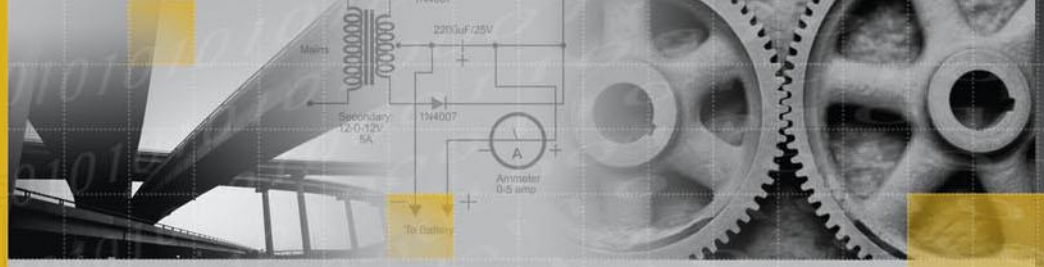
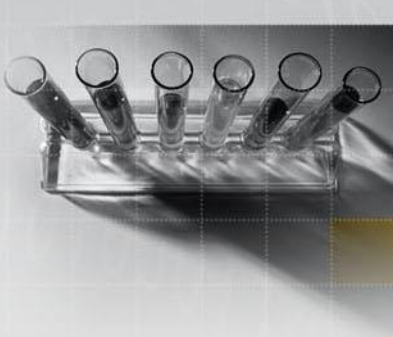


Two-Dimensional, Single Subset based Digital Image Correlation Using the Inverse Compositional Lucas-Kanade Algorithm

Devan Atkinson
17732913



2017

FAKULTEIT INGENIEURSWESE
FACULTY OF ENGINEERING

UNIVERSITEIT STELLENBOSCH UNIVERSITY

Unifying Framework for Gradient-Descent, Subset Based Digital Image Correlation and Digital Volume Correlation

MEng Research Thesis Proposal

Supervisor: Dr Thorsten Becker

Devan Atkinson

Department of Mechanical and Mechatronic Engineering

2017

Department of Mechanical and Mechatronic Engineering
Stellenbosch University

Declaration

I know that plagiarism is wrong.

Plagiarism is to use another's work (even if it is summarised, translated or rephrased) and pretend that it is one's own.

This assignment is my own work.

Each contribution to and quotation (e.g. "cut and paste") in this assignment from the work(s) of other people has been explicitly attributed, and has been cited and referenced. In addition to being explicitly attributed, all quotations are enclosed in inverted commas, and long quotations are additionally in indented paragraphs.

I have not allowed, and will not allow, anyone to use my work (in paper, graphics, electronic, verbal or any other format) with the intention of passing it off as his/her own work.

I know that a mark of zero may be awarded to assignments with plagiarism and also that no opportunity be given to submit an improved assignment.

I know that students involved in plagiarism will be reported to the Registrar and/or the Central Disciplinary Committee.

Name: .....

Student no: .....

Signature: .....

Date: .....

Abstract

Digital Image Correlation (DIC) and Digital Volume Correlation (DVC) are becoming widely used tools, in the field of material science, to measure the displacements and deformations of specimens. However the high cost of commercial DIC and DVC software and the limited control offered over the correlation process in both commercial and open-source software limits its widespread adoption. This paper proposes a project which aims to create a Matlab based program capable of performing 2D DIC and DVC on a given set of images while allowing the user control over the correlation process. It is hypothesised that the different DIC algorithms use different methods to perform the same tasks in the correlation process. Thus by analysing these algorithms it is possible to combine these different methods into one program which will allow the user to select which methods to use for the various correlation tasks. If successful, this project will produce a freely available program, which offers many of the correlation methods of the different DIC algorithms, so that users will be able to perform in depth DIC and DVC analyses.

Contents

List of Figures

Chapter 1

Introduction

1.1 Background

Digital Image Correlation (DIC) and Digital Volume Correlation (DVC) fall into the discipline of computer vision which aims to extract quantifiable information from images. This quantifiable information that DIC extracts is the optical flow that occurs between two images. It is capable of relating this optical flow to the displacements and deformations of the physical objects captured in the images. Thus it constitutes a tool for determining the displacements and deformations of 3D objects from images taken of these objects. Since its introduction its has found widespread use in many applications including measuring vein deformations, aerospace testing and obstacle avoidance for unmanned aerial vehicles to name a few.

Although there are other optical flow techniques; DIC and DVC are the most accurate of these methods which has allowed them to be successfully used in the field of material science where high accuracy is important. Material science mainly focuses on quantifying the characteristics of materials and using these quantified characteristics to predict material behaviour. These quantified characteristics are referred to as material properties which serve as parameters in constitutive equations. A constitutive equation is a mathematical relations which approximates the response of a material, such as deformation, to external stimuli, such as a force; where the material property determines the degree of the response of the material.

Thus these constitutive equations mathematically define the material properties such that they can be quantified. These constitutive equations and material properties can be used to predict the behaviour of a component to an external force that the component is expected to withstand during its use. The resulting behaviour that is predicted can then be used to determine whether the component is susceptible to failure during its intended use case.

However accurate material properties are required in order to reliably predict the response of a component to external stimuli. Mechanical material properties are determined by applying a force to a specimen, made from a specific material, and measuring the displacements or deformations that occur. Then these forces and resulting displacements or deformations are substituted into the constitutive equations so that the value of the mechanical material property can be solved for.

Therefore DIC is applied in the field of material science as a tool for measuring the displacements and deformations of specimens as they are being loaded. Then these displacements and deformations are used to determine material properties. Other methods of measuring displacements and deformations exist however DIC offers advantages over these

methods as is discussed in the next section.

This project focuses on researching DIC and DVC with the goal of creating Matlab program capable of determining the displacements and deformations of specimens from images captured of the specimens. The program is specifically aimed at material science applications and is intended to provide the user with full control over the correlation process. Thus the program is intended to become an open-source program which unifies multiple methods of performing correlation so that the user can choose the correlation configuration that suites them best or edit and create their own. This is valuable in the DIC and DVC community since currently limited software packages offer this functionality.

This proposal outlines the motivation behind this project and the problem to be solved. Objectives have then been proposed based on this and a research plan has been developed around these objectives. A time-line for the research plan and a budget are also provided.

1.2 Motivation

In order for the predicted material response to be reliable the material properties used must be accurate. However most conventional methods of measuring displacement and deformation can only do so at a specific point on the specimen. Thus this local measurement needs to be representative of what is happening over the whole material for it to provide a means of accurately determining a material property.

As such well developed standards for determining material properties have been created which document the experiments used to determine these properties. These standards place strict conditions on the specimens used, the measurement techniques employed and the calculations involved in determining material properties. The specimen is designed in such a way that the material property under investigation plays a significant role in the deformation and the measurement technique is employed at a location on the specimen which reliably represents the response of the material as a whole.

Although these standards provide reliable results they require multiple tests to ensure a consistency in the results, they usually require specimens of considerable size and only one material property can be determined per an experiment. These shortcomings arise due to the limitation of the conventional displacement and deformation measurement techniques employed.

In contrast DIC and DVC are full-field measurement techniques which means that they provide displacement and deformation data across the full-field of the specimen. This means displacement and deformation data over the whole surface (DIC) or whole volume (DVC) within the field of view. This is significant since it allows more complex methods of determining material properties by using full-field deformation data to be used. These methods of material property determination are attractive because they can determine more than one material property from a single experiment if the specimen is designed well. For example Grédiac [6] used DIC in conjunction with the Virtual Fields Method to determine multiple material properties of a composite from a single experiment.

Furthermore since DIC only requires images of the specimen to be taken during the experiment it is classified as a non-contact measurement method. This is advantageous since it allows this method of measurement to be used even when the specimen is exposed to a harsh environment during the test. For example if the specimen is exposed to high temperatures in a furnace then images can still be captured through a window.

Additionally, since these alternative methods of material property determination are capable of determining material properties using a wide range of specimen sizes and shapes, there has been much interest in using it to quantify the condition of operating components so that the failure of these components can be avoided [4]. This is done by cutting a small piece of material from the component and testing it to determine its material properties. These

material properties are then compared to the component's original material properties to determine how the condition of the component has deteriorated during its use. This research is being done within the Stellenbosch University's material science research group to help Eskom analyse the current working condition of their power stations so that shut-downs due to maintenance can be minimized.

Thus it is clear that DIC offers many advantages in the field of material science which is why it is becoming so widely adopted as a measurement technique.

Chapter 2

Problem statement

The rapid improvement in performance of computers and cameras with decreasing costs has made the necessary equipment for DIC more accessible. This coupled with the continuous improvement of DIC algorithms over the last 30 years, since its introduction, would suggest that this tool for displacement and deformation measurement shall become more widely adopted and accepted.

However there has been factors that have limited its use chief among which is the high cost of commercial software which puts it out of the reach of many research institutions and businesses that would otherwise benefit from its use. Open-source software is available but it has limitations in that the algorithm is set up in a specific way which cannot be altered. For example Ncorr (open-source DIC software) interpolates using a spline function and the user cannot use any other method.

This has caused users to view DIC and DVC as a sort of black box over which one has limited control. This is disadvantageous since it prevents users from developing a deeper understanding of DIC and DVC which would have otherwise allowed them to make better use of the tool. It also brings the reliability of the displacement results into question since how can the displacement and deformation results be trusted if the correlation method used is unknown.

Control over the correlation process refers to being able to choose what method is used to perform a specific task of the correlation process. The different algorithms that have been proposed over the last 30 years differ predominantly as a result of them using different methods to perform these specific tasks of the correlation process. Thus, by creating a code that allows a user to choose from the different methods of performing the correlation tasks, the code will essentially be able to incorporate many of the correlation codes into one. Some of the common algorithms to perform correlation are Lucas-Kanade [7], Farnebäck [5] and Uras [11].

Control over the correlation process is important since different algorithms perform better in certain situations since many algorithms involve a trade-off between efficiency and accuracy. Furthermore parameters that control the implementation of the algorithm such as subset size, allowable subset deformations and interpolation strategy can also have a significant impact on the results.

Thus it can be seen that there is a need for a freely available DIC program which offers multiple DIC correlation options and allows the user to control the parameters that affect the implementation of the chosen algorithm.

Chapter 3

Literature study

3.1 Overview of DIC and DVC

DIC and DVC are techniques which quantify the optical flow between images then relate the optical flow observed to displacements in the 3D world. Optical flow is the apparent motion of an object in an image caused by the actual motion of the object in the 3D world.

Thus these techniques solve two problems in order to determine displacements of 3D objects from image of the 3D objects. The first problem is determining the optical flow that is present between images. This is determined by identifying a unique pattern contained within a cluster of pixels, with these pixels representing a part of the 3D object, in the reference image and finding the same pattern within a cluster of pixels in the deformed image. Then the difference in position of these pixel clusters within the two images can be used to determine the displacement that this part of the 3D object underwent. Note that at this stage the displacement is in units of pixels.

This process is referred to as correlation. The correlation techniques focused on in this project use gradient-descent methods to determine the location of a pixel cluster, in the deformed image, that matches a cluster in the reference image. Additionally since they rely on tracking pixel clusters, called subsets, these algorithms are referred to as gradient-descent, subset based DIC and DVC.

The second problem to be solved is relating the displacement in units of pixels to the actual displacement in the 3D world in metric units. In order to do this a method of mathematically relating the 2D information stored within an image to the 3D scene that it captures is needed. This mathematical relation is well known and is discussed in section 3.5. The actual problem is to determine the value of the parameters that are involved in this mathematical equation. These are solved for by relating the known coordinates of a 3D object to the coordinates of the object within the image using this mathematical relation and solving for the parameters. This is an inverse problem and this process is referred to as calibration. Calibration also has to take into account distortions caused by the camera.

Therefore two image sets are required in order to perform DIC or DVC analysis. The first image set contains images of an object of known coordinates such that the calibration process can be performed. The second image set must contain images taken of an object as it is being deformed or displaced so that these images can be used for correlation.

3.2 Correlation

DIC is used in material science to determine the displacements that the surface of a test specimen experiences when it is deformed by an applied load. First pictures of the speckle

pattern applied to the surface of the specimen are taken before and after deformation. Then a cluster of pixels that contains a unique pattern in the reference image is identified and the same pattern in a cluster of pixels is found within the deformed image. By comparing the position of this unique pattern in the reference image to its position in the deformed image the displacement of the pattern can be determined.

The cluster of pixels is referred to as a subset. A subset is used instead of individual pixels because a single pixel in the reference image is likely to have multiple matches in the deformed image since any pixel with the same grayscale value in the deformed image is a possible match. Therefore a subset of pixels is used since it contains more information and is therefore more likely to contain a unique pattern that can be more definitively identified in the deformed image.

3.2.1 Correspondence problem and speckle patterns

Although subsets of pixels are usually easier to track than individual pixels, using subsets leads to a new issue. The correspondence problem is to do with the inability to track displacements as a result of the grayscale patterns within the image. To understand this consider the aperture problem which is a special case of the correspondence problem.

