SAND2020-13855C

Stereo Photogrammetry Camera Pose Optimization

Bryan L. Witt[1], J. Justin Wilbanks[2], Brian C. Owens[2], Daniel P. Rohe[1]

Sandia National Laboratories[*]

[1]Experimental Structural Dynamics Department
[2]Analytical Structural Dynamics Department

P.O. Box 5800 - MS0557
Albuquerque, NM, 87185

ABSTRACT

Stereo photogrammetry makes use of calibrated camera pairs to obtain three-dimensional information from two-dimensional images. The accuracy of the extracted measurements is extremely dependent on the selection and setup of the camera system. For a given test object and desired viewing orientation, there is no one "correct" stereo camera setup, but rather a range of potential setups with some approaching an optimal system with respect to maximizing the measurement resolution. The open-ended nature of this test design exercise is compounded by equipment availability and the fact that many of the setup parameters have dependent characteristics, e.g. changing focal distance will affect stand-off distance, field of view, and image projection, among others. This work describes a planning tool that utilizes projective and Euclidian geometry to iteratively estimate optimal camera poses for available equipment, determines the most efficient image size, and also performs checks for lens diffraction, minimum focal distance, and adequate depth of field. Integrating a finite element model with these calculations further extends planning capabilities by allowing (1) an accurate definition of the volume to be imaged and (2) the ability to estimate response displacements in pixels due to an arbitrary excitation applied to the test object. This latter capability is critical for pre-test determination of the chosen camera setup's ability to successfully extract three-dimensional measurements. The theory and workflow are presented along with an experimental demonstration.

Keywords: Stereo; Photogrammetry; Test planning; Camera pose; Displacement estimation

---

[*] Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and 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-NA-0003525.

1 INTRODUCTION

Photogrammetry is well established as a diagnostic capability for quasi-static displacement and strain measurements as well as large-scale (integer pixel) motion tracking [1-4]. Optical measurement techniques have many benefits, such as being non-contact and very fast to field. Photogrammetry's use in structural dynamics, where higher frequency displacements are typically sub-pixel, is gaining popularity as camera technology and bespoke data processing methods are established [5-8]. A recent study [9] demonstrated that carefully constructed experimental modal tests using DIC can directly extract modal information from displacements as small as 0.001 pixel.

The resolution of any optical system is dependent of the field of view (FOV); optimizing the number of pixels across an imaged area of interest (AOI) is critical for measuring sub-pixel displacements. However, because there are an endless number of practical considerations for each unique optical test (e.g. camera sensor selection, lenses available, physical space around the test object, etc.), there is no single truly "optimized" configuration. Typically, the practitioner must rely solely on their experience to select cameras and lenses and then determine their best pose in a stereo rig. Pose is defined here as the physical location and orientation of the cameras in space relative to each other as well as the test object.

It is entirely conceivable to set up a stereo system, collect and download images, and post-process the data only to find that the displacements of the test object never overcame the noise floor of the optical system. This work presents a photogrammetry pre-test planning workflow that will inform the selection of camera/lens pairs and the stereo poses that approximate their optimal setup in terms of measurement resolution. This process also considers lens minimum focus distances, depth of field (DOF) requirements, and checks for lens diffraction limitations.

Finally, we use a Finite Element Model (FEM) and the pose optimization results to determine what points in a bounding volume will be visible to both cameras and estimate their respective displacement amplitudes on a mode-by-mode basis for a given excitation. The modal displacement estimates can be directly correlated to the known (or estimated) noise floor of the camera system, easily identifying which modes may or may not be observable by the optical system. Armed with the information from the test planning workflow, the practitioner can easily iterate on the test setup (equipment, AOI, pose, excitation, etc.) to have the best opportunity to obtain the measurements of interest and meet the test objectives.

The following section provides background information on general stereo camera setups, including the coordinate systems which are utilized and a brief coverage of 3D-2D projective transformations. Section 3 presents the iterative pose optimization routine and demonstrates output results relative to an actual test. Section 4 provides a stand-alone FEM analysis to determine visible nodes and estimate displacements, using the final results from Section 3 as input. The reader is cautioned that these two sections use different variable definitions. The final section discusses conclusions and future work under consideration.

2 GENERAL STEREO PHOTOGRAMMETRY SETUP

The test planning workflow described in this work can be applied to a 2D photogrammetry using a single camera. However, a single camera test is usually much easier to setup and iterate upon; in this work we will focus on the more complex stereo camera setup utilizing two cameras for 3D photogrammetry measurements.

Consider a test article in 3D space which is defined in an arbitrary "world" Cartesian coordinate system, $\mathbf{O_w}$, as depicted in Figure 1. It is usually extremely convenient to let the world coordinate system match that of the FEM global coordinate system, as is done in this work. Two cameras, denoted Camera 0 and Camera 1, are positioned such that their optical axes are pointed in the direction of the test article and have local coordinate systems $\mathbf{O}_i$ where i indicates the camera index 0 or 1, respectively. The baseline (BL) is the line formed between the origin points of $\mathbf{O}_i$ and lies on the same plane containing all points along both optical axes, $\mathbf{z}_i$. The camera rig is a fourth coordinate system, $\mathbf{O_r}$, and is defined at the center of the baseline, oriented such that $\mathbf{x_r}$ points from Camera 0 to Camera 1 and $\mathbf{y_r}$ is normal to the plane containing the baseline and optical axes. Finally, the image planes are located at the focal distance f_i along the optical axis $\mathbf{z}_i$. Ideally, the image center would lie on the optical axis $\mathbf{z}_i$ such that $(c_x,c_y)_i = (0,0)$; typically there are slight offsets, but this is a reasonable approximation for a planning tool. By convention, the integer pixel indexing in each image space is defined such that $(u,v)_i = (0,0)$ is in the upper left corner as shown in Figure 1.

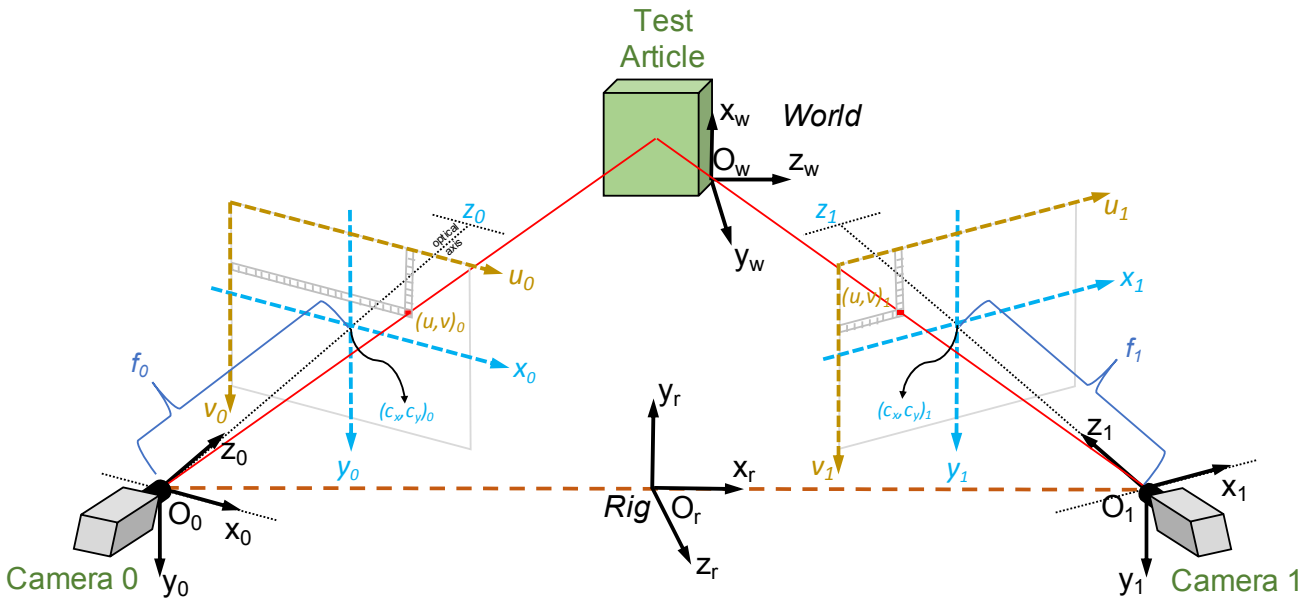

Figure 1: Stereo photogrammetry setup

As part of a stereo photogrammetry test, a camera calibration is performed prior to data collection, which establishes both the intrinsic and extrinsic parameters of the camera system. In the test planning workflow, we do not have equipment set up to perform a calibration, but the form of the results elucidates the parameters needed for planning purposes. The intrinsic parameters form $\mathbf{K}_i$ and comprise the optical center $(c_x,c_y)_i$, the focal lengths $(f_x,f_y)_i$ and a skew term q that is the tangent of the angle between the image axes. The extrinsic parameters form $\mathbf{[R|T]_{ir}}$ and comprise the rotation matrix and translation vector that transforms 3D coordinates $\mathbf{X}$ (in homogeneous coordinates) from $\mathbf{O_r}$ to $\mathbf{O}_i$. Thus, Equation (1) describes the projective transform between a 3D point in the rig coordinate system to 2D pixels $\mathbf{u}$ in the image frame. The $\mathbf{u}$ results are also in homogeneous coordinates and should be divided by the homogeneous scalar a to obtain final pixel values. The form of this projection will be used throughout the process.

$$\mathbf{u}_i = \mathbf{K}_i[\mathbf{R}|\mathbf{T}]_{i\mathbf{r}}\mathbf{X}_\mathbf{r}$$

$$\begin{bmatrix} au \\ av \\ a \end{bmatrix}_i = \begin{bmatrix} f_x & q & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}_i \begin{bmatrix} r_{xx} & r_{xy} & r_{xz} & t_x \\ r_{yx} & r_{yy} & r_{yz} & t_y \\ r_{zx} & r_{zy} & r_{zz} & t_z \end{bmatrix}_{ir} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}_r \tag{1}$$

3 STEREO POSE OPTIMIZATION

An overview of the workflow is provided in Figure 2. At a high level, the process is divided into four main segments: input, setup, pose estimation, and pose validation. The following section will elaborate on each segment, providing necessary details for those interested in recreating the tool in a programming language of choice.

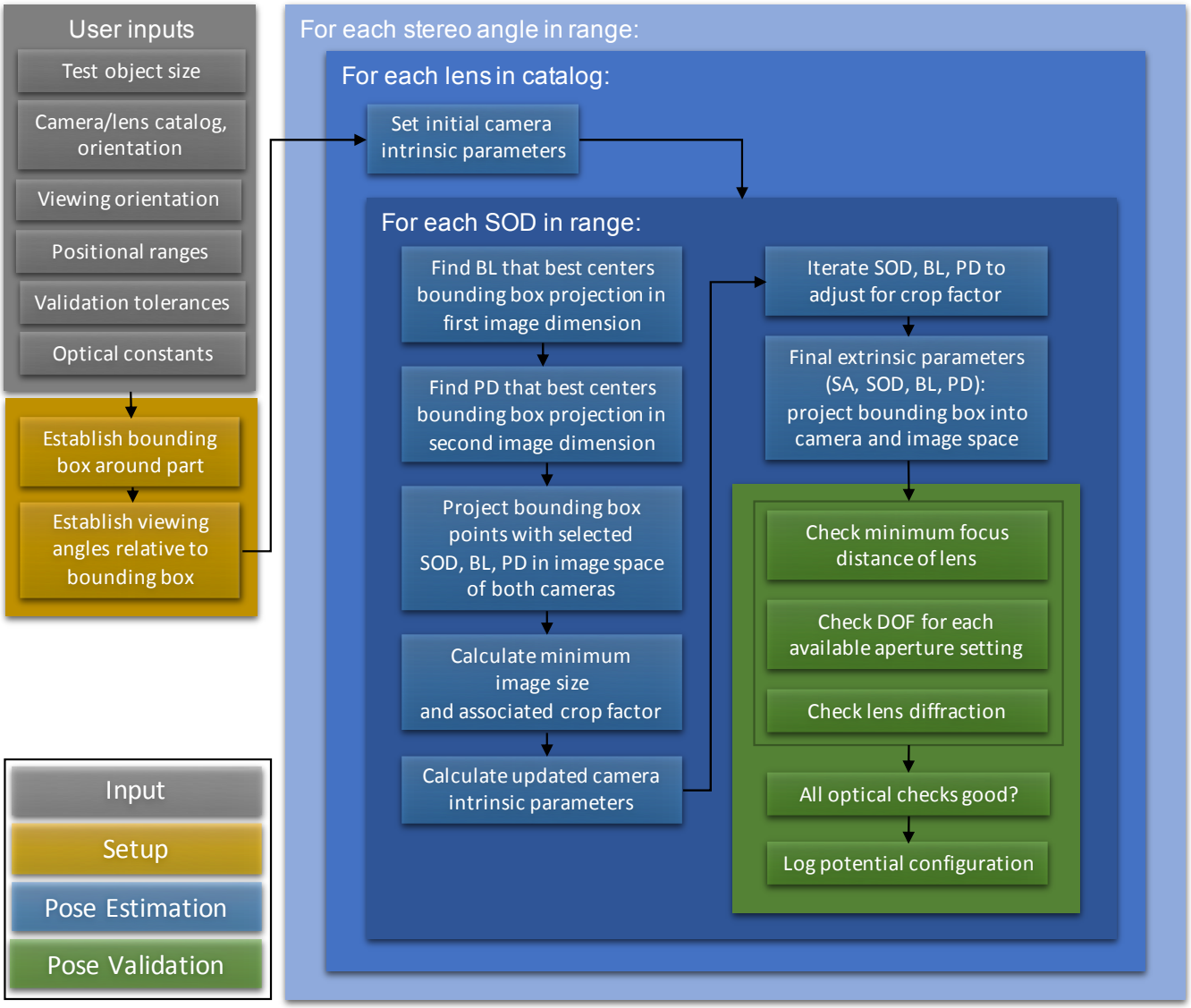

Figure 2: Stereo pose optimization flowchart

3.1 User Inputs

First, specifications regarding the available imaging hardware are gathered. Specifically, the following information must be compiled:

- Camera sensor specifications:
  - Full image size (s_x, s_y) (pixels)
  - Pixel size, ϵ (mm)
- Camera orientation (landscape or portrait, see Section 3.3)
- Camera stand orientation (horizontal or vertical, see Section 3.3)
- A list of lenses available and their properties:
  - Make and model
  - Nominal focal length f (mm)
  - Minimum focal distance, s_{min} (mm)
  - Aperture f-stop values, N (minimum and maximum)

With the exception of the above hardware specifications, all of the other user inputs presuppose a knowledge of the planning process, and will be described in detail in the following sections:

- Physical size or coordinates of a bounding box that encloses the surfaces to be imaged
- Bounding box margin
- Euler angles that describe the viewing orientation (i.e. how to "look at" test object)
- Space available around the part
    - Acceptable range for standoff distance (SOD)
    - Acceptable range for camera baseline (BL)
    - Acceptable range for camera perpendicular distance (PD)
- Acceptable range of camera stereo angles
- Tolerance ranges for optical validation parameters
    - Minimum focus distance
    - DOF
    - Lens diffraction
- Optical constants that will be used for DOF and lens diffraction estimations
    - Acceptable circle of confusion, c (mm)
    - Light wavelength, λ (mm)

There is a tradeoff between minimizing required user inputs and increasing the flexibility of the tool. The workflow presented here corresponds to the authors' current implementation but can easily be modified according to specific application needs.

