Technical Report: alg04-02
February 02,2004
Sang-Chul Lee, slee40@uiuc.edu
Peter Bajcsy, pbajcsy@ncsa.uiuc.edu

Automated Learning Group
National Center for Supercomputing Applications
605 East Springfield Avenue, Champaign, IL 61820

Multi-Instrument Analysis From Point and Raster Data

Abstract

In this report, we address the problem of multi-instrument analysis from point and raster data. A camera-based sensor acquires raster data (images). Point-based sensors are attached to the object of interest and report accurate measurements at a sparse set of locations. Our work tackles the problem of raster and point data integration, and includes sensor registration, point data interpolation, variable transformation, data overlay and value comparison. We describe a few necessary steps one has to perform in order to form two comparable data sets in terms of (1) their coordinate systems, (2) spatial resolution and (3) physical entities. The objective of the processing steps is to resolve the above issues by (a) spatial registration of both data sets, (b) B-spline interpolation of point data, and (c) variable transformations of point data according to structural engineering formulas. We present our preliminary results from the on-going research that is a part of the National Earthquake Engineering Simulation (NEES) project and conducted in collaboration with the Civil and Environmental Engineering Department, UIUC. Our future work will focus on developing a new uncertainty model for data fusion so that we can predict and optimize setup parameters in the future multi-instrument experiments, for example, point sensor spacing, camera distance, point sensor locations or spatial overlap of sensor measurements.

1. Introduction

Recently, the growing number of applications for many different areas created the need for new sensors that can carry out the specialized tasks with high performance. Multiple sensors of the same kind or various types can be used for more accurate and confident data collection in one experiment. Although these multi-instrument experiments lead to measured data with increased quality, they pose new challenges on multi-sensor data fusion. For example, a conventional camera and an infrared camera can outperform a single camera or multiple conventional cameras in the environment with unexpected change of lighting or atmospheric conditions. However, in order to benefit from the multi-camera systems, one has to resolve several data fusion issues and these issues are addressed in the rest of this document.

Multi-sensor fusion is the process of dealing with the association, correction, correlation and combination of the sensor data from multiple sources with different modalities (wavelengths, data types, reference coordinate systems, imaging mechanisms, etc). A typical sequence of multi-sensor fusion steps is as follows:

1. Acquisition: Observe the *same* scene while measuring data with two or more multi-modal sensors.
2. Transformation: Transform all multi-sensor observations into a geometrically conformal form, such as a raster image.
3. Registration: Match common features in all transformed data to establish a common coordinate system.
4. Uncertainty analysis: Estimate the uncertainty (inaccuracy and confidence) of each sensor measurement after transformation and registration steps.
5. Integration: Fuse multiple data sets into a new data set by minimizing the uncertainty.

In addition to more confident and accurate fused data, the multi-sensor data fusion can provide additional benefits, such as, an extended temporal and/or spatial coverage,

reduced ambiguity, enhanced spatial resolution, and increased dimensionality of the measurement space.

While an increasing number of sensors have been developed recently, one of the most popular sensors is still a camera-based sensor because of its low cost and multi-purpose applicability in practice. The camera-based sensors perform non-contact measurements of target objects. In general, a camera-based sensor generates raster data that can be described as measurements of a 3D scene projected to the camera plane and represented by a regular two-dimensional array of values. For example, a conventional camera with arrays of charge-coupled devices (CCD) measures three arrays of light values of particular red, green and blue wavelengths that are visible to human eyes. Other cameras may use different wavelength bands, e.g., a thermal infrared (IR) camera, or different wavelength (spectral) resolution, e.g., a hyper-spectral camera.

Regardless of a wavelength, the accuracy of camera-based sensors is limited due to (a) camera noise, (b) measurement distortion caused by varying scene and light medium characteristics, e.g., fast moving objects, airflow, temperature and lighting condition, or (c) a limited field of view. For example, a spatial resolution may significantly degrade the accuracy of measurements of a large scene due to a limited field of view. This is the case of a camera moving away from the target object in order to capture an entire large scene. One approach to this problem is the use of multiple cameras located close enough to the target in different spatial locations, and fusing the acquired images to generate a big image with large spatial coverage and high spatial resolution. Unfortunately, this approach introduces new problems of camera calibration and image alignment.

Another approach to enhance the data quality from camera-based sensors is by using auxiliary deployable sensors, such as Micro-Electro-Mechanical Systems (MEMS), measuring scalar values at multiple locations on target objects. The MEMS sensors are in contact with the target objects, and hence their measurement error is more predictable and accurate. The measured entity can be a coordinate with respect to the fixed coordinate system, e.g., Global Positioning System (GPS), or an orientation, e.g., motion trackers, or some other physical entities, e.g., temperature, vibration or stress. The major challenge of

this approach lies in the limited number of measurements due to the physical size of sensors placed on a target object.

