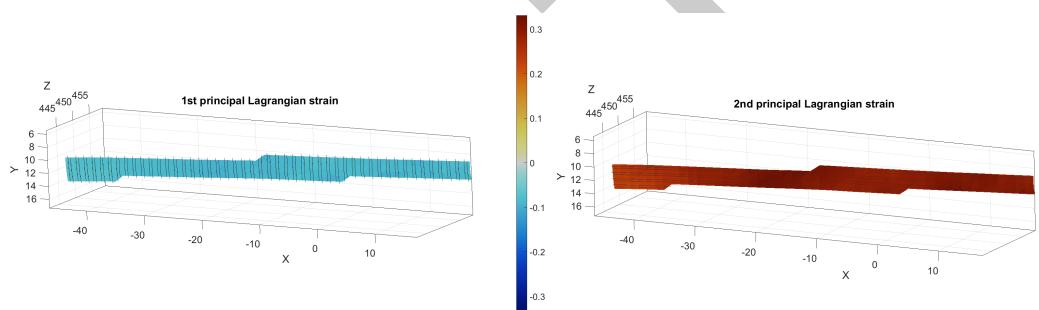


Figure 6: First (minimal) and second (maximal) principal strains plotted as 3D surfaces represented by triangular meshes, with face colors depicting the strain magnitude and the black lines depicting the strain direction.

124 Validation using a rigid body motion (RBM) test

125 To assess the metrological performance of DuoDIC, we performed a rigid body motion experiment,
 126 whereby a speckled cylinder was translated using a motorized linear translation stage (PT1/M-
 127 Z8, Thorlabs, USA), in 15 increments of 0.2 mm. simultaneous images were captured using
 128 the same camera setup described in the previous section. By comparing the displacements
 129 measured using the translation stage with those computed using DuoDIC, the displacement
 130 errors were quantified, as shown in [Figure 7](#). In addition, since strain should vanish for any RBM,
 131 any non-zero strains represent measurement errors. [Figure 8](#) plots the strains measured during
 132 the RBM translations. The results indicate that for this experimental setup, the translations
 133 were accurate to within $(1.3 \pm 1.1) \cdot 10^{-3}$ mm (mean \pm STD absolute translation error) and
 134 the strains errors were $(3.4 \pm 2.3) \cdot 10^{-4}$.

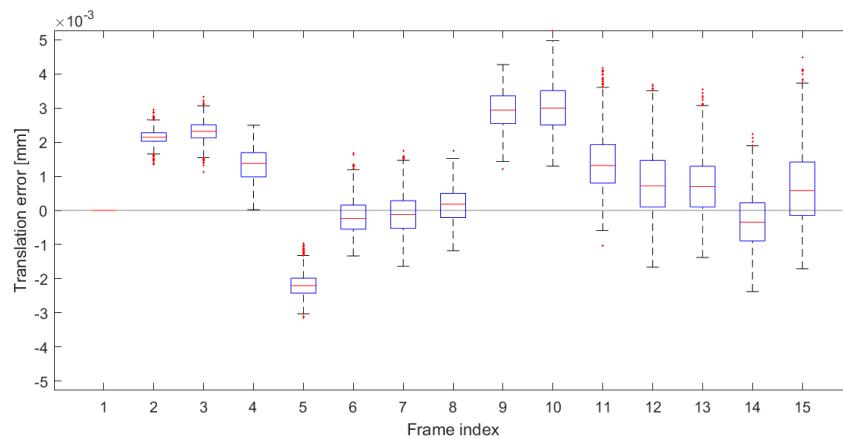


Figure 7: Translation errors representing the difference between the translations measured by the translation stage and those computed using DuoDIC. On each box, the central red line indicates the median, the bottom and top edges of the box indicate the 25th and 75th percentiles, the whiskers extend to the most extreme data points not considered outliers, and the outliers are plotted in red.