3.2 Setup: Establish Bounding Box

For the purpose of test planning, we define an imaginary bounding box that contains all surfaces of the test object that are to be imaged. Consider an example test object, such as the Box and Removable Component (BARC) [10], from which the areas of interest (AOI) include all camera-facing surfaces. The corresponding bounding box is shown in Figure 3 as a red rectangular cube encompassing those surfaces; the entirety of the test object does not need to be included in the bounding box volume, only the surfaces to be imaged. The bounding box has its own coordinate system $\mathbf{O_b}$, which is centered on the front surface as shown in Figure 3. The coordinates of the bounding box nodes are easily defined given its dimensions (b_x, b_y, b_z):

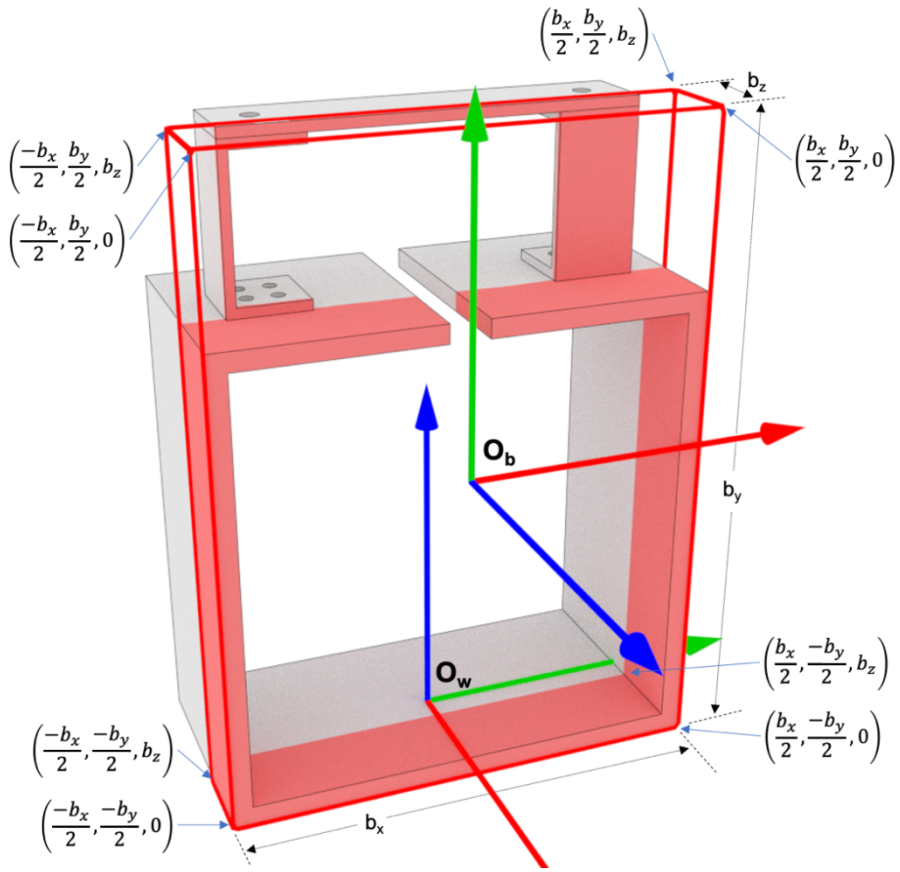

Figure 3: Bounding box nodes in local coordinates

In practice, the box node coordinate values may be given in the world/FEM coordinate system along with the dimensions (b_x, b_y, b_z). In this case, first use the bounding box dimensions to compute the box node positions as shown in Figure 3. Once the 3D coordinates are known in both $\mathbf{O_w}$ and $\mathbf{O_b}$, we compute the transformation between the two, $\mathbf{[R|T]_{bw}}$ for later use in Section 3.4. This is accomplished through least squares rigid motion (LSRM) transformation using multiple point correspondences [11].

Although the bounding box is defined by the test object's dimensions, in practice it is useful to allow the box to expand by some margin, which can be specified by the user in all three axes independently. This relaxes the degree of accuracy to which the cameras must be placed to retain all AOIs in the image frame. More importantly, this also allows for lens "focus breathing" which varies in both direction and severity from lens to lens (more details in Section 3.5). This is easily achieved by defining the bounding box "tight" on the test object as shown in Figure 3, then applying the specified margins to the box coordinates in $\mathbf{O_b}$.

The bounding box dimensions (b_x, b_y, b_z) are user-defined inputs. Similarly, the margin values associated with each box dimension are user-defined inputs.

3.3 Pose Estimation

Consideration must be given to the camera (sensor) orientation, relative to how it is defined by the user. For example, if the full image size is defined as (2048, 1920) pixels and the camera is mounted such that the 2048 pixels are in the *u* dimension of the image, the camera orientation is considered "landscape" (refer to Figure 1). Conversely, if the 2048 pixels lie along the *v* dimension of the image, the camera orientation is considered "portrait." Thus, the relationship between the image size and the camera orientation defines which axis of the sensor correspond to the image *u* and *v* dimensions. Either camera orientation may be used, and this is included as an option in the user-defined inputs.

Similarly, consideration is given to the camera stand orientation relative to the test object. When the cameras are aligned along the image *u* dimension, we refer to the stand orientation as "horizontal." Conversely, if the cameras are aligned along the image *v* dimension, we refer to the stand orientation as "vertical." This distinction will be important in determining the targets for centering the test object within the image frame in subsequent steps. The camera stand orientation is also a user-defined input.

The user is also responsible for the definition of the view angle with respect to $\mathbf{O_b}$. Consider the typical stereo pose where the camera rig is placed directly in front of the test object as depicted in Figure 4; in this case it is trivial to place the camera rig in space using $\mathbf{O_b}$. However, if the camera rig needs to be positioned somewhere other than directly in front of the part, we can define a new coordinate system $\mathbf{O_v}$ that makes this task easier. A user-defined set of Euler angles describe a rotation matrix, $\mathbf{R_{vb}}$, to transform from $\mathbf{O_b}$ to $\mathbf{O_v}$. For example, to look down on the bounding box from the right side, the Euler angles (in this case using the Tait-Bryan convention with intrinsic rotations about the axes) $(\theta_y,\ \theta_{x'},\ \theta_{z''}) = (-10,15,0)°$ would rotate into a coordinate frame in which the rig position is easily defined, as depicted in Figure 5. The superscripts $'$ and $''$ on the angle subscripts denotes the once and twice rotated coordinate frame, respectively.

To position the camera rig relative to the bounding box, three positioning dimensions are defined in $\mathbf{O_v}$:

- Standoff distance (SOD) is the distance along the *z*-axis of $\mathbf{O_v}$ to the camera rig.
- Baseline (BL) is two times the distance to either camera along the *x*-axis of $\mathbf{O_v}$ (we assume both cameras are equidistant from the rig origin of $\mathbf{O_r}$). This is equivalent to the normal definition of baseline which is simply the straight-line distance between camera origin points.
- Perpendicular distance (PD) is measured along the *y*-axis of $\mathbf{O_v}$.

The user defines the ranges that SOD, BL and PD may fall within based on the space available around the test object. The camera positions with respect to $\mathbf{O_v}$ are shown in Equation (2). The coordinates $\mathbf{C}_i$ are also the origin points of the respective camera coordinate systems $\mathbf{O}_i$. The stereo angle, θ_{stereo}, is the angle formed between the optical axes of the cameras (see Figure 4). Typical stereo DIC tests will be performed in the ~20-30° stereo angle range [2] as this provides a balance between resolving out-of-plane motions and image correlation breakdown. For the planning workflow, the user defines the stereo angle, or stereo angle range to consider. Referring back to Figure 2, note that the stereo angle is not changed when iterating the SOD, BL, or PD; some configurations will obviously not properly frame the bounding box within the image and are rejected.

$$\left.\begin{aligned} \boldsymbol{C_0} &= \left(\frac{-BL}{2}, PD, SOD\right)^T \\ \boldsymbol{C_1} &= \left(\frac{BL}{2}, PD, SOD\right)^T \end{aligned}\right\} \text{if stand orientation is horizontal}$$

$$\left.\begin{aligned} \boldsymbol{C_0} &= \left(PD, \frac{-BL}{2}, SOD\right)^T \\ \boldsymbol{C_1} &= \left(PD, \frac{BL}{2}, SOD\right)^T \end{aligned}\right\} \text{if stand orientation is vertical} \qquad (2)$$

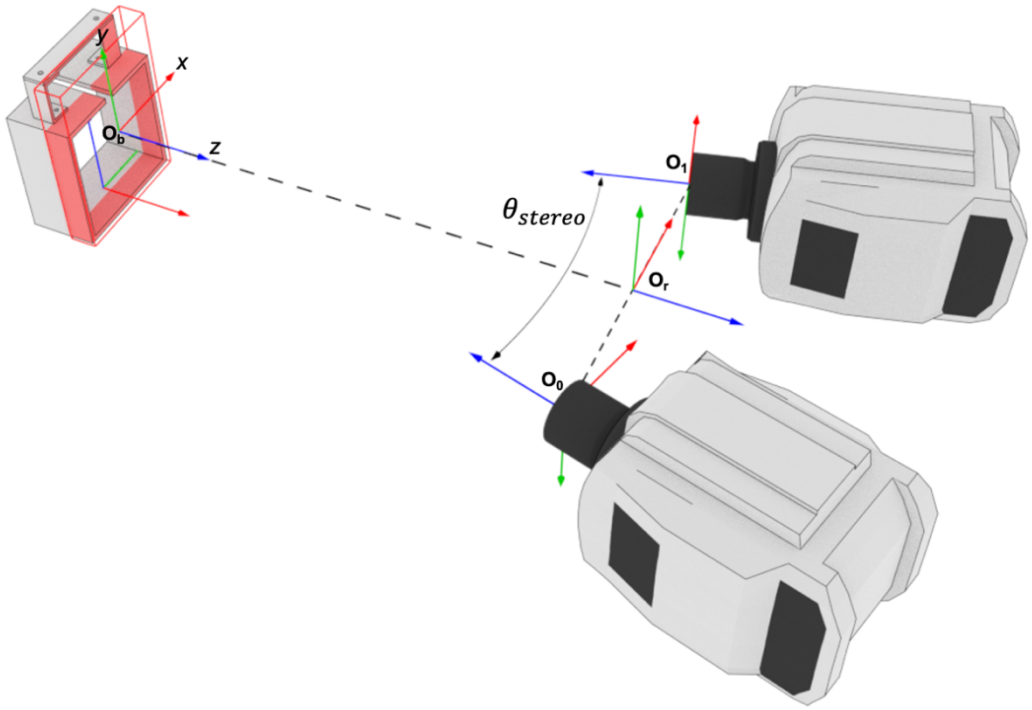

Figure 4: Typical stereo setup with no view angle change

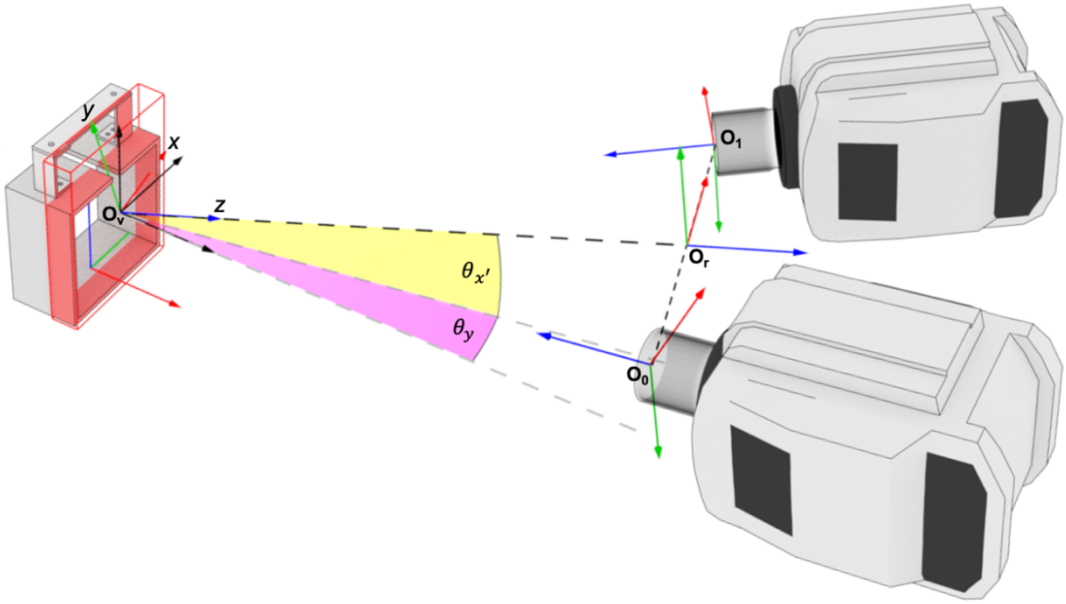

Figure 5: Stereo setup with example view angle change

We now have the tools necessary to project the bounding box 3D coordinates into image pixels for either camera. Referring to the form of Equation (1) and noting that transformation matrices can be chained together, the 3D-2D projection from the bounding box to image space is:

$$\mathbf{u}_i = \mathbf{K}_i\mathbf{R}_{i\mathbf{v}}[\mathbf{I}|\mathbf{C}]_i\mathbf{R}_{\mathbf{vb}}\mathbf{X}_{\mathbf{b}}$$

$$\begin{bmatrix} au \\ av \\ a \end{bmatrix}_i = \begin{bmatrix} f_x & q & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}_i \begin{bmatrix} r_{\mathrm{xx}} & r_{\mathrm{xy}} & r_{\mathrm{xz}} \\ r_{\mathrm{yx}} & r_{\mathrm{yy}} & r_{\mathrm{yz}} \\ r_{\mathrm{zx}} & r_{\mathrm{zy}} & r_{\mathrm{zz}} \end{bmatrix}_{i\mathrm{v}} \begin{bmatrix} 1 & 0 & 0 & C_{\mathrm{x}} \\ 0 & 1 & 0 & C_{\mathrm{y}} \\ 0 & 0 & 1 & C_{\mathrm{z}} \end{bmatrix}_i \begin{bmatrix} r_{\mathrm{xx}} & r_{\mathrm{xy}} & r_{\mathrm{xz}} & 0 \\ r_{\mathrm{yx}} & r_{\mathrm{yy}} & r_{\mathrm{yz}} & 0 \\ r_{\mathrm{zx}} & r_{\mathrm{zy}} & r_{\mathrm{zz}} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}_{\mathrm{vb}} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}_{\mathrm{b}} \tag{3}$$

where $\mathbf{R}_{i\mathbf{v}}$ takes into account the camera z-axis flip and stereo angle by rotating $\mathbf{O}_{\mathbf{v}}$ to $\mathbf{O}_i$ as described by the Euler angles (in radians):

$$\left.\begin{aligned} \left(\theta_x, \theta_{y'}, \theta_{z''}\right)_{0\mathrm{v}} &= \left(-\pi, \frac{-\theta_{stereo}}{2}, 0\right) \\ \left(\theta_x, \theta_{y'}, \theta_{z''}\right)_{1\mathrm{v}} &= \left(-\pi, \frac{\theta_{stereo}}{2}, 0\right) \end{aligned}\right\} \text{if stand orientation is horizontal}$$

$$\left.\begin{aligned} \left(\theta_x, \theta_{y'}, \theta_{z''}\right)_{0\mathrm{v}} &= \left(-\pi, \frac{\theta_{stereo}}{2}, 0\right) \\ \left(\theta_x, \theta_{y'}, \theta_{z''}\right)_{1\mathrm{v}} &= \left(-\pi, \frac{-\theta_{stereo}}{2}, 0\right) \end{aligned}\right\} \text{if stand orientation is vertical} \tag{4}$$

Since no actual camera calibration is performed in the planning workflow, we must make assumptions in order to populate $\mathbf{K}_i$. First, we assume the actual focal length of the lenses, f, is the nominal manufacturer value (e.g. 50 mm) and the pixel size is square such that $f_x = f_y = f/\epsilon$. Further, we assume the optical axis is aligned with the center of the image such that $c_x = s_x/2$ and $c_y = s_y/2$. Finally, in most test scenarios the skew is negligible, and one can assume $q = 0$.

