UTILIZING THERMO-ELASTIC STRESS ANALYSIS TO AID DEVELOPMENT OF TEST-TO-MODEL CORRELATION CRITERIA

Thesis

Submitted to

The School of Engineering of the

UNIVERSITY OF DAYTON

In Partial Fulfillment of the Requirements for

The Degree of

Master of Science in Mechanical Engineering

By

Caitlin Marie Jenkins, B.S.

UNIVERSITY OF DAYTON

Dayton, Ohio

December 2021

UTILIZING THERMO-ELASTIC STRESS ANALYSIS TO AID DEVELOPMENT OF TEST-TO-MODEL CORRELATION CRITERIA

Name: Jenkins, Caitlin Marie

APPROVED BY:

Dennis J. Buchanan, Ph.D.
Advisory Committee Chairman
Division Head, Structural Materials
Mechanical & Aerospace Engineering

Robert L. Lowe, Ph.D.
Committee Member
Assistant Professor
Mechanical & Aerospace Engineering

David H. Myszka, Ph.D.
Committee Member
Professor
Mechanical & Aerospace Engineering

Robert J. Wilkens, Ph.D., P.E.
Associate Dean for Research and Innovation
Professor, School of Engineering

Margaret F. Pinnell, Ph.D.
Interim Dean, School of Engineering

ABSTRACT

UTILIZING THERMO-ELASTIC STRESS ANALYSIS TO AID DEVELOPMENT OF TEST-TO-MODEL CORRELATION CRITERIA

Name: Jenkins, Caitlin Marie
University of Dayton

Advisor: Dr. Dennis J. Buchanan

Non-contact instrumentation methods are becoming more prevalent in the realm of structural testing for collecting experimental data to correlate to finite element models. Thermo-elastic stress analysis is a type of non-contact method that has been gaining popularity in its use, however, criteria for how to correlate this data to finite element models has not been developed. As this method produces an averaged image over a span of cyclical loading, assessing the quality of the image is the first step in determining how to develop criteria for correlation. Included herein is an experiment that employs a thermo-elastic stress analysis system that utilizes a microbolometer to capture infrared images from the heat produced from a dogbone specimen. These images are then compared to a reference image, and image quality indices and an error index are produced for each set of images. These values are evaluated and a determination is made on how to utilize them for correlating the model strain values to the strain values measured by the thermo-elastic stress analysis system.

ACKNOWLEDGMENTS

My special thanks are in order to Dr. Dennis Buchanan, my advisor, for providing technical guidance, directing this thesis, and bringing it to its conclusion with patience and expertise. Additionally, to Dr. TJ Spradlin for providing the time and equipment necessary for the work contained herein, and providing technical guidance.

I would also like to express my appreciation to everyone who has helped me with this work. This includes Tony McFall, Dewayne Gray, Wes Ault, William Boles, and Brian Smyers who provided technical assistance with conducting the experimentation and reviewing the data produced. I also deeply appreciate the help given by Adam DeVilbiss, who aided in reviewing the text and provided much needed time, patience, and encouragement throughout the completion of my thesis.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND NOTATIONS

α	Coefficient of Thermal Expansion
ρ	Density
C_p	Specific Heat at Constant Pressure
T	Absolute Temperature
ΔT	Change in Temperature
$\Delta\sigma$	Change in the Sum of the Principal Stresses
Q	Heat energy
$C_\in$	Specific Heat at Constant Strain
σ_{xy}	Covariance of factors x and y
σ_x , σ_y	Standard deviation of factor x and y, respectively
$\bar{x}$, $\bar{y}$	Mean of factor x and y, respectively
MiTE	Microbolometer Thermo-elastic Evaluation
TSA	Thermo-elastic Stress Analysis
SPATE	Stress Pattern Analysis by Thermal Emission
NETD	Noise Equivalent Temperature Difference
DIV	Difference in Variance
CC	Correlation Coefficient
ERGAS	*Erreur Relative Globale Adimensionnelle de Sysnthesis*
RASE	Relative Average Spectral Error
RMSE	Root Mean Square Error

CHAPTER I

INTRODUCTION

1.1 Research Areas

Two areas of study were addressed in the thesis; the first being a non-contact instrumentation method named thermo-elastic stress analysis; the other being a study of utilizing this non-contact instrumentation method to aid in developing test-to-model correlation criteria, specifically utilizing image quality indices to determine how well the data from this instrumentation method correlates to the nodal model data. This required the development of an experiment to capture the stress-state of a Ti-6Al-4V dogbone specimen utilizing both contact and non-contact instrumentation methods, as well as employing a Matlab image analysis program to compare the thermal images in each test sequence to a reference image, and comparing the strain data to the nodal strain data in the model. This would ultimately be utilized to further develop model correlation criteria for use when employing a microbolometer thermal camera to capture thermo-elastic stress data. Microbolometers are uncooled, smaller thermal detectors with a small Noise Equivalent Temperature Difference (NETD), or thermal sensitivity, which is a measure of the limit to which a useful signal can be resolved by the thermal detector, or the noise floor of the detector. This thermal sensitivity can affect the quality of the image, and as this value is typically higher in microbolometers when compared to larger, cooled photon detectors, the quality of the image is a factor that should be carefully considered when determining correlation criteria to a finite element model.

1.2 Thermo-elastic Stress Analysis

Thermo-elastic Stress Analysis (TSA) is a non-contact instrumentation technique that utilizes an infrared camera to capture the thermal energy released when a test specimen is subjected to cyclic mechanical loading, also referred to as the thermo-elastic effect. Due to the thermo-elastic effects of various materials (e.g., metals) when subjected to elastic loading, this thermal energy can be directly correlated to elastic strain energy upon calibration to a reference signal, typically a traditional foil strain gage placed within the field of view, specifically an area of constant or bulk (homogeneous) strain. Microbolometer Thermo-elastic Evaluation (MiTE) is a TSA capability that was developed by the Defense Science and Technology (DST) Group in Australia specifically for use with microbolometers. This type of uncooled thermal sensor enables the user to utilize a smaller thermal camera, in this case the FLIR A35sc, to capture the thermoelastic effect in areas where a larger, cooled camera would not be feasible, while providing comparable measurement sensitivity. Utilizing the MiTE software and multiple strain gages installed far-field from the stress concentration, the thermal energy can be converted into a full-field strain map. This full-field strain map can then be utilized to assess stress concentration values and various other geometric responses to loading.

1.3 Image Quality Analysis

Image quality analysis is a statistical method of determining how accurately a digital image is able to capture the details of the subject of the image. This requires a reference image upon which to compare the test images and determine the deviation values from the reference image. This can be accomplished for a multitude of image types, from black and white (or greyscale), to color images with varying levels of spectral color bands. Within this work, three image quality indices will be calculated over three color bands. These color bands are red, blue, and green, and correspond to the color bands utilized by the MiTE software for displaying the range of strain values on the calibrated strain map. For increasing load levels, such as what is present in this study, the change from the reference image should follow a slightly decreasing positive correlation for the red and green band, as the load will increase and the strain data should increase along with the load value causing a slight deviation as this value changes. The index values for the blue band, however, should remain fairly consistent, as the strain values for this band should not change much as strain values this low will likely not be present. Image indices can be useful in determining correlation criteria to a finite element model as the quality of the image is considered, as well as error, correlation, and variance. The image indices of most interest in this study are those of root mean square error (RMSE) and quality (which includes the correlation coefficient, as well as two distortion factors).

1.4 Model Correlation Criteria

Model correlation criteria are standards that assess the degree to which the model matches the real-world component. This can be accomplished in various ways, and the standards by which this metric is determined are subject to the application of the model. For this experimental work, the model in question is one in which the material response of a component is being modeled to enable the prediction of this response to various types of loading. To determine the validity of the model, experimental data is necessary to compare the response of the physical component to that of the one modeled based on established boundary conditions, material properties, and any environmental concerns relevant to the test environment, such as temperature, humidity, and general controllable factors that can be added as inputs to the model. For the purposes of this endeavor, the guidance for correlation of the finite element model to the strain data captured by the various instrumentation methods will be the *Air Force Structural Integrity Program Structures Bulletin, EN-SB-11-001* [1]. This document provides guidance on correlating finite element models to structural ground test data, which is indicative of the testing described in Chapter 3 of this document.

CHAPTER II

LITERATURE REVIEW / BACKGROUND

2.1 Thermo-elasticity

The thermo-elastic effect, or the change in temperature a material undergoes when subjected to elastic deformation, be that due to cyclic loading or pressure changes, is the underlying principle utilized by thermo-elastic stress analysis, or TSA. From this effect, a linear relationship was derived to express the connection between the change in temperature and the change in stress of the system, or component, as shown in Equation (1) [2]:

$$\Delta T = \frac{T}{\rho C_\epsilon}\left(\frac{\partial \sigma_{ij}}{\partial T}\epsilon_{ij}\right) + \frac{Q}{\rho C_\epsilon} \qquad with\ ij = 1,2,3. \tag{1}$$

where ΔT is the change in temperature, T is the absolute temperature, ρ is the mass density, $C_\in$ is the specific heat at constant strain, σ_{ij} is the stress tensor, $\in_{ij}$ is the strain tensor, and Q is the heat energy of the material. This is an equation that can be applied to various material types; however, this specific utilization of this effect requires some additional conditions be applied to the equation to simplify it. These additional conditions are the following:

1) No heat transfer into or out of the system, or adiabatic conditions,
2) Homogeneous and isotropic material, and
3) Loading limited to the linear elastic range of the material [2].

When these conditions are observed, Equation (1) can be reduced to the following:

$$\Delta T = -\frac{\alpha}{\rho C_p} T \Delta\sigma \tag{2}$$

where α is the linear coefficient of thermal expansion, ρ is the mass density, C_p is the specific heat at constant pressure, T is the absolute temperature, ΔT is the change in temperature, and $\Delta\sigma$ is the change the bulk stress, or sum of the principal stresses [3]. This equation is the crux of the thermo-elastic stress analysis system utilized in this work. Further explanation of this is included in the following section.

2.2 Thermo-elastic Stress Analysis

Thermo-elastic stress analysis (TSA) is a non-contact instrumentation method that utilizes an infrared camera to capture the thermo-elastic effect displayed by a material under elastic loading. The thermo-elastic effect occurs when a material undergoes a temperature change when subjected to loading, either tensile or compressive or a combination of the two, under adiabatic conditions (i.e., no transfer of heat into or out of the test system). This condition is an assumption made for the thermo-elastic effect, as the thermal camera utilized captures minute changes in temperature. The origin of TSA is that of Stress Pattern Analysis by Thermal Emission or SPATE. This method "relies on the infrared detection of minute temperature changes that accompany stress changes" [2]. An excellent discussion and overall overview of TSA as a general theory is well summarized in *A review of the general theory of thermoelastic stress analysis* by Pitarresi and Patterson [4].This work clearly demonstrates the ability of TSA to capture the first stress invariant (i.e., sum of principal stresses) of the surface of a component during adiabatic conditions, and under the required cyclic loading. This work also provides an in-depth discussion on the principles behind the detectors utilized in the thermal cameras to detect photon flux, which is radiated from the surface of the material.

2.2.1 Microbolometer Thermo-elastic Evaluation (MiTE)

Microbolometer Thermo-elastic Evaluation or MiTE is a TSA capability that was created specifically to be utilized with a microbolometer. A microbolometer is "a thermal detector that relies on absorption and thermal conductance for the transduction of radiant energy to an electrical signal" [5]. This specific detector differs from those typically utilized in larger TSA systems, as they use a cooled photon detector, which can either be cooled by a compressor, or in the older systems, liquid nitrogen. The microbolometer detectors are suggested by Rajic and Rowlands [5] as a comparable alternative to their photon detector counter parts for various reasons, among the most important is practicality. Included in their argument for practicality is the reduced size and lower cost of the camera when compared to the significantly larger, and more costly photon detectors. As TSA is further researched and utilized for model validation, and general structural testing, these factors become more important in its ability to be utilized in as broad of applications as possible. That being said, this specific TSA system was chosen to be utilized for this endeavor to evaluate the advantages of using a microbolometer system, as well as to develop criteria to correlate between the images produced by this system and finite element models that are created to predict component behavior.

2.2.2 TSA Applications

For model correlation, experimental data is required to determine how closely the model was able to predict the material response captured through experimentation. Depending upon the complexity of the component being tested, the boundary conditions being applied to the component, and the data required for model validation, the instrumentation requirements may simply be strain sensors, or thermocouples, or other types of traditional, contact methods of acquiring test data. However, the more complex the geometry, and the need for more than just point (local) measurements, the more a non-contact method of data acquisition is required. Thermo-elastic stress analysis is a type of non-contact instrumentation method that has been implemented on various types of tests that require the use of a non-contact instrumentation system to capture data. This type of non-contact instrumentation system is especially useful for capturing full-field data, as opposed to the traditional thin-film/foil strain gage that can only capture data on a point-by-point basis, averaged over the area of the grid section of the gage. This requires additional time in test set-up, as well as limits the area over which the data can be captured for smaller components. The TSA system utilized in this work was employed to study the bulkhead of the F/A-18 aircraft [5], as well as various locations on the F-35A aircraft [3]. While these are much larger applications for this system, there are also smaller uses for this system, such as coupons with a hole to measure stress concentration values, or four-point bending of a beam. Details of the experiments in which this was utilized can be seen in the journal articles by Rowlands and Rajic [5] and Rajic et al [3].

In addition, there have been experiments conducted that specifically compare the results of strain gage readings to that of TSA system readings to determine how well they compare. Casavola, Caterina, et al [6] employed a Stress Photonics DeltaTherm 1560 cooled photon detector to measure the evolution of loading in a butt welded joint to determine which would better explain local phenomenon occurring at the weld toe in terms of fatigue strength. This TSA system was able to explain local phenomena that was occurring, causing an increase in the strain field around the joint, and was able to aid in determining where failure occurred, and why, as it was not in the location predicted, proving the usefulness of tracking the change of phase when determining damage in a specimen, as well as crack initiation. From this experiment, they were able to ascertain the usefulness and reliability of the TSA system for “non-destructive inspection of complex structures under load service” [6]. Another application of the comparison from TSA to linear strain gages is demonstrated in a biomechanical application, in which the fatigue strength and shielding strength of implants and whole bones were determined. TSA is utilized to determine high stress areas that would put “areas of whole bone at risk of mechanical failure”, as well as areas of low stress to assess stress shielding risks [7]. Metals and biomedical materials (bone and implants) have proven the usefulness of TSA, but this technology can also be useful in composite materials. A study was done to compare this technique with that of an ultrasonic probe, fiber optic strain sensors, and digital image correlation on a stiffened composite structure. Traditional thin-film/foil strain gages were also installed on the surface, and the results were discussed in the work; however, the ultrasonic data was used as truth data. This was an experiment that proved TSA was able to estimate delamination areas well when compared to the truth data, and had an average percent error of 3.52% [8]. While this is not specifically a microbolometer system that is being utilized, the principle of the thermo-elastic effect and TSA are employed using a photon detector.

As demonstrated in the previous paragraph, there are a wide variety of uses of this system; however, the quality of the thermal images produced by this specific system has not yet been explored or quantified. As previously stated, this particular system as a higher NETD (thermal sensitivity) value, which could influence its propensity to noise. This could be demonstrated through contrast in the image, or a reflection due to the uncooled nature of the device. However, image quality specifically of thermal images has not been assessed with this TSA system. This is something that should be explored and quantified to aid development of correlation criteria for this non-contact technology.

2.3 Image Analysis

Thermo-elastic stress analysis is a non-contact instrumentation method that utilizes an infrared camera to capture the minute changes in surface temperature of an elastic component under cyclic loading. The specific system utilized in this experimental endeavor produces images captured over a set number of cycles, and averages them together to produce an image for that specific load level. To aid in determining the ability to correlate this data to the data produced from a finite element model, an analysis of the images produced is suggested, calculating image indices that would provide quantitative values for comparison and determination of image quality. This would make apparent any error within the image data produced when analyzed over three spectral bands, utilizing a multispectral analysis Matlab program that produces eight image indices, three of which are utilized in this work. An overview of this subject, its uses, and the application to this experiment are covered in the following sections [9], [10].

2.3.1 Multispectral Imaging

The electromagnetic spectrum is broken down into various frequencies of electromagnetic radiation, and ordered by wavelength and photon energy associated with each type. Along this spectrum we have electromagnetic radiation that is visible to the human eye, eg. the visible light spectrum, which includes three basic spectral bands: red, green, and blue. On the end of this visible light spectrum is the infrared light spectrum, which is not visible to the human eye, and requires the use of a photon detector, or infrared camera, to capture the data associated with this wavelength range of the electromagnetic spectrum. To visualize the data collected at the infrared wavelength, this data is displayed as an R-G-B visual image, with a scale from white to black. These digital images can be analyzed at these color bands, and image indices produced that enable the analysis of the quality of the images when compared to a reference image. Currently, this type of image analysis is utilized in satellite imagery to detect environmental changes. However, this technology is one that is applicable to graduated changes in loading, as the metadata stored in the images at each (x,y) coordinate pair is the calibrated strain value. Therefore, the strain changes will be the comparative factor across the load level and test sequences. This type of image analysis utilizes eight image indices, however, some of these are specific to multi-spectral analysis of satellite imagery. Therefore, only three of these indices were used, with the quality image being broken down into the three factors that make it up for further understanding of the sources of error in the quality index [9], [11].

2.3.2 Image Indices Utilized

The multispectral image analysis Matlab program [10] employed for this work calculated eight different image indices from the color image series input. To calculate these indices, a reference image is required, which is then compared to a desired number of test images. The reference image serves as the “x” factor and the test images individually serve as the “y” factor for the calculation of all the image indices. In these image index equations, the metadata contained in the images that is being compared is the calibrated strain value at each color band, as well as the average. These image indices are bias, difference in variance (DIV), correlation coefficient (CC), entropy, *erreur relative globale adimensionnelle de sysnthesis* (ERGAS) or Relative Dimensional Global Error, image quality (Q), relative average spectral error (RASE), and root mean square error (RMSE). This Matlab program was designed to calculate all eight of these indices, however, for this endeavor, additional lines were included to calculate the Luminance Distortion and Contrast Distortion factors within the quality (Q) index. Values for all 10 of these indices were produced and provided as an Excel spreadsheet; however, not all of these values were utilized in this work, therefore, only the factors used will be discussed. The equation and an explanation of the equation for each of the indices calculated and utilized is included in the following subsection [9].

2.3.2.1 Image Index Equations

The correlation coefficient (CC) image index is calculated using Equation (3) [9].

$$CC = \frac{\sigma_{xy}}{\sigma_x \sigma_y} \tag{3}$$

In this equation, the correlation coefficient is defined as the covariance of x and y divided by the product of the standard deviation of x and y. This value can aid in determining the linear correlation of the factors, x and y, due to the value of the correlation coefficient being a value between negative one and positive one. If the value is positive, then there is a positive linear correlation between x and y; however, if the value is negative, there is a negative linear correlation between the factors. If the value is negative one, zero, or positive one, then the factors are perfectly, negatively, linearly correlated, or there is no linear correlation, or the factors are perfectly, positively, linearly correlated, respectively. Therefore, for best agreement, the value of the correlation coefficient would need to be either highly negative, or highly positive.

The quality index (Q) is calculated by making use of Equation (4) [9].

$$Q = \frac{4\sigma_{xy}\bar{x}\bar{y}}{(\sigma_x^2+\sigma_y^2)(\bar{x}^2+\bar{y}^2)} \tag{4}$$

In this equation, σ_{xy} is the covariance of the factors x and y, $\overline{x}$ and $\overline{y}$ are the mean of x and y, and σ_x and σ_y are the standard deviation of the corresponding factors [9].

This equation was calculated both as a whole, and broken down into components of distortion, in accordance with the following equation from Wang and Bovik [12]:

$$Q = \left(\frac{\sigma_{xy}}{\sigma_x \sigma_y}\right) \cdot \left(\frac{2\bar{x}\bar{y}}{(\bar{x})^2 + (\bar{y})^2}\right) \cdot \left(\frac{2\sigma_x \sigma_y}{\sigma_x^2 + \sigma_y^2}\right) \tag{5}$$

From this equation, Wang and Bovik [12] have proposed to model the distortion in an image in three factors: "loss of correlation, luminance distortion, and contrast distortion." The first factor is the correlation coefficient, which is a measure of "the degree of linear correlation between x and y" [12]. This is a factor that is separately calculated in the image quality analysis. The additional two indices are the luminance distortion, which is a measure of the closeness of the mean luminance between the two factors, and the contrast distortion, which is a measure of how similar the contrasts are between the two factors [12].These factors should be considered when utilizing the microbolometer infrared camera because this has a higher NETD, or thermal sensitivity, value. This could make this particular thermal camera, a microbolometer, more susceptible to issues with noise within the image, which will show up as contrast noise. Evaluating this component of the quality index will enable the determination of sources of error that might degrade the quality of the thermal image. Utilizing this as an application to thermal images with color will be a novel expansion on the current use of these distortion factors within the quality image index, as these were previously explored on a black-and-white image only.

The root mean square error (RMSE) is calculated by using the following equation [9].

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(x_i - y_i)^2}{n}} \quad (6)$$

In this equation, the square of the difference between the test image and the reference image is summed over the number of images, and then the square root of this value is taken. This provides a value utilized to determine the difference between the selected value in the reference image and the test images, and how well they fit a regression model. The difference between the reference and test images should follow a positive correlation trend, or have a positive slope as the load increases, as the strain will increase as the load increases. This error term will provide a quantitative understanding of how much the strain values deviate from a line of best fit from the lower load level to the maximum load allowable to remain in the elastic regime of the material [9].

Each of these indices are calculated for each spectral band displayed in the digital image. In this case, the spectral color bands displayed are red, green, and blue, or in the Matlab program they are denoted as Band 1, Band 2, and Band 3, respectively. When the image is broken down into its metadata, it can be broken down into greyscale images. In addition to each band, these indices are calculated as an average for each comparison across the three spectral bands [9], [10].

2.4 Model Correlation

Model correlation is a process by which data gathered through experimental means is compared to a computer-based model of a component, and used to predict the behavior of said component under a variety of boundary conditions. The goal of model correlation is to show a relationship between the model and the experimental data, more specifically that they show good agreement between predicted and actual behavior in response to the metrics analyzed through experimental testing. This will require the consideration of what instrumentation will be required to capture the material response metrics being utilized for correlation. Traditional contact-based instrumentation used to capture data for model correlation are strain gages, thermocouples, and deflection sensors. Considerations for this are the number of sensors required, the location these need to be placed in, and proper installation to reduce error. However, for a non-contact instrumentation system, some of these concerns can be alleviated as channel count will not be required, and the surface emissivity of the component is the largest concern. Utilizing traditional strain sensors in the experimental portion of this work will provide necessary test data to determine percent error between the thermal images and the model data. From this data, the predicted value of the model data can be compared to the measured data of the strain gages and the TSA system. This can be done by plotting these values against one another and creating lines for +/- 10% of the perfect correlation line. Upon seeing how the model compares to the different instrumentation, the trends in the image indices can then be assessed and a determination of the values acceptable for model correlation can be determined. If the values do not correlate well, using the percent error can provide an error bound on the correlation value, and if this value is low in comparison, can provide additional criteria for exhibiting the lack of correlation [1].

2.5 Novelty to this work

While image analysis has been previously accomplished utilizing these image indices, this was specific to hyperspectral images or satellite imagery. This approach aims to begin development of test-to-model correlation criteria using three of the indices utilized in previous work, taking into account the distortion factors that make up the quality image index (luminance distortion and contrast distortion). These are important to understand, specifically the contrast distortion, as this type of uncooled thermal sensor has a higher thermal sensitivity value, and would be more sensitive to noise. This noise will appear in the image as contrast distortion, therefore, impacting the quality of the image produced.

CHAPTER III

EXPERIMENTAL METHOD

3.1 Test Article Description

The titanium alloy Ti-6Al-4V was chosen as the material to investigate for this experiment as it is a well-known metal with a wide variety of aerospace applications. Additionally, due to the cyclic requirements for this test endeavor, a material that is linearly elastic is an ideal choice. This specific alloy of Ti-6Al-4V was mill annealed and machined into a dogbone configuration for this application. The geometry of the specimen was designed such that multi-axial bending would be present in addition to the tension from axial loading, to create a complex stress-state.

3.1.1 Specimen Geometry

A Ti-6Al-4V mill annealed dogbone (Figure 1) contains a 0.281-in. through-hole longitudinally centered within the 3.604-in. gage section. Load was introduced through pin-loading by means of two holes at the ends of the specimen, equidistant from the centered hole, and offset from the transverse centerline to generate an axis of bending. Shims and bearings were placed within the holes to accommodate the use of a 0.75-in. pin-clevis fixture.

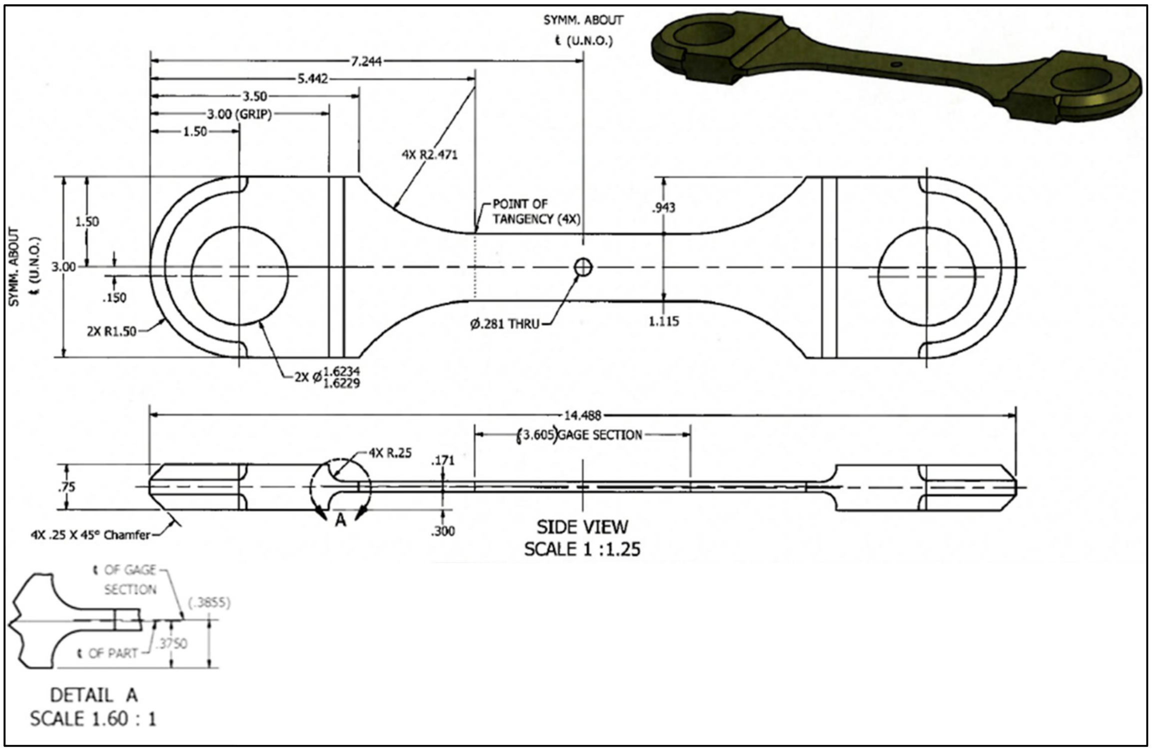


Figure 1. Dogbone Specimen Detailed Geometry

3.1.2 *Specimen Instrumentation*

The dogbone specimen was instrumented in two phases, first for initial constant-amplitude dynamic loading, and second for static loading to obtain the strain gradient within the gage section of the component. Each phase of instrumentation involved surface preparation and installation of thin-film/foil strain gages.

For the first phase of instrumentation, two CEA-05-125UW-350 uniaxial strain gages were installed on the front and back faces of the specimen, directly opposite of one another, and centered 1.00-in. below the 0.251-in. center hole. In addition, two WK-05-062AP-350 uniaxial strain gages were installed on the through-thickness faces of the specimen, directly opposite of one another, and aligned with the CEA gages. The layout for this instrumentation can be seen in Figure 2.

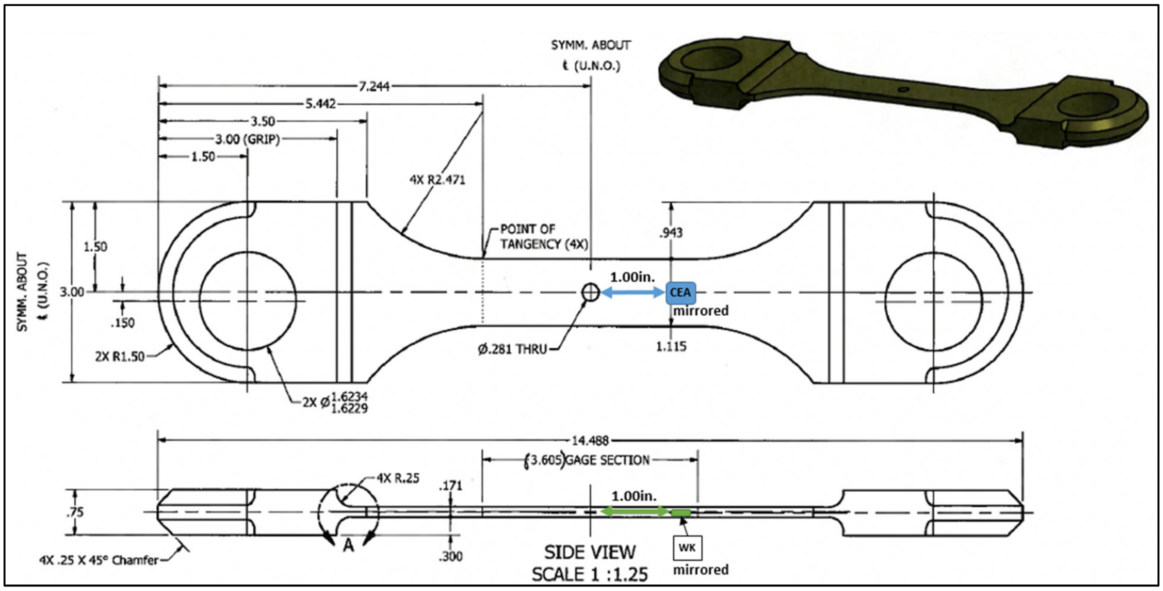


Figure 2. Dogbone Specimen Instrumentation Phase I Drawing

Both sets of strain gages were installed utilizing MBond 200 adhesive, in accordance with *Vishay MicroMeasurements Instruction Bulletin B-127-14* [13], wired with 4-foot, 30 American Wire Gage (AWG) wire, and RJ45 connectors installed for proper connection to the test frame signal conditioner. All strain gages were verified for proper functionality prior to test execution. To protect the gages from the coating necessary for proper data capture while using the TSA system, a thin layer of room temperature vulcanizing (RTV) silicone was applied to each strain gage.

For the second phase of instrumentation, sixteen EA-13-062AQ-350 uniaxial strain gages were installed on the front and back faces of the specimen, directly opposite one another so as to mirror the front to the back, with four gages on either side of the hole, the center of the grid on each gage aligned with the centerline of the centered hole, and the solder pads alternating top and bottom from the left edge of the specimen to the right edge of the specimen. The solder pads on the gages were alternated top and bottom to enable four gages to be placed on either side of the hole, as well as to prevent any chance of creating a solder bridge across two gages from the proximity to one another. Due to the limited width of the specimen and the footprint of the strain gages selected for this additional instrumentation, each gage was required to be cut down to a smaller size, trimming the footprint of the gage down to 0.106-in. This also required that the gages be installed in batches, or four at a time, corresponding to each side of the hole. This would enable the technician installing the gage to prepare the alignment of the gages on a piece of Kapton tape (DuPont) and then transfer them to the specimen surface. Also, an additional set of larger solder pads were installed 0.50-in. away from the strain gages to enable the use of 30 AWG wire and RJ45 connectors, as these are the standard wire size and connector type in the test environment used for this experimental work. The whiskers are very fine wires that connect the solder tabs from the grid of the gage to an exterior set of solder tabs. These whiskers from the gage were soldered to the larger set of solder pads, and 4-foot 30 AWG wire leads were soldered to the same pads (ensuring proper isolation for each gage), with RJ45 connectors. This set of gages and solder pads were installed using AE-10 adhesive, and verified for proper functionality prior to test execution. An image depicting this instrumentation layout can be seen in Figure 3.