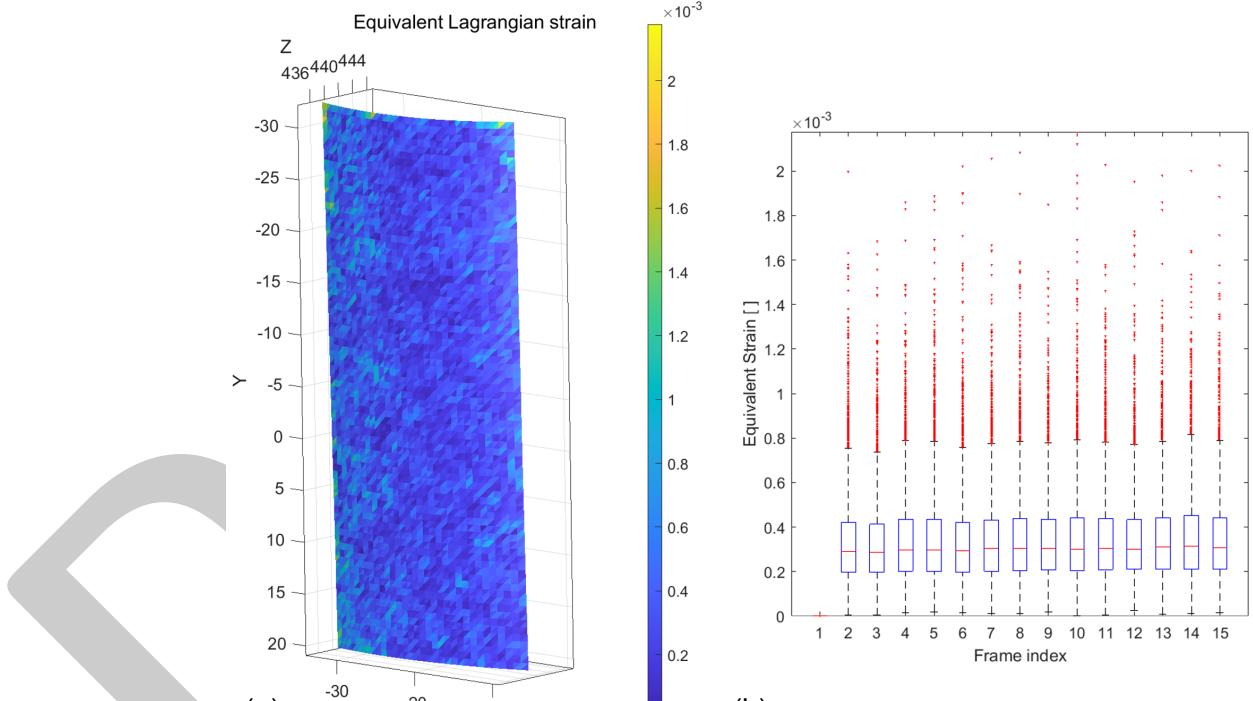


Figure 8: Strain values measurements during the RBM test represent the errors in the strain measurement. (a) 3D surface of the cylinder's ROI in the last translation step, showing the distribution of strains across the surface. (b) Statistical strain results representing all the faces at each translation step. On each box, the central red line indicates the median, the bottom and top edges of the box indicate the 25th and 75th percentiles, the whiskers extend to the most extreme data points not considered outliers, and the outliers are plotted in red.