From right to left, the terms in Equation (3) effectively take the 3D coordinates of the bounding box in $\mathbf{O}_{\mathbf{b}}$, transform them into $\mathbf{O}_{\mathbf{v}}$, translate them to the camera positions, rotate them into $\mathbf{O}_i$, and finally project them into image space. Thus, for a user-defined stereo angle and desired viewing angle, one can iterate over the camera positions $\mathbf{C}_i$ to find potential rig poses for which the bounding box is completely contained and centered in both camera images.

The iteration process is outlined in the blue boxes on the right side of Figure 2. For each stereo angle considered, the workflow will loop over all lenses specified in the catalog, allowing the intrinsic matrices $\mathbf{K}_i$ to be formed. For a first pass, we assume the full sensor size will be used (no sensor cropping). Next, the workflow loops over each SOD in the range specified by the user. At each SOD step, first assume a PD of zero and iterate over all BL values in the user-defined range to find which best centers the bounding box in both images using Equation (3). The camera stand orientation will change which image dimension will be affected by a change in BL: u if stand is horizontal (c_x is the center target) and v if vertical (c_y is the center target). Hence, the optimal BL will minimize the difference between the center target and the mean of the projected bounding box points. This must be done for both cameras which will have competing "optimal" values, meaning the center target errors from both cameras should be evaluated together (e.g. minimizing the RMS error values from both cameras). Similarly, the next step is to assume the BL just calculated and iterate over each PD in the user-defined range, finding the value that best centers the bounding box in the associated image dimension depending on stand orientation.

This process produces the optimal BL and PD for the current SOD iteration; however, there is no guarantee yet that together these $\mathbf{C}_i$ values will project the bounding box completely within the image space. Thus, the optimal values for this SOD are passed into Equation (3) and the projection is checked to see if all points fall within the (full) image size. At some instance, as SOD iterates outward, the entirety of the bounding box will fit within both camera images. Clearly, some configurations will run out of positional range before this occurs, in which case, the current SOD is abandoned, and the next iteration begins.

Even when the bounding box is properly framed in the image, there will often be "wasted pixels" where the bounding box does not fill the image in both directions. In such cases, it makes sense to crop the sensor appropriately and minimize the image size. The optimal image size is easily determined since the bounding box pixel positions, and thus extent in both image directions, has already been calculated. However, cropping the image size necessitates recalculating new intrinsic camera matrices, $\mathbf{K}_i$. Further, a subroutine must be performed in which the current SOD iteration is increased slightly to account for the zoom effect of cropping the sensor, and the BL and PD optimization iterations are re-performed. The end result is an optimized image size and camera positions (for the current θ_{stereo}, lens, and SOD). Note the user-defined resolution of the steps in the ranges provided for SOD, BL, and PD will determine how accurately the algorithm can center the bounding box.

3.4 Pose Validation

Once a potential pose is established, the optical viability must be evaluated (green box, bottom right of Figure 2). The current version of the planning workflow evaluates for lens minimum focal distance, DOF, and lens diffraction limitations.

The minimum focal distance and DOF checks can be easily accomplished by recognizing that the bounding box coordinates can be transformed into 3D camera coordinates $\mathbf{O}_i$ by leaving the $\mathbf{K}_i$ out of Equation (3) as:

$$\mathbf{x}_i = \mathbf{R}_{i\mathbf{v}}[\mathbf{I}|\mathbf{C}]_{\boldsymbol{i}}\mathbf{R}_{\mathbf{vb}}\mathbf{X}_{\mathbf{b}}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}_i = \begin{bmatrix} r_{\mathrm{xx}} & r_{\mathrm{xy}} & r_{\mathrm{xz}} \\ r_{\mathrm{yx}} & r_{\mathrm{yy}} & r_{\mathrm{yz}} \\ r_{\mathrm{zx}} & r_{\mathrm{zy}} & r_{\mathrm{zz}} \end{bmatrix}_{iv} \begin{bmatrix} 1 & 0 & 0 & C_{\mathrm{x}} \\ 0 & 1 & 0 & C_{\mathrm{y}} \\ 0 & 0 & 1 & C_{\mathrm{z}} \end{bmatrix}_i \begin{bmatrix} r_{\mathrm{xx}} & r_{\mathrm{xy}} & r_{\mathrm{xz}} & 0 \\ r_{\mathrm{yx}} & r_{\mathrm{yy}} & r_{\mathrm{yz}} & 0 \\ r_{\mathrm{zx}} & r_{\mathrm{zy}} & r_{\mathrm{zz}} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}_{\mathrm{vb}} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}_{\mathrm{b}} \tag{5}$$

Once all bounding box points have been transformed to the camera CS, we can find the center value of their z-coordinates:

$$s_i = \big(\max(\mathbf{z}_i) + \min(\mathbf{z}_i)\big)/2 \tag{6}$$

Ideally, the focused plane should lie at the center of the bounding box volume. Since the z-axis is the optical axis of the camera, s_i corresponds to the distance between the focused plane and the lens. Therefore, if the calculated focus distances are greater than or equal to the minimum focal distance of the lens, such that Equation (7) is true, the pose is considered valid. Generally, a relaxation of the equality is beneficial in actual application, in the form of the user-defined tolerance (e.g. ≥ 0.95). Note that we have dropped the i subscript denoting individual cameras, but the check must be performed for both cameras.

$$s/s_{min} \geq 1 \tag{7}$$

Next we consider DOF requirements to ensure the entire bounding box will be in focus for both cameras. Since the DOF is a function of the lens aperture setting, we must consider each f-stop value, N, of the lens under consideration. The estimated DOF for near and far points is [12]:

$$\begin{aligned} DOF_{near} &= \frac{Hs}{H+s} \\ DOF_{far} &= \frac{Hs}{H-s} \end{aligned} \tag{8}$$

where H is the hyperfocal distance, given as:

$$H = \frac{f^2}{Nc} \tag{9}$$

and c is the acceptable circle of confusion, which is a user-defined value. The circle of confusion is the largest diameter of a spot that will be perceived as a point; it can conservatively be set to the size of one pixel.

Using the coordinates of the bounding box in the camera CS from Equation (5), we can check if the extrema in the z-dimension of both cameras are bounded by the calculated DOF for each aperture f-stop to determine if the configuration is valid within a user-defined tolerance:

$$\begin{aligned} \frac{\min(\mathbf{z})}{DOF_{near}} &\geq 1 \\ \frac{DOF_{far}}{\max(\mathbf{z})} &\geq 1 \end{aligned} \tag{10}$$

Finally, if both the minimum focus distance and DOF checks are valid, the lens diffraction is estimated. In an optical system with a circular aperture, a point is not imaged as a point but rather as an Airy disk, which has one central lobe and sequentially degrading concentric diffraction rings [13]. The diameter of the Airy disk is:

$$D = 2.44\lambda N \tag{11}$$

where λ is the wavelength of the light entering the lens. Since the Airy diameter represents the smallest theoretical spot size that can be imaged, we consider the optical system to be diffraction (lens) limited when D is greater than the pixel size, such that a valid configuration will satisfy the following equality within a user-defined tolerance:

$$\epsilon / D \geq 1 \tag{12}$$

If a pose configuration passes all three optical validation checks, it is flagged as a viable and all relevant parameters are recorded. The process is repeated for the next iteration of SOD→lens→stereo angle.

The camera positions in $\mathbf{O_v}$ are already given by SOD, BL and PD values as given in Equation (2). It may be more convenient to also record these coordinates in the bounding box system, $\mathbf{O_b}$, which may be easier to physically measure when actually positioning the cameras for a test:

$$\mathbf{G}_i = \mathbf{R_{vb}}^{\boldsymbol{T}} \mathbf{C}_i \tag{13}$$

where ^{T} indicates the matrix transpose. Similarly, the camera positions can be recovered in the world (FEM) coordinate system $\mathbf{O_w}$ by use of the transformation between world and bounding box $\mathbf{[R|T]_{bw}}$ established in Section 3.2:

$$\mathbf{W}_i = \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{0} & 1 \end{bmatrix}_{\mathbf{bw}}^{-\mathbf{1}} \begin{bmatrix} \mathbf{G}_i \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} aw_x \\ aw_y \\ aw_z \\ a \end{bmatrix}_i = \begin{bmatrix} r_{\mathrm{xx}} & r_{\mathrm{xy}} & r_{\mathrm{xz}} & t_{\mathrm{x}} \\ r_{\mathrm{yx}} & r_{\mathrm{yy}} & r_{\mathrm{yz}} & t_{\mathrm{y}} \\ r_{\mathrm{zx}} & r_{\mathrm{zy}} & r_{\mathrm{zz}} & t_{\mathrm{z}} \\ 0 & 0 & 0 & 1 \end{bmatrix}_{\mathrm{bw}}^{-1} \begin{bmatrix} g_x \\ g_y \\ g_z \\ 1 \end{bmatrix}_{\boldsymbol{i}} \tag{14}$$

A list of the primary results to record for each viable pose would include:

- Lens settings (*N*, *s*)
- θ_{stereo}
- Optimized image size (pixels, pixels)
- $\mathbf{C}_i$ camera positions in $\mathbf{O_v}$ (SOD, BL, PD)
- $\mathbf{G}_i$ camera positions in $\mathbf{O_b}$
- $\mathbf{W}_i$ camera positions in $\mathbf{O_w}$
- Optical validation parameter results
- Camera intrinsic matrices
- Transformation matrices used

3.5 Evaluating Output and Final Pose Selection

The workflow, as presented, will likely output many (thousands) viable pose configurations. For example, for a given stereo angle there can be viable poses for each lens, at multiple SODs, and with multiple lens aperture settings. We can easily reduce the number of viable poses to a more refined set by realizing that only the pose with the minimum SOD is optimal in the sense that it maximizes the number of pixels across the bounding box. Simply flagging the first configuration with a SOD that results in a valid pose as optimal is recommended. Additionally, an "image" of how this pose projects the bounding box into the image space is useful and easily accomplished with Equation (3). This reduces the output to contain only the best pose for each lens in the catalog and its corresponding range of *N* that will provide adequate DOF and prevent lens diffraction. From this reduced list, the user can easily pick which lens and aperture setting make the most sense for the test at hand.

As an example, if we wished to set up a test similar to that shown in Figure 5 with the user inputs listed in Table 1, there are thousands of valid poses that are found (exact number will depend on the input parameters such as the number of lenses available, SOD/BL/PD range and step size, and number of stereo angles considered). The optimal set is reduced to the number of lenses in the user catalog (in this case 9). Examples from two different lenses are shown in Figure 6. The red face represents the front (camera-facing) surface of the bounding box, with a line drawn to the origin of $\mathbf{O_b}$.

Table 1: User-defined inputs for example viewing angle shown in Figure 5

Cameras	**Phantom v2640 UHS** -Sensor Size = [2048, 1920] px -Pixel Size = 0.0135 mm
Lenses	**Zeiss Milvus 2.8/18mm** -Minimum Focal Dist = 250 mm -Aperture Range = [22, 1.8]
	Zeiss Milvus 2.0/135mm -Minimum Focal Dist = 800 mm -Aperture Range = [22, 2.0]
Camera Orientation	Horizontal
Camera Stand Orientation	Horizontal
Stereo Angle (degrees)	30
Bounding Box [bx, by, bz] (mm)	[152.4, 206.4, 38.1]
Bounding Box Margin %(x,y,z)	[0, 0, 0]
Viewing Euler Angles (degrees; y,x',z'')	[-10,15,0]
SOD Range (mm)	[200:5:5000]
BL Range (mm)	[200:5:5000]
PD Range (mm)	[-20:5:30]
Minimum Focus Check Tolerance	0.95
DOF Check Tolerance	0.95
Lens Diffraction Check Tolerance	0.95
Acceptable Circle of Confusion (mm)	0.0135
Lighting Wavelength (mm)	6.00E-04

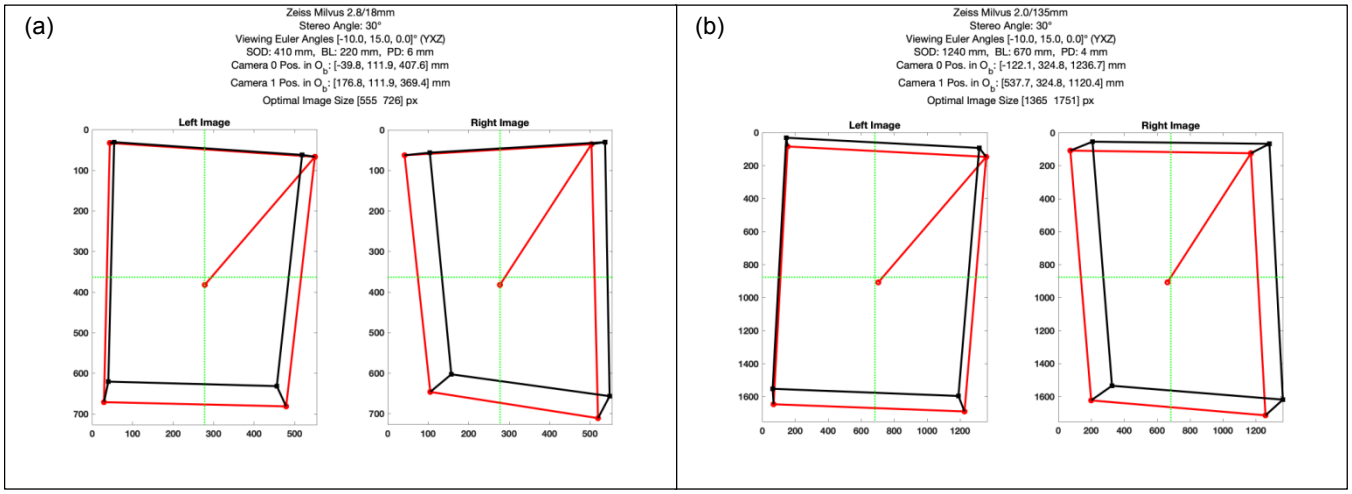

Figure 6: Example results for optimal pose configurations for the example viewing angle shown in Figure 5; (a) 18mm lens, (b) 135 mm lens

An example DIC test was performed on the BARC with the camera rig positioned directly in front of the test article, similar to the configuration depicted in Figure 4, as shown in Figure 7.

