AN INTERACTIVE COMPUTER GRAPHICS TOOL FOR VISUALIZATION OF

THREE-DIMENSIONAL PHOTOELASTIC STRESS FROZEN MODELS:

A STUDY OF TOTAL SHOULDER ARTHROPLASTY

GLENOID COMPONENTS

by

Matthew Ervin Harrow

A thesis submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Master of Science in the
Department of Computer Science

Charlotte
1999

Approved by:

_____

Dr. Kalpathi Subramanian

_____
_
Dr. Richard Peindl

_____

Dr. Taghi Mostafavi

ABSTRACT

MATTHEW ERVIN HARROW. An interactive computer graphics tool for visualization of three-dimensional photoelastic stress frozen models: A study of total shoulder arthroplasty glenoid components. (Under direction of DR. K.R. SUBRAMANIAN)

Photoelastic analysis of three dimensional models have historically been analyzed by complex combinations of software packages and computer platforms requiring experienced operator input. The purpose of this thesis was to utilize modern data representations, interactive computer graphics, and data visualization to create a tool for automating and accelerating the analysis of three-dimensional photoelasticity. Ideally, one program would be used for capturing stress patterns viewed through the polariscope, through data analysis, to solid model rendering of results, working in an inexpensive PC platform. Work presented here compares stress patterns observed in photoelastic models testing glenoid components of two types of total shoulder replacement systems (the 22mm Kirshner Neer system and a new 3-pegged Bio-modular total shoulder replacement system). Results demonstrate the completion of most functions for automating the analysis process. Procedures have been developed for image acquisition, calibration, contouring, translations, solid modeling, calculating effective stress, and model viewing. Additional work needs to be conducted in the resizing of data and calibration for complete analysis of glenoid component stress patterns, however preliminary comparisons are available. Viewing planar stress patterns along a single axis of the photoelastic models reveals the stress shielding pattern of the Kirshner Neer keel design. Stresses of the peg design are more distributed across the plateau of the implant near the joint surface, while the keel design transmits stresses through the tip of the keel farther into the bone.

ACKNOWLEDGMENTS

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

Total shoulder replacement surgery is becoming a common solution for patients

who suffer from degenerative arthritis of the shoulder joint. Orthopaedic devices are

inserted into the bone of the upper arm (humeral head) and the shoulder blade (glenoid

region of the scapula) to reconstruct the joint space providing pain relief and increased

range of motion(see Figures 2.1 and 2.2). Of particular interest is the development of

new devices that can be implanted into the glenoid and their associated failure

mechanisms.

The mechanisms for glenoid component loosening are not clearly defined but are

attributed to the anatomic and biomechanical factors that are unique to the glenoid. There

continues to be an evolution of glenoid component design focused on reducing

component loosening from the bone. Better understanding of the relationship between

glenoid component design and global stress distribution patterns in the remaining glenoid

bone should allow optimization of individual component designs with respect to the

minimization or dispersion of local stress regions in the glenoid. Three-dimensional

stress analysis data for total shoulder replacement (TSR) systems should also provide

useful information with regard to which component designs are best suited for specific

patterns of glenoid bone stock insufficiency as in posterior arthritic degeneration.

Judicious component selection and/or subtle improvements in component designs may

decrease the incidence of glenoid component loosening. Also, the technical difficulties

associated with

the reproducibility of glenoid component insertion may be reduced through modifications of installation instrumentation.

Photoelastic analysis is a method of two and three dimensional full field stress analysis that can be applied to highly complex statically or dynamically loaded models. The benefit of the photoelastic technique is full field analysis, allowing an entire region to be analyzed as opposed to the point-by-point measurement of other strain measuring devices. Stress-freezing is a technique for accurately measuring internal strains in a three-dimensional photoelastic model to determine the stresses during static testing. The stress-freezing technique using photoelastic materials has an established history of successful applications in virtually every field of manufacturing and construction where stress analysis is employed[5]. As the photoelastic materials are available in resin-based liquid form, virtually any prototype component shape (e.g. a human scapula) can be molded/cast. Often the option of using the photoelastic stress freezing method is overlooked because of the time required for developing a model and analyzing the data.

Automating the photoelastic analysis process has been a goal of many authors for nearly 50 years. Automated polariscopes are now available for fast point-by-point measurements of two-dimensional photoelastic models. With methods of automating the point measurements of full field photoelastic analysis almost complete the next step is to extend the analysis from the planar 2D into 3D volumes. A variety of image processing techniques have been used to automate the processing of light transmitted through a photoelastic model, however, very little has been done in the area of modern data representation and computer graphics for analysis of photoelastic modeling. Previous

methods of three-dimensional photoelastic analysis are filled with tedious time consuming tasks that can be minimized by developing modern hardware and software solutions. Similar to other three-dimensional analysis techniques like finite element modeling, photoelastic analysis requires the need for specialized high speed Unix based computers for collecting and analyzing data with independently developed programs for each analysis. If three-dimensional viewing is desired, analyzed data then has to be transferred to another program such as IDEAS or IBM's Data Explorer, for solid modeling and viewing. Difficulties involved with the analysis technique compounded with the problem of having to change data formats for multiple specialized software programs being run on specialized machines makes the analysis process more complicated than the initial testing. Even today authors opt to minimize their analysis by generating output of 3D models in the form of planar 2D views[6, 20] instead of a volumetric analysis.

The major goal of this project is to utilize a photoplastic model for the analysis and comparison of the three-dimensional stress patterns for two different glenoid component system designs subjected to clinically relevant loading conditions. The components tested are from the 22mm Kirshner Neer keel system and a new 3-pegged Bio-modular total shoulder replacement systems shown in Figure 1.1.

The data obtained can also be used as a base of information for comparative analysis involving component design alterations or instrumentation modifications. The specific goal of this thesis is the development of a PC based automated three-dimensional photoelastic analysis package to process photoelastic slices and create 3D geometric models for comparison of stress patterns in a simple computational environment. Such a

package would minimize the cost of computers, software, repetitive data transfers, and

the need to be proficient in the use of multiple software packages.
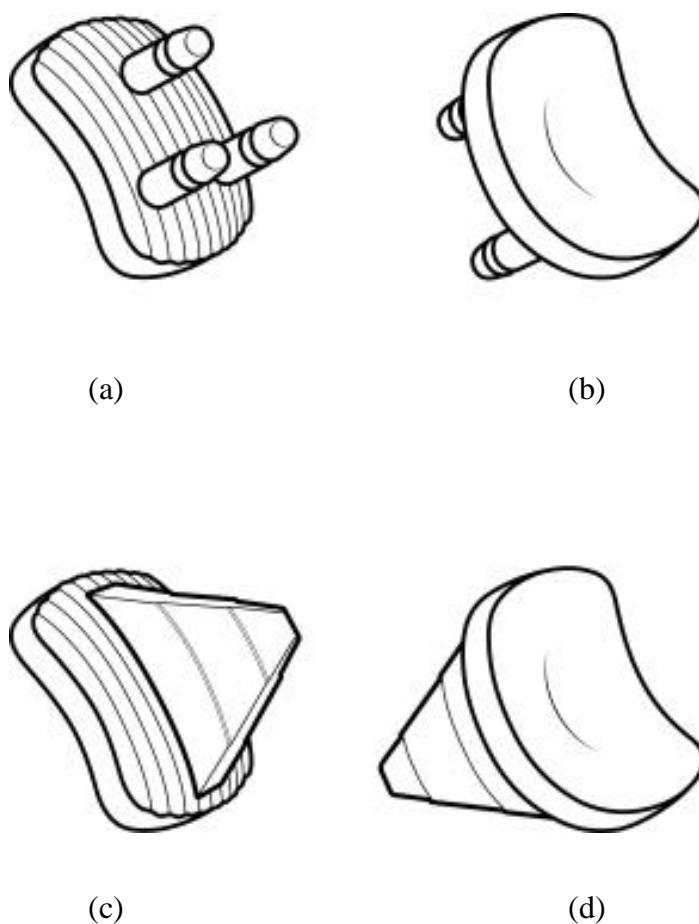
Figure 1.1: Glenoid components tested.
(a) and (b) New 3-pegged Bio-modular system
(c) and (d) Kirshner Neer keel system

1.1 Chapter Overview

The remaining text will be discussed in five chapters. Chapter 2 reviews pertinent

background information relating to total shoulder replacement surgery, photoelasticity,

and computer graphics techniques. Chapter 3 covers the materials and methods of

creating and testing a three-dimensional photoelastic model. In Chapter 4, a description

of the general functions of the automated photoelastic analysis program is provided along

with several of the processing difficulties. Results of the analysis process are provided in

Chapter 5, and a brief discussion of related work and future testing is contained in

Chapter 6. Finally, a more detailed description of the coding procedures developed for

this project is given in Appendix A.

CHAPTER 2: BACKGROUND

2.1 Anatomy

　　The shoulder is created by the junction of the clavicle, scapula, and humerus with

the scapula-humerus joint providing the majority of the motion (Figure 2.1). Numerous

soft tissues attached throughout these structures produce stability and motion, with

stability of the glenohumeral joint depending on central contact between the humeral

head and the glenoid fossa.
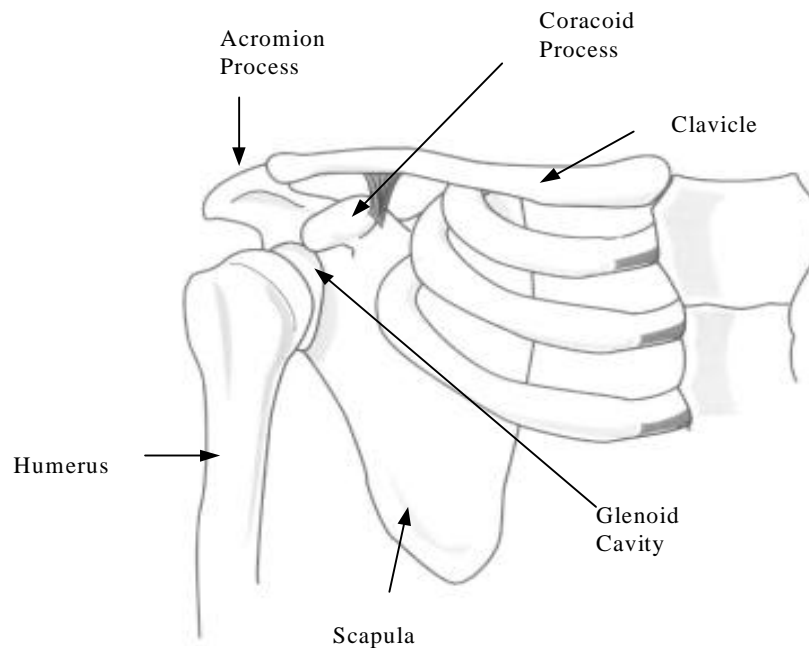
Figure 2.1: Anatomy of the shoulder.

Osteoarthritis, osteonecrosis, and rheumatoid arthritis cause deterioration of bone and

cartilage of joints throughout the body. Symptoms of this deterioration include pain,

recurrent dislocation, and limited motion. Treatment options vary greatly depending upon location of the joint, severity, soft tissues damage, and patient. In the shoulder, cases involving destruction of both the humeral head and glenoid cavity with minimal soft tissue damage the recommended treatment is total shoulder replacement surgery.

Today the most popular prosthesis for total shoulder replacement surgery is the Neer unconstrained design[4] (Figure 2.2). The design features a metalic stem inserted into the proximal humeral with minimal bone removal and a polyethylene component inserted into the glenoid region of the scapula. Components are matched in size and radius of curvature allowing duplication of the normal anatomy of the joint.
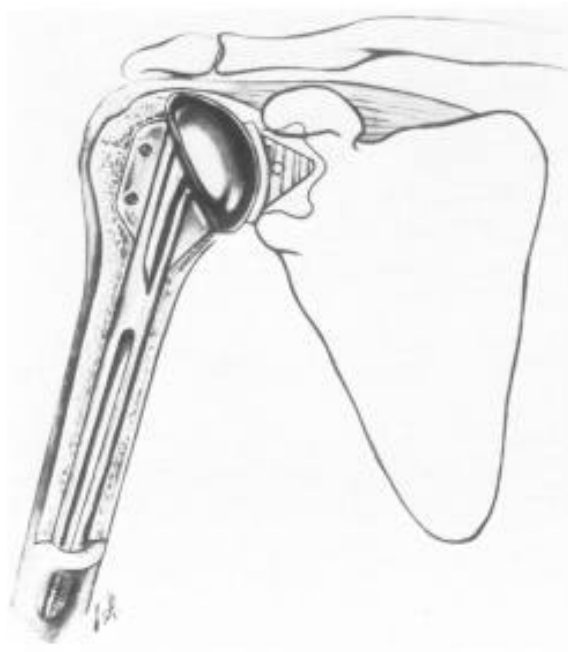
Figure 2.2: Total shoulder replacement prosthetics.

2.2 Photoelastic Theory

2.2.1 Photoelastic model

Photoelasticity involves analyzing the polarized light that is transmitted through a stressed model of polymeric material. Deformation of the polymeric material causes changes of the reflection and refraction of the light waves proportionally to the magnitude and direction of applied load. For a two dimensional model the applied load can be separated into two orthogonal principle stresses and a shear stress allowing a relationship to be made between the stress magnitude and the index of refraction or retardation. This relationship has been defined as:

$$\Delta = \Delta_2 - \Delta_1 = \frac{2\pi hc}{\lambda}(\sigma_1 - \sigma_2)$$

where Δ = relative retardation, h = thickness of the model, c = stress optic coefficient (constant for material determined by calibration), λ = wavelength of light being used, σ_1, σ_2 = orthogonal principal stresses.

Placing a term for the relative retardation of a complete cycle of retardation

(fringe value) $$N = \frac{\Delta}{2\pi}$$

then , $$(\sigma_1 - \sigma_2) = \frac{N\lambda}{hc}$$

so the principal stress difference, $(\sigma_1 - \sigma_2)$, can be solved for if N is known (an example of fringe patterns is shown in Figure 2.3a) . Utilizing a polariscope a variety of experimental techniques can be employed to determine a relative retardation value, N, that can be used to separate and determine the entire state of stress at every location

within the model. (Note: Further references to 'stress' in this paper will refer to the

principal stress difference unless otherwise specified.)

2.2.2 The Polariscope

2.2.2.1 Plane Polariscope

     A plane polariscope setup utilizes two plane polarized lenses, the polarizer and

the analyzer, arranged so the axis's of polarization are perpendicular. As light enters the

polarizer it can be referred to as $E = a\cos(\omega t)$ where $\omega = 2\pi f$ is the circular frequency of

the light wave. Light parallel with the polarizer axis will be transmitted as

$E_{pt} = a\cos(\omega t)\cos(\alpha)$ and light perpendicular to the axis will be reflected as

$E_{pr} = a\cos(\omega t)\sin(\alpha)$ where α is the angle from the polarizing axis to the major axis of the

original light. A stressed model acts like a wave plate, as light passes through the model

it propagates at velocities proportional to the principle strains in orthogonal directions.

$$E_1 = k\cos\alpha\cos(\omega t - \Delta_1) \qquad\qquad E_2 = k\sin\alpha\cos(\omega t - \Delta_2)$$

Light then entering the analyzer will be transmitted as

$$E_{at} = E_2' - E_1' = E_2\cos(\alpha) - E_1\sin(\alpha)$$

$$= k\sin 2\alpha\sin\frac{\Delta_2 - \Delta_1}{2}\sin\left(\omega t - \frac{\Delta_2 + \Delta_1}{2}\right)$$

Knowing that the intensity I of light is proportional to the square of the amplitude, and

that the time dependent term will average to a constant the light emerging from the

analyzer can then be referred to as

$$I = K\sin^2 2\alpha\sin^2\frac{\Delta}{2}$$

Note that the intensity will be zero when $\sin^2 2\alpha = 0$ or when $\sin^2(\Delta/2) = 0$. Therefore

intensity is dependent upon both the direction (α) and difference(Δ) of the principle

stresses. Light intensities patterns viewed through a plane polariscope are referred to as

isoclinics. Black regions in the output image identify principal stress directions that are

aligned with the axis of the analyzer. By rotating the analyzer through 180° the direction

of the principal stresses at every point in the photoelastic model can be determined.

2.2.2.2 Circular Polariscope

To eliminate the effect of principle stress direction on the output intensity two

additional lenses are added to the system making a circular polariscope. The additional

lenses are birefringent wave plates (or quarter wave plates) are aligned with their fast

axis 90 degrees offset and placed directly in front of and behind the model within a plane

polariscope setup.

