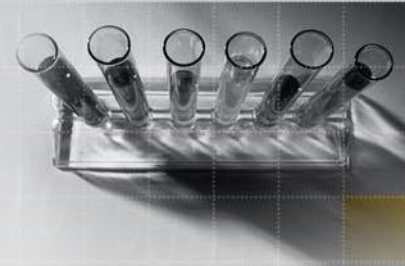




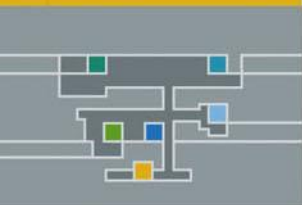
Unifying Framework for Gradient-Descent, Subset Based Digital Image Correlation and Digital Volume Correlation

Devan Atkinson
17732913



2017

FAKULTEIT INGENIEURSWESE
FACULTY OF ENGINEERING



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

Contents	5
List of Figures	6
1 Introduction	7
2 Correlation	8
2.1 Calibration	8
2.1.1 Inverse problem	8
2.1.2 Calibration plate	9
2.1.3 Homography	9
2.1.4 Estimating homography with direct linear transform	10
2.1.5 Absolute conic	10
2.1.6 Constraints on intrinsic parameters	11
2.1.7 Intrinsic parameters and the absolute conic	12
2.1.8 Closed form solution	13
2.1.9 Distortion in Calibration	13
2.2 Correlation criteria	14
2.3 Warp function	15
2.4 Interpolation	16
2.5 Subset matching	16
2.6 Algorithm outline	18
3 Conclusion	19
Bibliography	20

List of Figures

Chapter 1

Introduction

Chapter 2

Correlation

The Lucas-Kanade algorithm was used in combination with the zero-mean normalised sum of squared differences correlation criteria. This correlation criteria was chosen since it takes into account offset and scaling in the light intensities of the images making the overall algorithm more robust. The Lucas-Kanade algorithm assumes that the deformation of the image is constant over a area of the image referred to as the subset. As such this algorithm tries to find the deformation field that best corrects a subset of the deformed image so that it appears to be the same as the reference image.

2.1 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.

explain intrinsic and extrinsic better

take out distortion not used

2.1.1 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 typically 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.

2.1.2 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.

2.1.3 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 ?? 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, \mathbf{H} , for calibration is given by the following equation.

$$\begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix} = \alpha \begin{bmatrix} f_x & f_s & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ 0 \\ 1 \end{bmatrix} \quad (2.1)$$

This can be reduced to

$$\begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix} = \alpha \begin{bmatrix} f_x & f_s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ 1 \end{bmatrix} \quad (2.2)$$

$$= \alpha \mathbf{K} \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{t} \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ 1 \end{bmatrix} \quad (2.3)$$

$$= \alpha \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ 1 \end{bmatrix} = \alpha \mathbf{H} \begin{bmatrix} x_w \\ y_w \\ 1 \end{bmatrix} \quad (2.4)$$

Here \mathbf{r}_1 and \mathbf{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.

2.1.4 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 [1]. Equation 2.4 can be written out as

$$x_s = \alpha (h_1 x_w + h_2 y_w + h_3) \quad (2.5)$$

$$y_s = \alpha (h_4 x_w + h_5 y_w + h_6) \quad (2.6)$$

$$1 = \alpha (h_7 x_w + h_8 y_w + h_9). \quad (2.7)$$

The scale factor, α , can be eliminated by dividing equations 2.5 and 2.6 by 2.7 to get

$$x_s (h_7 x_w + h_8 y_w + h_9) = (h_1 x_w + h_2 y_w + h_3) \quad (2.8)$$

$$y_s (h_7 x_w + h_8 y_w + h_9) = (h_4 x_w + h_5 y_w + h_6) \quad (2.9)$$

These equations then reduce to

$$-h_y x_w x_s - h_8 y_w x_s + h_1 x_w + h_2 y_w + h_3 = h_9 x_s \quad (2.10)$$

$$-h_7 x_w y_s - h_8 y_w y_s + h_4 x_w + h_5 y_w + h_6 = h_9 y_s. \quad (2.11)$$

In order to avoid the trivial solution where every element in the homography 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 singularity [6].