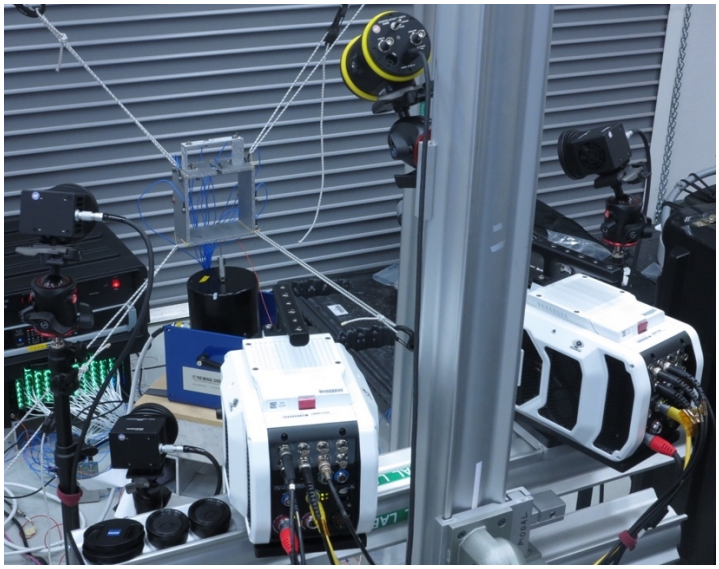

Figure 7: Example stereo DIC test of the BARC structure

The test planning tool was used to estimate the optimal camera position for this test using the settings in Table 1Table 3. The optimized pose results (specifically the camera intrinsic and extrinsic parameters that locate the cameras with respect to the world coordinate system) were imported into an image rendering software, Blender [14], along with the FEM of the BARC. The rendered images are overlaid on the planning tool results for the optimized setup in Figure 8, validating the selected pose for an idealized optical system and perfect camera placement in the real world.

However, in practice it is difficult to place the cameras in the exact locations specified by the planning tool output. Slight shifts in the actual placement are expected, along with asymmetry between the two cameras with respect to the rig origin. Another known difficulty with estimating the composition of the bounding box within the image frame is the phenomenon of lens "focus breathing" where the image composition changes as the focus distance is changed. Further confounding the issue is that, in some lenses, breathing will cause an apparent increase in magnification as focal distance increases while other lenses do exactly the opposite. The current tool only deals with this issue by applying a margin to the bounding box volume. As such, the user should be cognizant of the margin used and understand the degree to which the lenses available exhibit this behavior.

The as-tested pose configuration was quantified by performing a stereo camera calibration and importing the resulting intrinsic and extrinsic parameters (see Equation (1)) into Blender, as was done for the optimized configuration. The as-tested configuration was then synthesized within Blender allowing for the accurate calculation of the camera positions. The differences between the optimized and as-tested camera positions were found to be on the order of approximately 10%:

Table 2: Optimal vs. as-tested camera positions

Pose in Ob (mm)	Camera 0				Camera 1			
	Optimal	As-Tested	Error (mm)	Error (%)	Optimal	As-Tested	Error (mm)	Error (%)
X	-195	-219	24	13%	195	182	13	7%
Y	0	10	-10	--	0	9	-9	--
Z	728	768	-40	6%	728	777	-49	7%

The (non-optimal) output from the test planning tool that most directly corresponded to the as-tested configuration with the actual test images overlaid is shown in Figure 9. The effects of focus breathing are immediately observable, in that the estimated projections are not quite zoomed in far enough (apparent lens focal length is too small). The positioning errors are also observable, as evidenced by the actual test images not being exactly centered. Even with these issues, the test planning tool provided a very good first-order estimate of which lens to select and the corresponding camera positions which would properly frame the test object (refer to Figure 8); from this estimate a successful, but not perfect, test setup was easily created.

The positioning errors are strictly a function of how much time the user devotes to the test setup. The focus breathing phenomenon is the primary source of uncertainty (barring massive lens distortions). Future revisions of the planning tool may look to make the focal length of the lens a function of focus distance in the iterative optimization routine, although this would require very thorough experimental characterizations of each lens by the user. As a rough validation, the planning tool was re-

run with a lens focal length of 93 mm to account for focus breathing and the test images were artificially shifted to center the BARC within the image frames, as shown in Figure 10. While this is merely a qualitative comparison, it does provide a level of expectation that if focus breathing is accounted for and camera positioning is exact, the planning tool and actual images should align quite well.

Table 3: User-defined inputs for the stereo-DIC BARC test

Cameras	**Phantom v2640 UHS** -Sensor Size = [1536, 1920] px -Pixel Size = 0.0135 mm
Lenses	**Zeiss Milvus 1.4/85mm** -Minimum Focal Dist = 800 mm -Aperture Range = [16, 1.4]
Camera Orientation	Horizontal
Camera Stand Orientation	Horizontal
Stereo Angle (degrees)	30
Bounding Box [bx, by, bz] (mm)	[152.4, 206.4, 38.1]
Bounding Box Margin %(x,y,z)	[5, 5, 5]
Viewing Euler Angles (degrees; y,x',z'')	[0, 0, 0]
SOD Range (mm)	[200:5:5000]
BL Range (mm)	[200:5:5000]
PD Range (mm)	[-20:5:20]
Minimum Focus Check Tolerance	0.95
DOF Check Tolerance	0.95
Lens Diffraction Check Tolerance	0.75
Acceptable Circle of Confusion (mm)	0.0135
Lighting Wavelength (mm)	6.00E-04

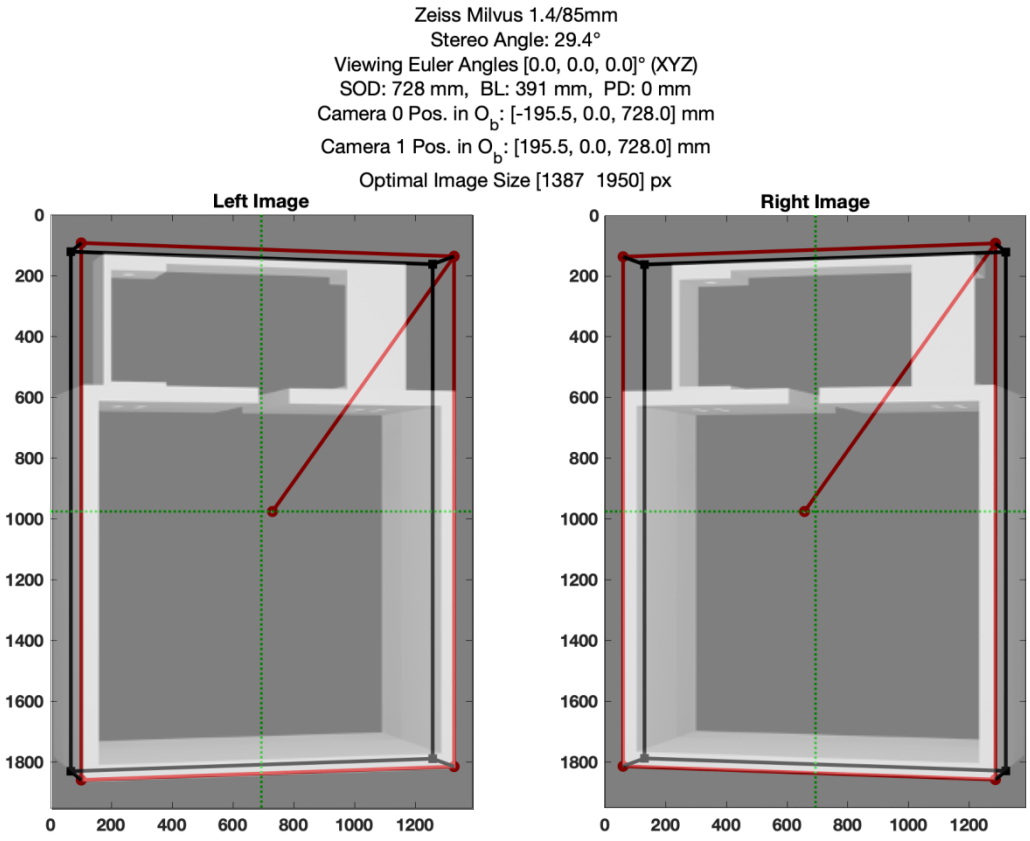

Figure 8: Optimized pose selected for BARC stereo DIC test

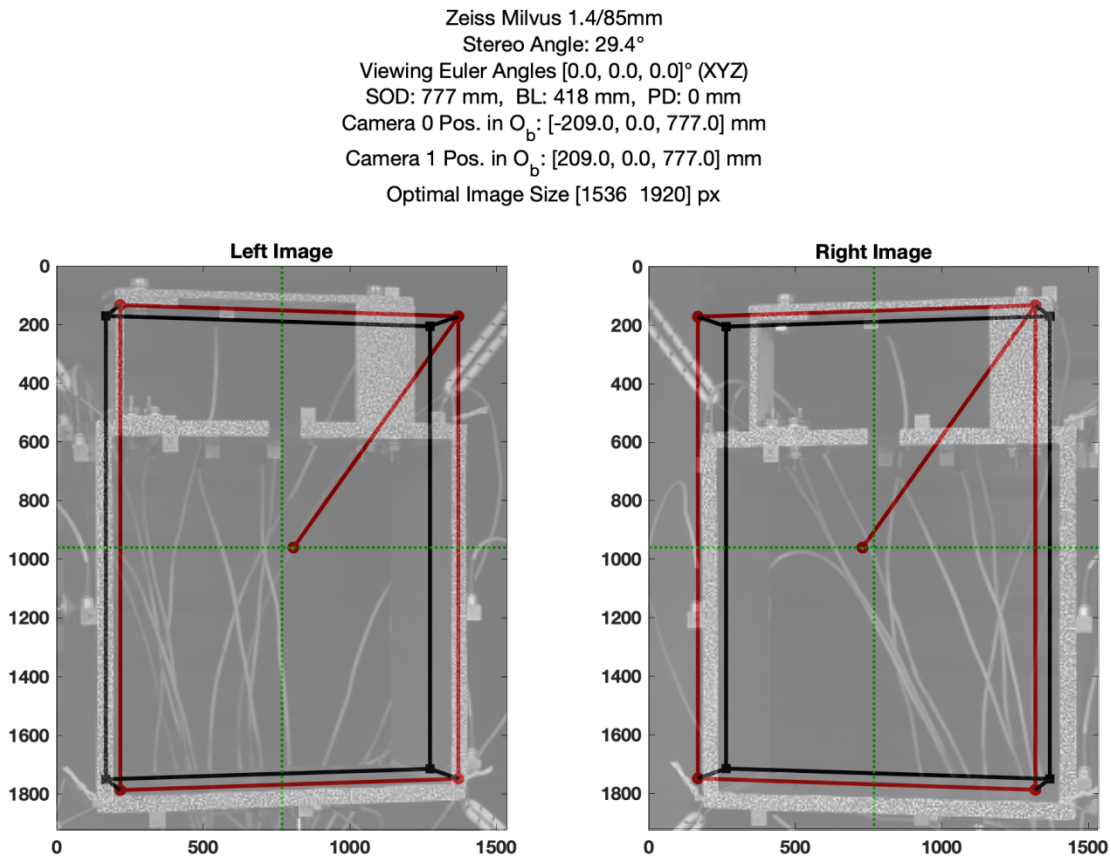

Figure 9: As-tested pose for stereo DIC BARC test

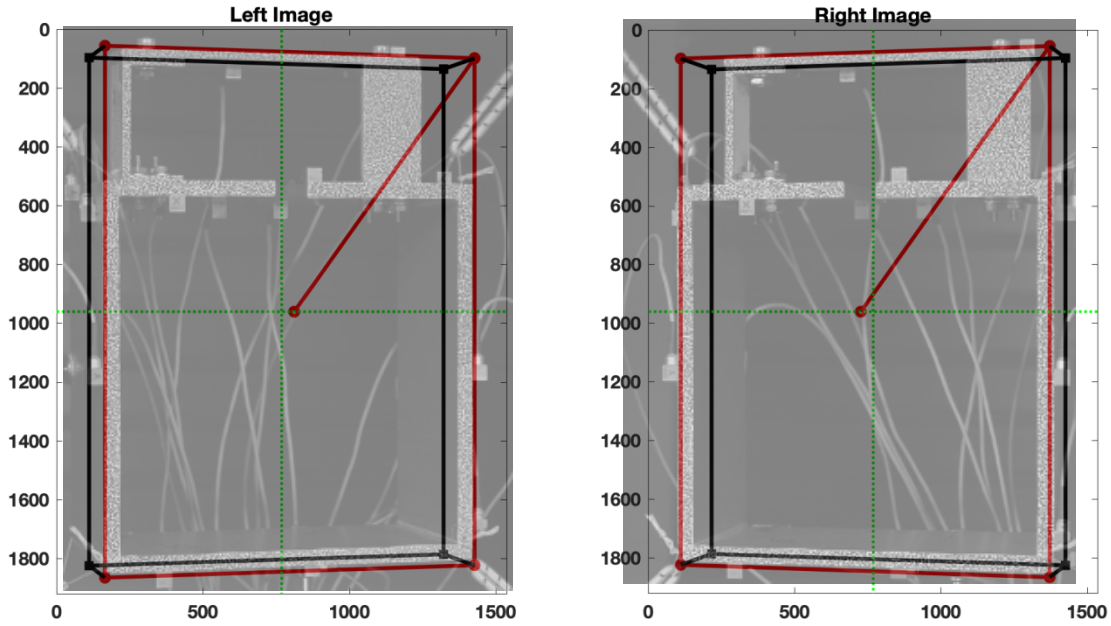

Figure 10: As-tested pose artificially adjusted for camera translation and focus breathing (f = 93 mm)

4 PREDICTING OBSERVABLE PIXEL DISPLACEMENTS WITH FEM

Once the pose optimization has been performed as outlined in Section 3, a final camera pose is selected, and the resulting parameters are fed into the next stage of pre-test analysis which utilizes the FEM. In this section, all quantities are defined in the world (FEM) coordinate system $\mathbf{O_w}$. As a separate analysis, this section utilizes different variable definitions from those provided in Section 3, and are defined at the point of use. This analysis would also be performed prior to setting up any test equipment; the test examples shown in the previous section were to demonstrate the effectiveness of the pose estimation algorithm.