The polariscope used is a circular polariscope (Measurements Group, Raleigh,

NC) in a dark field setup with a white light source. Without discussing in great detail, the

intensity of the light transmitted through a circular polariscope and a photoelastic model

will be proportional to the principal stress difference in the model (the isochromatic

fringe pattern). The exact equation is

$$I_i = \frac{1}{\lambda_{i2} - \lambda_{i1}} \int_{\lambda_{i1}}^{\lambda_{i2}} F_i(\lambda) I_0(\lambda) \cdot \sin^2\left(\frac{\pi \Delta C_\lambda}{\lambda C_0}\right)(1 - \cos^2 2\alpha \sin^2 \varepsilon) d\lambda \qquad \text{Equation 2.1}$$

where,

i = r, g, b
λ_{i1} and λ_{i2} = the lower and upper limits of the light source spectrum as it is
            acquired by the filter i

$F_i (i = \text{r, g, b})$ = the spectral responses of the red, green, and blue filters of the camera

I_0 = white light source intensity

C_λ / C_0 = the term which accounts for the dispersion of birefringence, being

C_λ and C_0 the stress optic coefficients at arbitrary wavelength λ and at

the reference wavelength λ_0 respectively

α = the isoclinic angle, the angle between the principle stresses

ε = the error of the retardation of the quarter wave plates

Note that the output is dependent upon the material used in the model, the light source, and the recording device. The correlation between RGB intensity and stress value can be more easily understood by plotting the values as shown in Figure 2.3.

(a)

(b)
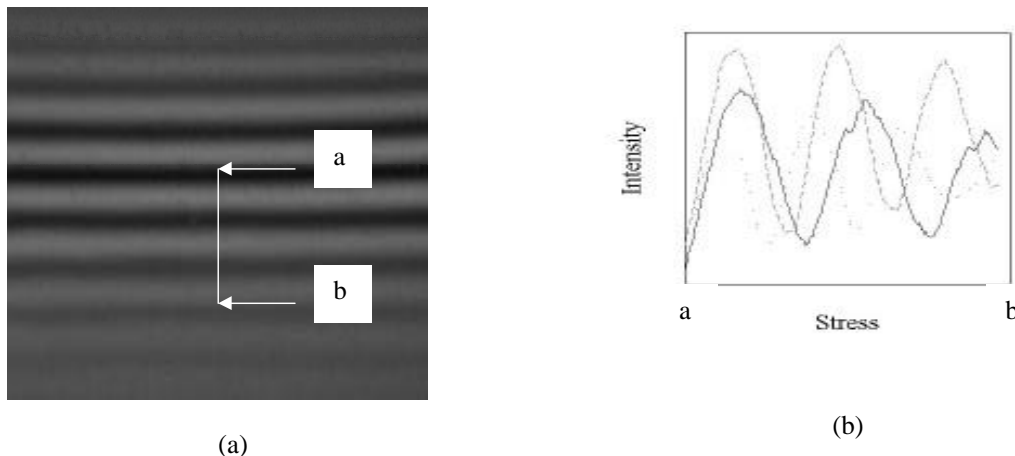
Figure 2.3: Center view of a calibration beam subjected to four-point bending with the top of the beam in compression and the bottom in tension(a), and associated stress intensity plot produced from line a-b (b).

The zero stressed region of any photoelastic material viewed with a light source can easily be identified as a dark black region immediately surrounded by white regions. Color bands are created from the wavelength dependent retardations, caused by increased deformation of the model. By following the change in intensity of each of the primary colors moving from a zero stress region (point a) to a high stress region (point b) it can

be seen that the intensity of each color varies sinusoidally with increasing stress. Also notice how the increasing the deformation creates a different frequency for each of the primary colors. The amplitude of the sinusoid will be shown to be dependent upon the modeling material, the surface texture, distance from the camera to the model, and the imaging board later in the text.

2.2.3 Stress freezing method

Stress freezing is a method of locking internal stresses in a 3D photoelastic model to analyze interior planes of the model. The material used in stress freezing is a diphase polymer. During polymerization a lattice of long primary bonds are formed first followed by shorter secondary bonds that will form between the primaries providing additional support. At room temperature both molecular bonds will act to resist the applied load. When the temperature of the polymer is raised to a critical temperature the secondary bonds will break leaving the primary bonds to support any load. The model will deform elastically by elongating the unsupported primary bonds. Then upon slow cooling the secondary bonds will reform locking the primaries into their deformed position. Stress frozen models may then be sliced without disturbing the locked-in stress patterns of the molecular bonds. Optical properties relating to deformed primary bonds of the model slices can then be analyzed to determine the internal stress patterns.

2.2.4 Effective Stresses

It is possible to determine the exact state of stress $\left(\sigma_x, \sigma_y, \sigma_z, \tau_{xy}, \tau_{xz}, \tau_{yz}\right)$ for every point within a photoelastic model, however the added expense of labor and extreme care and skill for such an analysis may not be required. Individual stress information may not be as useful in predicting failure in an object as a calculation of a

combined stress for each point would be. An alternative is to determine an effective (von Mises) stress that will provide indication of plastic yielding or possible fatigue failure. Principle stresses for a point can be determined from one view of the normal incidence of each of three mutually orthogonal directions of the isochromatic fringes (difference in principle stresses) and six views isoclinic fringes (15° rotations of the polariscope analyzer providing the direction of the principle stresses). The principle stresses can then be used to calculate the effective stress by

$$\sigma_E = \frac{1}{2}\left((\sigma_x - \sigma_y)^2 + (\sigma_y - \sigma_z)^2 + (\sigma_z - \sigma_x)^2\right)^{0.5} \qquad \text{Equation 2.2}$$

where,

σ_E = the effective stress

$\sigma_x, \sigma_y, \sigma_z$ = the principle stress values at a point viewed normal to the x, y, and z axis respectively.

Determining stress within the model requires obtaining the seven images from three orthogonal directions for each point in the model. This can be accomplished by slicing the model into sections, then slicing the sections into strips, and finally slicing the strips into small cubes, and viewing each cube from three directions. However, there is less loss of information and faster processing time when three models are used with each being sectioned along a separate axis. Working with cross-sectional views limits data collection to the accuracy to the image capturing equipment instead of the width of the saw blade.

2.2.5 Automated photoelasticity

Automated photoelasticity first began in the 1950's when Brown and Hickson[2] and Focht et al.[10] first attached a photoelectric photometer to a polariscope to measure intensity of the transmitted light through a mobile aperture. Zandman[61] attached a computer controlled servo motor to a Babinet compensator to automate the point-by-point measurements of the isoclinic parameter and isochromatic fringe order in 1959. The first attempt to use image processing for full-field analysis began in 1979 when two groups Mueller and Saackel[31], and Seguchi[48] used a video camera, image processor, and a computer to analyze grey scaled images and detect centerlines of fringes. From that point many authors worked on automated isoclinic fringe detection, ordering, and fringe multiplication to increase the accuracy of the analysis [15, 27, 31, 38, 40, 41,48,50, 59, 60]. In 1983 the idea of relating the stress magnitude to the intensity of the light transmitted through a photoelastic model began with half-fringe photoplasty (HFP) being detected by Voloshin and Burger[53]. The basis of Volshins method is the sinusoidal intensity of the transmitted light emerging from a circular polariscope darkfield. By limiting the maximum fringe order transmitted through the photoelastic model to ½ fringe a direct correlation can be made between light intensity and the fringe order. The benefit of this method is that it is a full field technique which requires no operator input. In 1991 Patterson and Wang[35] developed an automated polariscope that determined the entire state of stress for a single photoelastic slice by using a phase stepping method. The phase stepping method requires changing the light source intensity by rotation of the optical elements of the polarizer and monitoring the change in phase of the transmitted light. The drawback of this method is it still requires the entire state of stress at one point to be known. In the mid-90's

Ajovalasit et al.[1] and Ramesh et al.[37] improved on a concept of Voloshin and Berger to develop a system of full field analysis by using a white light source and analyzing the red, green, and blue (RGB) components of the polariscope output. Similar to HFP analysis, each method involves creating a calibration table that maps the RGB intensity variations to the relative retardation or isochromatic value. From this method principle strain differences can be determined quickly for every point in an image. Accuracy is limited to the size of the image recorded and can only be used for the first three to four fringes due to the attenuation of the RGB levels at higher retardations.

2.3 Volume Rendering

Scientific visualization is a technique of analyzing multivalued and multidimensional data to provide scientists with insight of complex phenomena. Volume visualization is the most active subset of scientific visualization providing the process of projecting multidimensional data onto a two dimensional image plane demonstrating the structures contained within the volumetric data. Methods of volume visualization must be selected based on the input data and the desired output. Ideally output must be understandable, easy to manipulate, and fast to render. Increasing size and complexity of data to be analyzed challenges hardware and software limitations. Compromises must be made, concerning the data characteristics, volume characteristics, and processing techniques, for the user to achieve satisfactory results.

2.3.1 Data characteristics

Data can be in the form of individual scalar values representing a physical measurements, vector values indicating a directional value for each point, or combinations of multiple instances of each. The amount of structure and complexity of

the dataset determines the type of visualization algorithm that needs be applied to generate the most informative results. Scalar values are generally acquired for comparing numerical and theoretical results from a grid-like structure requiring a more static visualization of the entire structure at one time. Vector values however, are generally signifying a time dependent event and are generally rendered one slice at a time in a dynamic or animated visualization.

2.3.2 Volume Characteristics

The organization of the data is key to understanding how areas between data points are treated during the interpolating and resampling steps of the rendering process. Volumes of data can be represented as voxels or cells. Voxels are volumetric elements of constant value. A central point has a defined value, and the area surrounding that point is considered to be of the same value. In a cell structure there is allowance for interpolation of data between two or more grid points. The advantage of the cell structure is that a more accurate approximation of the location of object edges can be made. Both voxels or cells can then be organized along Cartesian or non-Cartesian axis, in the form of tetrahedra, hexahedra, prism, pyramids, or curved surfaces.

2.3.3 Visualization Techniques

There are two types of fundamental visualization techniques direct volume rendering and surface fitting. The direct volume rendering is a high quality rendering process that accounts for not only location, lighting, color, and opacity of an object being viewed, but also material properties and the interaction of other objects contained within a scene. It is CPU intensive requiring the entire dataset to be traversed each time a scene is rendered. The most common example of direct volume rendering is ray casting. There

are several variations of ray casting[24,25,26,51,52], but in the general ray casting method each pixel of the screen plane is traversed casting a ray into the data volume. As the ray passes through the data volume pixel values are changed based upon the objects intersected by the ray.

Surface fitting algorithms speed the rendering process by minimizing the amount of data that is traversed through the creation of constant valued surface primatives or objects. A primitive is defined by selecting a threshold value and searching the dataset connecting adjacent points or contours that are equal to the threshold value. Points above or below the threshold are discarded and will not affect the output image. Disadvantages of surface fitting algorithms include false positive and negative artifacts in the output and the potential for large numbers of primitives slowing the process. Examples of surface connecting algorithms include contour connecting and marching cubes[28,58]. In contour connecting each slice of a volume is contoured individually, then through various methods the contours of adjacent slices are connected. The marching cubes method scans a dataset , four slices at a time, looking at all points comprising a cell cube that straddle a threshold value. Each corner point of the cell is evaluated as to being above, on or below the threshold value. Then based on a lookup table of possible combinations of cell corner values a triangular surface patch is created for that cell. The algorithm then continues through the remaining data creating and connecting the patches as it progresses.

CHAPTER 3: PHOTOELASTIC MODEL PREPARATION

3.1 Material Selection

Accurately modeling the stresses transmitted by glenoid component of a total

shoulder replacement (TSR) system requires the model that must simulate the bone

around the glenoid region of the scapula, the glenoid implant, and a bone cement that is

normally used in conjunction with the implant. Several factors were considered when

selecting the appropriate materials for models. Most importantly the ratio of elastic

moduli of each of the materials must be accounted for. The ratio of elastic moduli of the

modeling materials at 121°C(250°F) should be equivalent to the ratio of the bone, bone

cement, and the glenoid component would have at body temperature 37°C(see Figure

3.1).

TSR System (@ 37°C)	Model Components (@ 121°C)
UHMWPE (E = 1.38 GPa)	Delrin, 100ST (E = 0.20 GPa)
↕ 0.5 : 1	↕ 0.53:1
PMMA (E = 2.75 GPa)	Uni-heat Resin (E = 0.38 GPa)
↕ 8.09 : 1	↕ 8.35:1
Glenoid Bone (E = 0.34 GPa)	PLM-9 Plastic (E = 0.046 GPa)

Figure 3.1: Relative Youngs moduli (compressive) of interacting materials, with
interrelational stiffness ratios indicated.

The photoelastic material selected was PLM-9 by Measurements Group(Raleigh, NC). Delrin 100ST (Dupont) was then selected as the material of choice for the glenoid components based on the elastic moduli published by the manufacturer. Adequate material property data could not be obtained for an epoxy resin to simulate the polymethyl methacrylate bone cement. Several combinations of resin bases (Uniheat, and E-Cast F28 Black Resin) and hardeners (Hardener 118, 203, 215, and an Activator) (supplied by United Resin, Royal Oak, MI) were mixed and tested under compressive loading. Tests were conducted on cylinders of each polymer (0.5in. in diameter by 1in. length) at 250° F using an MTS Bionix 858 Testing System (MTS, Minneapolis, MN) equipped with an environmental chamber (Thermotron Corp., Holland, MI). The elastic moduli of each was calculated and the combination of Uniheat Resin and Hardener 118 were selected.

Delrin glenoid components were manufactured by Biomet(Warsaw, IN) based on the specifications of the 22mm Kirshner Neer system and a new 3-pegged Bio-modular total shoulder replacement systems.

3.2 Creating the molds

A master model (Figure 3.2b) was made from a synthetic scapula (Sawbones Inc.) with the acromion process, coracoid process, and medial edge of the scapula removed to minimize the size of the model, the amount of plastic required, and the complexity of the mold. An epoxy putty was applied to smooth large surface defects, and a plexiglass base was attached to provide flat surfaces for alignment after molding.

(a)   (b)
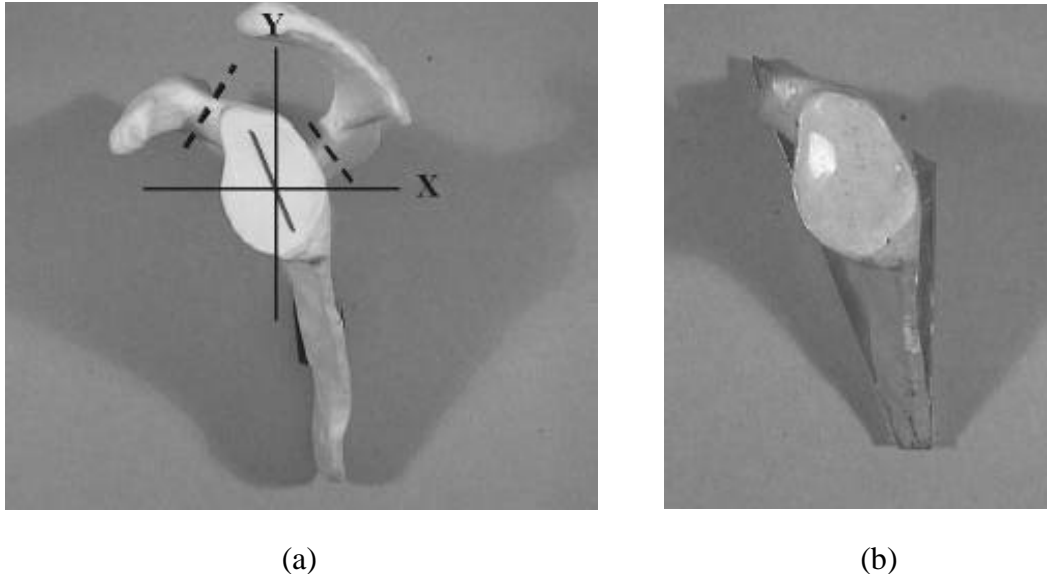
Figure 3.2: (a) Anatomical scapula showing the cutting lines of the acromium and coracoid, orientation of the implant, X and Y planes of sectioning. (b) Master model used to create molded specimens.

Original attempts of creating a mold used the two-part molding material Reprorubber (Flexbar, Islandia, N.Y.). The rubber was mixed and poured into a sheet-metal box having one side of the master, with base attached, pressed down into it. Once the rubber had hardened the process was repeated for the other side. The two sides were then clamped together and holes were cut at the top of the mold (medial edge of the scapula) for pouring of the photoelastic resin.