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 2.10 and 2.11. These are then combined into an equation of the form

$$\begin{bmatrix} x_{w_1} & y_{w_1} & 1 & 0 & 0 & 0 & -x_{w_1} x_{s_1} & -y_{w_1} x_{s_1} \\ 0 & 0 & 0 & x_{w_1} & y_{w_1} & 1 & -x_{w_1} y_{s_1} & -y_{w_1} y_{s_1} \\ x_{w_2} & y_{w_2} & 1 & 0 & 0 & 0 & -x_{w_2} x_{s_2} & -y_{w_2} x_{s_2} \\ 0 & 0 & 0 & x_{w_2} & y_{w_2} & 1 & -x_{w_2} y_{s_2} & -y_{w_2} y_{s_2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \\ h_7 \\ h_8 \end{bmatrix} = \begin{bmatrix} x_{s_1} \\ y_{s_1} \\ x_{s_2} \\ y_{s_2} \\ \vdots \end{bmatrix} \quad (2.12)$$

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.

speaking about
normalisation

2.1.5 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, $\tilde{\mathbf{x}}_{ac} = [x, y, z, w]^T$, that satisfies the following.

$$w = 0 \quad (2.13)$$

$$\mathbf{x}_{ac}^T \mathbf{x}_{ac} = x^2 + y^2 + z^2 = 0 \quad (2.14)$$

Here tilde is used to indicate homogeneous coordinates whereas $\mathbf{x}_{ac} = [x, y, z]^T$ 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 \mathbf{x}_{ac} in Euclidean space in the world coordinate system that lies on the absolute conic. It has homogeneous coordinates $\tilde{\mathbf{x}}_{ac} = [\mathbf{x}_{ac}^T, 0]^T$ in projective space. Applying a different rotation and translation to this point to obtain $\tilde{\mathbf{x}}'_{ac}$, which is equivalent to moving the camera in the world coordinate system, a corresponding point is obtained in the camera coordinate system.

$$\tilde{\mathbf{x}}'_{ac} = \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{0} & 1 \end{bmatrix} \tilde{\mathbf{x}}_{ac} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{ac} \\ y_{ac} \\ z_{ac} \\ 0 \end{bmatrix} \quad (2.15)$$

$$= \begin{bmatrix} \mathbf{R}\mathbf{x}_{ac} \\ 0 \end{bmatrix} \quad (2.16)$$

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 \mathbf{x}'_{ac} is on the same absolute conic.

$$\mathbf{x}_{ac}'^T \mathbf{x}'_{ac} = (\mathbf{R}\mathbf{x}_{ac})^T (\mathbf{R}\mathbf{x}_{ac}) = \mathbf{x}_{ac}^T \mathbf{R}^T \mathbf{R} \mathbf{x}_{ac} = \mathbf{x}_{ac}^T \mathbf{x}_{ac} = 0 \quad (2.17)$$

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

$$\tilde{\mathbf{m}}_s = \frac{1}{\alpha} \mathbf{K} \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{ac} \\ 0 \end{bmatrix} = \frac{1}{\alpha} \mathbf{K} \begin{bmatrix} \mathbf{R}\mathbf{x}_{ac} \\ 0 \end{bmatrix} \quad (2.18)$$

$$\therefore \mathbf{m}_s = \frac{1}{\alpha} \mathbf{K} \mathbf{R} \mathbf{x}_{ac}. \quad (2.19)$$

Checking whether this point satisfies equation 2.14 results in

$$\mathbf{m}_s^T \mathbf{K}^{-T} \mathbf{K}^{-1} \mathbf{m}_s = \frac{1}{\alpha^2} \mathbf{x}_{ac}^T \mathbf{R}^T \mathbf{R} \mathbf{x}_{ac} = \frac{1}{\alpha^2} \mathbf{x}_{ac}^T \mathbf{x}_{ac} = 0 \quad (2.20)$$

Thus the image of the absolute conic is itself an imaginary conic. The image of the absolute conic is defined by $\mathbf{K}^{-T} \mathbf{K}^{-1}$ [2]. 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.

2.1.6 Constraints on intrinsic parameters

According to Zhang [6] 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 [6]

$$\begin{bmatrix} \mathbf{r}_3 \\ \mathbf{r}_3^T \mathbf{t} \end{bmatrix}^T \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} = 0. \quad (2.21)$$

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 $[\mathbf{r}_1^T, 0]^T$ and $[\mathbf{r}_2^T, 0]^T$ are points on this line [6]. Thus it is known that any point on this line, x_c^∞ , is a linear combination of these two points.

$$x_c^\infty = a \begin{bmatrix} \mathbf{r}_1 \\ 0 \end{bmatrix} + b \begin{bmatrix} \mathbf{r}_2 \\ 0 \end{bmatrix} = \begin{bmatrix} a\mathbf{r}_1 + b\mathbf{r}_2 \\ 0 \end{bmatrix} \quad (2.22)$$