4.1 Algorithm to Determine Node Visibility from Camera Placements

The mesh of the test article of interest is created using CUBIT, a meshing toolkit produced by Sandia National Laboratories (SNL) [15]. Within CUBIT, the test article mesh is exported as an EXODUS file (*.exo) compatible with Sierra Structural Dynamics (SD) analyses, a SNL-produced module for the Sierra engineering mechanics simulation code suite [16]. In the definition of the mesh of the test article, the global placement of the mesh should be consistent with the coordinate system definitions and placement of the bounding box around the finite element model (FEM) in the camera setup tool discussed in Section 3. In the definition of the mesh, surfaces that are candidates for measurement with digital image correlation (DIC) can be defined as sidesets. The CUBIT mesh of the example test article is shown in Figure 11(a).

The mesh information, including node coordinate, connectivity, and model output data from a normal modes analysis are imported into MATLAB. The nodes on the external sides of the test article model can be identified from the connectivity information to form a set of planes that represent the exterior of the test article, which is shown Figure 11(b). This information along with the sideset representing prospective measurement nodes, shown as red points in Figure 11(c), comprises the user information required in order to predict the observable nodal displacements from a camera position.

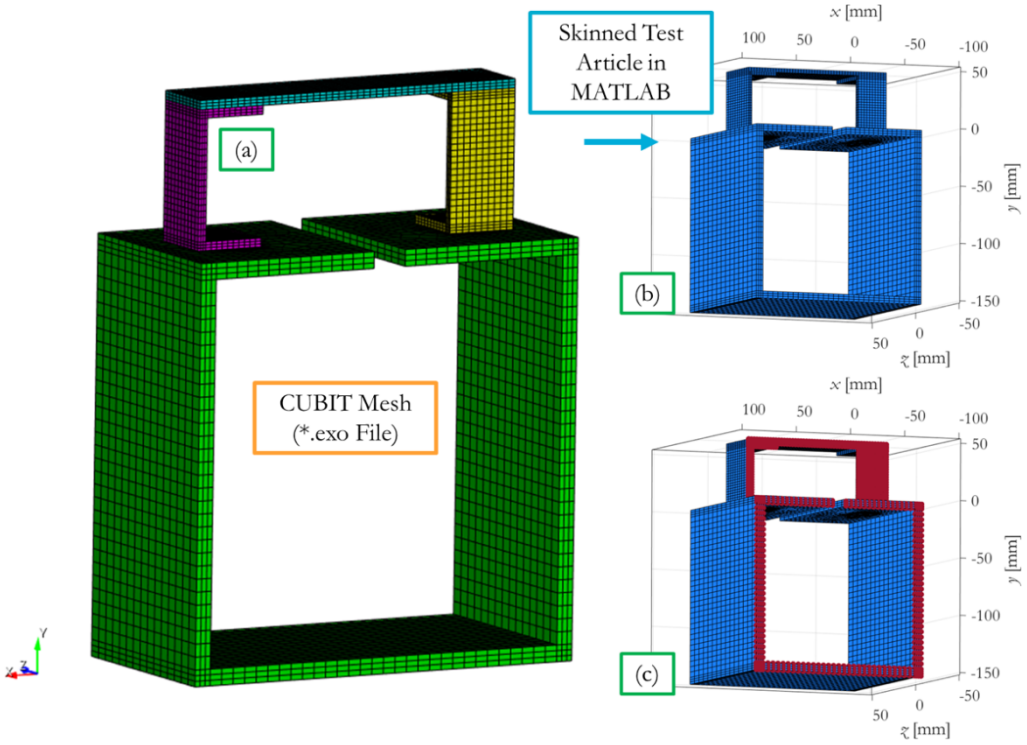

Figure 11: Initial steps to determine observable nodes in FEM of test article based on camera parameters and test setup with a (a) mesh defined in Cubit processed in MATLAB (b) without and (c) with the prospective measurement surface nodes highlighted

In addition to the definition of the set of nodes and planes that form the exterior surface, as well as the prospective measurement nodes, camera placements are designated to represent the test setup geometry as provided by the iterative optimization process from Section 3. Figure 12 shows how lines are formed from each of the cameras placed, Camera 1 (green point) and Camera 0 (yellow point), to each of the prospective nodes, shown as red points, for the example digital image correlation setup and determination of visible nodes. This tool currently assumes small displacements, such that excitation of the part does not cause nodes to go in/out of visibility.

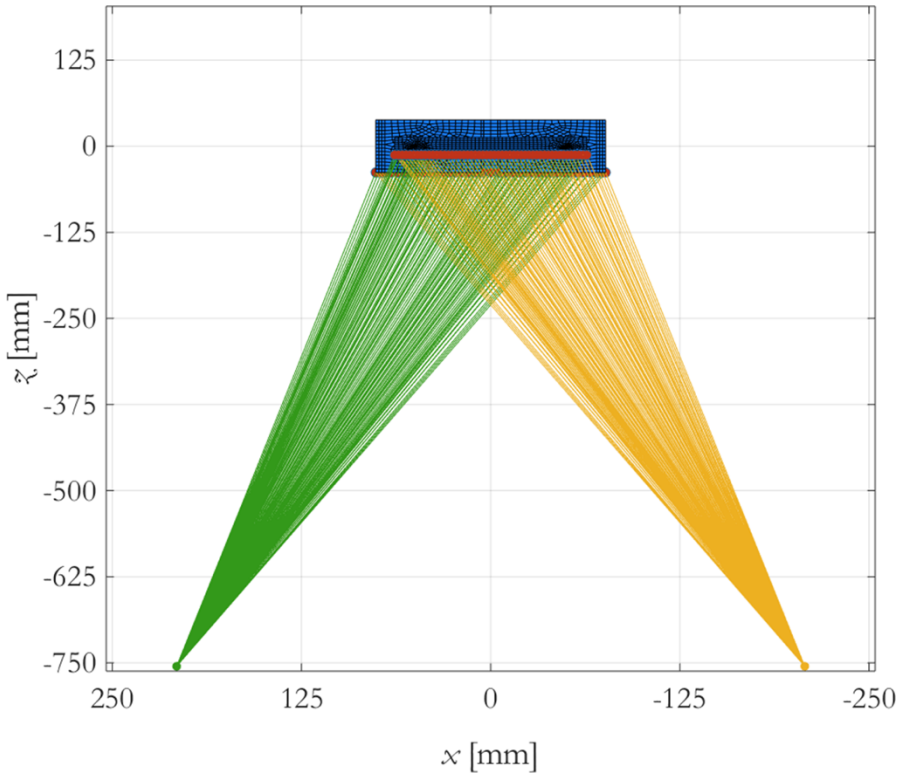

Figure 12: Camera placement and definition of lines representing camera views of each node in a prospective speckling area

The camera placements shown in Figure 12 are an effective set for observing the prospective measurement points chosen. Figure 13 shows a poorly placed camera position (orange point, behind the area of interest) viewing the single red measurement point to show the detection of nonvisible nodes due to view obstructions. The green inset of Figure 13 shows the view behind the poor camera placement, which highlights the detected intersection point in green.

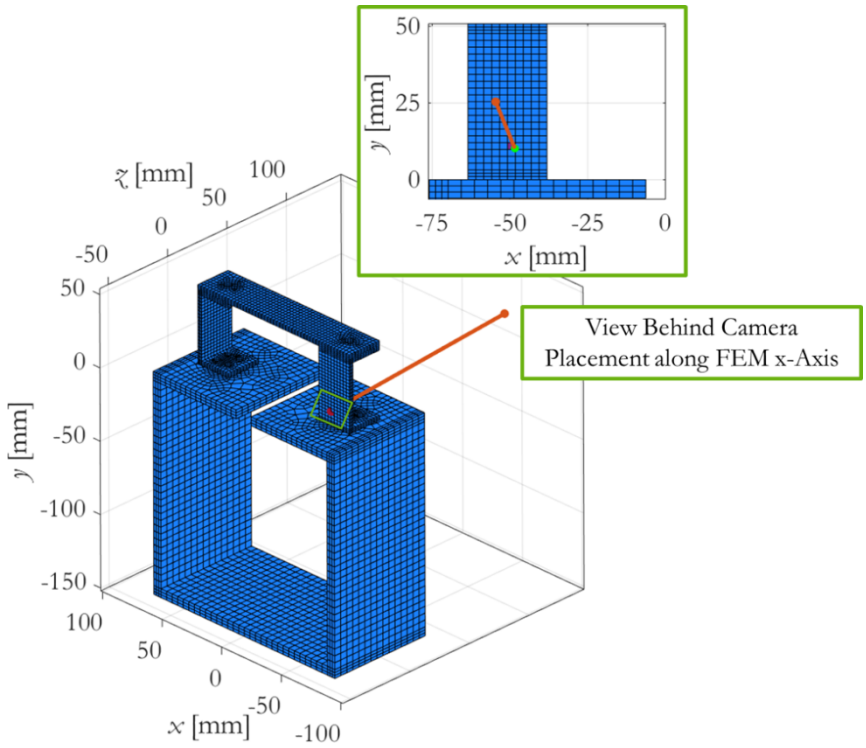

Figure 13: Definition of single view line from camera to a node of interest from a poorly placed camera relative to surface of interest

Any intersection of a view line with an exterior surface of the model is calculated by checking to see if the view line intersects with any of the planes forming the exterior surface and if the intersection point falls within the edges of the element face being represented by the plane. The intersection shown in the inset of Figure 13 is shown in greater detail in Figure 14. Once the test article is skinned and prospective measurement nodes are defined, the toolset starts with determining if any intersections occur for the view line, or ray, from the camera position to each of the nodes of the prospective measurement set. The toolset can be used for hexahedral and tetrahedral meshes without significant user input since the script detects what mesh is used for each component of the test article. A slightly different approach is leveraged to detect what prospective measurement nodes are obstructed depending on the mesh type.

The intersection example shown in Figure 14 is based on a hexahedral mesh, which is implemented on the BARC test article. The Edge 1, $\mathbf{v_{12}}$, is defined between Vertex 1 (Yellow Point) and Vertex 2 (Red Point) and Edge 2, $\mathbf{v_{14}}$, is defined between Vertex 1 (Yellow Point) and Vertex 4 (Gray Point). Using the edge definitions, the plane normal can be defined as:

$$\mathbf{n} = [D,E,F] = \mathbf{v_{12}} \times \mathbf{v_{14}} \tag{15}$$

where D, E, and F denote the components of the normal vector for the plane being intersected. Once the normal c=vector for the current plane is determined, the general form of the equation of the plane can be defined as:

$$D\mathbf{x} + E\mathbf{y} + F\mathbf{z} = 0 \tag{16}$$

which can be defined with the parametric definition of the camera view line to determine if the view line intersects the plane, is parallel to the plane, or exists in the plane when extending the plane to infinity.

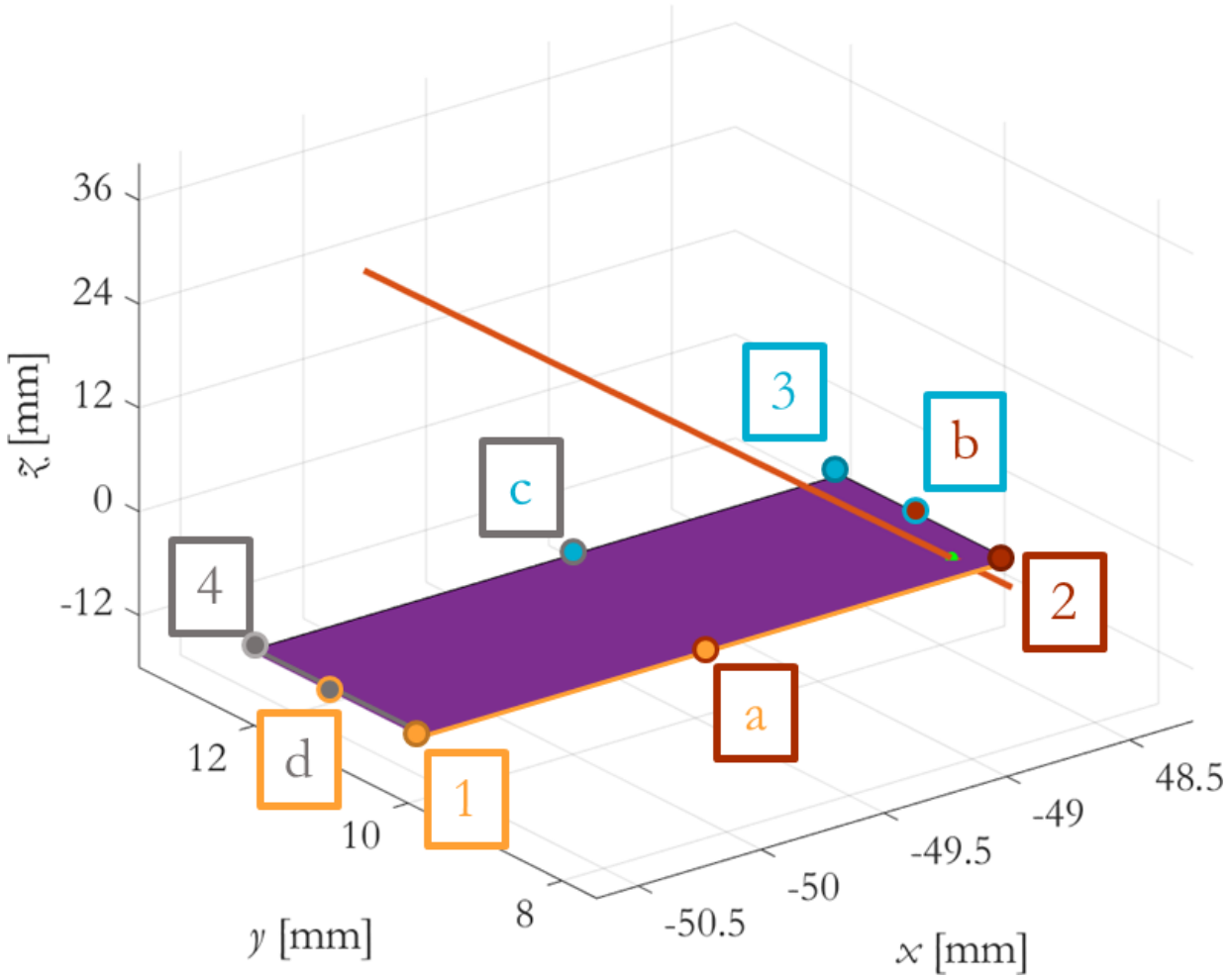

Figure 14: An example of an intersection point and intersect exterior plane on the FEM of the test article based on poor camera placement