Though the method seemed correct, but there were several flaws. Inconsistencies in the molding material curing made it unreliable. Large areas of the mold would not harden even after several days of curing. The walls of the box holding the molds were not strong enough producing inconsistent clamping of the boxes and molds of varying thickness. Additionally, it was determined that the base could not be molded with the rest of the model due to the photoelastic shrinkage during cure. The shrinkage only affects

the top of the mold and since the glenoid region is facing down and the base is facing up,

only the base was affected.

A more reliable method of creating molds used Silastic M RTV Silicon Rubber

Kit (Dow Corning, Midland, MI) as the molding material (Figure 3.3). Support for the

molds was provided by two 1/4 inch thick steel c-beams bolted together to create an oval

cylinder. By covering the bottom and suspending the master inside the entire model

could be cast at one time. One of the disadvantages of Silastic M is that it requires an
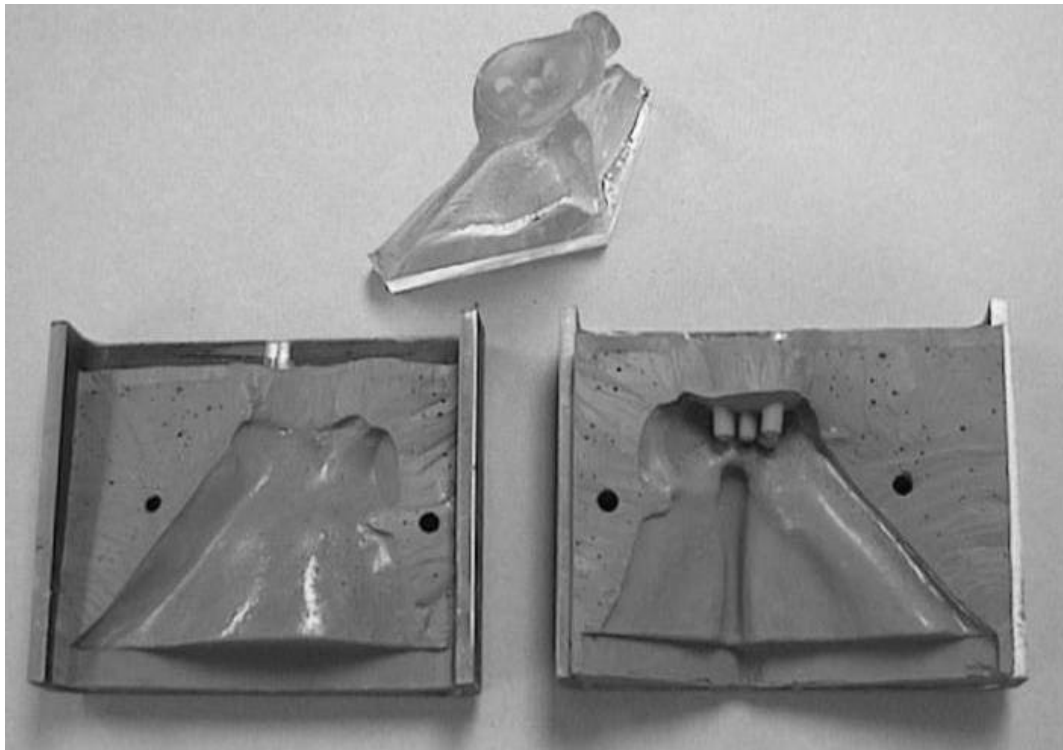
Figure 3.3: (top) Photoelastic model with attached plexiglass base
produced from molding. (bottom) One of three molds used in
creating scapula models for insertion of a pegged component.

additional step of degassing the mixture. After pouring, the molding rubber was placed

vacuum for 10 – 15 minutes to remove excess air and then allowed to harden at room

temperature. To avoid photoelastic shrinkage problems around the base several

plexiglass bases were machined separately. A photoelastic model was then made by placing a plexiglass base (slightly narrower than the master) into the mold and again clamping with the c-beams. Photoelastic resin was poured into the spaces at the ends of the base providing enough extra resin to insure bonding over the entire surface of the base. One of the advantages of this method is that if the glenoid surface needs to be changed (as in changing from a keeled mold to a pegged mold) the fixture could be inverted, and that section of the mold could be removed and repoured without changing the entire mold.

Initial calibration beams were molded from a rectangular aluminum cup, however this method was discarded because the PLM-9 had good adhesion to aluminum and was fracturing during removal from the mold. Calibration beams were then molded from a rubber mold that used a machined aluminum block as the master.

3.3 Molding the photoelastic model

From the master model two identical secondary master plastic models were produced from one mold: one for the keeled component, and one for the pegged component. Each was machined for the implanting of their respective glenoid devices by a surgeon using the same instruments and methods as used in the operating room. Three new rubber molds were then created using one of the machined secondary master models(pegged component configuration shown in Figure 3.3). Testing models were then produced in sets of three along with a calibration beam. Photoelastic resin was prepared as indicated by the instructions (Measurements Group). All molds were coated liberally with mineral oil to prevent adhesion and were preheated to $46^{o}C(115^{0}F)$. During pouring the molds were positioned at a slight incline while the resin was poured slowly to

minimize air pockets. Molds remained at room temperature with additional resin being

added as needed to fill the molds completely for the first 24 hours preventing shrinkage.

After the 48 hour settling time the entire set of molds was placed in the environmental

chamber where the temperature was raised to $43^{\circ}C(110^{0}F)$ and held for a four-hour post

cure. A second-stage post cure raised the temperature to $121^{\circ}C(250^{0}F)$ at $5^{\circ}C(10^{0}F)$ per

hour and held for three hours before cooling at $2^{\circ}C(3^{0}F)$ per hour until secondary bonds

are complete, around $93^{\circ}C(200^{\circ}F)$, then slightly faster returning to room temperature

(Figure 3.4).
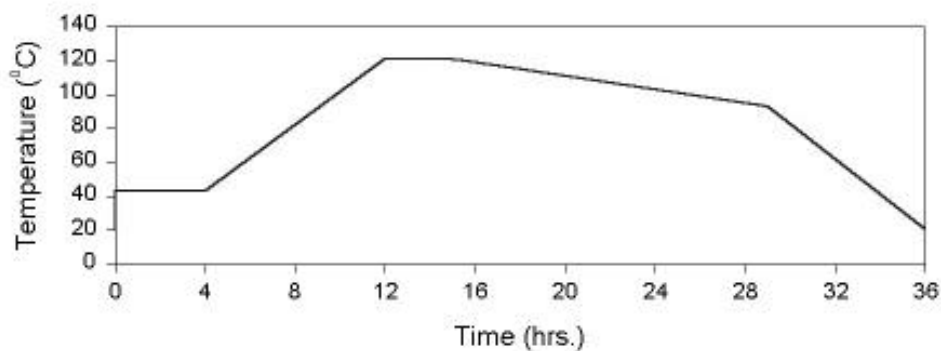
Figure 3.4: Cure cycle for photoelastic material.

3.4 Loading mechanisms

During testing the models were placed in a three-point (tripod) loading system,

developed by Derrek Johnson in his Master's Project[17], to distribute loads evenly to the

three models (Figure 3.5). The tripod system consisted of two central pivoting plates to

which the scapula models and corresponding humeral head components were attached.

Humeral components from the 22mm Kirshner Neer TSR system were loosely attached

to the upper plate allowing self-seating during load application. Positioning jack-screws, supporting the models affixed to the lower plate, provided an accurate means of controlling the direction of applied load to the models. The entire system is placed within the environmental chamber extending through holes in the top and bottom of the chamber and attached to the actuator of the MTS for control of temperature and load.

The rectangular calibration beam is prepared for testing by drilling four holes along the center of the beam spaced 4 and 6 inches apart. An amount of weight needed to create a desired stress pattern is calculated based on the dimensions of the beam. The predetermined weight is hung from the beam to create four point bending and the loaded construct is placed in front of the models within the environmental chamber.
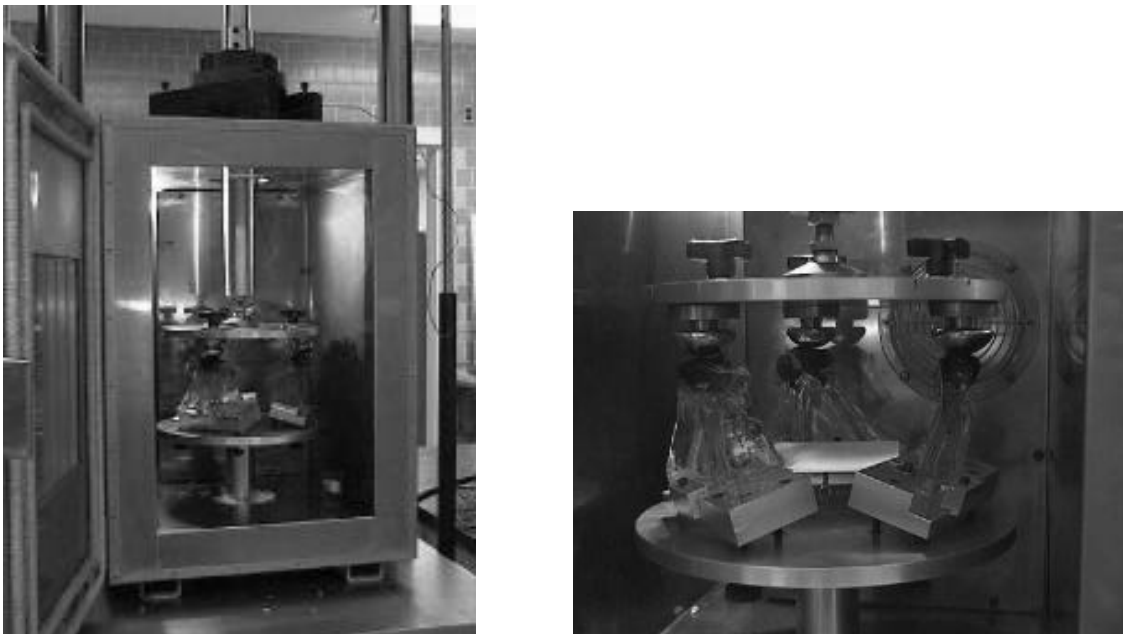
Figure 3.5: Testing fixture positioned in environmental
chamber mounted on MTS testing system.

3.5 The Environmental Chamber

One of the requirements of 3D photoelastic modeling is the ability to accurately control the temperature over long slow temperature gradients. This requires a computer controlled environmental chamber. The chamber available was large enough for the testing, but needed to be retrofit for a new controller. A microprocessor based Thermotron 4800 Programmer/Controller console was wired to the chambers conditioning systems and a type 'T' thermocouple via solid state relays. Analog and Transistor-Transistor Logic levels are input and output to control and monitor the system. Temperature profiles were programmed into the system for curing and testing. The testing profile consisted of a ramp from room temperature to near $121^{\circ}C(250^{\circ}F)$ at $5^{\circ}C/hr$. Temperature was then held for 3 hours before returning to room temperature at $2^{\circ}C/hr$ (Figure 3.6).
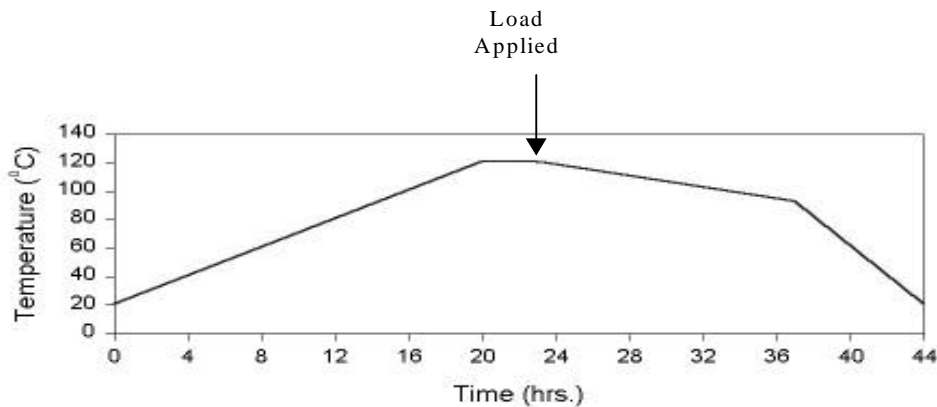
Figure 3.5: Stress freezing temperature profile used during testing.

3.6 Testing

Each of the three models of a test and associated calibration beam were subjected to the same temperature conditions throughout testing. The 4800 was programmed for temperature profile described above. As the temperature neared the $121^{o}C(250^{o}F)$ range the calibration beam stress patterns were observed with the use of a reflective polariscope to determine the critical temperature. Once the stress pattern reached an equilibrium the temperature was increase another $5^{o}C$ to the critical temperature. After reaching the critical point, the temperature was held for 3 hours before loading the models and returning to room temperature.

3.7 Sectioning the models

After testing, the lateral half of each scapula (i.e. the glenoid) was embedded into large blocks of epoxy resin that would provide fixation during the sectioning of the models. Models were cooled during the embedding process to minimize any exothermic reactions that may occur. Sectioning the models was a two step process. Truing the model was performed with 0.013inch loss of material using a Well precision diamond wire spool saw (model 3241, Delaware Diamond Knives, Wilmington, DE) requiring 15 to 30 minutes per cut. Blocks were cut to a predefined dimension then removed from the base. Less critical slicing was performed with a Jet horizontal/vertical band saw(model HVBS-463, Jet Equipment and Tools, Auburn, WA), modified with a hydraulic valve to control the rate of cut, requiring 1-2 minutes per cut. Epoxy resin blocks containing the modeled glenoid heads were sliced into 1mm thick sections with approximately 1.5mm being lost to kerf.

CHAPTER 4: ANALYSIS PROCESS

The focus of developing an analysis program was to provide an inexpensive (minimizing time and money) and easy to use alternative for photoelastic analysis. The goal was to develop a single PC based program that could be connected to an image database for input, provide image capturing and processing, solid modeling, and final analysis viewing methods. An ideal program would have functions for reading a calibration beam image and applying the calibration to other images to obtain stress information from pixel color values. It would also provide solid modeling capabilities for reconstructing three-dimensional objects from image slices including scaling, rotations, translations, color, and visibility changes for several objects independently. Finally, the program would provide the ability for stresses of three view dependent models of the same object to be combined into one effective stress model that could be viewed as multiple stress volumes.

Although the goals are considerably high expectations for a single project, much of the functionality has been developed providing a basis for building and improving the design. Most tasks were completed in one program, however the development of the image processing tasks were completed through the use of external software as a time and cost savings. The functions used in the processing are available in code form from an

image processing library and can be combined with the remaining analysis program in the future if desired. For ease of discussion the analysis process is separated into three sections: image processing, development of the analysis program providing spacing, alignment, and rendering, and then the calibration function (Figure 4.1).
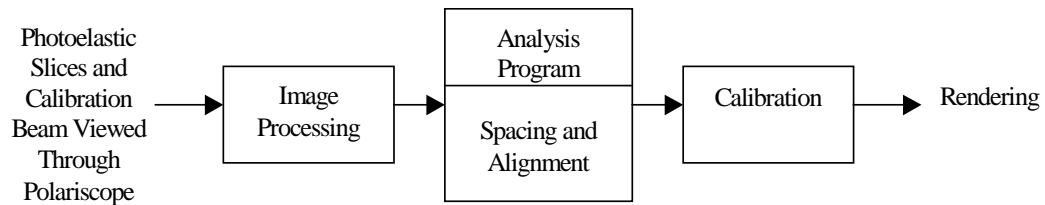
Figure 4.1: Analysis Process

4.1: Image Processing
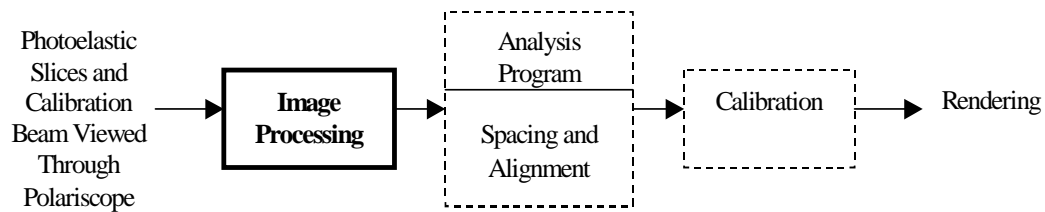
Originally the analysis program was expected to provide image capturing, processing and storage capabilities. Purchase of a development package, MIL, (Matrox Imaging Library, Matrox Electronic Systems Ltd., Dorval, Quebec, Canada) containing a variety of imaging handling functions was considered. The package would provide all the code required for all input, storage, and processing of images, however due to the cost and associated learning curve this solution did not seem appropriate. Instead it was decided to use independent software for the initial development of the process and at a later date this functionality can be added to the photoelastic analysis package as an upgrade. An image analysis program based on the MIL library, Visilog 5.0, (Noesis,

Ville St-Laurent, Quebec, Canada) was obtained that provided all of the imaging

capabilities (capturing, storing, editing, and data output) for a reasonable price.

