SAND2020-8808J

Experimental Techniques manuscript No.
(will be inserted by the editor)

Predicting 3D Motions from Single-Camera Optical Test Data

Daniel P. Rohe · Bryan L. Witt · Tyler F. Schoenherr

Received: date / Accepted: date

Abstract In a typical optical test, a stereo camera pair is required to measure the three-dimensional motion of a test article; one camera typically only measures motions in the image plane of the camera, and measurements in the out-of-plane direction are missing. Finite element expansion techniques provide a path to estimate responses from a test at unmeasured degrees of freedom. Treating the case of a single camera as a measurement with unmeasured degrees of freedom, a finite element model is used to expand to the missing third dimension of the image data, allowing a full-field, three-dimensional measurement to be obtained from a set of images from a single camera. The key to this technique relies on the mapping of finite element deformations to image deformations, creating a set of mode shape images that are used to filter the response in the image into modal responses. These modal responses are then applied to the finite element model to estimate physical responses at all finite

---

D. P. Rohe
Sandia National Laboratories*
P.O. Box 5800 - MS0557
Albuquerque, NM 87123
E-mail: dprohe@sandia.gov

B. L. Witt
Sandia National Laboratories

T. F. Schoenherr
Sandia National Laboratories

*Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

element model degrees of freedom. The mapping from finite element model to image is achieved using synthetic images produced by a rendering software. The technique is applied first to a synthetic deformation image, and then is validated using an experimental set of images.

Keywords Optical · Finite Element Expansion · Synthetic Images · Structural Dynamics

1 Introduction

Optical techniques are becoming popular due to the full-field, three-dimensional motions that they can obtain on the surfaces of structures. Traditionally, when 3D motions are desired in a structural dynamics test, a stereo pair of cameras is required to obtain the measurement. A single camera is only able to identify motions in the image plane, and any motions perpendicular to these directions are not captured. In many situations, only one camera may be present during an event, but it would be useful to be able to estimate 3D motions of a structure. Considering the example of a building during an earthquake: there may be cameras throughout the city that have a view of this building, but likely not a well-calibrated stereo pair. Still, engineers would be interested in the deformations of that building to determine if any damage occurred.

Knowledge of the test article can be used to supplement the lack of out-of-plane measurements in a single camera measurement system. If a plate is set in front of a single camera system and moved towards the camera, it will appear to grow larger in the camera image; in this case, a single camera cannot distinguish between out-of-plane motion and expansion of the part. However, given knowledge of the test article, it is possible to infer what these ambiguous motions typically mean. In this example, a plate very rarely grows uniformly in size due to a typical structural dynamic loading; it seems much more reasonable that the plate would be moving towards the camera. Applying knowledge of the physics of the test article in order to predict unmeasured responses is precisely the goal of finite element expansion techniques. In these techniques, there is some set of measured data, and from that data, unmeasured degrees of freedom can be predicted using a model. This work aims to apply finite element expansion techniques, particularly the System Equivalent Reduction Expansion Process (SEREP) technique [1], to expand to full-field, three-dimensional motions from a set of images from a single camera.

2 Finite Element Expansion using the SEREP Technique

SEREP as an expansion technique is essentially a spatial filter, using finite element shapes as the basis for the filter. A given response $\mathbf{x}_a$ on the part is measured at some a-set of degrees of freedom, and the responses are filtered by the finite element mode shape matrix $\mathbf{\Phi}_a$ partitioned to only those a-set degrees of freedom (using the pseudoinverse $^+$ to perform the filtering). This results in coefficients $\mathbf{q}$ corresponding to a linear combination of the finite element shapes active in the measured response, which can be multiplied by the full n-space finite element shapes $\mathbf{\Phi}_n$ to recover estimates of responses in the full finite element space $\mathbf{x}_n$.

$$\mathbf{q} = \mathbf{\Phi}_a^+ \mathbf{x}_a \tag{1}$$

$$\mathbf{x}_n = \mathbf{\Phi}_n \mathbf{q} \tag{2}$$

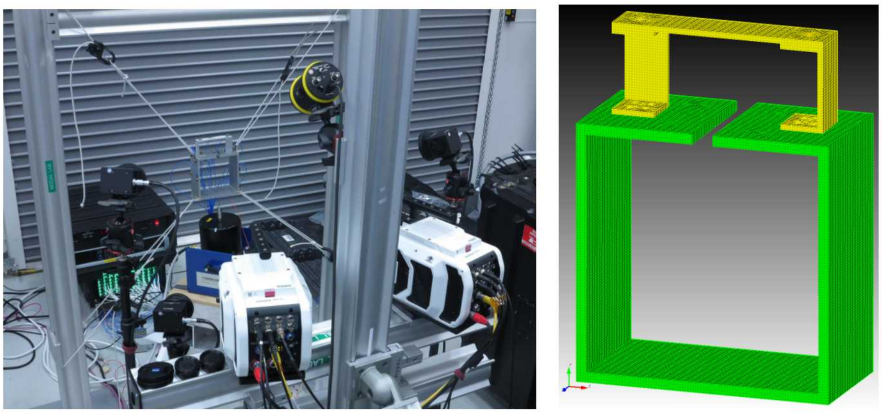
Combining these two equations gives the typical SEREP expansion equation

$$\mathbf{x}_n = \mathbf{\Phi}_n \mathbf{\Phi}_a^+ \mathbf{x}_a \tag{3}$$

What is not obvious in equation (3) is that the a- or n-set degrees of freedom need not correspond to physical displacements, they can instead correspond to any generalized degrees of freedom that correlate to the mode shapes. For example, in [2], the authors use SEREP to expand from translation degrees of freedom to full-field strain derived from a linear combination of strain shapes from the finite element model. In other words, if the same degrees of freedom that are measured during the test can be extracted or derived from the finite element shapes, and these finite element degrees of freedom define a unique linear combination of shapes to reproduce the test degrees of freedom, the SEREP technique can be used to expand those test degrees of freedom to full finite element space. Therefore, in order to use SEREP to expand to full-field, 3D deformations of the test article $\mathbf{x}_n$, the terms on the right-hand side of equation (3) must be developed.

The Box Assembly with Removable Component (BARC) [3], shown in Figure 1a, was used as a test article for this work. Contrast was applied using a computer-generated speckle pattern printed to label paper, which was adhered to the camera-facing surfaces of the structure. A finite element model was created of the test article, shown in Figure 1b, and used in the expansion process. The finite element deformations for each mode shape can be used directly to populate columns of the $\mathbf{\Phi}_n$ matrix. The $\mathbf{\Phi}_a$ degrees of freedom will require additional processing to compute, as they will depend on the $\mathbf{x}_a$ degrees of freedom extracted from the images.

To define the $\mathbf{x}_a$ degrees of freedom, a camera view of the test article was needed. Details of the experimental stereo camera setup are provided in Table 1. The stereo camera setup was necessary to produce 3D "truth data" against which to compare the expanded data; to perform the expansion to 3D motions, only the images from the left camera were used. Figure 2 shows an example test image. The degrees of freedom extracted from the left camera images will represent the $\mathbf{x}_a$ degrees of freedom. The final part of the equation, $\mathbf{\Phi}_a$, is obtained by transforming the finite element displacements into image degrees of freedom. This is an involved process that is described in Section 3.



(a) BARC test article set up for optical testing. (b) BARC finite element model.

Fig. 1: BARC test article and finite element model.