This report addresses the problem of multi-instrument (or multi-sensor) data analysis from raster (camera-based sensors) and point data. The multi-instrument data are generated during a material structural health analysis experiments, which is our application driver. Structural and earthquake engineers are interested in studying strain and stress values while loading a structure in multiple directions and with a variable load. The instruments for these experiments generate both raster-based and point-based information and the information corresponds to different physical quantities that can be related theoretically. For example, the camera-based Stress Photonic System generates a raster image where each pixel represents some shear strain values. This system can quickly produce full-field images of structural stresses and strains on the structure surface by measuring the ellipsity of the polarized light that is reflected by special photo-elastic coating on the test specimen. Another sensor, the point-based Krypton system, can measure three-dimensional coordinates of LED-based targets on a test specimen. The measured coordinates can be transformed to displacement values of the targets and the displacement values can be closely related to the strain and stress of the material of the target objects. The major limitations of the Stress photonic system include: (1) limited field of view and limited resolution of the image, and (2) some missing parameters for strain analysis, such as normal strains. We can overcome these limitations by using the Krypton system. However, we have to understand and analyze multi-instrument sensor data in order to improve the data spatial resolution and the accuracy of any derived or measured physical entity, for instance, by analyzing interpolation methods that have to be used. The aforementioned sensors and the associated data fusion problems are an integrated part of the Network for Earthquake Engineering Simulation (NEES) project conducted at the MUST-SIM facility at UIUC.

The outcome of our proposed analysis leads to the following advancements:

- **Increased confidence and reduced ambiguity**: The point-based sensors usually provide relatively more accurate and predictable measurements, and the raster data (images) can be calibrated by estimating the extrinsic parameters using the point measurements as reference points.

- **Extended spatial coverage**: Interpolating values based on a set of sparse measurements (point data) to raster data enhances possibly low-resolution raster images.
- **Efficient registration**: Point-based sensors attached on a test object can play a role of registration control points in a raster image during a template-based search for registration landmarks. There is no need to extra artificial landmarks for registration purposes.
- **Increased spatial resolution**: Lower accuracy of raster data can be enhanced by more accurate point data.

In this report, we demonstrate the above advancements by processing the data from Krypton System (point-based sensor) and Stress Photonic System (raster-based sensor). The project is an integrated part of the NEES MUST-SIM project and it is conducted as a collaboration of the National Center for Supercomputing Applications (NCSA) and the Department of Civil and Environmental Engineering at University of Illinois. The software documentation about the developed tools is a part of the Image to Knowledge (I2K) documentation [5]

2. Problem Formulation

Given raster data and point data, three steps can describe the general process for sensor fusion: data registration (calibration), interpolation of point data and data fusion. An automatic template-based data registration is based on defining (selecting an image example) a point sensor template (a point-sensor appearance in raster data), searching for the occurrences of the template in a raster image and transforming the data to one coordinate system by using the found locations of template matching as control points. If the automatic approach fails due to a large variation of point sensors then it is possible to define control points by mouse clicks. After the registration step, two data sets are converted into a conformal data form (image), and can be analyzed. Figure 1 show an overview of how two data sets are converted and fused. First, the raster data λ_1 is calibrated using point data λ_2 as reference points, and create a calibrated image λ_1'. To

fuse λ_2 with λ_1', we create a raster image λ_2' from λ_2 by interpolation, such as B-spline data fitting. While interpolating, boundary conditions must be known a priori due to the lack of data points close to the data boundaries. The interpolated images are in the same conformal form as the raster images. There are multiple strategy for data fusion of λ_1' and λ_2', and we will describe our strategy later in the text.

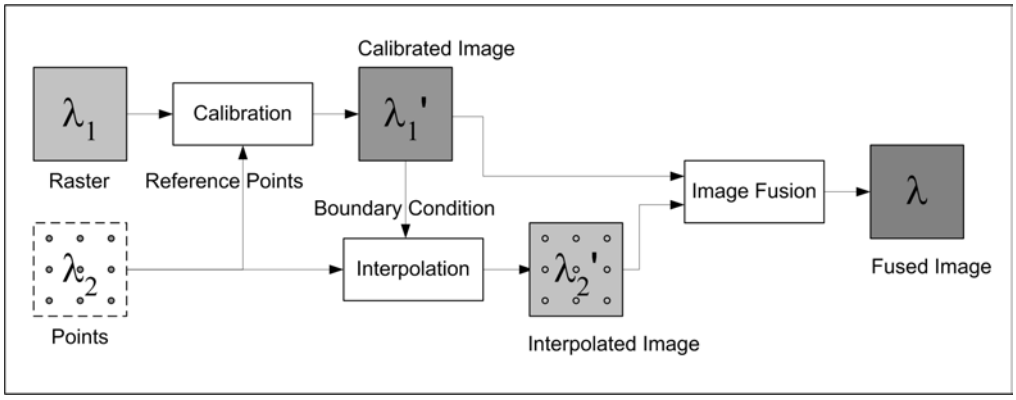


Figure 1: Data fusion of a raster image and a set of point data

In order to solve the data fusion problem for raster-based and point-based sensors, we decomposed the process into the following steps:

1. **Sensor setup**: For a successful experiment, prepare the sensors so that they acquire data λ_1 and λ_2 from the same spatial location. We assume that the point-based sensors are reasonably small and do not occlude any significant area of the scene. It is recommended to orient the camera perpendicular to the test object surface in order to eliminate any distortion due to a perspective view. Depending on the application, the point sensors can be arranged in a grid or in clusters. It is assumed that the point sensors are spatially sparse and locations of all point sensors are known a priori or can be directly measured during the experiment.
2. **Synchronization**: Synchronization is an important issue for data collection from multiple sensors to guarantee temporal coincidence of measurements. Generally,

synchronization can be achieved in two different ways. First, each sensor takes its measurement after being triggered by one centralized controller. This way, any collected data can be processed in real time since the data is available immediately. Second, each sensor collects data independently and stamps it with a global time stamp. One can process such data collection based on the global time stamp.