The aperture problem involves the ambiguity in attempting to track the motion of a one-dimensional spatial structure (such as a line) when viewed through an aperture (such as when considering a subset of pixels) that cuts off the view of the ends of the one-dimensional spatial structure [2]. This can be illustrated by considering the difficulty in tracking the motion of a line in the direction along the line when the end points of the line are not visible.

Similarly if the pattern in the image to be tracked is a repeating pattern with a constant grid spacing then the displacement cannot be uniquely determined since the image will contain the same pattern repeated. Thus the algorithm will determine the displacement to be within a multiple of the grid spacing from the actual displacement unless the subset contains a unique feature such as being at the edge of the specimen.

Thus it is clear that the patterns to be tracked in the images can have a significant impact on the performance of the correlation process. To avoid the aperture problem is is clear that the pattern should be isotropic such that it can be tracked equally well in all directions. Additionally to solve the correspondence problem the pattern should also be highly random and non-periodic so that each subset of an image contains a unique pattern.

This is achieved by applying a random speckle pattern to the surface of the specimen to be tested since these provide very unique patterns which are isotropic in nature. Additionally the high information density enables smaller subsets to be used than would otherwise have been possible.

3.2.2 Interpolation

Different applications of DIC place different requirements on the DIC algorithm. Within material science applications one of the main requirements is that the displacements be tracked accurately to a very high resolution. For example to create an accurate stress-strain curve for common engineering materials a resolution on the order of $10^{-5}m/m$ is required [9]. This thus requires displacements to be determined to sub-pixel resolutions.

The issue with this is that this requires the light intensity to be continuous whereas digital images are discrete. Interpolation is used to determine the light intensities between pixels so that an approximation to the continuous form of the intensity pattern on the surface of the specimen can be calculated.

3.2.3 Warp function

An additional requirement of applying DIC to material science is that the deformation characteristics of the materials should be taken into account. The deformation of the material causes the speckle pattern on its surface to deform in the same way. This is an issue since the pattern contained within a subset of the reference image will be deformed in the second image which can cause difficulties in matching the two subsets.

This is solved by allowing for the subsets to deform in a similar way to that of the material by defining the allowable deformations in a warp function. The purpose of these warp functions is to transform the pixel coordinates of the reference subset so that the resulting pattern is closer to the pattern in the deformed image. A common warp function which allows for affine transformations is given by

$$W(\boldsymbol{x}, \boldsymbol{p}) = \begin{bmatrix} x_{warp} \\ y_{warp} \end{bmatrix} = \begin{bmatrix} x + u + \frac{\partial u}{\partial x}\Delta x + \frac{\partial u}{\partial y}\Delta y \\ y + v + \frac{\partial v}{\partial x}\Delta x + \frac{\partial v}{\partial y}\Delta y \end{bmatrix}. \tag{3.1}$$

Here x and y represent the position of the pixel in the original image, u and v represent the displacements in the x and y directions respectively, $\frac{\partial u}{\partial x}$ and $\frac{\partial v}{\partial y}$ represent the elongation in the x and y directions respectively, $\frac{\partial v}{\partial x}$ and $\frac{\partial u}{\partial y}$ represent the shear deformation of the subset, Δx and Δy are the distances from the centre of the subset to the pixel under consideration and x_{warp} and y_{warp} are the modified pixel coordinates after the warp has been applied.

With most subset based DIC algorithms the more complex the warp function becomes the more computationally expensive each iteration is. This is because the correlation criteria must be optimized in terms of more variables leading to more work.

3.2.4 Correlation criteria

A method of determining how well the reference subset, G, has been matched to a subset in the deformed image, F, is needed in order for a DIC algorithm to know when correlation has been accomplished. This is done through equations called correlation criteria which describe the fit between the two subsets by a single number.

Correlation criteria fulfil an additional purpose. As images are taken, as the specimen is deformed, the intensity of the images can change as a result of changes in lighting, changes in reflectivity as the material strains and so on. Although this does not cause the patterns in the subsets to change it does cause the grayscale values of individual pixels, that make up the patterns, to change. Some correlation criteria account for this by accommodating for offset in intensity values or for scaling the intensity values.

Each correlation criteria is based off of a cost function which is usually a variation of the sum of squared differences between the intensities of the reference and investigated subset. They vary in the scaling and offset that is applied to the investigated subset's intensity values.

Sum of square difference

This is one of the more basic correlation criteria since it does not adjust for lighting. It is simply the sum of squared differences between the reference subset and investigated subset.

$$\chi^2_{SSD}(G) = \sum_i \left(G_i - F_i\right)^2 \tag{3.2}$$

Zero-mean sum of squared difference

To account for an offset in light intensity a constant is added to the investigated subset's intensities. This results in the following cost function.

$$\chi^2_{ZSSD}(G) = \sum_i \left(G_i + b - F_i\right)^2 \tag{3.3}$$

The offset b can be solved for by treating it as an additional parameter of the optimisation problem. However it is preferred to determine an optimal estimate for b by taking the derivative of the cost function with respect to b and setting it to zero.

$$\frac{\partial \chi^2_{ZSSD}}{\partial b} = 2\sum_i \left(G_i + b_{opt} - F_i\right) = 0 \tag{3.4}$$

$$0 = \sum_i \left(G_i\right) - \sum_i \left(F_i\right) + b_{opt} n \tag{3.5}$$

$$\therefore \quad b_{opt} = \frac{\sum_i F_i}{n} - \frac{\sum_i G_i}{n} = \bar{F} - \bar{G} \tag{3.6}$$

Here n is the number of pixels in the subset and $\bar{F}$ and $\bar{G}$ are the average intensity values for the reference and investigated subset respectively. Substituting b_{opt} for b in the cost function equation 3.3 the correlation criteria is obtained.

$$\chi^2_{ZSSD} = \sum_i \left(\left(G_i - \bar{G}\right) - \left(F_i - \bar{F}\right)\right)^2 \tag{3.7}$$

Normalised sum of squared difference

This correlation criteria accommodates for a scale change in the light intensity of the investigate subset. This is done by multiplying the investigated subset intensities by a constant a.

$$\chi^2_{NSSD} = \sum_i \left(aG_i - F_i\right) \tag{3.8}$$

Again an optimal estimate can be found for a such that it can be eliminated from the equation. This is done by taking the derivative of equation 3.8 with respect to a and setting it to zero.

$$\frac{\partial \chi^2_{NSSD}}{\partial a} = 2\sum_i \left(a_{opt} G_i - F_i\right) G_i = 0 \tag{3.9}$$

$$a_{opt} = \frac{\sum_i G_i F_i}{\sum_i G_i^2} \tag{3.10}$$

Substitution of a_{opt} for a in equation 3.8 gives the correlation criteria.

$$\chi^2_{NSSD} = \sum_i \left(\frac{\sum_i G_i F_i}{\sum_i G_i^2} G_i - F_i\right)^2 \tag{3.11}$$

Zero-mean normalised sum of square difference

Accounting for both offset and scale of the intensities leads to the zero-mean normalised sum of square difference correlation criteria. It has the following cost function.

$$\chi^2_{ZNSSD} = \sum_i \left(aG_i + b - F_i\right)^2 \tag{3.12}$$

The optimal estimators for a and b are found by taking the derivative of equation 3.12 with respect to a and b separately and setting this to zero.

$$\frac{\partial \chi^2_{ZNSSD}}{\partial a} = 2\sum_i \left(a_{opt}G_i + b - F_i\right) G_i = 0 \tag{3.13}$$

$$\Rightarrow a_{opt} = \frac{\sum_i \left(F_i - b\right) G_i}{\sum_i G_i^2} \tag{3.14}$$

$$\frac{\partial \chi^2_{ZNSSD}}{\partial b} = 2\sum_i \left(aG_i + b_{opt} - F_i\right) = 0 \tag{3.15}$$

$$\Rightarrow b_{opt} = \frac{\sum_i F_i - aG_i}{n} \tag{3.16}$$

These can be solved to obtain

$$a_{opt} = \frac{\sum_i \bar{F}_i\bar{G}_i}{\sum_i \bar{G}_i^2}, \qquad b_{opt} = \bar{F} - \bar{G}\frac{\sum_i \bar{F}_i\bar{G}_i}{\sum_i \bar{G}_i^2} \tag{3.17}$$

$$\text{where} \quad \bar{F}_i = F_i - \bar{F} \quad \text{and} \quad \bar{G}_i = G_i - \bar{G} \tag{3.18}$$

Substituting these into equation 3.12 results in the correlation criteria.

$$\chi^2_{ZNSSD} = \sum_i \left(\left(\frac{\sum_i \bar{F}_i\bar{G}_i}{\sum_i \bar{G}_i^2} G_i - \bar{G}\frac{\sum_i \bar{F}_i\bar{G}_i}{\sum_i \bar{G}_i^2} \right) - F_i + \bar{F} \right)^2 \tag{3.19}$$

3.2.5 Subset matching

The purpose of DIC is to find the warp function parameters which optimise the correlation criteria. Multiple methods of doing this have been proposed; each with their own advantages and disadvantages. The Lucas-Kanade method is explained for the case of correlating a reference subset F to the subset under investigation G.

Lucas-Kanade

The Lucas-Kanade method is one of the most widely used subset based DIC techniques. It makes use of intensity gradients to guide the search direction to optimise the correlation criteria. A limitation of the method is that it requires the displacement between the images to be small which is common in material science applications.

The warp function parameters, p, are iteratively improved to obtain a better correlation criteria value by a method of non-linear optimization similar to that of Newton's method.

$$p^{k+1} = p^k + \Delta p \tag{3.20}$$

The equation for Δp is dependent on the warp function, the correlation criteria and the gradient descent approximation used. For illustration purposes the method will be derived using the sum of squared difference correlation criteria, with the warp function given in equation 3.1 and Guass-Newton gradient descent approximation [1]. Thus the correlation criteria to be minimized is

$$\chi^2 = \sum_{\boldsymbol{x}} \left(G(W(\boldsymbol{x}, \boldsymbol{p} + \Delta\boldsymbol{p})) - F(\boldsymbol{x})\right)^2. \tag{3.21}$$

Taking the Taylor expansion of $G(W(\boldsymbol{x}, \boldsymbol{p} + \Delta\boldsymbol{p}))$, this equation reduces to

$$\chi^2 = \sum_{\boldsymbol{x}} \left(G(W(\boldsymbol{x}, \boldsymbol{p})) + \nabla G \frac{\partial W(\boldsymbol{x}, \boldsymbol{p})}{\partial \boldsymbol{p}} \Delta\boldsymbol{p} - F(\boldsymbol{x}) \right)^2 \tag{3.22}$$

where
$$\nabla G = \begin{bmatrix} \frac{\partial G(W(\boldsymbol{x},\boldsymbol{p}))}{\partial x} & \frac{\partial G(W(\boldsymbol{x},\boldsymbol{p}))}{\partial y} \end{bmatrix} \tag{3.23}$$

and
$$\frac{\partial W(\boldsymbol{x}, \boldsymbol{p})}{\partial \boldsymbol{p}} = \begin{bmatrix} \frac{\partial x_w}{\partial u} & \frac{\partial x_w}{\partial v} & \frac{\partial x_w}{\frac{\partial u}{\partial x}} & \frac{\partial x_w}{\frac{\partial u}{\partial y}} & \frac{\partial x_w}{\frac{\partial v}{\partial x}} & \frac{\partial x_w}{\frac{\partial v}{\partial y}} \\ \frac{\partial y_w}{\partial u} & \frac{\partial y_w}{\partial v} & \frac{\partial y_w}{\frac{\partial u}{\partial x}} & \frac{\partial y_w}{\frac{\partial u}{\partial y}} & \frac{\partial y_w}{\frac{\partial v}{\partial x}} & \frac{\partial y_w}{\frac{\partial v}{\partial y}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \Delta x & \Delta y & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta x & \Delta y \end{bmatrix}. \tag{3.24}$$

Here ∇G is the gradient of the investigated subsets intensity calculated along its own coordinate frame but it is evaluated at the position given by the warp function. The optimal $\Delta\boldsymbol{p}$ can be found by taking the derivative of equation 3.22 with respect to $\Delta\boldsymbol{p}$ and setting it equal to zero.

$$\frac{\partial \chi^2}{\partial \Delta\boldsymbol{p}} = 0 = 2 \sum_{\boldsymbol{x}} \left[\nabla G \frac{\partial W}{\partial \boldsymbol{p}} \right] \left[G(W(\boldsymbol{x}, \boldsymbol{p})) + \nabla G \frac{\partial W}{\partial \boldsymbol{p}} \Delta\boldsymbol{p} - F(\boldsymbol{x}) \right] \tag{3.25}$$

$$\Delta\boldsymbol{p} = H^{-1} \sum_{\boldsymbol{x}} \left[\nabla G \frac{\partial W}{\partial \boldsymbol{p}} F(\boldsymbol{x}) - \nabla G \frac{\partial W}{\partial \boldsymbol{p}} G(W(\boldsymbol{x}, \boldsymbol{p})) \right] \tag{3.26}$$

where
$$H = \sum_{\boldsymbol{x}} \left[\nabla G \frac{\partial W}{\partial \boldsymbol{p}} \right]^T \left[\nabla G \frac{\partial W}{\partial \boldsymbol{p}} \right] \tag{3.27}$$

Using this equation an incremental improvement to the warp function parameters can be found in each iteration thereby improving the warp function parameters until the convergence criteria is acceptable optimised. This procedure is carried out for many subsets in order to determine the displacements over the full surface of the specimen.