Now assume that this point, x_c^∞ , lies on the absolute conic. Then this point must satisfy equation 2.14. 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

$$x_c^\infty = a \begin{bmatrix} \mathbf{r}_1 \pm i\mathbf{r}_2 \\ 0 \end{bmatrix}. \quad (2.23)$$

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

$$x_s^\infty = \mathbf{K}(\mathbf{r}_1 \pm i\mathbf{r}_2) = \mathbf{h}_1 \pm i\mathbf{h}_2. \quad (2.24)$$

Substituting these points into equation 2.14 results in

$$(\mathbf{h}_1 \pm i\mathbf{h}_2)^T \mathbf{K}^{-T} \mathbf{K}^{-1} (\mathbf{h}_1 \pm i\mathbf{h}_2) = 0. \quad (2.25)$$

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

$$\mathbf{h}_1^T \mathbf{K}^{-T} \mathbf{K}^{-1} \mathbf{h}_2 = 0 \quad (2.26)$$

$$\mathbf{h}_1^T \mathbf{K}^{-T} \mathbf{K}^{-1} \mathbf{h}_1 = \mathbf{h}_2^T \mathbf{K}^{-T} \mathbf{K}^{-1} \mathbf{h}_2 \quad (2.27)$$

2.1.7 Intrinsic parameters and the absolute conic

An homography, \mathbf{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

$$\mathbf{B} = \mathbf{K}^{-T} \mathbf{K}^{-1} = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{12} & b_{22} & b_{23} \\ b_{13} & b_{23} & b_{33} \end{bmatrix} \quad (2.28)$$

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

$$\mathbf{h}_i^T \mathbf{B} \mathbf{h}_j = v_{ij} \mathbf{b} \quad (2.29)$$

where

$$v_{ij} = [h_{i1}h_{j1}, h_{i1}h_{j2} + h_{i2}h_{j1}, h_{i2}h_{j2}, h_{i3}h_{j1} + h_{i1}h_{j3}, h_{i3}h_{j2} + h_{i2}h_{j3}, h_{i3}h_{j3}]^T. \quad (2.30)$$

Then equation 2.26 can be rewritten as

$$\begin{bmatrix} v_{12}^T \\ (\mathbf{v}_{11} - \mathbf{v}_{22})^T \end{bmatrix} \mathbf{b} = \mathbf{0} \quad (2.31)$$

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

$$\mathbf{V} \mathbf{b} = \mathbf{0}. \quad (2.32)$$

The solution for \mathbf{b} , up to a scale factor λ , is known to be the eigenvector of $\mathbf{V}^T \mathbf{V}$ associated with the smallest eigenvalue [6]. Once \mathbf{b} has been determined it can be used to determine the intrinsic parameters of matrix \mathbf{K} . The relation between \mathbf{B} and \mathbf{K} is $\mathbf{B} = \lambda \mathbf{K}^{-T} \mathbf{K}^{-1}$. The intrinsic parameters are determined as follows.

$$c_y = (b_{12}b_{13} - b_{11}b_{23})/(b_{11}b_{22} - b_{12}^2) \quad (2.33)$$

$$\lambda = b_{33} - (b_{13}^2 + c_y(b_{12}b_{13} - b_{11}b_{23}))/b_{11} \quad (2.34)$$

$$f_x = \sqrt{\frac{\lambda}{b_{11}}} \quad (2.35)$$

$$f_y = \sqrt{\frac{\lambda b_{11}}{b_{11}b_{22} - b_{12}^2}} \quad (2.36)$$

$$f_s = \frac{-b_{12}f_x^2 f_y}{\lambda} \quad (2.37)$$

$$c_x = \frac{f_s c_y}{f_y} - \frac{b_{13}f_x^2}{\lambda} \quad (2.38)$$

Thereafter the extrinsic parameters can be determined.

$$\mathbf{r}_1 = \lambda \mathbf{K}^{-1} \mathbf{h}_1 \quad (2.39)$$

$$\mathbf{r}_2 = \lambda \mathbf{K}^{-1} \mathbf{h}_2 \quad (2.40)$$

$$\mathbf{r}_3 = \mathbf{r}_1 \times \mathbf{r}_2 \quad (2.41)$$

$$\mathbf{T} = \lambda \mathbf{K}^{-1} \mathbf{h}_3 \quad (2.42)$$

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

this section is too close to the paper

2.1.8 Closed form solution

Sections 2.1.3 through to 2.1.7 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 absolute conic which in turn can be used to determine the intrinsic and extrinsic camera parameters. This 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