Table 1: Stereo Camera Parameters

Parameter	Value
Camera	Vision Research Phantom v2640
Lens	Zeiss Milvus 85mm
Aperture	f/11
Image Resolution	1536×1920 pixels
Field of View	15×20 cm
Image Scale	8 pixel/mm
Stereo-angle	30 degrees
Stand-off Distance	80 cm
Image Acquisition Rate	4096 fps
Patterning Technique	Printed Sticker
Approximate Speckle Size	5 pixels

3 Creating Synthetic Images using a Finite Element Model

In order to expand from a measured *a*-set of degrees of freedom using SEREP, those same *a*-set degrees of freedom must be extracted from the finite element model for each mode. Therefore, to expand from arbitrary image degrees of freedom, each of the finite element shapes should be reproduced on-image so that the same degrees of freedom can be extracted. This process involves determining the intrinsic and extrinsic camera parameters in the finite element coordinate system, mapping the image texture to the finite element model, deforming the finite element model into a mode shape, and then rendering an image of each shape. This is performed using the process developed in [4] using the open-source Blender software [5] to render the final deformed images. The image processing is then performed on the synthetic images, using the identical technique to the actual response images, to generate the finite element *a*-set degrees of freedom necessary to evaluate Equation (3).

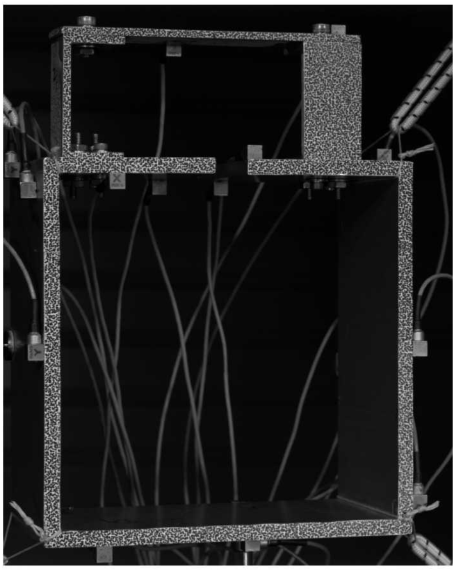


Fig. 2: Representative response image from the test.

3.1 Preparing the Finite Element Model

The first step is to prepare the finite element model for import into the Blender software. The Blender software uses surface meshes to model geometry, so the volumetric finite element mesh needs to be “skinned” to produce a surface mesh. This can be performed by creating a list of all faces of all volumetric elements in the model, and removing any face that appears twice: exterior faces will only appear in one volume element, whereas internal faces will always appear in two. These exterior faces are used to create a connectivity array that can be assembled in Blender using its Python programming interface.

3.2 Camera Setup

Once the finite element model is loaded into Blender, a camera must be placed so that there is a view point from which to render the scene. Blender uses a pinhole camera model that is widely used in computer graphics applications, where the equation that projects three-dimensional spatial (x, y, z) coordinates into the two-dimensional image (u, v) coordinates is

$$\begin{bmatrix} cu \\ cv \\ c \end{bmatrix} = [K]\,[R|t] \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{4}$$

Note that equation (4) is constructed using homogeneous coordinates, so the left hand side may be scaled by an arbitrary factor c, which is the last row of the resulting vector. Dividing the first and second rows of the left hand side by c (i.e. scaling the vector so the last row is 1) allows recovery of the image pixel coordinates u and v.

Here, $[K]$ is a 3×3 matrix of camera intrinsic parameters, which is developed from physical camera parameters as

$$[K] = \begin{bmatrix} \frac{p_u f_{mm}}{ss_{u,mm}} & s & (1/2 - s_u)p_u \\ 0 & \frac{p_v f_{mm}}{ss_{v,mm}} & (1/2 - s_v)p_v \\ 0 & 0 & 1 \end{bmatrix} \tag{5}$$

where p_u and p_v are the image resolution in pixels in the u and v dimensions, respectively, $ss_{u,mm}$ and $ss_{v,mm}$ are the camera sensor size in millimeters in the u and v dimensions, respectively, s_u and s_v are the normalized camera shift values in fraction of the image in the u and v dimensions, respectively, and f_{mm} is the lens focal length in millimeters. All of these camera parameters are specified in the Blender software. Some camera calibration routines return a camera skew s or a separate focal length f_{mm} for each of the u and v directions; however Blender's camera model cannot handle these parameters. If a significant skew or difference in focal length between dimensions is needed for the camera and lens combination used in the test, these distortions can be applied via postprocessing the output images from Blender.

$[R|t]$ is a 3×4 matrix of extrinsic camera parameters. The left partition $[R]$ is a 3×3 rotation matrix, and the right partition $[t]$ is a 3×1 vector of translations. These transformations take vectors specified in the global (finite element) coordinate system and transform them into the local camera coordinate system where the camera z direction is defined as the view axis of the camera. However, the camera rotation and position in the global coordinate system need to be specified in Blender. One important detail is that the local z directions of camera objects in Blender point *opposite* of the view axis, so an additional rotation $[R]_c$ must be specified that performs a 180 degree rotation about the local camera x axis.

$$[R]_c = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{6}$$

$$[R]_b = [R]^T [R]_c \tag{7}$$

$$[t]_b = -[R]_b [R]_c^T [t] \tag{8}$$

The matrices $[R]_b$ and $[t]_b$ are then the orientation and position of the camera in the Blender software. The orientation of an object must be specified using Quaternions or Euler Angles, so the orientation matrix $[R]_b$ must be transformed into one of those forms.

The intrinsic parameters of the test camera can be derived using a standard camera calibration that is common in DIC: a calibration target with a dot

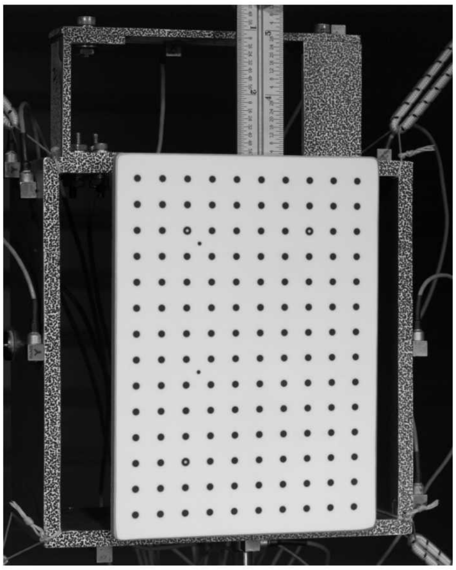


Fig. 3: Representative calibration image showing dot grid calibration target.

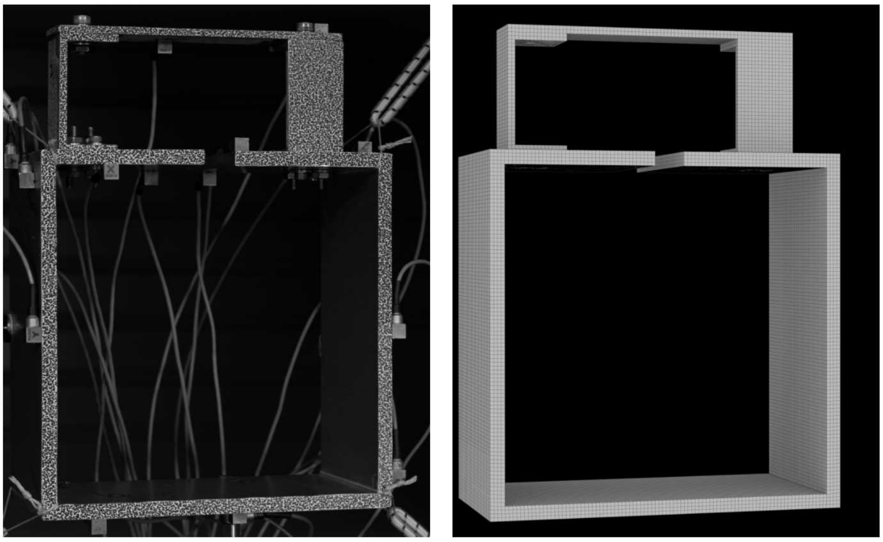
grid with known spacing is translated and rotated in front of the camera while a series of images are captured. Figure 3 shows a representative calibration image used for this test. Correlated Solutions' VIC3D software was used to perform the calibration, and the $[K]$ matrix was constructed from its output. Lens distortion parameters are also obtained from the calibration procedure. While Blender's pinhole camera model does not include distortions, the Open Computer Vision (OpenCV) [6] library has functions that can be used to remove distortion from test images to ensure that they are consistent with Blender's output.