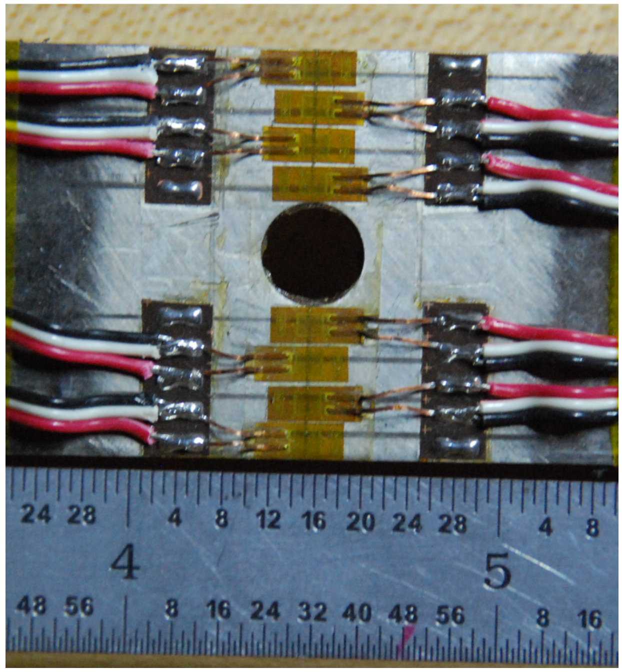


Figure 3. Phase II Instrumentation Layout per Side

3.1.3 Surface Preparation

The TSA non-contact instrumentation system utilized for this experimentation requires a surface coating for proper image and data capture. As there were two phases of instrumentation applied to the specimen, the TSA surface coating was applied, constant-amplitude testing executed, and the coating removed prior to the application of the second phase of strain gage instrumentation. The second phase of instrumentation required removal of the TSA coating so that the installation area could be properly cleaned and prepared for gage installation. However, the entirety of the coating was not removed; just the area required for installation around the centered hole.

3.1.3.1 TSA Surface Coating

The coating applied to the dogbone specimen was Dupli-Color, a flat black rubberized spray paint that provides a uniformly emissive surface for the TSA system. Prior to applying this coating, the surface of the specimen was thoroughly cleaned with isopropyl alcohol to ensure all debris had been removed from the surface, and the area of interest was taped off with 1.00-in.-wide FrogTape. A thin layer of Dupli-Color was applied, and allowed ample time to dry. This was repeated until an evenly coated surface was produced, with minimal thickness added to the specimen. An image depicting this surface coating can be seen in Figure 4.

Figure 4. TSA Surface Coating – Dupli-Color

3.2 Experimental Setup

To execute testing on this dogbone specimen, an MTS axial servo-hydraulic load frame with a 22-kip capacity load cell was used, as well as MTS FlexTest control software and signal conditioner for the strain gage channels. This load capacity requirement was determined through model analysis of the component, which determined that this would be a load within the elastic regime of the material and component geometry and allow for repeated testing to capture significant data. As is standard procedure for this test equipment, *ASTM E4: Standard Practices for Force Verification of Testing Machines* [14] was followed to verify the alignment of the test frame and calibrate/verify the load output to ensure accuracy of the load and strain measurements produced by the MTS machine and signal conditioner. The fixturing used for this experiment was a pin-clevis fixture designed to support a 0.75-in. pin. This specific clevis influenced the design of the dogbone longitudinal ends, requiring a rounded end to prevent interaction between the square ends and the interior of the clevis. This also required a 0.25-in. bushing to bridge the gap between the 1.0-in. inner diameter of the dogbone ends and the outer diameter of the clevis pin, ensuring a snug fit that could be shimmed, if necessary. It was a necessity to shim the clevis utilizing 0.015-in. thin metal shims, which were inserted between the spherical bearings and the bottom of the clevis.

The specimen was assembled prior to insertion into the clevis fixture, which included insertion of the spherical bearings and 0.25-in. bushings and installation of the various strain gages. The verification of their measurement accuracy and readout in the MTS FlexTest software was complete via a shunt calibration and verification of MTS FlexTest program setup. A procedure was created in accordance with the test matrix established for this experimental effort, as seen in Table 1, to include three test runs per load level. All testing was executed on the same specimen, with the maximum load levels varied from 1200-lbf to 4200-lbf.

Table 1. TSA Data Capture Test Matrix

Test Run	**Loading Type**	**Min. Load (lbf)**	**Max. Load (lbf)**	**R-value**	**Loading Frequency (Hz)**
1 -3	Tensile	120	1200	0.1	2.0
4 - 6	Tensile	220	2200	0.1	2.0
7 - 9	Tensile	320	3200	0.1	2.0
10 - 12	Tensile	420	4200	0.1	2.0

The procedure for testing the dogbone specimen included the following steps:

1. In accordance with *ASTM E4* [14], align the specimen within the test frame.

2. Define the load cases and loading frequency for interrogation through material property and FEA model investigation.
3. Execute tensile testing at each load case.
4. Acquire TSA data during cyclic loading of the specimen at each load case at a frequency of 2.0 Hz.
5. Repeat each load case, for a total of three replicates per load case per specimen face.

The load cell and displacement transducer on the MTS load frame were employed to track the mechanical response of the specimens during tensile loading. Strain gages were used to record local strains in areas of high interest to confirm coherency with the FEA model and for use in the calibration of the TSA system. In addition to the traditional strain measurement systems, a Microbolometer Thermoelastic Evaluation (MiTE) TSA system was used to capture the strain field on the test article. The setup for this data acquisition system is covered in the following section.

3.2.1 MiTE TSA System Setup

The MiTE TSA system is a *standalone system* that requires the following equipment: a standalone laptop with MiTE and MiTEViewer softwares installed, Power over Ethernet (PoE) injector, two Ethernet cables to connect the camera to the computer, an FLIR A35sc test kit (including an FLIR A35sc camera, blue focus ring, and PoE injector), mounting equipment, a National Instruments USB-6000 Multifunction Input/Output (I/O) data acquisition device, black moving blankets, brown packing paper, and green FrogTape. This experimental setup is shown in Figure 5.

Figure 5. MiTE TSA Camera System Setup

To setup the MiTE TSA system, Steps 3 through 5 in the *MiTE Installation and User Guide* [15] were followed, which provides adequate detail on how to pair the NI USB-6000 data acquisition device with the laptop, connect the FLIR A35sc camera (specifically addressed in Step 4B), and ensure power is being supplied to all devices in the system and that they are properly connected. Once the system was properly set up, the angle of the camera was established for proper data collection. To do this, Steps 7 and 8 [15] were followed to start the MiTE software, and “Live Mode” was utilized to focus the camera’s image. Prior to focusing the image, however, black moving blankets and brown packing paper were utilized to block out ambient light from the laboratory environment where the experiment was conducted. The ambient light in this environment can cause distortion in the image, and degrade the signal-to-noise ratio in the collected data. The camera was placed in front of the specimen at an upward angle, approximately 10° from the y-plane (face of the specimen), ensuring the centered hole was kept clearly within view, but avoiding a reflection of the thermal energy from the heat produced by the camera during use.

Microbolometers do not include a compressor, which enables a lighter, more compact device for data capture. However, this produces reflected thermal energy that requires the user to coat the surface of the material (this material specifically because of the mirror finish of the metal, which will produce a great deal of reflection without a high-emissivity coating applied to the low-emissivity surface of the specimen). An additional measure to mitigate the thermal reflection, and thereby heat build-up in the camera, being captured during data collection was to remove the power source to the camera, enabling the camera to cool. Caution needs to be taken when removing the Ethernet cable feeding the camera’s power so that the position of the camera is not adjusted, as the mount is not perfectly rigid. The MiTE software was then setup for data capture, which required determining the correct setting for “blocks to capture” and the “cycles per block”. It was determined that 800 cycles per load level would produce the appropriate amount of data images to be averaged together to produce a clear image within the MiTE software; therefore, the “cycles per block” was set to 30, and the “blocks to capture” were set to 20.

With the frequency of the load frame set to 2.0Hz, this would seem to produce 600 cycles, however, through experimentation, this aligned more closely to the 800 cycles utilized for this experiment. Finally, the camera calibration parameters must be input "in order to make unbiased thermoelastic response measurements" [15]. There are three parameters that must be input: line scan delay, time constant, and transport delay. The transport delay must be experimentally determined; however, the line scan delay and time constant are determined by the camera. The values for this particular set up were -63µs, 12.3ms, and 42.584ms, for the line scan delay, time constant, and transport delay, respectively. These values, and how to determine the transport delay, are included in Step 9 of the *MiTE Installation and User Guide* [15]. Three test runs were executed for each of the four load cases, and this was repeated for each face of the specimen, denoted "Side A" or "Side B". These sides describe the front and back faces of the specimen, respectively.

3.2.2 Strain Gage Setup

The testing on this specimen was conducted in two phases, requiring a slightly different strain gage set up for each test phase, as well as different size gages. The first phase involved two gages on the thru-thickness faces and two strain gages, one on the front face and one on the back face. The second phase involved sixteen gages, with a 1/16-in. grid size aligned on the centerline of the 0.281-in. centered thru-hole, with four strain gages on either side of the hole, creating a total of eight strain gages per face (front and back).

3.2.2.1 Strain Gage Setup Phase I

The first phase of instrumentation was required during data capture for the TSA data, as this requires a far-field strain value to calibrate the (axial) strain map, which can be converted to an axial stress map utilizing the modulus of elasticity of the material. This required determining the correct locations at which to install the strain gages. To ensure no edge effects from the centered hole were captured in the strain data, the two CEA-05-125UW-350 uniaxial strain gages were placed at a distance at least twice that of the diameter of the hole, or at least 0.562-in. from the bottom edge of the hole. For ease of strain gage layout and installation, this distance was increased to 1.00-in. from the bottom edge of the hole, where the center of the strain gage's grid section would be located. This portion of the strain gage is the area that is encompassed by the grid loops on the upper portion of the gage, above the solder tabs. The two CEA strain gages were installed in the exact same location on the front and back face, mirroring them, so that they could be compared with one another in later testing. This phase of instrumentation was utilized for the entirety of the three TSA testing sequences, and as such, created data sets for each set of TSA images collected, creating a total of twenty-four MTS data files for analysis and comparison. This will be discussed in both the "Results" and "Discussion" sections in Chapter 4 and 5 in further detail.

3.2.2.2 Strain Gage Setup Phase II

The second phase of instrumentation was required to determine the strain gradient across the centerline of the gage section of the specimen. This enabled the capture of a series of four points on either side, left or right, of the centered thru-hole, for comparison to the opposing face, as well as to the TSA non-contact data previously captured in this effort. The placement of these additional strain gages was determined, therefore, to be along the centerline of the centered hole, with the center of the strain gage grid area along this line for each strain gage. The signal for these strain gages was conditioned utilizing the MTS FlexTest software, just as the gages in the previous phase were, but, utilizing a different gage factor, and the channels verified for appropriate readings prior to test. Once verified, the specimen was subjected to loading in accordance with Table 2.

Table 2. Strain Gradient Data Capture Test Matrix

Test Run	**Loading Type**	**Max. Load (lbf)**
1	Tensile	1200
2	Tensile	2200
3	Tensile	3200
4	Tensile	4200

Each test conducted in the above test matrix had a duration of 10-15 seconds to allow for stabilization at the specified load.

CHAPTER IV

RESULTS

4.1 Results Overview

The results from the experimental testing of this dogbone specimen were compared in multiple ways, encompassing the TSA system and the strains from each phase of instrumentation. The strains from the strain gages for each side of the specimen were compared directly to one another at each load level and compared to those seen by the model. The TSA images were analyzed utilizing the MiTEViewer software, as well as compared to one another to establish the image quality indices at each load level when compared to the static image. These image quality indices were then utilized to aid determining the degree to which the images matched the model. Based on this comparison, model correlation criteria was developed for use in evaluating the validity of the model developed.

4.2 Strain Gage Data Results for Phase I

The instrumentation of this dogbone specimen occurred in two phases, the first being four strain gages installed, two on the front and back faces, and two smaller footprint gages installed on the through-thickness faces of the specimen. Data from each of the four thin-film strain gages that were installed on the gage section of the dogbone specimen was captured in a text file within the MTS FlexTest software and exported to an Excel file so that this data could be plotted and compared from one test run to another, as well as from face-to-face of the specimen (Side A vs. Side B), for each of the four load levels. Each test run provided a text file with data from each gage for the duration of testing. To begin analysis, the maximum strain value from each gage was determined utilizing the “MAX” function within Excel and this value was recorded into a separate Excel workbook. This process was repeated for each of the twenty-four test files that were created.

From the physical response of the strain gages located on the front and back faces, for Side A and Side B of the specimen, the maximum strains were recorded at each load level, and the averages taken. Due to the minimal difference in strain gage readings at each level between the test sequences, the average was recorded and can be seen in Table 3.

Table 3. Average Maximum Strain Values at Each Load Level for Side A & B

Load Level (lbf)	**Side A, SG-01 (με)**	**Side B, SG-03 (με)**	**Delta, Side A to B (με)**
1200	389	348	41
2200	712	653	59
3200	1038	956	82
4200	1360	1263	97

The average maximum strain values were then compared per side by plotting them at each load level, and comparing the slope of each line. These figures were then grouped per set of strain gages to be compared, the front/back face for Side A and B, and the through-thickness gages, SG-01 and SG-03, and SG-02 and SG-04, respectively. The through-thickness strain values were compared to determine how much of a factor bending in the specimen would influence the difference in strain values obtained for the front and back faces. These sets of graphs can be seen in Figure 6 through Figure 9.

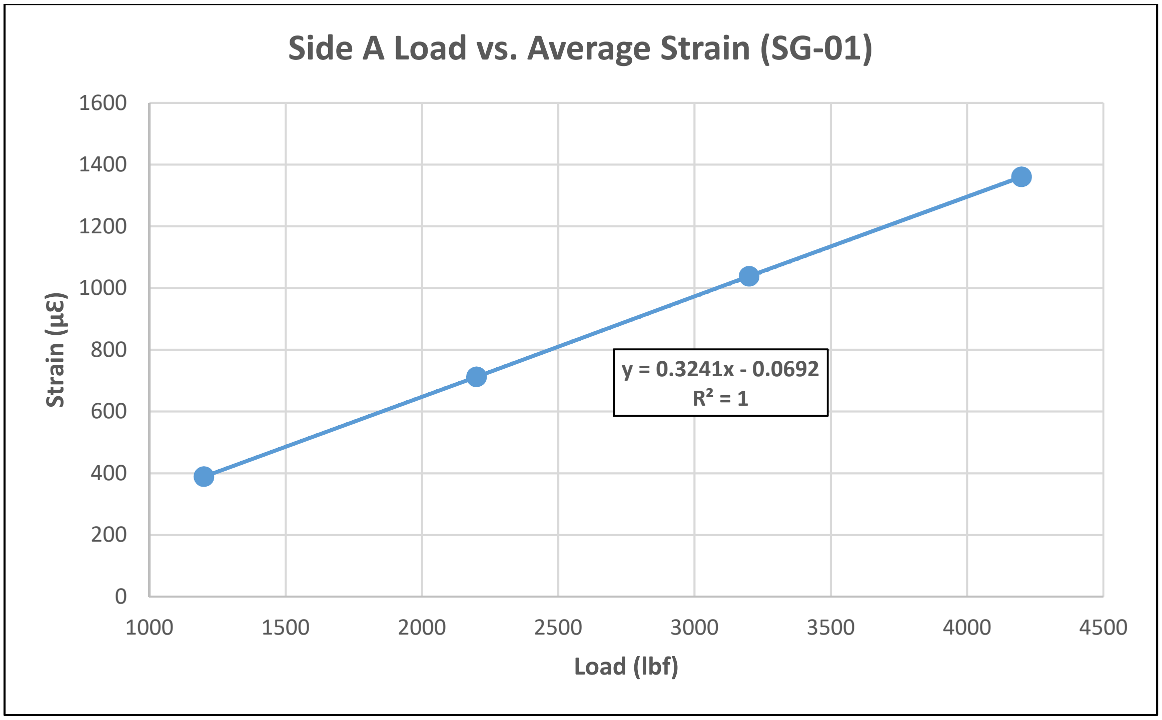


Figure 6. Side A (SG-01) Average Max Strain Value at Each Load Level

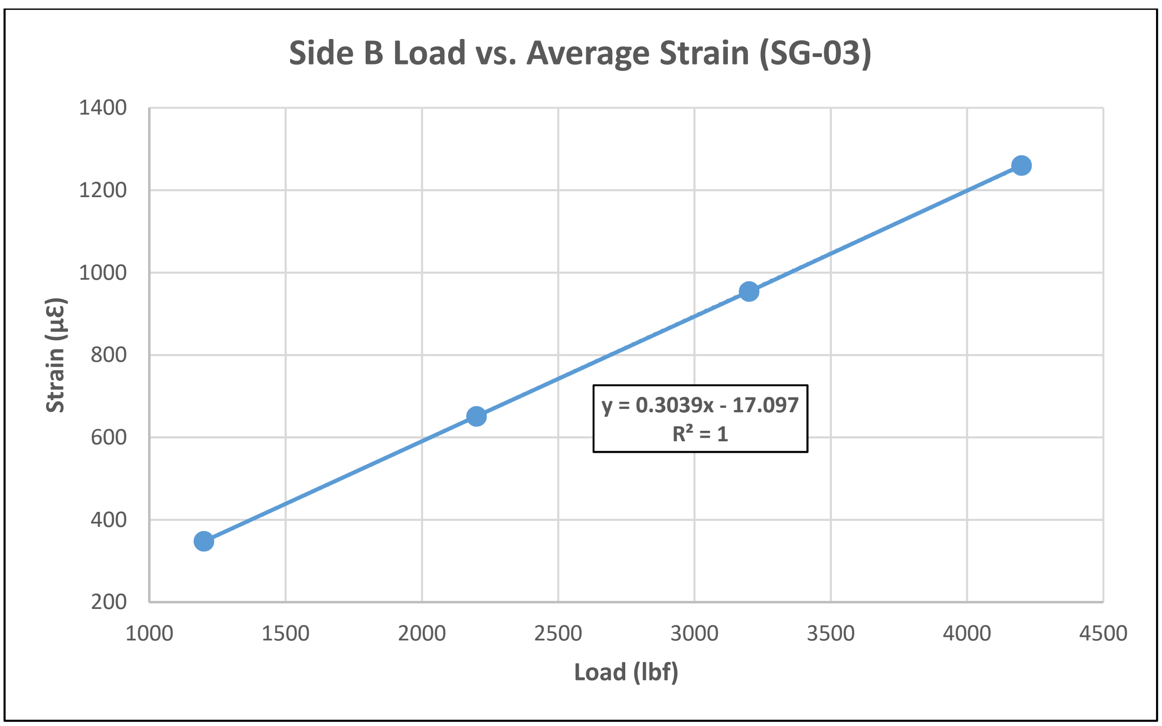


Figure 7. Side B (SG-03) Average Max Strain Value at Each Load Level

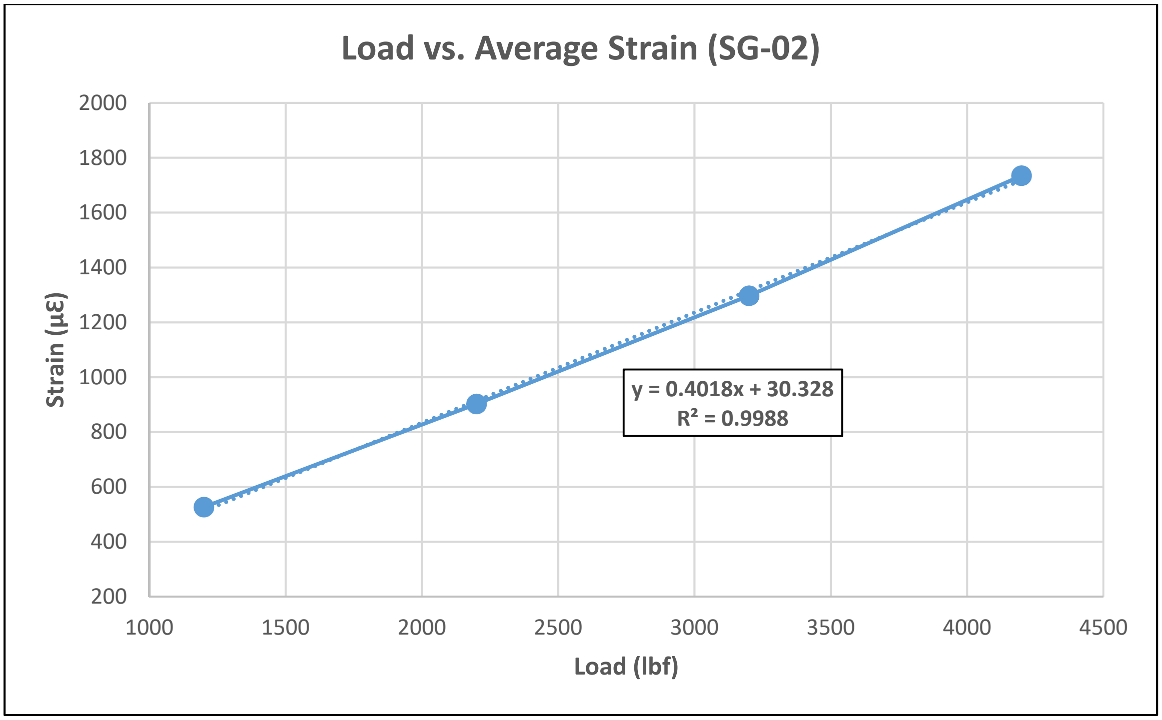


Figure 8. SG-02 Average Max Strain Value at Each Load Level

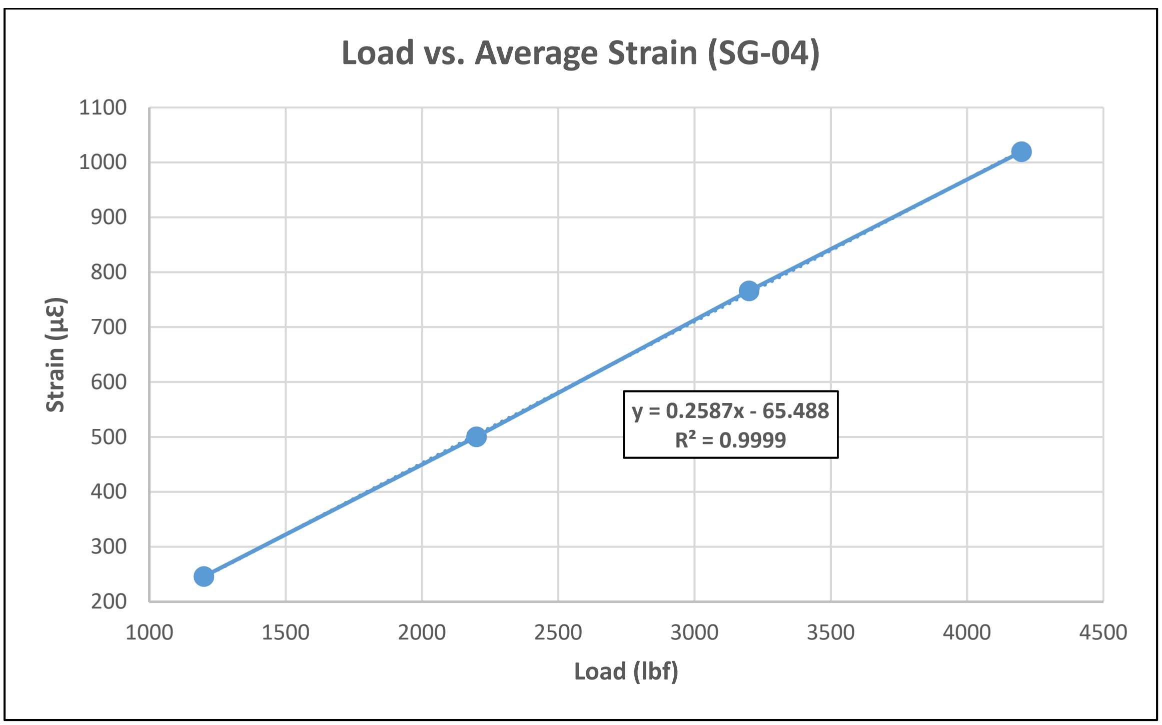


Figure 9. SG-04 Average Max Strain Value at Each Load Level

4.3 Strain Gage Data Results for Phase II

The second phase of instrumentation involved sixteen thin-film strain gages installed along the centerline of the centered hole in the dogbone specimen under constant-amplitude loading, with eight gages on the front face and eight gages on the back face. This was done to assess the strain gradient of the component for comparison to the TSA results obtained utilizing the MiTE software. To capture this data, the specimen was subjected to loading at each of the peak loads in Table 2 for a total of two test runs. Similar to the data from the previous four strain gages, the maximum value of strain recorded by each gage was calculated and recorded in a separate Excel file for further analysis. This was done for each test run. The data from each load level was then plotted in Excel to compare the gradient trends on either side of the hole, to compare the results from the front face to the back face, and to compare the results from test-to-test to ensure repeatability. These results are displayed in Figure 10 through Figure 13.

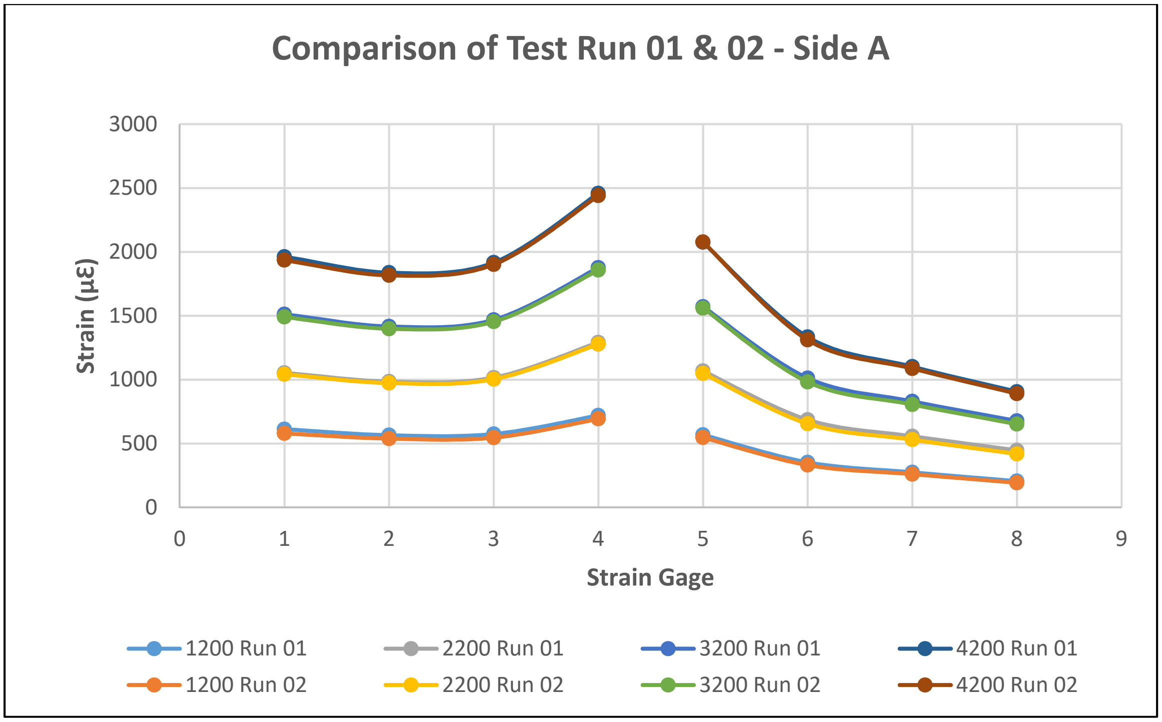


Figure 10. Comparison of Strain Results for Test Run 01 & 02 on Side A

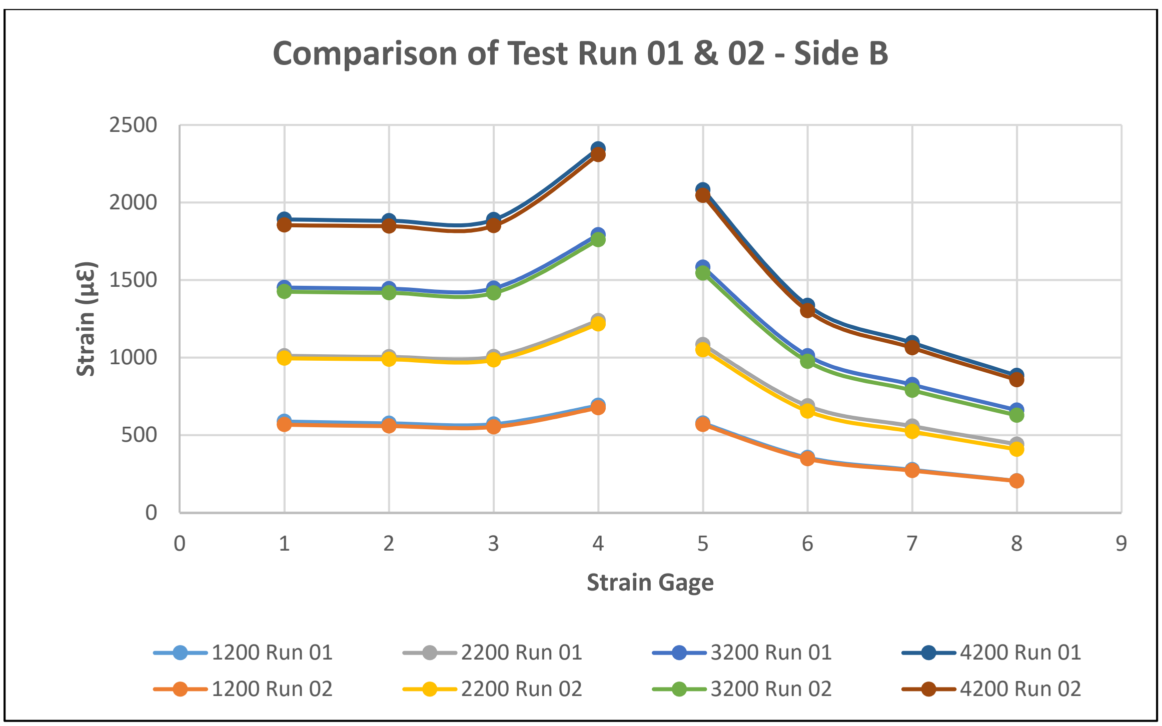


Figure 11. Comparison of Strain Results for Test Run 01 & 02 on Side B

Figure 12. Phase II Comparison of Side A to B for Test Run 01

Figure 13. Phase II Comparison of Side A to B for Test Run 02

The trends in the gages across the centerline from Side A to Side B appear to be in agreement, however, to verify this, the delta between the two test runs was determined, and the percentage difference was calculated. These values can be seen in Table 4 through Table 7.

Table 4. Strain Gage Max Value Delta (με) for Side A

Load Level (lbf)	SG-01	SG-02	SG-03	SG-04	SG-05	SG-06	SG-07	SG-08
1200	32.80	26.79	28.97	27.44	21.03	21.31	14.77	12.77
2200	11.71	11.86	12.75	13.92	20.11	30.65	26.75	28.86
3200	21.18	18.26	14.98	19.45	11.39	29.48	24.10	25.14
4200	26.91	21.43	14.83	18.16	-2.06	22.66	14.64	15.02

Table 5. Percent Strain Gage Max Value Delta (με) for Side A

Load Level (lbf)	SG-01	SG-02	SG-03	SG-04	SG-05	SG-06	SG-07	SG-08
1200	5.36%	4.74%	5.04%	3.81%	3.70%	6.04%	5.37%	6.20%
2200	1.11%	1.20%	1.25%	1.08%	1.88%	4.47%	4.80%	6.45%
3200	1.40%	1.29%	1.02%	1.04%	0.72%	2.91%	2.90%	3.71%
4200	1.37%	1.17%	0.77%	0.74%	-0.10%	1.70%	1.33%	1.66%

Table 6. Strain Gage Max Value Delta (με) for Side B

Load Level (lbf)	SG-01	SG-02	SG-03	SG-04	SG-05	SG-06	SG-07	SG-08
1200	20.57	17.57	17.96	14.95	8.83	8.04	5.78	0.72
2200	15.88	15.91	21.43	21.18	33.28	34.00	34.91	32.81
3200	26.58	26.26	31.15	31.56	37.90	36.77	36.59	33.25
4200	37.04	34.98	38.69	37.30	36.00	34.43	32.64	27.82

Table 7. Percent Strain Gage Max Delta (μƐ) for Side B

Load Level (lbf)	SG-01	SG-02	SG-03	SG-04	SG-05	SG-06	SG-07	SG-08
1200	3.50%	3.05%	3.15%	2.16%	1.53%	2.26%	2.09%	0.35%
2200	1.57%	1.58%	2.13%	1.71%	3.07%	4.94%	6.26%	7.44%
3200	1.83%	1.82%	2.15%	1.76%	2.40%	3.64%	4.43%	5.03%
4200	1.96%	1.86%	2.05%	1.59%	1.73%	2.58%	2.98%	3.15%

This procedure was repeated to compare from Side A to Side B from Test Run 01 to Test Run 02, providing the calculated delta between Side A and B for Test Run 01 and Test Run 02, and the percentage difference, or percent error, for the same data sets. These can be seen in Table 8 through Table 11.