Newton-Raphson

The Newton-Raphson method is one of the most popular methods for finding the roots of a function. Its uses the first two terms of the Taylor series approximation of a function to iteratively improve upon an initial estimate to the roots of a function. The Taylor series of function is given as

$$f(x_{i+1}) = f(x_i) + f'(x_i) \times (x_{i+1} - x_i) + \frac{f''(x_i)}{2!} \times (x_{i+1} - x_i)^2 + ... \tag{3.28}$$

The Newton-Raphson method is derived by considering only the first two terms of the Taylor series and setting the value of $f(x_{i+1}) = 0$.

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \tag{3.29}$$

Thus this method uses the tangent line of the function at the current estimate to find an improved estimate to the root of the function. The improved estimate is the intersection point of the tangent line with the x-axis. However for the purposes of subset matching this method needs to be modified to find the minimum of a function and not the root since it is unlikely that a root exists. A root to the correlation criteria does not exist since that would indicate perfect correlation which is not possible due to errors introduced in the correlation process as a result of noise in the image and interpolation used to sample light intensity values between pixel locations.

Since it is known that a minimum or maximum of a function exists where the derivative of the function is equal to zero the Newton-Raphson method is modified by using the derivative of the correlation coefficient as $f(x_i)$ in equation 3.29. In doing so the algorithm changes from a local root finding algorithm to a local minimum-maximum finding algorithm. As such, in order to ensure that the algorithm finds a minimum to the correlation coefficient, the initial guess supplied to the algorithm needs to be in the vicinity of a local minimum. Thus the Newton-Raphson minimization algorithm is given as

$$x_{i+1} = x_i - \frac{f'(x_i)}{f''(x_i)} \tag{3.30}$$

However this form of the algorithm is for an equation with one independent variable. Subset matching usually requires solving an equation with six or more independent variables. As a result the derivative $f'(x_i)$ is replaced by the Jacobian of the correlation coefficient, $\boldsymbol{J}$, and the second derivative $f''(x_i)$ is replaced by the Hessian of the correlation coefficient,$\boldsymbol{H}$, in terms of the $\boldsymbol{p}$ parameters. For the general case of $\boldsymbol{p}$ containing 6 parameters the Jacobian is

$$\boldsymbol{J} = \frac{\partial \chi}{\partial \boldsymbol{p}} = \begin{bmatrix} \frac{\partial \chi}{\partial p_1} & \frac{\partial \chi}{\partial p_2} & \frac{\partial \chi}{\partial p_3} & \frac{\partial \chi}{\partial p_4} & \frac{\partial \chi}{\partial p_5} & \frac{\partial \chi}{\partial p_6} \end{bmatrix} \tag{3.31}$$

Similarly the Hessian is of the form

$$\boldsymbol{H} = \frac{\partial^2 \chi}{\partial \boldsymbol{p} \partial \boldsymbol{p}} = \begin{bmatrix} \frac{\partial^2 \chi}{\partial p_1^2} & \frac{\partial^2 \chi}{\partial p_1 \partial p_2} & \frac{\partial^2 \chi}{\partial p_1 \partial p_3} & \frac{\partial^2 \chi}{\partial p_1 \partial p_4} & \frac{\partial^2 \chi}{\partial p_1 \partial p_5} & \frac{\partial^2 \chi}{\partial p_1 \partial p_6} \\ \frac{\partial^2 \chi}{\partial p_2 \partial p_1} & \frac{\partial^2 \chi}{\partial p_2^2} & \frac{\partial^2 \chi}{\partial p_2 \partial p_3} & \frac{\partial^2 \chi}{\partial p_2 \partial p_4} & \frac{\partial^2 \chi}{\partial p_2 \partial p_5} & \frac{\partial^2 \chi}{\partial p_2 \partial p_6} \\ \frac{\partial^2 \chi}{\partial p_3 \partial p_1} & \frac{\partial^2 \chi}{\partial p_3 \partial p_2} & \frac{\partial^2 \chi}{\partial p_3^2} & \frac{\partial^2 \chi}{\partial p_3 \partial p_4} & \frac{\partial^2 \chi}{\partial p_3 \partial p_5} & \frac{\partial^2 \chi}{\partial p_3 \partial p_6} \\ \frac{\partial^2 \chi}{\partial p_4 \partial p_1} & \frac{\partial^2 \chi}{\partial p_4 \partial p_2} & \frac{\partial^2 \chi}{\partial p_4 \partial p_3} & \frac{\partial^2 \chi}{\partial p_4^2} & \frac{\partial^2 \chi}{\partial p_4 \partial p_5} & \frac{\partial^2 \chi}{\partial p_4 \partial p_6} \\ \frac{\partial^2 \chi}{\partial p_5 \partial p_1} & \frac{\partial^2 \chi}{\partial p_5 \partial p_2} & \frac{\partial^2 \chi}{\partial p_5 \partial p_3} & \frac{\partial^2 \chi}{\partial p_5 \partial p_4} & \frac{\partial^2 \chi}{\partial p_5^2} & \frac{\partial^2 \chi}{\partial p_5 \partial p_6} \\ \frac{\partial^2 \chi}{\partial p_6 \partial p_1} & \frac{\partial^2 \chi}{\partial p_6 \partial p_2} & \frac{\partial^2 \chi}{\partial p_6 \partial p_3} & \frac{\partial^2 \chi}{\partial p_6 \partial p_4} & \frac{\partial^2 \chi}{\partial p_6 \partial p_5} & \frac{\partial^2 \chi}{\partial p_6^2} \end{bmatrix} \tag{3.32}$$

The Hessian and Jacobian matrices are determined analytically in two steps. First the chosen correlation coefficient equation is differentiated according to equations 3.31 and 3.32 up until the point that it is the light intensity values of the deformed subset that are being differentiated. This is done by hand and is then coded as a function. Thereafter the equation used to determine the light intensity values of the deformed subset is created symbolically in matlab. Matlab's Jacobian and Hessian function are then used to determine the appropriate derivatives of the light intensity values in terms of the $\boldsymbol{p}$ parameters which are then written as functions using matlabs built in functionality. This is done so that if the warp function is changed the algorithm can recompute the light intensity derivatives automatically while the derivative of the correlation coefficient remains the same. This allows the algorithm to autonomously determine equations (or matlab functions) for the Jacobian and Hessian matrices.

However since Newton-Raphson is being used to solve a six dimensional optimization problem a method of ensuring consistent improvement in the correlation coefficient after each iteration is needed. This is achieved by using the golden section method to search in the $\frac{f'(x_i)}{f''(x_i)}$ direction for the lowest correlation coefficient value. This improves the robustness of the method. First a bracketing method is used to find a section which is convex and then the golden section method is used to determine the optimal point within this section.

3.3 Digital cameras

Cameras at the basic level rely upon using an aperture and a lens to focus light rays that originate from objects onto a plane within the camera called the sensor plane. At the sensor

plane exist a Charge-Coupled Device (CCD) that consists of a matrix of light sensors. Each light sensor converts the light incident upon its surface into an electrical charge through the photoelectric effect. The charge is proportional to the light intensity. Then each light sensor's voltage is read by an analogue-to-digital converter which measures the voltage and assigns a digital value to it. These digital values are then stored at the corresponding position in a matrix which forms the digital image.

3.4 Camera optics

For an image of an object to be in focus the light incident upon each light sensor should originate from one point on the object's surface. However light is reflected by objects in many directions and so a means of focusing the light is required. This is accomplished through using lenses and an aperture. Throughout this project thin lenses are assumed. A thin lens is one in which its thickness is negligible in comparison with its focal length or radius of curvature [9]. Additionally the paraxial approximation is assumed which states that light rays passing through the lens do so with a small angle to the lens's optical axis and pass through the lens close to the optical axis. This leads to the small angle approximation.

$$\sin(\theta) \approx \tan(\theta) \approx \theta \tag{3.33}$$

Lenses are usually disk shaped pieces of glass with two convex surfaces. The convex surfaces are designed to bend light towards the optical axis with the degree of bending increasing with the distance from the light to optical axis at the lens mid plane. Thus diffuse light emanating from a point M $[x, y, z]^T$ on an object will pass through the lens and the light rays will converge at a point called the ideal image point M' $[x', y', z']^T$. Thereafter the light rays diverge again to points M" $[x'', y'', z'']^T$ as shown in figure 3.1. If the sensor plane is coincident with the ideal image points of the light rays the image that forms on the sensor is inverted due to the way the lens bends the light. As a result an inverted coordinate system is used for the sensor plane.

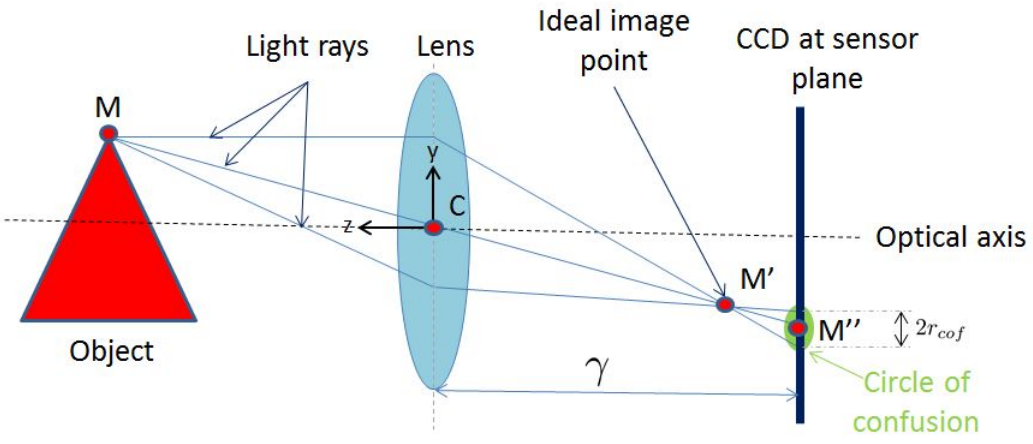


Figure 3.1: Illustration of how light rays are manipulated by the lens

The thin lens equation can be stated as

$$\frac{1}{|CM|} + \frac{1}{CM'} = \frac{1}{\bar{f}} \tag{3.34}$$

where $\bar{f}$ is the focal length; an inherent property of the lens. Additionally through similarity of triangles we have

$$\frac{y}{CM} = \frac{-y'}{CM'} \tag{3.35}$$

$$\frac{x}{CM} = \frac{-x'}{CM'} \tag{3.36}$$

$$\frac{z}{CM} = \frac{-z'}{CM'}. \tag{3.37}$$

Combining these a series of equations for the ideal image point can be obtained.

$$\begin{aligned} x' &= \frac{-\bar{f}x}{z-\bar{f}} \\ y' &= \frac{-\bar{f}y}{z-\bar{f}} \\ z' &= \frac{-\bar{f}z}{z-\bar{f}} \end{aligned} \tag{3.38}$$

3.4.1 Circle of confusion

It is impossible to align the sensor plane perfectly with the ideal image points and so the divergence of the light rays after the ideal image point causes the light rays to illuminate a circular area on the sensor plane. This blurred region is known as the circle of confusion. If this circular area is larger than the area spanned by an element of the sensor then this will result in blurring in the image as the light spills over multiple sensors. This blur could be eliminated by having the sensor plane the same distance from the lens as the the ideal image point but different points on the object will have different ideal image points.

The light rays that make up the outer perimeter of the circle of confusion are the rays that pass through the lens on the outer edges. Using these rays it is possible to determine the radius of the circle of confusion. Taking r to be the radius of the lens these outer light rays pass through the midplane of the lens at $[r\cos\beta, r\sin\beta, 0]^T$ where $0 < \beta < 2\pi$. Using trigonometry the y components of these light rays can be related.

$$\frac{y' - r\sin\beta}{z'} = \frac{y' - y''}{\gamma + z'} \tag{3.39}$$

Th distance between the lens and the sensor plane is given by γ. The same can be done for the x component. Using this and equation 3.38 an expression for the perimeter of the circle of confusion can be found.

$$\begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix} = \begin{bmatrix} r\cos\beta + r\cos\beta\left(1 + \frac{\gamma(\bar{f}-z)}{\bar{f}z}\right) \\ r\sin\beta + r\sin\beta\left(1 + \frac{\gamma(\bar{f}-z)}{\bar{f}z}\right) \\ 0 \end{bmatrix} + \begin{bmatrix} -\frac{\gamma x}{z} \\ -\frac{\gamma y}{z} \\ -\gamma \end{bmatrix} \tag{3.40}$$

The radius of the circle of confusion can be determine by taking the difference in the y components of y'' for $\beta = 90°$ and $\beta = 270°$.

$$2r_{cof} = \left[r\sin 90° \left(2 + \frac{\gamma\left(\bar{f}-z\right)}{\bar{f}z}\right)\right] - \left[r\sin 270° \left(2 + \frac{\gamma\left(\bar{f}-z\right)}{\bar{f}z}\right)\right] \tag{3.41}$$

$$r_{cof} = r\left(1 + \frac{\gamma\left(\bar{f}-z\right)}{\bar{f}z}\right) \tag{3.42}$$

Another way that the blur can be improved is by using an aperture. An aperture is at a basic level an opaque diaphragm with a hole in it which serves to reduce the amount of light that is incident upon the sensor. Thus by blocking off the light that passed through the outer edges of the lens the aperture reduces the size of the circle of confusion by effectively reducing the radius of the lens. The aperture can be place in front or behind the lens.