For a stereo DIC setup, the camera calibration described above will typically provide extrinsic camera parameters as well; however, with only one camera only the intrinsic parameters can be derived, and a separate approach is required to compute the camera pose. Here the Efficient Perspective-n-Point (EPnP) algorithm [7] was used, as implemented in the OpenCV library [6]. The EPnP algorithm takes as its inputs the camera intrinsic matrix $[K]$ and distortion parameters, as well as a series of point pairs in the image and spatial domains. Table 2 shows the point pairs used for this pose estimation. The output of the algorithm is the orientation and position of the camera, from which the matrix $[R|t]$ was derived. To check the accuracy of the complete camera projection, the 3D coordinates in Table 2 can be reprojected through equation (4) to produce estimated pixel locations. These can be compared to the 2D image locations in Table 2 to compute a reprojection error. In this case, the average error was 2.5 pixels, which is approximately 0.3 mm.

Once the intrinsic and extrinsic camera matrices are determined, a camera can be created in the Blender software using physical camera parameters from

Table 2: Point pairs used in the EPnP algorithm

Image Position		Spatial Position			Reprojection
u (px)	v (px)	x (in)	y (in)	z (in)	Error (px)
201	74	2.5	2.128	-0.5	0.8
1220	105	-2.5	2.128	-0.5	3.6
136	523	2.99	0.0	-1.5	2.4
1398	536	-2.99	0.0	-1.5	3.9
137	1860	2.99	-5.98	-1.5	2.3
1395	1814	-2.99	-5.98	-1.5	2.1



(a) Test Image **(b)** Rendered Finite Element Model

Fig. 4: Comparison of test image and finite element model projected to test image.

equations (5) and (7)–(8). A render of the finite element model from this camera compared to the test image appears in Figure 4.

3.3 Local Texture and Lighting

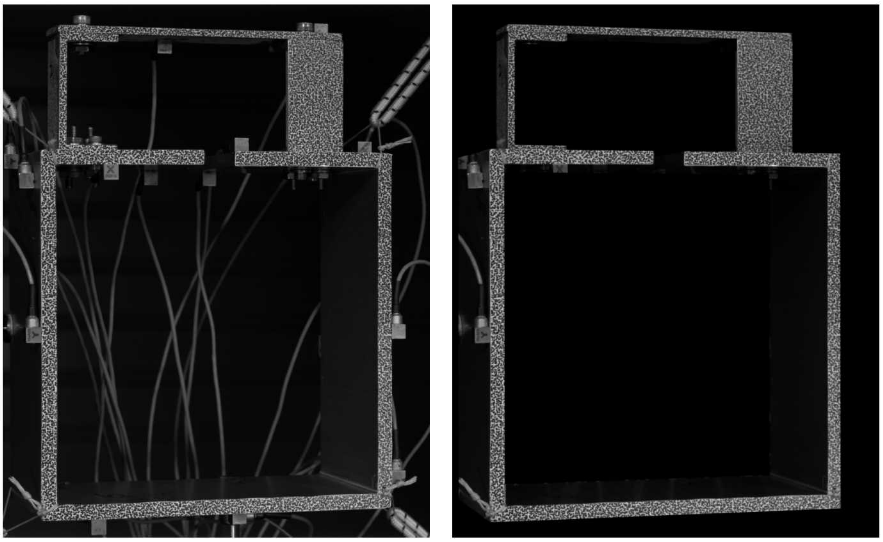
The next step to matching the test images is to apply a texture to the finite element model. This texture will be mapped to the undeformed finite element model; when the model is deformed, the texture will deform with it. If tracking subsets using DIC, it is not strictly necessary to reproduce the exact image from the test on the finite element model as long as the subset locations are chosen identically. However, for other optical techniques such as feature tracking, marker tracking, or phase-based processing, it will be necessary to have identical local contrast patterns to ensure the same degrees of freedom are being measured. Therefore, this work will proceed with mapping the test image onto the finite element model.

The general problem of texture mapping in computer graphics involves the mapping of a 2D image to a 3D model. One common approach, known as UV mapping in computer graphics applications, involves “unwrapping” or projecting the 3D mesh to 2D (u, v) image coordinates. Each face of the geometry mesh is mapped to a portion of the image, which is applied to the 3D mesh during the render. Note that the projection from 3D model to test image space is already defined from the camera projection developed in Section 3.2, so if the texture to be applied to the finite element model is the test image itself, the UV mapping is already defined. Blender performs this entire process in a few mouse clicks; the test reference image is loaded into the software as a new material and assigned to the geometry, the view in the software is set to the camera view, and the model is unwrapped via projecting from the current view.

The texture of the object is only half of what determines its final appearance in a render. The other half is the lighting applied to the texture. One may be tempted to try to reproduce the lighting that was used in the test, perhaps using spotlights positioned in approximately the same places pointing at the test article; however, this is not correct. The lighting, shadows, and specularities in the test image are already embedded into the texture applied to the model; therefore, only a flat-white uniform light is required. This is achieved via environmental lighting in the Blender software. Note that depending on which render engine is being used in the Blender software, the background color of the render may also change with the applied environment lighting, so it may be necessary to put a flat-black plane behind the test article in the Blender scene to accurately reproduce the image. Figure 5 shows a comparison between the test image and the rendered textured finite element image.

3.4 Deforming and Rendering Images

With the finite element mesh textured, it can be arbitrarily deformed within the scene and images can be rendered of those deformations. The deformations that will be rendered will be those of the finite element mode shapes, and one must chose at what scale to render these shapes in the images. If the rendered displacements are too small, the image degrees of freedom will be noisy, which would result in a poor inversion in equation (3). However, if displacements are too large, image processing techniques could break down. At large deflections, hidden surfaces may be revealed, and because the texture was applied from the test image, surfaces that are not visible in the test image will not be textured correctly. Also, for subset or feature/marker tracking approaches, the tracked set of pixels may become distorted due to large rotations or shearing. Taking these constraints into consideration, a peak displacement of 0.5 pixels was chosen, which resulted in a different scaling factor for each mode of the system (see Section 5 for a comparison of results using



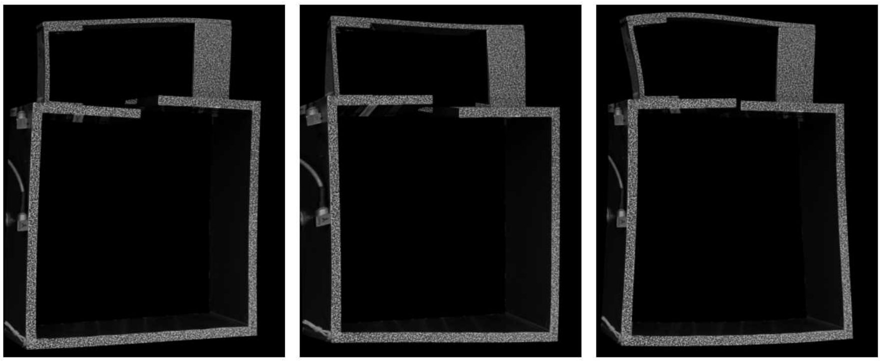
(a) Test Image (b) Rendered Finite Element Model