Table 8. Strain Gage Max Delta (μƐ) for Test Run 01

Load Level (lbf)	SG-01	SG-02	SG-03	SG-04	SG-05	SG-06	SG-07	SG-08
1200	24.73	10.00	4.62	29.71	8.08	1.88	1.39	1.16
2200	42.56	19.19	11.23	54.48	14.06	2.16	0.09	6.57
3200	60.03	27.01	21.81	85.98	10.71	0.69	4.87	16.39
4200	71.08	44.65	27.82	113.35	5.81	1.49	7.16	21.80

Table 9. Percent Strain Gage Max Delta (μƐ) for Test Run 01

Load Level (lbf)	SG-01	SG-02	SG-03	SG-04	SG-05	SG-06	SG-07	SG-08
1200	4.04%	1.77%	0.80%	4.12%	1.42%	0.53%	0.51%	0.56%
2200	4.04%	1.95%	1.10%	4.22%	1.32%	0.32%	0.02%	1.47%
3200	3.97%	1.91%	1.48%	4.58%	0.68%	0.07%	0.59%	2.42%
4200	3.62%	2.43%	1.45%	4.61%	0.28%	0.11%	0.65%	2.41%

Table 10. Strain Gage Max Delta (μƐ) for Test Run 02

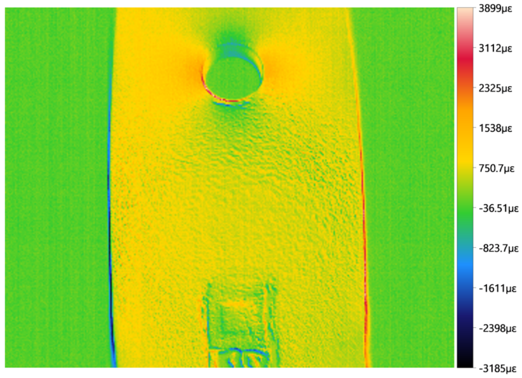
Load Level (lbf)	SG-01	SG-02	SG-03	SG-04	SG-05	SG-06	SG-07	SG-08
1200	12.49	19.22	6.38	17.22	20.28	15.16	10.38	10.89
2200	46.73	15.13	19.91	61.74	0.88	1.18	8.07	10.52
3200	65.42	19.02	37.98	98.09	15.80	7.98	17.36	24.51
4200	81.22	31.09	51.68	132.50	32.26	10.28	25.16	34.60

Table 11. Percent Strain Gage Max Delta (μƐ) for Test Run 02

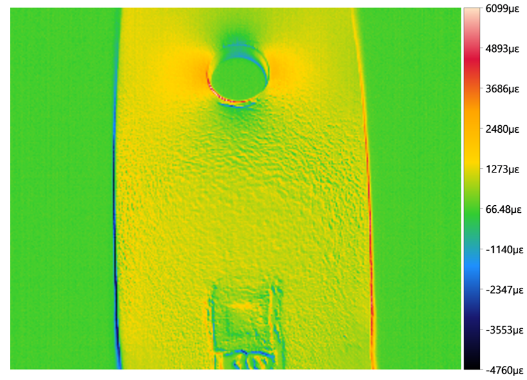
Load Level (lbf)	SG-01	SG-02	SG-03	SG-04	SG-05	SG-06	SG-07	SG-08
1200	2.15%	3.57%	1.17%	2.48%	3.70%	4.57%	3.99%	5.64%
2200	4.48%	1.55%	1.98%	4.83%	0.08%	0.18%	1.52%	2.51%
3200	4.39%	1.36%	2.61%	5.28%	1.01%	0.81%	2.15%	3.76%
4200	4.20%	1.71%	2.72%	5.43%	1.55%	0.78%	2.31%	3.88%

4.4 MiTE TSA System Results

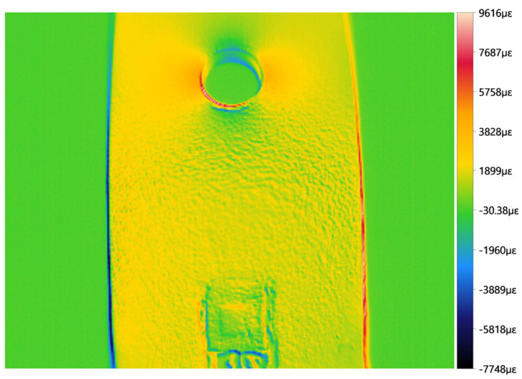
To analyze the individual images captured as data for the MiTE system, the onboard MiTEViewer software was utilized. The TSA images captured by the FLIR A35sc camera are an average of the images captured over the 800 cycles under which the specimen was sustaining loading from the MTS load frame. This is a raw image that requires an axial strain value, from an area of bulk strain or a known strain region, to calibrate the image into units of strain. This will then enable the visualization of the variation in stress across the specimen, as this strain map could easily be converted to that of stress utilizing the elastic modulus of the material. To calibrate the image with this known strain value, the value will be retrieved from the MTS system strain data for that test run and load level, and entered into the MiTEViewer software under the "Calibration" tab. Next, the pixel location within the image to place the calibration point was selected, converting the raw map of the thermal response into one of strain. The point chosen for these images was one on the strain gage within the image, located below the hole, typically near the top right corner. In addition to calibrating the in-phase component, the phase component (θ) also needed to be calibrated with in the corresponding data tab in the MiTEViewer software. To do this, "Calibrate to 0" in the left-hand pane was selected, and then the point to be used as this calibration point was selected. For this experimental effort, a point along the same horizontal line, but off to the right of the gage within the image was selected. This was repeated for each image, with some of the points of calibration being slightly different from one another. The images produced by this software for Test Sequence 01, both Side A and Side B, can be seen in Figure 14 and Figure 15.



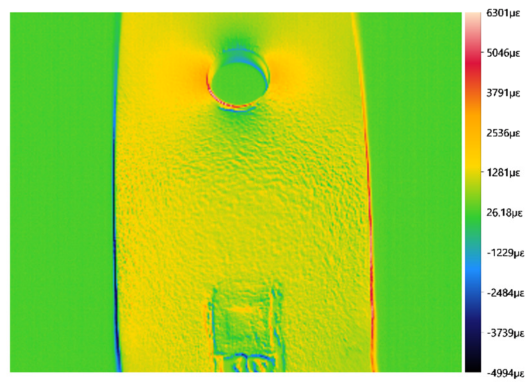
a) Side A 1200-lbf load case



b) Side A 2200-lbf load case



c) Side A 3200-lbf load case



d) Side A 4200-lbf load case

Figure 14. TSA Images for Side A for each load case, as labeled

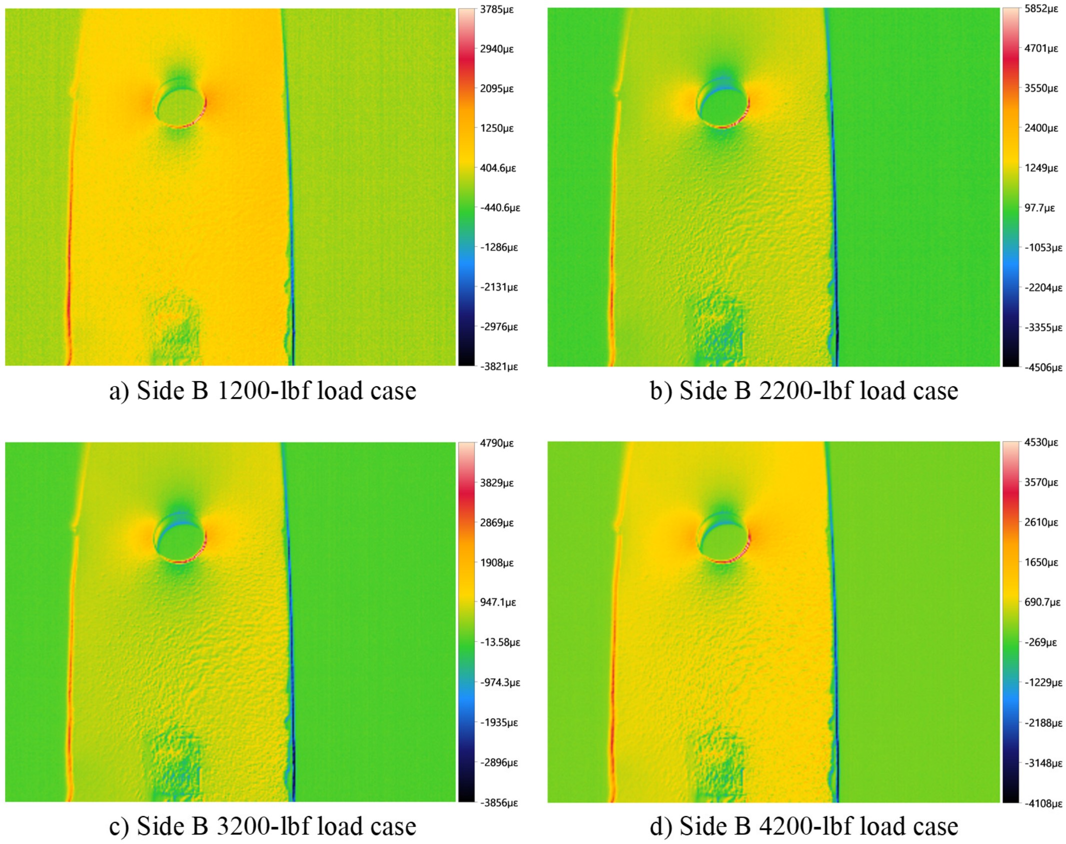


Figure 15. TSA Images for Side B for each load case, as labeled

Within the MiTEViewer, there are various tools to analyze the image, including a point measurement and line measurement. Due to the specimen geometry and the direction of loading, the highest values of stress were known to be located at the right and left longitudinal edges of the centered hole in the specimen, thus dictating that point values needed to be obtained at these locations. Within the software, under the Analysis tab, points can be selected and the X, Y, r, and theta value at that location will be provided. These values correspond to the in-phase strain value (X), the out-of-phase strain value (Y), and the phase of the image captured (θ). As this specimen was loaded in the direction perpendicular to the camera, the in-phase response will be captured and the phase calibration will be set to zero. For this experimental endeavor, the X-value, or in-phase strain value, was obtained at each edge of the hole. This was done for both Side A and Side B, with the values displayed in Table 12.

Table 12. Comparison of Hole Edge Values for Side A & B from MiTEViewer

Specimen Face	**Load Level (lbf)**	**Left Edge Point Value (με)**	**Right Edge Point Value (με)**
Side A	1200	887	305
	2200	2293	820
	3200	5198	2191
	4200	3934	2161
Side B	1200	952	2774
	2200	1408	4227
	3200	1253	3245
	4200	2146	4613

Due to the hole being a mirror image from Side A to Side B, the left edge point value for Side A would correspond to the right edge point value for Side B. From the data shown, the value taken at the 4200-lbf load case corresponds well from the front to the back face for the right edge, and is slightly off for the left edge point from front to back of the specimen. This led to a comparison of the strain gradient for the specimen from each side of the specimen, Side A and Side B. This was done by analyzing data along a line that is run through the center of the hole in the specimen within the TSA image within the MiTEViewer software. Under the "Analysis" tab, a line was drawn along the center of the hole within the image and a graph displaying the "Distance along Line (pixels)" versus "X(με)" was created within the software, at the bottom of the display window. This can be seen in Figure 16.

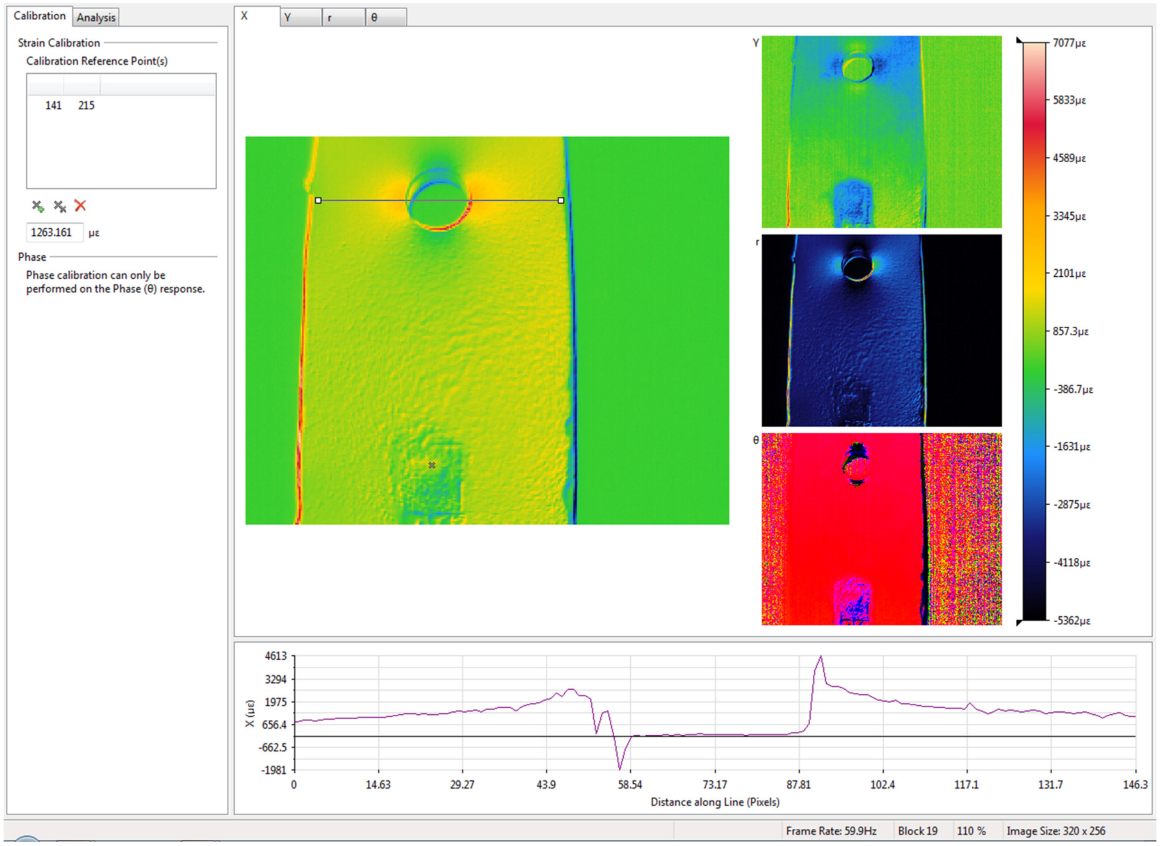


Figure 16. MiTEViewer TSA Data Analysis Image, Side B

The components of this line can then be exported to a .csv file, which can then be imported into Excel and utilized to create a line plot. This line plot can then be compared to the components of the model images, and from TSA image to TSA image. Due to the geometry of the component, peak data can be seen at the longitudinal edges of the centered hole, which decreased until a stable, far-field strain value was reached. The noise in the image appears to be low, due to the smoothness of the line created, which is a function of the focus of the camera, the length of data acquisition, and the proper sync between the MiTE system and the MTS load frame. This procedure was executed for each test run, ensuring to calibrate each image to the same pixel location on the strain gage and line coordinates for each image, making slight changes as needed for any change in camera placement.

4.5 Finite Element Model Results

A finite element model, provided by the University of Dayton Research Institute (UDRI), was created within the Abaqus modeling software for the purpose of analyzing the strain distribution in the gage section of the dogbone component. This analysis was to be done at peak load, and due to linear elastic assumptions that can be made due to the material, this analysis is scalable to the intermediate load states included in this work. Due to the symmetry in the component, and a reduction in computation time, the model utilized for this work was a half model, split at the centerline of the central hole being analyzed. Initially, to properly model the spherical bearing within the longitudinal, larger hole on the ends of the specimen, two boundary conditions were created: one being an applied load under tension in the axial direction of the component, and second a zero displacement boundary condition. The model was analyzed at each of the load levels and the simulated strain values in the x-direction obtained along the centerline of the centered hole, propagating from left to right. This was repeated for each side, Side A and Side B, for the component at each load level. The strain maps obtained for the 4200-lbf load case can be seen in Figure 17 and Figure 18.

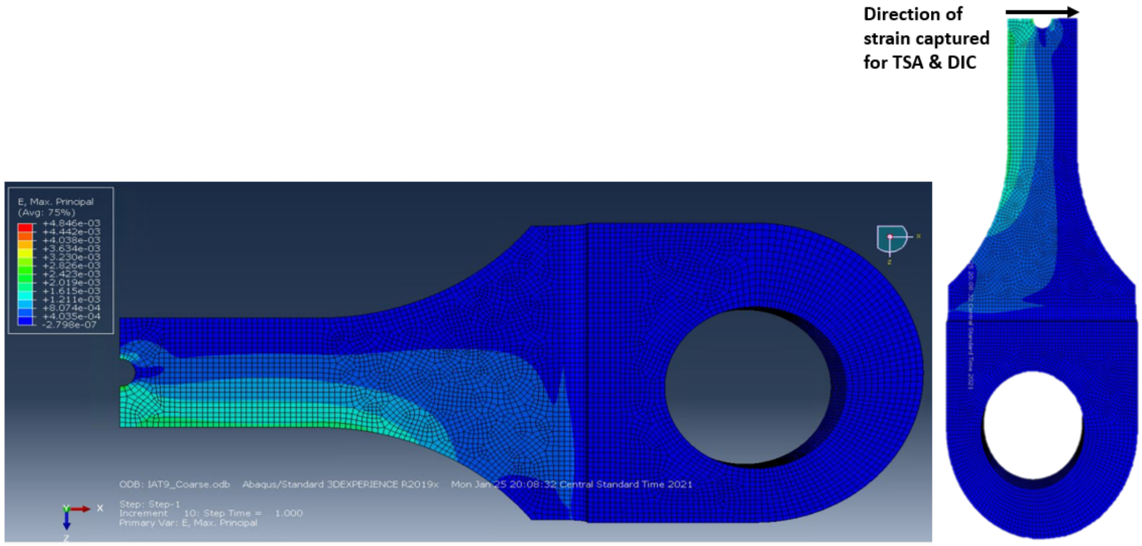


Figure 17. Side A Model Strain Map

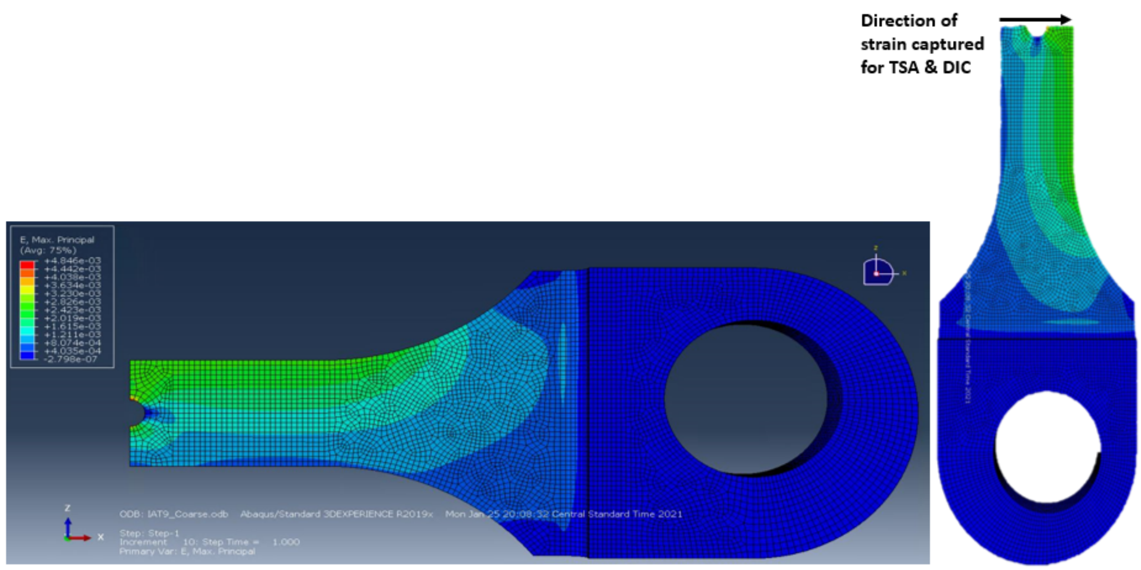


Figure 18. Side B Model Strain Map

4.6 Comparison Across Systems

To compare the results of the TSA system, the strain gages (both phases), and the finite element model, each system was scaled such that the values could be compared along the longitudinal centerline of the centered hole over a specified distance. This specified distance was determined by the width of the gage section of the specimen, and scaled to the appropriate pixel length in the TSA images and the appropriate nodal path in the model images. The length measurement utilized for Side A was 24.34mm, and the length measurement for Side B was 24.83mm. The strain response from all three systems, for Side A and Side B, individually, was plotted for comparison, and can be seen in Figure 19 and Figure 20.

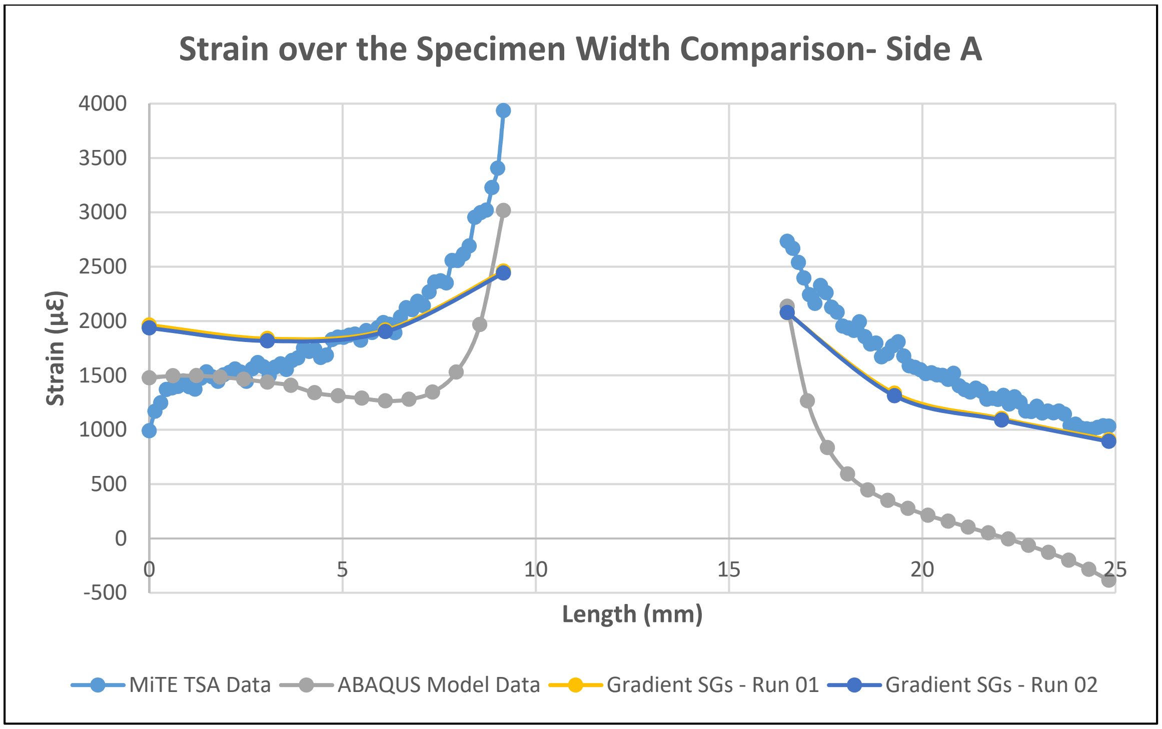


Figure 19. Strain Comparison Across 3 Data Sets for Side A

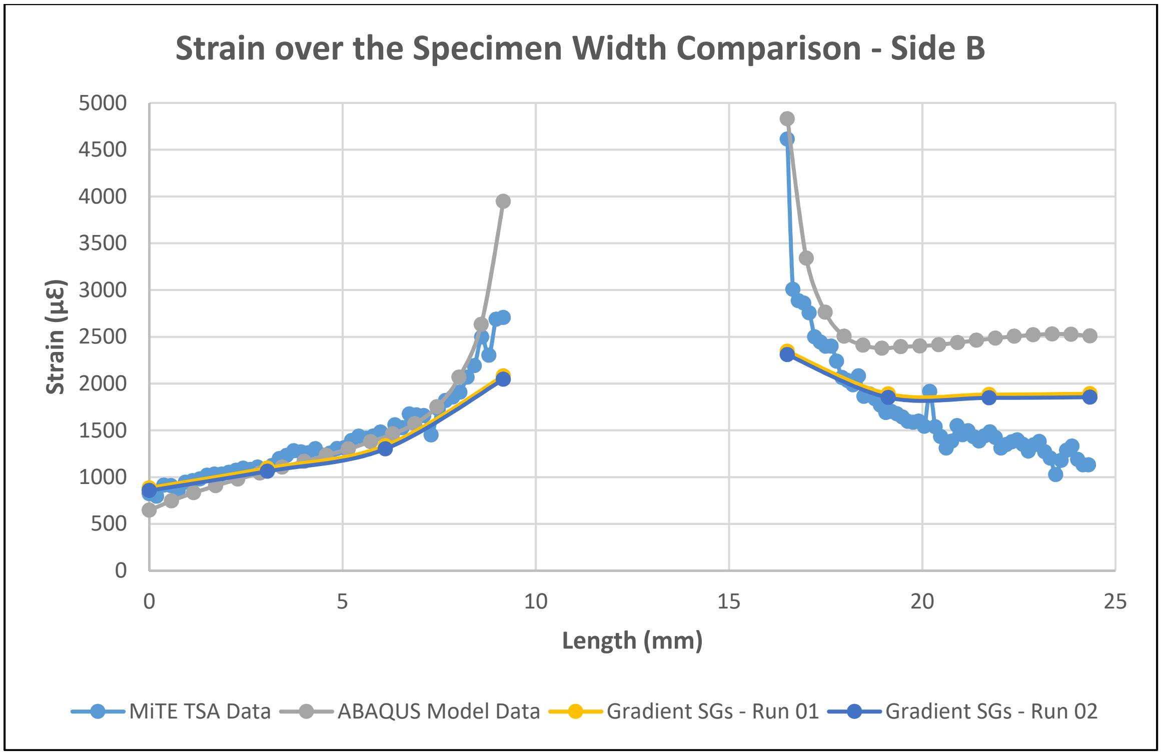


Figure 20. Strain Comparison Across 3 Data Sets for Side B

4.7 Image Quality Index Results

To further investigate the comparison from one method to another, the quality of the TSA images obtained from each test run were compared to one another utilizing a commercially available Matlab program [10] that utilizes various methods of comparing multiple aspects or indices of the image, including the correlation coefficient (CC), the quality (Q), the luminance distortion (LumDist), the contrast distortion (ConDist), and the root mean square error (RMSE). While this program is commercially available, the function for the quality (Q) index was modified to calculate the luminance and contrast distortions presented by Wang and Bovik [12]. To utilize this tool, the images of each of the test runs within the MiTEViewer system had to be saved as simply the image with no scaling or additional data. However, these needed to be images that were calibrated using the maximum strain value for that test run, which was calculated in the MTS data file and recorded in an Excel file for later use. Once these files were created, an image needed to be selected as the reference image. The reference image for each data series, Batch 01 being the comparison from the initial image in the test sequence to the remaining test sequences, or Batch 02 being the comparison from the initial image in the test sequence to the increasing load level cases, was the initial image captured for each side of the specimen, either Side A or Side B. More explicitly, the reference image for Side A is the 1200-lbf load case for the first test sequence, or 1200A. The reference image for Side B is the 1200-lbf load case for the first test sequence on Side B, or 1200B. These will be displayed separately as there are results for each of the color bands, as well as for the average of these bands in the following sections. While this Matlab program was able to calculate eight indices for comparison from image-to-image, only three of these were utilized in determining criteria for test-to-model correlation, as these were deemed to be the indices of higher value in determining this criteria and to narrow the scope of this endeavor. These indices include CC, Q (including the distortion components), and RMSE.

The data was processed in two data batches, however, the results were the same for both batches, so tables were created only for each set of image indices at the load levels for each test sequence. All of the values displayed in the initial tables are the average over three bands of the color spectrum, these bands being red (Band 1), green (Band 2), and blue (Band 3). As the thermal images produced have multiple colors to them, and are not simply greyscale or black and white, the program will analyze the image indices listed in the table over three color bands and produce four values, the first three being the individual index value per color band, and the fourth being the average over the three bands. Since there are three color bands, the values for each band are included in separate tables, after the average values.

For the first data batch, the initial data file, or the first 1200-lbf test run, in the set of three images at each load level was utilized as the reference image. This image was compared to all three images in the test sequences, for both Side A and Side B, and the results of these comparisons were recorded. To further compare the images, the 120 – 1200-lbf test run was utilized as the reference image for each side of the specimen and compared to each of the subsequent load level files for each test sequence for the corresponding side of the specimen. The results of both of these data batches are combined in Table 13 through Table 20, which includes the average values over each color band (1 through 3) and the individual values for each color band.

