Adaptive Digital Image Correlation Using Neural Networks

by
Devan James Atkinson

Dissertation presented for the degree of
Doctor of Philosophy
in the Faculty of Engineering at
Stellenbosch University

Supervisor: Prof. Thorsten Hermann Becker
Co-supervisor: Dr Melody Neaves

March 2023

Declaration

By submitting this dissertation electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

This dissertation includes 2 original papers published in peer-reviewed journals or books and 1 unpublished publications. The development and writing of the papers (published and unpublished) were the principal responsibility of myself and, for each of the cases where this is not the case, a declaration is included in the dissertation indicating the nature and extent of the contributions of co-authors.

Date: March 2023

Abstract

Subset size selection is crucial to the accuracy and precision of digital image correlation (DIC) measured displacements. Increasing the subset size improves noise suppression (reducing random errors) at the cost of spatial resolution (ability to accurately measure complex displacement fields). The tradition of global correlation parameter assignment is suboptimal because the speckle pattern quality and displacement field complexity can vary spatially. Dynamic subset selection (DSS), which assigns location specific optimal subset sizes, is challenging because the metrological performance of correlation is dictated by complex interactions between correlation parameters (subset size and shape function) and image set properties (noise, speckle pattern and displacement field complexity).

This dissertation uses an open-source DIC framework to investigate the potential of artificial neural networks (ANNs) for error prediction and DSS, prior to the DIC process, from purely image information. ANNs are capable of modelling complex relationships within noisy, incomplete data without imposing fixed relationships, inspiring their recent resurgence for DIC applications.

Despite the plethora of open-source DIC algorithms available, none offer spatially and temporally independent assignment of correlation parameters. Subsequently, a modular, open-source DIC framework capable of such flexibility is developed. This framework is predominantly consistent with current state-of-the-art practices and performs on par with well-established open-source and commercial DIC algorithms. Drawing direct links between the well-documented theory of DIC and its nuanced practical implementation, bridges this gap in literature which has acted as a barrier to newcomers intending to develop the capabilities of DIC. This framework, implemented in 117 and 202 lines of MATLAB code for 2D and stereo DIC, respectively, is attractive as a starting point to further the capabilities of DIC.

The feed-forward ANN developed using this DIC framework, predicts random errors based on the speckle pattern quality (contained within a subset) and standard deviation of image noise more accurately and precisely than established theoretical derivations. A DSS framework is developed which uses this ANN to appoint subset sizes, based on the local speckle pattern, that offer random errors consistent with a stipulated threshold value. Appropriate selection of the random error threshold offers a favourable compromise between noise suppression and spatial resolution for up to moderate displacement gradients. Consequently, in the presence of varying speckle pattern quality this framework outperforms the traditional approach of trial-and-error global subset size selection for the same mean subset size. Speckle pattern characteristics outside the training scope reveal the generalisability limitations of the DSS method, and associated ANN, as it performs on par with the traditional global subset size approach, motivating the need to broaden its training scope.

Investigation of convolutional neural networks for dynamic shape function selection is initiated, showing they are capable of quantifying displacement field complexity between image pairs to guide spatially and temporally independent shape function assignment.

The dissertation reveals that ANNs are an attractive approach to model the

metrological characteristics of the correlation process for the purpose of dynamic correlation parameter assignment. Furthermore, such models facilitate dynamic correlation parameter assignment from purely image information such that they can operate as a pre-process to DIC.

Opsomming

Seleksie van subsetgrootte is deurslaggewend vir die akkuraatheid en presiesheid van digitale beeld korrelasie (DBK) gemete verplasings. Die verhoging van die subsetgrootte verbeter geraas vermindering (verminder ewekansige foute) ten koste van ruimtelike resolusie (vermoë om komplekse verplasingsvelde akkuraat te meet). Die tradisie van globale korrelasieparametertoewysing is suboptimaal omdat die spikkelpatroonkwaliteit en verplasingsveldkompleksiteit ruimtelik kan varieer. Dinamiese subset-seleksie (DSS), wat liggingspesifieke optimale subsetgroottes toeken, is uitdagend omdat die metrologiese prestasie van korrelasie bepaal deur komplekse interaksies tussen korrelasieparameters (subsetgrootte en vormfunksie) en beeldstel eienskappe (geraas, spikkelpatroon en verplasingsveld kompleksiteit).

Hierdie proefskrif gebruik ’n oopbron-DIC-raamwerk om die potensiaal van kunsmatige neurale netwerke (KNN’e) vir foutvoorspelling en DSS, voor die DBK-proses, uit slegs beeldinligting te ondersoek. KNN’e is in staat om komplekse verhoudings binne raserige, onvolledige data te modelleer sonder om vaste verhoudings af te dwing, wat die onlangse herlewing van die gebruik van KNN’s vir DBK-toepassings inspireer.

Ten spyte van die oorvloed van oopbron DBK-algoritmes wat beskikbaar is, bied nie een ruimtelike en tydelike onafhanklike toewysing van korrelasieparameters nie. Vervolgens word ’n modulêre, oopbron DBK-raamwerk ontwikkel wat in staat is tot sulke buigsaamheid. Hierdie raamwerk stem oorwegend ooreen met huidige moderne praktyke en presteer op gelyke voet met goed gevestigde oopbron- en kommersiële DBK-algoritmes. Deur direkte verbande te trek tussen die goed gedokumenteerde teorie van DBK en die genuanseerde praktiese implementering daarvan, word hierdie gaping in literatuur, wat as ’n hindernis opgetree het vir nuwelinge wat van plan is om die vermoëns van DBK te ontwikkel, oorbrug. Hierdie raamwerk, geïmplementeer in 117 en 202 reëls MATLAB-kode vir onderskeidelik 2D en stereo DBK, is aantreklik as ’n beginpunt om die vermoëns van DBK te bevorder.

Die vooruitvoer KNN wat met hierdie DBK-raamwerk ontwikkel is, voorspel ewekansige foute gebaseer op die spikkelpatroonkwaliteit (vervat in ’n subset) en standaardafwyking van beeldgeraas meer akkuraat en presies as gevestigde teoretiese afleidings. ’n DSS-raamwerk word ontwikkel wat hierdie KNN gebruik om subsetgroottes aan te stel, gebaseer op die plaaslike spikkelpatroon, wat ewekansige foute bied wat ooreenstem met ’n gestipuleerde drempelwaarde. Toepaslike keuse van die ewekansige foutdrempel bied ’n gunstige kompromie tussen ruisonderdrukking en ruimtelike resolusie tot en met ‘n matige verplasingsgradiënte. Gevolglik, in die teenwoordigheid van wisselende spikkelpatroonkwaliteit, presteer hierdie raamwerk beter as die tradisionele benadering van proef-en-fout globale subsetgrootteseleksie vir dieselfde gemiddelde subsetgrootte. Spikkelpatroonkenmerke buite die opleidingsomvang openbaar die veralgemeenbaarheidsbeperkings van die DSS-metode, en geassosieerde KNN, aangesien dit op gelyke voet met die tradisionele globale subsetgrootte-benadering presteer, wat die behoefte motiveer om sy opleidingsomvang te verbreed.

Ondersoek van konvolusionele neurale netwerke vir dinamiese vormfunksieseleksie word geïnisieer, wat toon dat hulle in staat is om verplasingsveldkompleksiteit tussen beeldpare te kwantifiseer om ruimtelike en tydelike onafhanklike vormfunksietoewysing te lei.

Die proefskrif onthul dat KNN'e 'n aantreklike benadering is om die metrologiese kenmerke van die korrelasieproses te modelleer vir die doel van dinamiese korrelasieparametertoewysing. Verder fasiliteer sulke modelle dinamiese korrelasieparametertoewysing vanaf suiwer beeldinligting sodat hulle as 'n voorproses tot DBK kan funksioneer.

Acknowledgements

To my incredible supervisor Prof. Thorsten H. Becker: I thank you for all that you have done to enable me to get to this stage in my academic career; I would not have been able to do it without you. You have an tremendous tact for supervision which is why I think you have had so many successful MEng and PhD postgraduate students. You know how to bring someone under your wing, impart knowledge and wisdom to them while encouraging and developing their own unique skills and insights, and once they catch wind let them take flight.

I thank you for your unwavering dedication and effort towards bringing this dissertation, as well the contained articles, to completion; for nurturing my passion for research and development while imparting valuable skills and knowledge beyond the world of academics. I thank you for the countless conversations and discussions that sparked my scientific curiosity and compelled me to dive deep into down the rabbit hole, although often times too deep :). I wish you all the best in your future endeavours and in your recently new chapter of life.

To my amazing co-supervisor Dr Melody Neaves: I had no idea when you were tutoring my class as a Demi that you would eventually be guiding me in my academic pursuits. Thank you for all the effort and dedication you have put towards the completion of this dissertation; and for all your advice and input (for my academic career and beyond). I can see that you are going to inspire many great minds to contribute to the field of scientific knowledge.

To my friends: Gavin, Warren, Barry, Luca, Dyllan, Robby, Adie, Alec, Ivan, The Mav, Luke, Candice, Gerrit (and those who are slipping my sleep deprived mind). You have all helped me indirectly by simply being who you are, I would never have been able to get to this stage, nor be who I am without you. To a special lady, Carla thank you for all your support, understanding and keeping me sane.

Last, but certainly not least, I would like to thank my family: Edmund, Kim, Chris, Matthew and Sirrocco. Thank you for all the sacrifices, support and effort you have put in to allow me to pursue my studies to this point.

Dedication

I dedicate this work to my parents; Edmund and Kim Atkinson. Your hard work and determination has inspired and enabled me to pursue my dreams.

I am forever grateful.

Contents

List of Figures

List of Tables

Abbreviations

2D	two-dimensional
3D	three-dimensional
ANN	artificial neural network
ANNDSFS	artificial neural network based dynamic shape function selection
ANNDSS	artificial neural network based dynamic subset selection
BNN	biological neural network
CCD	charge-coupled device
CMOS	complementary metal oxide semiconductor
CNN	convolutional neural network
CS	coordinate system
CT	calibration target
DIC	digital image correlation
DICe	Digital Image Correlation engine
DSFS	dynamic shape function selection
DSS	dynamic subset selection
DVC	digital volume correlation
FEM	finite element method
FFT	fast Fourier transform
FIS1	first image series
FIS2	second image series
IC-GN	inverse compositional Gauss-Newton
iDICs	International Digital Image Correlation Society
LBFGS	limited-memory Broyden-Fletcher-Goldfarb-Shanno
MDPI	Multidisciplinary Digital Publishing Institute
ME	mean error
MSAC	m-estimator sample consensus
NR	Newton-Raphson
PCM	phase correlation method