3. **Data enhancement**: In case that the measured data value of the test target changes very slowly comparing with the sensor speed of the data measurement, multiple temporal steps of the measurements can be averaged to generate a more reliable and less noisy data set. The raster data can then be enhanced (calibrated) again by taking the point data as reference points.
4. **Registration**: As described in Step 1, the point sensor location is previously known or measurable during the experiment, e.g., manually or using tracking devices. Matching salient sensor points in raster data with known point sensor locations is used for performing a multi-sensor registration. In our sensor environment, the point based sensors play the role of the feature points, e.g., landmarks. The registration model in our application is an affine transform that compensates for translation, rotation and shear.
5. **Image generation**: The point data is too sparse to directly compare their values with the corresponding raster data. We estimate a denser image λ_2' by interpolating point data λ_2. Depending on the application needs, one can use bi-linear, bi-cubic, B-spline or other interpolation methods. In the case of B-spline based interpolation, boundary conditions can be provided by the registered and calibrated raster data λ_1'. The output is a set of raster images or an image with multiple bands where each image (or band) has different physical entity. Note that the raster image λ_1 and the generated image λ_2' are registered because the point data set is already registered with the raster data in the previous step.
6. **Value derivation**: We derive a new physical entity λ for value comparisons by transforming λ_1' and λ_2' according to appropriate physical laws.

7. **Uncertainty analysis**: We analyze the uncertainty of λ_1', λ_2' and λ that is caused by error propagation during the geometric image transformation (affine transformation), interpolation and value transformation [6].
8. **Image fusion**: We may have multiple data fusion scenarios depending on the test setup, and experimental hypothesis.
    - Value selection: Based on uncertainty analysis, create a new image by taking the more confident value from the generated data sets λ_1' and λ_2' for all spatial locations.
    - Value merging: In case that each sensor provides different physical entities as well as a common physical entity, they can be merged into the new image λ that contains both physical entities from λ_1' and λ_2'.
    - Spatial extension: In case that the point sensor measures larger area, interpolated data can be adjusted by raster data based on the spatial area where both measurements are available.

3. Test Setup and Data Acquisition

Our goal is to acquire data set from a test specimen with multiple instruments in order to derive very accurate and highly confident data. In this section, we present a test setup that is designed for acquiring data suitable for our data analysis. The list of setup requirements is summarized as follows.

- Set both image-based and point-based sensors to acquire data from the same spatial location. Depending on applications, one sensor may cover larger area than the other. It is recommended that there is sufficient spatial overlap of the data sets.
- In order to avoid occlusion, the size of point sensors is small with respect to the size of areas of interest. The problem with occlusion should be also avoided by spatially sparse distribution of point sensors.
- Viewing geometries of all camera-based sensors are the same. It is recommended to setup the camera-based sensors perpendicular to the surface of the test target.

- Point sensors can be arranged in a grid or in a cluster formation. Point sensors are firmly attached and their locations are known a priori.

3.1 A grid-based target arrangement

It is useful to arrange point-based sensors in a grid formation for mathematical modeling purposes and for uniform spatial coverage. In this configuration, the point sensors are arranged in a grid pattern aligned with the image row/column coordinate system. This configuration provides the following benefits:

- Efficient registration by using point sensors in a raster image as salient features for computing registration parameters. Point sensor detection in a raster image can be faster and more accurate since the known geometric layout of the point sensors improves robustness of automatic detection.
- Fast interpolation. Given equidistant point spacing, B-spline interpolation model can use uniform knot vectors and hence the interpolation is computationally less expensive.
- Minimum uncertainty variation across a given spatial coverage. Any configuration other than a uniform grid will lead to a larger uncertainty variation.
- Efficient transformation of point data to raster image. The point data can be directly transformed to a raster image with dimensions equal to the number of sensors times spacing distance along each direction. Later, low-resolution images can be interpolated to generate higher resolution images.

Assuming the grid configuration, we can automatically derive the sensor geometry by partitioning the coordinates set independently along x (column) and y (row) axis. Let $P_k^{(x)}$ be a set of points p_i such as:

$$\left\| x(p_i) - x(p_j) \le \varepsilon \right\|$$
$$\text{where } p_i, p_j \in P_k^{(x)}$$