Table 13. Average Image Index Value for 1200-lbf Load Case for Side A & B over 3 bands

Image Index	**Image 1A**	**Image 2A**	**Image 3A**	**Image 1B**	**Image 2B**	**Image 3B**
CC	1.0000	0.8169	0.4861	1.0000	0.8202	0.2109
Q	1.0000	0.7998	0.4849	1.0000	0.7191	0.1356
LumDist	1.0000	0.9988	0.9998	1.0000	0.8884	0.7618
ConDist	1.0000	0.9821	0.9985	1.0000	0.9953	0.9788
RMSE	0.0000	17.8733	25.7477	0.0000	32.6979	60.1377

Table 14. Image Index Value for Band 1-3 of the 1200-lbf Load Case for Side A & B

Image Index	**Band Number**	**Image 1A**	**Image 2A**	**Image 3A**	**Image 1B**	**Image 2B**	**Image 3B**
CC	1	1	0.9266	0.8302	1.0000	0.8831	0.4564
	2	1	0.8063	0.08265	1.0000	0.7035	-0.04959
	3	1	0.7179	0.5455	1.0000	0.8742	0.2258
Q	1	1	0.8807	0.8283	1.0000	0.8108	0.3377
	2	1	0.8062	0.08264	1.0000	0.6967	-0.04864
	3	1	0.7124	0.5439	1.0000	0.6499	0.1177
LumDist	1	1	0.9991	0.9997	1.0000	0.9207	0.7398
	2	1	1.0000	1.0000	1.0000	1.0000	0.9998
	3	1	0.9972	0.9997	1.0000	0.7446	0.5457
ConDist	1	1	0.9513	0.9981	1.0000	0.9973	1.0000
	2	1	0.9999	0.99990	1.0000	0.9904	0.9811
	3	1	0.9951	0.9974	1.0000	0.9984	0.9552
RMSE	1	0	31.1878	40.9329	0.0000	71.5507	122.5543
	2	0	7.1975	15.5413	0.0000	7.4034	16.9156
	3	0	15.2346	20.7689	0.0000	19.1397	40.9432

Table 15. Average Image Index Value for 2200-lbf Load Case for Side A & B over 3 bands

Image Index	**Image 1A**	**Image 2A**	**Image 3A**	**Image 1B**	**Image 2B**	**Image 3B**
CC	0.8494	0.8102	0.4754	0.8003	0.6989	0.2244
Q	0.8398	0.8019	0.4548	0.6557	0.4770	0.1178
LumDist	0.9922	0.9952	0.9826	0.8516	0.7356	0.7112
ConDist	0.9966	0.9953	0.9841	0.9746	0.9621	0.9182
RMSE	16.7292	17.5838	28.8315	40.2881	58.2394	68.4412

Table 16. Image Index Value for Band 1-3 of the 2200-lbf Load Case for Side A & B

Image Index	**Band Number**	**Image 1A**	**Image 2A**	**Image 3A**	**Image 1B**	**Image 2B**	**Image 3B**
CC	1	0.9655	0.9457	0.8251	0.9180	0.8506	0.4322
	2	0.7664	0.7104	0.0717	0.6505	0.4992	-0.0628
	3	0.8164	0.7746	0.5295	0.8322	0.7469	0.3037
Q	1	0.9558	0.9290	0.7790	0.7716	0.6105	0.3055
	2	0.7645	0.7089	0.0706	0.6481	0.4899	-0.05859
	3	0.7990	0.7679	0.5150	0.5474	0.3306	0.1065
LumDist	1	0.9900	0.9938	0.9746	0.8729	0.7177	0.7070
	2	1.0000	1.0000	1.0000	1.0000	0.9995	0.9990
	3	0.9865	0.9918	0.9733	0.6819	0.4894	0.4274
ConDist	1	1.0000	0.9885	0.9687	0.9628	1.0000	0.9997
	2	0.9976	0.9978	0.9845	0.9963	0.9817	0.9345
	3	0.9922	0.9995	0.9992	0.9646	0.9045	0.8204
RMSE	1	27.2697	28.8158	50.7389	87.3806	118.9813	128.0200
	2	8.2489	9.1335	14.5438	9.0358	12.9620	20.4563
	3	14.6689	14.8020	21.2119	24.4479	42.7749	56.8472

Table 17. Average Image Index Value for 3200-lbf Load Case for Side A & B over 3 bands

Image Index	**Image 1A**	**Image 2A**	**Image 3A**	**Image 1B**	**Image 2B**	**Image 3B**
CC	0.8125	0.7927	0.5092	0.8461	0.7942	0.2272
Q	0.8070	0.7865	0.4628	0.8258	0.7361	0.1647
LumDist	0.9999	0.9977	0.9481	0.9910	0.9445	0.8201
ConDist	0.9926	0.9940	0.9744	0.9826	0.9847	0.9582
RMSE	14.9409	16.6133	34.7759	12.1975	22.6593	54.1583

Table 18. Image Index Value for Band 1-3 of the 3200-lbf Load Case for Side A & B

Image Index	**Band Number**	**Image 1A**	**Image 2A**	**Image 3A**	**Image 1B**	**Image 2B**	**Image 3B**
CC	1	0.9611	0.9515	0.8209	0.9535	0.9255	0.5023
	2	0.6639	0.6261	0.0955	0.6992	0.5889	-0.0397
	3	0.8125	0.8006	0.6112	0.8857	0.8683	0.2190
Q	1	0.9610	0.9475	0.7761	0.9501	0.8996	0.4017
	2	0.6555	0.6194	0.0934	0.6642	0.5655	-0.0383
	3	0.8046	0.7925	0.5189	0.8631	0.7430	0.1308
LumDist	1	0.9999	0.9973	0.9522	0.9975	0.9737	0.8288
	2	1.0000	1.0000	0.9998	1.0000	1.0000	0.9999
	3	0.9998	0.9957	0.8921	0.9757	0.8597	0.6315
ConDist	1	0.9999	0.9985	0.9928	0.9990	0.9984	0.9648
	2	0.9874	0.9893	0.9786	0.9499	0.9602	0.9643
	3	0.9905	0.9941	0.9516	0.9988	0.9954	0.9455
RMSE	1	20.2627	24.5030	57.6359	20.3662	45.8047	109.6575
	2	10.4576	10.9175	17.8948	7.3915	8.3454	17.2273
	3	14.1025	14.4192	28.7969	8.8347	13.8279	35.5901

Table 19. Average Image Index Value for 4200-lbf Load Case for Side A & B over 3 bands

Image Index	**Image 1A**	**Image 2A**	**Image 3A**	**Image 1B**	**Image 2B**	**Image 3B**
CC	0.7834	0.7726	0.5100	0.7757	0.7628	0.2261
Q	0.7747	0.7641	0.4843	0.7312	0.7467	0.1688
LumDist	0.9999	0.9999	0.9785	0.9680	0.9882	0.8372
ConDist	0.9876	0.9876	0.9716	0.9779	0.9887	0.9535
RMSE	16.0338	16.3320	31.0077	16.4955	14.3759	52.2938

Table 20. Image Index Value for Band 1-3 of the 4200-lbf Load Case for Side A & B

Image Index	**Band Number**	**Image 1A**	**Image 2A**	**Image 3A**	**Image 1B**	**Image 2B**	**Image 3B**
CC	1	0.9591	0.9566	0.8384	0.9241	0.9330	0.5014
	2	0.5774	0.5495	0.0965	0.6202	0.5358	-0.0372
	3	0.8135	0.8117	0.5951	0.7827	0.8197	0.2141
Q	1	0.9579	0.9557	0.8230	0.8724	0.9277	0.4094
	2	0.5660	0.5388	0.0936	0.6189	0.5189	-0.0358
	3	0.8000	0.7979	0.5362	0.7023	0.7935	0.1327
LumDist	1	0.9998	0.9999	0.9837	0.9955	0.9966	0.8541
	2	1.0000	1.0000	0.9999	1.0000	1.0000	0.9999
	3	0.9998	0.9997	0.9518	0.9086	0.9681	0.6577
ConDist	1	0.9989	0.9991	0.9980	0.9483	0.9977	0.9559
	2	0.9803	0.9806	0.9702	0.9978	0.9684	0.9624
	3	0.9835	0.9832	0.9466	0.9874	1.0000	0.9421
RMSE	1	21.4414	21.8760	49.0685	28.6807	23.3605	105.2431
	2	12.0221	12.3854	18.0463	9.3358	9.0381	17.2263
	3	14.6378	14.7346	25.9084	11.4700	10.7292	34.4119

CHAPTER V

DISCUSSION

5.1 Discussion Overview

Throughout this experimental work, the strain results of the tensile testing of a dogbone specimen have been captured utilizing various methods, including various strain gages for a contact measurement system, and an MiTE TSA system for non-contact measurement. The data for this experimental work was captured in two phases, one for the initial strain gages and the TSA system, and the second for the additional strain sensors and strain gradient data capture. The results from each of these test phases were then compared in a traditional manner, plotting them in Microsoft Excel and comparing between the model, the strain gage data, and the TSA data. To take this a step further, a Matlab program designed to analyze multispectral images utilizing eight image indices was employed to compare the data between each test run and determine how well this data compared to the model data. From this comparison, criteria for model validation was suggested based upon the value returned from the Image Quality Index excursion.

5.2 Strain Gage to Model Correlation

As stated previously, the initial phase of instrumentation of this dogbone specimen included two different types of thin-film/foil strain gages, including two CEA-05-125UW-350 gages placed on the front and back face of the dogbone, and two WK-05-062AP-350 strain gages (of smaller footprint) on the through-thickness faces of the specimen. The response of these gages were compared across each of the three test sequences, as well as from Side A to Side B. The results of these comparisons can be seen in Section 4.2 and 4.3. The comparison from test run-to-test run for the centerline, or Phase II, gages demonstrates the repeatability of the strain data achieved by the strain gages, and the comparison from the strain gages to the additional systems appears to show good alignment, and a good ability to correlate between the larger gages, the TSA system, and the model.

Additionally, the strain gage values were compared to the model values utilizing the correlation criteria discussed in Section 2.4. The graphs of this comparison are shown in Figure 21 and Figure 22.

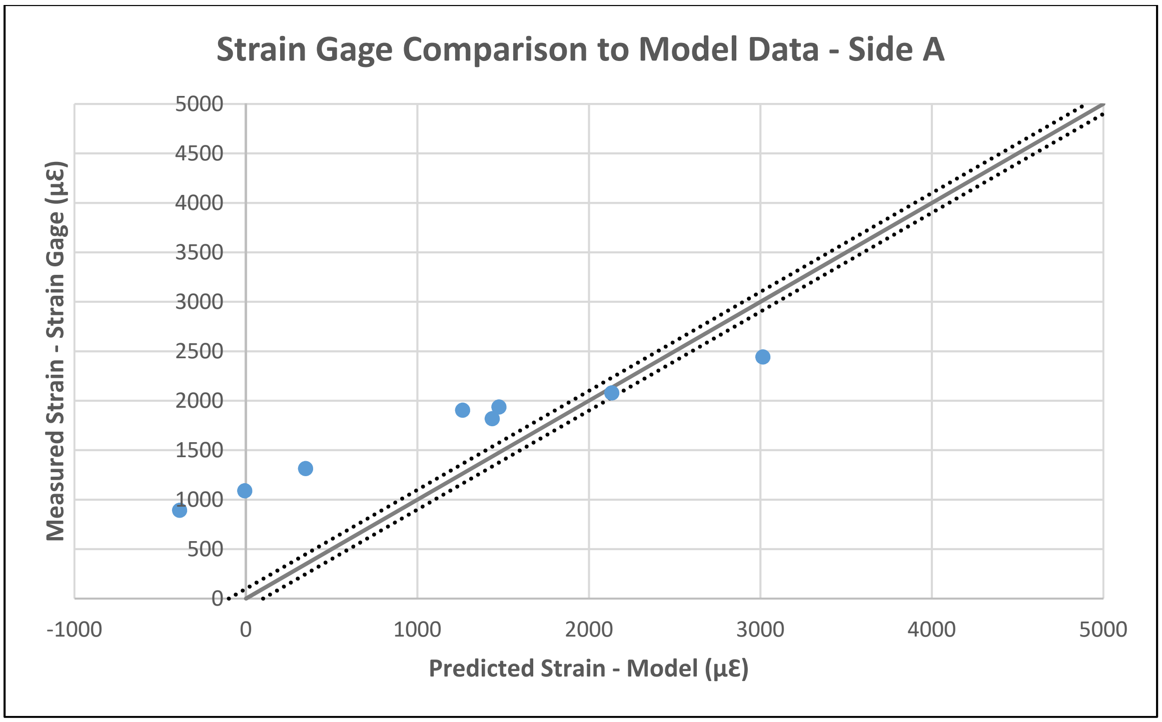


Figure 21. Strain Gage Comparison to Model Data for Side A

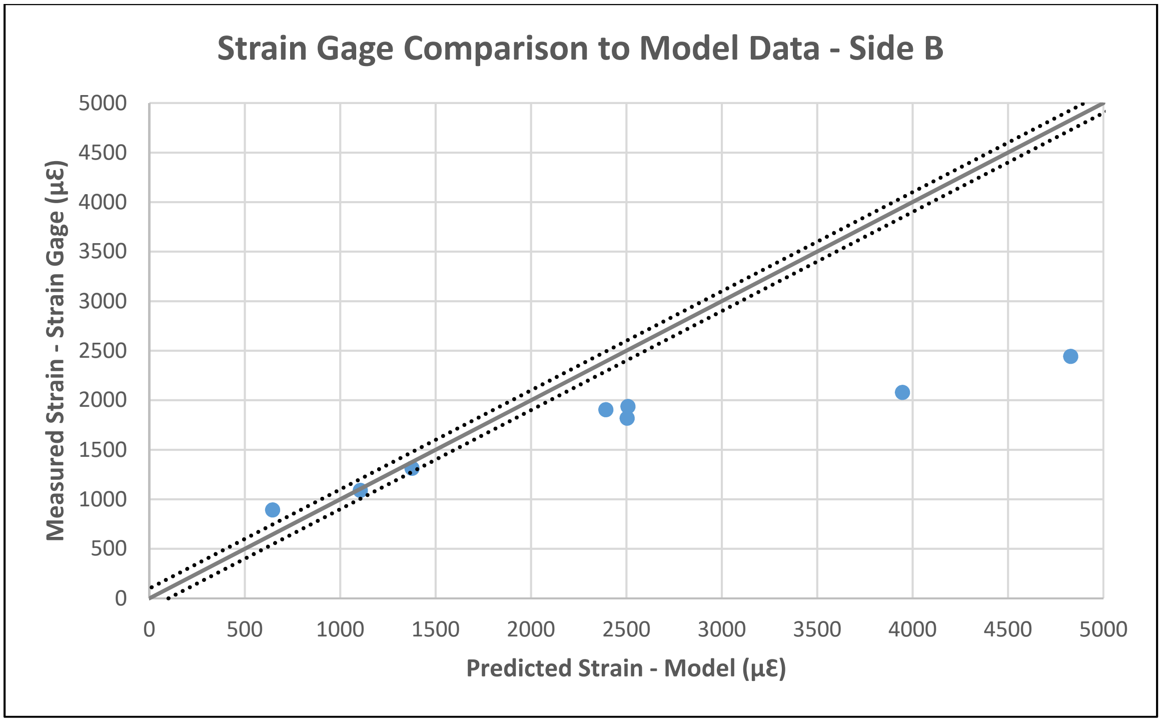


Figure 22. Strain Gage Comparison to Model Data for Side B

The strain gage values vs. the model values were plotted, along with a perfect correlation line (the solid line) and the dashed lines are the +/-10% error lines. As can be seen from this graph, the majority of the values do not match up well on Side A; however, Side B has more data points that fall within or very close to the error lines. While this is not a perfect correlation, this does speak to the differences in the data captured on each side, and how this has been consistently different. The linearity of the values for Side A, however, are quite a bit more defined when compared to Side B. This could possibly indicate an error in the model itself, as the values should align with the model fairly well.

5.3 TSA to Model Correlation

The TSA image data for the 4200-lbf test run was compared to the model data at the same load, and the results were compared to one another in correlation graphs, as seen in Figure 23 and Figure 24.

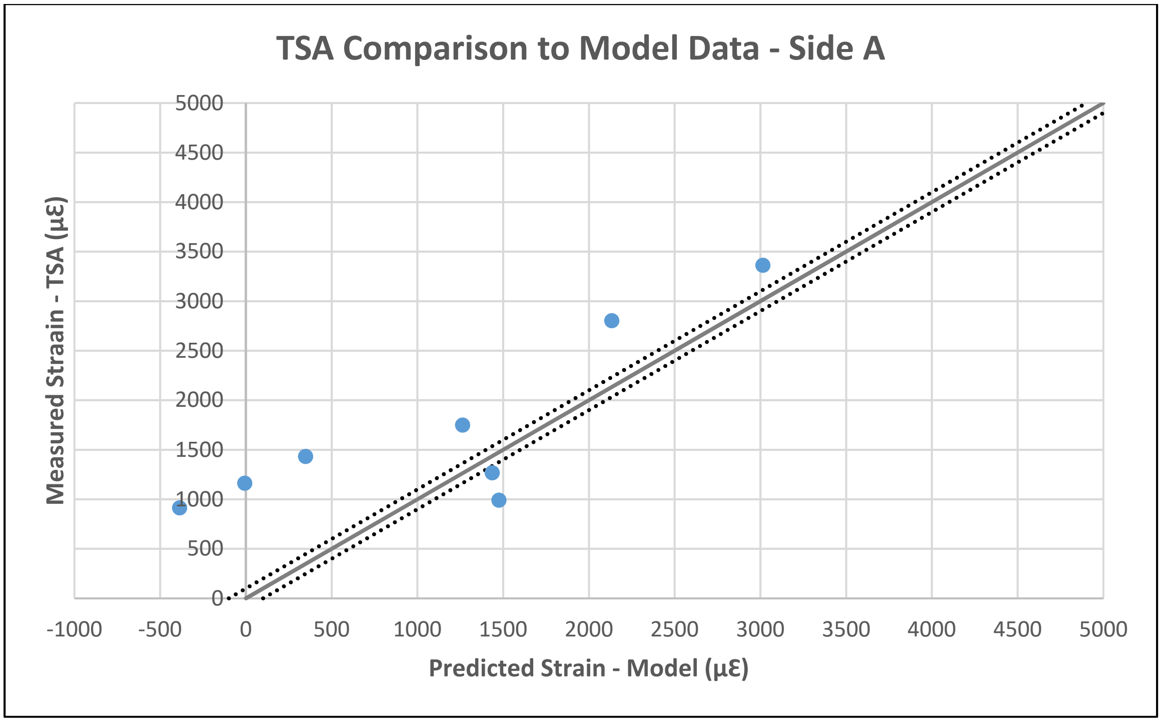


Figure 23. TSA Comparison to Model Data for Side A

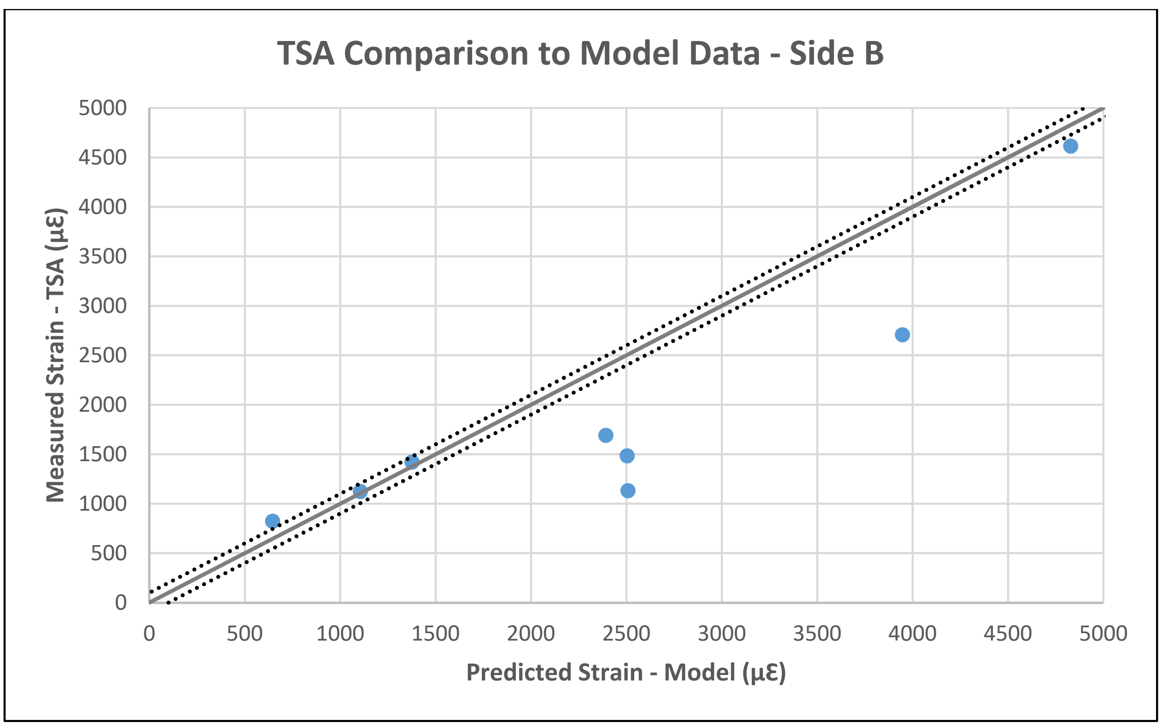


Figure 24. TSA Comparison to Model Data for Side B

The model does not appear to correlate well to either side of the specimen, however, there is clearly better correlation with Side B compared to Side A. The image index values at these points will need to be investigated to determine if they bolster this comparison, or if they appear skewed. The comparison for Side A, however outside the error bounds, does appear to follow a more linear trend, which would speak to better correlation. This is due to the fact that these values should be increasing in a linear trend, as they did for the strain gage comparison when comparing the linearity as the strains increased in conjunction with the increasing load.

5.4 TSA to Strain Gage Correlation

As another means of correlation, the strain gage data was compared to the TSA data to determine how well they correlated. This was done due to the fact that the model data did not appear to correlate well to either the strain gage data, or the TSA data, as the majority of the values were outside of the +/- 10% error lines. The resulting graphs can be seen in Figure 25 and Figure 26.

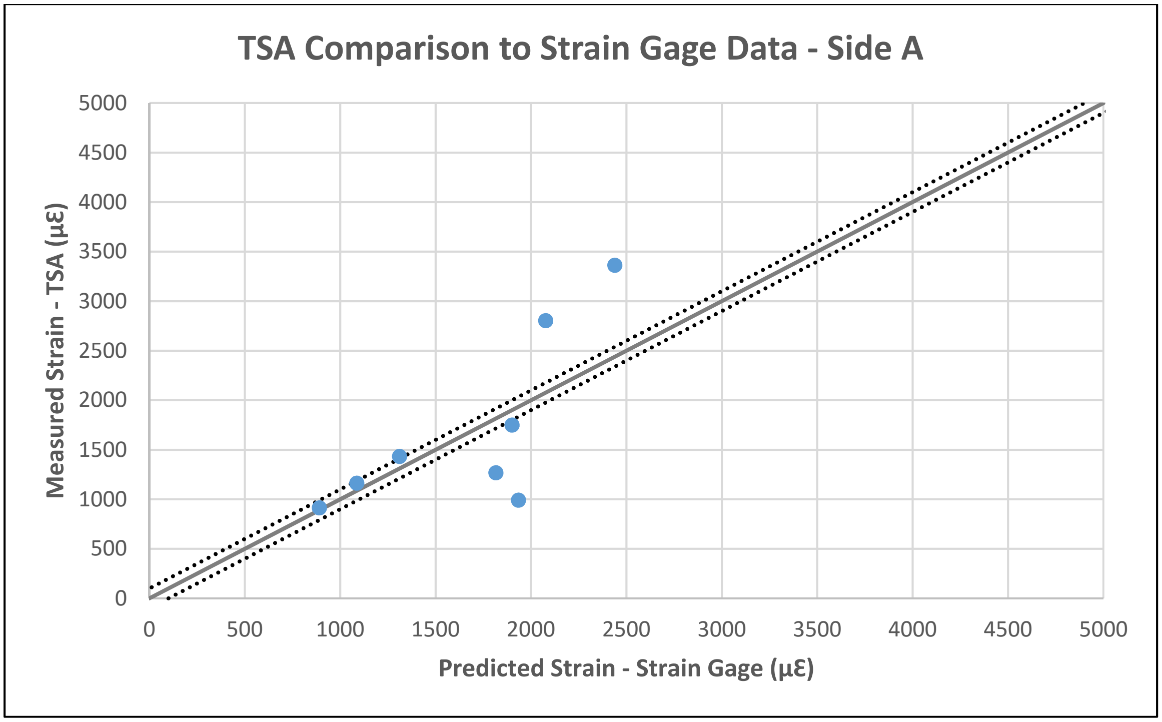


Figure 25. TSA Comparison to Strain Gage Data for Side A

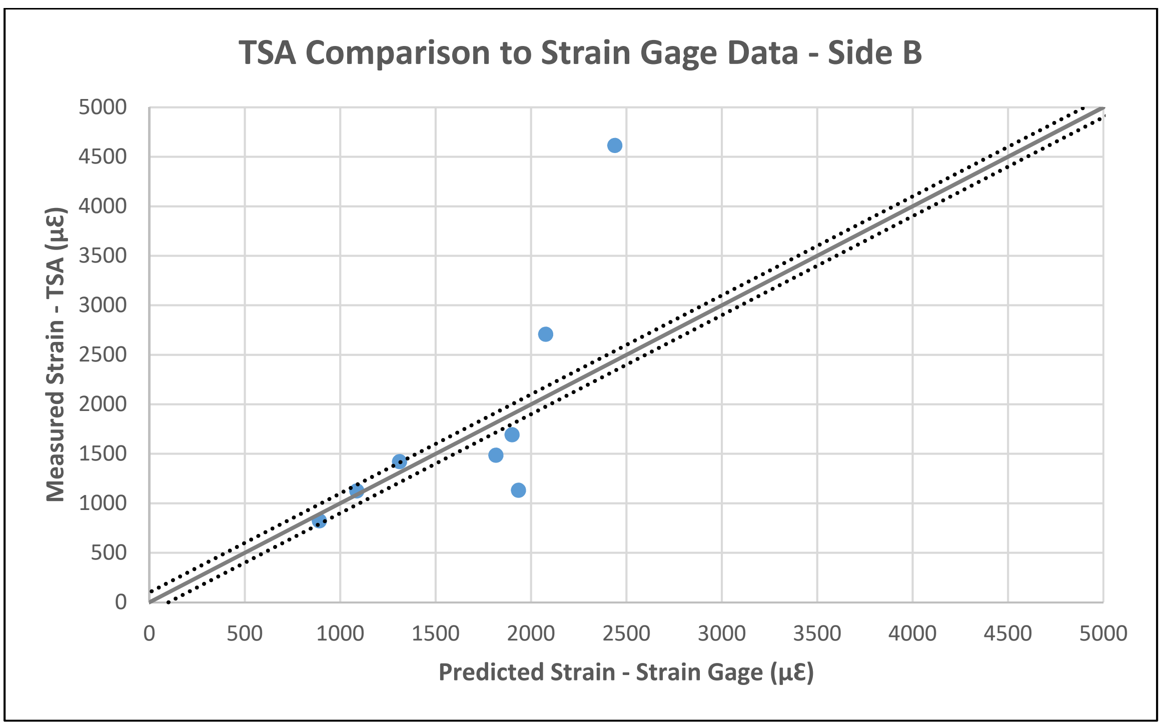


Figure 26. TSA Comparison to Strain Gage Data for Side B

The TSA data appears to correlate better to the strain gage data, with more data points falling within the +/-10% error lines. Additionally, the data appears to be clustered much closer to the correlation line and not spread out quite as much when compared to the model. The data from Side A to Side B appears to be nearly identical, with only two points that appear to differ, which speaks to the repeatability of both the TSA system and the strain gages. This would then indicate that there is an issue within the model.

5.5 Image Quality Analysis Discussion

For the image index results, the images were captured in two separate batches. The first batch compared the 1200-lbf image to each successive load level increase (1200 – 1200-lbf, 1200 – 2200-lbf, 1200 – 3200-lbf, and 1200 – 4200-lbf) and produced four sets of data for each side of the specimen, for a total of eight data sets. Each of these data sets were compared as a set to determine trends in the indices as the load levels changed, as these have a predicted pattern and can be assessed based on that as to whether the trends agree with the strain gage and model data, or if there appear to be anomalies in the image-to-image trends.

The second batch analyzed the 1200-lbf image of the first test sequence in comparison to the three test sequences, comparing the results at the same load level for each set of data, creating four sets of load level data comparisons. The first image was compared to itself to provide a baseline. This was done to provide a trend in the image indices from the first to the third test sequence.

While the Matlab program was created to determine eight image quality indices, for the purposes of this endeavor, only three were utilized for determining correlation criteria for the data captured utilizing the TSA system. The three image indices used for this work are the correlation coefficient (CC), the quality (Q), and the root mean square error (RMSE). These were determined to be the most applicable, and therefore, the remaining five were not included in the discussion of the results.

5.5.1 Batch 01 – Load Level Comparisons

The image indices produced for the initial batch of comparisons were split into groups of three and categorized by the test sequence and side of the specimen, for example 1A – 1200 corresponds to the first test run for Side A at the 120 – 1200lbf load level. This was done for each set of images until twelve data sets were created for each side of the specimen, to include data for the three bands (red, green, and blue) and the average over these three spectral bands. The data across each spectral band was compared to one another to determine trends in the data at these color spectra. The graphs of the comparison of each load level for each color band can be seen in the figures in the following subsections. These will be broken down according to image index to provide a visual comparison across the different color bands, with both Side A and Side B being included in the same graph.

5.5.1.1 Batch 01 – Correlation Coefficient (CC) Index

The correlation coefficient is a measure of the correlation between the strain in the reference image and the strain in each of the test images. This is a value that ranges from negative one to positive one, with a positive one being a perfect positive correlation, zero being no correlation, and negative one being a perfectly negative correlation. The closer to one, the better the correlation between the two values. A series of comparisons over the load cases, for both Side A and B, can be seen in Figure 27 through Figure 29.

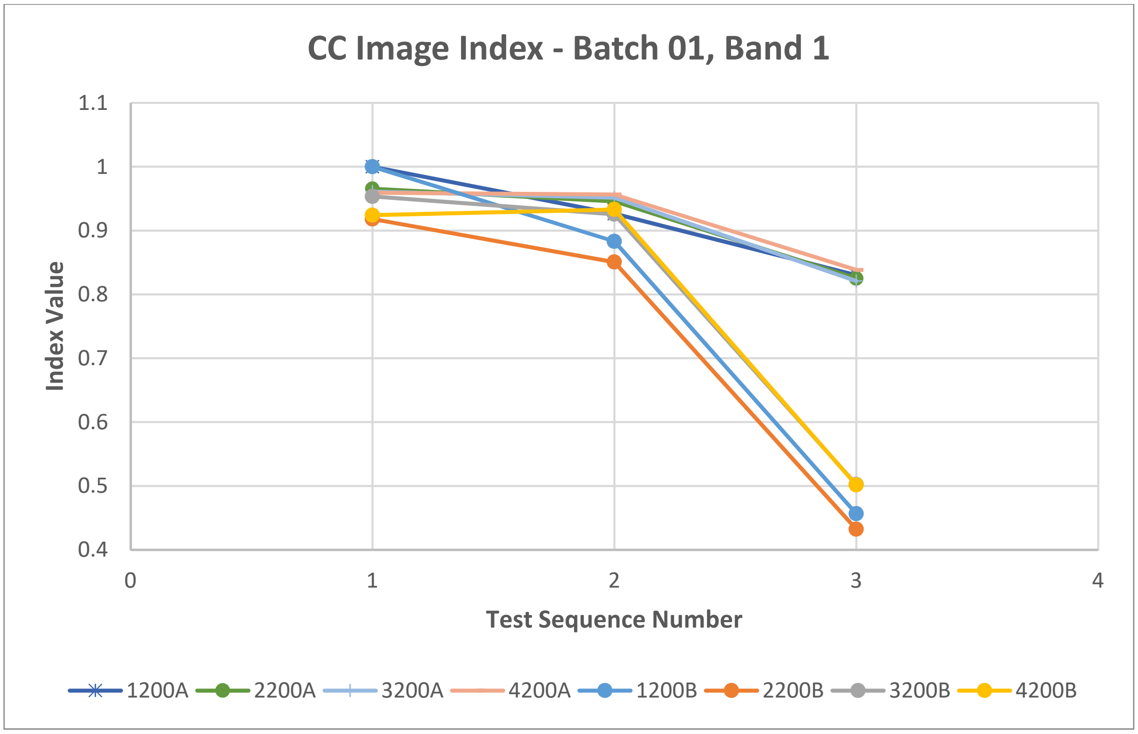


Figure 27. Correlation Coefficient Image Index for Band 1, Side A & B per Test Sequence

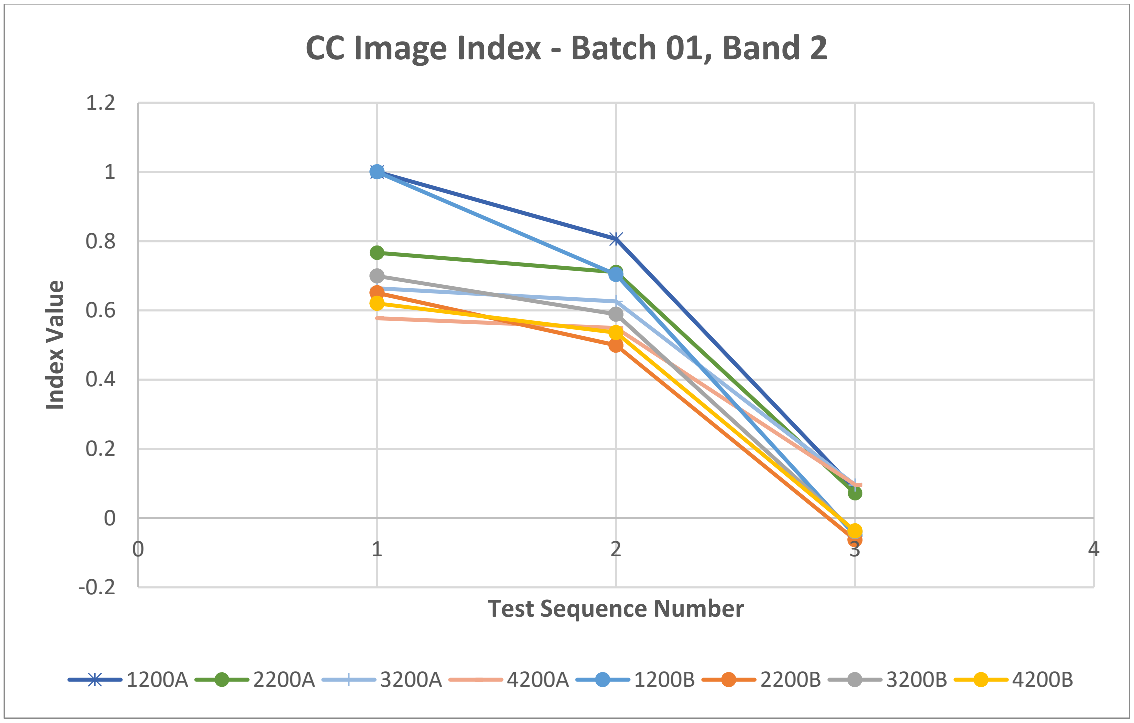


Figure 28. Correlation Coefficient Image Index for Band 2, Side A & B per Test Sequence

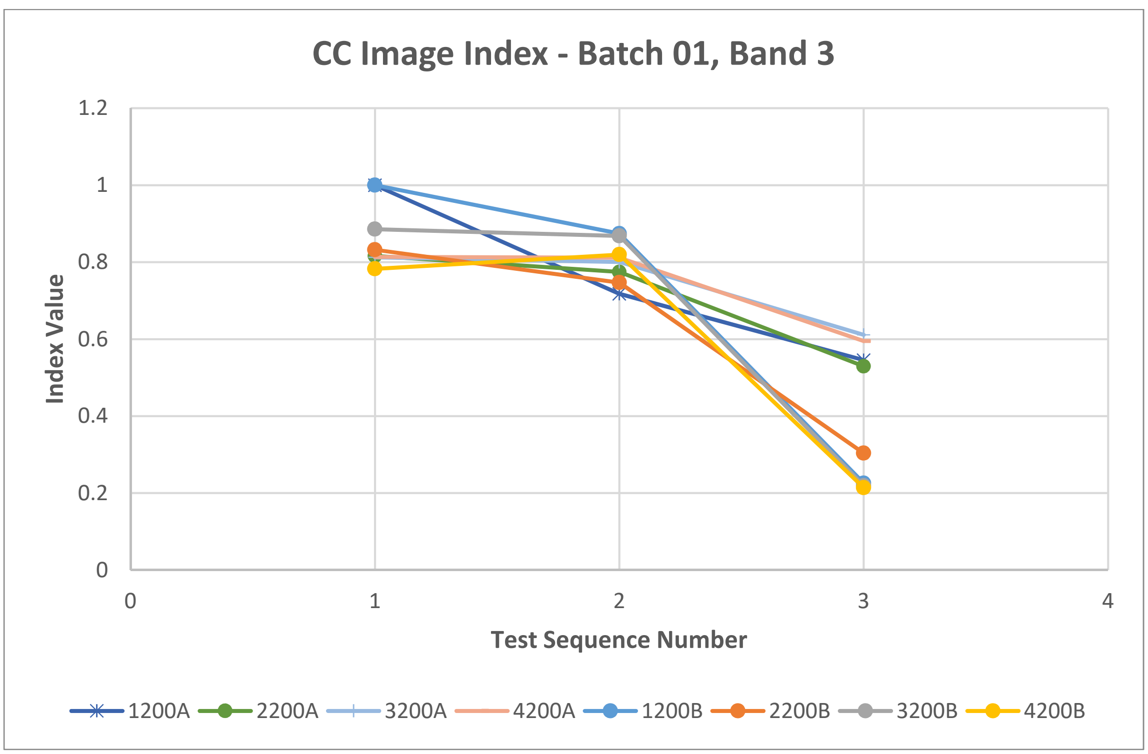


Figure 29. Correlation Coefficient Image Index for Band 3, Side A & B per Test Sequence

The trend for all of the test runs appears to be a slight decrease in correlation value from test sequence one to test sequence two, and a steeper decline in correlation value to test sequence three. As this is a component of the quality index, this would indicate that the quality of the images is declining over the test sequences. For the red band, or Band 1, the correlation values remain quite high for Side A, however, they appear to decrease significantly to 0.4322. This still indicates a moderate agreement, but calls into question the quality of the images from Side A to Side B for test sequence three. Additionally, Side A appears to have the same trend as Side B, however, the correlation values appear to remain more positive in test sequence three.