PIB pattern induced bias

pix pixel

PSF point-spread function

RBT rigid body translation

ReLU rectified linear unit

RGDT reliability-guided displacement tracking

RGW-DIC rotated Gaussian weighted digital image correlation

RMSE root-mean square error

ROI region of interest

SF shape function

SFAB shape function attenuation bias

SFP shape function parameter

SIFT scale-invariant feature transform

SPQM speckle pattern quality metric

SR spatial resolution

SSE subset Shannon entropy

SSSIG sum of square of subset intensity gradient

ZNCC zero-mean normalised cross-correlation

ZNSSD zero-mean normalised sum of squared difference

Nomenclature

$a_j^{(l)}$	The j^{th} neuron of the l^{th} layer of an artificial neural network (ANN)
$a_k^{(l-1)}$	The k^{th} neuron of the $(l-1)^{\text{th}}$ layer of an ANN
$\frac{\partial a_j^{(l)}}{\partial z_j^{(l)}}$	Derivative of the activation function of the j^{th} neuron of the l^{th} layer with respect to its input signal
$b_j^{(l-1)}$	Bias applied to the signal received by the j^{th} neuron of the l^{th} layer of an ANN
β_1	Portion of the first moment of the derivative of the loss function retained during ANN training by the Adam method
β_2	Portion of the second moment of the derivative of the loss function retained during ANN training by the Adam method
c_s	Skew of sensor coordinate system (intrinsic camera parameter)
CE_{loss}	Cross-entropy loss function
χ_i^{pred}	Output value predicted by ANN
χ_i^{true}	Target (true) output value desired from the ANN
d	Counter for images of a set
ε	Smoothing term used for Adam method
ϱ	Learning rate for ANN training
$\frac{\partial f_i}{\partial x}$	Intensity gradient in the x-direction for the i^{th} pixel of the reference subset
γ	Momentum term used during ANN training
g_t	The current derivative of the loss function with respect to the parameter under consideration
g_{t-1}	Derivative of the loss function with respect to the parameter under consideration for the previous iteration of training
L	Number of layers within an ANN (counter it l)
M	Subset size (counter for contained pixels is m)
ME_x	Systematic error in the x-direction
ME_x^{thresh}	Systematic error threshold in the x-direction

MSE_{loss}	Mean squared error loss function
m_t	First moment of the derivative of the loss function
m_0	Initial value of the first moment of the derivative of the loss function
$\hat{m}_t$	Bias corrected first moment of the derivative of the loss function
$N_{(l)}$	Number of neurons within the l^{th} layer of an ANN
N_L	Number of neurons in the output layer of an ANN
ϖ	Shift of sinusoidal function
ψ	Standard deviation of Gaussian noise
ϕ	Activation function for neurons of an ANN
ϕ'	Derivative of an ANN activation function
q	Counter for query points of a DIC analysis
σ^{thresh}	Random error threshold
σ_x	Random error in the x-direction displacements
σ_y	Random error in the y-direction displacements
σ_u^{Glorot}	Standard deviation of the uniform distribution for the Glorot weight initialisation scheme
σ_n^{Glorot}	Standard deviation of the normalised distribution for the Glorot weight initialisation scheme
σ_u^{He}	Standard deviation of the uniform distribution for the He weight initialisation scheme
σ_n^{He}	Standard deviation of the normalised distribution for the He weight initialisation scheme
θ_{t+1}	The updated weight or bias parameter value
θ_t	The current weight or bias parameter value
u	Displacement in the x-direction
$\frac{\partial^2 u}{\partial x^2}$	Second-order x-displacement gradient in the x-direction
$\frac{\partial^2 u}{\partial y^2}$	Second-order x-displacement gradient in the y-direction
Υ	Symbol representation of the loss function of an ANN
$\frac{\partial \Upsilon}{\partial w_{jk}^{(l-1)}}$	Derivative of the loss function with respect to the weight applied to the signal sent from the k^{th} neuron of the $(l-1)^{\text{th}}$ layer to the j^{th} neuron of the l^{th} layer of an ANN
$\frac{\partial \Upsilon}{\partial b_j^{(l-1)}}$	Derivative of the loss function with respect to the bias applied to the signal received by the j^{th} neuron of the l^{th} layer of an ANN

$\frac{\partial \Upsilon}{\partial a_j^{(l)}}$	Derivative of the loss function with respect to the activation function of the j^{th} neuron of the layer l^{th} layer of an ANN
$\frac{\partial^2 v}{\partial x^2}$	Second-order y-displacement gradient in the x-direction
$\frac{\partial^2 v}{\partial y^2}$	Second-order y-displacement gradient in the y-direction
v_t	Second moment of the derivative of the loss function
v_0	Initial value of the second moment of the derivative of the loss function
$\hat{v}_t$	Bias corrected second moment of the derivative of the loss function
ϑ	Period of a sinusoidal function
$w_{jk}^{(l-1)}$	The weight applied to the signal sent from the k^{th} neuron of the $(l-1)^{\text{th}}$ layer to the j^{th} neuron of the l^{th} layer of an ANN
x	Position within an image in x-direction
y	Position within an image in y-direction
$z_j^{(l)}$	Signal received by the j^{th} neuron of the l^{th} layer of an ANN
$\frac{\partial z_j^{(l)}}{\partial w_{jk}^{(l-1)}}$	Derivative of the input signal of the j^{th} neuron of the l^{th} layer with respect to the weight applied to this signal by the k^{th} neuron of the $(l-1)^{\text{th}}$ layer of an ANN
$\frac{\partial z_j^{(l)}}{\partial b_j^{(l-1)}}$	Derivative of the input signal of the j^{th} neuron of the l^{th} layer of an ANN with respect to its bias

Chapter 1
Introduction

The ability to effectively manipulate materials has been strongly linked by historians to the advancement of humanity, to the extent that eras of history are referred to by the materials from which mankind was capable of forging tools, utensils and weaponry [1]. Current society is no exception with the knowledge generated within the field of materials science prescribing what is possible at the current state-of-the-art limits of the manufacturing and construction industries; thereby defining the infrastructure of modern civilisation.

In particular, advances facilitated by materials science include: (i) semiconductors and the modern electronics industry [2]; (ii) implants in the medical industry [3, 4]; (iii) materials necessary for the construction of the various tools and equipment that has driven forward many fields of scientific research (Hadron collider [5], space exploration, space probes, Hubble telescope [6]); and (iv) nanotechnology [7], to name a few. Such advances are made possible through the use of material models and material properties, the development and documentation of which is a primary area of focus of materials science. Material models characterise the mechanisms and phenomena that define the response of a material to external stimuli while material properties dictate the degree of this response for specific materials.

Digital image correlation (DIC) has proven invaluable for the development of such material models. Its ability to measure full-field displacements and deformations aids the investigation of material mechanisms, expediting validation of material models. Furthermore, this full-field data is attractive for material property determination.

DIC involves a complex measurement chain consisting of the optical system which focuses light, the charge-coupled device (CCD) or complementary metal oxide semiconductor (CMOS) which captures this light as images, the geometric setup of the experiment dictating the relative position between the optical system and specimen, the correlation process which computes the pixel displacements and projection of these displacements onto the surface of the specimen. Although errors are introduced at each of these stages, advances in the equipment, techniques and algorithms have reduced these errors significantly. More specifically, competition within industry has encouraged great advancement in cameras (CCD arrays) and optical systems. Knowledge and wisdom gained by DIC practitioners and researchers has been compiled into extensive resources, such as the International Digital Image Correlation Society (iDICs) Good Practices Guide [8], to guide appropriate experimental setup and specimen preparation to obtain high quality image data. The collective efforts of the pioneers of DIC [9, 10, 11, 12, 13, 14, 15] have led to correlation algorithms that are robust, more resistant to image noise and offer high precision and accuracy (on the order of 0.01 pixels [16, 17, 18]).

Although these improvements have greatly reduced errors across all stages of the measurement chain, each correlation process is unique in that its performance is primarily influenced by the correlation parameters stipulated by the DIC practitioner. The appropriateness of these correlation parameters for the analysis dictates, to a large degree, the quality of the computed displacements. Moreover, the magnitude of the displacement error is governed by the complex interaction between the correlation parameters, image noise, speckle pattern quality and displacement field complexity; the latter three of which are properties of the image set. Although resources such as the iDICs's Good Practices Guide provides recommendations for selection of appropriate global correlation parameters, these guidelines are empirically based and seek to ensure sufficient noise suppression and stability of the correlation process. Thus, they do not guarantee the lowest overall displacement error. Additionally, such guidelines typically cater towards the field of experimental solid mechanics where paint applied to the specimen constitutes the speckle pattern. They are not directly applicable to broader DIC applications which rely on natural surface texture to constitute the speckle pattern.

Furthermore, literature suggests location specific optimal correlation parameters are required to obtain the best displacement measurement at each query point due to the spatially and temporally varying image set properties [19, 20, 21]. A lack of a widely accepted method for automatic assignment of appropriate correlation parameters spatially and temporally has been identified as an essential step to furthering the capabilities of DIC [22]. Such a method would ensure high quality displacement measurements[1] by appropriately setting up the correlation problem independently of the practitioner, giving confidence to results. The development of such a method has proven challenging due to the complex interaction between the correlation parameters, image set properties and the resulting displacement error.

Artificial neural networks (ANNs) have been outperforming traditional methods in many instances for modelling complex relationships [23] and have shown great success in the field of computer vision [24]. Subsequently, recent research efforts within the DIC community have investigated the potential of ANNs for DIC applications [25, 26]. ANNs offer a promising approach to model the complex relationship between the correlation parameters, image set properties and the resulting displacement errors to facilitate appropriate assignment of location specific optimal correlation parameters.

This doctoral project extends from an earlier MEng project conducted by the author which focused on the development of a modular DIC open-framework for 2D DIC. Following this MEng project, work on the doctoral project commenced in 2019 at Stellenbosch University. The first year was spent refining the DIC framework by extending its modularity as well as reviewing existing literature. Subsequent efforts focused on extending the framework to stereo DIC. Thereafter, research shifted towards the investigation of ANNs for the purpose of dynamic assignment of correlation parameters.

This chapter serves to introduce the project providing the context of the work. This begins by establishing the basic concepts of DIC aiding comprehension of the ensuing topics, such as the success of DIC in experimental solid mechanics as well as many other scientific fields and industry. Subsequently, the main barriers preventing assignment of appropriate correlation parameters are acknowledged prior to discussing the accomplishments of ANNs within computer vision and DIC. Finally, the aim of the project is summarised before discussing the outline of the dissertation.

---

[1] High quality displacement measurements refers to those of low systematic and random errors.

1.1 DIC nomenclature

DIC determines the displacement and deformation experienced by a specimen from images captured of it. More specifically, images captured of the specimen undergoing load application visually record the displacement and deformation it experiences which are conveyed by the high contrast speckle pattern applied to its surface. The first of these images, the reference image representing the specimen in its unloaded state, is broken up into clusters of pixels called subsets. Each subset contains a unique light intensity pattern due to the speckle pattern applied. Subsequent images are referred to as deformed images and represent the specimen in its loaded state.

The correlation process of DIC considers a single image pair at a time; composed of the reference image and the deformed image under consideration. The two commonly employed types of DIC perform correlation in fundamentally different ways. Local DIC tracks each subset independently between the reference and deformed image. Large subset sizes are used as a form of regularisation to ensure the contained light intensity pattern is distinctive such that it can be uniquely tracked. Subsequently, it makes use of a local shape function (SF) which enables these subsets to deform as well as displace. Accounting for the deformation experienced by the subset enables local DIC to obtain accurate displacement measurements in the presence of displacement gradients. Traditionally, local DIC assigns a global subset size and SF across all subsets of the analysis. Thus, the subset size and SF are the correlation parameters which setup the correlation problem.

Conversely, global DIC treats the subsets of the reference image as nodes within a finite element method (FEM) type mesh and tracks all subsets between the reference and deformed image simultaneously [27]. This FEM mesh, functioning as a global SF, provides a form of regularisation that enforces a degree of continuity in the displacements of neighbouring subsets. Typically, global DIC makes use of smaller subsets relative to local DIC.

The goal of correlation, in both instances, is to determine the shape function parameters (SFPs) which minimise the discrepancy between the reference and deformed subsets or images for local or global DIC, respectively. This discrepancy is quantified by the correlation criterion. Thus, regardless of the type of DIC, correlation relies upon the image intensity gradients, imparted by the speckle pattern, to guide the iterative updates applied to the SFPs to optimise the correlation criterion. In doing so, the correlation process determines the displacements experienced by the query points, which correspond to the subset centres, resulting in a displacement field.