3.4.2 Depth of field

Depth of field is defined as the distance ahead and behind the object that is in focus. For a particular camera system the ideal image point of an object will fall upon the sensor plane if the object is at an ideal focal length from the lens. If the object is any closer or further from the lens it will cause a circle of confusion on the sensor plane as opposed to a point. If the circle of confusion is small enough it will not spill significant light over multiple sensors which will result in a clear image.

However if the circle of confusion is large enough it will cause blurring in the image. The largest circle of confusion which still results in a sharp image is referred to as the acceptable circle of confusion [9]. Thus the depth of field can be considered as the distance along the z-axis ahead or behind the ideal focus length which results in an acceptable circle of confusion.

The largest degree of freedom for a specific camera system occurs when the object is at the hyperfocal length, H, from the camera. The hyperfocal length is defined as the closest an object can be to the camera such that the depth of field extends to infinity behind the object. In this situation the depth of field starts at a distance of $\frac{H}{2}$. The hyperfocal length can be approximated as

$$H \simeq \frac{\bar{f}^2}{2Nr_{cof}} \tag{3.43}$$

Here N is given by $\frac{\bar{f}}{D_p}$ where D_p is the diameter of the entrance pupil for the camera system. Letting s be the distance from the camera to the object such that the camera is ideally focused at a distance s. The distance from the camera to the near limit of the depth of field, D_N, and the distance from the camera to the far limit of the depth of field, D_F, can be approximated.

$$D_N \simeq \frac{Hs}{H+s} \tag{3.44}$$

$$D_F \simeq \frac{Hs}{H-s} \tag{3.45}$$

These approximations assume that the object distance is large compared to the lens focal length. The depth of field can then be determined.

$$DOF = D_F - D_N = \frac{2Hs^2}{H^2 - s^2} \tag{3.46}$$

Combining equation 3.43 and 3.46

$$DOF = \frac{4Nr_{cof}\bar{f}^2 s^2}{\bar{f}^4 - 4N^2 r_{cof}^2 s^2}. \tag{3.47}$$

Thus it is clear that the depth of field can be controlled by altering the focal length of the lens, by altering N by changing the aperture or changing the distance between the camera and the object.

3.4.3 Field of view

The field of view is the extent of the world that the camera is capable of capturing in an image. It is quantified as the largest angle that a light ray, that is incident upon the sensor, makes with the optical axis. This angle is referred to as the angle-of-view.

Using the pinhole camera model with a distance of L between the sensor and the lens and taking the lens height to be d a relation for the angle-of-view, α, can be derived using trigonometry.

$$\tan\left(\frac{\alpha}{2}\right) = \frac{d}{2L} \tag{3.48}$$

$$\alpha = 2\arctan\left(\frac{d}{2L}\right) \tag{3.49}$$

However for the best picture quality the distance between the sensor and the lens should be equal to the focal length f.

$$\alpha = 2\arctan\left(\frac{d}{2f}\right) \tag{3.50}$$

3.4.4 Transformation to image plane

The relation between an object point and the projection of the object point onto the sensor can be derived from equation 3.40 by eliminating the circle of confusion component. Additionally when the camera system is set up properly z' will be equal to γ so that the sensor plane is approximately coincident with the ideal image points.

$$\begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix} = \begin{bmatrix} -\frac{\gamma}{z} & 0 & 0 \\ 0 & -\frac{\gamma}{z} & 0 \\ 0 & 0 & -\gamma \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{3.51}$$

This equation presents two issues. Firstly the dependence of the sensor positions on z. Secondly all the terms in the equation have metric units whereas the image coordinate system has dimensions measured in pixels. These issues are fixed by using a homogeneous from of equation 3.51.

$$\alpha \begin{bmatrix} -x'' \\ -y'' \\ 1 \end{bmatrix} = \begin{bmatrix} \gamma & 0 & 0 & 0 \\ 0 & \gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{3.52}$$

Here α is a scale factor which allows for the conversion from metric units for the world coordinate system to pixels used in the sensor coordinate system. This equation is referred to as perspective projection [9].

3.4.5 Front image plane model

A new imaging model that is often preferred for computer vision applications can be obtained by translating the sensor plane a distance of 2γ along the optical axis. In this case the sensor plane is in front of the lens. Treating the sensor coordinates M'' of an object as the intersection of the light ray with the sensor plane; equation 3.52 remains valid for this configuration.

This imaging model is advantageous in that the sensor plane coordinates are no longer inverted. This means that the scene being imaged is also not inverted.

3.5 Coordinate systems

Images are only capable of storing two dimensional information whereas we live in a three dimensional world. Thus cameras convert three dimensional information from the world coordinate system into two dimensional information in the sensor coordinate system when a picture is taken as shown in figure 3.2. The mathematical relationship between these two coordinate systems is discussed here.

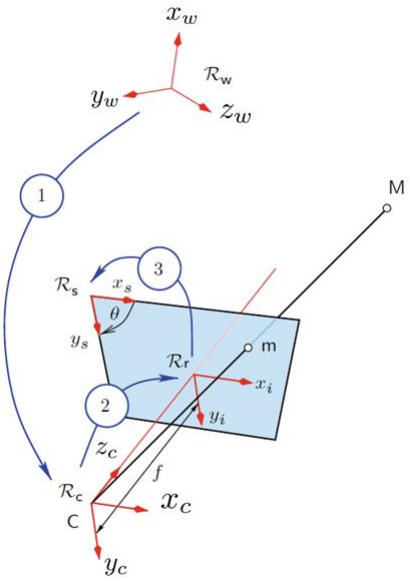


Figure 3.2: The conversion between coordinate systems that occurs when an image is taken

3.5.1 Homogeneous coordinates

It is common knowledge that any object's shape can be fully defined using distances and angles in 3D Euclidean space. However when an image is taken of this object these distances and angles become distorted. For example railway tracks consist of two beams that remain parallel to one another at a set distance apart in Euclidean space but in an image of the railway track (projective space) these beams appear to get closer and closer to one another as seen in figure 3.3. Therefore the parallelism between the beams in Euclidean space is distorted in projective space. This occurs as a result of reducing 3D information to a 2D image.

Homogeneous coordinates make it possible to take a 3D coordinate in Euclidean space into a 4D coordinate in projective space. This is necessary since a transformation matrix for 3D coordinates that applies both rotation and translation has 4 columns and so can only be applied to a 4D coordinate. Conversion to homogeneous coordinates involves adding an additional coordinate w to the coordinate vector and setting $w = 1$. To convert back from homogeneous coordinates requires dividing the x, y and z coordinates by w and then eliminating w.

Figure 3.3: How projective space distorts parallelism present in Euclidean space

3.5.2 World to camera coordinate system

Converting from the world coordinate system to the camera coordinate system involves rigid transformations of translation T and rotation R. The world coordinate system is simply the coordinate system that would be used to classify the position and orientation of objects in the real world. The world coordinate systems orientation and origin is somewhat arbitrary and it is usually classified according to the object to be imaged.

The camera coordinate system is as the name suggests fixed according to the cameras position and orientation. The camera coordinate system's z axis is the optical axis of the camera. The conversion between the two coordinate systems can be represented as

$$\boldsymbol{X}_c = \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{T} \\ \boldsymbol{0} & 1 \end{bmatrix} \boldsymbol{X}_w. \tag{3.53}$$

Scaling of dimensions between the two coordinate systems also needs to be taken into account. The parameters in $\boldsymbol{R}$ and $\boldsymbol{T}$ are referred to as extrinsic parameters since they are not dependent on the camera system's hardware but rather the camera's position and orientation in the world coordinate system.

3.5.3 Camera and imaging plane coordinates

The 3D object defined in the camera coordinate system is converted to its projection on imaging plane. This conversion is from a 3D coordinate system to a 2D coordinate system as given below using homogeneous coordinates.

$$\boldsymbol{X}_i = \alpha \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \begin{bmatrix} \gamma & 0 & 0 & 0 \\ 0 & \gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} \tag{3.54}$$

The variable α is an arbitrary scale factor.

3.5.4 Image plane and sensor coordinates

At this point the locations of where the light rays originating from the object will intersect the imaging plane are known. Now it is necessary to mathematically represent how the sensor would interpret these light rays incident upon the sensor into the form of pixels. In order to do this a relation between the position of a point within a coordinate system and the pixel within an image is necessary. Additionally the sensor array is not guaranteed to the orthogonal and so a skewed coordinate system must be taken into account.

The transformation from the image plane coordinate system to a temporary skewed coordinate system with an angle of ϕ between the two axes can be represented by

$$\begin{bmatrix} x_{temp} \\ y_{temp} \end{bmatrix} = \begin{bmatrix} 1 & -\cot\phi \\ 0 & \frac{1}{\sin\phi} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix}. \tag{3.55}$$

It is assumed that the two principle directions in the sensor coordinate system have different scale factors, S_x and S_y,which have units of pixels per unit length. Applying these to the coordinates calculated in equation 3.55 and accounting for the translations $\hat{c}_x$ and $\hat{c}_y$ to convert to the origin of the sensor coordinate system results in the sensor coordinates below.

$$\begin{bmatrix} x_s \\ y_s \end{bmatrix} = \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \begin{bmatrix} x_{temp} \\ y_{temp} \end{bmatrix} - \begin{bmatrix} S_x\hat{c}_x - S_x\hat{c}_y\cot\phi \\ \frac{S_y\hat{c}_y}{\sin\phi} \end{bmatrix} = \begin{bmatrix} S_x & -S_x\cot\phi \\ 0 & \frac{S_y}{\sin\phi} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} - \begin{bmatrix} S_x\hat{c}_x - S_x\hat{c}_y\cot\phi \\ \frac{S_y\hat{c}_y}{\sin\phi} \end{bmatrix} \tag{3.56}$$

This can be rewritten in homogeneous form so that it is consistent with equation 3.54.

$$\boldsymbol{X}_s = \begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix} = \begin{bmatrix} S_x & -S_x\cot\phi & -S_x\left(\hat{c}_x - \hat{c}_y\cot\phi\right) \\ 0 & \frac{S_y}{\sin\phi} & -\frac{S_y\hat{c}_y}{\sin\phi} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \boldsymbol{A}\boldsymbol{X}_i \tag{3.57}$$

3.5.5 World to sensor coordinates

The conversions between coordinate systems described in the previous sections can be combined into one conversion from the world coordinate system to the sensor coordinate system.

$$\boldsymbol{X}_s = \alpha \begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix} = \begin{bmatrix} S_x & -S_x\cot\phi & -S_x\left(\hat{c}_x - \hat{c}_y\cot\phi\right) \\ 0 & \frac{S_y}{\sin\phi} & -\frac{S_y\hat{c}_y}{\sin\phi} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \gamma & 0 & 0 & 0 \\ 0 & \gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{3.58}$$

This can be simplified by combining the first two matrices.

$$\boldsymbol{X}_s = \alpha \begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix} = \begin{bmatrix} \gamma S_x & -\gamma S_x\cot\phi & -S_x\left(\hat{c}_x - \hat{c}_y\cot\phi\right) & 0 \\ 0 & \frac{\gamma S_y}{\sin\phi} & -\frac{S_y\hat{c}_y}{\sin\phi} & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{3.59}$$

Replacing the elements of the first matrix in equation 3.59 with single variables for the purposes of simplicity the equation can be rewritten as

$$\alpha \begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & f_s & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{3.60}$$

$$= \boldsymbol{K}\boldsymbol{V}\boldsymbol{X}_w. \tag{3.61}$$

Here the parameters relating the world coordinate system to the sensor coordinate system are separated into two matrices. The first matrix $\boldsymbol{K}$ contains the intrinsic parameters which are fixed for a specific camera. The second matrix $\boldsymbol{V}$ contains the extrinsic parameters. The extrinsic parameters change when the camera's position and orientation relative to the world coordinate system changes.