5.5.1.2 Batch 01 – Image Quality (Q) Index

The quality image index is a measure of the quality of the image, and is a product of the correlation coefficient, and two distortion factors, luminance and contrast. A series of comparisons over the load cases, for both Side A and B, can be seen in Figure 30 through Figure 32.

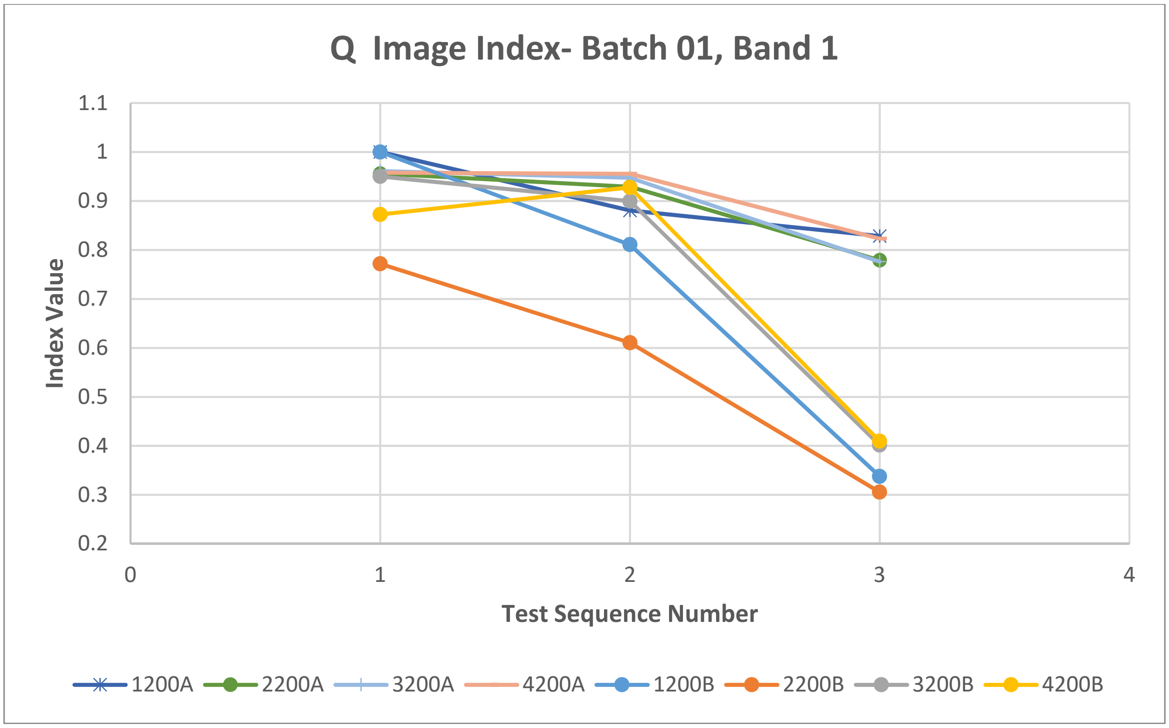


Figure 30. Quality Image Index for Band 1, Side A & B per Test Sequence

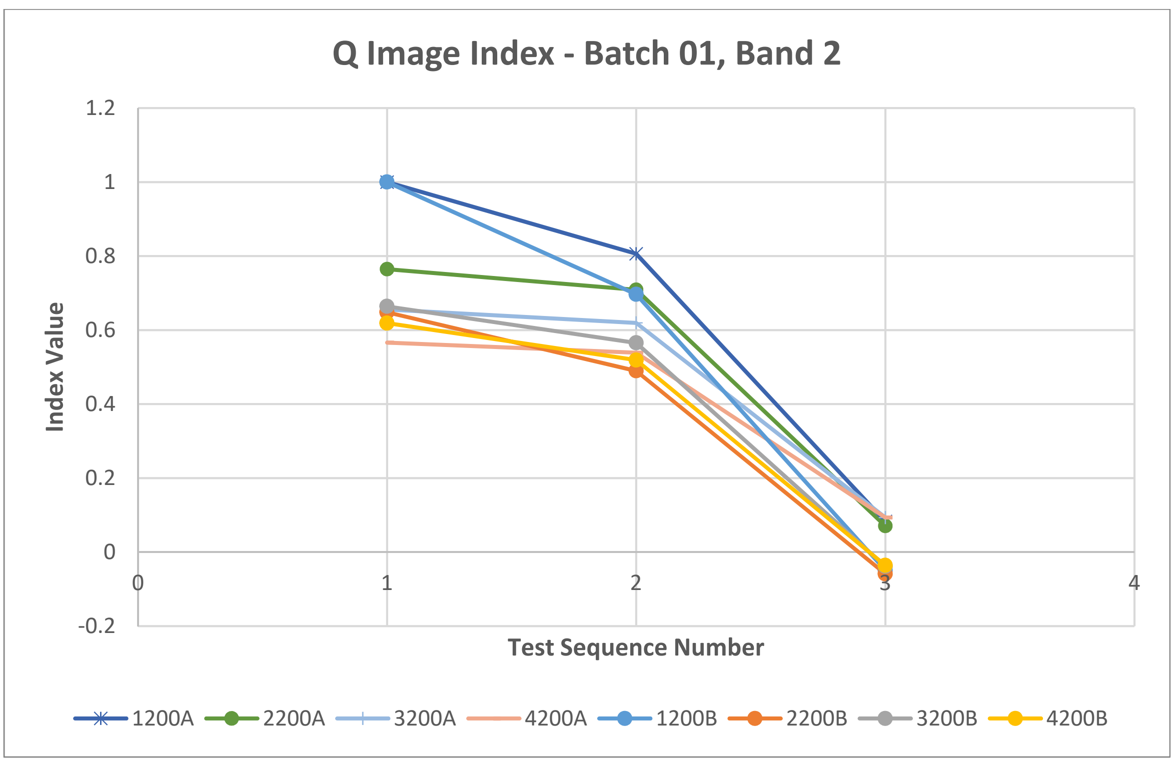


Figure 31. Quality Image Index for Band 2, Side A & B per Test Sequence

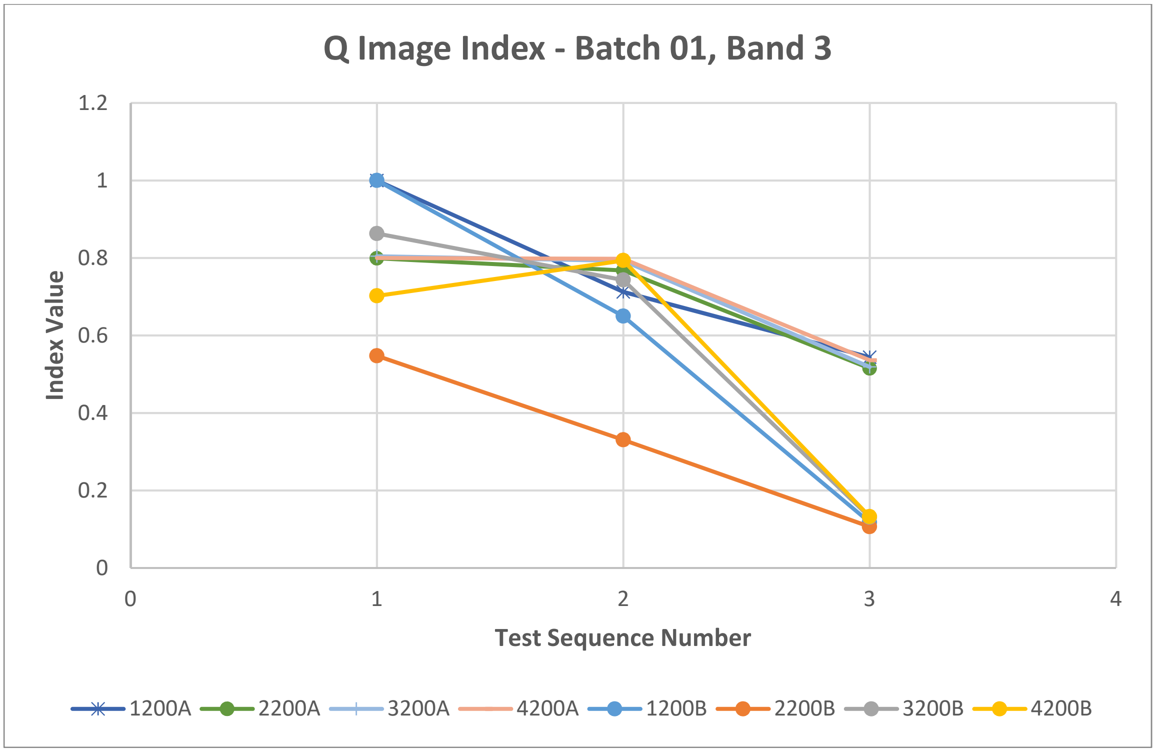


Figure 32. Quality Image Index for Band 3, Side A & B per Test Sequence

The quality of the images produced for this endeavor appear to be significantly higher for Side A compared to Side B across the color bands. However, much like the correlation coefficient, the quality values appear to slightly decrease from the first test sequence to the second, with a much more significant decrease in the third sequence. This steeper decline is particularly present in the green and blue bands, or Band 2 and Band 3. These are the bands with the greatest amount of fluctuation in the strain values, therefore, they have the highest likelihood of error or variance in values.

5.5.1.3 Batch 01 – Luminance Distortion (LumDist) Index

The luminance distortion image index is a measure of the "mean luminance between the factors x and y" [12], or the strain in the reference image and the strain in the test image. A series of comparisons over the load cases, for both Side A and B, can be seen in Figure 33 through Figure 35.

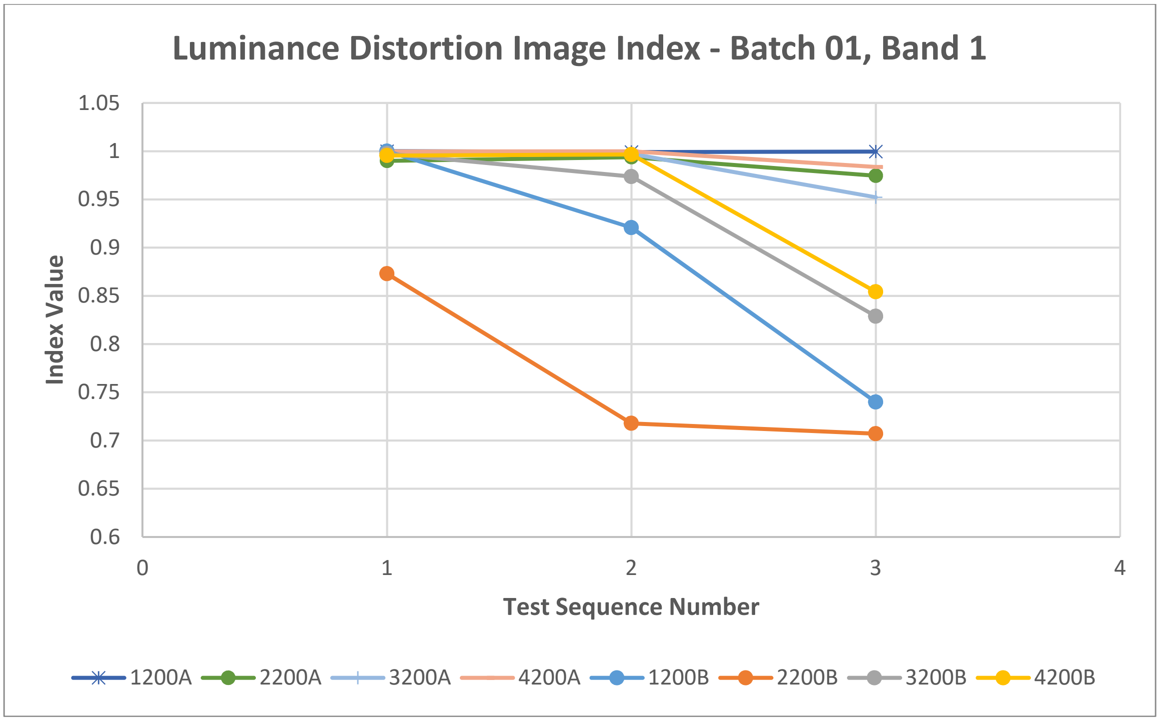


Figure 33. Luminance Distortion Image Index for Band 1, Side A & B per Test Sequence

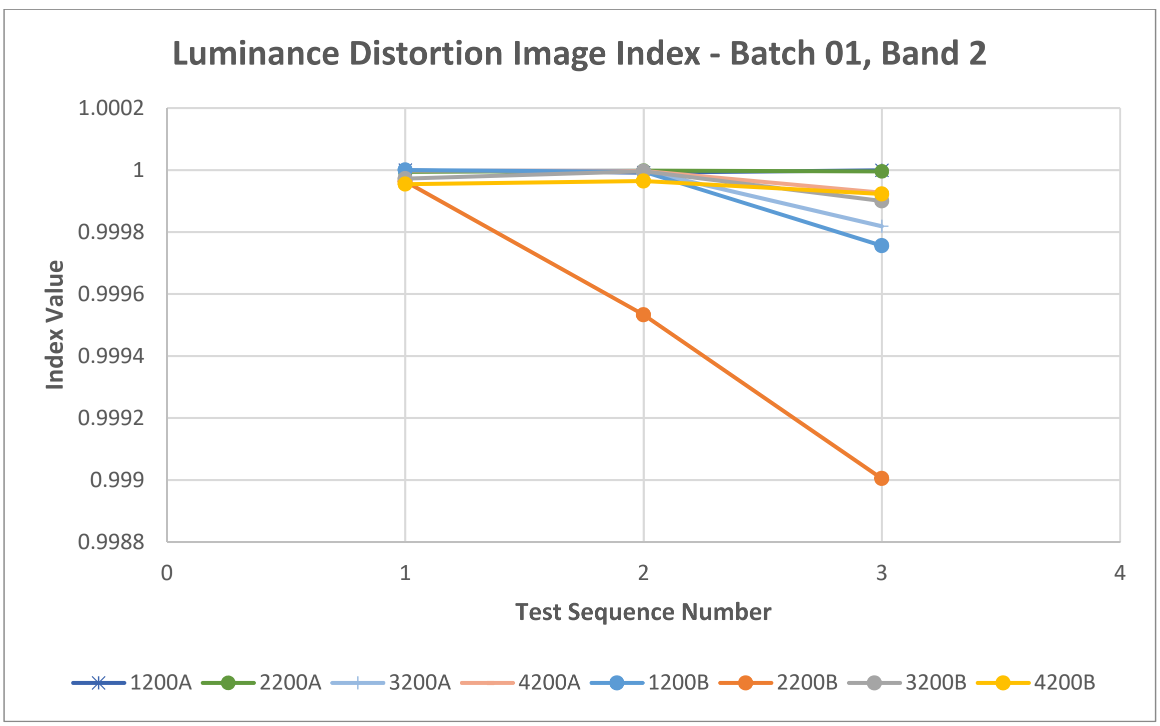


Figure 34. Luminance Distortion Image Index for Band 2, Side A & B per Test Sequence

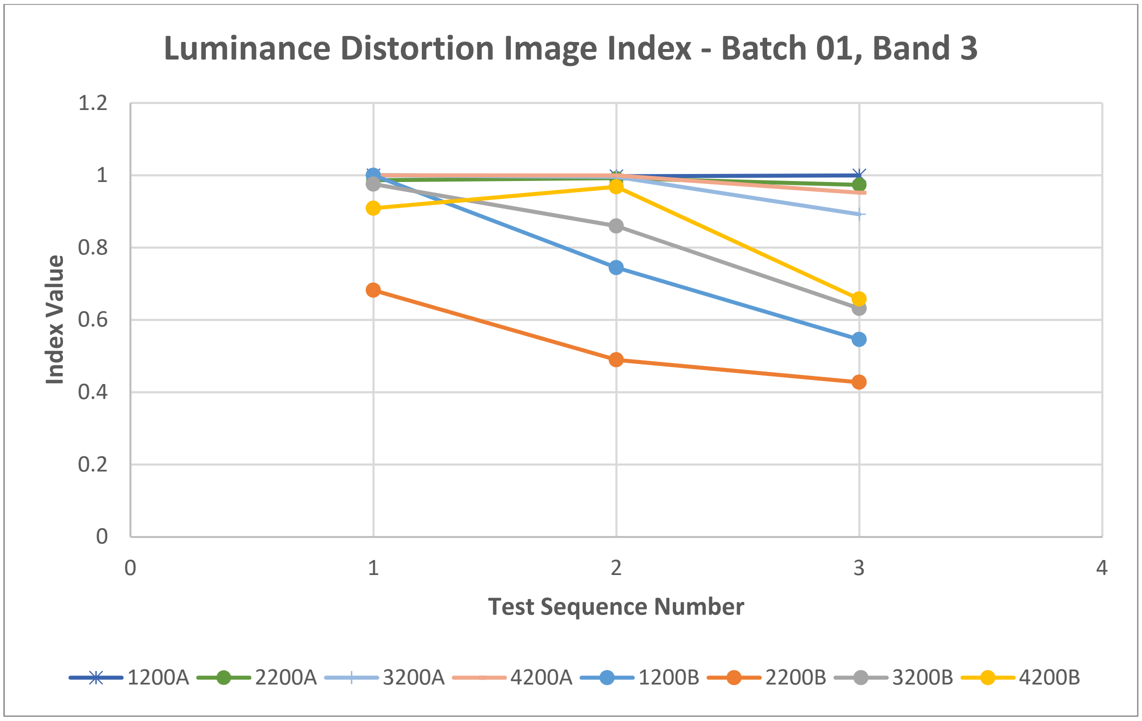


Figure 35. Luminance Distortion Image Index for Band 3, Side A & B per Test Sequence

When comparing the trends from Side A to Side B, the values for Side A are consistently higher and appear to remain consistent for test sequence one and two, however, they slightly diverge at the third test sequence. For Side B, the first, third, and fourth load cases appear to follow a consistent trend, however, the 2200-lbf load case is a consistently lower value. This appears to be an outlier in the data and would require further analysis of the images to determine the cause of the deviation.

5.5.1.4 Batch 01 – Contrast Distortion (ConDist) Index

The contrast distortion image index is a measure of the degree to which the contrast in the reference image and the test image are similar, or the same [12]. A series of comparisons over the load cases, for both Side A and B, can be seen in Figure 36 through Figure 38.

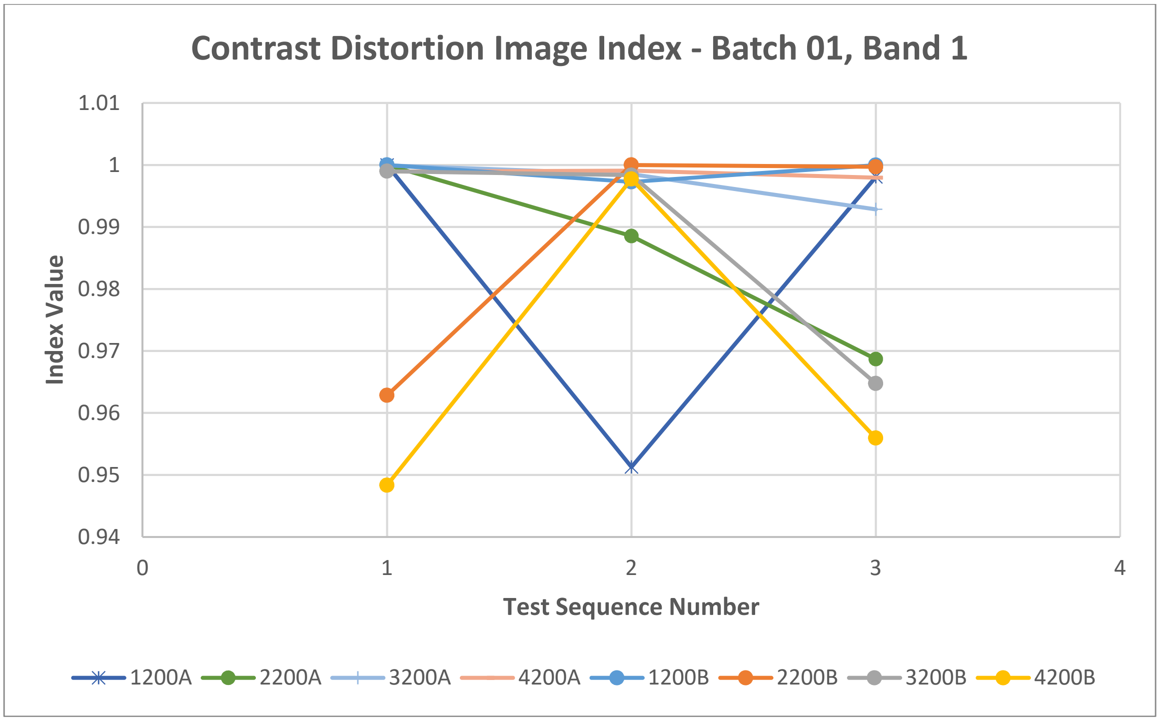


Figure 36. Contrast Distortion Image Index for Band 1, Side A & B per Test Sequence

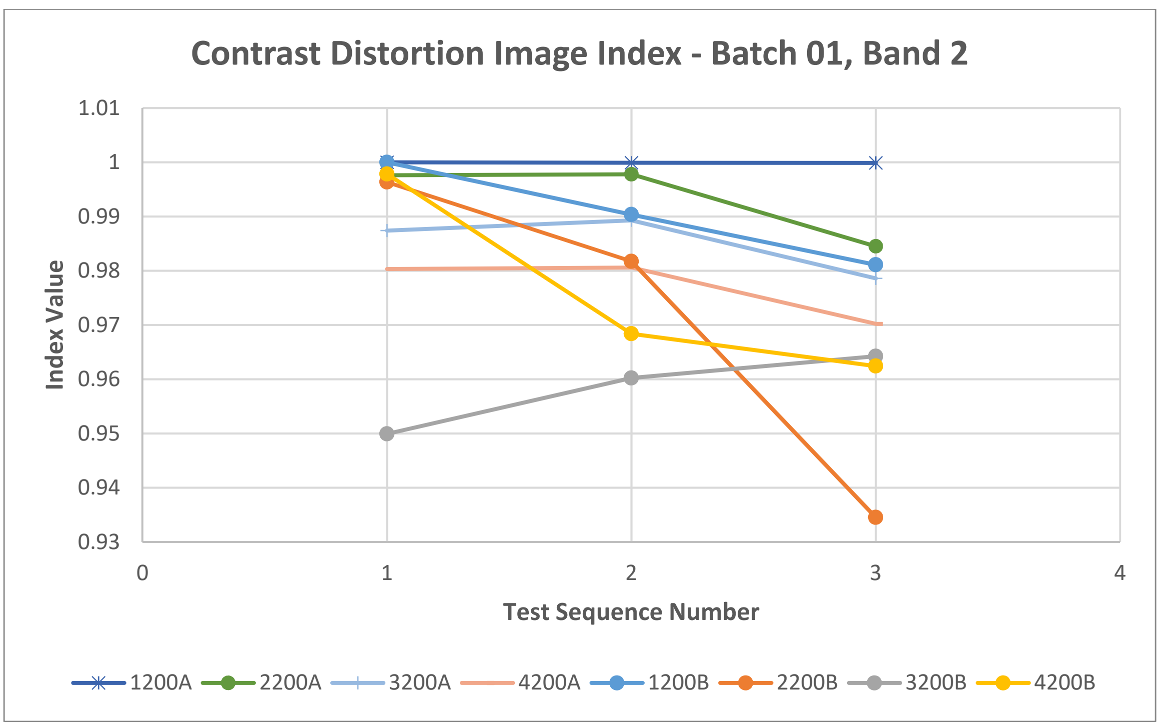


Figure 37. Contrast Distortion Image Index for Band 2, Side A & B per Test Sequence

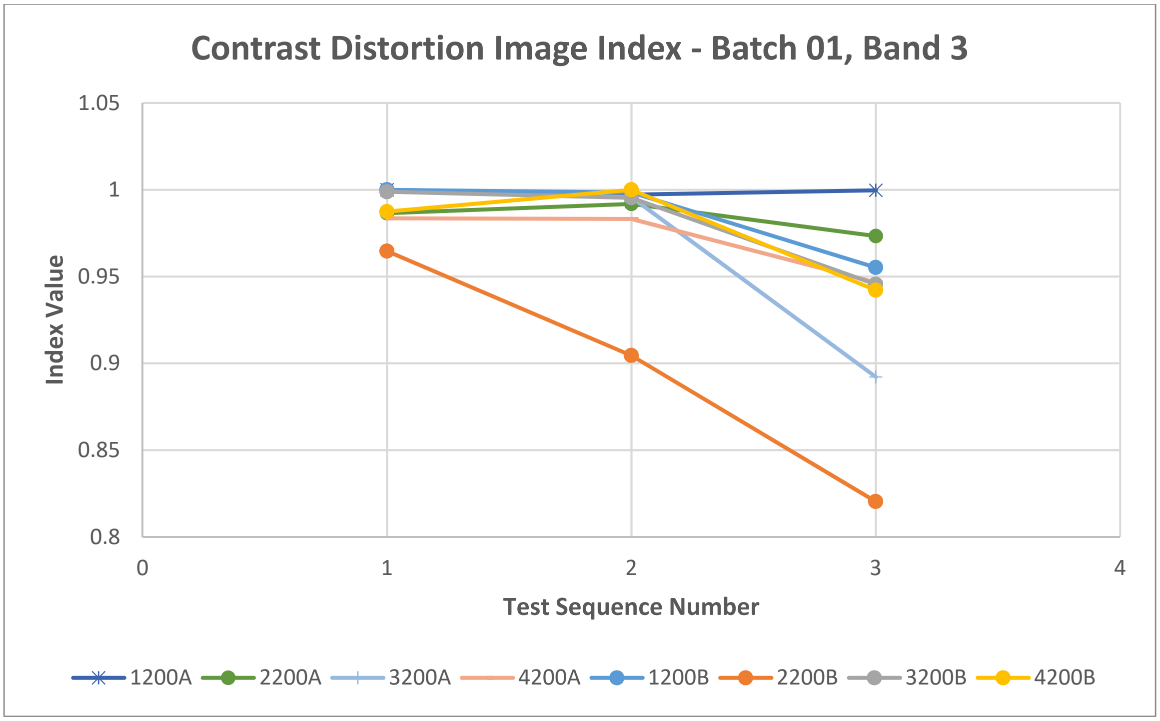


Figure 38. Contrast Distortion Image Index for Band 3, Side A & B per Test Sequence

For Side A, the 1200-lbf load cases appears to be the outlier in the red band, however it remains consistent for the green and blue band at a value of one. The values for Side A seem to remain consistent until the third test sequence, where they diverge at the 2200-lbf load case. For Side B, there appears to be a divergence in trends between the 1200-lbf to 2200-lbf load cases and the 3200-lbf to 4200-lbf load cases. However, the values are consistently above 0.85 for all of the test sequences and load levels, with the exception of 2200B in the Band 3 graph in Figure 38. The blue band appears to have the least amount of variability in the test sequence trends, however, the scaling is slightly different due to the 2200B outlier.

5.5.1.5 Batch 01 – Root Mean Square Error (RMSE) Index

The RMSE image index is a measure of the deviation from the linear regression line and can have a direct relationship with the correlation coefficient. A series of comparisons over the load cases, for both Side A and B, can be seen in Figure 39 through Figure 41.