Performing correlation across all deformed images results in a set of displacement fields which record the displacement experienced by the specimen at points in time corresponding to the acquisition of the deformed images. Displacement transformation converts these full-field, pixel displacements to metric displacements in the real world. This relies upon the determination of calibration parameters which describe the positional relationship between points on the specimen's surface and corresponding coordinates within the image. The calibration and displacement transformation process differs based on the number of cameras used. 2D DIC uses a single camera to determine in-plane displacements while stereo DIC uses two or more cameras to account for both in-plane and out-of-plane motion. Additionally, DIC was extended to digital volume correlation (DVC) [28] which uses volumetric images, such as X-Ray tomography-computed images, to determine three-dimensional (3D) displacement fields accounting for displacements in all three dimensions.

Appropriate differentiation of this displacement field provides the strain field experienced by the specimen. Although global DIC enforces locally heterogeneous deformation, it is computationally expensive and computes displacement fields of poorer precision relative to local DIC [27]. This project focuses on local DIC.

1.2 Applications of DIC

Adoption of DIC was initially slow with over a decade passing between the introduction of the core concept by Anuta [29] in 1970 and the first instance of its use in an experimental mechanics context in 1983 [30]; where it was shown to be capable of measuring the displacement of a cantilever beam. This prompted development of the method which through the collective efforts of the pioneers of the field [9, 10, 11, 12, 13, 14, 15] led to improved robustness, efficiency, accuracy and precision. Subsequently, it has become an essential tool within the field of experimental solid mechanics as noted by Pan [22]:

> *"In the experimental mechanics community, DIC has been considered as a revolutionary advance; the most important advance since the strain gauge. It will continue to be the most practical and powerful deformation measuring tool for the foreseeable future."*

Experimental solid mechanics, a subset of material science, focuses on the experimental investigation of the mechanical response of solids subjected to imposed boundary conditions for the purpose of developing material models that quantify these relationships. Providing the fixed boundary conditions (forces applied) and tolerable response (displacement and deformation) of a component as inputs and outputs to such models, respectively, guides appropriate material selection and component dimensioning. These models are essential for the modern approach of component design enabling reliable and efficient components to be designed at the conception stage.

Experimental solid mechanics links theoretical and applied mechanics to: (i) experimentally investigate the response of materials to external stimuli to understand the underlying phenomena and mechanisms responsible such that theoretical (constitutive) equations and models can be developed to characterise the material behaviour; (ii) prove the effectiveness and applicability of these models through experimental validation; and (iii) determine the material specific parameters (material properties) of these models to build a library of material properties so that these models can be readily utilised at the conception stage of component design. DIC has been shown to be well suited to these processes.

Firstly, the full-field displacement and deformation information measured by DIC allows for effective investigation of the complex interactions between different mechanisms and phenomena dictating the response of the material to boundary conditions. Read [31] used DIC in his investigations of bituminous paving mixtures to directly record the crack path. He subsequently discovery that cracks travel along a favoured route between the matrix and coarse aggregate; in particular favouring the shortest path between the location of crack initiation and site of load application. Such behaviour was previously not observable through traditional equipment such as crack opening displacement gauges and crack foil circuits which indirectly measure crack depth.

Employing DIC on scanning electron microscopy topography images, Kang et al. [32] measured the microscopic strain distribution, at the grain level, of dual phase steels. Subsequently, partitioning of strain between the ferritic and martensitic regions was discovered. Furthermore, it was observed that tempering of the metal alters the microscopic strain distribution altering the tensile properties, damage accumulation and ductility. Additionally, high-speed cameras have enabled the investigation of non-quasi-static behaviour of materials such as the behaviour of composites upon impact [33], the deformations occurring during cold rolling [34] and deformation of tire walls [35].

Secondly, the full-field displacement and strain fields are effective for spatial validation of theoretical predictions and models. In particular, DIC has been successfully employed to validate FEM models predicting the strain anticipated in proximal femur subjected to gait loading [36]. Seo et al. [37] and Chehab et al. [38] employed DIC to extend the validation of viscoelastoplastic continuum damage model, of Kim and Little [39] and Kim et al. [40], to forecast strains beyond localisation up to the point of macro-crack formation and propagation, not previously feasible with linear variable differential transformers.

Lastly, its ability to determine full-field displacements makes DIC particularity attractive for material property determination. Conventional displacement and deformation measurement equipment takes measurements at a localised region (strain gauge) or between two points (extensometer). This necessitates the development of meticulous standards for the determination of material properties. Such standards include the ASTM Standard E8 [41] which relies upon using an extensometer to determine yield and tensile strength and the ASTM Standard E132 [42] which uses a strain gauge to determine Poisson's ratio at room temperature.

These standards place strict requirements on the specimen geometry such that deformation corresponding to the measurement area is directly dependent on the material property to be measured while mitigating that resulting from other material properties. This is done so that specialised constitutive equations which are calibrated for the specimen geometry, in terms of how the displacement is measured and how other material properties affect the deformation within the measurement zone, can be used to accurately quantify the material property. Consequently, a different specimen is required to accurately determine each material property.

However, the full-field displacement and deformation data offered by DIC has allowed for new approaches to material property determination, such as the Virtual Fields Method [43] and Finite Element Model Updating Method [44]. These utilise more complex models, accounting for the impact of a collection of material properties on the resulting deformation, to determine multiple materials properties from a single experiment. These approaches relax the restrictions on the specimen geometry to only require that the material properties under investigation play a significant enough role in the deformation of the material while the impact of other material properties are mitigated or easily accounted for.

DIC's non-contact nature presents several advantages. It allows for the investigation of the response of soft materials without altering their material properties; not possible using traditional metrology equipment such as strain gauges. Lee et al. [45] utilised this advantage to investigate the impact of the mixing ratio and curing temperature on the coefficient of thermal expansion of polydimethylsiloxane. Additionally, this enables investigating the response of a material exposed to harsh environments since DIC only requires a clear view of the specimen to perform displacement measurement. Bolivar et al. [46] used this to conduct in-situ experiments on 600 Alloy exposed to a tetrathionate solution to investigate stress corrosion crack interactions. By utilising DIC and applying a spatially varying temperature field, van Rooyen and Becker [47] extracted multiple temperature specific Poisson's ratios and elastic moduli in the range of 480 °C to 600 °C from a single specimen.

Although most of the development of DIC has focused on improving it for experimental solid mechanics applications, its versatility has led to its successful application and growing interest in many other fields of research and industry. In the geotechnical engineering community, it has been identified as a transformative tool for analysing failure mechanisms and characterising soil and soil-structure interactions both within laboratory testing and on-site monitoring [48]. DIC has gained interest as a time efficient, cost effective and non-destructive quality control tool capable of defect detection in pre-stressed concrete railroad ties [49] and identification of manufacturing defects in elastomeric bridge bearings [50].

DIC has shown great potential in the medical field including: (i) DIC for full-field human pulse monitoring [51, 52]; (ii) utilisation of DIC to develop an inverse finite element analysis framework for investigation of strain distributions and local pulmonary force on lungs due to ventilation strategies [53] ; (iii) verification of the, decades long assumed, incompressibility of brain tissue [54]; and (ii) measurement of strain distributions in multi-vertebra spine segments [55].

1.3 Barriers to correlation parameter assignment

Accurate and precise displacement measurements are crucial for all applications of DIC outside preliminary investigations of the phenomena governing the kinematic response of an imposed boundary condition. In particular, experimental solid mechanics requires accurate and precise displacement measurements such that model validation and material property determination are reliable. The DIC technique has been refined to such a degree that the dominant factor affecting the displacement error is the appropriateness of the assigned subset size and SF order for the image set properties as detailed in Section 2.1.

Here it suffices to understand that increasing the subset size, and SF order, improves the random error at the potential expense of the systematic error, and vice versa [56]. Thus, since increasing the subset size and SF order have counteracting effects on the displacement error, there exists an optimal combination of the two which results in the lowest overall displacement error. This ideal combination is dictated by the image set properties in the vicinity of the query point.

Although image noise is consistent throughout an image set as it is dictated by the camera sensor utilised [57, 58], the speckle pattern quality and displacement field complexity vary. Speckle pattern quality typically varies spatially due to the difficulty experienced by the practitioner in maintaining factors that affect the consistency of the speckle size and coverage during the application of sprayed paint. The displacement field complexity varies spatially, due to the location of load application and the specimen geometry, and temporally, due to the time dependent magnitude of load applied. This suggests a location and image specific optimal subset size and SF order combination to realise the best (most accurate and precise) displacement measurement.

However, a global subset size and SF are typically used in practice across all subsets of an analysis, due to the two barriers hindering assignment of such optimal correlation parameters on a per-subset and per-image basis. Firstly, the lack of a corroborated and robust framework for determining the optimal correlation parameters. Although assigning the optimal subset size and SF as the combination that returns the lowest displacement error is a simple concept, this optimal combination is obscure and challenging to determine without a model of the complex interaction between correlation parameters, image set properties and resulting displacement error.

Attempts have been made to circumvent this by publications [59, 19, 21] which put forth metrics that quantify the quality of the speckle pattern contained within the subset and illustrating that these speckle pattern quality metrics (SPQMs) are related to the resulting displacement error. These publications propose dynamic subset selection (DSS) by appointing the location specific subset size as the smallest that offers a SPQM value satisfying a threshold value. However, these methods are entirely based on empirical observation in terms of both relating the SPQMs to the displacement error and the selection of an appropriate threshold. As such, they have found limited use in real world applications.

Conversely, Pan et al. [60] employed a theoretical error analysis to derive a SPQM, termed the sum of square of subset intensity gradient (SSSIG), which is directly related to the random error in the displacements. This was extended by Wang and Pan [61] to derive the random displacement errors as a function of the image noise, SSSIG and SF order. This provides crucial knowledge of the role that correlation parameters play in dictating the displacement error. However, the assumptions made and simplistic correlation algorithm utilised (to ease the complexity of the mathematical derivation) prevents these relations from being directly applicable to most DIC instances currently in use. Furthermore, as stated by Pan [22]:

> *"Since the actual deformation occurring in each subset cannot be known in advance in practical applications, an approach is still lacking for the adaptive selection of optimal subset size and shape function for each calculation point, which can comprehensively consider local speckle pattern quality and local complex deformation state."*

Secondly, to the authors knowledge, no DIC algorithms currently available allow for the per-subset and per-image assignment of these correlation parameters. The capabilities of DIC and its wide range of applications and use cases has led to the development of a plethora of DIC algorithms. Many commercial DIC software packages exist which offer sophisticated DIC solutions. Although these commercial DIC packages offer efficient and robust displacement measurement for established applications, their high cost (in the face of falling prices and improved capabilities of cameras and computers) has driven the development of open-source DIC algorithms to improve the accessibility of DIC. The design philosophy of these open-source DIC algorithms, focusing on user-friendliness as well as robust and efficient computation, has made them a viable alternative to commercial packages. However, these algorithms, like their commercial counterparts, fail to offer full control of the correlation parameters allowing independent assignment of the subset size and SF order on a per-subset and per-image basis.

1.4 ANNs in computer vision and DIC

ANNs are programs designed to mimic the way the human brain processes information by using networks of interconnected artificial neurons. They are capable of identifying complex, non-linear patterns occurring within representative data without the need to establish rigid relationships portraying the underlying mechanisms [62]. Furthermore, they are simple to implement, robust if developed appropriately, and have the ability to deal with noisy data [63].