2.1.9 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 parameters 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 account during calibration [4,5]. This is beneficial since it is possible to solve

add reference for this

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

$$\begin{bmatrix} x_{i_{distorted}} \\ y_{i_{distorted}} \end{bmatrix} = 1 + k_1 r^2 + k_2 r^4 \begin{bmatrix} x_i \\ y_i \end{bmatrix} \quad (2.43)$$

$$\text{where } r = \sqrt{x_i^2 + y_i^2}. \quad (2.44)$$

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 [1]. This error is represented as

$$d_1 = x_{s_{actual}} - x_{s_{calculated}} \quad (2.45)$$

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

$$\tilde{x}_i = u_c + (x_i - u_c) * (1 + k_1 r^2 + k_2 r^4) \quad (2.46)$$

$$= u_c + x_i - u_c + (x_i - u_c) * (1 + k_1 r^2 + k_2 r^4) \quad (2.47)$$

$$= x_i + (x_i - u_c) * (1 + k_1 r^2 + k_2 r^4) \quad (2.48)$$

$$= x_i + d_{2_i} \quad (2.49)$$

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.

2.2 Correlation criteria

A correlation criteria is a mathematical way of quantifying how well a subset in one image matches to a subset in another image. It is used to determine whether the deformation field that has been solved for is sufficiently accurate to explain how the deformed image is related to the reference image. As such it is important that the correlation criteria reliably indicates the fit between the two subsets.

The zero-mean normalised sum of squared differences (ZNSSD) correlation criteria is one of the most robust options for a correlation criteria since it accounts for both lighting offset and scaling between the reference and deformed images. Correlation criteria are based off of a cost function which is usually a variation of the sum of squared difference between the pixel values of a subset of the reference image and a subset of the deformed image. The ZNSSD correlation criteria has the following cost function.

$$\chi_{ZNSSD}^2 = \sum_i (aG_i + b - F_i)^2 \quad (2.50)$$

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

this to zero.

$$\frac{\partial \chi_{ZSSD}^2}{\partial a} = 2 \sum_i (a_{opt} G_i + b - F_i) G_i = 0 \quad (2.51)$$

$$\Rightarrow a_{opt} = \frac{\sum_i (F_i - b) G_i}{\sum_i G_i^2} \quad (2.52)$$

$$\frac{\partial \chi_{ZSSD}^2}{\partial b} = 2 \sum_i (a G_i + b_{opt} - F_i) = 0 \quad (2.53)$$

$$\Rightarrow b_{opt} = \frac{\sum_i F_i - a G_i}{n} \quad (2.54)$$

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. These can be solved to obtain

$$a_{opt} = \frac{\sum_i \bar{F}_i \bar{G}_i}{\sum_i \bar{G}_i^2}, \quad b_{opt} = \bar{F} - \bar{G} \frac{\sum_i \bar{F}_i \bar{G}_i}{\sum_i \bar{G}_i^2} \quad (2.55)$$

$$\text{where } \bar{F}_i = F_i - \bar{F} \quad \text{and} \quad \bar{G}_i = G_i - \bar{G} \quad (2.56)$$

Substituting these into equation 2.50 results in the correlation criteria.

$$\chi_{ZSSD}^2 = \sum_i \left(\left(\frac{\sum_i \bar{F}_i \bar{G}_i}{\sum_i \bar{G}_i^2} G_i - \bar{G} \frac{\sum_i \bar{F}_i \bar{G}_i}{\sum_i \bar{G}_i^2} \right) - F_i + \bar{F} \right)^2 \quad (2.57)$$

2.3 Warp function

When applying DIC to material science the deformation behaviour of materials under load must be taken into account. This is because 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. For this analysis the common warp function which accounts for affine transformations is used since it is consistent with the strains that the material is expected to experience. Thus by determining the warp function parameters for a specific subset; the strains for that subset are also determined.

$$W(\zeta, \mathbf{p}) = \begin{bmatrix} \Delta x_{warp} \\ \Delta y_{warp} \end{bmatrix} = \begin{bmatrix} 1 + \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} & u \\ \frac{\partial v}{\partial x} & 1 + \frac{\partial v}{\partial y} & v \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ 1 \end{bmatrix}. \quad (2.58)$$