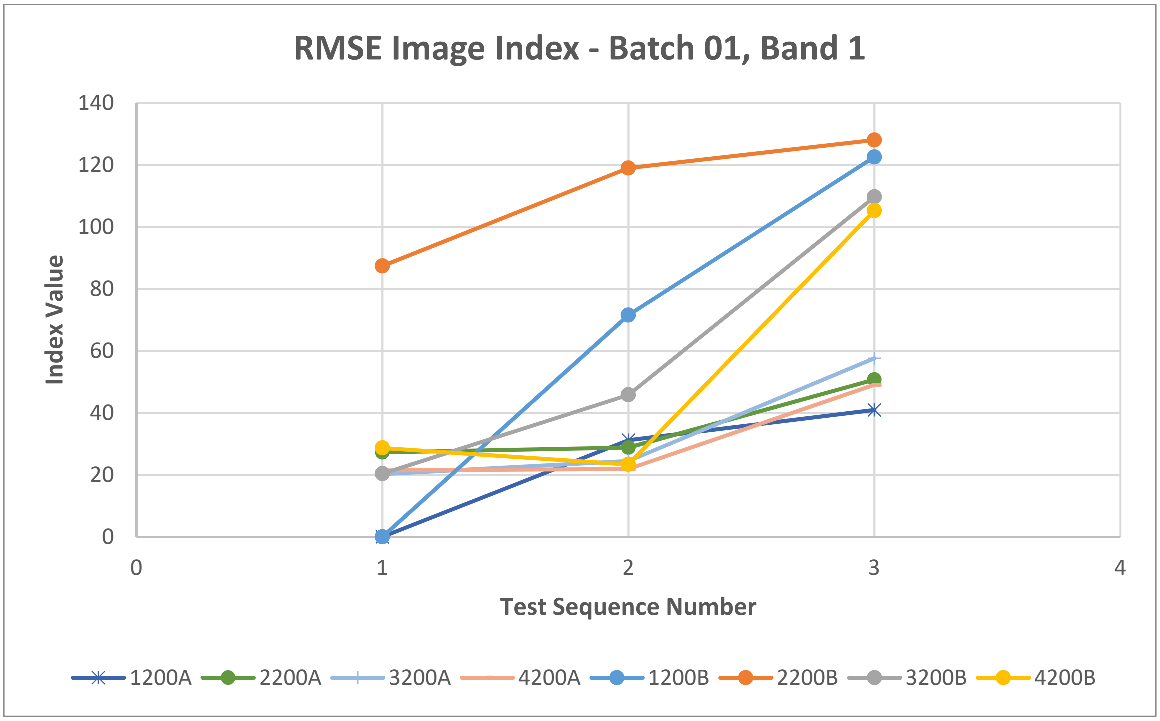


Figure 39. RMSE Image Index for Band 1, Side A & B per Test Sequence

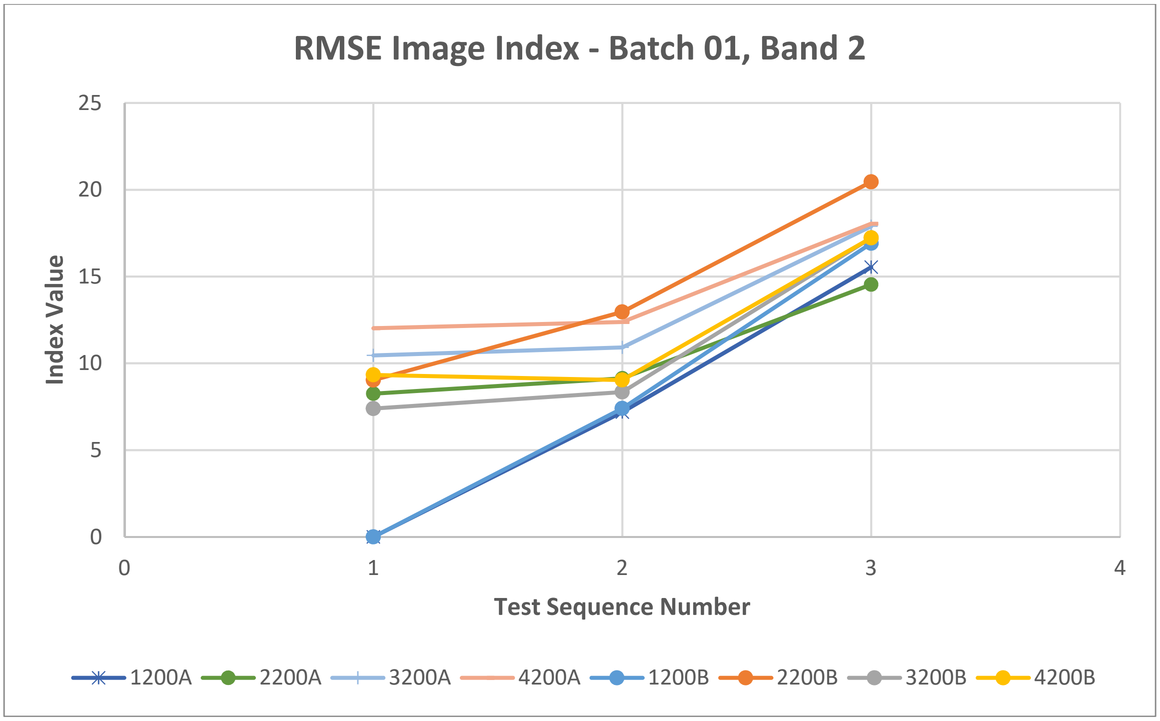


Figure 40. RMSE Image Index for Band 2, Side A & B per Test Sequence

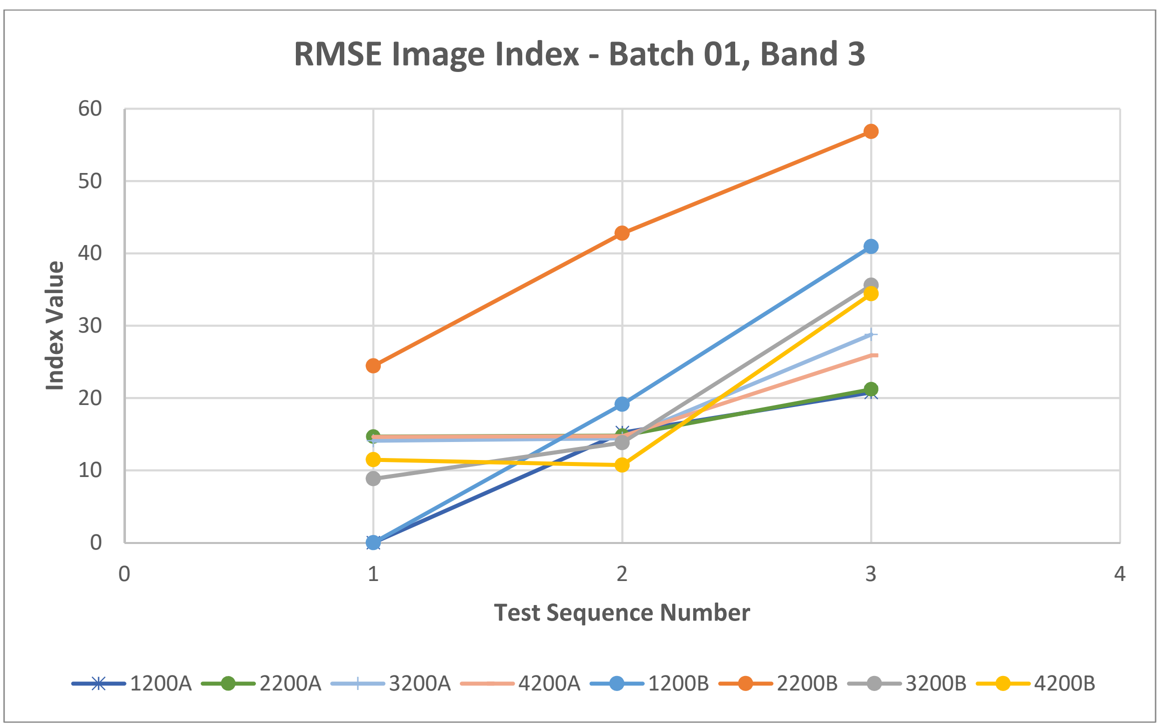


Figure 41. RMSE Image Index for Band 3, Side A & B per Test Sequence

The RMSE values for Side A appear to be consistently lower than that of Side B, however, in all cases, the error value increases from the first test sequence to the third. Side B appears to have a much higher error value in the red band when compared to the green or blue band. This correlates well with the correlation coefficient for both of these bands, as this value is consistently much lower for Side B. This is a good indication of how these indices are interconnected.

5.5.2 Batch 02 – Test Sequence Comparisons

An additional comparison of the images was performed, looking at the data sets as test sequences, instead of comparing each load level. This would enable the comparison of the output of strain values to the load level and determine if there is a positive correlation between them, or if there is a deviation to this prediction. To do this, the test sequence data was ordered similarly to that of the previous batch, where the images were categorized by the test sequence and the load level, however, the groups of data were larger as they would have three data sets with four images as opposed to four data sets with three images. For comparison, the delta value from each load level was determined for each spectral band, as well as the average, and these were plotted against the load level so that they could be compared in a graph across the test runs for each band and average value.

5.5.2.1 Batch 02 – Correlation Coefficient (CC) Index

The correlation coefficient (CC) index indicates the measure of the correlation between the strain in the reference image and the strain in each of the test images. This is a value that ranges from negative one to positive one, with a positive one being a perfect positive correlation, zero being no correlation, and negative one being a perfectly negative correlation. The closer to one, the better the correlation between the two values. A series of comparisons over the test sequences, for both Side A and B, can be seen in Figure 42 through Figure 44.

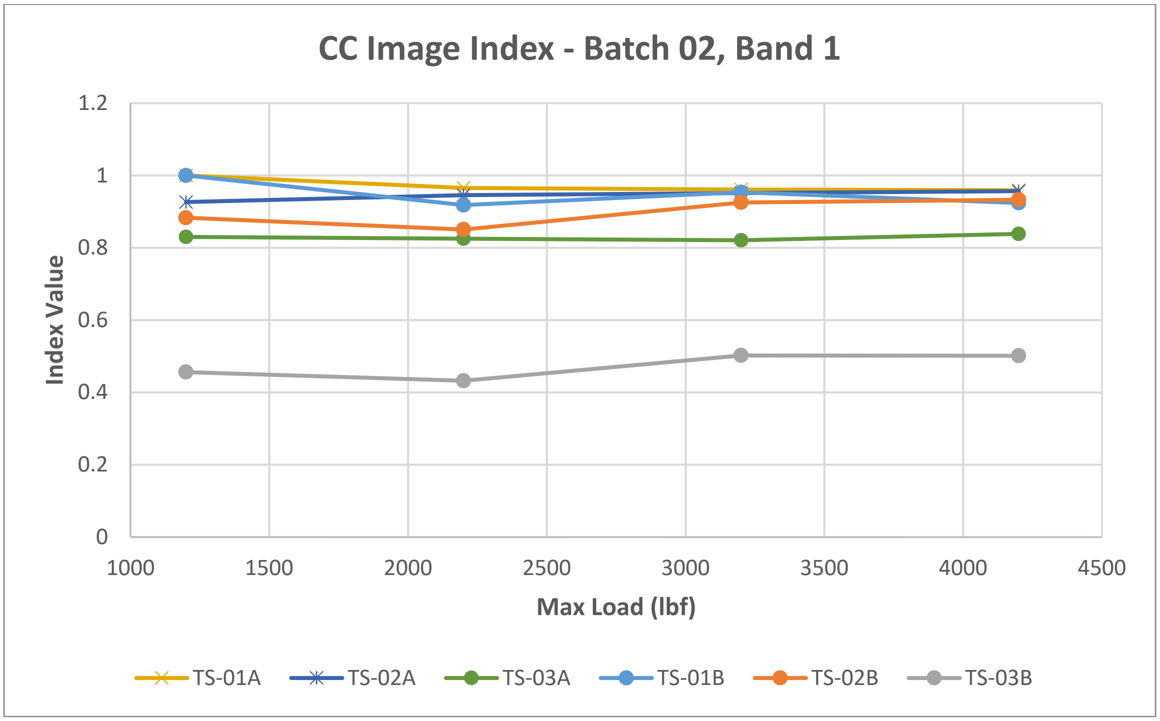


Figure 42. Correlation Coefficient Image Index for Band 1, Side A & B per Load Case

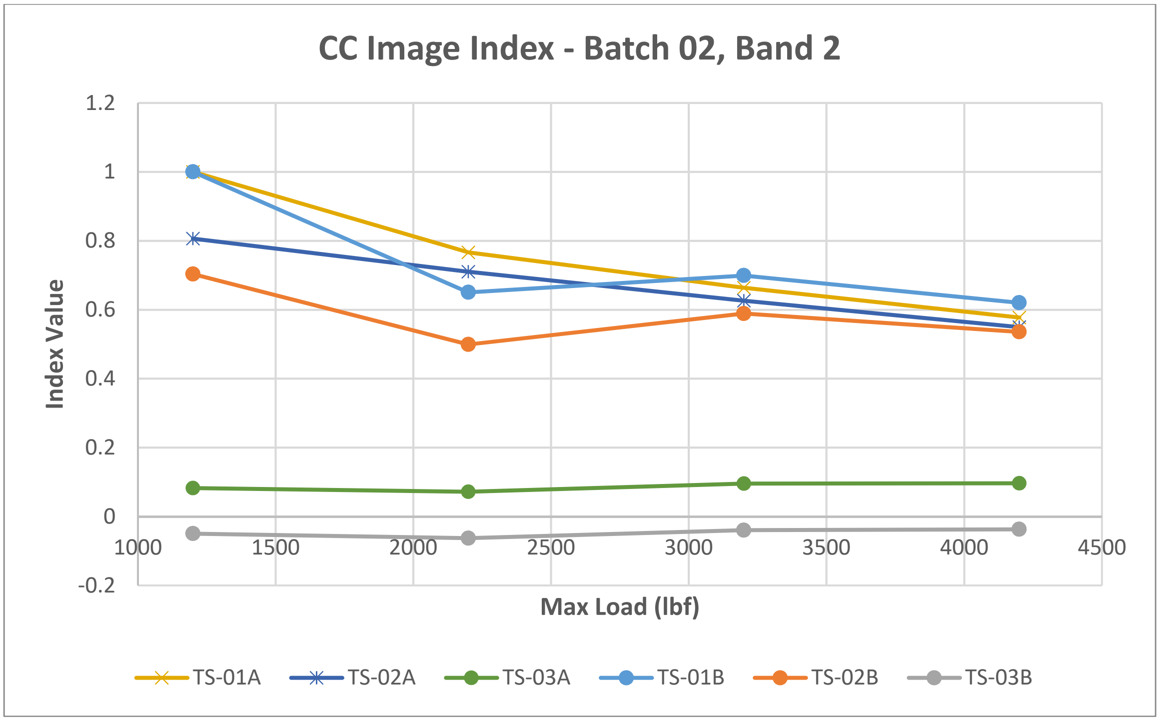


Figure 43. Correlation Coefficient Image Index for Band 2, Side A & B per Load Case

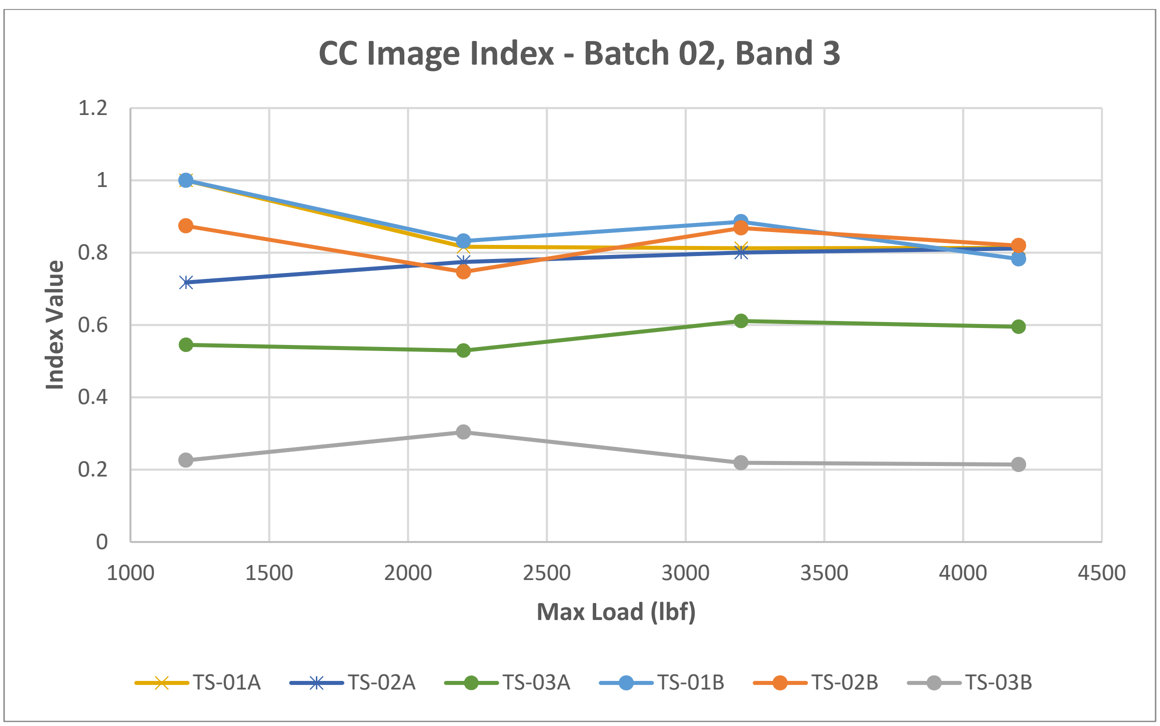


Figure 44. Correlation Coefficient Image Index for Band 3, Side A & B per Load Case

For Side A, the correlation values appear to remain fairly consistently clustered together from the 1200-lbf load case to the 4200-lbf load case. These values do decrease quite a bit more for the green band as opposed to the blue band. For the red band, they appear to remain fairly high. For both sides of the specimen, the third test sequence seems to produce a much lower value, specifically for Side B. The remaining values appear to follow a similar trend from side-to-side. The third test sequence for Side B seem to produce the lowest values for correlation, even shifting to a negative number for the green band.

5.5.2.2 Batch 02 – Image Quality (Q) Index

The quality index indicates the quality of the image, and is a product of the correlation coefficient, and two distortion factors, luminance and contrast. A series of comparisons over the test sequences, for both Side A and B, can be seen in Figure 45 through Figure 47.

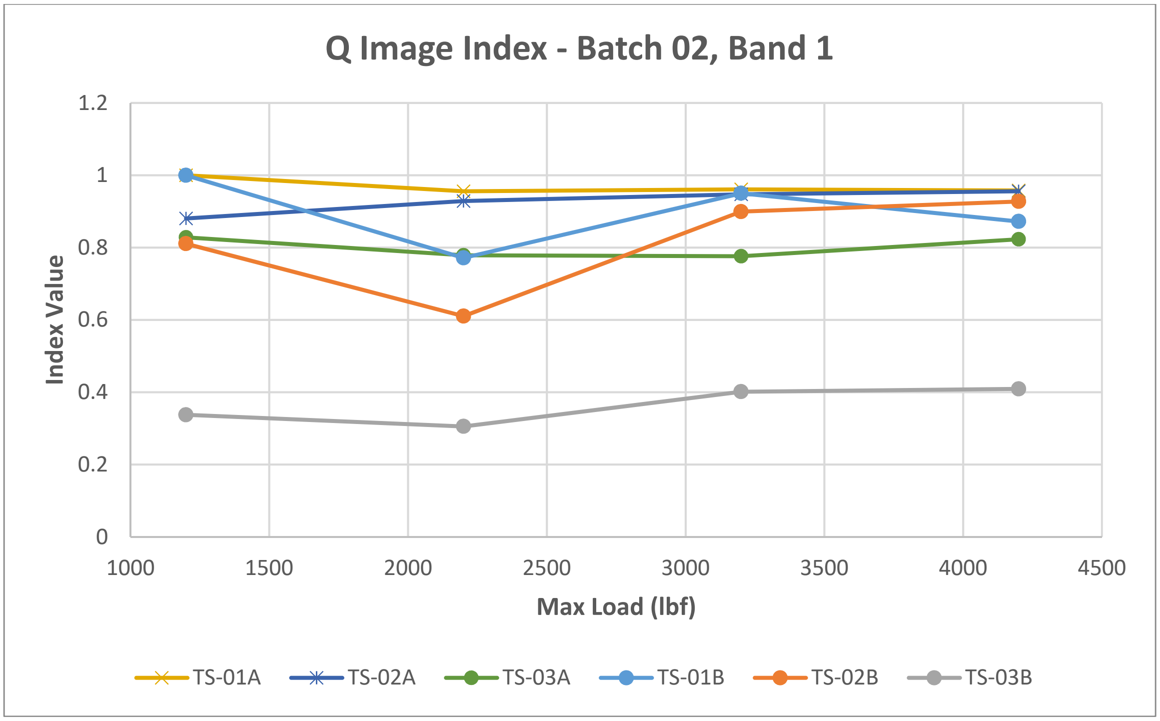


Figure 45. Quality Image Index for Band 1, Side A & B per Load Case

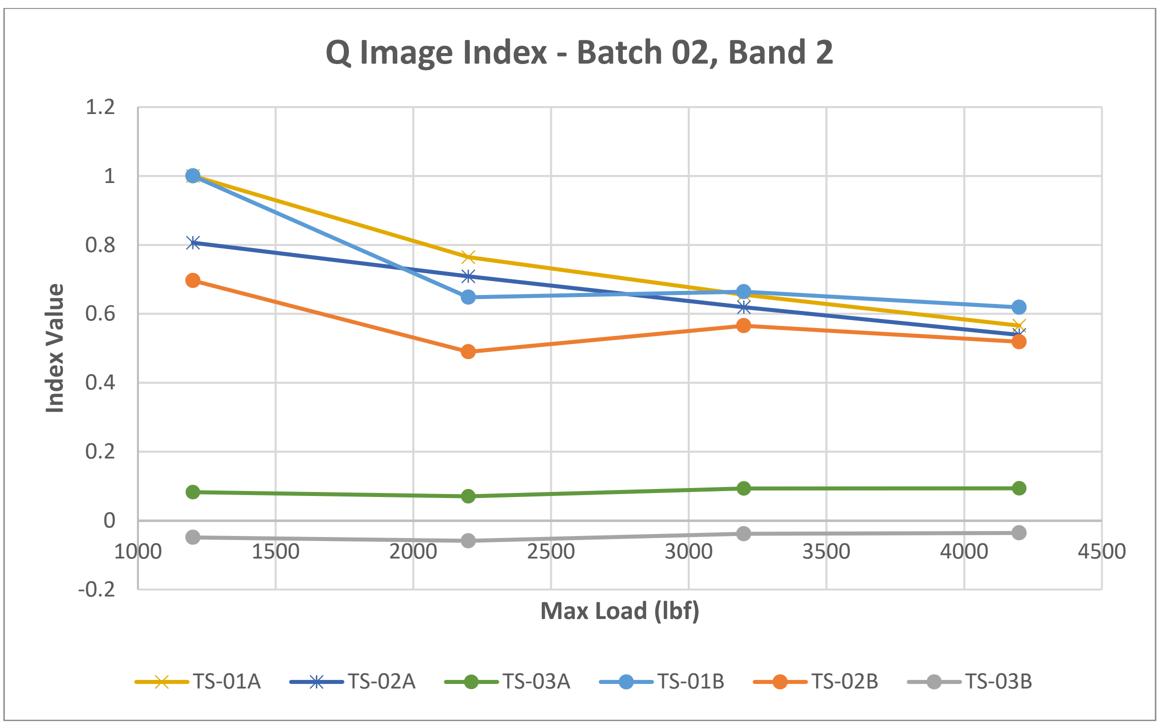


Figure 46. Quality Image Index for Band 2, Side A & B per Load Case

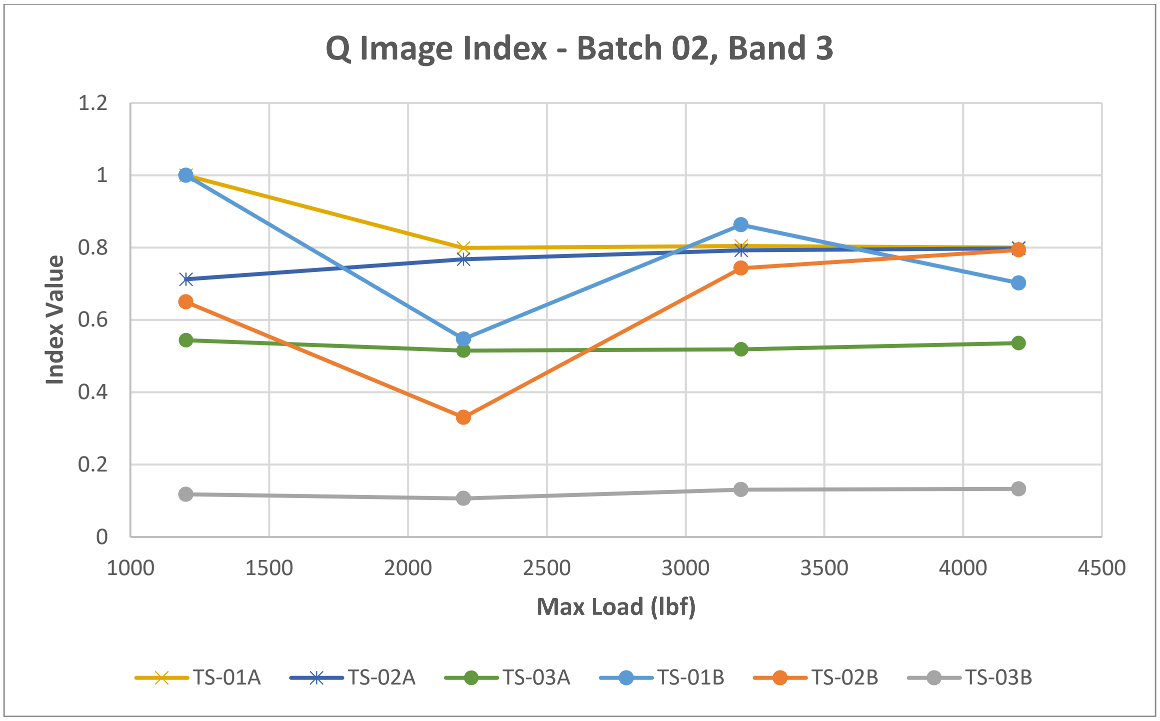


Figure 47. Quality Image Index for Band 3, Side A & B per Load Case

The quality value of the images from the red to the blue band appear to consistently decrease at the 2200-lbf load case and then level off for the red and green band, but in the blue band, they appear to cycle back up for Side B. Side A appears to level off after the decline. Once again, the values of the third test sequence appear lower than the previous two, which would indicate some sort of error or variance specifically in that test sequence.

5.5.2.3 Batch 02 – Luminance Distortion (LumDist) Index

The luminance distortion index indicates a measure of the "mean luminance between the factors x and y" [12], or the strain in the reference image and the strain in the test image. A series of comparisons over the test sequences, for both Side A and B, can be seen in Figure 48 through Figure 50.

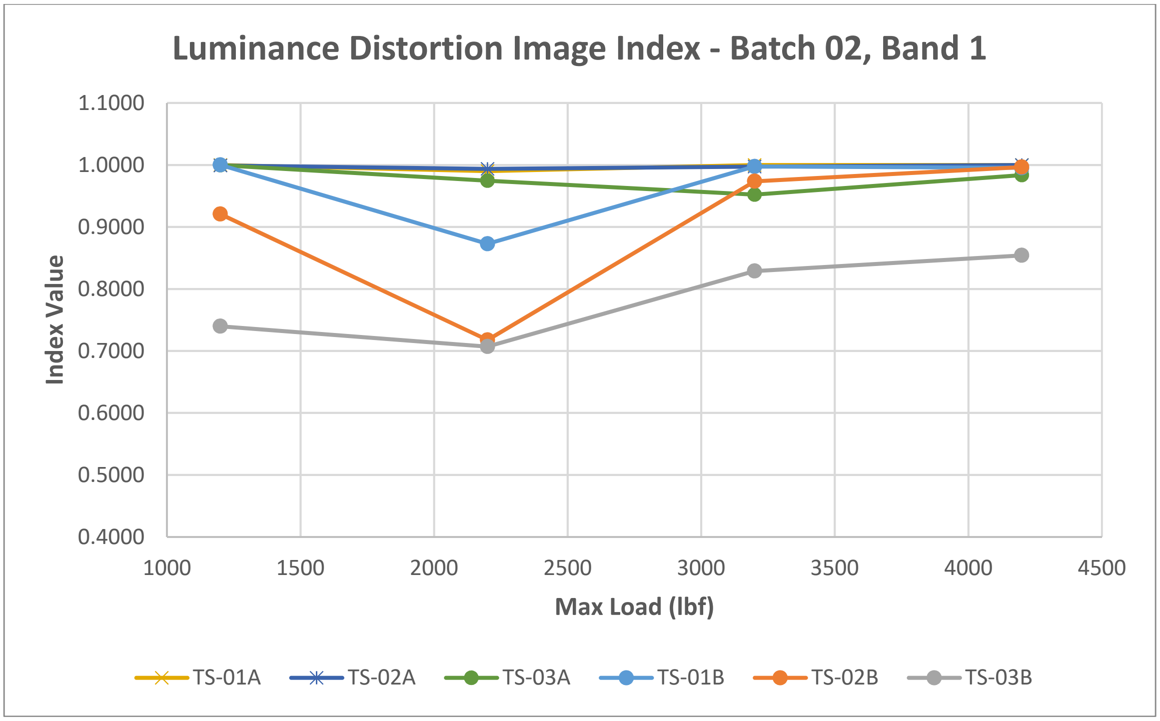


Figure 48. Luminance Distortion Image Index for Band 1, Side A & B per Load Case

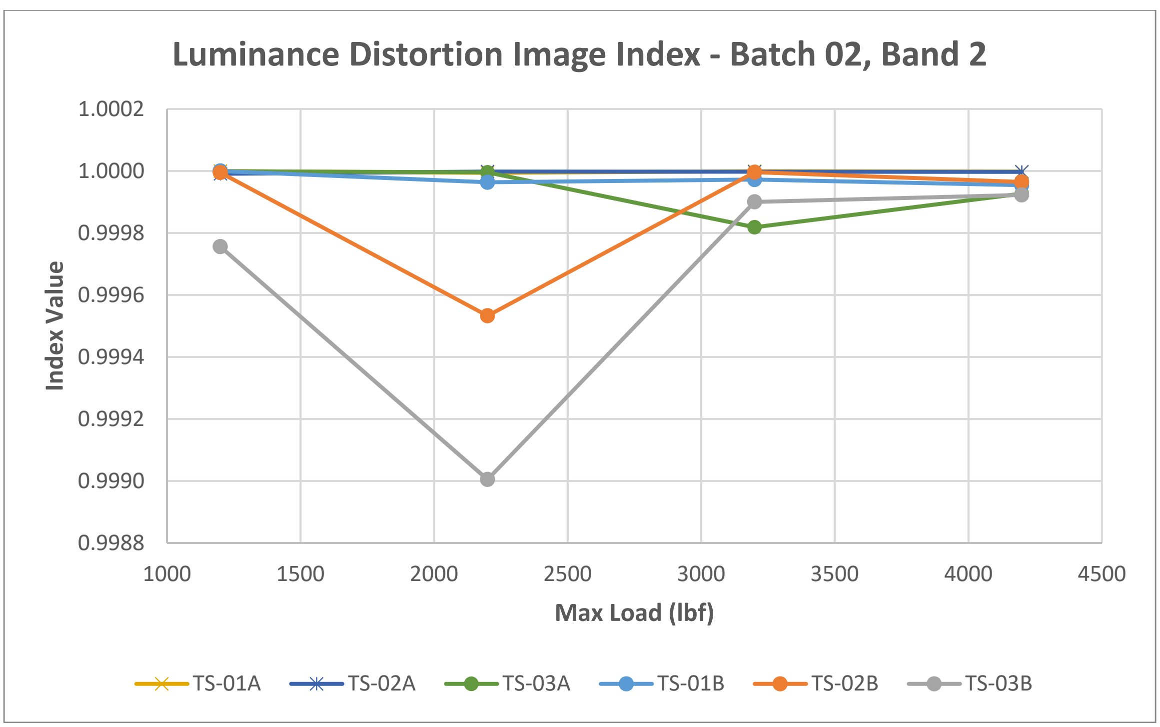


Figure 49. Luminance Distortion Image Index for Band 2, Side A & B per Load Case

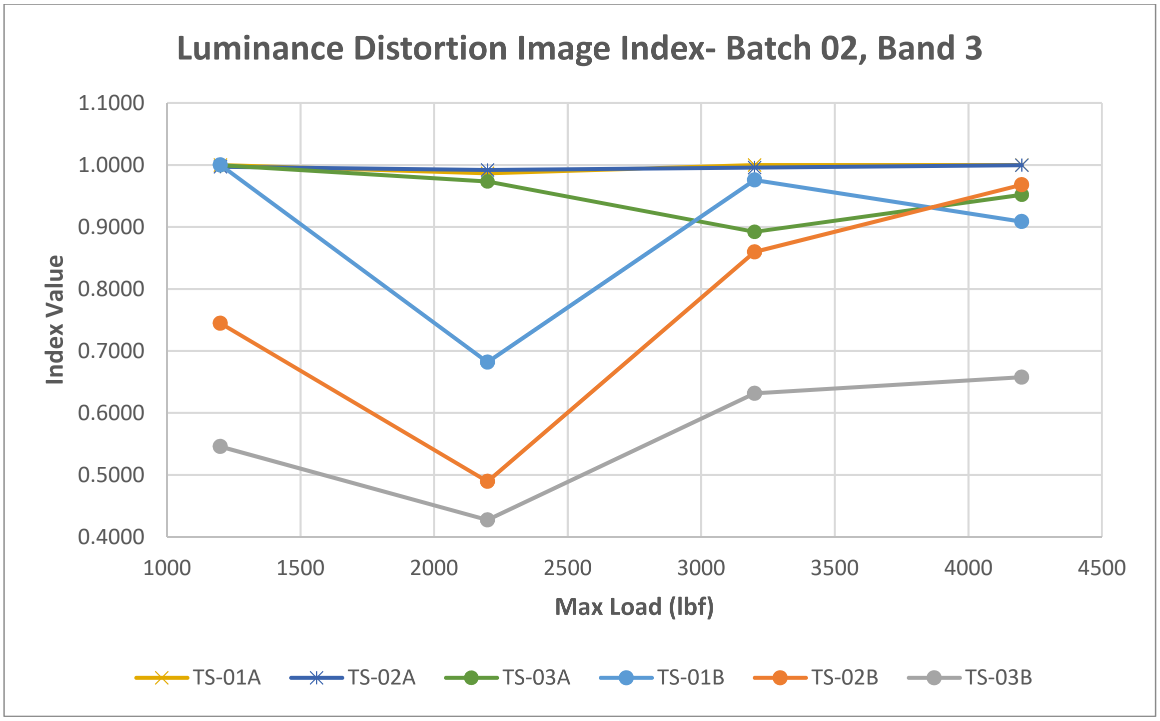


Figure 50. Luminance Distortion Image Index for Band 3, Side A & B per Load Case

The luminance distortion values have a similar decrease in values at the 2200-lbf load case, which is very pronounced specifically for Side B. Side A appears to be very linear and consistently near one. This would then call to question why this would vary quite so much from one side to another if the control of the ambient light was treated the same.