ANNs have been particularly successful in image analysis applications. This began in 1980 when Fukushima [64] introduced the neocognitron, the predecessor of the modern convolutional neural network (CNN), capable of pattern recognition tasks such as identifying hand written Japanese characters. However, most advances in image based ANNs occurred after 2012 when the CNN AlexNet [65] outperformed traditional image classification algorithms in the ImageNet Large Scale Visual Recognition Competition [66]. This stimulated image based ANN research with CNNs dominating this competition ever since.

This research has led to ANNs revolutionising the field of computer vision, providing efficient solutions to problems that had remained unresolved for extensive periods [67], including: (i) image classification based on image content [68]; (ii) locating and identifying objects within images [69]; (iii) semantic segmentation of images identifying pixels belonging to various objects or classes [70]; (iv) upsampling images from low to high resolution [71, 72]; (v) colourisation of grey-scale images [73, 74]; (vi) style transfer enabling an image to be transformed into *artwork* in the style of a particular artist [75]; (vii) image synthesis applying specific modifications to an image [76]; and (viii) reconstruction of partial or corrupted images [77].

These capabilities have enabled successful application of ANNs in many practical image processing tasks. In the medical field, ANNs are attractive to detect biological anomalies in medical imagery. For instance ANNs have been used to identify regions, within high-resolution computed tomography imagery, containing interstitial lung disease [78]. Self-driving cars leverage the capabilities of CNNs for tasks such as lane recognition, path planning, scene classification and understanding, pedestrian detection, motion control, traffic sign recognition and obstacle detection [79]. Additionally, ANNs have been utilised for quality control; particularly in the food industry [80]. This widespread success of ANNs for image processing has led to their emergence in the field of DIC and materials research in two significant ways.

Firstly DIC computed displacements and strains have been successfully utilised as input data for ANNs for the purpose of structural health monitoring. DIC data is attractive for this purpose since: (i) the kinematic response of structures is directly related to their damage state; (ii) there is a large pool of structural health monitoring related image data available to train these ANNs (while FEM provides an additional option to bolster the training set inexpensively); and (iii) the ability of DIC to adapt temporally and spatially to capture the anticipated damage mechanisms affords a large amount of representative features.

As such, using DIC computed data, ANNs have been developed for crack path detection in AA2024-T3 rolled sheets based on displacement fields [81], high accuracy crack length estimation in aluminium from strain fields [82] and damage detection in composite sandwich structures [83] as well as carbon fibre reinforced plastics [84] from vibration and strain data, respectively.

Secondly, ANNs have been implemented within the DIC process for displacement and strain computation. Initial attempts, as early as 2001 [85], utilised a hybrid method where integer pixel displacements were determined by traditional approaches, such as the phase correlation method (PCM) [86], while an ANN was used to estimate the sub-pixel component of displacement [87, 88]. These techniques have not found widespread adoption since, despite their computational efficiency, their accuracy and precision was on par with that of established sub-pixel algorithms of the time.

Late 2019 [26] saw a resurgence in ANN-based DIC research utilising a fundamentally different approach inspired by the success of CNNs for optical flow estimation [89, 90, 91]. In particular, Min et al. (2019) [26], Boukhtache et al. (2021) [25], Ma et al. (2021) [92], Yang et al. (2022) [93], Wang et al. (2022) [94] and Duan et al. (2023) [95] utilised CNNs to compute the underlying displacement fields directly from an image pair.

These ANN-based DIC methods are attractive in that they avoid the need for setting up the correlation problem such that the displacement results are independent of the practitioner, compute pixel-wise displacement fields, can account for complex deformations [25] and are computationally efficient (in some cases offering essentially real-time measurement [94]). Additionally, Deep DIC [93] and StrainNet [25] computes strains directly from image pairs.

Although these methods seem to resolve the need to set up the correlation problem, the generalisability of the ANN, largely dictated by the scope of its training dataset, directly affects its performance. In particular, Deep DIC [93] performs poorly for simple rigid body translation (RBT) since they were not included in the training dataset while StrainNet [25] (which computes both displacements and strains) is incapable of handling displacements larger than a pixel. Furthermore, these training datasets are composed entirely of synthetic images due to the difficulty and cost of obtaining experimental images of known complex deformations. Thus, the applicability of these ANNs to real world applications is questionable, particularly in the cases where the natural surface texture of the specimen is used to constitute the speckle pattern.

Consequently, the teething problems of ANN-based DIC techniques suggest that their adoption will be gradual, indicating that traditional DIC algorithms will remain prevalent for the foreseeable future. However, the resurgence of ANNs in the field of DIC, offering appealing benefits and capabilities, indicates that they are likely to disrupt the status quo as has occurred in many other computer vision related fields. This is substantiated by recent publications (October 2022) that use CNNs to determine SFP initial estimates [96] and attempt to quantify global speckle pattern quality [97].

1.5 Outline of aim

ANNs could offer a feasible approach to: (i) model the complex relationship between the subset size, the contained speckle pattern quality, image noise and the resulting displacement error; and (ii) spatially quantify the underlying displacement field complexity between image pairs. The aim of the project is to *investigate the viability of ANNs for dynamic correlation parameter assignment from purely image information using an open-source DIC framework*. Chapter 3 is dedicated to developing this aim into objectives and defining the project scope.

1.6 Dissertation outline

This section briefly presents the philosophy behind the structure of the dissertation serving as a guide for those who wish to take a front-to-back approach, while functioning as a roadmap for individuals who prefer to focus on specific chapters.

Chapters 4, 5 and 6, which align with the first three objectives of this project as detailed in Section 3.3, form the core of this dissertation. These chapters are presented in their original (pre-review) form, contain a preamble detailing the contributions of the co-authors and are concluded with a post-submission discussion of the insights gained during the peer review process and literature published post-submission. Each of these core chapters is self-contained including an overview of the background literature pertinent to the objective addressed and a list of chapter specific symbols.

As such, Chapter 2 presents background reading summarising contextual information relevant to the project. This includes a summary of the most pertinent DIC errors and a discussion of the theory of ANNs. Chapter 3 develops the dissertation's core hypothesis by motivating the research directions pursued, and refining these research directions into the objectives which collectively give rise to the framework used to investigate the validity of the project's hypothesis. The hypothesis is then clarified prior to establishing the scopes and limitations of the project.

Chapter 7 discusses the core chapters in terms of their strengths and shortfalls highlighting how they contribute to the aim of the dissertation. Additionally, it presents an investigation into the viability of CNNs for dynamic shape function selection (DSFS) and outlines the foundation for future work directions. Finally, Chapter 8 closes off the dissertation with conclusions summarising the key findings and contributions of this work, highlighting their alignment with the project's objectives detailed in Section 3.3.

Chapter 2
Background reading

The following chapter introduces the metrological characteristics of DIC and discusses the role of this knowledge in guiding its effective use. Thereafter, the theoretical background of ANNs is presented.

2.1 DIC metrological characteristics

Considerable research efforts undertaken since the 1990's have lead to the identification and quantification of various error sources present in the DIC error chain. The metrological characteristics of DIC are dictated by these error sources which are categorised as either experimental or numerical errors as detailed below. The way in which knowledge of these error sources has guided the development and effective use of DIC is presented before this section is concluded with a short discussion of the challenges faced in analysing the metrological characteristics of DIC.

2.1.1 Experimental errors

Experimental errors are introduced during the image acquisition phase of DIC as a result of experimental setup and equipment used. For instance, noise is introduced within the images during their acquisition as a result of: (i) dark current noise [98] which occurs when thermally generated electrons increase the charge of an element of the CCD array despite it not being exposed to light (consequently spurious light intensity levels are captured in the image); (ii) photon shot noise [99] which is a consequence of the variation in the photons' rate of arrival when they strike the CCD array; and (iii) read noise [100] caused by the conversion of the charge of an element of the CCD array into a voltage signal and the subsequent analogue-to-digital conversion of this signal. Image noise greatly affects the measurement errors as it reduces the signal-to-noise ratio; the intensity gradient caused by the speckle pattern (or speckle pattern information) being the signal of interest.

Additionally, the quality of the contained speckle pattern within the image is affected by blur and variations in illumination. Blur is caused by camera or sample motion, limited depth of field or defocussing of the camera setup and lens imperfections [101]. Although modern correlation criteria can account for offset and scaling changes in illumination making them robust against variations in ambient light [102], regions reflecting high intensity light can cause random displacement errors in these regions to increase five fold [103].

As shown by Lava et al. [104], lens distortions can cause systematic errors in the measured displacements. It is for this reason that radial and tangential distortions are included within the camera model to reduce their impact on the displacement error. However, regardless of the scope of factors affecting the optical setup that the camera model is designed to account for, poor quality calibration, causing significant errors in the computed camera and distortion parameters, will cause substantial systematic errors in the computed displacements [105].

Self-heating of cameras during image acquisition is a common phenomenon. Ma et al. [106] noted that this can lead to appreciable systematic errors in computed displacements as a result of thermal deformations of the camera's mechanical components. Additionally, heat sources in the experimental setup, present in many DIC applications such as the sun in field work [107] and furnace for testing heated samples [47], cause light to pass through air of heterogeneous temperature and density. The resulting heterogeneous index of refraction of the air distorts the location and form of objects being imaged such that spurious displacements are computed [108].

Out-of-plane motion will causes errors for 2D DIC as it attributes this motion to the computed in-plane displacement components [109]. Although stereo DIC offers robustness against out-of-plane motion, it has been shown that the accuracy of its displacement errors are subject to the angle between the cameras (stereo angle) [105, 110].

2.1.2 Numerical errors

Numerical errors refer to those arising during the numerical process of correlation. DIC is fundamentally an ill-posed problem since it attempts to determine displacement magnitude and direction from scalar image information [111]. Additionally, displacements can only be determined in the direction of the image intensity gradient [112]. That is, the image gradient guides the correlation process indicating the direction and magnitude of the iterative adjustments that need to be made to the SFPs such that the correlation criterion is optimised.

Consequently, local DIC tracks subsets of pixels (instead of individual pixels) to provide intensity gradients in both the x- and y-directions such that displacements in these directions can be uniquely determined. It is for this reason that many publications recommend that a subset contain at least 3-4 pixels such that it can be uniquely tracked [8, 20]. Although this form of regularisation symptomatically addresses the ill-posed problem, it does not guarantee that the measured displacement matches the true displacement. Moreover, irrespective of its quality, all speckle patterns act as filters allowing only specific parts of the real underlying displacement field to pass through such that they can be accurately tracked by DIC [111]. As such, for a large enough subset which can be uniquely tracked, the error associated with this ill-posed problem is dictated by the speckle pattern and underlying displacement field of the image set.

Interpolation is used within the correlation process to sample intensity values at non-integer locations within the images enabling sub-pixel displacements to be measured. Interpolation changes the phase and amplitude of the intensity pattern of the original image introducing errors within the DIC computed displacements. Schreier et al. [12] showed that lower order interpolation methods suffer due to phase errors. Although higher order interpolation reduces these phase errors, this comes at the cost of errors due to poor amplitude attenuation. As such, they recommend the use of higher order interpolation methods, in particular bi-quintic b-spline interpolation, to reduce phase errors while using low-pass filtering of the images to mitigate errors due to poor amplitude attenuation.

Use of a low-pass filter additionally benefits the temporal variance error [113] caused by the image noise introduced by experimental error sources. The intensity pattern within the image is a combination of that due to the speckle pattern, which conveys the displacement experienced by the specimen, and random image noise. Differentiation amplifies the noise present within the computed intensity gradient which is used to guide the correlation process. Consequently, noisy updates are iteratively applied to the SFPs, as the correlation process attempts to track both the image noise and the underlying displacement conveyed by the speckle pattern, causing random errors in the final displacements.