Fig. 5: Comparison of test image and finite element model projected to test image.

different peak displacement values). To determine each mode's scaling factor, the undeformed and deformed 3D coordinates of the finite element model were passed through equation (4), resulting in pixel locations corresponding to the undeformed and deformed model for each mode shape. Subtracting the deformed from the undeformed pixel locations gives pixel displacements for each shape. The mode shape was then scaled by the reciprocal of the maximum pixel displacement times the desired maximum pixel displacement. Note that equation (4) is not linear due to the normalization that must occur in the third row of the equations to maintain the homogeneous coordinates. Therefore, in order for this approach to work, the mode shapes were first scaled to a small value in order to generate an approximately linear relationship between 3D and image displacements, then scaled to the linear value. Figure 6 shows the first three elastic modes of the finite element model; these are rendered with deformations set to 100× the deformation used in the analysis to ensure the deformations are visible to the reader.

4 Choosing Image Degrees of Freedom

Once the finite element shapes were rendered, degrees of freedom from the images were selected. Ideally these degrees of freedom will result in linearly independent columns of the $\mathbf{\Phi}_a$ matrix so the pseudoinverse in equation (3) can be accurately performed. In this case, DIC subset displacements were chosen as the degrees of freedom to correlate. Because only one camera image



(a) Mode 7 **(b)** Mode 8 **(c)** Mode 9

Fig. 6: Synthetic images created from deforming mode shapes. Note that the deformations are scaled 100× compared to the half-pixel displacement images used in the analysis in order to make them visible to the reader.

was available, 2D DIC was performed, resulting in u and v displacements in pixels for each subset solved. The open-source Digital Image Correlation Engine (DICe) software [8] was used to perform the DIC analysis. Areas of interest were defined on the speckle pattern as shown in Figure 7. Subset and step sizes of 21 and 20 pixels, respectively, were used for the analysis. The shape functions used in the correlation allowed for translation, rotation, normal stretch, and shear stretch. These parameters resulted in 814 subsets solved with two degrees of freedom per subset for each of the rendered mode shape images. The first 26 modes (up to 3,000 Hz) were solved, which included the six rigid body modes. Figure 9 shows the subset displacements for each mode, scaled so they are visible. A Modal Assurance Criterion (MAC)[9] matrix is shown in Figure 8. The 1628×26 $\mathbf{\Phi}_a$ matrix had a condition number of 148.

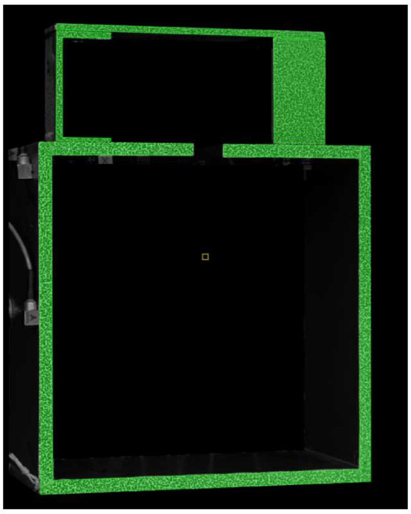


Fig. 7: Areas of Interest and Subset Size defined in DICe.

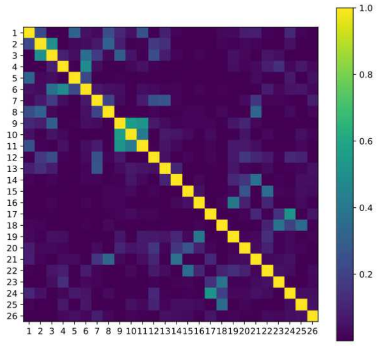


Fig. 8: MAC of the $\mathbf{\Phi}_a$ matrix.

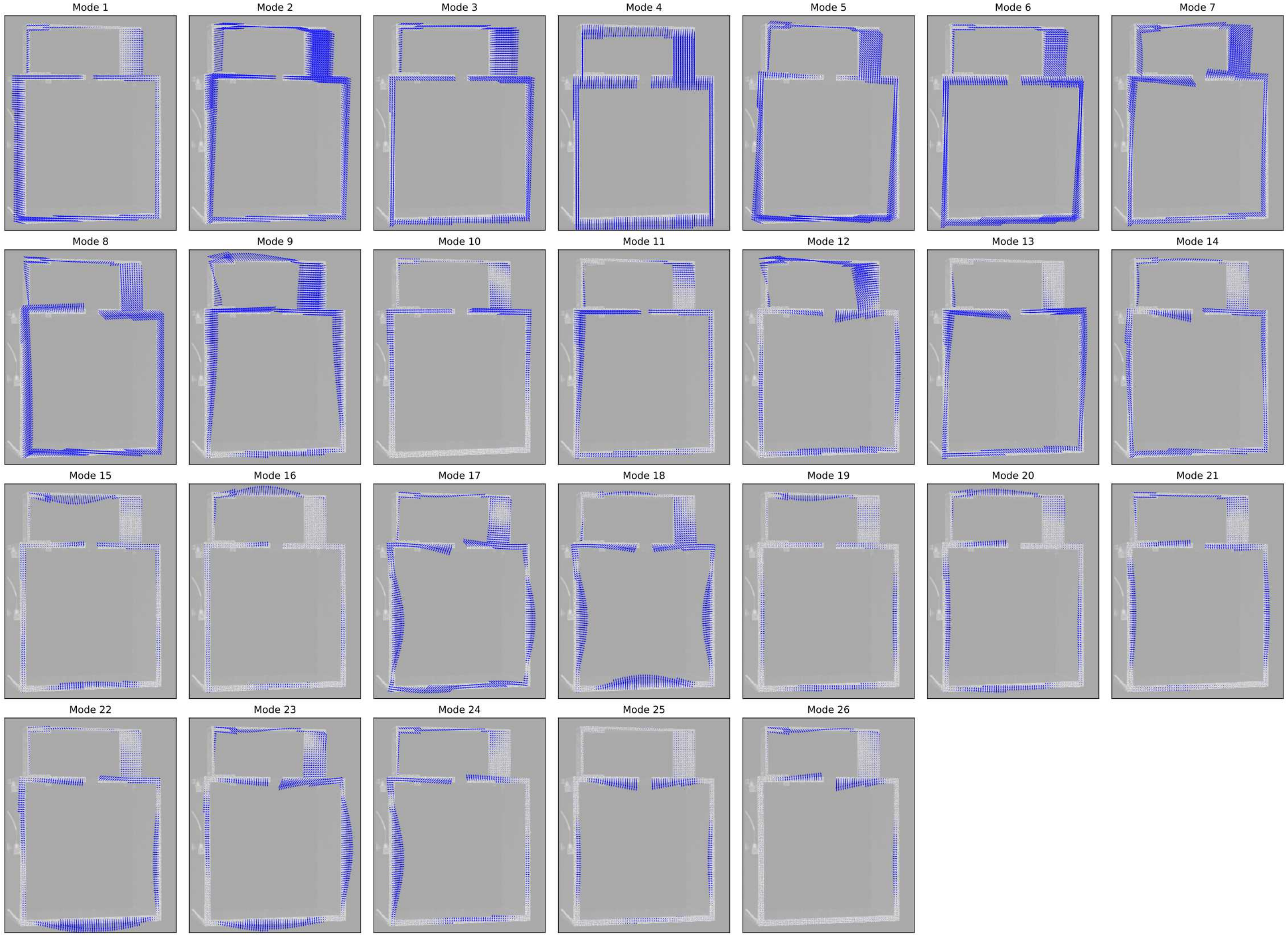
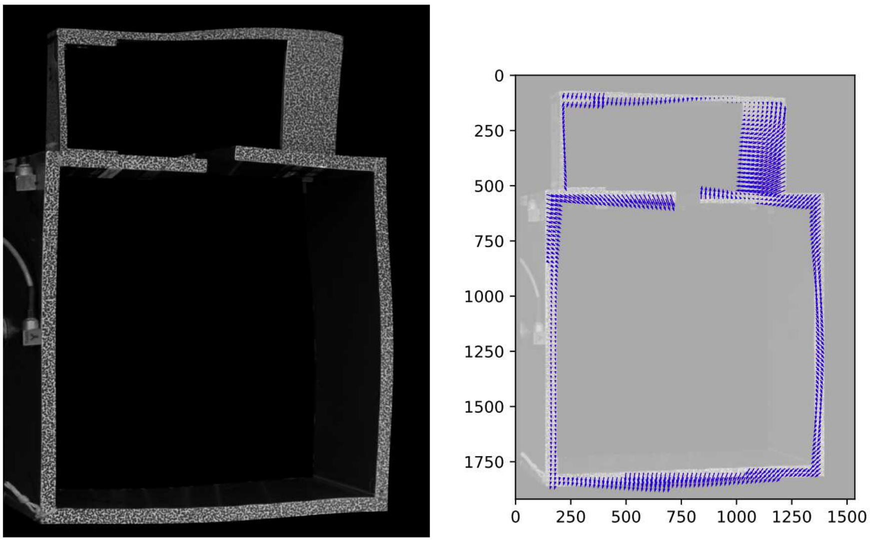