3.6 Distortion

Distortion refers to a collection of phenomena that cause the actual image to differ from that of the idealised image expected from the pinhole camera model. This happens because lenses cannot be manufactured and assembled perfectly and so some misalignment and defects exist in the imaging system. Therefore in order to calculate accurate displacement information the distortions must be accounted for and corrected prior to the correlation process.

Distortion can be separated into many types as done below. The overall distortion of the image can then be mathematically represented as the linear combination of the mathematical expressions for the individual types. A brief explanation of each distortion type and the equation that accounts for the distortion it creates is given below.

3.6.1 Spherical distortion

Spherical distortion is when light rays originating from a point on an object intersect at different points on the sensor plane. This is as a result of the lens not having the perfect curvature required for all the light rays to intersect at the same point on the sensor plane. It is assumed to be axis symmetric relative to the axis passing through the centre of the sensor and to be a function of the radial distance.

$$\boldsymbol{D} = \kappa_1 \rho^4 \boldsymbol{e}_r \tag{3.62}$$

Here $\boldsymbol{e}_r$ is the radial unit vector and ρ is the distance from the origin of the sensor coordinate system to the point under consideration.

3.6.2 Coma distortion

Coma distortion affects light rays that travel towards the lens at an angle to the optical axis. Light rays that go through the centre portion of lens refocus to a point on the sensor plane, whereas light rays that pass through the outer portion of the lens don't refocus fully and intersect the sensor plane further (positive coma) or closer (negative coma) to the optical axis. This results in light from a point on an object creating a comet-like shape on the sensor plane. It is corrected with the following equation.

$$\boldsymbol{D} = \kappa_2 \rho^3 \cos\left(\bar{\theta} - \bar{\theta_c}\right) \boldsymbol{e}_r \tag{3.63}$$

Here $\bar{\theta_c}$ is the orientation of the projected lens tilt angle in the sensor plane.

3.6.3 Astigmatism

Astigmatism is caused by a lens having curvature that varies when measured along perpendicular planes. For instance the vertical plane of the lens has a different curvature to the horizontal plane of the lens. The different curvatures have different focal lengths and so light passing through the vertical portion of the lens will focus at a different distance from the lens than light that passes through the horizontal portion of the lens. This results in line like blurs on the sensor plane for the curvature that is not in focus. The degree to which the astigmatism affects the image increases further away from the centre of the image.

$$\boldsymbol{D} = \kappa_3 \rho^2 \cos\left(\bar{\theta} - \bar{\theta}_A\right) \boldsymbol{e}_r \tag{3.64}$$

Where $\bar{\theta}_A$ is the orientation of the projected astigmatic plane within the sensor plane.

3.6.4 Curvature of field

Curvature of field is a distortion that results due to the curved nature of the optical elements such as the lens. As light rays pass through the lens with a small angle to the optical axis they refocus at the sensor plane. However light rays that pass through the lens at a larger angle to the optical axis refocus at a point that is closer to the lens. This results in the light rays refocusing on a curved plane much like a shallow dome. Since the sensor is planar (flat) this causes the centre of the image to be in focus while the edges of the image are not in focus.

Curvature of field is assumed to be symmetric with respect to the optical axis and to be a quadratic function of the radial position [9].

$$\boldsymbol{D} = \kappa_4 \rho^2 \boldsymbol{e}_r \tag{3.65}$$

Here κ_4 is the amplitude of the curvature of distortion in the sensor plane measured in pixels^{-1}.

3.6.5 Linear

This non-symmetric distortion component is assumed to be a linear function of radial position and is dependent on the angular position [9].

$$\boldsymbol{D} = \kappa_5 \rho \cos\left(\bar{\theta} - \bar{\theta_L}\right) \boldsymbol{e}_r \tag{3.66}$$

Here $\bar{\theta_L}$ is the angular orientation of the linear distortion axis.

3.6.6 Radial

Radial distortion is caused by the lens having different magnification levels based on the angle of the light rays to the optical axis. As a result the image can experience a decrease in magnification with increasing distance from the optical axis (barrel distortion) or the image can experience increasing magnification with increasing distance from the optical axis (pincushion distortion). This distortion is symmetric with respect to the optical axis.

$$\boldsymbol{D} = \kappa_6 \rho^3 \boldsymbol{e}_r + \kappa_7 \rho^5 \boldsymbol{e}_r + \kappa_8 \rho^7 \boldsymbol{e}_r \tag{3.67}$$

3.6.7 De-centering

De-centering distortion is caused by the lens not being in perfect alignment with the rest of the camera system. Usually this type of distortion is less severe than radial or spherical distortions.

$$\boldsymbol{D} = \kappa_9 \rho^2 \left[3 \sin\left(\bar{\theta} - \bar{\theta_d}\right) \boldsymbol{e}_r + \cos\left(\bar{\theta} - \bar{\theta_d}\right) \boldsymbol{e}_t\right] \tag{3.68}$$

Here $\boldsymbol{e}_t$ is the tangential unit vector and $\bar{\theta_d}$ is the orientation of the axis for maximum tangential distortion.

3.7 Calibration

Calibration is a necessary process that must be completed in order to extract metric information from images. The calibration process solves for parameters that define the optical characteristics of the camera, parameters that define the orientation and position of the camera coordinate system to the world coordinate system and parameters that define the distortions that must be corrected in the images.

The parameters that define the optical characteristics and the distortions are referred to as intrinsic parameters since they are fixed for a specific camera. The parameters that describe the orientation and position of the camera within the world coordinate system are extrinsic parameters because they change if the camera is moved.

Multiple methods of calibration exist however calibration using a calibration plate is used in this project since it is one of the most popular methods and making a calibration plate is relatively simple and inexpensive. Calibration for distortion parameters will not be focused on in this section.

3.7.1 Inverse problem

Calibration is essentially a method of finding the parameters that allow 3D coordinates in the world coordinate system to be accurately related to 2D coordinates in the image coordinate system. Thus the inputs, 3D world coordinates, and outputs, 2D image coordinates, needed to be used to solve for the parameters which describe the relationship between the two which constitutes an inverse problem.

Inverse problems are typlically hard to solve and calibration is no exception. In order to solve for the calibration parameters more reliably the camera model that relates world points to sensor points is broken down to the simple pinhole camera model initially in order to solve for as few parameters as possible initially. These parameters are solved for using a closed form solution which gives good estimates to the parameters. Thereafter once these parameters have been estimated the camera model is made more complex by introducing radial distortion in order to account for imperfections in the lens system of the camera. These radial parameters are first estimated and once each parameter has a corresponding estimate all the parameters are optimized in an iterative manner.

This inverse problem is sensitive to errors in the 3D world coordinates (inputs) and 2D image coordinates (outputs) used. Thus these need to be known to a high degree of accuracy in order to solve for the intrinsic and extrinsic camera parameters reliably. This is achieved by using a calibration plate.

3.7.2 Calibration plate

A calibration plate is an object with a flat surface containing a high-contrast, regular pattern. The pattern is such that it contains definitive, point-like features which can be located to a high degree of accuracy within images taken of it. For example a checker board pattern allows for accurate calculation of the points at the corners of the squares. Thus the coordinates of these point-like features on the calibration plate will be known (inputs) and the coordinates of these point-like features in the image can be determined to a high degree of accuracy (outputs).

Since images inherently contain some level of noise it is best to have an overdetermined system of equations. This is accomplished by taking multiple images of the calibration plate and changing the relative position and orientation between the calibration plate and the camera for each image. This effectively reduces the effects of noise; making the system more robust.

3.7.3 Homography

Homography is a transformation that can be applied to points on a plane to bring it into alignment with another plane. It is used to bring the points on the calibration plate in the world coordinate system into alignment with their location in the image in the sensor coordinate system. The transformation from world coordinates to sensor coordinates in equation 3.61 is a type of homography.

Treating the calibration plate such that it lies in the x-y plane of the world coordinate system; the homography, $\boldsymbol{H}$, for calibration is given by the following equation.

$$\begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix} = \alpha \begin{bmatrix} f_x & f_s & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ 0 \\ 1 \end{bmatrix} \tag{3.69}$$

This can be reduced to

$$\begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix} = \alpha \begin{bmatrix} f_x & f_s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ 1 \end{bmatrix} \tag{3.70}$$

$$= \alpha \boldsymbol{K} \begin{bmatrix} \boldsymbol{r}_1 & \boldsymbol{r}_2 & \boldsymbol{t} \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ 1 \end{bmatrix} \tag{3.71}$$

$$= \alpha \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ 1 \end{bmatrix} = \alpha \boldsymbol{H} \begin{bmatrix} x_w \\ y_w \\ 1 \end{bmatrix} \tag{3.72}$$

Here $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ are the first and second columns of the rotation matrix. It is clear that the homography matrix contains both intrinsic and extrinsic parameters and thus it is different for each image taken of the calibration plate. Additionally note that the homography matrix is defined up to a scale factor.

3.7.4 Estimating homography with direct linear transform

The homographies that relate the world coordinates of the calibration plate targets to the targets in the image can be estimated using direct linear transformation [3]. Equation 3.72 can be written out as

$$x_s = \alpha \left(h_1 x_w + h_2 y_w + h_3 \right) \tag{3.73}$$

$$y_s = \alpha \left(h_4 x_w + h_5 y_w + h_6 \right) \tag{3.74}$$

$$1 = \alpha \left(h_7 x_w + h_8 y_w + h_9 \right). \tag{3.75}$$

The scale factor, α, can be eliminated by dividing equations 3.73 and 3.74 by 3.75 to get

$$x_s \left(h_7 x_w + h_8 y_w + h_9 \right) = \left(h_1 x_w + h_2 y_w + h_3 \right) \tag{3.76}$$

$$y_s \left(h_7 x_w + h_8 y_w + h_9 \right) = \left(h_4 x_w + h_5 y_w + h_6 \right) \tag{3.77}$$

These equations then reduce to

$$-h_y x_w x_s - h_8 y_w x_s + h_1 x_w + h_2 y_w + h_3 = h_9 x_s \tag{3.78}$$

$$-h_7 x_w y_s - h_8 y_w y_s + h_4 x_w + h_5 y_w + h_6 = h_9 y_s. \tag{3.79}$$

In order to avoid the trivial solution where every element in the homogrpahy matrix is equal to zero; constraints need to be placed on the elements of the homography matrix. In this case the element h_9 is set equal to 1 however other constraints are also possible such as $h_7^2 + h_8^2 + h_9^2 = 1$. Note that if the true value of h_9 is close to zero then this assumption will introduce a singluarity [13].

Each target, having points x_{w_i} and y_{w_i}, on the calibration plate that is captured within an image of the calibration plate, having points x_{s_i} and y_{s_i}, provides both an equation 3.78

and 3.79. These are then combined into an equation of the form

$$\begin{bmatrix} x_{w_1} & y_{w_1} & 1 & 0 & 0 & 0 & -x_{w_1}x_{s_1} & -y_{w_1}x_{s_1} \\ 0 & 0 & 0 & x_{w_1} & y_{w_1} & 1 & -x_{w_1}y_{s_1} & -y_{w_1}y_{s_1} \\ x_{w_2} & y_{w_2} & 1 & 0 & 0 & 0 & -x_{w_2}x_{s_2} & -y_{w_2}x_{s_2} \\ 0 & 0 & 0 & x_{w_2} & y_{w_2} & 1 & -x_{w_2}y_{s_2} & -y_{w_2}y_{s_2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \\ h_7 \\ h_8 \end{bmatrix} = \begin{bmatrix} x_{s_1} \\ y_{s_1} \\ x_{s_2} \\ y_{s_2} \\ \vdots \end{bmatrix} \tag{3.80}$$

This overdetermined system of equations then can be used to solve for the homography matrices of each calibration image using least squares. This is the first step involved in calibration.

3.7.5 Absolute conic

As mentioned before two objects that are parallel in Euclidean space appear to intersect each other in projective space. The intersection of these two lines in projective space occurs at a point that lies on the plane at infinity. If a point lies upon the plane at infinity its w is equal to zero in homogeneous coordinates.

The absolute conic lies on the plane at infinity and is defined by the set of points, $\tilde{\boldsymbol{x}}_{ac} = [x, y, z, w]^T$, that satisfies the following.

$$w = 0 \tag{3.81}$$

$$\boldsymbol{x}_{ac}^T \boldsymbol{x}_{ac} = x^2 + y^2 + z^2 = 0 \tag{3.82}$$

Here tilde is used to indicate homogeneous coordinates whereas $\boldsymbol{x}_{ac} = [x, y, z]^T$ would be the point in Euclidean space.