5.5.2.4 Batch 02 – Contrast Distortion (ConDist) Index

The contrast distortion index indicates a measure of the degree to which the contrast in the reference image and the test image are similar, or the same [12]. A series of comparisons over the test sequences, for both Side A and B, can be seen in Figure 51 through Figure 53.

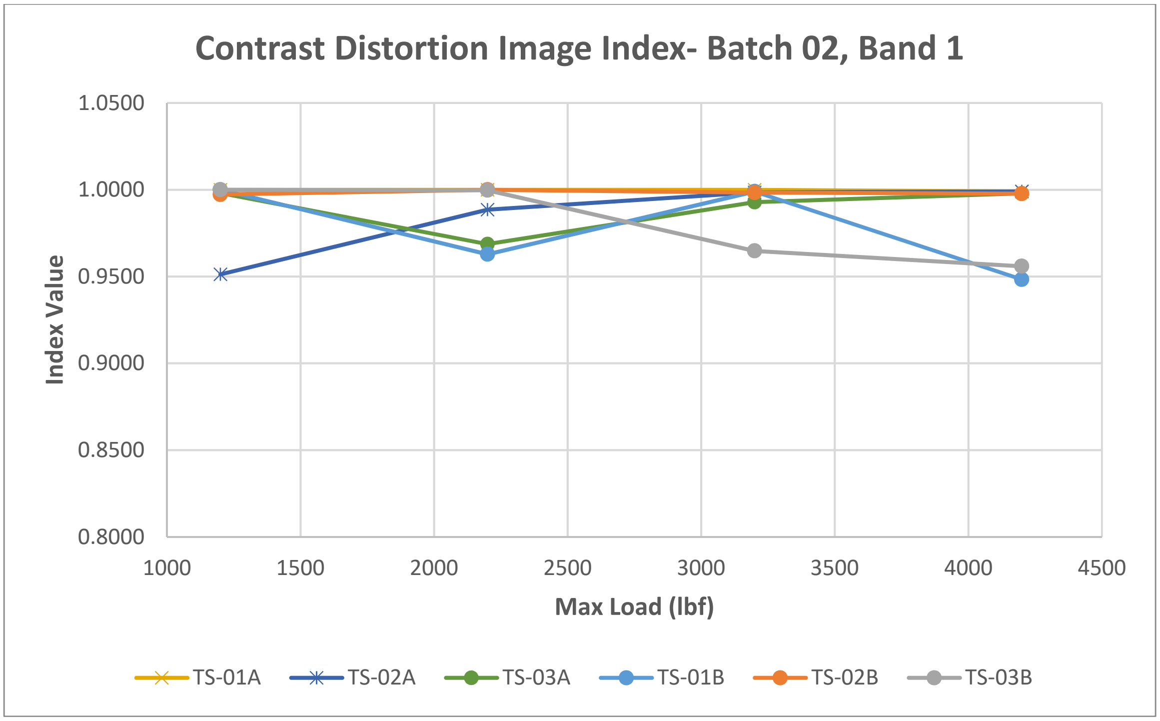


Figure 51. Contrast Distortion Image Index for Band 1, Side A & B per Load Case

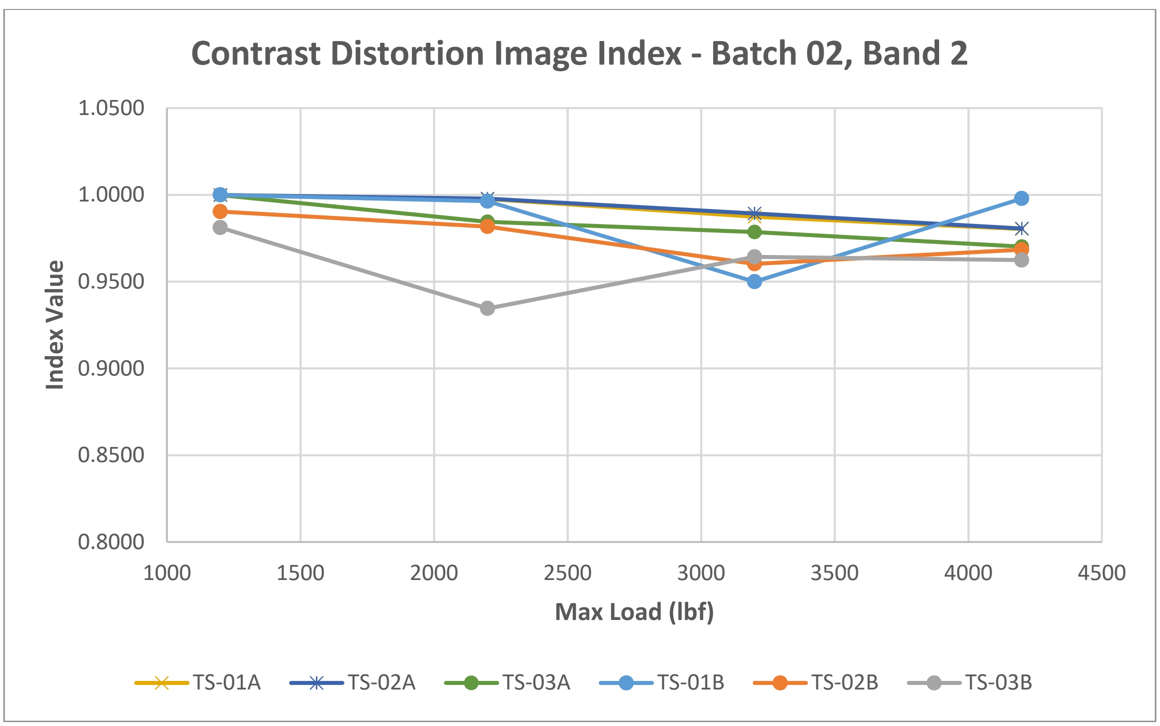


Figure 52. Contrast Distortion Image Index for Band 2, Side A & B per Load Case

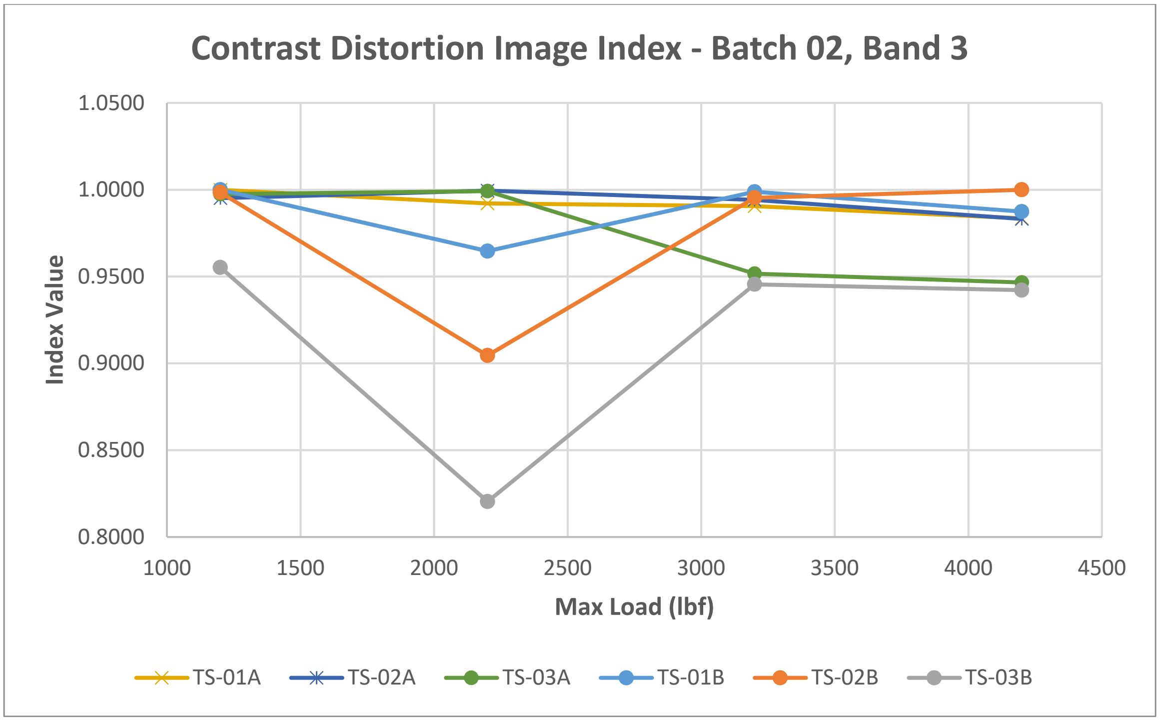


Figure 53. Contrast Distortion Image Index for Band 3, Side A & B per Load Case

The contrast distortion values appear to deviation more significantly from band to band, with the blue band having the most significant drop in value at the 2200-lbf load case. These values are consistently lower in the Side B data, where the Side A data appears to be flat line with a slight downward slope for the 4200-lbf load case.

5.5.2.5 Batch 02 – Root Mean Square Error (RMSE) Index

The root mean square error (RMSE) index indicates a deviation from the linear regression line, and is a direct relationship to the correlation coefficient. A series of comparisons over the test sequences, for both Side A and B, can be seen in Figure 54 through Figure 56.

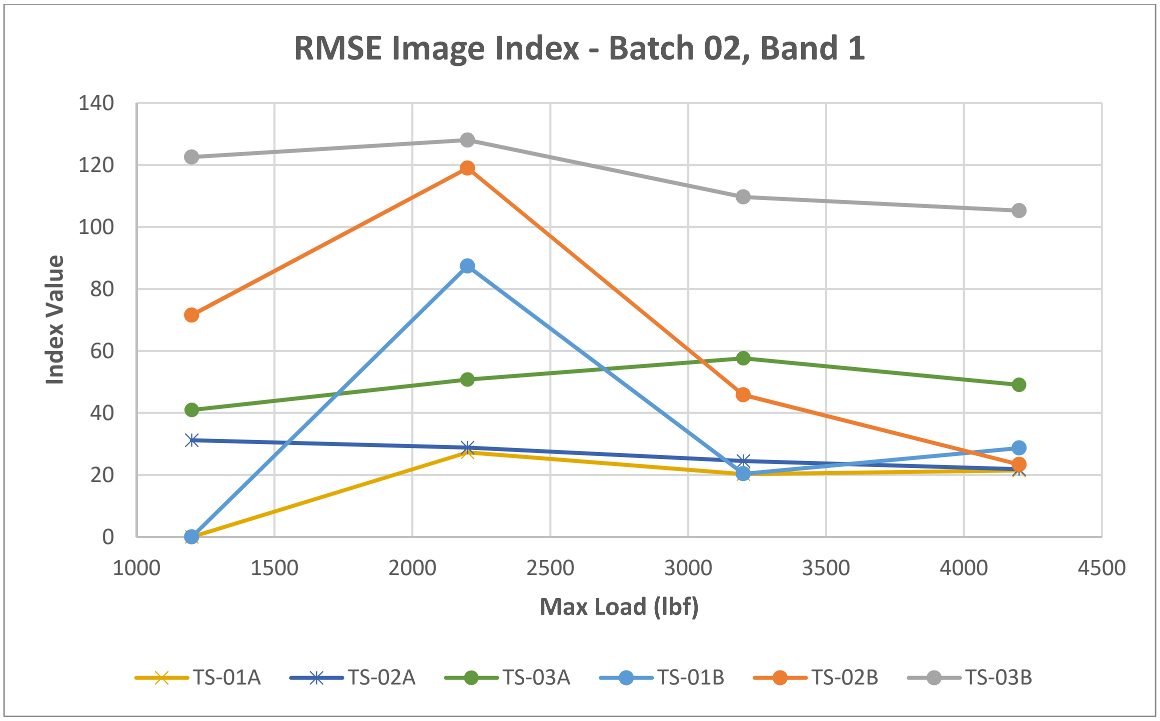


Figure 54. RMSE Image Index for Band 1, Side A & B per Load Case

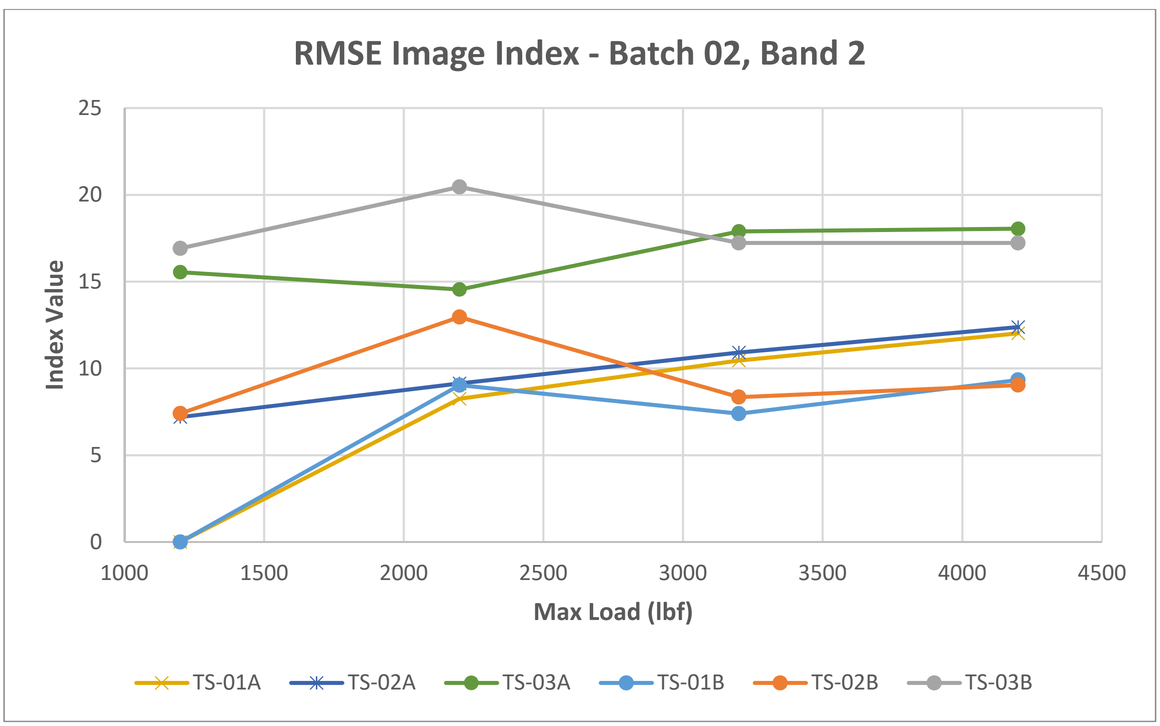


Figure 55. RMSE Image Index for Band 2, Side A & B per Load Case

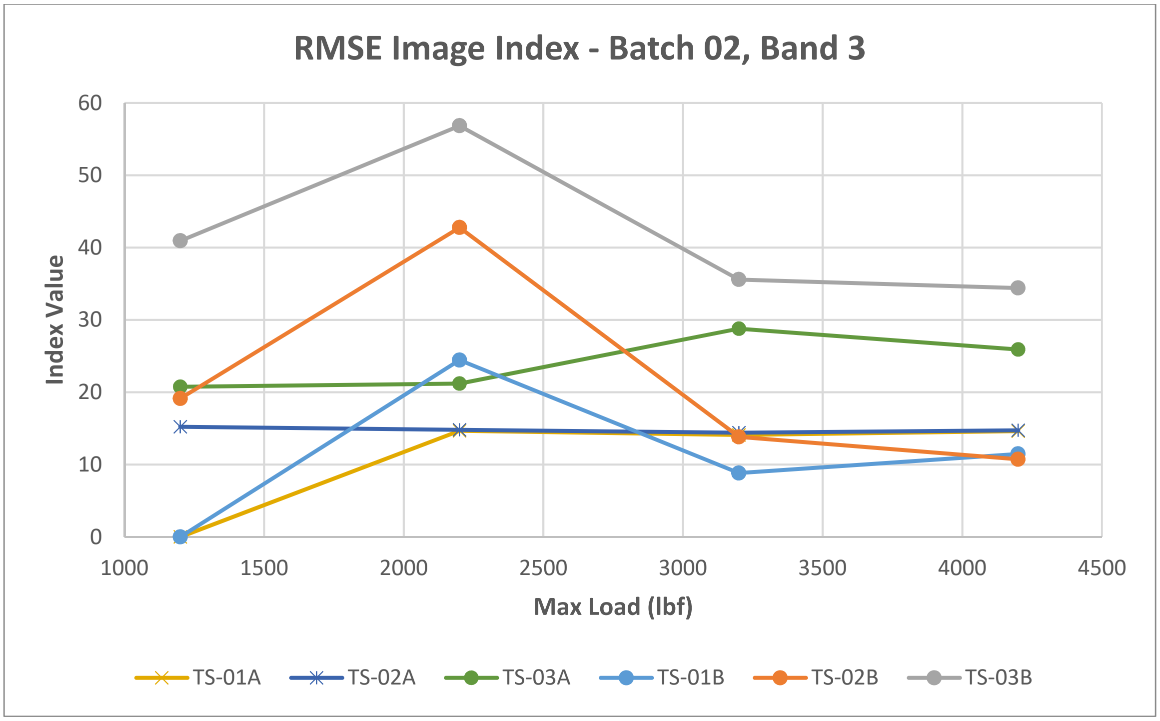


Figure 56. RMSE Image Index for Band 3, Side A & B per Load Case

The error value appears to peak in the 2200-lbf load case for each spectral band, indicating that the images captured have some sort of variation, however, this is consistent across the test sequences, and therefore, the cause is something that would require further investigation. Once again, the third test sequence appears to produce outlier data when compared to the first and second test sequences. This is the case for each band except the blue band, where there is a bit more variability in how the test sequences from side-to-side are clustering.

5.6 Percent Error and Index Values

To properly determine how to utilize the image quality index values, they must be compared to the percent error calculations for the 4200-lbf load case for both Side A and Side B of the specimen. The percent error between the TSA system data and the Model data can be seen in Table 21 and Table 22.

Table 21. Percent Error Calculations for Comparison of TSA System to Model for Side A

TSA Measured Strain (µƐ)	Model Predicted Strain (µƐ)	(Measured-Predicted)/Predicted	% Error
989	1477	-0.3300	-33.0%
1265	1437	-0.1192	-11.9%
1747	1264	0.3822	38.2%
3361	3015	0.1148	11.5%
2801	2134	0.3122	31.2%
1431	349	3.1049	310.5%
1161	-6	-193.4330	-19343.3%
912	-386	-3.3647	-336.5%

Table 22. Percent Error Calculations for Comparison of TSA System to Model for Side B

TSA Measured Strain (µƐ)	Model Predicted Strain (µƐ)	(Measured-Predicted)/Predicted	% Error
822	646	0.2713	27.1%
1123	1106	0.015288	1.5%
1419	1378	0.03009	3.0%
2705	3947	-0.3146	-31.5%
4613	4830	-0.04487	-4.5%
1690	2393	-0.2938	-29.4%
1483	2505	-0.4080	-40.8%
1130	2509	-0.5497	-55.0%

To compare the TSA and model values to those of the image quality indices, we have to consider the spectral bands that are in play for each area of the specimen that has been measured. The outer six gages are going to fall in the green band, while the center two values will fall in the red band. The specific images used for the quality index will be from test sequence three for Side A, and test sequence one for Side B. The values produced for Side A for the CC index are 0.8384 for the red band and 0.0965 for the green band. These make sense as the percent error values for the green band are significantly higher than that of the red band, with one of the points being extraordinarily high. This would indicate a very low, 0.0965, correlation coefficient value, and bolsters the lack of correlation in the data shown. For Side B, the values of error are highly negative for the red band, suggesting that the measured values are actual quite a bit lower than what is modeled. In fact, there quite a few values that are largely negative errors, suggesting that the model has an element that is creating a component that would see higher strains. The CC index value for the red band is 0.9241, and for the green band it is 0.6202. This seems to be in agreement with the deviations from the +/-10% error bands, as the green band has a much larger deviation. The red band and the point transitioning from the red band to green band, or orange in the image, is staying within the error bands of +/-10%. However, it is increasing across the centerline gradient to the edges. The correlation coefficient is the driving factor in the quality of the image, which would indicate why they are very close in value from Side A to Side B. Particularly, for Side A, the values are nearly identical with very high distortion values, at nearly one and not diverging below 0.97. Values of 0.95 and up would indicate a low issue with distortion, as the bounds for this are zero to one. For Side B, however, the contrast distortion in the red band has deviated below 0.95. This could be a contributing factor in the higher root mean square error of 28.6807 when compared to 9.3358 for the green band. The luminance distortion does not seem to be a factor for Side B, as this value is at one for the green band and 0.9955 for the red band. This would indicate that the contrast distortion between the reference image and the 4200-lbf load case is significant enough to reduce the quality of the image to 0.8724. The high correlation coefficient values and quality values in the red band, or the area in the image seeing the highest strain, would indicate that the TSA data is producing good quality image data for use in comparison to the model data. When comparing this to the model data, there seems to be quite a discrepancy with the model; however, this is also seen with the strain gages. This would indicate that the model is not correlating well to either method of capturing experimental data, and would require a closer look at the boundary conditions input for the model to see if this would remedy the discrepancies.

CHAPTER VI

SUMMARY AND CONCLUSIONS

6.1 Summary

The goal of this experimental endeavor was to capture TSA data, analyze the data produced by the MiTE TSA system using a Matlab image analysis program which calculated eight image indices (three of which were utilized and two components added), analyze the trends produced by these image indices, and determine how they could be utilized to aid in developing model correlation criteria when compared to a finite element model. Traditional comparison techniques were utilized to compare the data from strain gages, TSA images, and Abaqus model values. This work took the analysis a step farther and utilized statistical modeling equations to compare the images produced by the TSA system to determine their correlation trends, and how to apply these to aid in creating test-to-model correlation criteria when utilizing non-contact instrumentation such as TSA. The conclusions drawn from each type of analysis will be presented and discussed in the following sections.

6.2 Strain Gage Conclusions

Of the three different types of instrumentation utilized in this experimental endeavor, the strain gages are the easiest to quantify error, and they are a standard utilized across the mechanical testing community. As mentioned previously, two phases were utilized in providing strain data, however, the second phase with additional, smaller footprint gages provided a better comparison as this allowed the capture of the strain trends along the centerline of the hole, and enabled direct comparison to the TSA images and the finite element model. The values from Side A to Side B were compared to determine how well they correlated and determine the percent difference and error between them. As can be seen in Table 4 through Table 7, the percent difference in values for the majority of the strain gages between the test runs are at or below 5%. This provides high confidence in the validity of these values, especially since they are repeatable within 5%. This can then provide the truth data for comparison between the systems, as seen in Chapter 5.

6.3 Abaqus Model Conclusions

The values produced by the Abaqus model are slightly higher than that of the TSA images, as seen in the comparison between the systems for Side A and Side B in Figure 19 and Figure 20. However, they are following the same trend line, with the values being slightly lower than that of the strain gages and TSA system. This was determined to be due to the model being created with a free moving greased bearing, as opposed to a seized bearing experiencing friction, which would match the test conditions. With this being considered as a factor, the analytical results produced by the model are comparable to the experimental results. The correlation between the two will need to incorporate a larger margin for error, however, they correlate well, especially on Side B. To determine model correlation, the results from the 4200-lbf load case were compared to the results for the TSA system, as well as the strain gages installed along the centerline. In accordance with the *Air Force Structural Integrity Program Structures Bulletin, EN-SB-11-001* [1], the percent error was calculated for comparisons of the TSA to Strain Gages, TSA to Model, and SG to Model. The model appears to have decent correlation to the strain gage data and the TSA data. The data on the right side of the hole on Side A and the left side of the hole on Side B appear to correlate well with one another, and they are back to back faces.

However, the opposite side of the hole for both Side A and Side B both have a distinct offset or discrepancy in their comparison to the strain gage and TSA data sets.

6.4 Image Analysis Conclusion

The values of the image indices of correlation coefficient, quality, luminance distortion, contrast distortion, and root mean square error were compared to the values calculated for the percent error along the center line of the component and the location of the points in reference to the spectral bands was considered during comparison. These values at their respective bands were determined to bolster the percentage error, as they followed the trend of both a lower correlation/quality value, and a higher error value, indicating a deviation from the linear regression. Bounding these values much like the percentage error would aid in determining the correlation between the model and the TSA data.

6.5 Suggestions for Model Correlation

To correlate from the image index results produced, values for the image indices will require bounds, or a range of values between which the image index is able to fall to enable proper correlation to the finite element model. As the range for the correlation coefficient is negative one to positive one, with the goal of achieving a value of one, this value will need to fall between zero and one, more specifically, bounding this value like the percent error value, 0.90 to one would ensure a high correlation value, and a greater correlation to the model data. For the quality index, the same bounding would apply, as this value is highly dependent upon the correlation coefficient. The distortion factors for this experimental method seemed to have a very low effect on the quality value, however, these would be required to be bounded as well. Limiting these values to 0.95 to one would help ensure the distortion between the images is low, and the quality remains high enough to determine correlation. Further investigation into these bounds and how well they determine correlation to a finite element model is required, however, this work provides a foundation upon which to build.

6.6 Suggestions for Future Work

The work captured in this document details a statistical method to quantify the correlation coefficient, quality (including distortion factors), and root mean square error of thermal images produced by a microbolometer thermal camera. This quantification is based upon statistically determined image indices that are employed within a Matlab program and used to compare a reference image to test images captured at various load steps within the experiment. These image indices were then compared to the percentage error between the various instrumentation methods to determine if there is a correlation between these values when comparing the instrumentation systems to the model. To further investigate, and develop more quantifiable correlation criteria for utilizing this system for finite element model validation, it is suggested that the quality of the images is assessed as more local areas within the image, possibly using a sliding window in the image analysis. This would better enable the comparison at areas of higher strain, and for more direct comparison at areas of bulk or constant strain. For example, possibly utilizing a window size of 8x8 pixels to quantify the quality of the image at the strain gage within view, as the values on the strain gage are not all exactly the same. Or examine the quality of the image at the hole location. Taking a more localized approach to analyzing the image could help reduce any noise or error induced from the background, and enable a better analysis of the component when compared to a corrected model. Additionally, a consideration could be made for utilizing a different microbolometer, such as the FLIR A655sc to do a comparison between the images produced by each microbolometer. This is a slightly larger model, however, it is also compatible with the MiTE software and could be another microbolometer option.

BIBLIOGRAPHY

[1] Air Force Structural Integrity Program. *Guidance on Correlating Finite Element Models to Measurements from Structural Ground Tests*. Structures Bulletin. WPAFB: Wright Patterson Air Force Base, 2011.

[2] Oliver, David E. "Stress Pattern Analysis by Thermal Emission." *Journal of Experimental Mechanics* (1988): 610-620.

[3] Rajic, Nik, et al. "In situ thermoelastic stress analysis - an improved approach to airframe structural model validation." Quantitative InfraRed Thermography Journal (2018): 1-26.

[4] Pitarresi, G and E A Patterson. "A review of the general theory of thermoelastic stress analysis." Journal of Strain Analysis for Engineering Design 38.5 (2003): 405-417.

[5] Rajic, Nik and Rowlands, David. "Thermoelastic stress analysis with a compact low-cost microbolometer system." Quantitative InfraRed Thermography Journal 10.2 (2013): 135-158. <https://doi.org/10.1080/17686733.2013.800688>.

[6] Casavola, Caterina, et al. "Comparison of local strain measurements done by strain gage and differential thermographic technique." *Proceedings of SEM annual conference and exposition on experimental and applied mechanics, St. Louis (USA)*. 2006.

[7] Bagheri, Zahra S., Habiba Bougherara, and Radovan Zdero. "Thermographic Stress Analysis of Whole Bones and Implants." Experimental Methods in Orthopaedic Biomechanics. Academic Press, 2017. 49-64.

[8] Clay, Stephen, et al. "Comparison of diagnostic techniques to measure damage growth in a stiffened composite panel." Composites Part A: Applied Science and Manufacturing 137 (2020): 106030.

[9] Vaiopoulos, A. D. "Developing Matlab scripts for image analysis and quality assessment." Proceedings of SPIE. n.d.

[10] Vaiopoulos, Aristidis. "Hyperspectral Image Index Analysis." 2021. MATLAB Central File Exchange. <https://www.mathworks.com/matlabcentral/fileexchange/32637-hyperspectral-image-index-analysis>.

[11] Shippert, Peg. "Introduction to Hyperspectral Image Analysis." May 2021. https://spacejournal.ohio.edu/. Document.

[12] Wang, Zhou and Bovik, Alan C. "A Universal Image Quality Index." *IEEE Signal Processing Letters* 9.3 (2002): 81-84.

[13] Vishay MicroMeasurements. "Vishay Micro-Measurements Instruction Bulletin B-127-14." 2017.

[14] ASTM E4-16. "Standard Practices for Force Verification of Testing Machines." *ASTM International* (2016).

[15] Defense Science and Technology Group. "MiTE Installation and User Guide." 2013.