Fig. 9: Subset displacements for each mode shape.



(a) Analytic deformation image scaled 50x so the deformations are visible to the reader.

(b) Solved subset displacements for the synthetic deformation.

Fig. 10: Analytic deformations on which the technique will be demonstrated.

5 Application to Synthetic Deformation

To demonstrate the technique on perfect data, a synthetic deformation image was created using a random linear combination of modal coefficients $\mathbf{q}$ for the first 20 modes, sampled from a uniform distribution between -0.3 and 0.3. These coefficients were multiplied by unit-pixel scaled mode shapes to produce deflections that would be on the order of 0.3 pixels per mode. The synthetic deformation was then created by summing these shapes. Figure 10 shows the analytic displacement, with Figure 10a showing the synthetic render of the deformations at 50× scale and Figure 10b showing the solved subset displacements on the image. These subsets were identical to the subsets used in Section 4.

The deformed subset displacements from the analytic deflection image (Figure 10b) and the subset displacement “shapes” (Figure 9) were passed into equation (1) to solve for the shape coefficients of the displacement. Note that 26 modes were used to predict the response, even though only the first 20 modes were used to create the analytic deflection shape, so the predicted coefficients for the last 6 modes were expected to be zero. These predicted coefficients were compared directly to the prescribed modal coefficients used to create the deflection image. This comparison is shown in Figure 11. The true displacements were then compared to the prescribed displacements by passing the prescribed and estimated modal coefficients $\mathbf{q}$ through equation (2) to predict full-field response. This comparison is shown in Figure 12.

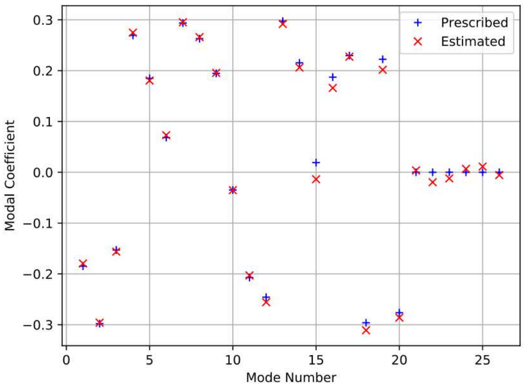


Fig. 11: Comparison of Prescribed and Estimated modal coefficients.

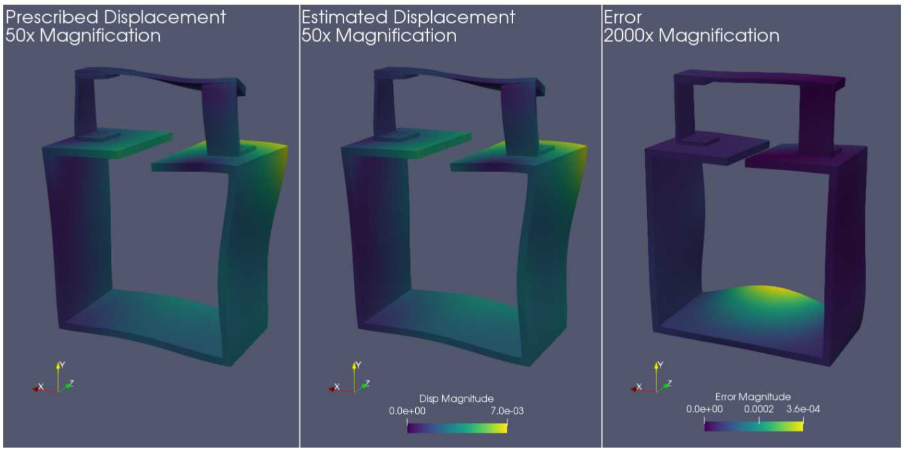


Fig. 12: Comparison of prescribed and estimated full field shapes, including the error between the two shapes.

Treating the coefficients, $\mathbf{q}$, as a vector, the magnitude of the error vector was computed to be 0.058, which was 5.8% of the magnitude of the displacement vector, 0.996. For responses, the magnitude of the error vector was only 3.0% of the magnitude of the response vector, and as seen in Figure 12, the majority of the error is located at points far from the surfaces where the subsets were solved. This gave confidence that the technique would be able to be applied to experimental data.

Because the analytical deformations applied during this analysis were known, it allowed the processing parameters to be investigated to determine which might provide a better estimate of the modal coefficients, and in turn the 3D displacements. The maximum pixel displacement of the mode shape images was of interest as it had been selected using intuition as discussed in Section

Table 3: Magnitude of the Modal Coefficient Error Vector for given maximum pixel displacement values.

Pixel Displacement	Error Magnitude
3	0.1252
1	0.0725
0.5	0.0580
0.25	0.1422
0.125	0.2010

3.4. Mode shape images were rendered at maximum deflections of 3, 1, 0.5, 0.25, and 0.125 pixel. Table 3 shows the magnitude of the error over these maximum pixel displacement values. A maximum mode shape pixel displacement of 0.5 was found to give the lowest error of the values investigated, so it was used to extract 3D motions from the test data.

6 Experimental Results

Given the favorable results using synthetic deformation images, the process was demonstrated using experimental data. The BARC was excited with a flat-force (versus a typical flat-voltage) pseudorandom input at 1 lb RMS from 100 to 2000 Hz. The flat-force input was created by measuring the transfer function between the shaker excitation force and amplifier input voltage in the test configuration and computing a shaped voltage spectrum that would result in a flat force spectrum. This voltage spectrum was then translated into a pseudorandom time history that could be played using the arbitrary source capabilities of the B+K LAN-XI data acquisition system used in this test. The data acquisition system also collected accelerometer and force data, though none of these sensors were used for the expansion; they were only applied as diagnostics. The cameras were run at 4096 frames per second with an exposure time of 50 μs to acquire test images. Each measurement frame was 1 second (4096 images) long and a total of ten frames were collected for averaging purposes. Image averaging was possible because the input excitation force was repeated identically for each frame once the start-up transients had died away. This reduced the total number of images that must be processed in the DIC software from 40960 to 4096.