Thus it is the conic of purely imaginary points that lies upon the plane at infinity. The importance of the absolute conic is that it is invariant under any Euclidean transformations. In other words the relative position of the absolute conic to a moving camera is unaffected by extrinsic parameters. Consider a point $\boldsymbol{x}_{ac}$ in Euclidean space in the world coordinate system that lies on the absolute conic. It has homogeneous coordinates $\tilde{\boldsymbol{x}}_{ac} = [\boldsymbol{x}_{ac}^T, 0]^T$ in projective space. Applying a different rotation and translation to this point to obtain $\tilde{\boldsymbol{x}}'_{ac}$, which is equivalent to moving the camera in the world coordinate system, a corresponding point is obtained in the camera coordinate system.

$$\tilde{\boldsymbol{x}}'_{ac} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{T} \\ \boldsymbol{0} & 1 \end{bmatrix} \tilde{\boldsymbol{x}}_{ac} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{ac} \\ y_{ac} \\ z_{ac} \\ 0 \end{bmatrix} \tag{3.83}$$

$$= \begin{bmatrix} \boldsymbol{R}\boldsymbol{x}_{ac} \\ 0 \end{bmatrix} \tag{3.84}$$

It is clear that this point still lies on the plane at infinity since its w is equal to zero. More importantly it can be proven that this point $\boldsymbol{x}'_{ac}$ is on the same absolute conic.

$$\boldsymbol{x}_{ac}'^T \boldsymbol{x}'_{ac} = (\boldsymbol{R}\boldsymbol{x}_{ac})^T (\boldsymbol{R}\boldsymbol{x}_{ac}) = \boldsymbol{x}_{ac}^T \boldsymbol{R}^T \boldsymbol{R} \boldsymbol{x}_{ac} = \boldsymbol{x}_{ac}^T \boldsymbol{x}_{ac} = 0 \tag{3.85}$$

Thus the absolute conic is invariant to Euclidean transformations. Now consider again a point, $\boldsymbol{x}_{ac}$, lying on the absolute conic. The corresponding point, $\boldsymbol{m}_s$, in the sensor plane is

given by

$$\tilde{\boldsymbol{m}}_s = \frac{1}{\alpha}\boldsymbol{K}\begin{bmatrix}\boldsymbol{R} & \boldsymbol{T}\\ \boldsymbol{0} & 1\end{bmatrix}\begin{bmatrix}\boldsymbol{x}_{ac}\\ 0\end{bmatrix} = \frac{1}{\alpha}\boldsymbol{K}\begin{bmatrix}\boldsymbol{R}\boldsymbol{x}_{ac}\\ 0\end{bmatrix} \tag{3.86}$$

$$\therefore \boldsymbol{m}_s = \frac{1}{\alpha}\boldsymbol{K}\boldsymbol{R}\boldsymbol{x}_{ac}. \tag{3.87}$$

Checking whether this point satisfies equation 3.82 results in

$$\boldsymbol{m}_s^T\boldsymbol{K}^{-T}\boldsymbol{K}^{-1}\boldsymbol{m}_s = \frac{1}{\alpha^2}\boldsymbol{x}_{ac}^T\boldsymbol{R}^T\boldsymbol{R}\boldsymbol{x}_{ac} = \frac{1}{\alpha^2}\boldsymbol{x}_{ac}^T\boldsymbol{x}_{ac} = 0 \tag{3.88}$$

Thus the image of the absolute conic is itself an imaginary conic. The image of the absolute conic is defined by $\boldsymbol{K}^{-T}\boldsymbol{K}^{-1}$ [8]. Thus since the image of the absolute conic is dependent only on intrinsic camera parameters it can be used to solve for the intrinsic camera parameters.

3.7.6 Constraints on intrinsic parameters

According to Zhang [13] there are two constraints placed upon the intrinsic parameters of the camera. These are important later on in the solving for these intrinsic parameters. The plane of the calibration plate in the camera coordinate system is given by [13]

$$\begin{bmatrix}\boldsymbol{r}_3\\ \boldsymbol{r}_3^T\boldsymbol{t}\end{bmatrix}^T\begin{bmatrix}x\\ y\\ z\\ w\end{bmatrix} = 0. \tag{3.89}$$

Here w is zero for points on the plane at infinity and one for those that are not. This plane intersects the plane at infinity on a line and it happens that $[\boldsymbol{r}_1^T, 0]^T$ and $[\boldsymbol{r}_2^T, 0]^T$ are points on this line [13]. Thus it is known that any point on this line, x_c^∞, is a linear combination of these two points.

$$x_c^\infty = a\begin{bmatrix}\boldsymbol{r}_1\\ 0\end{bmatrix} + b\begin{bmatrix}\boldsymbol{r}_2\\ 0\end{bmatrix} = \begin{bmatrix}a\boldsymbol{r}_1 + b\boldsymbol{r}_2\\ 0\end{bmatrix} \tag{3.90}$$

Now assume that this point, x_c^∞, lies on the absolute conic. Then this point must satisfy equation 3.82. This would require that $a^2 + b^2 = 0$ which results in the solution $b = \pm ai$, where $i = \sqrt{-1}$. As a result it can be seen that the two points along this line intersect the absolute conic at

$$x_c^\infty = a\begin{bmatrix}\boldsymbol{r}_1 \pm i\boldsymbol{r_2}\\ 0\end{bmatrix}. \tag{3.91}$$

Since these points lie on the absolute conic they are invariant under Euclidean transformations. Their projection, up to a scale factor, in the sensor coordinate system is given by

$$x_s^\infty = \boldsymbol{K}(\boldsymbol{r}_1 \pm \boldsymbol{r}_2) = \boldsymbol{h}_1 \pm i\boldsymbol{h}_2. \tag{3.92}$$

Substituting these points into equation 3.82 results in

$$(\boldsymbol{h}_1 \pm i\boldsymbol{h}_2)^T\boldsymbol{K}^{-T}\boldsymbol{K}^{-1}(\boldsymbol{h}_1 \pm i\boldsymbol{h}_2) = 0. \tag{3.93}$$

Requiring both the imaginary and real parts of this equation to equal zero results in two constraints on the intrinsic camera parameters.

$$\boldsymbol{h}_1^T\boldsymbol{K}^{-T}\boldsymbol{K}^{-1}\boldsymbol{h}_2 = 0 \tag{3.94}$$

$$\boldsymbol{h}_1^T\boldsymbol{K}^{-T}\boldsymbol{K}^{-1}\boldsymbol{h}_1 = \boldsymbol{h}_2^T\boldsymbol{K}^{-T}\boldsymbol{K}^{-1}\boldsymbol{h}_2 \tag{3.95}$$

3.7.7 Intrinsic parameters and the absolute conic

An homography, $\boldsymbol{H}$, between the calibration plate and the image of the calibration plate can be estimated. This homography can then be used to solve for the image of the absolute conic. Once the image of the absolute conic is known it can be used to solve for the intrinsic camera parameters. The image of the absolute conic can be represented as

$$\boldsymbol{B} = \boldsymbol{K}^{-T}\boldsymbol{K}^{-1} = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{12} & b_{22} & b_{23} \\ b_{13} & b_{23} & b_{33} \end{bmatrix} \tag{3.96}$$

Since $\boldsymbol{B}$ is symmetric its contents can be represented by a 6D vector $\boldsymbol{b} = [b_{11}b_{12}b_{22}b_{13}b_{23}b_{33}]^T$. Let $\boldsymbol{h}_i = [h_{i1}h_{i2}h_{i3}]^T$ to be the i$^{\text{th}}$ column of $\boldsymbol{H}$. Then

$$\boldsymbol{h}_I^T \boldsymbol{B} \boldsymbol{h}_j = \boldsymbol{v}_{ij}\boldsymbol{b} \tag{3.97}$$

where

$$\boldsymbol{v}_{ij} = [h_{i1}h_{j1}, h_{i1}h_{j2} + h_{i2}h_{j1}, h_{i2}h_{j2}, h_{i3}h_{j1} + h_{i1}h_{j3}, h_{i3}h_{j2} + h_{i2}h_{j3}, h_{i3}h_{j3}]^T. \tag{3.98}$$

Then equation 3.94 can be rewritten as

$$\begin{bmatrix} \boldsymbol{v}_{12}^T \\ (\boldsymbol{v}_{11} - \boldsymbol{v}_{22})^T \end{bmatrix} \boldsymbol{b} = \boldsymbol{0} \tag{3.99}$$

A separate version of this equation exist for each image taken of the calibration plate and these equations can be stacked, in $\boldsymbol{V}$, to give

$$\boldsymbol{V}\boldsymbol{b} = \boldsymbol{0}. \tag{3.100}$$

The solution for $\boldsymbol{b}$, up to a scale factor λ, is known to be the eigenvector of $\boldsymbol{V}^T\boldsymbol{V}$ associated with the smallest eigenvalue [13]. Once b has been determined it can be used to determine the intrinsic parameters of matrix $\boldsymbol{K}$. The relation between $\boldsymbol{B}$ and $\boldsymbol{K}$ is $\boldsymbol{B} = \lambda\boldsymbol{K}^{-T}\boldsymbol{K}^{-1}$. The intrinsic parameters are determined as follows.

$$c_y = (b_{12}b_{13} - b_{11}b_{23})/(b_{11}b_{22} - b_{12}^2) \tag{3.101}$$

$$\lambda = b_{33} - (b_{13}^2 + c_y(b_{12}b_{13} - b_{11}b_{23}))/b_{11} \tag{3.102}$$

$$f_x = \sqrt{\frac{\lambda}{b_{11}}} \tag{3.103}$$

$$f_y = \sqrt{\frac{\lambda b_{11}}{b_{11}b_{22} - b_{12}^2}} \tag{3.104}$$

$$f_s = \frac{-b_{12}f_x^2 f_y}{\lambda} \tag{3.105}$$

$$c_x = \frac{f_s c_y}{f_y} - \frac{b_{13}f_x^2}{\lambda} \tag{3.106}$$

Thereafter the extrinsic parameters can be determined.

$$\boldsymbol{r}_1 = \lambda\boldsymbol{K}^{-1}\boldsymbol{h}_1 \tag{3.107}$$

$$\boldsymbol{r}_2 = \lambda\boldsymbol{K}^{-1}\boldsymbol{h}_2 \tag{3.108}$$

$$\boldsymbol{r}_3 = \boldsymbol{r}_1 \times \boldsymbol{r}_2 \tag{3.109}$$

$$\boldsymbol{T} = \lambda\boldsymbol{K}^{-1}\boldsymbol{h}_3 \tag{3.110}$$

At this point it is necessary to incorporate distortions and use them to find better approximations for the intrinsic and extrinsic parameters.

3.7.8 Closed form solution

Sections 3.7.3 throught to 3.7.7 illustrate that estimates to the intrinsic and extrinsic parameters can be obtained by first determining the homographies for each calibration image, then these can be used to determine the image of the abolute conic which in turn can be used to determine the intrinsic and extrinsic camera parameters. This is closed form solution is the first part of the calibration process

put inverse problem in calibration plate section

explain homography and absolute conic

then explain closed form solution and optimization - give closed form and explain optimization

3.7.9 Distortion in Calibration

The closed form solution given above does not take distortion into account since it is based on the pinhole camera model. It is only once the closed form solution to the extrinsic and intrinsic parameters is optimised that distortion can be accounted for during the calibration process. As already presented, there are many types of distortion that occur when an image is taken. However it is seldom possible to take all of these distortion types into account since the more distortion paramters introduced into the camera model; the more likely the optimization process becomes unstable.

It has been found that good calibration results can be achieved by taking only radial distortion into accound during calibration [10, 12]. This is beneficial since it is possible to solve for initial guesses to the radial distortion parameters which helps the optimisation process to avoid local minima as a result of the distortion parameters used.

The distortion applied to the points on the image plane is of the form

$$\begin{bmatrix} x_{i_{distorted}} \\ y_{i_{distorted}} \end{bmatrix} = 1 + k_1 r^2 + k_2 r^4 \begin{bmatrix} x_i \\ y_i \end{bmatrix} \tag{3.111}$$

$$\text{where} \qquad r = \sqrt{x_i^2 + y_i^2}. \tag{3.112}$$

At this point the intrinsic and extrinsic parameters have been determined using the pinhole camera model and the error between the predicted calibration plate targets and the actual location of these in the images is attributed to radial distortion [3]. This error is represented as

$$d_1 = x_{s_{actual}} - x_{s_{calculated}} \tag{3.113}$$

An undistorted point, x_i is distorted to point $\tilde{x_i}$ according to

$$\tilde{x_i} = u_c + (x_i - u_c) * (1 + k1r^2 + k2r^4) \tag{3.114}$$

$$= u_c + x_i - u_c + (x_i - u_c) * (1 + k1r^2 + k2r^4) \tag{3.115}$$

$$= x_i + (x_i - u_c) * (1 + k1r^2 + k2r^4) \tag{3.116}$$

$$= x_i + d_{2_i} \tag{3.117}$$

The distortion parameters can be estimated by minimizing the difference between d_1 and d_2. Thus the least squares solution to the overdetermined system is what is needed.

3.8 Displacement fields

In this section some displacement fields are presented in order to illustrate what information DIC and DVC are capable of extracting from image sets.

3.8.1 DIC displacement fields

The figures below show the displacement fields in the region of the crack of a compact tension sample as it is loaded in the y-direction. As such the x-displacement magnitudes are at most 10 % of the displacements in the y-direction. It becomes clear in analysing these figures that the displacement fields determined by DIC are quite intuitive and can be easily interpreted to understand the deformation that a specimen has undergone.