The magnitude of these random errors are related to the standard deviation of the image noise (ψ), the SSSIG value of the subset, and the SF order [61] as shown in Figure 2.1. Here $\sum_{i=1}^{M}\sum_{j=1}^{M}\frac{\partial f(i,j)}{\partial x}^{2}$ represents the SSSIG in the x-direction for a square subset of size M, where $\frac{\partial f(i,j)}{\partial x}$ is the intensity gradient in the x-direction at pixel location (i,j).

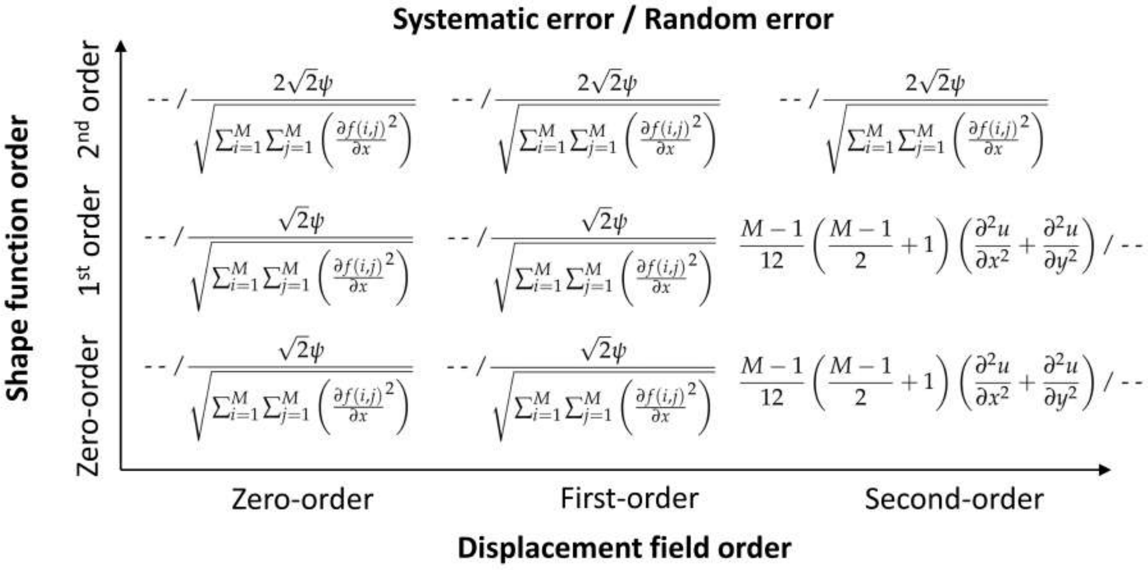


Figure 2.1: Theoretically derived relations for the dominant systematic and random errors in the x-direction adapted from [22] (relations are similar in the y-direction). "- -" indicates that the systematic or random errors are negligible for the associated combination of SF order and underlying displacement field complexity.

Applying a low-pass filter to the image smooths out the high-frequency noise (reducing ψ), while retaining the bulk of the speckle pattern information, thereby reducing the random displacement errors [114]. Alternatively, increasing the subset size, such that it contains more speckle pattern information (increasing SSSIG while ψ remains unchanged), effectively averages out the impact of noise reducing the temporal variance error.

However, increasing the subset size can increase systematic errors due to shape function attenuation bias (SFAB) [115]. SFAB occurs for under-matched SFs which are of lower order than the underlying displacement field such that they are incapable of accounting for the deformation experienced by the subset. Subsequently, in an attempt to improve the correlation criterion the correlation process spuriously shifts the subset, causing systematic errors, to compensate for the portion of the deformation not accounted for by the SF. Increasing the subset size such that it incorporates pixels farther from the query point, which experience increasingly incongruent displacements relative to the subset centre, leads to greater errors due to SFAB.

Xu et al. [116] theoretically derived that for a first-order SF and second-order displacement field the SFAB error is related to the second-order derivative of the displacement field and the subset size as shown in Figure 2.1. Here $\frac{\partial^2 u}{\partial x^2}$ and $\frac{\partial^2 u}{\partial y^2}$ are the second-order derivatives of the x-displacement (u) in the x- and y-directions, respectively, at the centre of the subset. As such, higher order SFs are used when the displacement field within an image set is known to be complex *a priori*. However, the increase in the number of SFPs leads to reduced noise suppression and higher random errors due to temporal variance errors. In particular, the equations of Figure 2.1 show that the random error of the second-order SF is double that of the first-order.

Within this work the impact of SFAB on the displacement errors is often discussed in terms of spatial resolution (SR) which is the accuracy to which DIC can track the underlying displacements in regions of high displacement field complexity (high displacement gradients). Local DIC measures the displacement at the centre of a subset as the average of that experienced across the subset. Thus, the accuracy to which DIC can measure the displacement at the subset centre decreases as the displacement gradients increase, particularly for under-matched SFs. SR is typically investigated using imposed sinusoidal displacement fields and is quantified as the lowest period of the sinusoidal displacement for which DIC can accurately capture the displacement amplitude within a certain percentage of its true value [56]. For an in-depth discussion of SR refer to [117].

Pattern induced bias (PIB) [118] occurs when a subset contains an imbalance in the distribution of speckle pattern information with respect to its centre. For under-matched SFs, the spatially imbalanced distribution of image gradients causes the measured displacements to be offset from the true displacement in the direction of the region of the speckle pattern with the higher concentration of intensity gradient information.

2.1.3 Metrological characteristics informing effective use of DIC

Knowledge of the experimental error sources guides appropriate experimental design and image acquisition, according to resources such as the Good Practices Guide [8], such that the high quality image sets and calibration parameters can be obtained (relative to what is achievable). Additionally, knowledge of the numerical error sources has guided the design of the correlation algorithm. In particular, modern correlation algorithms favour higher order interpolation methods and make use of more robust correlation criteria to reduce associated errors.

Recent literature recommends the use of regular patterns, such as checkerboard patterns, to eliminate PIB [119, 120]; however, this is contrary to conventional requirement of an isotropic, non-periodic speckle patterns of high information density [121] for unique tracking. Thus, it is most effective to mitigate PIB, along with SFAB, by ensuring appropriate selection of the SF and subset size for an analysis to avoid under-matched SFs.

Additionally, in practical applications the subset size required to address temporal variance errors ensures that the subset size is large enough to effectively mitigate errors due to the ill-posed problem. Therefore, numerical errors are predominantly dictated by how appropriate the subset size and SF are for the query point to be analysed. In fact, inappropriate assignment of these parameters will cause errors which outweigh those introduced by experimental error sources (provided guidelines are followed for experimental design and image acquisition).

Subsequently, the traditional approach of appointing a global subset size and SF order is becoming outdated. More specifically, this approach relies on appointing a global subset size that is large enough to mitigate the temporal variance errors for the region of the speckle pattern of poorest quality. Similarly, the SF order assigned needs to account for the highest expected displacement field complexity which is typically for a limited number of query points corresponding to stress concentrations and regions of plastic deformation (which develops at the onset of failure towards the end of the image set). As such, the region of poorest speckle pattern quality and highest displacement field complexity, which directly affect the minority of query points analysed, dictates the quality of the measured displacement field.

Although the theoretical equations relating the SSSIG to the temporal variance error and the second-order displacement gradient to the SFAB error offer an attractive alternative for dynamic correlation parameter assignment, assumptions made during their theoretical derivation limit their applicability in this regard for practical applications. In particular, the SSSIG relation ignores the contribution of image noise to the SSSIG value and assumes interpolation errors to be negligible.

Similarly, the SFAB relation only predicts the systematic error associated with the use of a first-order SF for a quadratic displacement field. Displacement fields seldom fall within such strict definitions and quantifying the complexity of the underlying displacement field remains a challenge as noted by Pan [22].

2.1.4 Challenges for metrological characterisation of DIC

Although the traditional approach of mitigating or removing known error sources to identify or investigate others has lead to the knowledge of metrological characteristics of DIC currently available, this approach is not without limitations. Firstly, this does not benefit understanding the full scope of how these error sources interact with one another to produce the final displacement error. More specifically, Balcaen et al. [105] observed that the DIC error chain involves complex interactions between error sources in which they can either amplify one another or cancel each other out.

Secondly, this method can hide the presence of error sources which lead to similar errors as those already documented. For instance, 18 years passed between PIB initially being noticed by Schreier and Sutton [115] and it being acknowledged within the DIC community. Fayad et al. [118] hypothesised that PIB was overlooked because it produces errors of similar magnitude and spatial randomness (due to the spatially varying speckle pattern) to that of temporal variance errors. Thus, it can be argued that there is a need for an alternative approach to investigate the metrological characteristics of DIC such that undiscovered error sources can be identified and the complex interaction of error sources can be understood. Such discoveries will lead to an improved understanding of the DIC error chain aiding the development of additional corrective measures and informed guidelines advocating the effective use of DIC.

Another challenge in the metrological characterisation of DIC is the differing approaches that publications use to quantify the metrological performance of DIC; both in terms of the images used to analyse the performance (with many publications using self generated images) and the metrics used to quantify the performance. In this work DIC performance is investigated using images sets provided in the DIC Challenge version 1 [122] and the Stereo DIC Challenge [123] such that the results reported are reproducible and comparable to established DIC algorithms. Furthermore, in accordance with the DIC Challenge version 1, the accuracy and precision of a DIC analysis is quantified as the mean and standard deviation of the displacement errors, respectively. It should be noted that during this work the DIC Challenge was updated to version 2 [124] which proposed updated image sets and a new approach of quantifying the metrological performance of DIC algorithms using the *metrological efficiency indicator* metric according to the work of Blaysat et al. [56].

2.2 Background on ANNs

ANNs refers to a subset of artificial intelligence which is composed of computational algorithms which aim to mimic the way in which the brains of invertebrates and mammals processes information through a biological neural network (BNN). They are capable of learning complex relationships and patterns within representative data. ANNs can be divided into two categories based on whether they are trained through unsupervised or supervised learning.

In unsupervised learning, the output for the provided data is unknown. Thus, instead of the ANN trying to model a specific relationship between inputs and outputs, it is expected to perform operations on the data to discover the hidden relationships within. In particular, unsupervised ANNs typically perform one of three tasks: (i) clustering by which the unlabelled data is grouped depending on how similar or dissimilar the datapoints are; (ii) dimensionality reduction to decrease the dimensionality of the unlabelled data while keeping the integrity of the data intact such that the data can be represented in a compressed format; and (iii) association to discover the underlying structure of the dataset whereby relationships and patterns between variables of the dataset are discovered. Thus, unsupervised ANNs complete their tasks upon conclusion of the learning process.

For supervised learning the target outputs for a set of inputs are known and the ANN is trained to map the relationship between them. Supervised learning typically results in ANNs of higher accuracy relative to unsupervised learning but requires more involvement from the deep learning practitioner to curate the training dataset. This higher accuracy is necessary since supervised ANNs are typically designed for deployment. More specifically, once a supervised ANN can accurately model the relationship of interest, within the training dataset, they are deployed on real world data for which the true output is unknown.

Although ANNs were first introduced by McCulloch and Pitts in 1943 [125], the lack of a viable method for training multilayer ANNs lead to stagnation of the field. It was not until 1986 when Rumelhart et al. [126] proposed the concept of backward propagation (backpropagation) that multilayer ANNs could be trained. This combined with the greater accessibility to computational resources and the large database of information made available by the internet enabled more complex and powerful ANNs to be effectively trained. The resulting success drove evermore increasing interest and investment into the field such that the current body of research contains a plethora of ANN types and architectures.