The identical processing that was applied in Section and 5 was then applied to the left camera images from the experiment. 2D DIC was performed using the subsets from Section 4 to extract (u, v) displacements for each subset at each time step. Because the pseudorandom excitation only excited the test article above 100 Hz, a 4th-order high-pass Butterworth filter with 100 Hz cut-off frequency was applied to the subset displacements with the goal of reducing erroneous signals due to heat waves or other low-frequency optical distortions. The temporally-filtered subset displacements were then spatially filtered using the finite element shapes $\mathbf{\Phi}_a$ to extract a set of modal coefficients $\mathbf{q}$ at each time step, per equation (1). These coefficients were then multiplied

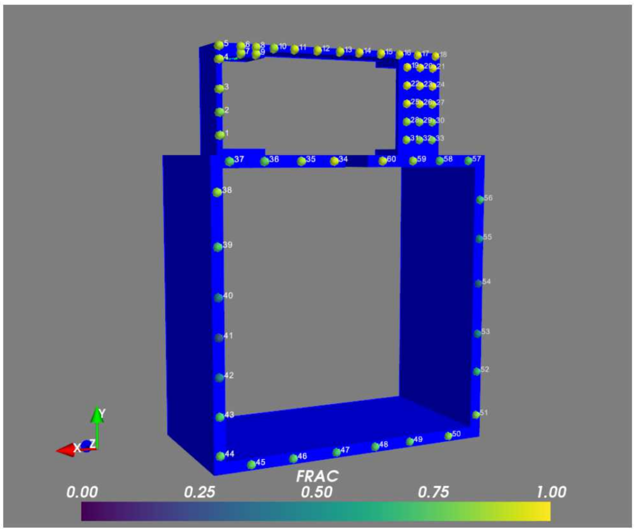


Fig. 13: 60 comparison locations shown on the BARC. The locations are colored by the FRAC between the z-direction responses of the expanded and truth data.

by the full finite element model shapes to produce full-field, 3D motions at each time step per equation (2).

Sixty locations were selected to compare expanded results to truth data from the stereo DIC analysis. At each location and for each coordinate direction the Time Response Assurance Criterion (TRAC) [10] and Frequency Response Assurance Criterion (FRAC) [11] were computed to compare the time responses and frequency spectra of the expanded and truth signals. The TRAC and FRAC are generalizations of the MAC using deflection shapes at each time step or frequency line rather than mode shapes to perform the correlation. Figure 13 shows the locations of each of these comparison points, and Figures 14 and 15 show the TRAC and FRAC values for all measurement locations for the x, y, and z directions. Figure 13 shows a triad that defines these directions. Note that the x and y directions were most closely aligned with the image plane of the camera, and the z direction is oriented primarily out of the image plane. Figure 16 shows time and frequency response comparison of the worst correlation, and Figure 17 shows the same comparison of the best correlation.

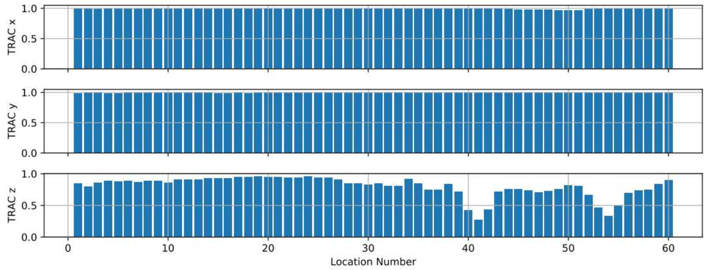


Fig. 14: TRAC comparing expanded to truth data for each coordinate direction for each location.

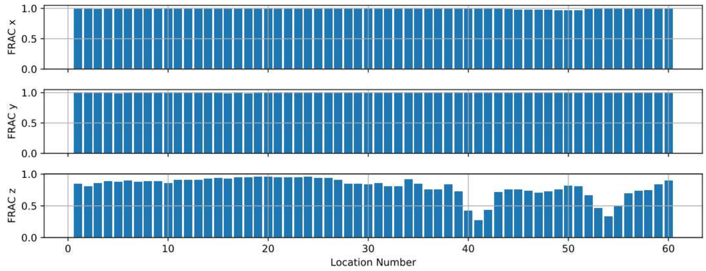


Fig. 15: FRAC comparing expanded to truth data for each coordinate direction for each location.

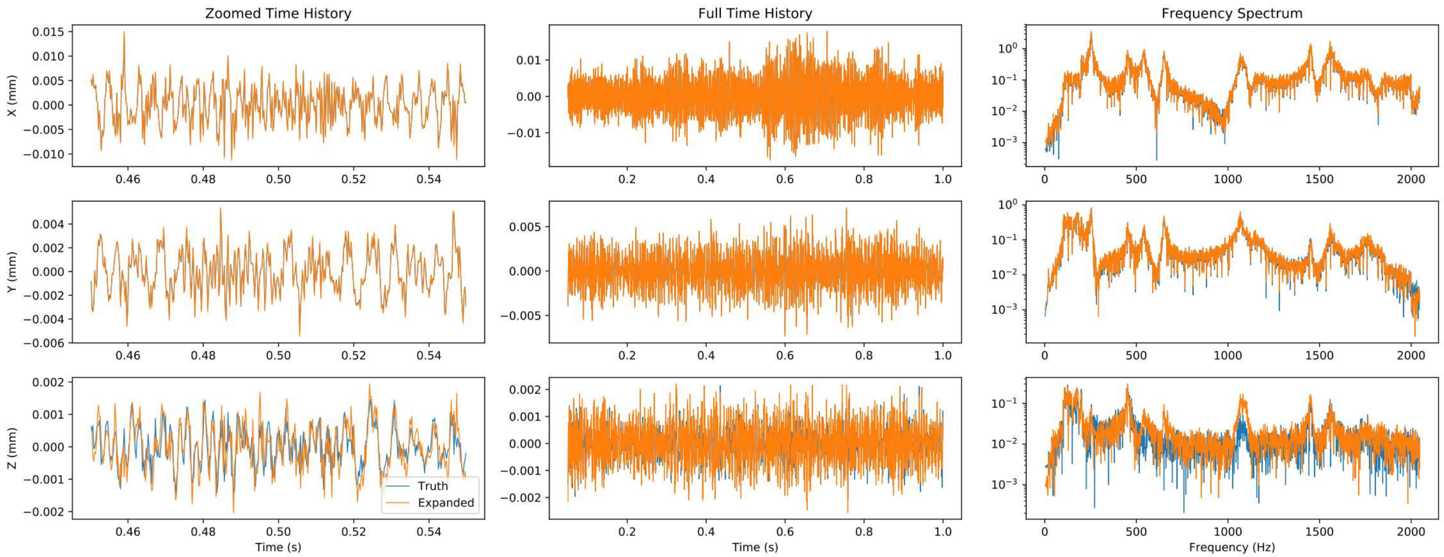


Fig. 16: Comparison of Location 41 data. This point had the worst correlation in the z direction.