135 Acknowledgements

136 This project was supported by Dana Solav's Jacques Lewiner Career Advancement Chair.

References

- 137
- 138 Atkinson, D., & Becker, T. H. (2021). Stereo digital image correlation in MATLAB. *Applied Sciences (Switzerland)*, 11(11), 4904. <https://doi.org/10.3390/app11114904>
- 139
- 140 Blaber, J., Adair, B., & Antoniou, A. (2015). Ncorr: Open-Source 2D Digital Image Correlation Matlab Software. *Experimental Mechanics*, 55(6), 1105–1122. <https://doi.org/10.1007/s11340-015-0009-1>
- 141
- 142
- 143 Bougue, J.-Y. (2013). *Camera Calibration Toolbox for Matlab: Calibration examples*. http://www.vision.caltech.edu/bouguetj/calib_doc/htmls/example.html
- 144
- 145 Bradski, G., & Kaehler, A. (2008). *Learning OpenCV* (First Edit). O'Reilly Media, Inc. ISBN: 978-0-596-51613-0
- 146
- 147 Heikkila, J., & Silven, O. (1997). A four-step camera calibration procedure with implicit image correction. *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1106–1112. <https://doi.org/10.1109/CVPR.1997.609468>
- 148
- 149
- 150 Moerman, K. M. (2018). GIBBON: The Geometry and Image-Based Bioengineering add-On. *The Journal of Open Source Software*, 3(22), 506. <https://doi.org/10.21105/joss.00506>
- 151
- 152 Pan, B. (2018). Digital image correlation for surface deformation measurement: Historical developments, recent advances and future goals. *Measurement Science and Technology*, 29(8). <https://doi.org/10.1088/1361-6501/aac55b>
- 153
- 154
- 155 Reu, P. (2015). THE ART AND APPLICATION OF DIC DIC : A Revolution in Experimental Mechanics. *Experimental Techniques*, 39(Dic), 1–2. <https://onlinelibrary.wiley.com/doi/pdf/10.1111/ext.12173>
- 156
- 157
- 158 Solav, D., Camomilla, V., Cereatti, A., Barré, A., Aminian, K., & Wolf, A. (2017). Bone orientation and position estimation errors using Cosserat point elements and least squares methods: Application to gait. *Journal of Biomechanics*, 62, 110–116. <https://doi.org/10.1016/j.jbiomech.2017.01.026>
- 159
- 160
- 161
- 162 Solav, D., Meric, H., Rubin, M. B., Pradon, D., Lofaso, F., & Wolf, A. (2017). Chest Wall Kinematics Using Triangular Cosserat Point Elements in Healthy and Neuromuscular Subjects. *Annals of Biomedical Engineering*, 45(8), 1963–1973. <https://doi.org/10.1007/s10439-017-1840-6>
- 163
- 164
- 165
- 166 Solav, D., Moerman, K. M., Jaeger, A. M., Genovese, K., & Herr, H. M. (2018). MultiDIC: an Open-Source Toolbox for Multi-View 3D Digital Image Correlation. *IEEE Access*, 6, 30520–30535. <https://doi.org/10.1109/ACCESS.2018.2843725>
- 167
- 168
- 169 Solav, D., Rubin, M. B. B., Cereatti, A., Camomilla, V., & Wolf, A. (2016). Bone Pose Estimation in the Presence of Soft Tissue Artifact Using Triangular Cosserat Point Elements. *Annals of Biomedical Engineering*, 44(4), 1181–1190. <https://doi.org/10.1007/s10439-015-1384-6>
- 170
- 171
- 172
- 173 Solav, D., Rubin, M. B. B., & Wolf, A. (2014). Soft Tissue Artifact compensation using Triangular Cosserat Point Elements (TCPES). *International Journal of Engineering Science*, 85, 1–9. <https://doi.org/10.1016/j.ijengsci.2014.07.001>
- 174
- 175
- 176 Sutton, M. A., Matta, F., Rizos, D., Ghorbani, R., Rajan, S., Mollenhauer, D. H., Schreier, H. W., & Lasprilla, A. O. (2017). Recent Progress in Digital Image Correlation: Background and Developments since the 2013 W M Murray Lecture. *Experimental Mechanics*, 57(1), 1–30. <https://doi.org/10.1007/s11340-016-0233-3>
- 177
- 178
- 179
- 180 Turner, D. (2015). *Digital Image Correlation Engine (DICe) Reference Manual*. Sandia Report. <https://github.com/dicengine/>
- 181

- ¹⁸² Zhang, Z. (2000). A Flexible New Technique for Camera Calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11), 1330–1334. <https://doi.org/10.1109/34.888718>

DRAFT