Unfortunately, capturing an image requires more than software. During the

development of the initial process, equipment was selected from that available with the

expectation to upgrade piece by piece as is deemed necessary. An Osprey100-H frame

grabbing board, originally designed for telemedicine applications, was used to capture

images through a video input port. The design feature that makes it suitable for

telemedicine is the biggest flaw for capturing RGB images. To recreate flesh tones the

board has internal filters designed to enhance the red and green while diminishing the

blue of an image. Another drawback of this board is it can not be used with the image

processing software described above, however the board is provided with software for

capturing a 640x480 windows bitmap image that can be imported into the image

processing package. Tests of repeatability showed that even though images captured are

not the exact red, green, and blue intensities seen in an object they are consistently the

same, making the board suitable for use in developing the functionality of an analysis

program.

Originally the camera selected was a Toshiba CCD color camera (model 1K-

627AT) with a cs-mount lens. This camera and lens proved ineffective due to the small

distance required between the camera and the viewing object. A second camera, a JCV

RGB color video camera (model TK-1070U) was selected for better quality and

resolution. Several lenses were tested with this camera before selecting a Navitar 7000

TV Zoom lens providing an adjustable focusing and magnification at small focal

distances.

Different white light sources were also tested in conjunction with the image capturing equipment. Overhead projection type systems were found to have non-uniform distribution of the light and created a large yellow-white spot in the center of the image. If the light was positioned at a slight angle to avoid the 'hot spot' there remained an intensity gradient across the image that prevented any repetition in the analysis. A more dependable uniform light source was obtained from Bretford Acculight slide viewing box. Light is evenly distributed across a flat surface providing an excellent light source for a table top polariscope arrangement.

An x-y platform was fabricated to accurately position the slices within a Measurements Group model 080 table-top polariscope. The polariscope was mounted on the Acculight lightbox positioned beneath the camera for viewing of light transmitted through the model slices. Images were captured using a JVC (RGB camera) connected to an Osprey frame grabbing board via a video connection and saved as windows bitmaps (BMP or DIB). Calibration beam images were obtained from the same set-up. Images for calibration beams were then imported into the image processing program Visilog 5 where data was collected for the RGB intensity changes with respect to increasing strain. The raw data was then imported into a data analysis package SigmaPlot 5.0 (SPSS Inc., Chicago, Ill.) where it was smoothed with a low-pass filter and output into a calibration look-up table.

4.2 Analysis Program

```
                       ┌─────────────┐
Photoelastic           │  Analysis   │
Slices and   ┌ ─ ─ ─ ┐ │  Program    │ ┌ ─ ─ ─ ┐
Calibration    Image    │             │   Calibration      Rendering
Beam Viewed →│Processing│→│ Spacing and│→│        │→
Through      └ ─ ─ ─ ┘ │  Alignment  │ └ ─ ─ ─ ┘
Polariscope            └─────────────┘
```
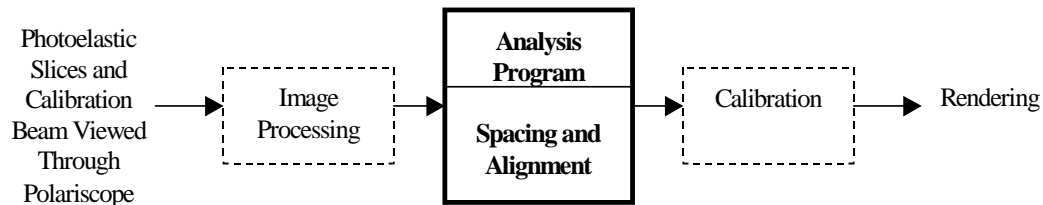
After capturing images and processing the calibration beam, the image files and

the calibration look-up table are used as inputs to program for completion of the

remaining analysis and rendering.

4.2.1 Software Used

The analysis program was developed using distributed freeware of The

Visualization Toolkit (vtk)[45, 46, 47]. Vtk is a system independent collection of

programming tools for learning and applying visualization techniques. Functions are

provided in a C++ class library and can also be accessed through a prototyping library in

the scripting language Tcl/Tk. The purpose of vtk is to offer the fundamentals of

visualization by providing functionality of the basic methods of computer graphics,

imaging, and visual data representation and processing, along with several commonly

used advanced functions of computer science, numerical analysis, and statistical

methods.

The interpreted or scripting language Tcl/Tk (Tool Command Language and

graphical user interface toolkit) is a precompiled language that executes commands

without the need of a compiler. Tcl is connected to C++ through wrapper functions that

allow the precompiled C++ commands (.dll libraries) to be called from within tcl.

Interpreted code is designed for rapid prototyping and development of a program. Tk is a

graphical user interface widget library implemented using Tcl to provide a working window type of environment.

It was decided to use the functions of The Visualization Toolkit (vtk) as the basis for the rendering program because of its ease of us, extensibility, and low cost. An additional benefit of vtk is its portability across multiple platforms providing development on PC or Unix based machines. The final program was written using a Pentium II 350MHz. PC, 64MB RAM, Windows 95 operating system, and the combination of Microsoft Visual C++ 5.0, vtk 2.3, tcl/tk 8.0.

4.2.2 Program Overview

The remaining analysis required includes the ability to combine the data collected from three photoelastic models into one contoured 3D model of the combined effective stresses. The three models were fabricated and tested, in the manner described in the previous sections, to produce the exact same stress patterns in each. Calculating an effective stress requires views from three orthogonal directions for every point within the model, so each model was then sliced along the x, y, or z direction. The analysis program is designed to input a series of slices and reconstruct 3D volumes for each of the view dependent models. Reconstructed volumes are then scaled, rotated, and translated to have an overlapping alignment. Once aligned all points of each model are calibrated to a stress value depending upon RGB values. An effective stress volume is calculated by sampling the structured grid defining the stress values of the three view dependent volumes of a test. Contours are then generated for different magnitudes of effective stress and rendered with varying opacity to show internal and surface measurements.

4.2.3 Program Design

In creating a graphical program with vtk there are several formats data must be passed through to get to the final rendering (Figure 4.2). Input data collected from the windows bitmap images of the model slices are first stored in an image cache (a 2D array of 8-bit unsigned characters). Data is then passed to a

Figure 4.2: Data Pipeline

structured points array for sizing, alignment, and contouring. The structured points data set is simply an array, representing a 3D grid, that is indexed by knowing the size and spacing of each of the x, y, and z dimensions. It is an efficient way to store data when the relationships of the individual data points have not yet been defined. Contour functions then search the structured grid and interpolate to find exact locations of points that equal a user defined threshold value. The Marching Cubes algorithm developed by Lorenson and Cline[28] was used to generate all surface contours. Once a contoured region is generated the data is reduced into a collection of points, lines, and triangles that is stored in a polygonal dataset. The polydata set is a list containing connectivity and location of points representing unstructured or minimally structured primitives that are used in high speed rendering. Viewing a collection of contour primitives is accomplished by mapping the primitives into the appropriate rendering system and applying the object properties (color, opacity, transformation...) stored in the actor.

Along with understanding the flow of data in vtk, designing the analysis program involved understanding how data is passed from one process to the next, and methods of minimizing memory usage. One method used to manage memory efficiently was designing the program to use polydata sets as the controlling element. Input data would be processed to the point where a polydata set was generated and then the original data would be deleted. Resizing and reslicing each volume of images reduced the number of data points used, saving memory and computations while creating a uniform structured grid for sampling data points. Another controlling factor in the design of the program was the necessity of object transformations to occur before contouring. Generally any movements of the objects displayed on the screen is completed in the final stages of the imaging pipeline through functions of the actors and all contouring functions act only on structured points data sets occurring earlier in the pipeline. Therefore, additional filters needed to be added to allow alignment of the three view dependent models before an effective stress could be calculated and contoured.

The steps of the processing pipeline (Figure 4.3 a, b, and c) are provided below. Data originates as three view dependent volumes of windows bitmap images and is passed along the path of Figure 4.3a. Each volume presents a view of stresses from a different direction for the same model. Images of a volume are read by specifying a file name and a numerical range for the filename's three digit extension. Spacing between images can also be set during the input, or in the next step of resizing. Resampling or resizing is a function for setting a magnification scale for each of the axis individually. If there is a desired spacing in the 3D grid containing the image data following resampling then reslicing should be used. The reslicing interpolates data to fit a predefined grid size

and spacing. Using the resampling and reslicing together creates a uniform structured

grid of the data volume by transforming from 640x480x15 (15 images at 640x480 pixels)

to

(a)

(b)

(c)

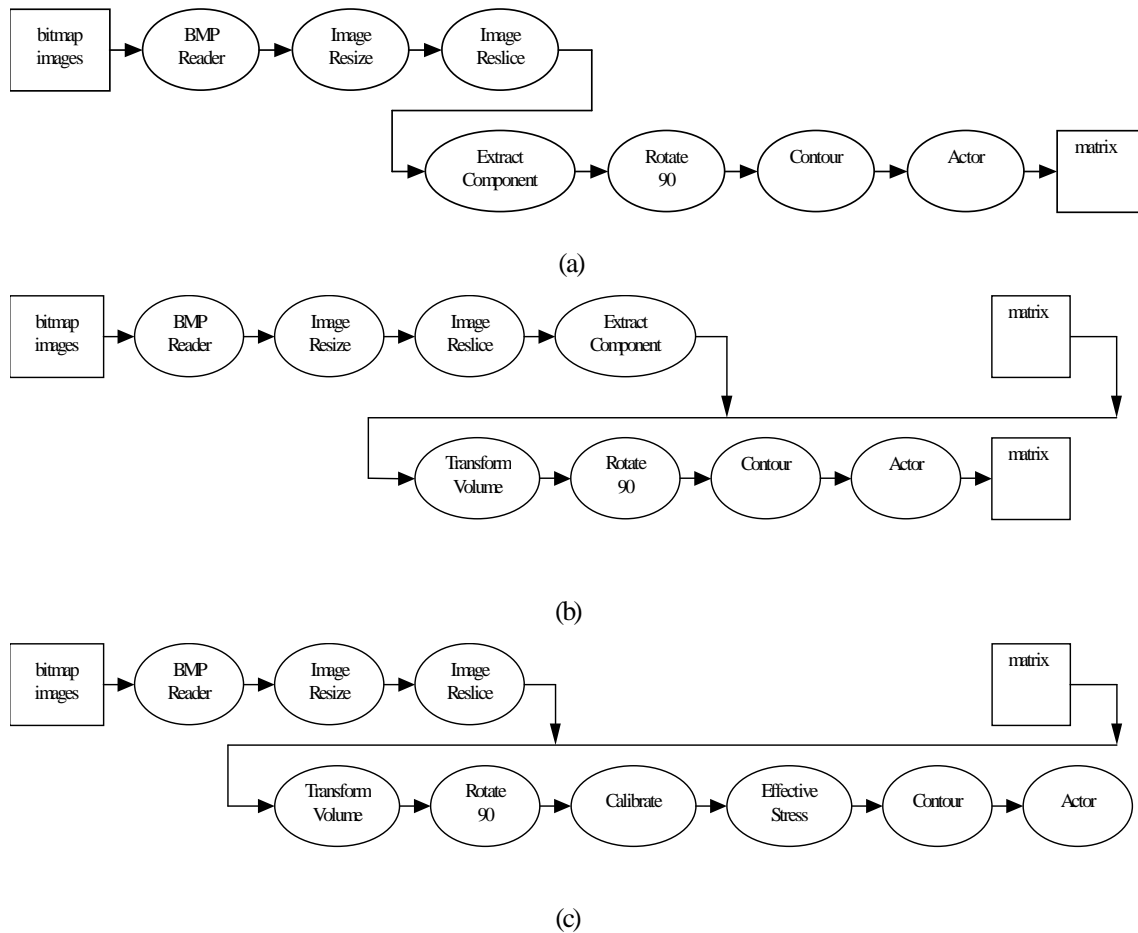Figure 4.3 a, b, and c: Image pipelines of the analysis program.

64x64x64. Currently, resampling in the z-direction (between images) creates incorrect

data and additional work needs to be conducted to maintain dependable results. Present

functions reduce the x and y axis' by sampling one of every four image pixels. The red

component of the data is then extracted to minimize computation during the remaining

alignment process. Vtk only allows images to be input as x-y slices and stacked along the

z direction when creating an image volume. So, the next step is either a y or z rotation to

correctly orient the view dependent volumes. After returning the volumes to their

specified viewing directions the volumes may still need slight translations to align the

data. The alignment is accomplished by contouring at a low value to create a shell of the

perimeter of the model. The outer contour of the three models are then rendered and

aligned using actor transformations. When a transformation is selected the object

transformation matrix of the actor is then retrieved and passed up the pipeline. The

volume transform function then reads the matrix and applies it to the data from the red

extraction (Figure 4.3b). In transforming the data contained within the structured grid

care was taken to assure the transformation is made on the data contained within the grid

without changing the original grid. Output of the transformed volume is then returned to

the rotation and contouring functions to be rendered again in case more transformations

are desired. When alignment is complete the final transformation matrix is applied to the

data directly after the reslicing, skipping the extracting process, and passing the data to

the appropriate rotation function (Figure 4.3c). The effective stress function reads the

three volumes and scans each point in the structured grid volume of each calibrating the

data and calculating an overall effective stress. A final output is then passed to the

contouring functions and rendered.

4.2.4 Segmentation

Segmentation refers to displaying reduced sections of the dataset to provide the

user with better viewing of object features. Due to difficulties with the calibration

functions to be described later, a separate program was developed for viewing stress

contoured models. Once calibration functions are determined acceptable then all facets of

the program will be combined into one executable.

Currently, output of the calibration program produces polydata set contours of

various stress magnitudes. These contours are then imported into the segmentation

program either individually or in multiples. Multiple contours can be appended together

allowing each contour to have individual color and opacity values while providing the

ability to view all contours simultaneously. Segmentation is performed by the use of two

clipping planes. The first plane allows viewing of any data points existing behind it,

while the second plane allows viewing of data existing in front. Initially the planes are

positioned as front and back clipping planes allowing the user to view entire datasets

held between. User defined segmentation is then performed by moving the planes

through the data with arbitrary orientations. Each plane is identified by a central point

and a vector designating the view plane normal. Menu functions provide the user with

the ability to modify the location of central points and vector normals independently,

thereby controlling each planes location and orientation. In a vtk render window,

rendered data can be rotated and zoomed at any time, the addition of two independent

clipping planes allows the user to create a cornering segmentation to provide better

isolation of internal areas of interest during rotation

4.3 Calibration
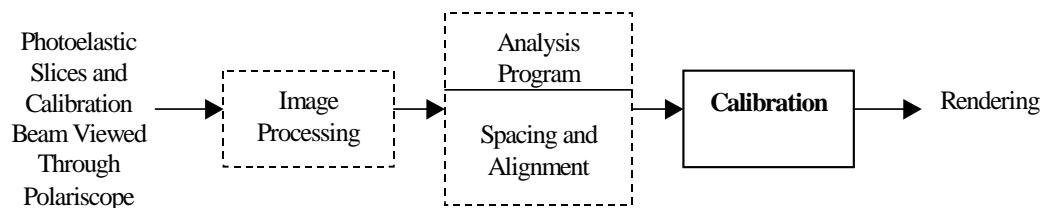
4.3.1 Introduction

The most important and difficult parts of automating photoelastic analysis is

converting what is seen in an image to usable stress related values. As seen in Figure 2.1

the sinusoidal intensities of the red, green, and blue components of light transmitted

through a photoelastic model can be related to stress magnitudes within the model. By

knowing the size and load applied to the calibration beam an exact stress can be

calculated for every point on the beam. Stresses can then be related to the various colors

observed in the beam, when view through the polariscope, based on their location.

Calibrating an image is a matter of matching the colors of the image to the colors of the

calibration beam and determining what the corresponding stress value is. Using plots

similar to Figure 1.1 images are calibrated by selecting the RGB intensities for each pixel

of the image and comparing with an RGB calibration curve to determine a stress value.