The parametric line definition for the camera view is:

$$
\begin{aligned}
x_{CV}(s) &= x_c + Us \\
y_{CV}(s) &= y_c + Vs \\
z_{CV}(s) &= z_c + Ws \\
\mathbf{u_{Camera}} = [U,V,W] &= [x_{mn} - x_c, y_{mn} - y_c, z_{mn} - z_c]
\end{aligned}
\tag{17}
$$

where $\mathbf{u_{Camera}}$ denotes a vector representing the camera optical axis, $[x_c, y_c, z_c]$ the camera position in cartesian coordinates shown as the orange point in Figure 13, and $[x_{mn}, y_{mn}, z_{mn}]$ is the location of the measurement node shown as the red point in Figure 13. If $\mathbf{u_{Camera}}$ is perpendicular to $\mathbf{n}$, the view line lies within the plane or is parallel to the plane at an offset. If the following equality is satisfied with $\mathbf{u_{Camera}}$ perpendicular to $\mathbf{n}$:

$$
\mathbf{n} \cdot \mathbf{r_{Camera}} - \mathbf{n} \cdot \mathbf{r_1} = 0 \tag{18}
$$

where $\mathbf{r_1}$ is the vector from the origin of the global cartesian coordinate frame to Vertex 1 and $\mathbf{r_{Camera}}$ the vector from the origin to the camera placement, then the camera view line is contained within the plane. Otherwise, $\mathbf{u_{Camera}}$ being perpendicular to $\mathbf{n}$ implies that the camera view line is parallel to the current plane being analyzed on the exterior surfaces of the test article.

If the dot product of $\mathbf{u_{Camera}}$ and $\mathbf{n}$ is nonzero, the infinite representation of the plane and camera optical axis intersect at a single point. The intersection point is defined as:

$$
\mathbf{r_{Intersection}} = \mathbf{r_{Camera}} - \mathbf{u_{Camera}}\left(\frac{-\mathbf{n} \cdot \mathbf{r_1} + \mathbf{n} \cdot \mathbf{r_{Camera}}}{\mathbf{n} \cdot \mathbf{u_{Camera}}}\right) \tag{19}
$$

where $\mathbf{r_{Intersection}}$ defines the vector from the origin to the intersection point (I), which is shown as the green point in the plane in Figure 14. Up to this point in the calculations, the only difference between processing tetrahedral and hexahedral meshes is the number of edges of the plane being reduced by one, due to the element faces being triangles instead of quadrilaterals, and Edge 2 being defined between Vertex 1 and Vertex 3 in a tetrahedral mesh in the above calculations.

When determining if the intersection point is contained within the edges of an exterior element face of the model, the methods differ slightly depending on the mesh type. For a hexahedral mesh, Varignon's theorem is used alongside Heron's formula to determine if the intersection point is contained on the element face of interest [17]. Varignon's theorem is used to calculate the area of the face of a hexahedral mesh that can be a general quadrilateral, which states for Figure 13 that the area of the element face being represented by the finite plane is given as the sum of the area of the parallelogram $\overline{abcd}$ and triangles $\overline{1ad}$, $\overline{2ba}$, $\overline{3cb}$, and $\overline{4dc}$. The points *a, b, c,* and *d* in Figure 14 are defined as follows:

$$
\begin{aligned}
\mathbf{r_a} &= \frac{1}{2}(\mathbf{r_1} + \mathbf{r_2}) \\
\mathbf{r_b} &= \frac{1}{2}(\mathbf{r_2} + \mathbf{r_3}) \\
\mathbf{r_c} &= \frac{1}{2}(\mathbf{r_3} + \mathbf{r_4}) \\
\mathbf{r_d} &= \frac{1}{2}(\mathbf{r_4} + \mathbf{r_1})
\end{aligned}
\tag{20}
$$

where $\mathbf{r_a}$, $\mathbf{r_b}$, $\mathbf{r_c}$, and $\mathbf{r_d}$are the vectors from the origin to points *a*, *b*, *c*, and *d*, respectively. In Equation (20), $\mathbf{r_2}$, $\mathbf{r_3}$, and $\mathbf{r_4}$ denote the vectors from the origin to points 2, 3, and 4 in Figure 14, respectively.

This area can then be compared to the sum of the areas of the four triangles formed from the intersection point and the vertices of the plane of interest ($\overline{1I2}$, $\overline{2I3}$, $\overline{3I4}$, and $\overline{4I1}$), which denotes that the intersection point is on the element face if equal. The area of all the triangles can be calculated with Heron's formula:

$$
\begin{aligned}
s_P &= \frac{1}{2}(T_{S1} + T_{S2} + T_{S3}) \\
A_T &= \sqrt{s_P(s_P - T_{S1})(s_P - T_{S2})(s_P - T_{S3})}
\end{aligned}
\tag{21}
$$

where s_P denotes the semiperimeter of the triangle, A_T the area of the triangle, and T_{S1}, T_{S2}, and $T_{S3}$the side lengths of the triangle [17]. In the case of a tetrahedral mesh, Heron's formula is used in place of Varignon's theorem to calculate the area of the plane representing the current element face being analyzed based on the camera view line. Singularities associated with Heron's formula are not expected to cause issues when a quality mesh is used with low aspect ratios throughout. The above process is iterated throughout all the exterior faces of the test article mesh for each view line specified based on the optimized camera rig position (the final selection from the pose optimization routine in Section 3) and prospective measurement nodes. Additional constraints include accounting for intersection points that occur beyond the length of the camera view line (i.e. behind the measurement node) and verifying that the intersection point is the prospective measurement node. The edge of the camera view is enforced by leveraging the information about the image size and pixel transformation from the camera setup tool discussed in Section 3.

Figure 15 shows an example of the impact of camera placement on the viewable nodes in a prospective set of measurement nodes. The red lines in Figure 15 are the sideset representing the nodes that are contained in the area that will be patterned for DIC and the purple, orange, and green points represent possible camera placements denoted as 1, 2, and 3, respectively. Placing a camera in a obscured view or too close to the part doesn't allow for the complete set of measurement nodes to be observed and causes the nodes with maximum overall displacement to be missed in the camera, which is shown by comparing the view from Camera Position 3 (green inset) to either Camera Position 1 (purple inset) or Camera Position 2 (orange inset). This simple example shows the toolset is functional with minimum user input, which allows for it to be leveraged in iterative optimization process as well as automated setup procedures.

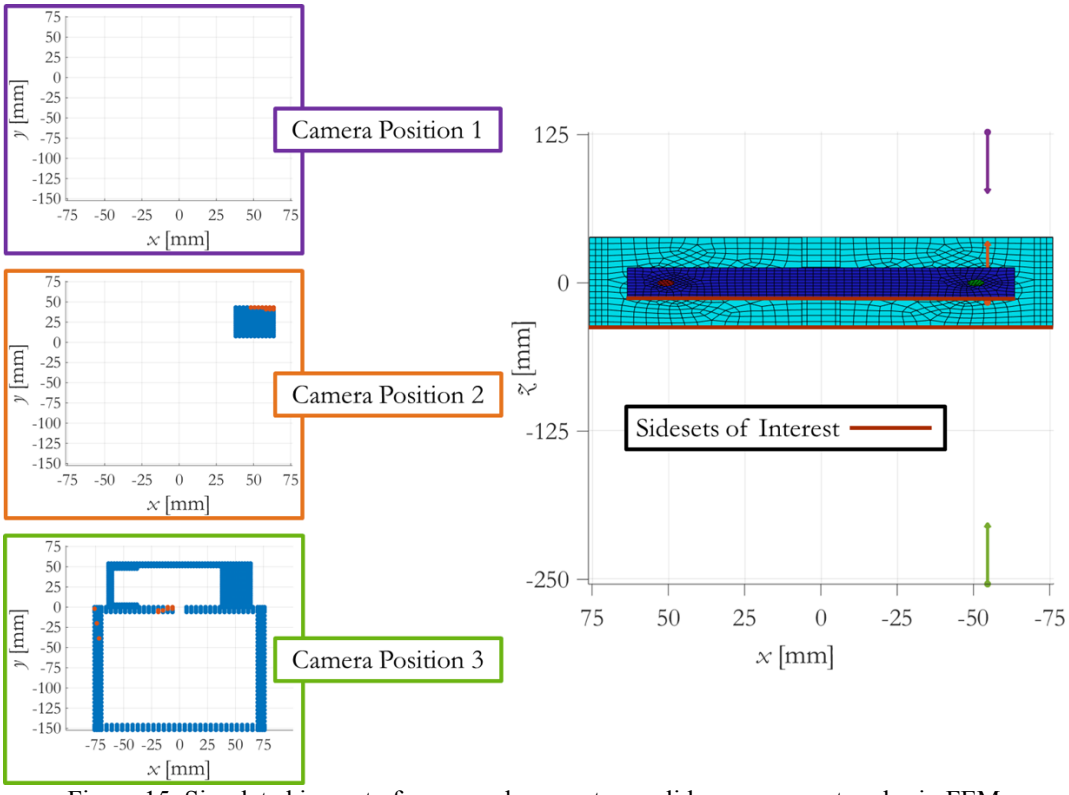

Figure 15: Simulated impact of camera placement on valid measurement nodes in FEM

4.2 Definition of Frequency Response Functions to Assess Arbitrary Inputs

Figure 16 shows the representative model definition for the BARC test setup shown in Figure 7. As stated in Section 4.1, the mesh was created in CUBIT. The final hexahedral mesh is shown in blue in Figure 16 and consists of 8-node hexahedral elements (HEX8). Once the mesh was created, an eigenanalysis in Sierra SD was utilized to determine the modes of the structure using the *eigen* solution type and to inform a subsequent *modalfrf* study [16]. Figure 16 also shows the prospective measurement nodes in red.

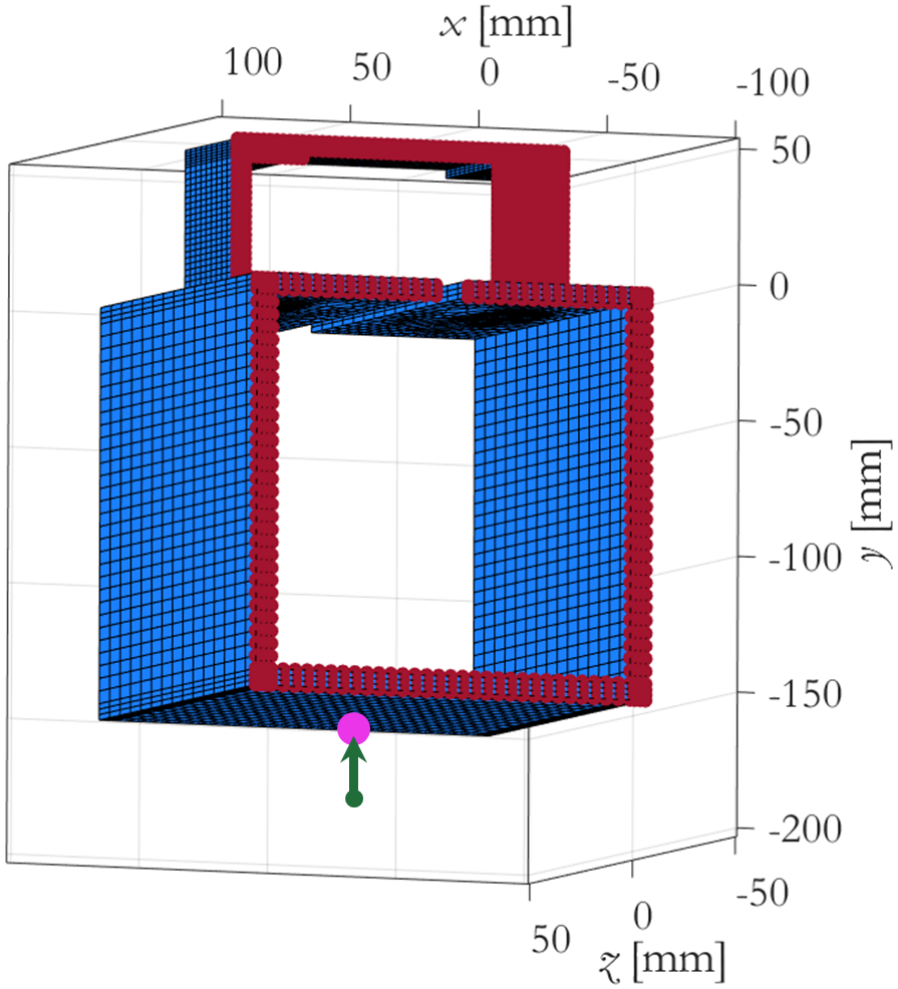

Figure 16: Representation of BARC FEM for Sierra SD analysis

The modal analysis for the BARC test article in Sierra SD was defined with free-free boundary conditions to replicate the test configuration:

$$0 = ((\mathbf{K} - \sigma\mathbf{M}) - \boldsymbol{\mu}\mathbf{M})\boldsymbol{\phi} \tag{22}$$

where $\mathbf{K}$ denotes the stiffness matrix, $\mathbf{M}$ the mass matrix, $\boldsymbol{\phi}$ the eigenvectors, $\boldsymbol{\mu}$ the eigenvalues of the shifted problem with shift σ, which must be a large negative value to ensure $\mathbf{K} - \sigma\mathbf{M}$ is nonsingular without being too large to impact the eigenvalue solver [16]. The shifted problem shown in Equation (22) is used with *eigen* in Sierra SD since $\mathbf{K}$ is singular in the unshifted problem. Original eigenvalues can be obtained with the simple expression:

$$\boldsymbol{\omega} = (\boldsymbol{\mu} + \sigma)^2 \tag{23}$$

where $\boldsymbol{\omega}$ are the original eigenvalues of the unshifted problem. Figure 17 provides the first six flexible modes from the modal analysis of the BARC model in Sierra SD.