Additionally these images illustrate the full-field nature of the displacement data that DIC is capable of determining. This coupled with its high accuracy and high spatial resolution makes DIC a very attractive displacement measurement tool.

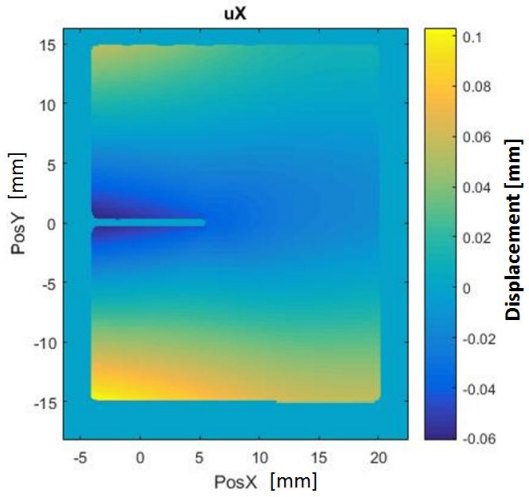


Figure 3.4: X-direction displacement field of a compact tension specimen (DIC)

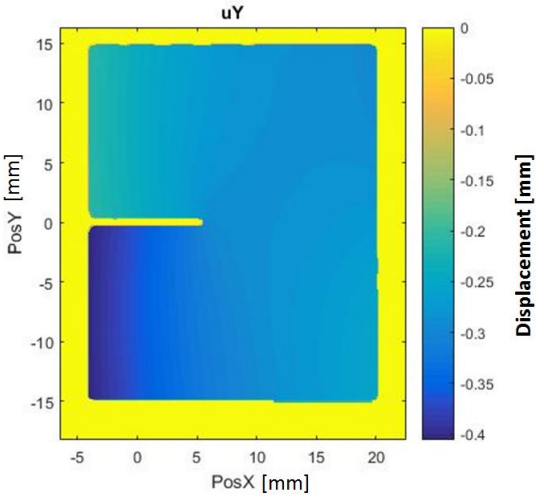


Figure 3.5: Y-direction displacement field of a compact tension specimen (DIC)

3.8.2 DVC displacement fields

DIC displacement fields consist of x and y displacements of points defined in two dimensions. This is because DIC only captures the deformation on the surface of the specimen. However DVC displacement fields consist of x, y and z displacements of points defined in three dimensions. This is because DVC captures deformation information through the thickness of the specimen.

As such in order to visualise DVC displacement fields a plane through the thickness (or on the surface) of the specimen must be chosen. Then the displacements can be visualised on this plane. This plane can then be moved through the thickness of the specimen in order to determine how the displacement fields differ in that direction.

This is illustrated in figures 3.6 and 3.7 by the change in location of the crack front. These figures show the z direction displacement fields of a specimen in the x-y plane at two different heights in the z direction. This is a rectangular specimen that contains a crack front which is not perpendicular to any of the specimens surfaces. These two figures show how the x-y position of the crack front changes with a change in the z position. Thus DVC allows for more of the materials response to be captured.

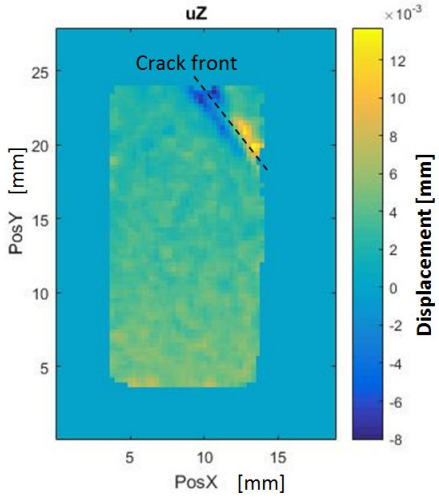


Figure 3.6: Z-direction displacement field on the x-y plane at a height of 13.95 mm in the z direction (DVC)

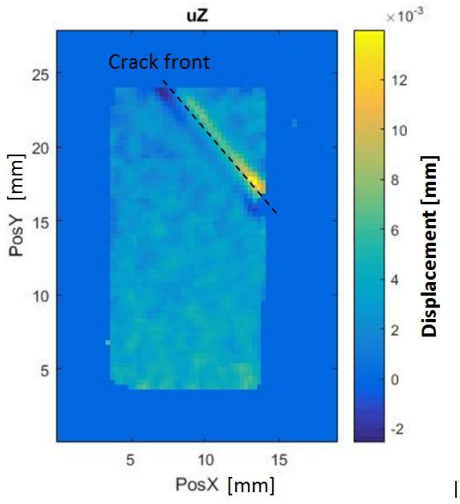


Figure 3.7: Z-direction displacement field on the x-y plane at a height of 15.45 mm in the z direction (DVC)

Chapter 4

Objectives

The main aim for this project is to create a unifying program which is capable performing 2D Digital Image Correlation (DIC) or Digital Volume Correlation (DVC) on a given set of images while allowing the user full control over what methods are used to perform each task during the correlation process. The program is to consist of a basic Graphical User Interface (GUI) and a backend which will handle calibration and correlation. The focus of this project is on the backend.

4.1 Literature review

Conduct an in-depth literature review on 2D subset based DIC and DVC. The purpose of this is to obtain a good understanding of the overall calibration and correlation processes involved in gradient-descent, subset based DIC and how these are related to DVC. In doing so special attention will be paid as to how the various DIC algorithms perform the same tasks in the correlation process using different methods. It is these different methods that are to be included in the program as options for how the correlation process can be controlled by the user.

4.2 Framework and preliminary backend code

The next objective is to use the knowledge gained from the literature review to design a framework for how the backend will perform 2D gradient-descent, subset based DIC. This framework will focus on how the different tasks of the correlation process can be separated so that modular code for each method of performing these tasks, identified in the literature review, can be interchanged to perform correlation. This will form the basis of how the user will be capable of controlling how the correlation process is performed.

Once a good framework has been designed the various methods of performing the tasks of the correlation process will be broken down into programmable functions so that a code can be created for each in Matlab. Then these codes will be connected according to the framework to form a first version of the backend of the program for DIC applications.

Thereafter a framework will be designed for DVC which will involve how the DIC framework can be changed to incorporate an extra dimension in the analysis. This framework will then be used to create code for the backend which enables DVC analysis to be performed.

4.3 Validating the preliminary backend code

Once the preliminary backend code has been completed it will be tested to identify bugs and problems. Image sets from previous DIC and DVC experiments will be analysed using this preliminary code to determine displacements. These calculated displacements will be compared to the displacements calculated by the commercial DIC software (LaVison) so that areas of poor correlation can be identified. Simpler experiments such as basic tension tests will be used first so that basic issues and bugs in the code can be identified and fixed. Thereafter more complex experiments such as specimens containing cracks will be used to see what problems arise in the code.

4.4 Improve preliminary backend code

Problems that arise in the preliminary backend code must be fixed. First problems that affect the execution of the code are to be fixed. Thereafter issues specific to material science applications are to be investigated. These are likely to include difficulty in dealing with discontinuous displacement fields, high strain gradients, applied masks and noise robustness. To improve the codes ability with regards to these types of problems additional functions will need to be designed such that they deal with these issues. Once this objective is complete a good backend for 2D DIC and DVC should result.

4.5 Creating the GUI

With the backend functioning properly a framework for communication between the GUI and backend is to be designed. This should be designed in such a way that it will still allow users to communicate directly with the backend without a need for the GUI. Once the framework is finished the backend will be edited to conform to this framework and the GUI layout will be designed according to what codes are available to perform the different correlation tasks and what input parameters are needed for these codes. Thereafter the GUI will be coded in Matlab according to this design.

4.6 Validate the program

Validate the program by quantifying its performance against that of commercial DIC software. In order for the proposed program to be of use it needs to perform at a similar level to that of commercial software. Two types of tests will be used for this. First generated speckle patterns will be deformed according to known displacement fields and the images of these will be correlated using both the proposed and commercial software. The similarity between the calculated displacement fields and the known displacement fields that were applied will give an indication of the performance.

The second test will involve deforming specimens as images are captured using the commercial DIC system. These images will then be analysed using both the commercial and proposed DIC program. Finally the results of these two will be compared in order to determine whether the proposed program performs to an acceptable level.

Chapter 5

Scope

This project is limited to the aspects listed below. Note that the proposed program is not intended to capture images or perform 3D DIC.

- Developing the backend of the program such that it is capable of performing 2D gradient descent, subset based DIC and DVC on a given set of images to output the displacements present in the images. The backend is to allow the user to select what methods are used during the correlation process.
- Developing a GUI for the program that allows the user import images, set up the type of analysis, set up the configuration for the correlation process, tell the backend to perform the correlation process and display the results.
- Testing the proposed program in order to validate it. The program will be tested using synthetic and real world data to determine how well it performs.
- Design specimens for the experiments that are to be performed to validate the program.

Chapter 6

Research planning

The research plan consists of activities that are closely based on the objectives of the project. The activities are first described and then the time frame, requirements, outputs, cost and milestones of the activity are given.

6.1 Literature review

An in-depth literature review on 2D gradient descent, subset based DIC and DVC are to be conducted in order to develop a good understanding of how these methods work and what steps are involved in the calibration and correlation processes. DIC will be investigated first since it is the simplest type of DIC and so will be easiest in terms of learning the basic concepts. Once a good understanding of DIC is acquired research will move to DVC which is closely related to 2D DIC.

As the different proposed 2D DIC algorithms are investigated special attention will be paid to the ways in which these algorithms use different methods to perform the necessary correlation tasks. Thus as the literature review is conducted a map of how the correlation process works and what tasks need to be completed during its execution will be developed. Then when the different algorithms are investigated their methods of performing these tasks will be documented and a list of the possible ways of performing each task of the correlation process will be created.

Time frame 01/06/2017 - 30/08/2017

Requirements Literature material is necessary for this step. The majority of this material is easily accessible using the internet and other material can be obtained through the Stellenbosch library.

Outputs Good understanding of how 2D DIC and DVC work. A map of how the correlation process works and a list of the different methods that can be used to perform the various correlation steps.

Cost No cost is expected to be incurred during this activity.

Milestones Gain sufficient understanding in order to start developing a framework for DIC and DVC.

6.2 DIC framework

The map of the correlation tasks and the list of the different methods of performing these tasks will be used to create a framework for the DIC part of the backend of the program. First the tasks that need to be performed during correlation, the order in which these tasks need to be performed, the methods that can be used to perform these tasks and what information transfer each of these methods involve will be used to create a detailed structure of how the tasks can be arranged and connected in such a way that these different methods of performing the tasks can be used interchangeably.

Thereafter each method of performing the correlation tasks will be analysed and broken down into programmable functions. The connection and communication between these functions will then be used to create a map for each method. The detailed structure of the correlation tasks and these maps for all the methods will then be combined to create a framework for the overall backend of the program.

This framework will then be used as a basis to develop a similar DVC framework. These frameworks will be similar to one another with the DVC framework basically including an extra dimension to be analysed. These two frameworks will then be combined to give the overall backend framework.

The purpose of this backend framework is to break down the overall correlation problem into the straight forward functions that need to be performed. These straightforward functions are then relatively easy to program and connect together to form the backend of the program.

Time frame 01/09/2017 - 10/10/2017

Requirements The map of how the correlation process works and a list of the different methods that can be used to perform the various correlation steps are needed in order to develop the framework.

Outputs A comprehensive framework for 2D DIC and DVC that allows user control over options for the correlation process.

Cost No cost is expected to be incurred during this activity.

Milestones Complete framework for backend.

6.3 Program the backend

In this activity the functions identified in the backend framework are to be written into code. Then once code exist for all these functions the codes will be connected according to the framework to create the backend of the program. This backend is to be capable of both 2D DIC and DVC analysis.

As the codes are being developed they will be tested by using synthetic tests. In these tests an image of a generated speckle pattern will be deformed according to a chosen displacement function to produce a deformed image. Then these images will be analysed using the backend and the difference between the calculated displacements will serve as a means of identifying faulty code and bugs. These issues will be fixed as they are identified.

Matlab is to be used as the programming language since most research institutions use it to analyse DIC data. Thus it is reasoned that anyone who want to use DIC software which allows this level of control over the correlation algorithm is likely to have access to Matlab.

Time frame 11/10/2017 - 30/11/2017

Requirements The framework for the backend is needed as an input for this activity. Additionally Matlab and a computer are required for programming purposes.

Outputs A working backend for the program coded in Matlab.

Cost Since Matlab and a computer are provided by the university no cost are anticipated.

Milestones Create a working backend code to perform 2D DIC and DVC analysis.

6.4 Validate the backend

Once the backend is finished it will be tested on actual DIC and DVC image sets from material science experiments. The purpose of this is to identify any fundamental issues in the backend and to identify issues specific to material science applications. A list of the issues is to be created so that they can be improved on. Issues will be identified by areas of poor correlation and discrepancies between the results of the proposed program and that of the commercial software (LaVision).