$x(p)$ represents the x-coordinate of the point p, and ε can be manually decided for the maximum allowed gap for a cluster. Then, we can define a sorted set $\mathbf{P}^{(x)}$ and $\mathbf{P}^{(y)}$ for each axis as:

$$\mathbf{P}^{(x)} = \{P_0^{(x)}, P_1^{(x)}, P_2^{(x)}, ..., P_c^{(x)}\}$$
$$\text{where } x(p_i) < x(p_j) < ... < x(p_k),$$
$$p_i \in P_0^{(x)}, p_j \in P_1^{(x)}, \text{and } p_k \in P_c^{(x)}$$

$$\mathbf{P}^{(y)} = \{P_0^{(y)}, P_1^{(y)}, P_2^{(y)}, ..., P_r^{(y)}\}$$
$$\text{where y}(p_l) < y(p_m) < ... < y(p_n),$$
$$p_l \in P_0^{(y)}, p_m \in P_1^{(y)}, \text{and } p_n \in P_r^{(y)}$$

The point P_{ij} in a two dimensional location, where i is the row index and j is the column index, can be found by satisfying:

$$p_{ij} = p_k, \text{ where } p_k \in P_j^{(x)}, p_k \in P_i^{(y)}$$

After finding the set of points P_{ij}, we can map the unique ID of each sensor in the scene with the raster image coordinate system. We refer to this operation as *sensor localization*.

3.2 Test setup for the NEES experiment

In the NEES MUST-SIM experiments, we used both Krypton (point-based sensor) and Stress photonic system (raster-based sensor) for analyzing a test specimen. Figure 2 shows an example of the test setup. The structure is first coated by an epoxy to reflect polarized light capture by the Stress photonic camera. Next, Krypton LED targets are attached in a grid pattern on top of the epoxy coating. Both the Krypton camera and the Stress photonic camera are viewing the same region of a test specimen. Although the LED targets occlude some area of the epoxy coating, the missing region of the Stress photonic data can be recovered by using interpolation methods, for instance, the nearest

neighborhood interpolation method. We view the Krypton system as a point-based sensor regardless of the camera sensing LED locations.

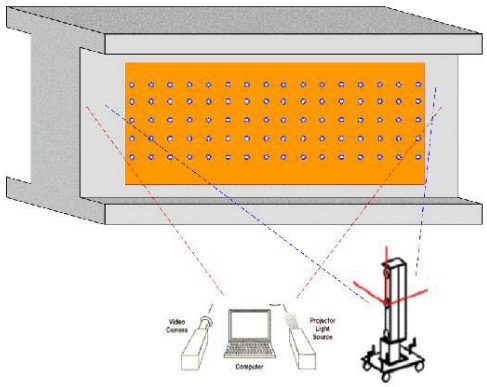


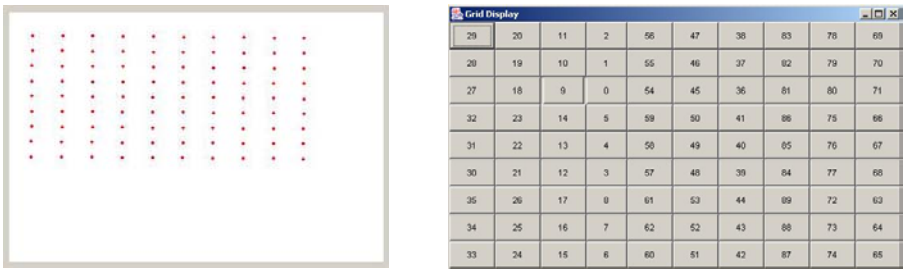
Figure 2: Test setup for NEES experiment

The Krypton system is capable of measuring 3D coordinates of LED targets in real time. By placing the Krypton LED targets systematically on a grid, we can acquire a useful data set for finite element analysis. Krypton provides 3D coordinates for up to 256 LEDs on a test specimen. The raw output of Krypton is by default in a MATLAB data file format. The file consists of a set of three-dimensional coordinates of LED targets associated with time stamps. Figure 3 shows an example of raw data. The first column contains a time stamp and the following columns contain x, y and z coordinates of individual LEDs ordered based on their IDs. The coordinate values are reported in millimeters. The coordinates are normalized to the local origin of the Krypton system, which is predefined by the provided probing device at initialization procedure.

	LED1			LED2			
time	x_1	y_1	z_1	x_2	y_2	z_2	
0.000000	183.019803	-9.303310	42.436430	189.003514	-10.318793	163.885000	
0.200000	183.030240	-9.304754	42.437568	189.011545	-10.334166	163.889341	
0.400000	183.017366	-9.301389	42.435610	189.007048	-10.325589	163.886182	...
0.600000	183.033665	-9.270475	42.433992	189.013636	-10.315177	163.887482	
0.800000	183.024098	-9.278258	42.432052	189.006470	-10.305221	163.884431	
1.000000	183.032185	-9.278146	42.439778	189.013927	-10.316019	163.889854	
				...			

Figure 3: Krypton RODYM data format

Figure 4(a) shows a visualization of the test data. The red dots denote LED targets and form a grid pattern. Figure 4(b) shows the result of the dynamic detection of LED identification numbers.