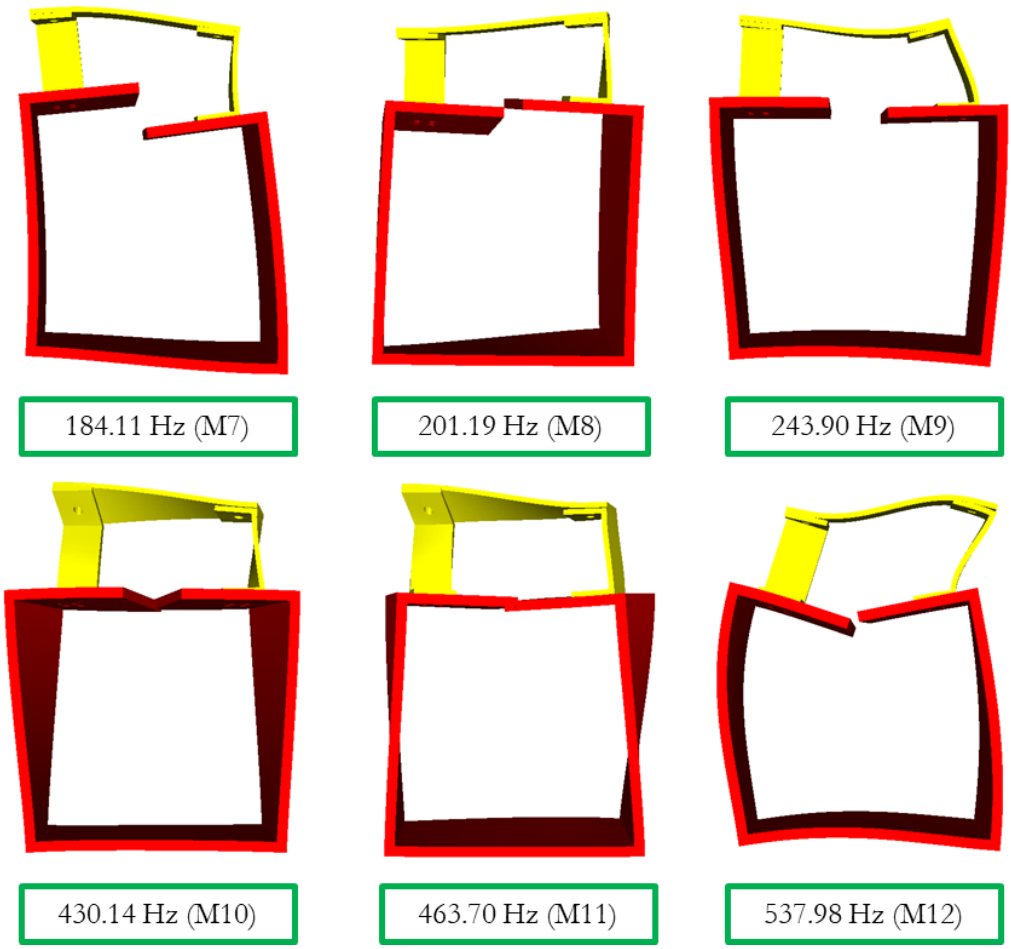

Figure 17: First six analytical flexible modes of the BARC structure

The inputs for the simulation are applied to the node closest to the location where the shaker is attached in the experimental test setup being represented, which is shown as the pink point in Figure 16. The axis of the shaker is parallel to the *y*-axis, which is shown as the green arrow in Figure 16. The mass of the input location is approximated as a portion of the mass of the elements containing the node located at the pink point in Figure 16. Bolts of the joints connecting the removable component (yellow component in Figure 17) are defined using Joint2G elements, which provides the ability to define the constitutive behavior for each degree of freedom of the bolt represented by a node pair. The bolt representation is then connected to the bolt holes in CUBIT using the Spider command, which uses rigid bar elements, RBARS in Sierra SD, to represent the connections [15, 16]. A uniform modal damping ratio of 0.02 is applied to all real modes.

To determine the motion of each of the prospective measurement nodes, a frequency response function (FRF) study is completed in Sierra SD with the *modalfrf* solution type to define the transfer function matrix relating the input acceleration, or force, from the shaker to the displacement of the prospective measurement nodes. The response, or output, spectral density function matrix, $\mathbf{S_{mm}}$, due to an input acceleration at the shaker drive point, $\mathbf{S_{nn}}$, is expressed as:

$$\mathbf{S_{mm}}(\omega) = \mathbf{H_{mn}}(\omega)\mathbf{S_{nn}}(\omega)\mathbf{H_{mn}^{H}}(\omega) \tag{24}$$

where $\mathbf{H_{mn}}$ denotes the transfer function matrix between the response accelerations at the prospective measurements points and the input acceleration due to the shaker obtained with the FRF study with Sierra SD, and $^{\mathbf{H}}$ denotes the complex conjugate transpose [18, 19]. Figure 18 provides the input auto-power spectral density (APSD) for the drive point acceleration measured in test to the node closest to the attachment point. Figure 19 shows a representative example for the resulting response of a prospective measurement node (Node 12188), whose location is shown in the inset, for the *x*-axis (blue line), *y*-axis (orange line), and *z*-axis (yellow line).

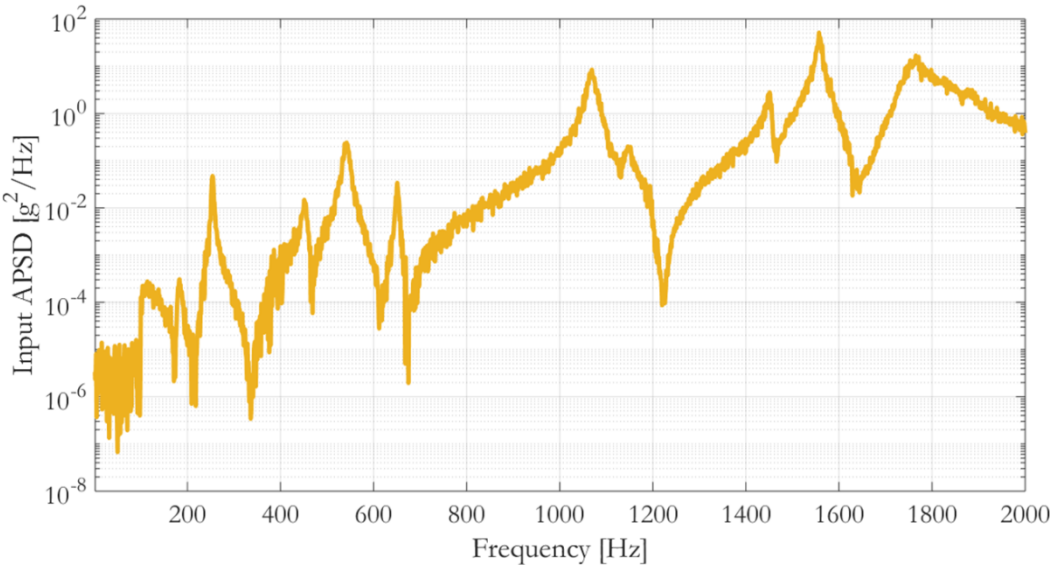

Figure 18: Auto-spectral density function of input acceleration

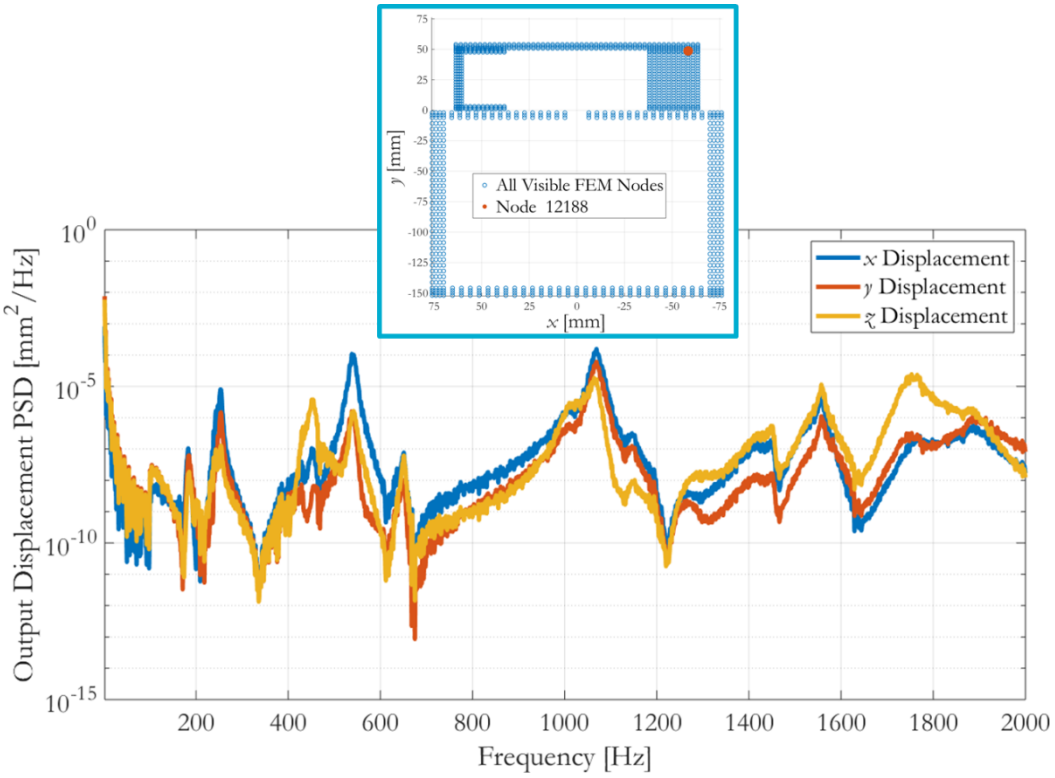

Figure 19: Output APSD for a single node of the BARC FEM

4.3 Transformation to Modal Coordinates to Acquire Pixel Displacements for Modes of Interest

The root mean square (RMS) values arising from the responses at each measurement node obtained from the results of the FRF study can be used to determine which of the prospective nodes displace enough to surpass an expected noise floor of the optical system [8]. From experience, a displacement of 1/100 of a pixel can be used to extract modal information (in optimal cases as low as 1/1000 of a pixel). The BARC test setup had an average of 8.8 px/mm for both cameras, resulting in an estimated noise floor of approximately 1.1E-3 mm. Modal filtering can be used to obtain the displacement of each pixel due to a mode of interest, which may then be compared directly to the noise floor to determine if that mode will be extractable. In the example shown in this section, the modes of interest are shown in Figure 20.

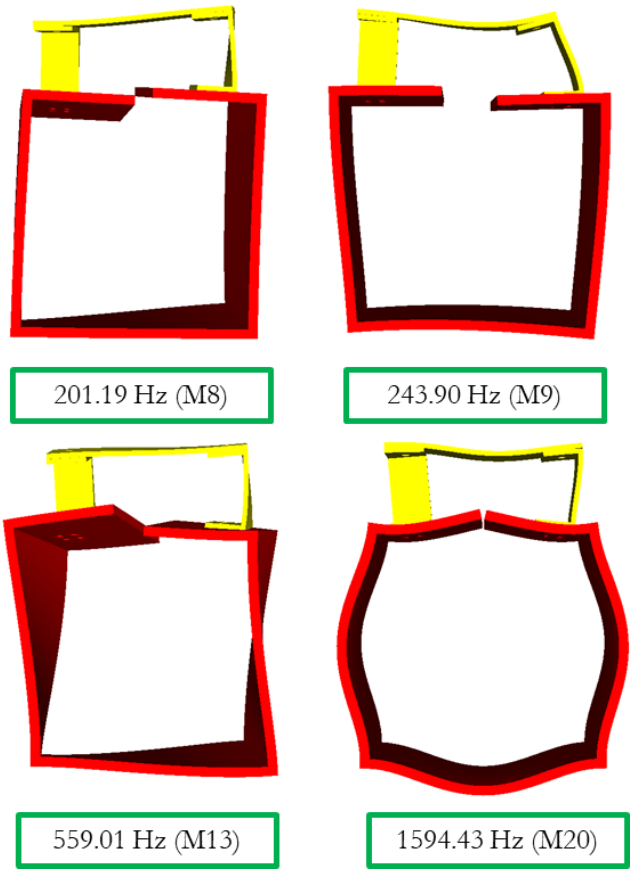

Figure 20: Example mode shapes of interest in determining what nodal displacements surpass the assumed noise floor

A full modal filter can be applied to obtain the responses at each of the prospective measurement points in modal coordinates with the complete set of modes obtained with Sierra SD:

$$\mathbf{x} = \mathbf{\Phi q} \tag{25}$$

where $\mathbf{x}$ denotes the displacements for prospective measurement nodes in cartesian coordinates due to the applied APSD, $\mathbf{q}$ denotes the displacements for prospective measurement nodes in modal coordinates, and $\mathbf{\Phi}$ is the complete mode shape matrix containing the rigid body and flexible modes. The displacements in the modal coordinates can be obtained by calculating the pseudoinverse of $\mathbf{\Phi}$, or $\mathbf{\Phi}^{+}$, and pre-multiplying Equation (25) by the resulting matrix:

$$\mathbf{q} = \mathbf{\Phi}^{+}\mathbf{x} \tag{26}$$

The displacement in cartesian coordinates for a specific mode can then be obtained as:

$$\mathbf{x}_{i\mathrm{RMS}} = \mathrm{RMS}(\mathbf{\Phi}_i \mathbf{q}_i) \tag{27}$$

where $\mathbf{\Phi}_i$ denotes the mode shape for a single mode at the prospective measurement nodes, $\mathbf{q}_i$ the response displacements for the measurement nodes due to the current mode of interest i in modal coordinates, and $\mathbf{x}_{i_{\mathrm{RMS}}}$ the root-mean-square (RMS) response displacement due to the mode of interest in physical cartesian coordinates. Using Equations (25)-(27), the displacements of the prospective measurement nodes due to the modes of interest shown in Figure 20 are calculated and compared to the noise floor to determine which nodes experience displacements that are observable; this result is shown in Figure 21.