The difficult part is locating the unique location of the calibration curve were the RGB

intensities of the pixel being analyzed are equal to the calibration RGB intensities. Added

to that our studies have identified a variety of other factors that influence the shape and

size of the calibration curves. Several factors affecting the calibration were previously

described with respect to Equation 1.1. Tests performed confirmed several of these and

other factors as well as identifying how each affect the intensity of light viewed through

a photoelastic model.

4.3.2 Material Effects

Accurate calibration of a stress frozen photoelastic model requires a good

calibration beam. In order for the beam to provide useful data it not only needs to be

molded and processed at the same time the models are, but also needs to be of the exact

same mixing batch, have the same surface texture, and be viewed with the same camera-

object focal distance. Figure 6a and 6b show how changing materials can alter the

wavelengths of transmitted light differently. A noticeable shift in the magnitude of the

blue and green output intensity patterns is seen by two calibration curves from different

(a)   (b)
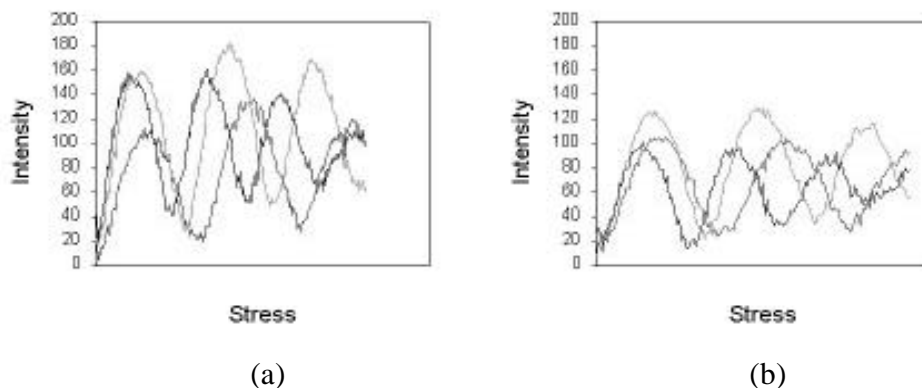
Figure 4.4 a and b: Output intensity plots of two different materials.

materials. The sequential peaks are altered in the red and green wavelengths with the

material on the left having a higher second peak for each color. Figure 4.5a and b

demonstrate the intensity changes of different batches of the same material. Here the

initial region of the curve of 4.5b does not approach zero as in 4.5a with the blue

intensity being greater than the green in 4.5b. Amplitudes of the green intensities are also

greater in Figure 4.5b.

<center>(a)   (b)</center>
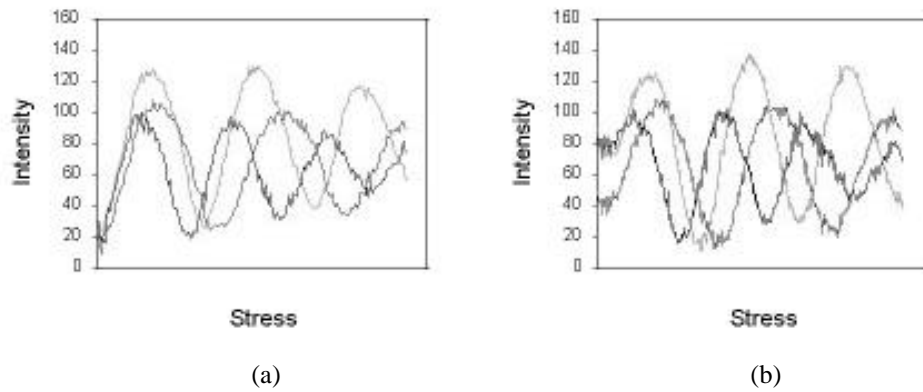
Figure 4.5 a and b: Intensity vs. Stress plots of different
batches of the same material.

4.3.3 Surface Prep

Comparing Figures 4.6 a and c to 4.6 b and d demonstrates the changes caused by

polishing a surface. Polishing a surface minimizes the reflections and refractions of

transmitted light caused by the surface irregularities. Colors of a polished surface are

more distinguishable because the maximum and minimum of the intensity amplitudes are

more pronounced.
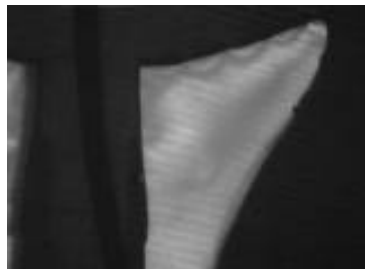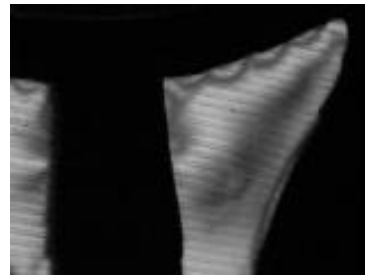
<center>(a)   (b)</center>
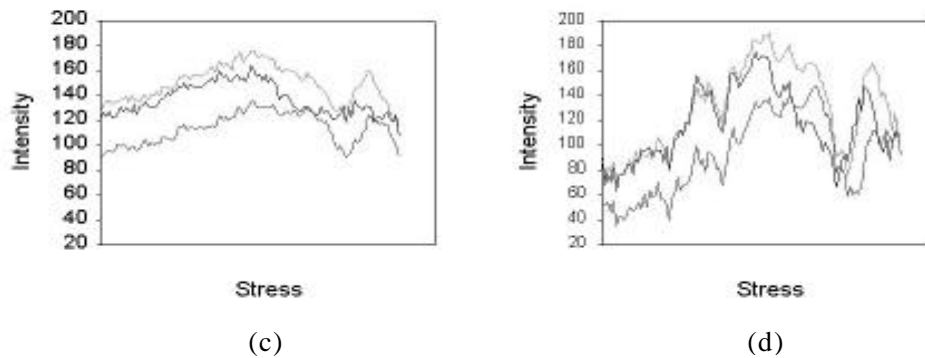
Figure 4.6 a, b, c, and d: Images and Intensity vs Stress plots
of different surface textures.
4.6 a and c: Unpolished   4.6 b and d: Slightly polished

4.3.4 Focal Length

Another factor influencing the transmitted light intensity recorded from a photoelastic model is the object-camera distance. Focal length changes the measured intensity opposite of what might be expected. An increased focal length allows more ambient light to be recorded as well as blurring intensity changes of small regions. The effect is increased amplitudes for each of the primary colors (Figure 4.7a and b).

Note: The large low gradient stress regions used to generate these plots do not represent the effects of small rapid changing stress regions.
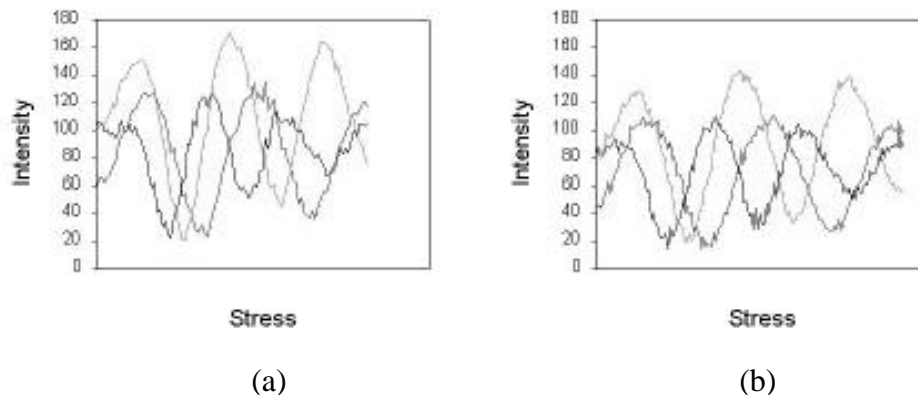
Figure 4.7 a and b: Intensity plots of different focal distances.
4.7a: Large focal distance 4.7b: Small focal distance

4.3.5 Hardware

Recording devices are a concern when comparing specific photoelastic results with the work of other authors. Filters of the camera and frame grabbing board can significantly alter the images retrieved. As mentioned previously the Osprey frame grabbing board may not produce results of the quality of a Matrox frame grabbing board. However, tests were conducted to measure repeatability and found no changes from measurements obtained at different times. Eventually an upgrade in capturing boards will be made to provide comparisons with work of other authors, but for the purposes of designing a photoelastic analysis package the current system is adequate.

4.3.6 Software

Unfortunately late in the program development several errors were detected in the functions provided by vtk that affected calibration. A windows BMP image file has a format that is dependent upon the header and can take numerous forms. The vtk

functions that read these files were coded for only two basic formats. In many cases this does not create a problem, however if a modern image processing software package is used to save the images then it may or may not save it in one of the formats readable by vtk. Several image slices used in this project were stored in a more recent BMP format. When read into vtk the indexing of the data was shifted causing the vtk blue to index the image red, the vtk red to index the image green of the previous pixel, and the vtk green to index the image blue of the previous pixel. Once detected the indexing could be changed within the code, however it is possible that this may have to be changed for every image of a volume independently. Additional errors were detected in the functions designed to resample and reslice the image volume. During the resampling process the data is not necessarily resampled it may be approximated from nearby pixels. Normally this would not cause a problem, however due to the sinusoidal nature of the intensity values and the amount of volumetric change desired this function may in effect smooth the data creating mapping errors during the calibration.

4.3 7 Calibration Methods: Divided Regions

Early calibration tests used a region dividing method that separated the calibration curves into regions based upon the locations where the intensity of two colors were equal (Figure 10). Scanning from the beginning of the curve (left side equal to zero stress) a vertical line is drawn each time two color intensities cross. Vertical lines then define stress regions based upon intensity order (i.e. 'red > green > blue' or 'red > blue > green' or 'green > blue > red' ...) defined as 0 through 15 in Figure 4.8. When calibrating an image a simple check of the order of a test pixels RGB intensities can minimize the number of possible calibration stress regions that the pixel could belong. By further

identifying differences in the order separated regions of the calibration curve the number of possible regions can again be minimized. For example, if the magnitude of the intensities of the pixel being calibrated are 'blue > red > green' this combination can occur in two regions 4-5 or 10-11. A decision can then be made to determine in which of the two regions the test pixel belongs by observing the differences of the two regions. It can be seen that a simple check of the magnitude of the test pixel's green intensity is enough to determine which is the correct region for the pixel to belong. If the pixel has a green intensity less than 40 it will belong the 4-5 region, if it is greater it will belong to the 10-11 region. Once a single region has been identified for that pixel a stress magnitude is assigned to it and the calibration continues to the next pixel. This method has one disadvantage in that there can be non-unique or closely matching regions that could cause larger errors in selecting the appropriate stress region. For example mistakenly selecting group 1-2 instead of 7-8 will cause what should have been a mid-range stress calibration to be a very low stress value.
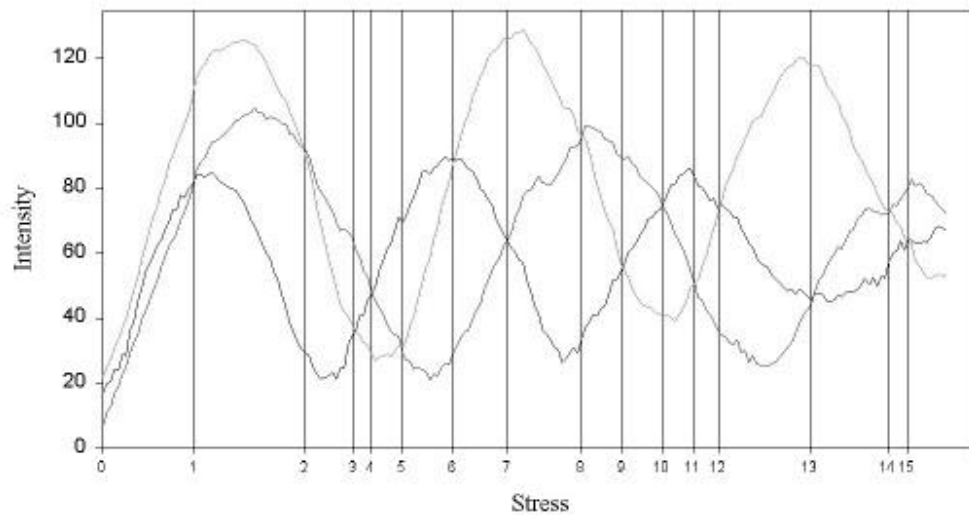
Figure 4.8: Stress regions defined by RGB

4.3.7 Least Square Error

In an attempt to increase accuracy the method of calibrating using least squared

errors suggested by Ajovalasit et al.[1] was tested. A look-up table was produced from the

calibration beam data mapping RGB intensities to stress values. RGB pixel intensities

from an image being calibrated were then compared with the table by a least squared

error calculation (Equation 4.1). Searching the calibration table for the index that

minimizes the error determines the corresponding stress value.

$$E_i = (R_i - R_m)^2 + (G_i - G_m)^2 + (B_i - B_m)^2 \qquad \text{Equation 4.1}$$

i = index of calibration table
m = pixel being measured

Analyzing the results of this function detected errors similar to those of the dividing

ranges. Pixels that should represent the highest stresses were getting calibrated to the

lowest stresses. This error appears to be caused by the surface properties of the photoelastic slices combined with very small high stress regions. Model slices are unpolished causing a damping of the amplitudes of the transmitted light. Viewing entire slices in a single image requires the camera to be positioned further from the model surface causing intensities produced from high stress gradients in small areas to be faded into intensities similar to zero stress regions.

4.3.8 Modified Least Error

A slight improvement in calibration was made by using a modified error method described by Ramesh et al.[37]. In this method an attempt is made to produce more distinct curves to use in the calibration beam table. By changing values of the table from a red intensity to a red – green intensity it is possible to minimize the number of non-unique data points in the calibration table. In our tests several combinations of color differences were tested before selecting a combination of red, green-blue, and blue as values composing the look-up table. This method improved the calibration, however output images still produced several areas of concern. The method does work well when the image being tested was the same image that produced the calibration curves. (This was actually a large improvement over the other methods.) However, when attempting to calibrate a different image several areas would continue to map to the wrong stresses. It was at this point that errors due to the software problems listed above were first detected. Due to multiple factors influencing the errors it was difficult to determine if the calibration function was actually working correctly or not. In locating the source of the errors an additional calibration method was devised.

4.3.9 Combination Calibration

It was noticed in the previous tests that the relationship of the ordering of the color intensities remained the same even though the actual amplitudes of the intensities varied among the unpolished image slices of a single test. Combining all of the methods above created a final calibration function. The calibration curves were divided into the various ordered intensity stress ranges and the ranges were included as part of the calibration table. When an image pixel is calibrated it first checks the ordering of the RGB intensities. Based on the ordering of intensities a scan of the calibration table can identify a minimal set of stress ranges where the point could exist. The modified error function is then applied to this minimal data set to determine the representative stress value. Coding a modified pixel indexing, and calibrating images at 640x480 without resampling or reslicing eliminated software errors. Output errors at high stress magnitudes continue to exist with the use of this function, however the numbers are much less. These remaining errors are believed to be caused by the model slice surface texture and focal distance. The irregular surface texture caused by machine tool marks in the unpolished model slices causes the light to reflect and refract differently (Figure 4.6 a and b). Any calibration based on the transmitted light intensity can be predicted to fail in regions around these tool markings. In the high stressed areas the focal distance also produces inaccurate intensity measurements by blending and amplifying the actual values.

CHAPTER 5: RESULTS

Images obtained for analysis were similar to those shown in Figure 2.3a (for a calibration beam), and Figure 4.6 a and b (for slices of the photoelastic model). From these images the model volumes were first reconstructed as a solid model to demonstrate the ability of the contour functions to differentiate between the model, the background, and the glenoid component. Selecting a series of slices taken along the y-axis of one model, Figures 5.1 a, b, and c were generated from 15 (640x480 pixel) image slices. Images were read in as a single stacked volume with a 20 pixel space between each slice. The long rectangular artifact displayed at the bottom of each image volume, produced from a laser positioning system used to align photoelastic slices during acquisition, was accidentally recorded as part of each image slice. Comparing these shell contours with an original model (Figure 5.2) it can be seen that the geometry appears to be correct.