(a) (b)

Figure 4: Dynamic grid detection: (a) Point visualization and (b) dynamic grid detection

4. Affine Parameter Estimation for Registration

The first step of registration is to find the best corresponding features in two coordinate systems. In this report, we automatically find the LED sensor locations in a row-column based coordinate system of the raster image. In a grid point sensor layout, we can find the correspondences by using the method suggested in Section 3.1. We assumed that the camera-based sensor is perpendicular to the test target, and therefore a simple affine

transformation model is sufficient for registering two coordinate systems. The affine transformation is defined as:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

Given a set of corresponding points (x, y) and (x', y'), called *feature points or registration control points*, we estimate the affine transformation parameters a, b, c, d, t_x and t_y. Each pair of corresponding points provides two constraints for x and y, and therefore we need at least three corresponding points to estimate all six affine transformation parameters. The transformation matrix can be rewritten as a set of linear equations.

$$ax + by - x' + t_x = 0$$
$$cx + dy - y' + t_y = 0$$

We rewrite the two equations in a matrix form:

$$\begin{bmatrix} x & y & x' & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ -1 \\ t_x \end{bmatrix} = 0; \begin{bmatrix} x & y & y' & 1 \end{bmatrix} \begin{bmatrix} c \\ d \\ -1 \\ t_y \end{bmatrix} = 0$$

To be able to cope with more than three pairs of feature points, we used the least squared method described in [3]. For each *x*-parameter set and *y*-parameter set, the least squared equation of the affine transform can be written as:

$$\begin{bmatrix} x \\ y \\ x' \\ 1 \end{bmatrix} \begin{bmatrix} x & y & x' & 1 \end{bmatrix} q_x = 0 \;\; ; \;\; \begin{bmatrix} x \\ y \\ y' \\ 1 \end{bmatrix} \begin{bmatrix} x & y & y' & 1 \end{bmatrix} q_y = 0$$

where $q_x = \begin{bmatrix} a & b & -1 & t_x \end{bmatrix}^T$ and $q_y = \begin{bmatrix} c & d & -1 & t_y \end{bmatrix}^T$. To estimate q_x, we rewrite the matrix as:

$$\begin{bmatrix} x^2 & xy & xx' & x \\ xy & y^2 & x'y & y \\ xx' & x'y & x'x' & x' \\ x & y & x' & 1 \end{bmatrix} q_x = 0$$

Sum of this squared matrix is written as:

$$\begin{bmatrix} \sum x^2 & \sum xy & \sum xx' & \sum x \\ \sum xy & \sum y^2 & \sum x'y & \sum y \\ \sum xx' & \sum x'y & \sum x'x' & \sum x' \\ \sum x & \sum y & \sum x' & n \end{bmatrix} q_x = 0$$

where $n >= 3$, which makes the matrix non-singular. To estimate q_x and minimize the sum of the least squared error, we take the eigenvector of the smallest eigenvalue computed from the covariance matrix. The estimation for q_y is performed the same way. The final estimated affine parameters are used for interpolating point-sensor data to match the resolution of raster data.

5. Data Transformation

Point sensors acquire a collection of values at multiple spatial locations. By configuring the point sensors in a grid pattern, we can generate easily dense raster data based on continuous interpolation models. The reason for selecting continuous interpolation models comes from our understanding of physical phenomena. For example, many materials in structural engineering are assumed to follow an elastic model, or in other words, a model that assumes a smooth spatial variation of certain physical properties due to material loading. Based on this type application understanding, we used a B-spline interpolation method since it satisfies C-2 continuity as required by our application. In the rest of this section, we assume the grid arrangement of point sensors as described in 3.1.

5.1 Image generation by B-spline based interpolation

Due to the spatial size of point sensors, the point data are usually at coarser spatial resolution than the raster data. To acquire point data at higher resolution, one can either increase the number of sensors or can estimate the values between sensors by creating

imaginary sensor readings, called *interpolation*. The first approach is straightforward and provides more accurate data values. However, it is not only the spatial dimension of point sensors that limits the density of point sensors but also the material occlusion of point sensors in raster data that voids the usefulness of raster data for data fusion purposes. With the interpolation approach, one can overcome the above problems although the trade-off between spatial density of measurements and accuracy of measurements remains to be addressed. Different interpolation methods can be used in this approach. A bi-linear interpolation method can be used to linearly estimate one or multiple values inside of each cell by taking four boundary points. The main problem of bi-linear interpolation is the discontinuity of the values at the edges of adjacent cells. Our assumption is that there are no cracks in a test structure and the test object most likely follows a smooth distribution of certain physical entities. The rest of this section describes the B-spline based interpolation method.

We assume that we are given a set of three-dimensional points (x, y, v_k), where (x, y) are from a two-dimensional xy-plane, and $v_k \in V$ is the measurement at (x, y). The B-spline interpolation procedure follows a standard surface-fitting algorithm as described in [2]. For evenly spaced point sensors, we used uniform B-splines rather than the standard B-spline with individually assigned knot vectors. The uniform B-spline leads to faster and simpler computation. The description of our procedure for the three-dimensional surface fitting is presented next.

1. Construct a curve network: Fit a set of curves (U-curves) for the points along x-axis (rows), and a set of curves (V-curves) along y-axis (columns).
2. Convert the curve network of B-splines from Step 1 into a new curve network represented by Bezier curves.
3. For each cell in the curve network, construct Bezier patch by estimating four missing control points in each cell by doing bi-linear interpolation.
4. Join all created Bezier patches to form the whole interpolation surface.