Presenting a detailed account of the theoretical background that is applicable to such a wide range of ANNs is beyond the scope of this project. As such, the following section presents the theory applicable to ANNs relevant to this project; that is ANNs which are trained via supervised learning and propagate information in a single direction through the network (feedforward). Moreover, this section outlines the functionality of artificial neurons drawing parallels with their biological counterparts, presents the commonly used activation functions, details the considerations involved in the design stage of ANN architecture, discusses the process used to train ANNs via supervision, and outlines why ANNs are advantageous over other machine learning methods due to their ability to extract multilevel representations from the input data.

2.2.1 Artificial neurons and layering

In simplified terms a BNN is a highly interconnected network of biological neurons linked to one another by synaptic connections. Synaptic connections facilitate the one directional flow of information between neuron pairs in the form of electrical or chemical signals. A neuron is capable of both receiving a signal (input) and generating a signal (output). A neuron will "fire" generating an output signal if the amalgamation of the input signals it receives from one or more preceding neurons is sufficient to excite it. The signal it generates is sent via the synaptic connection to its neighbouring, connected neurons.

The adaptability of the synaptic connections is crucial to the ability of BNNs to effectively process information. As noted by Hebb in 1949 [127], neurons which frequently facilitate the firing of a neighbouring neuron will strengthen the synaptic connection between them, expediting the flow of information. Consequently, a neuron which repeatedly causes a neighbouring neuron to fire will lead to increased likeliness of this action in the future. In this way chains of firing neurons, which respond strongly to the input signal, form neural pathways which become adept at processing particular types of information.

ANNs are more structured than their biological inspiration; consisting of L many layers of interconnected artificial neurons with $N_{(l)}$ many neurons in the l^{th} layer, as shown in Figure 2.2. The first and last layer are referred to as input and output layers, respectively, with hidden layers falling in-between. The input layer receives the information that is to be processed by the ANN, each of its neurons receiving a component of the input signal. This information, typically of numerical form, is processed as it traverses the hidden layers which perform operations such as data transformation and feature detection. Finally, this processed information is passed to the output layer which reorganises it into the desired output format.

The interconnectivity between the layers dictates how the information flows between neurons. In general, ANNs make use of fully connected layers, introduced by Rossenblatt in 1957 [128], where each neuron of the preceding layer is connected to each neuron of the subsequent layer.

Although the mathematical operations occurring between layers of neurons are simple in conception, they become convoluted as layers are stacked in deeper ANN

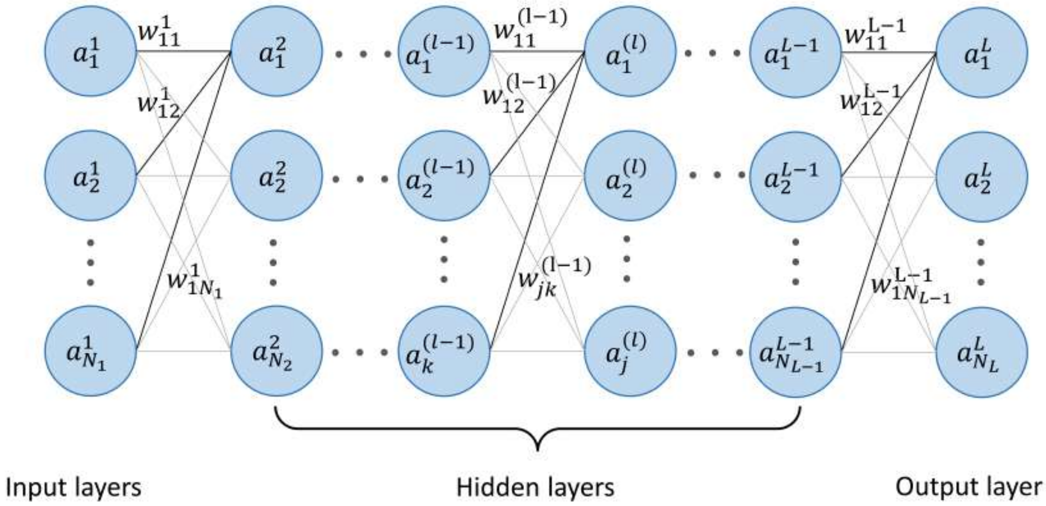


Figure 2.2: Schematic illustration of the structure of a typical feed-forward ANN composed of input, hidden and output layers.

architectures. As such, these operations are discussed in terms of occurring between two adjacent arbitrary layers; layer $l-1$ representing the preceding layer and layer l representing the subsequent layer. Using this convention the j^{th} neuron of the l^{th} layer is represented as $a_j^{(l)}$ while $a_k^{(l-1)}$ represents the k^{th} neuron of the preceding layer, as shown in Figure 2.2.

Artificial neurons mathematically mimic the functionality of their biological counterpart receiving input signals from one or more neurons of the preceding layer. Each input has a weight applied to it representing the strength of their connection similar to the synaptic connection strength of a BNN. The weight between $a_j^{(l)}$ and $a_k^{(l-1)}$ is represented as $w_{jk}^{(l-1)}$. Thus, the signal received by $a_j^{(l)}$, represented as $z_j^{(l)}$, is given as

$$z_j^{(l)} = \sum_{k=1}^{K} w_{jk}^{(l-1)} a_k^{(l-1)} + b_j^{(l-1)} \tag{2.1}$$

where $b_j^{(l-1)}$ represents the bias applied to $a_j^{(l)}$. Upon receiving this input, the neuron applies its activation function (ϕ) to determine the degree to which the input signal stimulates the artificial neuron as

$$a_j^{(l)} = \phi\left(z_j^{(l)}\right) \tag{2.2}$$

Non-linear activation functions are typically employed such that the resulting ANN is capable of mapping complex, non-linear relationships existing in the data on which it is trained. By shifting the activation function the bias term effectively dictates the range of input values over which the activation function is stimulated.

2.2.2 Activation functions

Some of the main activation functions are listed in Table 2.1. The first activation function proposed in 1943 was that of the threshold logic unit [125], which is similar to the heaviside step function. It closely mimics the behaviour of biological neurons activating once the summation of the weighted inputs is above a certain threshold. It was proven to be capable of modelling linearly separable functions within a limited number of iterations, which was of importance at the time due to the lack of computational resources [129].

However, it fell out of favour as its gradient, being undefined at $z = 0$ and zero otherwise, prohibited effective training of deeper ANNs. In particular, the analytical differentiability of activation functions, being neither zero nor undefined, is crucial to the ability of the backpropagation algorithm to train multilayer ANNs, as detailed in Section 2.2.5.

Consequently, non-linear, continuous activation functions have become favoured. Their non-linearity enables effective modelling of complex, non-linear relationships within the data. Additionally their continuous nature, indicating the degree of activation of the artificial neuron, allows for continuous gradients aiding training stability.

The sigmoid activation function was initially used to process information within hidden layers due to its ability to squash real valued inputs from $(-\infty, \infty)$ to $[0, 1]$. Furthermore, its continuous gradient offered improved training relative to the threshold logic unit and identity activation function. However, its gradient being close to zero outside its operating range of $-1 \leq z \leq 1$ meant poor training of deeper networks as the gradient would vanish during backpropagation. As such, it has fallen out of favour for use within hidden layers [130] and is almost exclusively utilised within output layers for binary classification tasks.

Table 2.1: Details of commonly employed activation functions.

Name	Function $\phi(z)$	Derivative $\phi'(z)$	Continuity
Binary	$\begin{cases} 0 & \text{if} \quad z < 0 \\ 1 & \text{if} \quad z \geq 0 \end{cases}$	$\begin{cases} 0 & \text{if} \quad z \neq 0 \\ ? & \text{if} \quad z = 0 \end{cases}$	C^{-1}
Identity	z	1	C^{∞}
Sigmoid	$\dfrac{1}{1+e^{-z}}$	$\phi(z)\,(1-\phi(z))$	C^{∞}
Hyperbolic tangent	$\dfrac{e^{z}-e^{-z}}{e^{z}+e^{-z}}$	$1-\phi(z)^2$	C^{∞}
ReLU	$\begin{cases} 0.01z & \text{if} \quad z \leq 0 \\ z & \text{if} \quad z > 0 \end{cases}$	$\begin{cases} 0 & \text{if} \quad z < 0 \\ 1 & \text{if} \quad z > 0 \\ ? & \text{if} \quad z = 0 \end{cases}$	C^{0}
Leaky ReLU	$\begin{cases} 0 & \text{if} \quad z \leq 0 \\ z & \text{if} \quad z > 0 \end{cases}$	$\begin{cases} 0.01 & \text{if} \quad z < 0 \\ 1 & \text{if} \quad z > 0 \\ ? & \text{if} \quad z = 0 \end{cases}$	C^{0}

Despite the hyperbolic tangent function suffering from this vanishing gradient problem, it remains a viable activation function for hidden layers due to its zero-centring characteristic. That is, it brings the mean of its outputs closer to zero essentially centring the data it passes on to subsequent layers of neurons. This mitigates undesirable zig-zag dynamics of gradient descent optimisation algorithms utilised during training.

The rectified linear unit (ReLU) is more computationally efficient to compute, leading to faster evaluation of the ANN as well as reduced training times; Krizhevsky noted that using ReLU activation functions offered a speed up in training of up to 6 times [65]. Additionally, by returning zero for $z \leq 0$ neurons not stimulated by the input data will not be activated leading to sparse networks.

Furthermore, ReLUs can lead to more robust ANNs. As noted by Bengio [131], a core purpose of ANNs is to disentangle the variables responsible for the patterns emerging in the data analysed. Dense representations are extremely interconnected such that small perturbations in the input can lead to large changes in the output. In contrast, a sparse representation which is reliable (possible through the use of ReLUs) more robustly represents the desired pattern since it is less susceptible to small perturbations in the input.

However, the property of returning zero for $z \leq 0$ comes with a caveat as improper training parameters can cause neurons to operate exclusively in this region. Such neurons become permanently inactive; not firing regardless of the input. The leaky ReLU [132] was created to mitigate this by introducing a shallow slope for $z \leq 0$.

The softmax function, given in Table 2.2 is a unique activation function used in the output layer for classification tasks. It normalises the output signals of the previous layer, for which each neuron corresponds to a class, to return distributed probabilities over these classes so that their summation is equal to one.

Table 2.2: Details of the softmax activation function.

Function $\phi(z)$	Derivative $\phi'(z)$	Continuity
$\frac{e^{z_j}}{\sum_{j=1}^{J} e^{z_j}}$	$\frac{\partial\phi(z_j)}{\partial z_j} = \begin{cases} \phi(z_j)(1-\phi(z_n)) & \text{if} \quad i = n \\ -\phi(z_j)\phi(z_n) & \text{if} \quad i \neq n \end{cases}$	C^{∞}

2.2.3 Convolution activation function

The convolution activation function is a unique activation function which operates directly on images. CNNs, which are a type of ANN which leverage the capabilities of convolution activation functions, are effective at extracting information from images or image series.

The convolution activation function is in the form of a matrix, of specific height and width, referred to as a kernel. The convolution operation applies a kernel to an input image which results in a feature map as illustrated in Figure 2.3. This involves applying the kernel to a region of the input image, called a window, of dimensions equivalent to that of the kernel. In doing so each element of the input window is multiplied by the corresponding element of the kernel and the summation of these values is stored in a corresponding element of the feature map.