Here 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 whereas $\frac{\partial v}{\partial x}$ and $\frac{\partial u}{\partial y}$ represent the shear deformation of the subset. All of these variables that explain how the subset is warped and translated are stored within the vector $\mathbf{p} = \left[u \quad \frac{\partial u}{\partial x} \quad \frac{\partial u}{\partial y} \quad v \quad \frac{\partial v}{\partial x} \quad \frac{\partial v}{\partial y} \right]$ which are referred to as the warp function parameters. Additionally Δx and Δy are the distances from the centre of the undeformed subset to the pixel under consideration and are contained within the variable $\zeta = [\Delta x \quad \Delta y \quad 1]^T$. The outputs Δx_{warp} and Δy_{warp} are the modified distances from the centre of the subset to the pixel under consideration. Thus it is the

pixel locations within the subset that are modified in order to induce warp and not the subset itself which is warped by this function. In other words the Δx and Δy are warped by the function which has the effect of warping the subset itself since it changes the locations within the subset that are to be evaluated. Thus the warp function applied to a subset is mathematically expressed as

$$F_{warped} = F(\mathbf{x}_0 + W(\zeta, \mathbf{p})) \quad (2.59)$$

where $\mathbf{x}_0 = [x_0 \ y_0]^T$ are the x and y coordinates, in units of pixels, of the centre of the subset.

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.

keep this?

2.4 Interpolation

It is clear that warping a subset according to equation 2.59 requires that the light intensity values between pixel locations can be determined. This is an issue because images store this information in a discrete manner. This is solved by using interpolation to determine the light intensity values between pixels based on the light intensity values of neighbouring pixels. For the purposes of this project Matlab's built in interp2 function was used to perform linear interpolation to determine the light intensity values for non-standard pixel locations where needed.

2.5 Subset matching

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

check

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 Lucas-Kanade method can be implemented in two distinct ways that differ in correlation criteria function. The first, the original way, is referred to as the forward additive method which applies the current estimate of the warp function and the iterative improvement to the warp function parameters to the investigated subset to deform it in an attempt to obtain a subset closer to the reference subset. This deformed subset is then compared with the reference subset and the difference between the two is used to incrementally improve the warp function parameter estimates. In this method the Hessian matrix is dependent on the investigated subset's light intensity values, G , and as such it needs to be computed for each iteration of the solver.

The second method, the inverse compositional method, is a modification to the forward additive method in an effort to decrease the computational cost per iteration. For this method the current estimate for the warp function parameters, \mathbf{p} , is applied to the investigated subset while the iterative improvement to the warp function parameters, $\Delta \mathbf{p}$, is applied to the reference subset. This is done so that the Hessian that needs to be determined depends only on the reference subset and is independent of the warp function parameters

which allows it to be pre-computed outside of the iterations. As a result this method is more efficient which means it takes less time to converge to the solution.

The algorithm explained here is based on the one proposed by B. Pan [3]. It is the inverse compositional Lucas-Kanade method using the Gauss-Newton approximation to the Hessian matrix. The algorithm used to solve for the optimal parameters of the warp function is derived from the correlation criteria used as given below.

$$\chi_{ZNSSD}^2 = \sum_{\zeta} \left[\frac{F(\mathbf{x}_0 + W(\zeta, \Delta \mathbf{p})) - \bar{F}}{\Delta F} - \frac{G(\mathbf{x}_0 + W(\zeta, \mathbf{p})) - \bar{G}}{\Delta G} \right]^2 \quad (2.60)$$

Notice that the current approximation for the warp function parameters, \mathbf{p} , is applied to the investigated subset while the iterative improvement to the warp function parameters, $\Delta \mathbf{p}$, is applied to the reference subset. Taking the first-order Taylor expansion of equation 2.60 in terms of $\Delta \mathbf{p}$ gives

$$\chi_{ZNSSD}^2 = \sum_{\zeta} \left[\frac{F(\mathbf{x}_0 + \zeta) + \nabla F \frac{\partial W}{\partial \mathbf{p}} \Delta \mathbf{p} - \bar{F}}{\Delta F} - \frac{G(\mathbf{x}_0 + W(\zeta, \mathbf{p})) - \bar{G}}{\Delta G} \right]^2 \quad (2.61)$$

where $\nabla F = \begin{bmatrix} \frac{\partial F(\mathbf{x}_0 + \zeta)}{\partial x} & \frac{\partial F(\mathbf{x}_0 + \zeta)}{\partial y} \end{bmatrix}$ is the gradients of the reference subset in the x and y directions and $\frac{\partial W}{\partial \mathbf{p}} = \begin{bmatrix} 1 & \Delta x & \Delta y & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta x & \Delta y \end{bmatrix}$ is the Jacobian of the warp function.