(a)   (b)   (c)
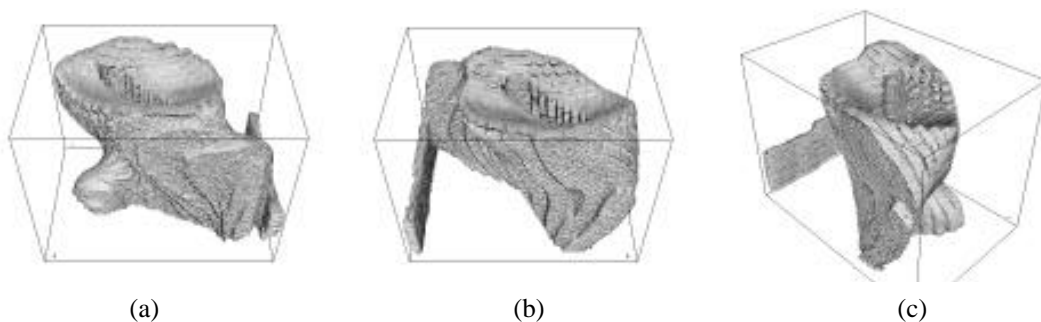
Figure 5.1: Model surface contour generated from 15 image slices at 640x480.

(a)   (b)   (c)
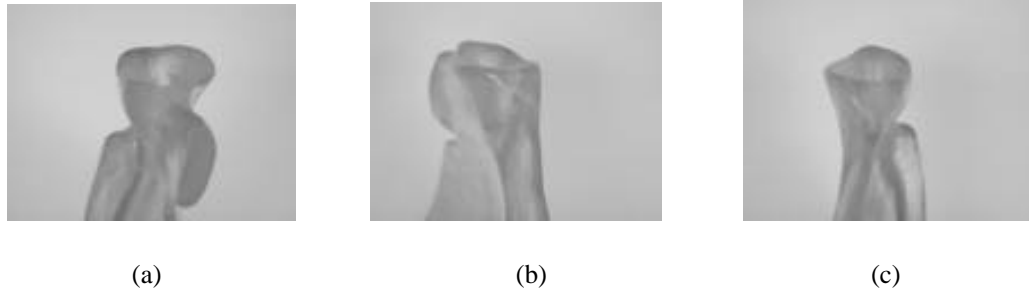
Figure 5.2: Molded photoelastic model.

During the generation of the surface contours computer memory and processing

speed deficiencies were first detected. Purchase of additional hardware for memory,

processor, and graphics accelerator upgrades was considered and dismissed due to

funding difficulties. A decision was then made to reduce the size of the dataset by

resampling the 640x480x15 image volumes to generate equally spaced data in a

64x64x64 volume. Advantages of the smaller dataset were two-fold; reduced processing

led to faster rendering, and an equally spaced data structure provided the ability to

combine data sets of variable length produced from different viewing directions.

Examples of the surface contours generated from the same input as in Figure 5.1 with the

additional step of resampling to a 64x64x64 volume are shown in Figure 5.3.

(a)   (b)
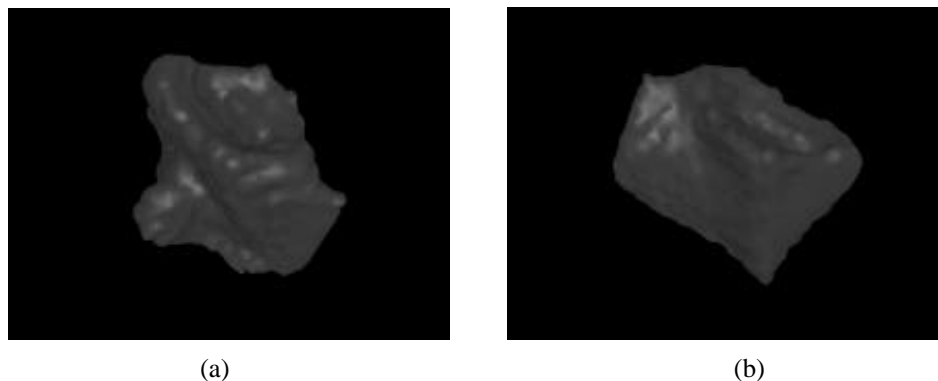
Figure 5.3: Surface contour of image
volume resampled to 64x64x64.

Comparing images of Figure 5.1 and 5.3 differences of the surface texture can be detected. Reducing the number of points within the volume creates a smoother transition from one slice to the next, however a gridded structure can be seen within the surface of the smaller volume. Additionally, loss of data around the rim of the glenoid has created notches in the surface. Overall, the geometry appears to be maintained and the rendering speed and memory savings allow multiple volumes to be viewed and rotated in one scene.

Figure 5.4 shows the alignment process of overlapping the surface geometries detected by volume renderings of three models of a single keeled glenoid component test. Each model was sliced into either axial, coronal, or sagittal sections. Spacing between slices of a single model was adjusted to align with orthogonal views observed from the other two models of a set. Each model was transformed independently to achieve overlapping of all three models. Minor incongruent differences in each of the volumes can be detected and appears to be caused by the resampling process. Viewing the slot where the keeled glenoid component is positioned (Figure 5.4c) shows the three volumes align adequately and should produce reasonable results when calculating the effective stress.

(a)   (b)   (c)

Figure 5.4: Alignment of three view dependent volumes
of a single keeled glenoid component test.

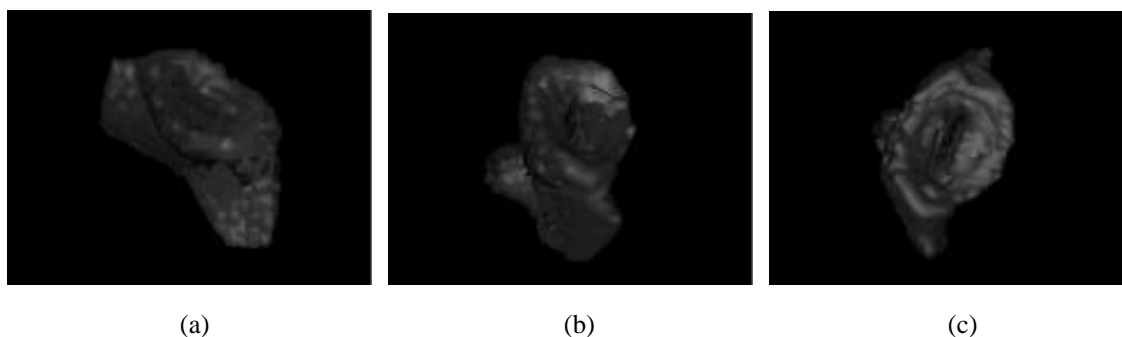A test of the effective stress calculation was then made by assuming the red component, of each of the volumes shown in Figure 5.4, was the stress value defining the volume. Each point of the simulated effective stress dataset was then calculated based on the value of the red component at the same location from each of the three input volumes. Three contours of varying opacity were generated from the effective stress volume. The order of the contours is: red represents the highest magnitude, followed by green as a midrange value, and yellow representing the surface of the entire object. Unfortunately, only one image was saved demonstrating this first test (Figure 5.5). The orientation would be similar to viewing the right side of Figure 5.2c. Many errors can be detected from this image. Most obviously, the surface is almost unrecognizable and contains several holes. Additionally, the red region shows many random artifacts that are fragmented and discontinuous. The favorable results of this test are the ability to demonstrate multiple opacity contours of several colors, as well as generating a surface that is close to what is expected, therefore, proving the indexing of the three volumes must be correct .

At this point several of the errors, listed in the previous chapter, were first detected. It is apparent from Figure 5.5 that calculations of the effective stress can not be conducted on a reduced size dataset. During the resampling process the original geometry of the surface is deteriorated and it can be assumed that any internal structures of the data will also be lost. Upon testing the calibration function, the errors of using different materials, mixing batches, surface textures, and viewing distances were also detected. Additional problems were associated with Vtk's ability to only read one type of

"bmp" file. Numerous tests were then performed to debug the process of calibrating test images while avoiding any influence from the other errors listed.
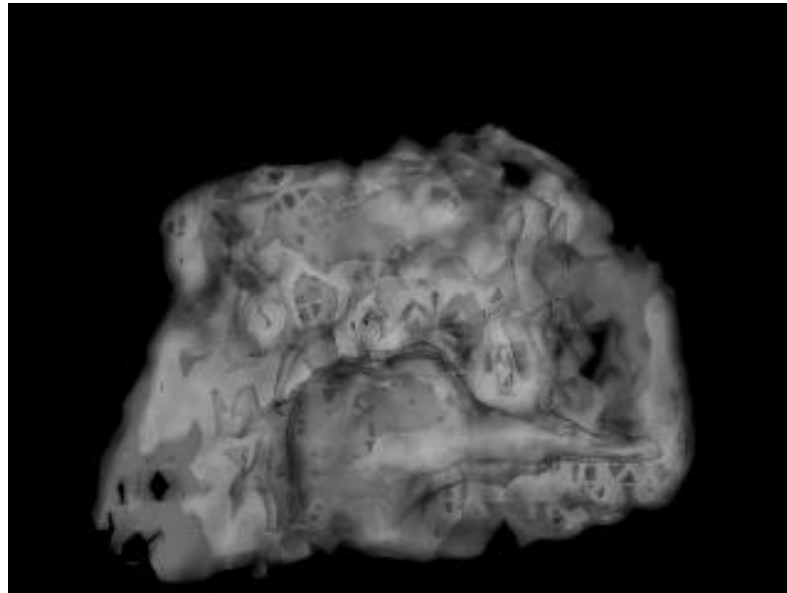
Figure 5.5: Original test of combining data of
three independent volumes into one.

Two test samples were machined from one block of photoelastic material. A four point bending calibration beam and 'C' shaped test sample were made produced with similar surface textures in each. Load was applied to each generating three fringe stress patterns (Figures 5.6 a and b). Data of the transmitted light intensity variations produced by the calibration beam were then recorded and separated into pseudo-stress ranges as defined in section 4.3.9.

A look-up table was then created by mapping the transmitted light intensities of the calibration beam to the pseudo-stress ranges. Calibration of the test image was then performed by measuring the color intensities of each pixel and comparing with the intensities of the look-up table to locate the corresponding stress value. Stress values were then stored in an output dataset. The output dataset was then used as input to

contouring functions that separated the final image into four stress regions. Figure 5.7 is

the contoured output of the calibration of Figure 5.6b with the data collected from Figure

5.6a. In this image bright red represents the highest stress, followed by dark red, then

green, then blue representing the lowest stress.

(a)   (b)

Figure 5.6: Calibration beam and test sample produced with identical
surface properties machined from one piece of photoelastic material. Both
images showing bottom in tension and top in compression. (a) Calibration
beam in four point bending. (b) 'C' beam test sample.
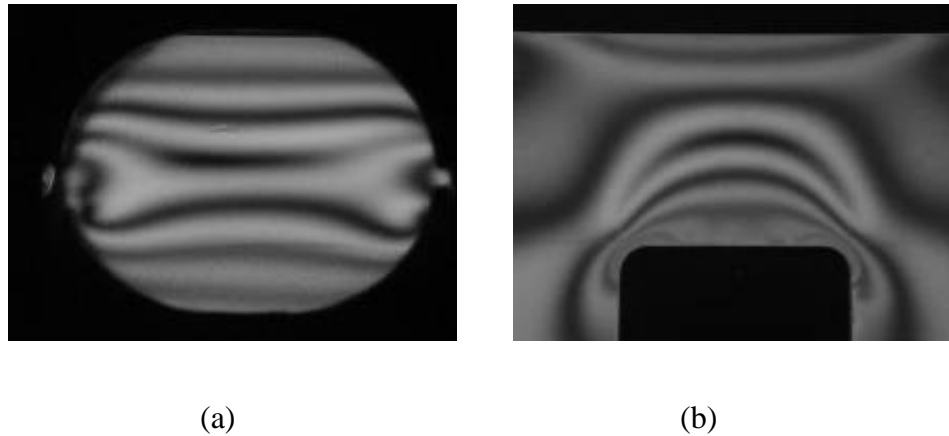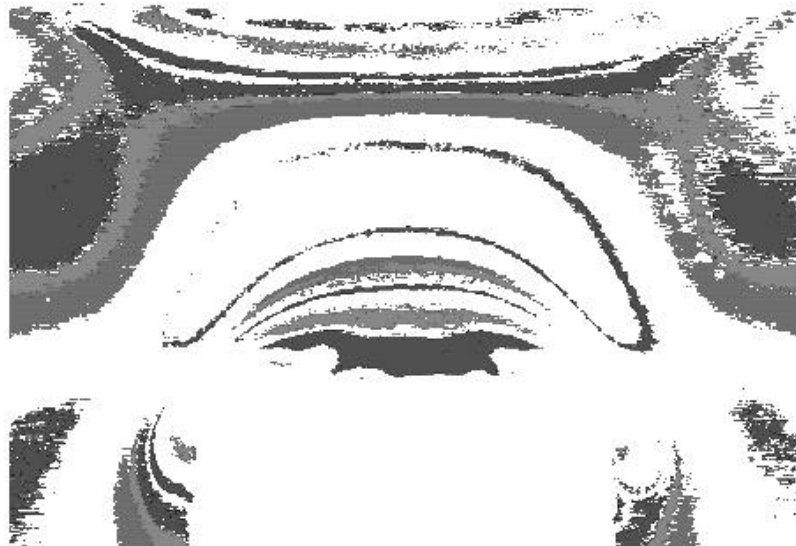

Figure 5.7: Contoured image produced by calibrating
image 5.6b with data collected from image 5.6a.

Calibration producing image 5.7 was very successful. The highest stress regions of the bright red are not visible, but this may be due to a incorrect choice of a threshold value for generating the contour. High stress regions do appear in dark red, and except for the kidney shaped loop in the center of the image, all regions are where they would be expected. As stresses decrease to the center of the image the color sequence of red – green – blue is followed correctly. Large open white space appearing in the center of the image is caused by a function of the calibration that removes any data falling below a specified threshold and considered insignificant. The only error is the small kidney shaped region of dark red in the center. This is caused by an attempt to calibrate the higher stress regions. There are two conditions that cause red, green, and blue intensities to be numerically close to each other. One is located around the black zero stress region that changes into white as stress begins to increase. Here all intensities start at zero (black) and then increase together to a maximum of 255 (white). The other area where intensity magnitudes converge is at high stress levels where the amplitudes of the intensities are being depressed. When using a look-up table to compare intensity measurements of each pixel of an image, slight variations caused by noise in the image recording can be expected to cause a failure in the calibration function in these regions.

Having debugged the calibration function the next step was to return to the images obtained from the photoelastic scapula models. A new look-up table was created from the calibration beam corresponding to the image dataset used in the beginning of the chapter. The values of the table were modified slightly to account for some polishing differences and applied to the dataset. Working with full 640x480 images, the pixel intensities were again compared with the look-up table to produce a stress gradient. In

this series the number of contour was limited to three: red is highest, then green, and blue representing the lowest stresses. The output is shown in Figure 5.8 a and b.
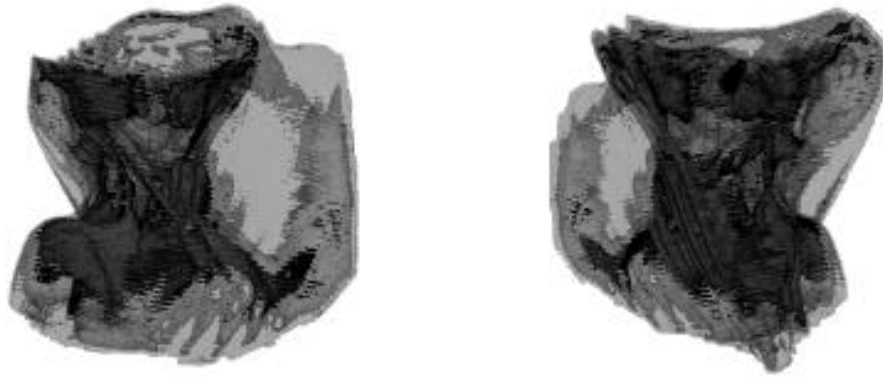
Figure 5.8: Two views of the calibrated
volume of a keeled glenoid component test.

Stress patterns of the rendering of Figure 5.8 correspond with the patterns of individual slices (Figure 4.6 a and b). Error can be detected as nearly horizontal lines created from remaining machine tool marks of the model slices. The threshold of the highest stress region contour is lower than desired, creating large regions making comparisons of stress patterns of two different models difficult, however attempts to reduce the size of the regions creates a more fragmented image that is difficult to analyze.