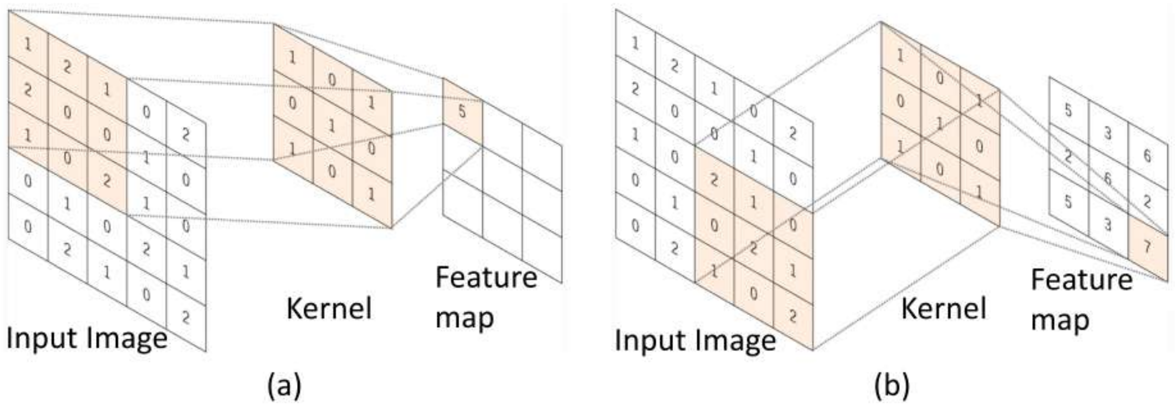


Figure 2.3: Schematic illustration of the convolution process of a CNN for the (a) first and (b) last element of the feature map (adapted from [133]).

The same operation is performed on a neighbouring window of the image. That is, the same kernel is applied to a new window of the input image which is located at a position that has been shifted relative to the previous window's position by the stride size of the convolution activation function. Repeating this process across all windows of the input image, defined by the stride, results in a feature map as shown in Figure 2.3(b). The feature map reflects the degree to which the regions of the image, corresponding to the windows considered, activate the kernel of the convolution function. The kernel values (elements of the matrix which make up the kernel) are the weights of the convolution function which are fine tuned during training to detect features relevant to the purpose of the CNN.

A single convolution layer can consist of multiple kernels (filters), the number of which is referred to as the depth of the layer. Each filter, having its own kernel matrix and associated weights, identifies separate features of the image. A convolution layer is typically followed by a ReLU or Leaky ReLU layer which modifies the elements of the feature map. CNNs can function as powerful image processing systems by linking pairs of convolution and ReLU layers. Each of these pairs will process the feature maps of the previous pair, detecting features within the feature maps of the preceding layer. Average and max pooling layers are typically used between convolution-ReLU pairs to reduce the dimension of the feature map to reduce the number of parameters to train.

The success of CNNs over the last decade has lead to an ever-growing pool of knowledge on their effective implementation and use. For more information on convolution activation functions, and CNNs in general, refer to the work of Albawi et al. [134] and Milosevic [135].

2.2.4 ANN architecture

The architecture of an ANN is determined by the structural hyperparameters which include: the number of layers, the number of neurons per layer, the interconnectivity between neurons and the activation functions of these neurons. These structural hyperparameters must be appropriately stipulated for the intended task since they directly affect the resulting ANN's accuracy, noise robustness and training stability.

Knowledge of the intended format of the input and output data streamlines structural hyperparameter selection of the input and output layers. The dimensionality of the input data directly dictates the number of neurons required for the input layer which typically employ the identity activation function such that the input data is passed to the first hidden layer without being manipulated. Similarly, the desired dimensionality and format of the output data dictates the activation functions and number of neurons of the output layer where:

- Binary classification ANNs sort inputs into two classes and make use of a sigmoid activation function and a single output neuron.
- Multi-class classification sorts input data into three or more classes by employing a softmax activation function with a neuron for each class.
- Regression ANNs attempt to model functions by predicting the function output based on the input values. They make use of linear activation function with a neuron per output variable.
- Multi-label classification tasks differ from multi-class classification in that classes are not mutually exclusive such that the ANN needs to determine which labels (potentially more than one) are relevant to the data instead of which class the data falls under. They employ the sigmoid activation function with a neuron for each label.

Appropriate architectural design of the hidden layers is more complex with theoretical recommendations contradicting empirical observations. In particular, the universal approximation theorem [136] has shown that an ANN containing a single hidden layer consisting of a finite number of neurons can approximate any continuous function. This theory is supported by the findings of Ba and Caruana [137], which empirically showed that single hidden layer ANNs can achieve similar performance to state-of-the-art deep ANNs for phoneme[2] and image recognition tasks.

Despite this, they noted that deeper networks are better suited to current state-of-the-art training algorithms. Additionally, deeper ANNs are capable of representing the same functions with fewer neurons, and thus fewer weights and biases, making them easier to train. This is significant since training ANNs for novel applications is generally challenging. By stacking layers of non-linear activation functions, deep ANNs are capable of modelling complex, non-linear relationships as explained in Section 2.2.6. This ability to approximate more complex functions is referred to as the flexibility of the ANN. As such, deeper networks, offering more flexibility, are generally favoured within the ANN community.

However, the appropriate depth remains a question without a definitive answer. It is generally accepted that the shallowest network which is capable of accurately mapping the inputs to the outputs of the training dataset, thereby modelling the function which they represent, is most desirable. This results in an ANN which has

---

[2] A phoneme, for example the "p" in "tap" and "b" in "tab", is the smallest "unit" of sound in a language that is capable of carrying a specific meaning.

the lowest number of weights and biases which is advantageous since this limits memory requirements, improves computational efficiency (evaluation of the ANN), leads to more stable training and aids generalisation ability (generalisability)[3] [138].

Interconnectivity between neurons of the hidden layer is an area of research that is still in its infancy. Until recently it has been common practice to employ fully connected layers throughout an ANN, only deviating from this where a priori knowledge of the problem suggests otherwise. However, in the last decade there has been a growing trend of employing skip connections to improve ANN training stability and generalisability [139]. Skip connections, introduced by He et al. [140] and Srivastava et al. [141], bypass one or more layers, connecting the output of a neuron to a neuron of a layer to which it is not adjacent. These skip connections offer additional channels for information transfer which are advantageous for backpropagation, alleviating the issue of vanishing gradients.

It is clear that appropriate design of the hidden layers is a complex task that lacks well-established rules or guidelines. As such, a number of approaches have been proposed to aid hidden layer structural hyperparameter selection. Benardos and Vosniakos [142] summarise these as: (i) pruning and constructive methods [143, 144] which remove and add, respectively, neurons from a preliminary starting architecture (neurons are added for poor accuracy or long training time and removed if they do not aid these aspects sufficiently); (ii) genetic algorithms which alter the number of hidden layers and neurons per hidden layer and assess the different architectures according to an objective function [145]; (iii) statistical and empirical methods [146, 147] which identify the impact of specific structural hyperparameters on the resulting ANN performance such that the best performing combination is retained; and (iv) the use of fuzzy logic[4] [149], in which an ANN is understood as an adaptive fuzzy system or is capable of working with fuzzy numbers rather than real ones.

Despite the availability of these, most deep learning practitioners make use of available research efforts, identifying a high performing ANN architecture which solves a problem similar to the one at hand. The input and output layers of this architecture are adapted to function with the training dataset available while the hidden layers are adjusted and refined through trial-and-error until the best performing network is obtained [138, 150].

2.2.5 Training

The values of the weights and biases of the hidden and output layers dictate the response of the neural network to its inputs. Training is the process by which these weights and biases are fine tuned to enable the ANN to accurately and reliably perform its intended task. Training of supervised ANNs is composed of: (i) initialisation of the weights of the ANN; (ii) forward propagation of the training data through the network to predict the output; (iii) computation of the loss representing the accuracy of the ANN in its current state; (iv) backpropagation to propagate the gradient of the loss function, with respect to the weights and biases, backward through the ANN;

---

[3] Generalisation ability of an ANN refers to how well it performs on unseen data (data it is not trained on).

[4] Fuzzy logic [148] is an alternative interpretation to Boolean logic (where something is either true (1) or false (0)) based on interpreting truth as partial; where the truth of a variable can be any real number between 0 and 1.

and (v) gradient descent optimisation to update the weights and biases according to their corresponding loss gradient. Points (ii)-(v) are iteratively repeated until convergence is achieved. These aspects are detailed before discussing nuances of the training process including mini-batches and early stopping.

2.2.5.1 Weight initialisation

The solution sought by the gradient descent optimisation algorithm corresponds to a local minima in the vicinity of the starting point defined by the initial weight values [151].Consequently, the initialisation of weights is crucial since it dictates how many iterations are required to locate this local minima affecting training stability and the quality of this local minima which directly translates to the ANN's performance.

Weights are typically initialisation in a random manner to "break symmetry" between neurons [152]. More specifically, neurons of the same layer typically make use of the same activation function and are connected to the same neurons of the preceding layer. If their initial weights are identical, their weights will be updated in the same way during training resulting in identical weights after training. As such, information passed through the trained network will be processed by these neurons in an identical manner which is redundant. In other words, the initial weights of neurons of the same layer need to be unique such that the features which these neurons seek are unique. Weights are sampled from a random distribution, with a mean of zero, to satisfy this.

The standard deviation of this random distribution is an important consideration having significant impact on the stability and speed of training. In particular, the standard deviation of the activation function's output signal must remain consistent across layers of the network to reduce the risk of vanishing or exploding gradients[5]. The standard deviation of the activation function's output is directly related to the standard deviation of the initial weights. Furthermore, the standard deviation of the output from the activation function increases with the number of inputs connected to the neuron.

As such, schemes have been developed for determining the standard deviation of the random distribution from which the initial weights are sampled. These schemes operate on a per layer basis, determining the values for a set of weights connecting the preceding layer, containing $N_{(l-1)}$ neurons, and the next layer, containing $N_{(l)}$ neurons, which receives the weighted signals. The formulae of these schemes are reported for both uniform and random distributions since there is no consensus within the ANN community as to which is preferred.

The Glorot [130], otherwise known as Xavier, scheme is appropriate for both sigmoid and hyperbolic tangent activation functions. The standard deviation of the uniform and normal distributions, represented as σ_u^{Glorot} and σ_n^{Glorot}, respectively, are given as

---

[5] Vanishing and exploding gradients refers to the phenomena whereby the gradient of the loss function becomes progressively smaller or larger, respectively, as it is backpropagated through the ANN. This causes the gradient descent algorithm to have negligible effect or diverge, respectively, for initial layers of the ANN.

$$\sigma_u^{Glorot} = \sqrt{\frac{6}{N_{(l-1)} + N_{(l)}}} \quad \text{and} \quad \sigma_n^{Glorot} = \sqrt{\frac{2}{N_{(l-1)} + N_{(l)}}} \tag{2.3}$$

The He scheme, relevant to the ReLU and leaky ReLU activation functions, have standard deviations of uniform (σ_u^{He}) and normal distributions (σ_n^{He}) of

$$\sigma_u^{He} = \sqrt{\frac{6}{N_{(l)}}} \quad \text{and} \quad \sigma_n^{He} = \sqrt{\frac{2}{N_{(l)}}} \tag{2.4}$$

2.2.5.2 Forward propagation

Forward propagation is the process by which input data is fed through the layers of the ANN which process this information and return a corresponding output. Once deployed, a trained ANN uses forward propagation to fulfil its function making predictions based on the inputs. During training it is used to determine how appropriate the current weights and biases of the ANN are for its intended function by comparing the returned output to the desired output.