To construct the curve network according to Step 1, we use a uniform B-spline as defined below:

$$\begin{aligned} Q_i(u) &= \sum_{i=0}^{m} P_i B_i(u) \\ & \quad B_i(u): \text{ Basis function} \end{aligned}$$

$Q_i(u)$ is a parametric representation of the i th curve segment with respect to u, where $0 \le u \le 1$, and P is a set of control points. We can represent this formula in a matrix form as:

$$\begin{aligned} Q_i(u) &= UM_{BS}P \\ &= \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \frac{1}{6} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \begin{bmatrix} p_i \\ p_{i+1} \\ p_{i+2} \\ p_{i+3} \end{bmatrix} \end{aligned}$$

The curve fitting in Step 2 is the process of finding the B-spline $Q(u)$ that passes through the data points. Given a set of data points on the curve $Q(u)$, the curve fitting requires computing the set of control points P for each spline segment. Thus, we rewrite the B-spline equation as:

$$\begin{aligned} \mathbf{Q}_i(u_p) &= \sum_{i=0}^{m} \mathbf{P}_i B_i(u_p) \\ &= D_p \text{ for all } p = 3, ..., m+1 \end{aligned}$$

D_p represents the set of data points, and u_p is the knot value which corresponds to the data point. From this formula, we have a system of equations in a matrix form:

$$\begin{bmatrix} B_0(u_3) & \dots & B_m(u_3) \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ B_0(u_{m+1}) & \dots & B_m(u_{m+1}) \end{bmatrix} \begin{bmatrix} \mathbf{P}_{x0} \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{P}_{xm} \end{bmatrix} = \begin{bmatrix} D_{x3} \\ \cdot \\ \cdot \\ \cdot \\ D_{xm+1} \end{bmatrix}$$

By solving this system of linear equations, we compute a set of curves with respect to each column and row in the curve network.

In Step 3, the goal is to convert the set of control points of B-splines into a set of control points of equivalent Bezier curves. Given Bezier control points, we can define the Bezier patches as:

$$Q_i(s,t) = \sum_{i=0}^{3}\sum_{j=0}^{3} p_{ij} B_i(s) B_j(t)$$

The union of all Bezier patches is achieved in Step 4. The joined Bezier curves satisfy the C-2 continuity property since they are identical to the B-spline. Thus, the joined patches are continuous with the C-2 continuity, as well.

5.1.1 Data interpolation for strain analysis

The Krypton system enables users to utilize relatively large number of LED targets in comparison with classic sensors, such as displacement transducers. Nevertheless, the density of measured data is still spatially coarse in comparison with the raster data generated by Stress photonic system. This fact leads to the need for spatial interpolation of Krypton data in the NEES experiments and the steps are described next.

1. At time t, compute displacement values δx and δy with respect to the initial loading step for every LED target.
2. Form two sets of 3D points by mapping δx and δy to the z coordinate.
3. For each coordinate set $(x, y, \delta x)$ and $(x, y, \delta y)$, apply surface fitting algorithm to recover the two three-dimensional surfaces, δx-surface and δy-surface.
4. Sample points from each constructed surface to generate raster images.
5. Compute ε_x, ε_y, γ_{xy} and $\gamma_{\max}$ according to structural engineering formulas from the two interpolated raster data sets to create new raster images representing these new physical entities. The detailed formulas are provided in the next section.

In this section, we showed how to use the general data interpolation method to construct a strain analysis model by interpolating displacement values δx and δy. Figure 5 shows the exaggerated deformation of the LED targets on the test structure at $t = 6$. The left-end of the target structure was fixed, and the loading was applied toward the right-end bottom of the test structure. From the red points shown in Figure 5, we constructed raster images

of δx and δy, illustrated in Figure 6. We can observe that the pseudo-colored images show smooth contour curves with respect to the displacement values.

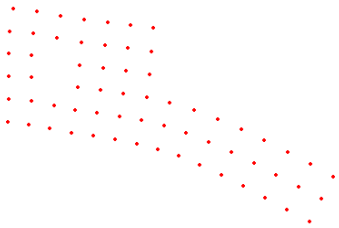
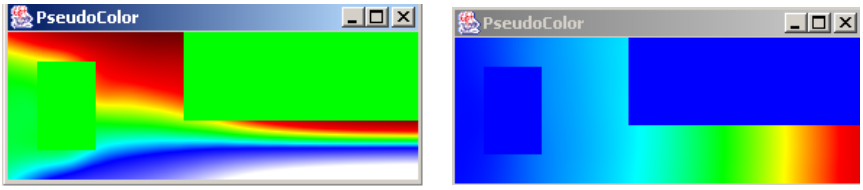


Figure 5: Exaggerated deformation of the LED targets



(a) (b)

Figure 6: Interpolated displacement image: (a) δx (color range: -0.002mm ~ +0.002mm) and (b) δy (color range: 0.00mm~0.05mm)

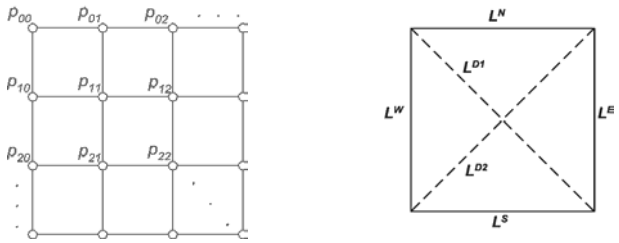
5.2 Value transformation

5.2.1 Strain analysis using a finite element model

5.2.1.1 Point data transformation (Krypton system)

In order to perform a finite element analysis of a test structure, we computed displacement changes at time t with respect to the initial loading for each grid cell defined by the four closest point sensors. To compute displacements at the grid cell

edges, we calculated the Euclidian distances of LED target coordinate measurements between adjacent points on the grid. Figure 7, (a) shows the LED target layout for the analysis. A point $p_{ij} = (x_{ij}, y_{ij}, z_{ij})$ represents a three-dimensional coordinate of one LED target at i, j in xy-plane.