Polishing samples to remove remaining machine tool marks requires additional equipment and time. To remove the artifacts caused by the machine tool marks, images were manually edited. Original images were separated into four stress regions based on color (Figure 5.9). Black, white, yellow were individual regions, and any other color representing the higher stresses was considered to be in the fourth region. Images were then re-colored using only shades of red. This is similar to the calibration process in which the original image intensities are translated into a single scalar value of stress. The

manual processing exchanges the reduction of image acquisition errors for an additional operator dependent error and time.

(a)   (b)
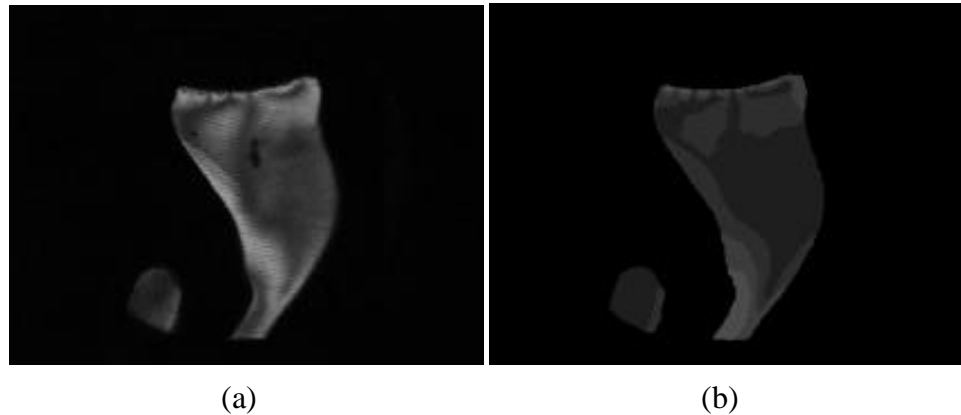
Figure 5.9: Manually calibrating images to stress ranges.
(a) Original images slice from model. (b) Manually edited
image showing four stress regions.

Manual calibration coloring of images was time consuming, however results obtained were found to provide several benefits. Minimizing error and reduction of data size was achieved as expected, having well defined contour regions led to additional data reduction, and demonstrated deficiency in the three dimensional contouring. With larger data regions and only four possible values, data could now be resampled along the x and y directions without severely altering the output. All images of Figure 5.10 were sampled once for every four pixels in the x and y directions of each image slice. Volume contouring revealed unexpected errors due to the curvature of the scapula model and the material loss during the sectioning process. Figure 5.10 shows the small regions of similar stress are incongruent from one slice to the next due to the high curvature and loss of data in the region. Images of 0, 30, 90, and 180 degrees were obtained from the pegged and keeled component models of a single view dependent volume for

comparisons. Patterns show stresses of the pegged component to be more distributed around the plateau, while the keel transfers load into a high stressed region at it's tip.

Keel   Peg

0°

30°

90°

180°
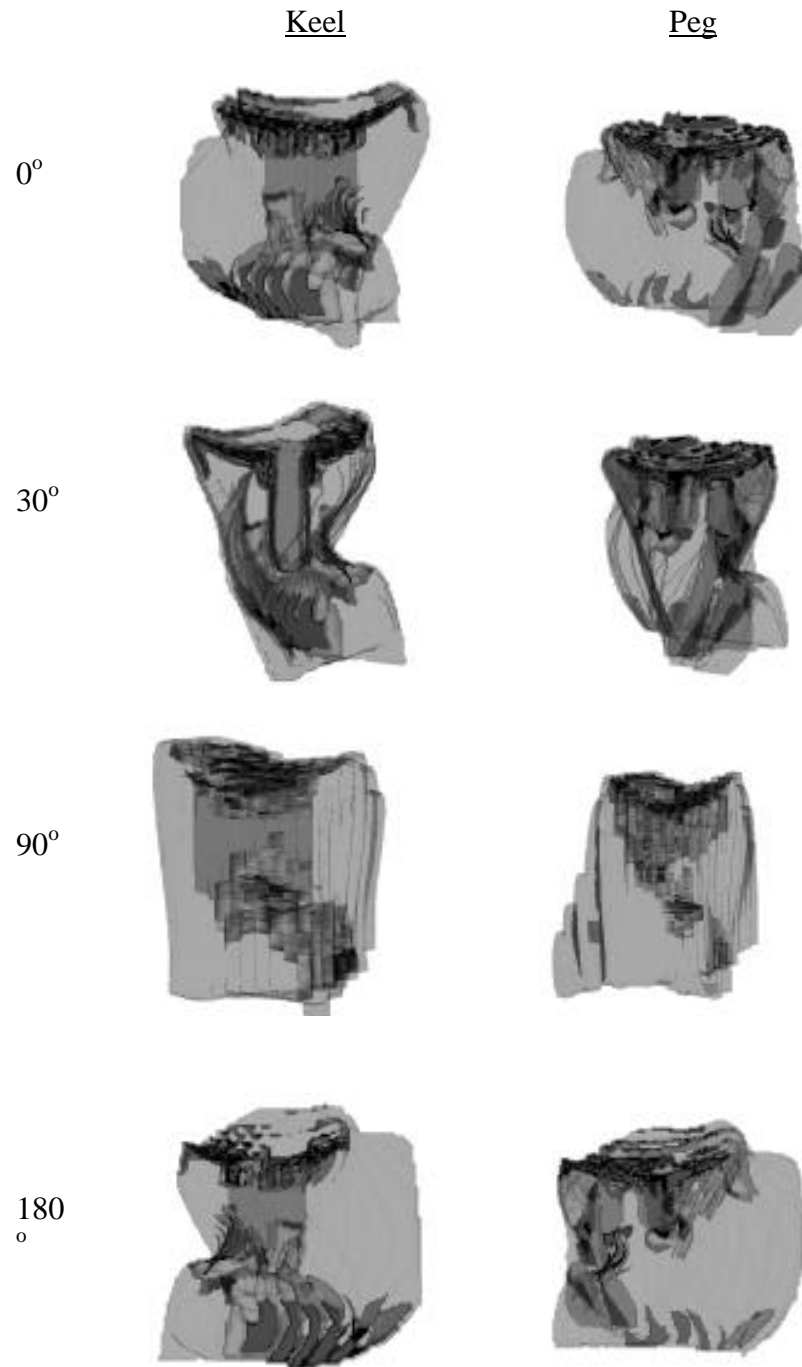
Figure 5.10: Views of peg and keel stress
patterns for a single view dependent volume.

CHAPTER 6: DISCUSSION

The original expectations of the analysis program were to develop one PC based

program that would provide the following:

- image capturing
- image processing
- images database storing and retrieval
- calibration functions to translate the RGB values of the images to magnitudes of stress
- reconstruction of 3D models from a series of images
- transformations of 3D volumes for alignment
- combine image volumes constructed from three orthogonal view directions into a single overall effective stress volume
- view contours within a volume
- sectioning of the volume for various methods of viewing

Almost all of the functionality has been developed. Procedures have been defined for all

tasks needed for analyzing 3D photoelastic stress frozen models. Image capturing and

storage is performed with the use of external software, and all the analysis is performed

by PC based programs. Additional functions for database access have not been

implemented, however the functions are available in toolkit libraries and can be added

later. The analysis programs have been developed on a PC platform, but are also portable

to other platforms. Errors remaining are part of the calibration function and are due to

unpolished samples.

6.1 Related Work

Results for comparison of combined effective stress regions of the two types of glenoid components are not complete. However, at this point differences can be seen from views of one view dependent volume. In the keel stresses appear to be transmitted to the bone through the tip of the keel creating stress shielding in the areas directly beneath the plateau. The peg design shows some transfer of stresses through the tips of the pegs, however the majority of the load is distributed to the bone closer to the joint surface.

The results for the keeled component are similar to the work of other authors in finite element studies of numerous glenoid component designs[9,22,23,34,49]. Orr[34] and Friedman[9] observed stress shielding of the keeled design using 2-dimensional models. Stone also found uncemented metal backed components produced stress shielding conducting 2D tests in two perpendicular planes. Lacroix etal.[23] with the only finite element study that uses a 3D model also observed stress shielding of the glenoid component, unfortunately his work is recorded in an abstract form and the entire paper could not be located. Advantages of the finite element method are executed in each of these studies with the use of several materials and material properties. Molding several layers of plastic to produce a photoelastic model that represents the multiple bone types and stiffnesses would be very difficult and is easily accomplished in a finite element model. The limitations of the finite element model are a lack of knowledge of true boundary conditions and the need to keep models small and 2-dimensional for computing purposes. Overall, the two modeling methods seem to agree for the keel results. The peg component is a recent design and no other data is available.

6.2 Future Improvements

The basic methods of automated 3-D photoelastic analysis are complete. From here improvements can be made in the analysis software, hardware, model preparation, and further model testing.

6.2.1 Software Improvements

Currently, there are three programs providing the alignment, calibration, and rendering procedures. Once final decisions can be made as to the ideal calibration function, the three programs can be combined into one. At this point it is easier to modify and recompile the smaller calibration program independently. Functional improvements of the program include adding stress direction measurements to the effective stress calculation, additional importing and exporting capabilities, and comparison viewing capabilities. Acquiring six more images per model slice would provide information as to the direction of the principle stresses within each slice, and increase the accuracy of the effective stress calculation. Functions to optimally determine the direction of principle stresses at each point will have to be written. Allowing a variety importing and exporting format options would make the program data more accessible by others, and having multiple view ports would allow users to compare two models side-by-side.

Improvements on the calibration procedures are dependent upon the quality of the models and the imaging equipment. Therefore, as improvements are made in the physical aspects of photoelastic analysis, modifications of the analytical functions of the calibration must reflect these changes. Additionally, future work may look at attempting

to differentiate the calibration of tensile and compressive strains within the photoelastic model. Tension and compression areas of simple 2D models can be located by a relatively inexperienced user, however in complex 3-D models this may be difficult even for the highly experienced user. It may also be beneficial to calibrate the images before importing the data into the solid modeling and analysis procedures. Memory requirements would be cut by two thirds by calibrating RGB values to individual stress values.

Changes to the Vtk library will also provide a benefit to the analysis program. Currently image reading functions of BMP and other PC-based formats are very limited and out dated. Future versions of Vtk should include additional interfacing formats making it more user friendly to the PC programmers. Difficulties associated with resampling and reslicing the dataset will also need to be looked at in more detail. Currently, the resampling process changes the data through gradient and averaging calculations. Additional functions of volume segmenting, in the resampling process, allowing users to identify small regions of interest that should not be included in the resampling would also be a benefit. Time spent on improving Vtk's online help functions will also benefit future additions and debugging. Online help has proven to be extremely useful, however explanations are often vague and misleading. Reading object oriented code to identify all of the possible functions of a class while trying to determine which functions act on which data types is time consuming. Simply listing all functions inherited from one class to another as well as re-writing some of the function explanations describing their purpose would be a great benefit to future programmers.

6.2.2 Hardware

As funds become available hardware upgrades could increase the accuracy and minimize analysis time. Imaging could be improved by changing the acquisition board from an Osprey (general purpose) board to a Matrix (advanced imaging) board. The Osprey board is an inexpensive board design that modifies colors to view skin tones of people in telemedicine applications. Changing to a board that is designed to measure the actual colors of objects will improve the accuracy of the recorded image and also provide a means of comparing image data with that recorded by other investigators. Camera, light source, and polariscope changes could increase accuracy, however the current devices are of good quality and do not appear to be causing any problems. Additional upgrades of the computer speed, memory, and a graphics accelerator could drastically change the analysis speed. Increasing memory will allow more information to be processed, thereby minimizing information loss during resampling.

6.2.3 Model Preparation

Most of the error in the analysis process is caused by the surface texture of the model slices. Ideally, all retardations of the light waves should be generated by the deformation of the polymer chains within the model and not by the surface features. Polishing the slices to remove machine tool marks will reduce the surface reflections and refractions while increasing the amount of light transmitted through each slice. This should minimize error as well as amplify the signal recorded, hopefully making the calibration procedure less complex. Currently, no equipment is available for polishing the model slices, however future plans do include the purchase of a polishing wheel.

6.2.4 Further Testing

Testing has already been completed for two additional load patterns of the two glenoid component designs of this study. Future work will begin with analysis of those models. From there testing can be conducted on other types of glenoid component designs for comparison, or the analysis can be used on any 3-dimensional photoelastic problem. Testing a scapula model with no implant inserted would provide information as to how loads are normally distributed in healthy individuals. Comparing stress region changes from intact to the instrumented data would be useful in determining cause of implant failure. Stress region changes could be compared to bone resorption changes indicated in CT reconstructions associated with failed components. Component design changes could then be altered to minimize specific stress concentrations or stress shielding.

Following the work of Ramesh[39] enhancements to the testing can be made by combining finite element analysis and photoelastic analysis. Photoelastic models could relate information of boundary conditions and regions of stress concentrations. Then finite element meshes could be constructed from the photoelastic data to obtain more accurate results. Another method by Ramesh[42] combines a volume rendering technique with image acquisition to obtain more detailed information in high stress gradient regions. Optically enhanced tiling divides an image into subsections. If additional information is needed for a given subsection it is divided again with each region now being related to a subsection of a magnified image. The purpose is to minimize the data used by starting with the largest possible grid size and reducing it only in areas that require more definition.

REFERENCES

1. Ajovalasit, A., Barone, S., Petrucci, G., "Towards RGB photoelasticity: Full-field automated photoelasticity in white light", Exp. Tech., 3 ,(35): 193-201, 1995.

2. Brown, A.F.C., Hickson, V. M., "Improvements in photoelastic technique obtained by using a photometric method", British Journal of Applied Physics, (1):, 29-44, 1950.

3. Bulletin 315-A, Student Manual on the Photoelastic Coating Technique, Measurements Group, Inc. Raleigh, NC, 1984.

4. Chapman, M.W., Operative Orthopaedics, Second Edition, Volume 2, J.B. Lippincott Company, Philadelphia, Pennsylvania, 1998.

5. Dally, J.W., Riley W.F., Experimental Stress Analysis, New York: McGraw-Hill, 1978.

6. Del Schutte, H.,Jr., Salehi, A., Goldstein, W. M., Weaver, K. D., "Finite Element and 3D Photoelastic Strain Analysis of Polyethylene at Different Degrees of Abduction", A Sientific Ehibit at the 1998 AAOS Meeting, New Orleans, Lousiana.

7. Elvins, T.T., "A Survey of Algorithms for Volume Visualization" Computer Graphis, vol. 26, no. 3, August, 1992.

8. Foley, J.D., van Dam, A., Feiner, S.K., Hughes, J.F., Computer Graphics Principles and Practice, Second Edition, Addison-Wesley Publishing Co., Reading, Massachusetts, 1990.

9. Friedman, R.J., LaBerge, M., Dooley, R.L., O'Hara, A.L., "Finite element modeling of the glenoid component: Effect of design parameters on stress distribution", J. Shoulder Elbow Surg., Sept., 261-270, 1992.

10. Frocht, M. M., Hui, P., Landsberg, D., "The use of photometric devices in the solution of the general three-dimensional photoelastic problem",Opt. Commun, (XII): 181-190, 1954.

11. Gamma, E., Helm, R., Johnson, R., Vlissides, J., Design Patterns Elements of Reusable Object-Oriented Software, Addison-Wesley Publishing Company, Reading, Mass., 1996.

12. Haake, S.J., Patterson, E.A., "Photoelastic analysis of frozen stressed specimens using spectral-contents analysis", Exp.Mech., (9): 266-272, 1992.

13. Haake, S.J., Patterson, E.A., Wang, Z.F., "2D and 3D seperation of stresses using automated photoelasticity", Exp. Mech., 3 (36):269-277, 1996.

14. Haake, S.J., Wang, Z.F., Patterson, E.A., "Evaluation of full field automated photoelastic analysis based on phase stepping", Exp. Tech., (11): 19-25, 1993.

15. Han, B., Wang, L., "Isochromatic fringe sharpening and multiplication", Exp. Tech., (11): 11-13, 1994.

16. Huang, Y.M., Lin, C.H., Suhling, J.C., Rowlands, R.E., "Determining the three individual stress components from measured isochromatic fringes", Exp. Tech., (12): 310-318, 1991.

17. Johnson, D. R., Stress Distributions Induced by Total Shoulder Replacement Glenoid Components: Intrumentation and Design, Mechanical Engineering Master's Thesis, University of North Carolina at Charlotte, 1998.

18. Karduna, A.R., Williams, G.R., Iannotti, J.P., Williams, J.L., "Total shoulder arthroplasty biomechanics: A study of the forces and strains at the glenoid component", Journal of Biomechanical Engineering, vol 120, Feb., 1998.