Various types of material science experiments will be used in order to identify as many material science related issues as possible. Anticipated issues include difficulty in dealing with displacement discontinuities, applied masks, high strain gradients and noise.

Various correlation process configurations of the different methods of performing the correlation tasks will be used in order to identify which methods have the greatest impact on the results and to identify issues that arise with using certain methods.

Time frame 01/12/2017 - 30/12/2017

Requirements A working backend which is created in the previous activity. Commercial DIC software which is available at Stellenbosch University. Matlab and a computer which has been provided by Stellenbosch University. Material science experiment image sets, to be analysed, which are already available on the DIC computer of the material science research group.

Outputs A list of issues to be fixed in the backend. These issues include problems specific to material science applications. Additionally an understanding of what types of displacement fields cause the biggest issues for the backend.

Cost All of the required equipment is already available at Stellenbosch University and so this activity is not expected to any costs.

Milestones Identify issues that need to be fixed or improved within the backend code.

6.5 Specimen design

The specimens required for validating the program are to be designed and manufactured. This is done at this stage in the project in order to allow ample time for the specimens to be manufactured so that no delays occur later in the project when time restrictions are strict. Additionally after completing the testing of the algorithms in the previous activity

the displacement fields that are unfavourable for the DIC code will become apparent. The specimens can then be designed in such a way that they will produce similar displacement fields so that the DIC algorithms can be put to the test.

All the specimens are to be designed such that they are loaded in tension. The standard dog-bone tension specimen is to be used as a starting point for the specimens and its geometry will be modified in order to create interesting displacement fields. An example of this would be a dog-bone specimen with a hole in the middle. FEM will be used to ensure that the proposed specimen designs will result in the desired displacement fields.

Time frame 01/01/2018 - 08/01/2018

Requirements Material for the specimens. Knowledge of what displacement fields are unfavourable for the backend (obtained from previous activity). CAD software for designing the specimens which is provided by Stellenbosch University. FEM software for developing the specimen designs is provided for by Stellenbosch University.

Outputs The specimens to be tested later on in the project to validate the program.

Cost The cost of the materials and the cost of the manufacturing the specimens. - R2000

Milestones Creation of specimen designs for manufacturing.

6.6 Improve backend

The issues identified in the backend are to now be solved. Fundamental issues will be solved first since they are specific to the codes already developed for the backend. Thereafter methods of solving issues specific to material science applications will be investigated.

Once methods have been developed to improve the performance of the code in material science applications it will be tested by using the experimental image sets used to first identify issues. The the previous results will be compared to the new results to determine whether there has been an improvement.

Time frame 09/01/2018 - 14/03/2018

Requirements A list of issues that have been identified in the backend.

Outputs A more robust backend code that is less susceptible to errors when being used for material science applications.

Cost This activity involves no cost.

Milestones Produce a backend capable of performing 2D DIC and DVC analysis for material science applications.

6.7 Create the GUI

A GUI is to be created to allow the user to easily control the backend without the need to use command prompt. Before work on the GUI is started a method of communication between the GUI and backend is to be designed. This method of communication should be flexible so that the user can still make use of the backend without the GUI if they so desire. With this method of communication finalised the backend is to be refined to conform to this communication method. This will likely only involve minor changes.

Once a method of communication has been developed the GUI can be designed. The GUI layout will be designed according to the structure of the backend. That is the GUI layout is to be arranged such that the different options available for the correlation process are presented in a logical way. Once a layout has been decided upon the GUI will be programmed in Matlab. Matlab's GUI environment offers sufficient GUI elements for the creation of a GUI of this type. Finally the GUI will be connected to the backend. At this point the proposed program should be fully operational.

Time frame 15/03/2018 - 15/04/2018

Requirements Fully functioning backend. A list of the correlation options to be provided; which will be included in the framework developed for the backend.

Outputs A GUI and a refined version of the backend which together form the proposed program.

Cost No cost is expected to be incurred during this activity.

Milestones Create the final working version of the program (GUI and backend).

6.8 Validate the program

The proposed program is to be validated by comparing its performance to that of commercial software (LaVision) in two types of tests. The first test is a synthetic test similar to that used during the backend's development to check that the code operates correctly. However this time the calculated displacement fields of the various algorithms will be compared to the actual applied displacement field with the difference between the two giving an indication of the accuracy of the algorithm.

Different applied displacement fields will be used in order to identify shortcomings in the developed algorithms. This will be done by using some displacement fields which are not favourable for DIC and DVC such as those containing significant rotation, displacement discontinuities and large straining. These shortcomings are important since commercial software usually includes ways of dealing with these difficulties. Thus comparing the performance between the commercial software and proposed program in this regard is important.

Additionally in order to evaluate how well these algorithms deal with noise a second version of the synthetic test will be conducted. In this second version noise will be added to the deformed image that is to be generated and the deviation of the results from the noise free case will be used to indicate sensitivity to noise.

The second test is more representative of real world use of the program. For this test a speckle pattern is applied to each of the aforementioned specimens before securing them in a tensile testing rig. Then images are taken of the speckle pattern using the available commercial system (LaVision) as the specimen is loaded in tension. These images are then

analysed using the commercial software and the proposed program in order to determine the underlying displacement fields.

The same images will be used for all the algorithms so that they can be compared solely on their correlation performance. The commercial system is used to capture images in order to give the commercial DIC software its best use case so that its results can be treated as the golden standard by which the proposed algorithms are compared.

Once these tests have been conducted the results will be used to decide whether the proposed algorithms perform acceptably well compared to the commercial software. Note that the results of the second test will be relative to the commercial software and not an absolute indication of performance.

Time frame 16/04/2018 - 15/05/2018

Requirements Fully functioning version of the proposed program as created in the previous activity. Matlab, a computer and the commercial DIC software (LaVision) which are all available at Stellenbosch University. The specimens which have been designed and manufactured. A tensile test rig and a commercial DIC camera system which are already available at Stellenbosch University. Spray paint for applying speckle patterns to the specimens.

Outputs Data indicating the performance of the propose program and the commercial DIC software (LaVision) for the same tests.

Cost Spray paint - R200

Milestones Obtain data on the performance of the proposed program.

6.9 Write up thesis

In this activity the information gained throughout this project is to be organised and broken down so that it can be reported on in the thesis.

Time frame 16/05/2018 - 01/08/2018

Requirements Literature review must be conducted prior to this. Results indicating the performance of the proposed program. Latex to write the report.

Outputs A report summarising the important aspects of the project.

Cost Since latex is free no costs will result.

Milestones Complete the report.

Chapter 7

Budget

The table below provides an outline of the expected costs for this project organised according to the activities in which the expenses are expected to occur. Since most of the resources required for the completion of this project are already available from Stellenbosch University the overall cost of the project is rather low.

Activity	Resource	Cost
Literature review	Internet and research papers	0
DIC framework	NA	0
Program backend	Matlab	0
Validate the backend	Commercial software	0
	Matlab	0
Specimen design	Materials and manufacturing	2000
	CAD software	0
	FEM software	0
Improve backend	Matlab	0
Create GUI	Matlab	0
Validate program	Tensile testing rig	0
	Commercial DIC system	0
	Spray paint	200
Write up thesis	Latex	0
Total		2200

Chapter 8

Time-frame

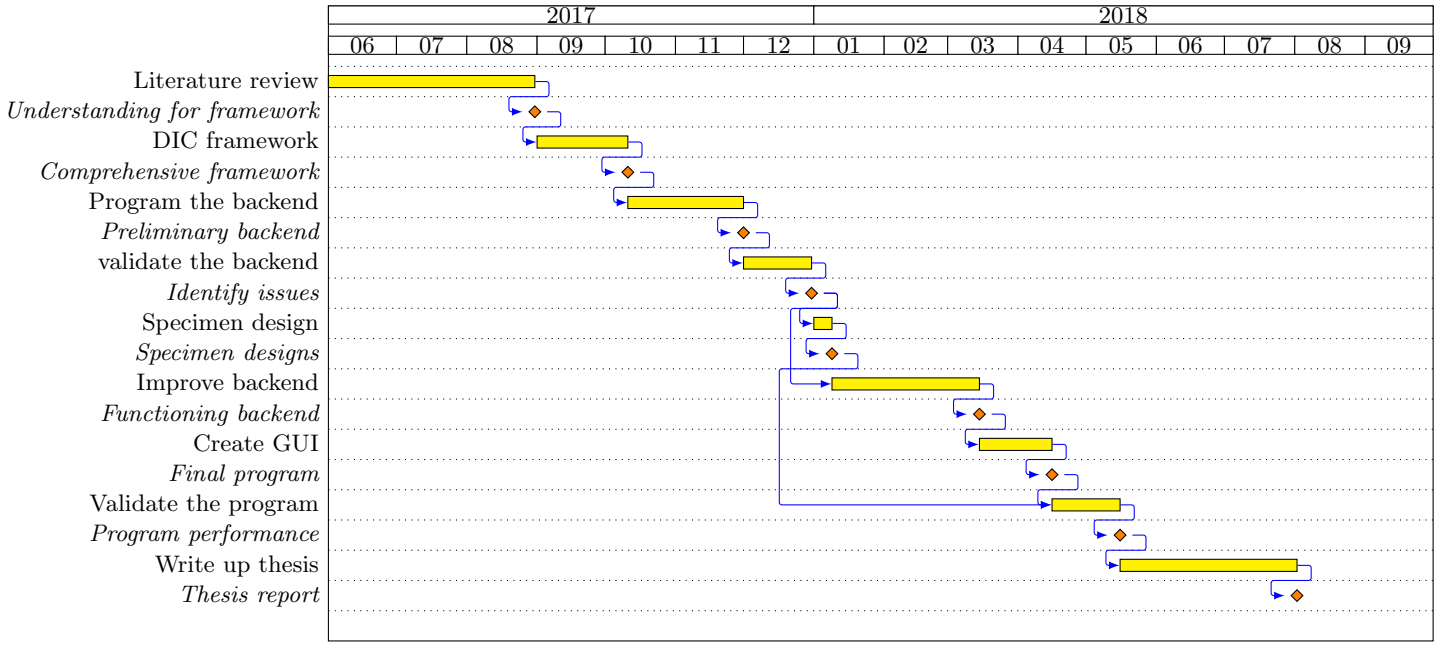


Figure 8.1: Gantt chart

Chapter 9

Conclusion

This document proposes a research project aimed at creating a program for Matlab that is capable of performing DIC or DVC on a given set of images. The proposed program is aimed at providing the user with full control over the correlation process so that users no longer need to treat DIC and DVC as a black box. This project is not intended to have any direct application in industry but rather to provide researchers with a comprehensive tool to perform correlation to determine material deformation.

The motivation of the project has been outlined along with literature review on DIC. The objectives of the project have also been outlined with a research plan which is aimed at achieving these objectives. A time-line for the activities of the research plan is also provided. It is evident that sufficient knowledge and resources are available for the successful completion of this project.

Bibliography

[1] S. Baker and I. Matthews. *Lucas-Kanade 20 Years on: A Unifying Framework*. Number pt. 1 in Technical report. Carnegie Mellon University, The Robotics Institute, 2002.

[2] Marc D. Binder, Nobutaka Hirokawa, and Uwe Windhorst, editors. *Aperture Problem*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[3] Wilhelm Burger. Zhangs camera calibration algorithm: In-depth tutorial and implementation. Technical Report HGB16-05, University of Applied Sciences Upper Austria, School of Informatics, Communications and Media, Dept. of Digital Media, Hagenberg, Austria, May 2016.

[4] JH Conradie, DZ Turnerb, and TH Beckerc. Characterising damage in structural materials using digital image correlation and peridynamics.

[5] Gunnar Farnebäck. Two-frame motion estimation based on polynomial expansion. *Image analysis*, pages 363–370, 2003.

[6] M Grédiac, F Pierron, and Y Surrel. Novel procedure for complete in-plane composite characterization using a single t-shaped specimen. *Experimental Mechanics*, 39(2):142–149, 1999.

[7] Bruce D Lucas, Takeo Kanade, et al. An iterative image registration technique with an application to stereo vision. 1981.

[8] Q-T Luong and Olivier D Faugeras. Self-calibration of a moving camera from point correspondences and fundamental matrices. *International Journal of computer vision*, 22(3):261–289, 1997.

[9] M.A. Sutton, J.J. Orteu, and H. Schreier. *Image Correlation for Shape, Motion and Deformation Measurements: Basic Concepts,Theory and Applications*. Springer US, 2009.

[10] Roger Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal on Robotics and Automation*, 3(4):323–344, 1987.

[11] Sergio Uras, Federico Girosi, Alessandro Verri, and Vincent Torre. A computational approach to motion perception. *Biological Cybernetics*, 60(2):79–87, 1988.

[12] Guo-Qing Wei and Song De Ma. Implicit and explicit camera calibration: Theory and experiments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):469–480, 1994.

[13] Zhengyou Zhang. *Camera Calibration*. Prentice Hall PTR, 2004.