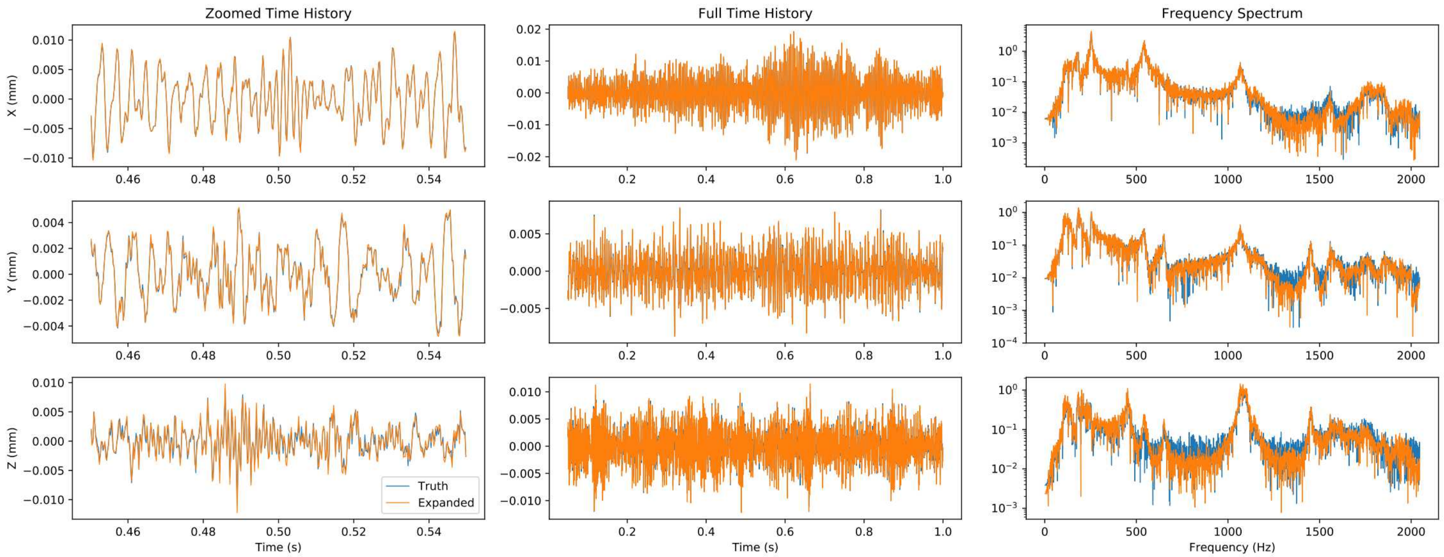


Fig. 17: Comparison of Location 24 data. This point had the best correlation in the z direction.

7 Discussion

While good agreement between truth and expanded signals was demonstrated in Section 6, there were distinct areas on the test object where the expansion process struggled. This section discusses some of those issues and potential remedies.

7.1 SEREP Expansion

For all checked locations, motion in the x and y directions showed very good agreement between the expanded and truth data. This was expected, as these degrees of freedom correspond most closely with the subset displacements used to perform the expansion. The out-of-plane z-direction responses were not predicted as accurately. Figure 13 shows the expansion struggled most on the vertical members of the “box” portion of the test article. Predictions on the component mounted to the box were much more accurate. While a FRAC or TRAC value of less than 0.5 seemingly indicates poor correlation, examination of the time and frequency responses of the worst points (e.g. as shown in Figure 16) does not paint as bleak of a picture. The majority of the frequency band seems to be accurately predicted, with the main sources of error coming from the frequency band between 1,050 and 1,100 Hz and to a lesser extent the frequency band around 250 Hz.

There can be several reasons that SEREP may incorrectly expand experimental data: (1) the finite element modes are not accurate representations of the true structure modes and therefore do not form adequate interpolation functions between the measured data to estimate unmeasured response, (2) the finite element model contains modes which are not necessary to expand the experimental data, or (3) insufficient degrees of freedom were measured to uniquely identify all modes which are being expanded.

To investigate, the 3D DIC truth data were combined with force and accelerometer diagnostic measurements, and modes were fit to the data using experimental modal analysis. This resulted in 13 experimental elastic shapes. Two SEREP case studies were performed to identify which of the potential expansion issues listed above were occurring. In the first case, the experimental mode shapes were filtered by the finite element mode shapes per equation (1) using all image and accelerometer degrees of freedom to create a set of coefficient vectors that represented the linear combination of finite element shapes to produce each experimental shape, shown in Figure 18a. In the second case, the accelerometer degrees of freedom were not used in the expansion; instead only the 3D image degrees of freedom from the truth DIC on the front surfaces of the part were used. This is shown in Figure 18b

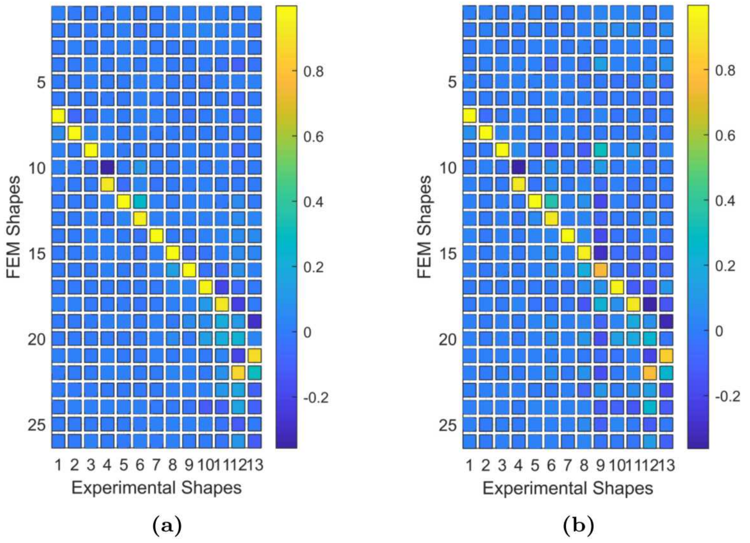
Because the accelerometers are spread evenly over the part, the set of coefficients from the first case is thought to represent the true linear combination of finite element shapes to create each experimental shape. Note that for each experimental mode, there is primarily one finite element mode that directly corresponds to it with smaller amounts of other modes also included to handle the small differences between the finite element model and test. Much work has been performed to understand the sensitivity of SEREP to model inaccuracies, and it has been shown to be very robust against even egregious model errors [12,13]. Therefore, for this work, the finite element model was thought to be sufficiently accurate. Additionally, all the finite element mode shapes used in the expansion showed significant contributions to the experimental mode shapes, so the finite element model was not thought to contain any modes that were unnecessary to expand to the experimental data.

If a set of degrees of freedom is adequate for a given expansion problem, one would not expect to see the shape coefficients computed by expanding from that set of degrees of freedom change from the truth value. When the accelerometer degrees of freedom were removed from the expansion in the second case study, the modal coefficients were similar to the truth case for many of the modes; however, the coefficients to reproduce experimental mode 9 have changed significantly. The main contribution to experimental mode 9 from finite element mode 16 has dropped from nearly 1 to less than 0.8, and several other modes are now contributing significantly to the shape. Mode 8 shows similar changes though they are not as drastic. When the coefficients to reproduce mode 9 are multiplied by the finite element shape matrix, the points far from the imaged surface no longer agree with the accelerometer data measured in the test. This indicates that the optically measured subset of degrees of freedom are by themselves insufficient to accurately expand modes 8 and 9. These modes, which have natural frequencies at 1,058 and 1,123 Hz, are likely the reason that poor expansion was obtained in the z direction in the 1,050–1,100 Hz frequency band.