Taking the derivative of equation 2.60 with respect to $\Delta \mathbf{p}$ and setting it to zero gives the least squares solution for finding $\Delta \mathbf{p}$.

$$\frac{\partial \chi_{ZNSSD}^2}{\partial \Delta \mathbf{p}} = 0 = 2 \sum_{\zeta} \left[\frac{F(\mathbf{x}_0 + \zeta) + \nabla F \frac{\partial W}{\partial \mathbf{p}} \Delta \mathbf{p} - \bar{F}}{\Delta F} - \frac{G(\mathbf{x}_0 + W(\zeta, \mathbf{p})) - \bar{G}}{\Delta G} \right] \frac{\nabla F \frac{\partial W}{\partial \mathbf{p}}}{\Delta F} \quad (2.62)$$

$$\Delta \mathbf{p} = -H^{-1} \times \sum_{\zeta} \left[(\nabla F \frac{\partial W}{\partial \mathbf{p}})^T \times \left[(F(\mathbf{x}_0 + \zeta) - \bar{F}) - \frac{\Delta F}{\Delta G} (G(\mathbf{x}_0 + W(\zeta, \mathbf{p})) - \bar{G}) \right] \right] \quad (2.63)$$

$$\text{where } H = \sum_{\zeta} \left[(\nabla F \frac{\partial W}{\partial \mathbf{p}})^T (\nabla F \frac{\partial W}{\partial \mathbf{p}}) \right] \quad (2.64)$$

Here H is the Hessian matrix which remains constant since it depends only on the reference subset. Thus the Hessian matrix can be computed before the algorithm starts iterating in order to reduce computational costs. However in order to achieve this reduced computational cost the iterative improvement to the warp function parameters was applied to the reference subset. This means that the iterative improvement cannot simply be added to the old warp function parameters to obtain the updated warp function parameters. Instead the updated warp function parameters must be determined using the following equation.

$$W(\zeta, \mathbf{p}_{new}) = W(\zeta, \mathbf{p}_{current}) \times W(\zeta, \Delta \mathbf{p})^{-1} \quad (2.65)$$

$$= \begin{bmatrix} 1 + \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} & u \\ \frac{\partial v}{\partial x} & 1 + \frac{\partial v}{\partial y} & v \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 + \Delta \frac{\partial u}{\partial x} & \Delta \frac{\partial u}{\partial y} & \Delta u \\ \Delta \frac{\partial v}{\partial x} & 1 + \Delta \frac{\partial v}{\partial y} & \Delta v \\ 0 & 0 & 1 \end{bmatrix}^{-1} \quad (2.66)$$

The new approximation to the warp function parameters can then be extracted from this warp function matrix $W(\zeta, \mathbf{p}_{new})$.

2.6 Algorithm outline

The steps performed within the algorithm in order to achieve correlation are listed here in order to illustrate how the theory presented here is related to the code.

1. First \bar{F} is determined for the reference subset.
2. Then ∇F is calculated using Matlab's built in image gradient function (`imgradientxy`) and the Jacobian of the warp function is calculated. These are then used to determine the Hessian matrix.
3. Thereafter ΔF is determined since it too remains constant.
4. The initial guess for the warp function parameters is then used to calculate the Δx_{warp} and Δy_{warp} for the investigated subset G using equation 2.58.
5. Δx_{warp} and Δy_{warp} are used in equation 2.59 to obtain the warped version of the investigated subset.
6. \bar{G} and ΔG are then determined.
7. With all the necessary values determined; equation 2.63 can be used to calculate Δp .
8. Then equation 2.66 is used to determine the updated approximation to the warp function parameters.
9. Step 4 until step 8 are repeated until convergence or until the warp function parameters are considered good enough.

Chapter 3

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] Wilhelm Burger. Zhangs camera calibration algorithm: In-depth tutorial and implementation. *month*, 2016.
- [2] 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.
- [3] B Pan, K Li, and W Tong. Fast, robust and accurate digital image correlation calculation without redundant computations. *Experimental Mechanics*, 53(7):1277–1289, 2013.
- [4] 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.
- [5] 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.
- [6] Zhengyou Zhang. *Camera Calibration*. Prentice Hall PTR, 2004.