(a) (b)

Figure 7 (a) A layout of Krypton LED targets and (b) one cell considered for the finite element analysis

For each cell with four points $p_{ij}, p_{i+1j}, p_{i+1j+1}$ and p_{ij+1}, we calculate six Euclidian distances for finite element analysis, as it is illustrated in Figure 7(b).

$$
\begin{aligned}
L_{ij}^{\eta} &= \{L_{ij}^{N}, L_{ij}^{S}, L_{ij}^{E}, L_{ij}^{W}, L_{ij}^{D1}, L_{ij}^{D2}\} \text{where } \eta = \{N, S, E, W, D1, D2\} \\
& L_{ij}^{N} = ||p_{ij} - p_{ij+1}|| \\
& L_{ij}^{S} = ||p_{i+1j} - p_{i+1j+1}|| \\
& L_{ij}^{E} = ||p_{ij+1} - p_{i+1j+1}|| \\
& L_{ij}^{W} = ||p_{ij} - p_{i+1j}|| \\
& L_{ij}^{D1} = ||p_{ij} - p_{i+1j+1}|| \\
& L_{ij}^{D2} = ||p_{i+1j} - p_{ij+1}||
\end{aligned}
$$

Based on all Euclidian distances at time t, the strain along each edge is calculated as:

$$
\begin{aligned}
\Delta L^{\eta}(t) &= L^{\eta}(t) - L^{\eta}(0), \\
\epsilon^{\eta}(t) &= \frac{\Delta L^{\eta}(t)}{L^{\eta}(0)}
\end{aligned}
$$

Following formulas show some elemental strains based on the edge strains at time t.

$$
\varepsilon_x(t) = \frac{\epsilon^{N}(t) + \epsilon^{S}(t)}{2}
$$

$$
\varepsilon_y(t) = \frac{\epsilon^{E}(t) + \epsilon^{W}(t)}{2}
$$

$$\gamma_{xy}(t) = \frac{\epsilon^{D1}(t) - \epsilon^{D2}(t)}{2}$$

$$\gamma_{max}(t) = \sqrt{(\varepsilon_x(t) - \varepsilon_y(t))^2 + (\gamma_{xy}(t))^2}$$

$\varepsilon_x(t)$ and $\varepsilon_y(t)$ are the average normal strains along x-axis and y-axis respectively. A positive value of the normal strain means a *tensile strain* and a negative value means *compressive strain*. The shear strain γ_{xy} can be calculated by measuring diagonal strains, $D1, D2$, which is closely related to the shape of the element. All four values generated above are essential for constructing the Mohr's circle that is frequently used in strain analysis. For more details, see [1].

5.2.1.2 Raster data analysis (Stress Photonic system)

The Stress Photonic system measures two types of shear strains: γ_{45} and γ_0. The shear strain γ_{45} corresponds to the shear strains on the $\pm 45°$ inclined planes and γ_0 represents the shear strains on the $0/90°$ planes. These two entities can be transformed into the maximum shear strain $\gamma_{\max}$ defined below.

$$\gamma_{max} = \sqrt{\gamma_0^2 + \gamma_{45}^2}$$

The shear strain maximum is one of the comparable physical entities that one could derive from Krypton and Stress Photonics data. While the shear strains γ_{45} and γ_0 represent a point on the Mohr's circle, the maximum shear strain $\gamma_{\max}$ corresponds to the Mohr's circle radius. For more detail, see [4].

Although the Stress photonic system provides relatively accurate shear strain measurements, it is not possible to recover the average normal strains ε_x and ε_y from the raster data. In this case, the fusion of point data and raster data can not only improve an accuracy of shear strain, e.g., $\gamma_{\max}$ but also expand the list of accessible variables, e.g., ε_x and ε_y, for finite element analysis.

5.2.1.3 Experimental result

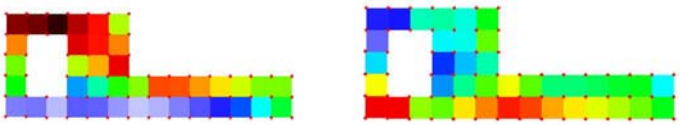
5.2.1.3.1 Krypton System

Figure 8 shows a test specimen in (a) and the LED IDs at their detected locations in (b). The developed LED detection method is capable of finding a partial grid, as well as, a complete grid. For example, we could detect point-sensor locations on a L-shaped structure by masking missing targets (see Figure 8).