19. Kenny, B., Patterson, E.A., "The use of a microdensitometer in photoelastic analyses", Strain, (2): 13-18, 1985.

20. Kihara, T., Unno, M., Kitada, C., Kubo, H., Nagata, R., "Three-dimensional stress distribution measurement in a model of the human ankle joint by scattered-light polarizer photoelasticity", Applied Optics, (24), 20: 3363-3367, 1985 and (26), 4: 643-649, 1987.

21. Kohan, S.G., Programming in ANSI C, Hayden Books, Indianapolis, Indiana, 1989.

22. Lacroix, D, Prendergast, P.J., "Stress analysis of glenoid component designs for shoulder arthroplasty", Proc Insts Mech Engrs, vol 211 part H, 467-474,1997.

23. Lacroix, D, Prendergast, P.J., "3D Finite element analysis of glenoid prostheses for total shoulder arthroplasty", 11th onference of the ESB, July 8-11, Toulouse, France, 1998.

24. Levoy, M., "Display of Surfaces from Volume Data", IEEE Computer graphics and Applications, vol. 8, no.3, pg. 29-37, March, 1988.

25. Levoy, M., "Volume Rendering, A Hybrid Ray Tracer for Rendering Polygon and Volume Data", IEEE Computer graphics and Applications, vol. 10, no. 2, pg. 33-40, March,1990.

26. Levoy, M., "Efficient Ray Tracing of Volume Data", ACM Transactions on Graphics, vol. 9, no. 3, pg. 245-261, July, 1990.

27. Liu, X., Yu, Q., "Some improvement on digital fringe - multiplication methods", Exp. Tech., (1): 26-29, 1993.

28. Lorensen, W.E., Cline, H.E., "Marhing Cubes: A High Resolution 3D Surface Construction Algorithm", Computer Graphics, vol. 21, no. 4, pg. 163-169, July, 1987.

29. Morimoto, Y., Fujisawa, M., "Fringe-pattern analysis by phase shifting method using extraction of characteristic", Exp. Tech., (7): 25-29, 1996.

30. Morimoto, Y., Morimoto, Y. Jr., Hayashi, T., "Separation of isochromatics and isoclinics using Fourier transform", Exp. Tech., (9):, 13-17, 1994.

31. Mueller,R.K., Saackel, L.R., "Complete automatic analysis of photoelastic fringes", Exp. Mech., (7): 245-251, 1979.

32. Manfuz, H., Wong, T.L., Case, R.O., "Photoelastic data acquisition by the digitizing technique over arbitrary boundries", Exp. Tech., (5): 39-42, 1992.

33. Ning, P.T. Peng, W.L., "Automatic analysis of Moire fringe patterns by using an image - processing system", Exp. Mech., (12): 350-355, 1988.

34. Orr, T.E., Carter, D.R., Schurman, D.J., "Stress analysis of glenoid component designs", Clinical Orthopaedics and Related Research, no. 232, July, 217-224, 1988.

35. Patterson, E.A., Wang, Z.F. "Towards full field automated photoelastic analysis of complex components", Strain, 27: 49-53 1991.

36. Poon, C.Y., Kujawinska, M., Ruiz, C., "Automated fringe pattern analysis for Moire interferometry", Exp. Mech., (10): 234-241, 1993.

37. Ramesh, K., Deshmukh, S.S., "Three fringe photoelasticity - use of colour image processing hardware to automate ordering of isochromatics", Strain, (8): 79-87 (1996).

38. Ramesh, K., Kelkar, A.A., "Automatic ordering of isochromatic fringes - a new methodology", Strain, (8): 95-99, 1995.

39. Ramesh, K, Pathak, P.M., "Role of photoelasticity in evolving discretization schemes for FE analysis", Experimental Techniques, July: 36-38, 1999.

40. Ramesh, K., Pramod, B.R., "Digital image processing of fringe patterns in photomechanics",Optical Engineering, 31 (7): 1487-1499, 1992.

41. Ramesh, K., Ganesan, V.R., Mullick, S.K., "Digital image processing of photoelastic fringes - a new approach", Exp. Tech., (9): 41-47, 1991.

42. Ramesh, K, Sreedhar, D., "Optically enhanced tiling (OET) in digital fringe pattern analysis", Strain, Nov., 127-131, 1998.

43. Redner, S., "New automatic polariscope system", Exp. Tech., (12): 486-491, (1974).

44. Redner, A.S., "Photoelastic measurements by computer-assisted spectral-contents analysis", Exp. Mech., (5): 148-153, 1985.

45. Schroeder, W., Martin, K., Lorensen, B., The Visualization Toolkit An Object-Oriented Approach to 3D Graphics, Prentice Hall, Upper Saddle River, NJ, 1996.

46. Schroeder, W., Martin, K., Lorensen, B., The Visualization Toolkit An Object Oriented Approach to 3D Graphics, 2nd Edition, Prentice Hall, Upper Saddle River, NJ, 1998.

47. Schroeder, W., Martin, K., The Visualization Cookbook vtk User's Guide, Distributed by the authors, December 1998.

48. Seguchi, Y., Tomita, Y., Watanabe, M., "Computer - aided fringe - pattern analyzer - A case of photoelastic fringe", Exp. Mech, (10): 362-370, 1979.

49. Stone, K.D., Grabowski, J.J., Cofield, R.H., Morrey, B.F., An, K.N., "Stress analysis of glenoid components in total shoulder arthroplasty", J Shoulder Elbow Surg., March, 151-158, 1999.

50. Toh, S.L., Tang, S.H., Hovanesian, J.D., "Computerized photoelastic fringe multiplication", Exp. Tech., 21-23, 1990.

51. Tuy, H.K., Tuy, L.L., "Direct 2-D Display of 3-D Objects", IEEE Computer Graphics and Applications, vol. 4, no. 10, pg. 29-33, October 1984.

52. Upson, C., Keeler, M., "The V-Buffer: Visible Volume Rendering", Computer Graphics, vol. 22, no. 4, pg. 59-64, July, 1990.

53. Voloshin, A. S., Burger, C.P., "Half-firnge photoelasticity: A new approach to whole-field stress analysis", Exp. Mech., (9): 304-313, 1983.

54. Voloshin, A.S., Redner, A.S., "Automated measurement of birefringence: Development and experimental evaluation of the techniques", Exp. Mech., (9): 252-257, 1989.

55. Wasowski, J.J., Wasowski, L.M.,"Computer - based optical differentiation of fringe patterns", Exp. Tech., (3): 16-18, 1987.

56. Watt, A., 3D Computer Graphics, Addison-Wesley, Reading, Mass., 1995.

57. Welch, B. B., Practical Programming in Tcl and Tk, Second Edition, Prentice Hall, Upper Saddle River, NJ, 1997.

58. Wyvill, G., McPheeters, C., Wyvill, B., "Data Structure for Soft Objects", The Visual Computer, vol. 2, no. 4, pg. 227-234, August, 1986.

59. Yao, J.Y. "Digital image processing and isoclinics", Exp. Mech., (9): 264-269, 1990.

60. Yao, J.Y., "Extracting isoclinics using image - processing techniques", Exp. Tech., (10): 30-31, 1987.

61. Zandman, F., "Photostress analysis", Product Engineering, (3): 1959.

APPENDIX A: CODE

A.1 C++ CODE

Most of the functions needed for the analysis program were provided by vtk,

however additional functionality was obtained by writing the functions

"vtkExtractTransformVOI", "vtkCreateYData", "vtkCreateZData", "vtkMerge3Filter",

and "vtkEffectiveStress". Calibration functions will be described separately.

A.1.1 vtkExtractTransformVOI

Purpose: To input a structured points dataset and a transformation matrix to produce a

subsampled dataset and/or a transformed data set. The output of the filter is also a

structured points dataset.

Description: The ExtractTrasformVOI function is simply a modification of the vtk

function ExtractVOI for extracting a subvolume of a dataset. By adding a matrix input

and transformation subroutines, the function can be changed to provide transformations

to an extracted volume. Adding a protected vtkMatrix4x4 pointer to the class and

creating the subroutine "void SetMatrixPointer(vtkMatrix4x4 *inMatrix)" provides the

transformation matrix input. A copy of the input matrix is made to avoid deleting the

original in case it is used by other functions. Setting the volume of interest with the

"SetVOI" command may be used to perform subsampling if a reduced dataset is desired.

The "VOI" is a six element array containing the minimum and maximum values of the x,

y, and z component of the structured grid respectively. Transformations are computed by

indexing through the structured grid with the defined output VOI variables and calling a

reverse transform to sample the original dataset. Input matrices were retrieved from a

transformation of an object to a desired location. Transformation of the data contained within a structured grid dataset without moving the dataset itself requires a subroutine that will index through the grid and transform the data from the input dataset. The reverse transform is obtained by multiplying each of the indexed output grid locations by the inverse of the input matrix and using the result as the index to the original dataset. Once the input and output indexes are obtained the data is copied from one point to another using the "CopyData" command.

A.1.2 vtkCreateYData and vtkCreateZData

Purpose: To provide a rotation of the data contained within a structured grid data set without changing the grid.

Description: The CreateYData and CreateZData functions are similar to the ExtractTransformedVOI providing a specific rotation instead of a user defined translation. Data is input and output in the same structured grid pattern. The rotations are not y and z-axis rotations, but are rotations about the y and x axis needed to produce the correct orientation of volumes constructed from image slices perpendicular to the y and z-axis. Rotations of the data contained within the grid are performed by indexing the output grid, determining the appropriate index to the input, and copying the data. For the Y rotation the input index is obtained by switching the x and z-axis and indexing the x-axis in reverse of the output index. Z rotation is obtained by switching the y and z with reverse indexing of the y-axis.

A.1.3 vtkMerge3Filter

Purpose: To create a class that would accept three image cache or structured points dataset inputs and one structured points output.

Description: There are no classes that allow three inputs in vtk, so this class was developed to serve as a parent class for the "vtkEffectiveStress" function. The filter defines three commands "SetInputX", "SetInputY", and "SetInputZ". Pointers from the input datasets are stored as protected "vtkDataObject" pointers. The difficulty of creating this class is the need to check modified times of all inputs with the execution time of the filter to assure data is up to date and the output can be computed.

A.1.4 vtkEffectiveStress

Purpose: To read in three view dependent data volumes, calibrate every data point contained within the volumes, and calculate an effective stress for each point of an equal sized output volume.

Description: The effective stress calculation should be made after all the data has been appropriately aligned and each data set is of the same size and spacing. The three volumes of image slices representing the three orthogonal views of the same model are input through the inherited commands "SetInputX", "SetInputY", and "SetInputZ". Dimensions and spacing are checked to be sure that all are equal. If no errors are encountered the output data file is passed the dimensions, spacing, and origin of the x-input and the data analysis begins. An index is started to access every point within the structured grid defined by the input datasets. For each element of the grid corresponding points of the three input grids are accessed. The RGB components of each input point is then passed to a calibration function that determines a stress value corresponding to the magnitude of the principal stress differences for that point. After determining a stress value for each of the three input points an effective stress is calculated as defined previously in Equation 2. At this time principal stress angles are assumed to be aligned

with the viewing axis. Future improvements will add an isoclinic analysis to increase the

accuracy of the effective stress calculation. Effective stress values are then inserted into

the element of the output structured grid corresponding to the element of the input being

analyzed.

A.2 Tcl/tk CODE: User Interface

Once the low-level C++ functions have been tested they are combined into

dynamic linked libraries (.dll) and accessed from the scripting code of tcl. A menu driven

windows environment program, "Automated Photoelastic Object Modeling", was then

developed using the combination of tcl and tk to provide a graphical user interface to the

C++ functionality. The program design is based on the program "Decimate.tcl" that was

provided with vtk. Upper level menu selections include "File, Edit, View, Tools, and

Options" each of the associated submenus and commands will be described below.

A.2.1 File

The File menu contains the typical Open and Save and Exit commands. Open

functions are currently hard coded to open multiple slices composing each of the three

view dependent volumes. Once an open command is called the series of images

corresponding to that volume are opened and passed through the pipeline described in

figure 2a to create a polydata set which is then rendered. Save functions allow the user to

specify the type of output file and the filename.

A.2.2 Edit

After opening a volume and creating a polydata set several functions become

available in the edit menu. The move command allows the current active volume to be

translated along each of three axis. The magnitude of the translations are limited because

the transformations are applied to the input dataset without changing the dataset indices.

Therefore, if the image is translated beyond the extents of the image boundaries the data

extending beyond the boundaries will be deleted. Other functions of the edit menu such

as smoothing and cleaning provide various enhancements to polydata sets and should

only be applied to the final effective stress model.

A.2.3 View

The view menu contains all commands associated with objects displayed on the

screen and how they are displayed. An active volume is used in the program to represent

the dataset that functions act upon. Switching between the objects displayed on the

screen is accomplished by setting that volume as the active volume through the view

menu commands. Similarly, objects stored in the program may be viewed or hidden at

any time through a Hide command. The active volume may be set to view wireframe,

boundary, or surface edges using the Surface command. Several built in view directions

are listed under the Direction command if an exact view direction is desired. Otherwise

rotations of the camera around the objects is accomplished by holding the right mouse

button down and moving the cursor. The left mouse button controls zoom features of the

rendered scene. Together these functions allow the user to visually accomplish

alignments by selecting the objects that a move command will operate on, and view or

hide additional objects as necessary to complete the alignment process.

A.2.4 Tools

Currently the Effective Stress calculation is the only command associated with

the tools menu. After completing the alignment process and the outer contours of the

three view dependent volumes are positioned appropriately the Effective Stress command

will calibrate each dataset into stress values and combine the stresses using the effective

stress calculation. Predetermined threshold values will then be used to separate the

effective stress volume into stress contours. Each contour will be designated a color and

opacity allowing internal contours to be viewed through semitransparent external

contours.

A.2.5 Options

Functions contained within the Options menu are unchanged from the original Decimate

program that was used as a starting point for this program. These functions provide

object and background color changes as well as viewing splitting the render window into

two separate views. In the future these functions will be modified to allow comparison of

two effective stress volumes simultaneously.

A.3 Compiling Keys

There were also several compiling tasks that are worth noting. Vtk can be

downloaded from the internet now in version 2.4 from "http://www.kitware.com".

Installing instructions can be found at the same site and should be read before attempting

to install. Some installations now provide a precompiled vtkdll.dll, however usually this

will need to be recompiled on the computer where it is installed. In a Windows based PC

compiling is conducted in a DOS window. The directory should be changed to the vtkdll

directory that was created during the install. Connections to the compiler are made by

setting the path command. For Microsoft C++ the path is usually the same

"path=c:\progra~1\devstu~1\vc\bin;c:\progra~1\devstu~1\shared~1\bin". This should be

typed with no spaces, and if located in another directory with a filename greater than six

characters then the "~1" should be used. When adding a new function to vtk then the

makefile within the vtkdll directory should be modified to include the name of the file and locations of other files that will be linked to the function. Before compiling it is important to look in the subdirectory "obj" to see if there is an object file associated with any file that has been modified. If so it should be deleted along with any "vtkdll" files remaining in the "vtkdll" directory. When ready to compile type "nmake" and the compiling process should start. When adding a new vtk function to tcl/tk library similar steps should be taken. The makefile for vtk in tcl is "vtktcl.dll" located in the vtktcl directory. New files should be added to the makefile as well as several "vtktcl.cxx" and "vtkgraphics.h" files located in various subdirectories. It is important to remove any old object and "vtktcl" files before executing the nmake command.

When adding commands to Tcl there are several locations in the software where edits are required. First, new functions should be saved in one of the current vtk directories for easiest referencing. Functions can be placed in other directories, however additional changes to the makefiles as well as the C++ compiler settings make this step a little more time consuming. Then finding a vtk....Tcl.cxx file from the vtkbin\vtktcl\src directory with methods close to the function created. Copy the file naming it vtkMyFileNameTcl.cxx and search inside the file changing any reference of the name of the previous function to the new function. At the end of the file is a list of the commands which needs to be changed to the commands of the new function. Next locate the files vtktcl.cxx (there may be several), graphics.h, and the makefiles in the vtkbin\vtktcl and vtkbin\vtkdll. Again search these files for any reference to the previous function copy what is listed and edit it to work with the new function, its location, and any included files. Then delete any files named vtkdll or vtktcl in the \vtkdll and \vtktcl directories. If

changing a previously compiled function the associated .obg file in the \src directories

needs to be deleted. Then in DOS mode set the path for the compiler and run nmake

inside the \vtkdll and \vtktcl directories.