In this case, the fact that the degrees of freedom were insufficient to expand specific modes was determined by comparison with truth data; however, in practice there might not be truth data available. Instead, the authors would encourage users of this technique to investigate the expansion analytically, as is shown in Section 5, perhaps even going as far as to perturb the model to understand the sensitivity of the expansion from the given set of degrees of freedom to modeling errors. Alternatively, a small number of "truth" sensors could be placed on the part to investigate how well the process expands to those sensor locations.

7.2 View Angles and Camera Lenses

This process attempts to re-create 3D motion from 2D motion using a finite element model to inform the missing degrees of freedom. This transformation relies on perspective to create motion when the part is moving towards or away from the cameras. Perspective effects will be largest with the test article closer to the camera with a wide angle lens. As the lens focal length increases the view angle decreases, so for a given translation towards the camera the relative enlargement of the part's image on the camera sensor will be smaller. In the



(a) **(b)**

Fig. 18: Comparison of Modal Coefficients using accelerometer and image degrees of freedom (a) and only image degrees of freedom (b). Note the change in composition of experimental mode 9.

limit that the focal length goes to infinity (e.g. an orthographic view which can be achieved using a telecentric lens), there is zero enlargement due to perspective. It is clear that this expansion process would break down in such a case, as at least one of the translation rigid body modes would be indistinguishable from the linear combination of the other two. Given this argument, if this work were to be repeated, the authors would investigate using a wider angle lens than the 85 mm lenses used in this proof-of-concept. Additionally, given the fact that motions in the x and y directions were extracted nearly perfectly but the motion in the z direction was less accurate, the authors would try using a view that was more oblique to the part xy plane so there would be a larger z component in the image plane. Note that the proximity of the part to the camera may be limited by the minimum focusing distance of a given lens. It may also be limited by the required depth of field, which for a given aperture will become narrower as the part moves closer to the camera.

7.3 Optical Errors due to Heat Waves

This technique uses finite element shapes as a basis for expanding measured data. It assumes that all the measured motions can be described by those shapes; however, there are some issues that may arise in optical testing that can invalidate this assumption. Heat waves are a well-known phenomenon in DIC analysis often active at temporal frequencies below 100 Hz [14], and these can distort a given measurement shape. These distorted shapes will not likely be found within the finite element model, so the expansion process suffers. Initially, no high-pass temporal filter was applied to the data, and very poor results were obtained from the expansion. Once the high-pass filter was added, as described in Section 6, which removed the effect of the heat waves, the data matched much better to the truth data.

7.4 Multi-view Analysis

One final point of discussion is the applicability of multiple views to this technique. As noted in Section 7.1, having all degrees of freedom on one or a few surfaces may not be sufficient to expand to motions on the entire part. One option is to supplement with accelerometers in locations that are not in the optical system line-of-sight. Alternatively, to stay with optical measurements only, another camera could be used to view additional degrees of freedom (in this case, a camera viewing the opposite side of the test article would offer the most improvement), synthetic finite element images could be created and 2D subset displacements could be extracted from that view as well. These additional degrees of freedom could simply be appended to equation (1) as extra rows of the $\mathbf{\Phi}_a$ and $\mathbf{x}_a$ matrices, and the modal coefficients $\mathbf{q}$ could be found for both views simultaneously. The addition of one or more extra views could significantly improve the expansion process. Note that the entire expansion process can be investigated analytically, as shown in Section 5, so the camera views and degrees of freedom used in the expansion could be optimized prior to the test.

8 Conclusions

This paper introduced a technique to extract 3D motions from a set of images from a single camera using finite element expansion techniques to estimate the out-of-plane displacements that have traditionally been considered missing from a single camera test. This technique utilized the SEREP technique to perform the expansion, creating synthetic finite element images from which arbitrary image degrees of freedom could be extracted. The synthetic finite element images were created such that they matched both the camera viewpoint as well as the local texture of the part. Because the synthetic images matched the test images, identical image processing techniques could be performed on the test and finite element images, meaning identical degrees of freedom could be produced with which the filtering could be performed. The technique was first tested using a synthetic deformation image, which was produced using a randomized set of modal coefficients. These coefficients were accurately reproduced by the expansion technique, so the prescribed 3D displacements were accurately estimated by the technique. The technique was then applied to an experimental dataset, and compared to truth data obtained using stereo DIC. Very good agreement was found in the x and y directions, which corresponded closely to the image plane directions. The z-direction motions were well predicted, but larger errors were found in some frequency bands due to the degrees of freedom used in the expansion process. Overall, this technique was successful in predicting 3D motions from a single set of 2D images. Future work will involve applying this technique to radiographic images, where degrees of freedom extracted from an image will not be limited to visible surfaces.

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. J. C. O'Callahan, P. Avitable, and R. Riemer, "System equivalent reduction expansion process," in *Proceedings of the Seventh International Modal Analysis Conference*, (Las Vegas, NV), Feb 1989.
2. B. L. Witt, D. P. Rohe, and T. F. Schoenherr, "Full field strain shape estimations from 3D SLDV," in *Rotating Machinery, Optical Methods & Scanning LDV Methods* (C. Niezrecki, J. Baqersad, and D. Di Maio, eds.), vol. 6 of *Conference Proceedings of the Society for Experimental Mechanics*, (Cham), pp. 31–45, Springer, 2019.
3. D. E. Soine, R. J. Jones, Jr., J. M. Harvie, T. J. Skousen, and T. F. Schoenherr, "Designing hardware for the boundary condition round robin challenge," in *Topics in Modal Analysis & Testing* (M. Mains and B. Dilworth, eds.), vol. 9 of *Conference Proceedings of the Society for Experimental Mechanics Series*, (Cham), pp. 119–126, Springer, 2019.
4. D. P. Rohe, "An optical test simulator based on the open-source Blender software," in *Proceedings of the 38th International Modal Analysis Conference*, (Houston, TX), Feb. 2020.
5. The Blender Foundation, "Blender.org." `https://www.blender.org/`. Accessed 2019-02-24.
6. G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, vol. 25, pp. 120–125, 2000.
7. V. Lepetit, F. Moreno-Noguer, and P. Fua, "EPnP: An accurate O(n) solution to the PnP problem," *International Journal of Computer Vision*, vol. 81, p. 155, Jul 2008.
8. D. Z. Turner, "Digital image correlation engine (DICe) reference manual," Sandia Report SAND2015-10606 O, Sandia National Laboratories, 2015.
9. R. J. Allemang and D. L. Brown, "A correlation coefficient for modal vector analysis," in *Proceedings of the 1st international modal analysis conference*, vol. 1, pp. 110–116, SEM Orlando, 1982.
10. P. Avitabile and P. Pingle, "Prediction of full field dynamic strain from limited sets of measured data," *Shock and Vibration*, vol. 19, pp. 765–785, Apr. 2012.
11. R. J. Allemang, "The modal assurance criterion–twenty years of use and abuse," *Sound and vibration*, vol. 37, no. 8, pp. 14–23, 2003.
12. L. Thibault, A. Butland, and P. Avitabile, "Variability improvement of key inaccurate node groups – VIKING," in *Topics in Modal Analysis II, Volume 6* (R. Allemang, J. De Clerck, C. Niezrecki, and J. Blough, eds.), (New York, NY), pp. 603–624, Springer New York, 2012.
13. D. Fowler, R. Schultz, B. Zwink, and B. Owens, "Flight environments demonstrator: Part III – sensitivity of expansion to model accuracy," in *Proceedings of the 37th International Modal Analysis Conference*, Society for Experimental Mechanics, 2019.
14. E. Jones and P. Reu, "Distortion of digital image correlation (DIC) displacements and strains from heat waves," *Experimental Mechanics*, vol. 58, pp. 1133–1156, Sep 2018.