(a) (b)

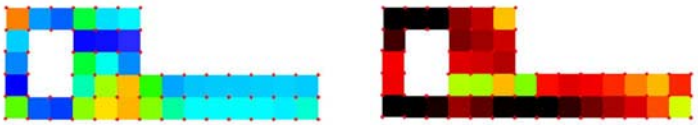
Figure 8: L-shaped structure: (a) LED layout and (b) grid detection

After computing average normal and shear strains along edges of each cell, element strains can be visualized as pseudo-colored images in Figure 9 and Figure 10. Each cell is pseudo-colored with respect to the horizontal or vertical strain value according to the Equations presented in the previous section. The presented test data were generated using ABAQUS software that simulated the Krypton output (vector data).



(a) (b)

Figure 9: Normal strain: (a) horizontal strain (ε_x) and (b) vertical strain (ε_y).

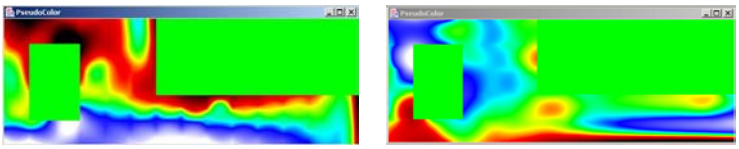


(a) (b)

Figure 10: Shear strain: (a) shear strain (γ_{xy}) and (b) maximum shear strain ($\gamma_{\max}$).

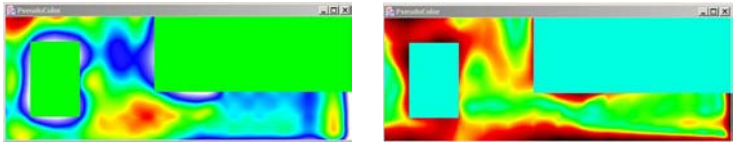
5.2.1.3.2 Interpolated result of the Krypton System

Given an interpolation model, we can create a raster data set of directional displacements, as shown in Figure 6, at any raster resolution (density). Thus, we can generate a much denser representation of the images shown in Figure 9 and Figure 10 by applying transformation formulas to the interpolated data set. Figure 11 and Figure 12 show raster images derived from the interpolated data at higher spatial resolution from the data shown in Figure 6.



(a) (b)

Figure 11: Interpolated normal strains: (a) horizontal strain (ε_x) and (b) vertical strain (ε_y)

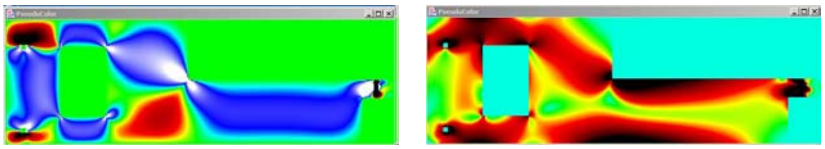


(a) (b)

Figure 12: Interpolated shear strains: (a) shear strain (γ_{xy}) and (b) maximum shear strain ($\gamma_{\max}$)

5.2.1.3.3 Stress Photonic System

Figure 13 shows the visualization of ABAQUS predictions for Stress Photonic system data. Figure 13 (a) shows the shear strain γ_{45}, which is equivalent to γ_{xy} in the displacement based strain calculation, and Figure 13 (b) shows the derived maximum shear strain $\gamma_{\max}$.



(a) (b)

Figure 13: Shear strain: (a) shear strain (γ_{45}) and (b) maximum shear strain ($\gamma_{\max}$).

6. Conclusion

In this report, we showed a framework for multi-instrument data analysis from point and raster data. The presented work involved sensor registration, point data interpolation, variable transformation, and value comparison. We successfully generated two comparable data sets in terms of (1) coordinate system locations, (2) spatial resolution and (3) physical entities by (a) B-spline interpolation of point data, (b) variable transformations of point data and (c) spatial registration of both data sets. Our future

work will focus on uncertainty analysis as a function of spatial locations of the interpolated raster image. The contribution to measurement uncertainty will be analyzed with respect to (a) instrument error, (b) interpolation error, and (c) error propagation from value transformations. We hope to develop a new uncertainty model for data fusion so that we can optimize instrument setup parameters, for example, point sensor spacing, camera distance, point sensor locations or spatial overlap of sensor measurements, in the future multi-instrument experiments involving raster-based and point-based sensors.

References

[1] J. Gere and S. Timoshenko, "Mechanics of Materials," *PWS publishing company*, Fourth edition

[2] A.Watt and M. Watt, "Advanced Animation and Rendering Techniques" *Addision-Wesley*, Workingham, England. 1992.

[3] E. Kang, I. Cohen and G. Medioni, "Robust Affine Motion Estimation in Joint Image Space using Tensor Voting," *International Conference on Pattern Recognition*, Quebec City, Canada, August 2002

[4] Stress Photonics Instrument, URL: http://www.stressphotonics.com

[5] P. Bajcsy et. al., "Image To Knowledge", software documentation at URL: http://alg.ncsa.uiuc.edu/tools/docs/i2k/manual/index.html; StrainAnalysis Tool.

[6] Taylor, John R. "An Introduction to Error Analysis: The Study of Uncertainties if Physical Measurements," *University Science Books*, 1982