APPENDIX A

Modified Matlab Program Used

```
% Image index Analysis
% imanalysis
% 10/08/2011 - Version 1.0
% 22/08/2011 - Version 1.2 /Fixed a compatibility issue for older versions.
% 08/08/2011 - Version 1.7 /Made the filenames input easier with uigetfile,
%                          /dealt with invalid inputs.
% Author: Aristidis D. Vaiopoulos
% Acknowledgement: This program uses progressbar.m to display the estimated
% time left. Author of progressbar.m, is Steve Hoelzer.
% Modification:
% Edited: 2021 10 15
% Modifications added are to include the Luminance & Contrast Distortion
% components into the indices calculated and written (Author: Caitlin Jenkins)
% -----------------------------------------------------------------------
% ====================
% Program Description:
% ====================
% This is a program which utilizes the included functions in order to
% calculate 8 image indices (Bias, Correlation coefficient, DIV, Entropy,
% ERGAS, Q, RASE and RMSE). The purpose of the program is to produce the
% results fast, easily and in a convenient way for the user (see Outputs).
% Initially, its purpose was to perform index analysis in hyperspectral and
% multispectral satellite imagery. It has been used and tested in fused
% hyperspectral products for quality assessment of the spectral fidelity.
% However, it is estimated that it can be used for image comparison of
% similar or processed images, of completely different origin. Every
% included function can be used separately.
%
% Program Structure:
% ------------------
% 0) User runs the program by typing 'imanalysis' in the command window.
%
% Inputs:
% 1) User must provide the program with the number (nin) of the test images
% (test) he desires to compare, with the original image (orig).
% *** All images must have the exact same resolution ***
% 2) After image inputs, user is being asked for the h/l ERGAS ratio.
% 3) Then, the user has to input the filenames, first that of the original
% image and afterwards, those of the (nin) test images declared in step 1.
%
% Index analysis:
% 4) Program performs computation of all eight indices for every image and
% for every band, by using seven independent functions. The average value
% is calculated for every index. Total values are also computed for
% Entropy, ERGAS and RASE indices.
%
```

```
% Outputs:
% 5) Program outputs an Excel file, containing each index analysis results
% in a homonymous spreadsheet. For ease, or later statistical operations,
% a column has been added to the left, numbering the bands of the tested
% imagery and a row above, containing the filename. User of course, can
% examine and plot the index results from Matlab command window. By typing
% before the index 'c' and after the index 's', the cell array containing
% the certain index is shown. For example, to display ERGAS index, we must
% type 'cergass'. See lines 326-333, for every index (2nd arg in xlswrite).
%
% -Compatibility-
% -Oldest Matlab version tested: 7.0.1 (R14SP1). Bear in mind that you will
% not be able to analyze hyperspectral images with this version.
% -Oldest Matlab version known to have full functionallity: 7.6 (R2008a).
%
% *This program does NOT use sliding windows in index computations.*
%------------------------------------------------------------------------
% -Uncomment the line below, if your Matlab version has clearvars function-
% clearvars -except p excel_name
format short g;
format compact;
disp(' ');
disp('Image index Analysis');
disp('--------------------');
disp(' ');
% Try counter
tr_c      = 0;
nin       = 0;
resratio   = 0;
% Inputs...
% 1. Number of test datasets
while nin <= 0 | ~isreal(nin) | ischar(nin) | numel(nin)~=1  %#ok<*OR2>

  nin = input('How many images will you use?...              ');
  disp(' ')
  if (nin <= 0 | ~isreal(nin) | ischar(nin) | numel(nin)~=1)

    tr_c = tr_c + 1;
    if tr_c < 3
    disp('-Invalid input. Please try again.')
    disp(' ')
    end
    if tr_c == 3
      disp('-Invalid input.')
      disp(' ')
    return;
    end
  end
end
nin = round(nin);
% 2. ERGAS ratio
```

```
while resratio <= 0 | ~isreal(resratio) | ...
        ischar(resratio) | numel(resratio)~=1

    resratio = input('Please enter resolution ratio (H/L) for ERGAS...  ');
    disp(' ')

    if (resratio <= 0 | ~isreal(resratio) | ...
        ischar(resratio) | numel(resratio)~=1)

        tr_c = tr_c + 1;
        if tr_c < 3
        disp('-Invalid input. Please try again.')
        disp(' ')
        end
        if tr_c == 3
            disp('-Invalid input.')
            disp(' ')
        return;
        end
    end
end
% 3. Data inputs
impaths = cell(nin+1,1);
flnames = cell(nin+1,1);
i       = 1;
% Original data
disp('Please input the ORIGINAL image.....')
[filename, pathname] = uigetfile(...
{'*.bmp;*.gif*.jpg;*.png;*.tif',...
'Image Files (*.bmp *.gif *.jpg *.png *.tif )';
'*.*',  'All Files'}, ...
'Select the ORIGINAL Dataset');
impaths(i) = {[pathname filename]};
flnames(i) = {filename};
if filename == 0
    disp('-Program exits: Data input was canceled by user.')
    disp(' ')
    return;
end
disp(['                                    ', '"',filename,'"']);
disp(' ')
% Test data
for i = 1:nin

disp(['Please input the TEST image No. ', num2str(i),'...   ']);
[filename, pathname] = uigetfile(...
{'*.bmp;*.gif*.jpg;*.png;*.tif',...
'Image Files (*.bmp *.gif *.jpg *.png *.tif )';
'*.*',  'All Files'}, ...
'Select a TEST Dataset');
disp(['                                       ', '"',filename,'"']);
```

```
impaths(i+1) = {[pathname filename]};
flnames(i+1) = {filename};
if filename == 0
   disp('-Program exits: Data input was canceled by user.')
   disp(' ')
   return;
end
end
disp(' ')
%%
tic;
disp('=========================')
disp('Please wait. Working.....')
disp('=========================')
disp(' ')
disp('Parsing Original image...')
orig = importdata(char(impaths(1)));
disp('Original image parsed.')
disp(' ')
% Find number of bands
sizi = size(orig);
if max(size(size(orig))) == 2
   bands = 1;
else
   bands = sizi(1,3);
end
% Preallocation
% Indices per band
bias        = zeros(bands,nin+1);
cc          = ones(bands,nin+1);
div         = zeros(bands,nin+1);
Epb         = zeros(bands,nin+1);
ergas_pb    = zeros(bands,nin+1);
qs          = ones(bands,nin+1);
lds         = ones(bands,nin+1);
cds         = ones(bands,nin+1);
rase_pb     = zeros(bands,nin+1);
rmses       = zeros(bands,nin+1);
% Average indices
av_bias     = zeros(1,nin+1);
av_cc       = ones(1,nin+1);
av_div      = zeros(1,nin+1);
Eav         = zeros(1,nin+1);
av_ergas    = zeros(1,nin+1);
av_q        = ones(1,nin+1);
av_ld       = zeros(1,nin+1);
av_cd       = zeros(1,nin+1);
av_rase     = zeros(1,nin+1);
av_rmse     = zeros(1,nin+1);
% Total indices (whole image)
Etl         = zeros(1,nin+1);
```

```
ergas_tl    = zeros(1,nin+1);
rase_tl     = zeros(1,nin+1);
% Find if the function progressbar exists, if the user wishes to use it
fp = 0;
p_ex = 0;
if ~exist('p') || p ~= 0 %#ok<EXIST>
   p = 1;
   p_ex = exist('progressbar.m'); %#ok<EXIST>
end
if p_ex == 2
   fp = 1;
end
if p ~= 0 && fp == 0
   disp('*Cannot find progressbar.m')
   disp('*Estimated time left will not be shown.')
   disp(' ')
end
if p == 1 && fp == 1
   progressbar
end
for m = 2:nin+1

   disp(['Parsing Test image # ', num2str(m-1),'...'])
   test = importdata(char(impaths(m)));
   disp(['Test image # ', num2str(m-1), ' parsed.'])
   disp(['Analysing Test image # ', num2str(m-1), '....'])
   disp(' ')
   disp('-Calculating index 1 of 10: Bias...')
   [bias(:,m), av_bias(m)]                 = bias_f(orig,test);
   disp('-Bias has been computed.')
   disp('-Calculating index 2 of 10: DIV...')
   [div(:,m), av_div(m)]                   = div_f(orig,test);
   disp('-DIV has been computed.')
   disp('-Calculating index 3 of 10: Entropy...')
   [Epb(:,m), Eav(m), Etl(m)]              = entropia_f(test);
   disp('-Entropy has been computed.')
   disp('-Calculating index 4 of 10: ERGAS...')
   [ergas_pb(:,m), av_ergas(m), ergas_tl(m)] = ergas_f(orig,test,resratio);
   disp('-ERGAS has been computed.')
   disp('-Calculating indices 5 + 6 of 10: CC + Q...')
   [qs(:,m), av_q(m), cc(:,m), av_cc(m)]      = q_f(orig,test);
      disp('-Calculating indices 7 + 8 of 10: LumDist + ConDist...')
   [lds(:,m), av_ld(m), cds(:,m), av_cd(m)]      = distort_f(orig,test);
   disp('-LumDist and ConDist have been computed.')
   disp('-Calculating index 9 of 10: RASE...')
   [rase_pb(:,m), av_rase(m), rase_tl(m)]    = rase_f(orig,test);
   disp('-RASE has been calculated.')
   disp('-Calculating index 10 of 10: RMSE...')
   [rmses(:,m), av_rmse(m)]                = rmse_f(orig,test);
   disp('-RMSE has been calculated.');
   disp(' ')
```

```
    % Since nin is relatively small, this check does not decrease
    % significantly the execution time.
    if p == 1 && fp == 1
        progressbar((m-1)/nin)
    end

end
disp('-Calculating Entropy of Original image...')
[Epb(:,1), Eav(1), Etl(1)]              = entropia_f(orig);
disp('-Entropy of Original image computed.')
disp(' ')
disp('-Image index analysis completed.')
disp(' ')
ttime = toc;
%%
% Write excel file with all index analysis results
disp('-Preparing Excel Output...')
% Disable AddSheet Warning
warning off MATLAB:xlswrite:AddSheet
% Band count
cbnd    = num2cell(1:bands)';
cbnda   = [ {'BAND #'}; cbnd ; cell(1,1) ; {'AVERAGE'} ];
cbndt   = [ cbnda; cell(1,1) ; {'TOTAL'} ];
e_line =  cell(1,nin+1);
% Prepare cell per index
% Bias
cbiass = [ flnames' ; num2cell(bias); e_line; num2cell(av_bias)];
cbiass = [ cbnda, cbiass ];
% DIV
cdivs = [ flnames' ; num2cell(div); e_line; num2cell(av_div)];
cdivs = [ cbnda, cdivs ];
% Entropy
centropys = [ flnames' ; num2cell(Epb); e_line; num2cell(Eav);...
            e_line   ; num2cell(Etl)                    ];
centropys = [ cbndt, centropys ];
% ERGAS
cergass = [ flnames' ; num2cell(ergas_pb); e_line; num2cell(av_ergas);...
           e_line  ; num2cell(ergas_tl)                  ];
cergass = [ cbndt, cergass ];
% CC
cccs = [ flnames' ; num2cell(cc); e_line; num2cell(av_cc)];
cccs = [ cbnda, cccs ];
% Q
cqss = [ flnames' ; num2cell(qs); e_line; num2cell(av_q)];
cqss = [ cbnda, cqss ];
% Luminance
cldss = [ flnames' ; num2cell(lds); e_line; num2cell(av_ld)];
cldss = [ cbnda, cldss ];
% Constrast Distortion
ccdss = [ flnames' ; num2cell(cds); e_line; num2cell(av_cd)];
ccdss = [ cbnda, ccdss ];
```

```
% RASE
crases = [ flnames' ; num2cell(rase_pb); e_line; num2cell(av_rase);...
         e_line ; num2cell(rase_tl)                ];
crases = [ cbndt, crases ];
% RMSE
crmses = [ flnames' ; num2cell(rmses); e_line; num2cell(av_rmse)];
crmses = [ cbnda, crmses ];
% Check if there is a prefered name for excel output, else use default
if ~exist( 'excel_name' ) %#ok<EXIST>
   excel_name = 'Image Index Analysis_L+D.xls';
end
% Write every index in the homonymous excel spreadsheet
xlswrite( excel_name, cbiass,      'Bias')
xlswrite( excel_name, cdivs,       'DIV')
xlswrite( excel_name, cccs,        'CC')
xlswrite( excel_name, centropys,   'Entropy')
xlswrite( excel_name, cergass,     'ERGAS')
xlswrite( excel_name, cqss,        'Q')
xlswrite( excel_name, cldss,       'LumDist')
xlswrite( excel_name, ccdss,       'ConDist')
xlswrite( excel_name, crases,      'RASE')
xlswrite( excel_name, crmses,      'RMSE')
disp('-All indices have been written successfully in Excel file.')
disp(' ')
```

APPENDIX B

Bias Index Function

```
function [bias av_bias] = bias_f(x,y)
% Bias calculator
% Formula: 1 - mean(fused image)/mean(original image)
% (Ideal value = 0)
% 07/03/2010 Version 1.0
% 25/06/2010 Version 1.2    - Excel Output option
% 04/08/2011 Version 1.2F   - Function Version
% Author: Aristidis D. Vaiopoulos
% Find the number of bands
bands = size(x);
if length(bands) == 3
   bands = bands(1,3);
else
   bands = 1;
end
% Preallocation
mx = zeros(1,bands);
my = zeros(1,bands);
% Mean value calculation
for i = 1:bands
   xt = double(x(:,:,i));
   yt = double(y(:,:,i));
   mx(i) = mean(xt(:));
   my(i) = mean(yt(:));
end
% Bias calculation
bias = 1 - (my./mx);
bias = bias';
av_bias = mean(bias);
end
```

APPENDIX C

Modified Distortion Indices Function

```
distort_f.m
function [lds av_ld cds av_cd] = distort_f(x,y)
% Q index
% Universal image quality calculator
% 11/08/2009 Version 1.0
% 02/12/2009 Version 1.5    - Hyperspectral support
% 05/12/2010 Version 3.0    - Double precision, RAM efficiency, Progressbar
% 25/06/2010 Version 3.2    - Excel Output option
% 07/08/2011 Version 3.4F   - Function Version
% Author: Aristidis D. Vaiopoulos
% Modification:
% Edited: 2021 10 15
% Edited to include luminance and contrast distortion sections (Author: Caitlin Jenkins)
% Find number of bands
bands = size(x);
if length(bands) == 3
   bands = bands(1,3);
else
   bands = 1;
end
% Preallocation
meansx = zeros(bands,1);
meansy = zeros(bands,1);
sdsx   = zeros(bands,1);
sdsy   = zeros(bands,1);
for i = 1:bands;
    xt = double(x(:,:,i));
    yt = double(y(:,:,i));
    % Statistics for each band
    meansx(i) = mean(xt(:));
    meansy(i) = mean(yt(:));
    sdsx(i)   = std2(xt);
    sdsy(i)   = std2(yt);

end

% Luminance Distortion
lds = ((2.*meansx.*meansy)./(meansx.^2 + meansy.^2));

% Contrast Distortion
cds = ((2.*sdsx.*sdsy)./(sdsx.^2 + sdsy.^2) );

% Calculate mean luminance and constrast distortions
av_ld = mean(lds);
av_cd = mean(cds);
end
```

APPENDIX D

Divergence Index Function

```
div_f.m
function [div, av_div] = div_f(x,y)
% DIV calculator
 %Difference In Variance
% Formula: 1 - variance(fused image)/variance(original image)
% (Ideal value = 0)
% 07/03/2010 Version 1.0
% 25/06/2010 Version 1.2    -   Excel Output option
% 06/08/2011 Version 1.2F   -   Function version
% Author: Aristidis D. Vaiopoulos
%Find the number of bands
bands = size(x);
if length(bands) == 3
   bands = bands(1,3);
else
   bands = 1;
end
%Preallocation
sdx = zeros(bands,1);
sdy = zeros(bands,1);
%Standard deviations
for i = 1:bands
   xt = double(x(:,:,i));
   yt = double(y(:,:,i));
   sdx(i) = std2(xt);
   sdy(i) = std2(yt);
end
%Variance = (Standard deviation)^2
varx = sdx.^2;
vary = sdy.^2;
%Calculate DIV
div = 1 - (vary./varx);
%Average DIV
av_div = mean(div);
end
```

APPENDIX E

Entropy Index Function

```
entropia_f.m
function [Epb Eav Etl hst] = entropia_f(I,Iclass)
% Entropy (E) of intensity image
% 06/08/2011    - Version 4.0F
% Formula: E = -sum(p.*log2(p));
% Author:           Aristidis D. Vaiopoulos
% Acknowledgement:  Mathworks MATLAB entropy function
% If image class is not forced by user, detect and use native class
if nargin == 1;
    Iclass = class(I);
end
% Construct the bin values and convert image according to Iclass
switch Iclass
   case {'logical'}
%       cs = 1;
      cl = 2^1;
      I = logical(I);
      bv = [0 1];
   case {'uint8'}
%       cs = 2;
      cl = 2^8;
      I = im2uint8(I);
      bv = 0:(2^8)-1;
   case {'int16'}
%       cs = 3;
      cl = 2^16;
      I = im2int16(I);
      bv = -( (2^16)/2 ):( (2^16)/2-1 );
   case {'uint16'}
%       cs = 4;
      cl = 2^16;
      I = im2uint16(I);
      bv = 0:(2^16)-1;
   case {'single','double'}
%       cs = 5;
      cl = 2^16;
      I = im2uint16(I);
      bv = 0:(2^16)-1;
    otherwise
%       cs = -1;
      disp('-Unsupported data class.')
    return;

end
% Transpose bin values
bv = bv';
```

```
% Find the number of bands
bands = size(I);
if length(bands) == 3
    bands = bands(1,3);
else
    bands = 1;
end
% Calculate histogram counts per band
p = zeros(cl,bands);
for b = 1:bands
    p(:,b) = imhist(I(:,:,b),cl);
end
% Give histogram
if nargout == 4
    hst = [bv p];
end
% normalize p so that sum(p) is one.
p = p ./ (numel(I)/bands);
% normalize p for whole image.
pt = sum(p,2)/bands;
% logarithmization
lp = log2(p);
lpt= log2(pt);
% nullify -Inf values due to logarithmization of zeros in p
lp(isinf(lp)) = 0;
lpt(isinf(lpt)) = 0;
% Entropy per band
Epb = -sum(p.*lp,1)';
% Average Entropy
Eav = mean(Epb);
% Total Entropy (whole image)
Etl = -sum(pt.*lpt);
end
```

APPENDIX F

ERGAS Index Function

```
ergas_f.m
function [ergas_pb av_ergas ergas_tl] = ergas_f(x,y,resratio)
% ERGAS calculator
% 19/08/2009 Version 1.0
% 02/12/2009 Version 1.5    - hyperspectral support
% 05/03/2010 Version 2.0    - Double precision, RAM efficiency
% 07/03/2010 Version 3.0    - Progressbar
% 07/08/2011 Version 3.0F   - Function Version
% Author: Aristidis D. Vaiopoulos
%Find the number of bands
sizi = size(x);
if max(size(size(x))) == 2
   bands = 1;
else
   bands = sizi(1,3);
end
%RMSE part
nres = sizi(1,1)*sizi(1,2);
%Variable preallocation
meansx  = zeros(bands,1);
%meansy  = zeros(bands,1);
RMSE = zeros(bands,1);
for i = 1:bands
   xt = double(x(:,:,i));
   yt = double(y(:,:,i));
   %Mean value calculation for ERGAS
   meansx(i,1) = mean(xt(:));
   %meansy(i,1) = mean(yt(:));
   %RMSE
   RMSE(i) = sqrt((sum(sum((xt - yt).^2)))/nres);
end
%End of RMSE part
%ERGAS part
presratio = 100*resratio;
ergasroot = sqrt(  (RMSE.^2)./(meansx.^2)   );
ergas_pb  = presratio*ergasroot;
av_ergas  = mean(ergas_pb);
ergasroot = sqrt((sum((RMSE.^2)./(meansx.^2))) / bands);
ergas_tl  = presratio*ergasroot;
end
```

APPENDIX G

Modified Quality Index Function

```
q_f.m
function [qs av_q cc av_cc] = q_f(x,y)
% Q index
% Universal image quality calculator
% 11/08/2009 Version 1.0
% 02/12/2009 Version 1.5    - Hyperspectral support
% 05/12/2010 Version 3.0    - Double precision, RAM efficiency, Progressbar
% 25/06/2010 Version 3.2    - Excel Output option
% 07/08/2011 Version 3.4F   - Function Version
% Author: Aristidis D. Vaiopoulos
% Modification:
% Edited: 2021 10 15
% Edited to include luminance and contrast distortion sections (Author: Caitlin Jenkins)
% Find number of bands
bands = size(x);
if length(bands) == 3
   bands = bands(1,3);
else
   bands = 1;
end
% Preallocation
meansx = zeros(bands,1);
meansy = zeros(bands,1);
sdsx   = zeros(bands,1);
sdsy   = zeros(bands,1);
cc     = zeros(bands,1);
for i = 1:bands;
   xt = double(x(:,:,i));
   yt = double(y(:,:,i));
   % Statistics for each band
   meansx(i) = mean(xt(:));
   meansy(i) = mean(yt(:));
   sdsx(i)   = std2(xt);
   sdsy(i)   = std2(yt);
   % Correlation Coefficient for each band
   cc(i) = corr2(xt,yt);
end
% Quality for each band
qs = (  ( cc .* ( (2.*meansx.*meansy) ./ (meansx.^2 + meansy.^2)  ) ...
          .* ( (2.*sdsx  .*sdsy  ) ./ (sdsx.^2 + sdsy.^2) ) )  ) ;

% Calculate mean quality and mean correlation coefficient
av_q  = mean(qs);
av_cc = mean(cc);
end
```

APPENDIX H

RASE Index Function

```
rase_f.m
function [rase_pb av_rase rase_tl] = rase_f(x,y)
% RASE calculator
% 21/08/2009 Version 1.0
% 02/12/2009 Version 1.5    - hyperspectral support
% 05/03/2010 Version 2.0    - Double precision, RAM efficiency
% 08/08/2011 Version 2.2F   - Function Version
% Author: Aristidis D. Vaiopoulos
% RMSE part
% Find the number of bands
sizi = size(x);
if max(size(size(x))) == 2
   bands = 1;
else
   bands = sizi(1,3);
end
nres = sizi(1,1)*sizi(1,2);
% Preallocation
rmses = zeros(bands,1);
Ms   = zeros(bands,1);
for i = 1:bands
   xt = double(x(:,:,i));
   yt = double(y(:,:,i));
   rmses(i) = sqrt((sum(sum((xt - yt).^2)))/nres);
   % Mean xs for RASE
   Ms(i)      = mean2(x(:,:,i));
end
% End of RMSE part
% RASE part
rmsesquared = rmses.^2;
srmsesq     = sum(rmsesquared);
M           = mean(x(:));
% Total RASE
rase_tl     = (100/M)*(sqrt(srmsesq/bands));
% RASE per band
rase_pb     = (100./Ms).*sqrt(rmsesquared);
% Average RASE
av_rase     = mean(rase_pb);
% End of RASE part
End
```

APPENDIX I

RMSE Index Function

rmse_f.m

```
function [rmses, av_rmse] = rmse_f(x,y)
% RMSE calculator
% 02/12/2009 Version 1.5    - Hyperspectral support
% 05/03/2010 Version 2.0    - Double precision, RAM efficiency
% 25/06/2010 Version 2.2    - Excel Output option
% 08/08/2011 Version 2.4F   - Function Version
% Author: Aristidis D. Vaiopoulos
sizi = size(x);
if max(size(size(x))) == 2
   bands = 1;
else
   bands = sizi(1,3);
end
nres = sizi(1,1)*sizi(1,2);
rmses = zeros(bands,1);
for i = 1:bands
   xt = double(x(:,:,i));
   yt = double(y(:,:,i));
   rmses(i) = sqrt( (sum(sum((xt - yt).^2)))/nres );
end
av_rmse = mean(rmses);
end
```

APPENDIX J

Correlation Coefficient Index Function

```
function [cc av_cc] = ccc_f(x,y)
% Correlation Coefficient Calculator
% 07/03/2010 Version 1.0
% 25/06/2010 Version 1.2    - Excel Output option
% 04/08/2011 Version 1.2F   - Function Version
% Author: Aristidis D. Vaiopoulos
% Find the number of bands
bands = size(x);
if length(bands) == 3
   bands = bands(1,3);
else
   bands = 1;
end
% Preallocation
cc = zeros(bands,1);
% Correlation Coefficient calculation
for i = 1:bands
   xt = double(x(:,:,i));
   yt = double(y(:,:,i));
   cc(i) = corr2(xt,yt);
end
% Average CC
av_cc = mean(cc);
end
```

APPENDIX K

Progress Bar Matlab Script

progressbar.m

```
function progressbar(varargin)
% Description:
%   progressbar() provides an indication of the progress of some task using
% graphics and text. Calling progressbar repeatedly will update the figure and
% automatically estimate the amount of time remaining.
%   This implementation of progressbar is intended to be extremely simple to use
% while providing a high quality user experience.
%
% Features:
%   - Can add progressbar to existing m-files with a single line of code.
%   - Supports multiple bars in one figure to show progress of nested loops.
%   - Optional labels on bars.
%   - Figure closes automatically when task is complete.
%   - Only one figure can exist so old figures don't clutter the desktop.
%   - Remaining time estimate is accurate even if the figure gets closed.
%   - Minimal execution time. Won't slow down code.
%   - Randomized color. When a programmer gets bored...
%
% Example Function Calls For Single Bar Usage:
%   progressbar               % Initialize/reset
%   progressbar(0)            % Initialize/reset
%   progressbar('Label')      % Initialize/reset and label the bar
%   progressbar(0.5)          % Update
%   progressbar(1)            % Close
%
% Example Function Calls For Multi Bar Usage:
%   progressbar(0, 0)         % Initialize/reset two bars
%   progressbar('A', '')      % Initialize/reset two bars with one label
%   progressbar('', 'B')      % Initialize/reset two bars with one label
%   progressbar('A', 'B')     % Initialize/reset two bars with two labels
%   progressbar(0.3)          % Update 1st bar
%   progressbar(0.3, [])      % Update 1st bar
%   progressbar([], 0.3)      % Update 2nd bar
%   progressbar(0.7, 0.9)     % Update both bars
%   progressbar(1)            % Close
%   progressbar(1, [])        % Close
%   progressbar(1, 0.4)       % Close
%
% Notes:
%   For best results, call progressbar with all zero (or all string) inputs
% before any processing. This sets the proper starting time reference to
% calculate time remaining.
%   Bar color is choosen randomly when the figure is created or reset. Clicking
% the bar will cause a random color change.
%
```

```
% Demos:
%    % Single bar
%    m = 500;
%    progressbar % Init single bar
%    for i = 1:m
%      pause(0.01) % Do something important
%      progressbar(i/m) % Update progress bar
%    end
%
%    % Simple multi bar (update one bar at a time)
%    m = 4;
%    n = 3;
%    p = 100;
%    progressbar(0,0,0) % Init 3 bars
%    for i = 1:m
%        progressbar([],0) % Reset 2nd bar
%        for j = 1:n
%            progressbar([],[],0) % Reset 3rd bar
%            for k = 1:p
%                pause(0.01) % Do something important
%                progressbar([],[],k/p) % Update 3rd bar
%            end
%            progressbar([],j/n) % Update 2nd bar
%        end
%        progressbar(i/m) % Update 1st bar
%    end
%
%    % Fancy multi bar (use labels and update all bars at once)
%    m = 4;
%    n = 3;
%    p = 100;
%    progressbar('Monte Carlo Trials','Simulation','Component') % Init 3 bars
%    for i = 1:m
%        for j = 1:n
%            for k = 1:p
%                pause(0.01) % Do something important
%                % Update all bars
%                frac3 = k/p;
%                frac2 = ((j-1) + frac3) / n;
%                frac1 = ((i-1) + frac2) / m;
%                progressbar(frac1, frac2, frac3)
%            end
%        end
%    end
%
% Author:
%   Steve Hoelzer
%
% Revisions:
% 2002-Feb-27   Created function
% 2002-Mar-19   Updated title text order
```

```
% 2002-Apr-11   Use floor instead of round for percentdone
% 2002-Jun-06   Updated for speed using patch (Thanks to waitbar.m)
% 2002-Jun-19   Choose random patch color when a new figure is created
% 2002-Jun-24   Click on bar or axes to choose new random color
% 2002-Jun-27   Calc time left, reset progress bar when fractiondone == 0
% 2002-Jun-28   Remove extraText var, add position var
% 2002-Jul-18   fractiondone input is optional
% 2002-Jul-19   Allow position to specify screen coordinates
% 2002-Jul-22   Clear vars used in color change callback routine
% 2002-Jul-29   Position input is always specified in pixels
% 2002-Sep-09   Change order of title bar text
% 2003-Jun-13   Change 'min' to 'm' because of built in function 'min'
% 2003-Sep-08   Use callback for changing color instead of string
% 2003-Sep-10   Use persistent vars for speed, modify titlebarstr
% 2003-Sep-25   Correct titlebarstr for 0% case
% 2003-Nov-25   Clear all persistent vars when percentdone = 100
% 2004-Jan-22   Cleaner reset process, don't create figure if percentdone = 100
% 2004-Jan-27   Handle incorrect position input
% 2004-Feb-16   Minimum time interval between updates
% 2004-Apr-01   Cleaner process of enforcing minimum time interval
% 2004-Oct-08   Seperate function for timeleftstr, expand to include days
% 2004-Oct-20   Efficient if-else structure for sec2timestr
% 2006-Sep-11   Width is a multiple of height (don't stretch on widescreens)
% 2010-Sep-21   Major overhaul to support multiple bars and add labels
%
persistent progfig progdata lastupdate
% Get inputs
if nargin > 0
   input = varargin;
   ninput = nargin;
else
   % If no inputs, init with a single bar
   input = {0};
   ninput = 1;
end
% If task completed, close figure and clear vars, then exit
if input{1} == 1
   if ishandle(progfig)
      delete(progfig) % Close progress bar
   end
   clear progfig progdata lastupdate % Clear persistent vars
   drawnow
   return
end
% Init reset flag
resetflag = false;
% Set reset flag if first input is a string
if ischar(input{1})
   resetflag = true;
end
% Set reset flag if all inputs are zero
```

```
if input{1} == 0
   % If the quick check above passes, need to check all inputs
   if all([input{:}] == 0) && (length([input{:}]) == ninput)
      resetflag = true;
   end
end
% Set reset flag if more inputs than bars
if ninput > length(progdata)
   resetflag = true;
end
% If reset needed, close figure and forget old data
if resetflag
   if ishandle(progfig)
      delete(progfig) % Close progress bar
   end
   progfig = [];
   progdata = []; % Forget obsolete data
end
% Create new progress bar if needed
if ishandle(progfig)
else % This strange if-else works when progfig is empty (~ishandle() does not)

   % Define figure size and axes padding for the single bar case
   height = 0.03;
   width = height * 8;
   hpad = 0.02;
   vpad = 0.25;

   % Figure out how many bars to draw
   nbars = max(ninput, length(progdata));

   % Adjust figure size and axes padding for number of bars
   heightfactor = (1 - vpad) * nbars + vpad;
   height = height * heightfactor;
   vpad = vpad / heightfactor;

   % Initialize progress bar figure
   left = (1 - width) / 2;
   bottom = (1 - height) / 2;
   progfig = figure(...
      'Units', 'normalized',...
      'Position', [left bottom width height],...
      'NumberTitle', 'off',...
      'Resize', 'off',...
      'MenuBar', 'none' );

   % Initialize axes, patch, and text for each bar
   left = hpad;
   width = 1 - 2*hpad;
   vpadtotal = vpad * (nbars + 1);
   height = (1 - vpadtotal) / nbars;
```

```
    for ndx = 1:nbars
        % Create axes, patch, and text
        bottom = vpad + (vpad + height) * (nbars - ndx);
        progdata(ndx).progaxes = axes( ...
            'Position', [left bottom width height], ...
            'XLim', [0 1], ...
            'YLim', [0 1], ...
            'Box', 'on', ...
            'ytick', [], ...
            'xtick', [] );
        progdata(ndx).progpatch = patch( ...
            'XData', [0 0 0 0], ...
            'YData', [0 0 1 1] );
        progdata(ndx).progtext = text(0.99, 0.5, '', ...
            'HorizontalAlignment', 'Right', ...
            'FontUnits', 'Normalized', ...
            'FontSize', 0.7 );
        progdata(ndx).proglabel = text(0.01, 0.5, '', ...
            'HorizontalAlignment', 'Left', ...
            'FontUnits', 'Normalized', ...
            'FontSize', 0.7 );
        if ischar(input{ndx})
            set(progdata(ndx).proglabel, 'String', input{ndx})
            input{ndx} = 0;
        end

        % Set callbacks to change color on mouse click
        set(progdata(ndx).progaxes, 'ButtonDownFcn', {@changecolor, progdata(ndx).progpatch})
        set(progdata(ndx).progpatch, 'ButtonDownFcn', {@changecolor, progdata(ndx).progpatch})
        set(progdata(ndx).progtext, 'ButtonDownFcn', {@changecolor, progdata(ndx).progpatch})
        set(progdata(ndx).proglabel, 'ButtonDownFcn', {@changecolor, progdata(ndx).progpatch})

        % Pick a random color for this patch
        changecolor([], [], progdata(ndx).progpatch)

        % Set starting time reference
        if ~isfield(progdata(ndx), 'starttime') || isempty(progdata(ndx).starttime)
            progdata(ndx).starttime = clock;
        end
    end

    % Set time of last update to ensure a redraw
    lastupdate = clock - 1;

end
% Process inputs and update state of progdata
for ndx = 1:ninput
    if ~isempty(input{ndx})
        progdata(ndx).fractiondone = input{ndx};
        progdata(ndx).clock = clock;
    end
```

```
end
% Enforce a minimum time interval between graphics updates
myclock = clock;
if abs(myclock(6) - lastupdate(6)) < 0.01 % Could use etime() but this is faster
   return
end
% Update progress patch
for ndx = 1:length(progdata)
   set(progdata(ndx).progpatch, 'XData', ...
      [0, progdata(ndx).fractiondone, progdata(ndx).fractiondone, 0])
end
% Update progress text if there is more than one bar
if length(progdata) > 1
   for ndx = 1:length(progdata)
      set(progdata(ndx).progtext, 'String', ...
         sprintf('%1d%%', floor(100*progdata(ndx).fractiondone)))
   end
end
% Update progress figure title bar
if progdata(1).fractiondone > 0
   runtime = etime(progdata(1).clock, progdata(1).starttime);
   timeleft = runtime / progdata(1).fractiondone - runtime;
   timeleftstr = sec2timestr(timeleft);
   titlebarstr = sprintf('%2d%%    %s remaining', ...
      floor(100*progdata(1).fractiondone), timeleftstr);
else
   titlebarstr = ' 0%';
end
set(progfig, 'Name', titlebarstr)
% Force redraw to show changes
drawnow
% Record time of this update
lastupdate = clock;
% ----------------------------------------------------------------------------
function changecolor(h, e, progpatch) %#ok<INUSL>
% Change the color of the progress bar patch
% Prevent color from being too dark or too light
colormin = 1.5;
colormax = 2.8;
thiscolor = rand(1, 3);
while (sum(thiscolor) < colormin) || (sum(thiscolor) > colormax)
   thiscolor = rand(1, 3);
end
set(progpatch, 'FaceColor', thiscolor)
% ----------------------------------------------------------------------------
function timestr = sec2timestr(sec)
% Convert a time measurement from seconds into a human readable string.
% Convert seconds to other units
w = floor(sec/604800); % Weeks
sec = sec - w*604800;
d = floor(sec/86400); % Days
```

```
sec = sec - d*86400;
h = floor(sec/3600); % Hours
sec = sec - h*3600;
m = floor(sec/60); % Minutes
sec = sec - m*60;
s = floor(sec); % Seconds
% Create time string
if w > 0
   if w > 9
      timestr = sprintf('%d week', w);
   else
      timestr = sprintf('%d week, %d day', w, d);
   end
elseif d > 0
   if d > 9
      timestr = sprintf('%d day', d);
   else
      timestr = sprintf('%d day, %d hr', d, h);
   end
elseif h > 0
   if h > 9
      timestr = sprintf('%d hr', h);
   else
      timestr = sprintf('%d hr, %d min', h, m);
   end
elseif m > 0
   if m > 9
      timestr = sprintf('%d min', m);
   else
      timestr = sprintf('%d min, %d sec', m, s);
   end
else
   timestr = sprintf('%d sec', s);
end
```