Additionally, the transformation to pixel coordinates can be completed to determine what nodes are indistinguishable from one another based on the camera placements, which is also enforced during this stage of the process. This transformation can be defined using the derivations from Section 3:

$$\mathbf{u}_i = \mathbf{K}_i\mathbf{R}_{i\mathbf{v}}[\mathbf{I}|\mathbf{C}]_i\mathbf{R}_{\mathbf{vb}}\mathbf{R}_{\mathbf{bw}}\mathbf{X}_\mathbf{b}$$

$$\begin{bmatrix} au \\ av \\ a \end{bmatrix}_i = \begin{bmatrix} f_x & q & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}_i \begin{bmatrix} r_{xx} & r_{xy} & r_{xz} \\ r_{yx} & r_{yy} & r_{yz} \\ r_{zx} & r_{zy} & r_{zz} \end{bmatrix}_{iv} \begin{bmatrix} 1 & 0 & 0 & C_x \\ 0 & 1 & 0 & C_y \\ 0 & 0 & 1 & C_z \end{bmatrix}_i \begin{bmatrix} r_{xx} & r_{xy} & r_{xz} & 0 \\ r_{yx} & r_{yy} & r_{yz} & 0 \\ r_{zx} & r_{zy} & r_{zz} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}_{vb} \begin{bmatrix} r_{xx} & r_{xy} & r_{xz} & t_x \\ r_{yx} & r_{yy} & r_{yz} & t_y \\ r_{zx} & r_{zy} & r_{zz} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}_{bw} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}_w \quad (28)$$

Accounting for the pixel transformation to Camera 1 (green point in Figure 12) only reduces the number of visible nodes by 2.3% before comparing the displacements to the noise floor. It is estimated that 91% of the total prospective measurement points have observable displacements for Mode 8 (201 Hz) in comparison to 58%, 64%, and 29% for Modes 9 (243 Hz), 13 (559 Hz), and 20 (1594 Hz), respectively. A majority of the points do not have observable motion in the removable component for Mode 20 since the motion is dominated by the bottom box.

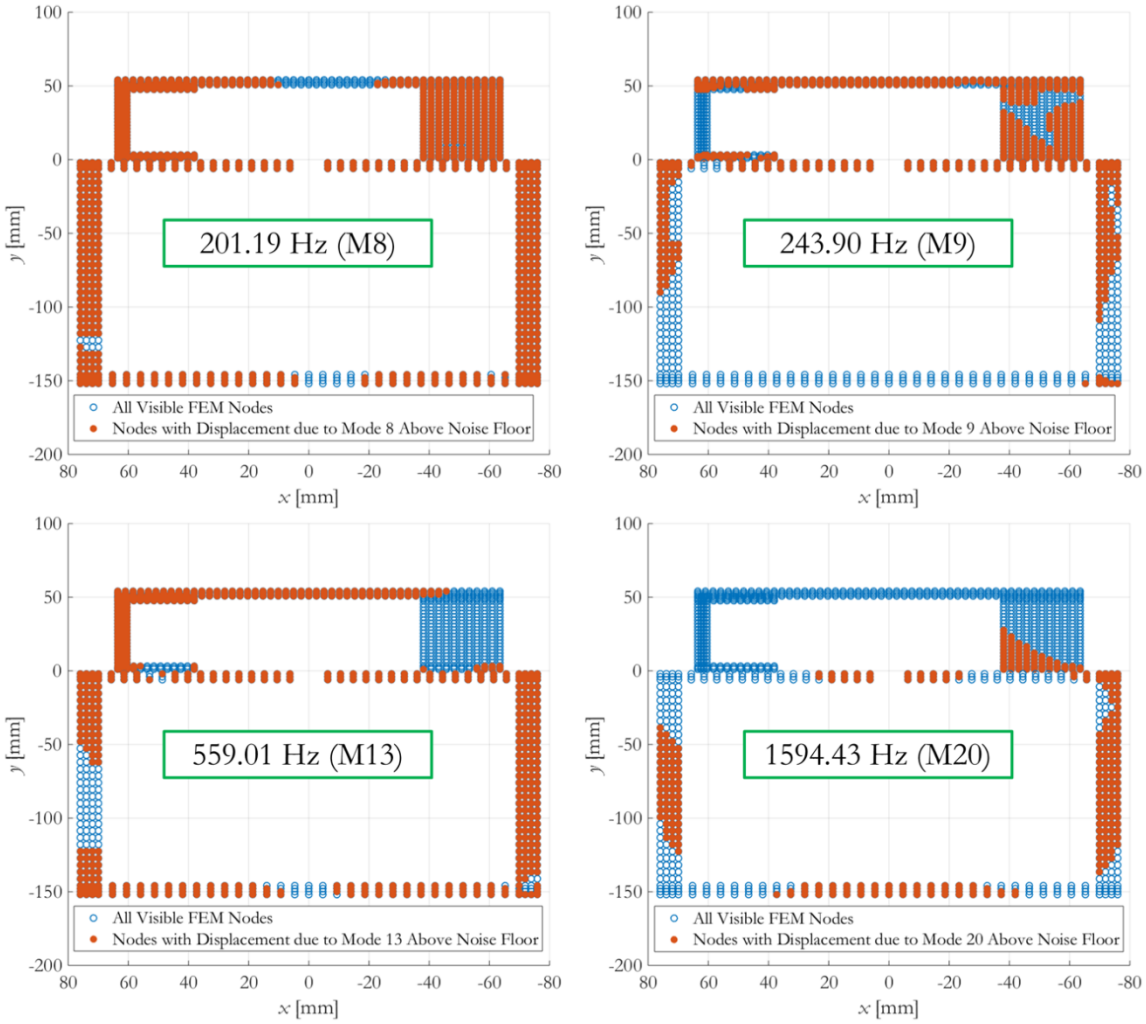

Figure 21: Comparing nodal displacement to noise floor for example modes of interest

Alternatively, specific measurement points can be chosen to determine what modes do not result in observable motion for the chosen point. The locations of Node 8174 (green point) and Node 16296 (yellow point) are shown in Figure 22. The resulting RMS displacements for Nodes 8174 and 16296 as a function of mode index is provided in Figure 23. Node 8174 experiences detectable motion for 56% of the first 50 modes compared to the 50% for Node 16296. The percentage of modes that are indicated to be observable via this evaluation is highly dependent on which node is selected.

Evaluating the observable modes for prospective measurement nodes allows for the best area or points of the model to be chosen to measure specific modes of interest. The tools developed in this section can be used to validate measurement surfaces

chosen or to leverage alongside an iterative optimization process to automate the selection of the optimal measurement points/area for a mode of interest.

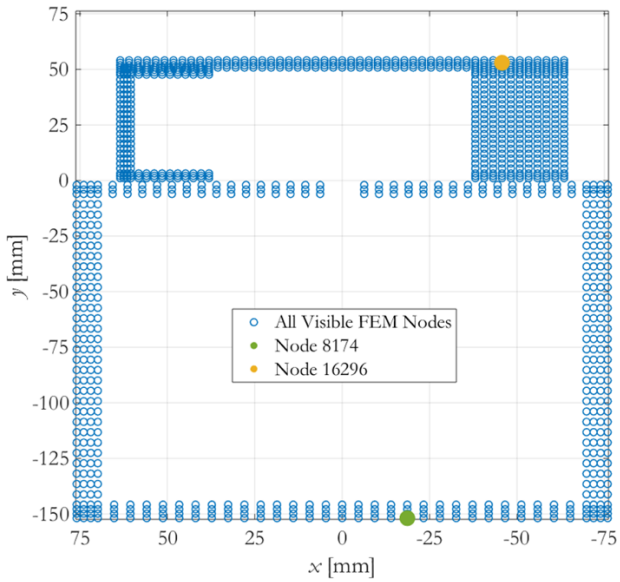

Figure 22: Locations of nodes used in modal displacement comparison to the RMS noise floor for the first 50 flexible vibration modes

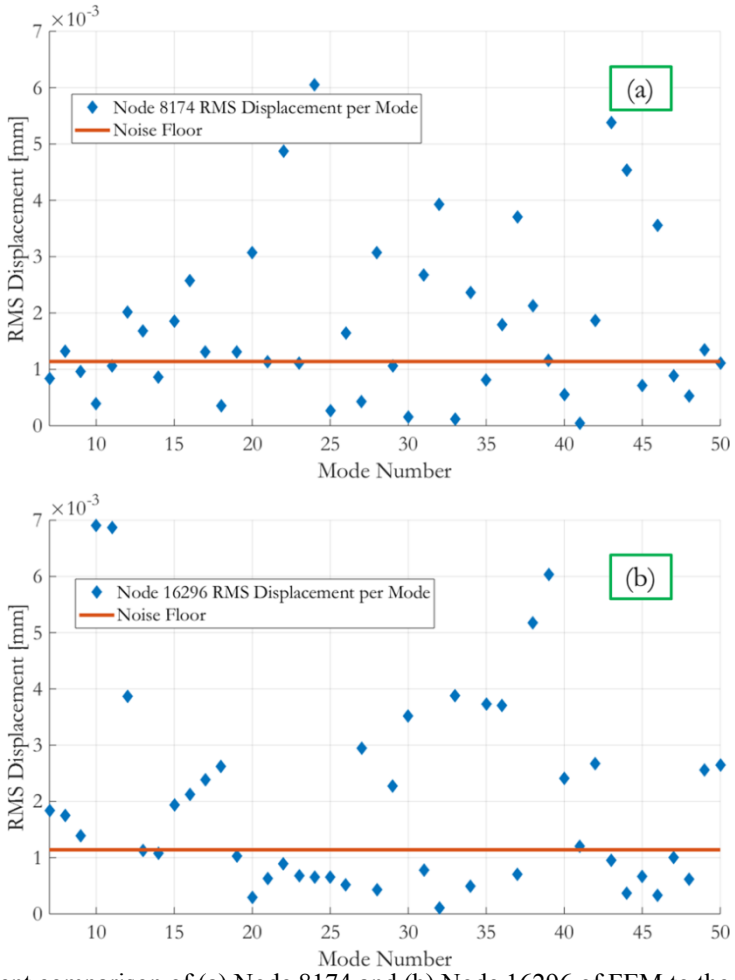

Figure 23: Modal displacement comparison of (a) Node 8174 and (b) Node 16296 of FEM to the RMS noise floor for the first 50 flexible vibration modes

5 CONCLUSIONS AND FUTURE WORK

A framework for iteratively determining a first-order approximation of a set of optimized stereo camera poses, given the optical hardware available to the practitioner, has been presented and demonstrated experimentally. The two primary difficulties lie in the ability to actually place the cameras in the exact locations specified by the planning tool results, and in the phenomenon of lens "focus breathing."

The placement of the cameras is not a technical issue, but rather one of practicality; getting the cameras within centimeters of the optimal position was not found to be overly difficult, but certainly the time that would be spent getting much more accurate would outweigh the benefits. The authors are currently exploring the use of a portable coordinate measurement machine for test setups requiring very precise positioning of the cameras relative to the test object.

The current implementation of the planning tool only deals with lens breathing through the implementation of a margin around the bounding box volume. While effective, it leaves room for improvement, the most straightforward of which could be to treat the lens focal length as a function of the focus distance within the iteration. Unfortunately, due to the variations between lens designs, this would require the user to conduct a characterization of each lens to establish this relationship.

The pose optimization framework by itself is useful to a practitioner, but the addition of the FEM analysis presented in Section 4 provides an additional level of utility. The ability to check for occluded areas before testing can potentially save valuable test time. Clearly, the end analysis in this work is oriented towards experimental modal testing but could be applied to any analysis where displacements are a quantity of interest. Future developments could easily include strain estimations as well. Another future development being investigated is to use the FEM to determine the measurement points anywhere around a test object that are required for modal characterization (e.g. effective independence), then determine the minimal set of camera poses necessary to measure them all.

The ultimate goal of a stereo camera pre-test analysis should be to not only determine where to put the equipment, but also if the locations and quantities of interest will be observable – this can only be achieved with the additional FEM analysis. Considering the alternative for pre-test planning is a "guess-and-check" method where the practitioner must rely on their experience to create a viable test setup, the tools presented herein mark an appreciable improvement, even in their current state. Given the high level of automation and minimal user inputs, the pre-test analysis presented has been found to save more time that it consumes.

6 REFERENCES

1 Sutton M, Orteu JJ, Schreier H (2009) Image correlation for shape, motion and deformation measurements: basic concepts, theory and applications. Springer, US. https://doi.org/10.1007//978-0-378-78747-3.

2 Jones EMC, Iadicola MA (eds.) (2018) A good practices guide for digital image correlation. International Digital Image Correlation Society. https://doi.org/10.32720/idics/gpg.ed1

3 Reu P (2012) Introduction to digital image correlation: best practices and applications. Experimental Techniques 36:3-4. https://doi.org/10.1111/j.1747-1567.2011.00798.x

4 Reu P, Miller T (2008) The application of high-speed digital image correlation. Journal of Strain Analysis for Engineering Design 43:673-688. https://doi.org/10.1243/03093247JSA414

5 Helfrick M, Niezrecki C, Avitabile P, Schmidt T (2011) 3D digital image correlation methods for full-field vibration measurement. Mechanical Systems and Signal Processing 25:917-927. https://doi.org/10.1016/j.ymssp.2010.08.013

6 Molina-Viedma AJ, Lopez-Alba E, Felipe-Sese L, Diaz FA (2017) Full-field modal analysis during base motion excitation using high-speed 3D digital image correlation. Measurement Science and Technology 28(10):105402. https://doi.org/10.1088/1361-6501/aa7d87

7 Lorenzo ED, Lava P, Balcaen R, Manzato S, Peeters B (2018) Full-field modal analysis using high-speed 3D digital image correlation. Journal of Physics: Conference Series 012007:1149. https://doi.org/10.1088/1742-6596/1149/1/012007

8 Witt B, Rohe D (2020) Digital image correlation as an experimental modal analysis capability. Experimental Techniques. https://doi.org/10.1007/s40799-020-00420-6

9 Rohe D, Quintana R, Witt B, Halls B (2021) Structural dynamic measurements using high-speed X-ray digital image correlation. In: Proceedings of the 39th International Modal Analysis Conference, Virtual.

10 Rohe D, Smith S, Brake MR, deClerk J, Blanco MA, Schoenherr TF, Skousen T (2019) Testing summary for the box assembly with removable component structure. In: Proceedings of the 37th International Modal Analysis Conference, Orlando, FL

11 Arun KS, Huang T, Blostein S (1987) Least-squares fitting of two 3-D point sets. IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-9(5):698-700

12 Allen E; Triantaphillidou S (2011). The manual of photography. Taylor & Francis. ISBN 978-0-240-52037-7

13 Greivenkamp J (2004). Field Guide to Geometrical Optics. SPIE Press, Bellingham, WA. ISBN 9780819452948

14 Rohe D (2020) An optical test simulator based on the open-source Blender software. In: Proceedings of the 38th International Modal Analysis Conference, Houston, TX

15 Sandia National Laboratories (2020) CUBIT 15.7 User Documentation. Sandia Technical Report, SAND2020-4156 W

16 Sandia National Laboratories (2020) Sierra/SD – Theory Manual. Sandia Technical Report, SAND2020-3687

17 Coxeter HSM, Greitzer SL (1967) Geometry revisited. pp. 52-54. MAA, Washington, ISBN 978-0-88385-619-2

18 Wirsching PH, Paez TL, Ortiz K (2013) Random vibrations: theory and practice. Dover Publications Inc., Mineola, NY, ISBN 9780486450155

19 Lin YL (1967) Probabilistic theory of structural dynamics. McGraw-Hill Book Co., New York, NY, ISBN 978-0882753775