Equations (2.1) and (2.2) perform the forward propagation of the signal of the previous layer through a single neuron of the subsequent layer. Performing this operation between all neurons of layers $(l-1)$ and (l), thereby propagating the signal forward from the $(l-1)^{\text{th}}$ to the l^{th} layer, is represented as

$$\begin{pmatrix} a_1^{(l)} \\ a_2^{(l)} \\ \vdots \\ a_j^{(l)} \end{pmatrix} = \phi\left(\left[\begin{pmatrix} w_{1,1}^{(l-1)} & w_{1,2}^{(l-1)} & \dots & w_{1,k}^{(l-1)} \\ w_{2,1}^{(l-1)} & w_{2,2}^{(l-1)} & \dots & w_{2,k}^{(l-1)} \\ \vdots & \vdots & \ddots & \vdots \\ w_{j,1}^{(l-1)} & w_{j,2}^{(l-1)} & \dots & w_{j,k}^{(l-1)} \end{pmatrix} \begin{pmatrix} a_1^{(l-1)} \\ a_2^{(l-1)} \\ \vdots \\ a_k^{(l-1)} \end{pmatrix} + \begin{pmatrix} b_1^{(l-1)} \\ b_2^{(l-1)} \\ \vdots \\ b_j^{(l-1)} \end{pmatrix}\right]\right) \tag{2.5}$$

This process is repeated between each neighbouring pair of layers, propagating the signal forward, until the output layer is reached at which point the ANN presents its prediction.

2.2.5.3 Loss computation

The loss function quantifies the discrepancy between the desired output and that returned from forward propagation to reflect how accurately the ANN models the relationship between the inputs and outputs. The aim of the training process is to minimise the loss function so as to improve the performance of the ANN. As such, the loss function employed needs to be appropriate for the design goal of the ANN such that improvement in the loss function faithfully represents improved performance of the ANN for its intended task.

ANNs performing regression typically employ the mean squared error loss function (MSE_{loss}) given as

$$MSE_{loss} = \frac{1}{N_L} \sum_{i=1}^{N_L} \left(\chi_i^{pred} - \chi_i^{true}\right)^2 \tag{2.6}$$

where there are N_L many output neurons, the i^{th} neuron returning a predicted output

χ_i^{pred} which is compared to the target (true) output of χ_i^{true}. The derivative of MSE_{loss} with respect to the output of the i^{th} neuron is given as

$$\frac{\partial MSE_{loss}}{\partial \chi_i^{pred}} = 2\left(\chi_i^{pred} - \chi_i^{true}\right) \tag{2.7}$$

For binary classification tasks, the binary cross-entropy loss function(CE_{loss}) is computed as

$$CE_{loss} = \frac{1}{N_L}\sum_{i=1}^{N_L} - \left(\chi_i^{pred} log(\chi_i^{true}) + \left(1 - \chi_i^{pred}\right) log(1 - \chi_i^{true})\right) \tag{2.8}$$

Its derivative with respect to output of neuron i is given as

$$\frac{\partial CE_{loss}}{\partial \chi_i^{pred}} = -\left(\frac{\chi_i^{pred}}{\chi_i^{true}} - \frac{1 - \chi_i^{pred}}{1 - \chi_i^{true}}\right) \tag{2.9}$$

The value computed by the loss function is referred to as the loss of the ANN.

2.2.5.4 Backpropagation

Backpropagation determines the portion of the loss that each neuron is responsible for. Neurons with a low associated loss detect features beneficial to the objective of the network (as defined by the loss function). The connections to these neurons are strengthened by increasing their weights. Conversely, neurons of high associated loss have their weights decreased to reduce the bias they impose on the computed outcome.

In particular, the backpropagation algorithm computes the gradient of the loss function (Y) with respect to each weight ($\frac{\partial Y}{\partial w_{jk}^{(l-1)}}$) and bias ($\frac{\partial Y}{\partial b_j^{(l-1)}}$) using the chain rule

$$\frac{\partial Y}{\partial w_{jk}^{(l-1)}} = \frac{\partial z_j^{(l)}}{\partial w_{jk}^{(l-1)}} \frac{\partial a_j^{(l)}}{\partial z_j^{(l)}} \frac{\partial Y}{\partial a_j^{(l)}} \tag{2.10}$$

$$\frac{\partial Y}{\partial b_j^{(l-1)}} = \frac{\partial z_j^{(l)}}{\partial b_j^{(l-1)}} \frac{\partial a_j^{(l)}}{\partial z_j^{(l)}} \frac{\partial Y}{\partial a_j^{(l)}} \tag{2.11}$$

Here $\frac{\partial Y}{\partial a_j^{(l)}}$ is the derivative of the loss function with respect to the activation function of the j^{th} neuron of layer l. Starting with the last layer, since the loss is to be propagated backwards through the network, l is set to equal the number of layers of the ANN as $l = L$. As such, $\frac{\partial Y}{\partial a_j^{(l)}}$ is the derivative of the loss function with respect to the activation function of the output layer which is given in Equations (2.7) and (2.9) for the mean squared error and binary cross-entropy loss functions, respectively. The derivative of the activation function of the j^{th} neuron of the l^{th} layer with respect to its input signal ($\frac{\partial a_j^{(l)}}{\partial z_j^{(l)}}$) is given as

$$\frac{\partial a_j^{(l)}}{\partial z_j^{(l)}} = \phi'(z_j^{(l)}) \tag{2.12}$$

where ϕ' is the derivative of the activation function used for layer l which are listed in Table 2.1 for various possible activation functions. The derivative of the input signal with respect to the weight, $\frac{\partial z_j^{(l)}}{\partial w_{jk}^{(l-1)}}$, is computed from Equation (2.1) as

$$\begin{aligned}\frac{\partial z_j^{(l)}}{\partial w_{jk}^{(l-1)}} &= \frac{\partial}{\partial w_{jk}^{(l-1)}}\left(\sum_{k=1}^{N_{(l)}} w_{jk}^{(l-1)} a_k^{(l-1)} + b_j^{(l-1)}\right) \\ &= a_k^{(l-1)}\end{aligned} \tag{2.13}$$

Similarly, the derivative of the input signal with respect to the bias, $\frac{\partial z_j^{(l)}}{\partial b_j^{(l-1)}}$, is

$$\begin{aligned}\frac{\partial z_j^{(l)}}{\partial b_j^{(l-1)}} &= \frac{\partial}{\partial b_j^{(l-1)}}\left(\sum_{k=1}^{N_{(l)}} w_{jk}^{(l-1)} a_k^{(l-1)} + b_j^{(l-1)}\right) \\ &= 1\end{aligned} \tag{2.14}$$

At this point the gradient of the loss function, also referred to as the loss gradient, has been propagated backwards to the weights and biases connecting the output layer and last hidden layer. To determine the derivatives of the weights and biases of the preceding layer the derivative of the loss function with respect to the output of each neuron of layer $l-1$ needs to be computed as

$$\frac{\partial Y}{\partial a_k^{(l-1)}} = \sum_{j=1}^{N_{(l)}} \frac{\partial z_j^{(l)}}{\partial a_k^{(l-1)}} \frac{\partial a_j^{(l-1)}}{\partial z_j^{(l)}} \frac{\partial Y}{\partial a_j^{(l)}} \tag{2.15}$$

$$\begin{aligned}\text{where} \quad \frac{\partial z_j^{(l)}}{\partial a_k^{(l-1)}} &= \frac{\partial}{\partial a_k^{(l-1)}}\left(\sum_{k=1}^{N_{(l)}} w_{jk}^{(l-1)} a_k^{(l-1)} + b_j^{(l-1)}\right) \\ &= w_{jk}^{(l-1)}\end{aligned} \tag{2.16}$$

Thereafter the derivative of the loss function with respect to the weights and biases of between the penultimate and final hidden layers can be computed using Equations (2.10-2.11) by setting $l = L-1$. This process is repeated to progressively build up a library of loss gradients until the derivative of the loss function with respect to each trainable parameter is known.

2.2.5.5 Gradient descent

The loss gradients of Equations (2.10-2.11) are used within a gradient descent algorithm to minimise the loss function by iteratively updating the weights and biases. Let g_t represent the derivative of the loss function with respect to the parameter under consideration for the current iteration. For basic gradient descent algorithm the updated parameter value (θ_{t+1}) is obtained from the current value (θ_t) as

$$\theta_{t+1} = \theta_t - \varrho g_t \tag{2.17}$$

where ϱ is the learning rate. The learning rate dictates the portion of the loss gradient which is applied when updating the weights and biases thereby dictating the speed at which the ANN learns. Typically the learning rate is below unity since if the learning rate is too large the ANN will rapidly reach convergence with sub-optimal biases and weights. Conversely, small learning rates increase the time to convergence and can lead to poor performance of the ANN as the training path becomes stuck in sub-optimal local minima. Thus, the learning rate used directly affects not only the training time but the resulting performance of the trained ANN.

Equation (2.17) has poor performance for non-convex loss functions typical for ANNs. It tends to oscillate in regions of the loss surface containing ravines (where the curve of the loss surface is more pronounced in certain dimensions relative to others [153]). Furthermore, it is susceptible to becoming trapped within sup-optimal local minima due to its difficulty in dealing with saddle points [154, 155].

Gradient descent with momentum uses a portion of the loss gradient of the previous iteration (g_{t-1}) to improve convergence as

$$\theta_{t+1} = \theta_t - \varrho g_t - \gamma \varrho g_{t-1} \tag{2.18}$$

where γ is the momentum term dictating the portion of the previous loss gradient to be utilised. This has the effect of increasing the magnitude of the update applied to parameters having a similar loss gradient direction for the current and previous iterations. Conversely, it reduces the magnitude of the update for parameters having loss gradients of opposing directions between iterations. This has the effect of dampening the oscillations occurring between iterations leading to an improved convergence rate [156].

Despite improving upon the standard gradient descent algorithm, the training performance is still dependent on the learning rate and momentum term which are challenging to select appropriately. Additionally, these methods do not adapt the learning rate on a per-parameter basis which is essential for sparse data[6].

Adaptive moment estimation (Adam) [157] offers such functionality while offering high computational efficiency and low memory requirements. It combines the concepts of gradient descent with momentum and root mean squared propagation to control the rate of gradient descent such that it bypasses local minima while limiting oscillations within the region of the global minima. Subsequently it outperforms root mean squared propagation in the final phase of optimisation as loss gradients become sparse [158].

Firstly, it approximates the first (m_t) and second moments (v_t) of the loss gradient (representing the mean and variance) as

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{2.19}$$

$$v_t = \beta_t v_{t-1} + (1 - \beta_2) g_t^2 \tag{2.20}$$

---

[6] Sparse data occurs when an input or output variable is zero for many of the data points of a dataset. As such, it has low information content.

where β_1 and β_2 are training hyperparameters which dictate the portion of the previous loss gradient which is used alongside the current loss gradient to update the weights and biases. These terms are biased towards zero, particularly for initial iterations, since they are initialised as $m_0 = 0$ and $v_0 = 0$. This is corrected for by computing the bias-corrected first ($\hat{m}_t$) and second moment ($\hat{v}_t$) as

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \tag{2.21}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \tag{2.22}$$

The updated parameter is obtained as

$$\theta_{t+1} = \theta_t - \frac{\varrho}{\sqrt{\hat{v}_t} + \varepsilon} \hat{m}_t \tag{2.23}$$

where ε is a smoothing term to avoid division by zero. The self-adaptive nature of this algorithm means that training performance is less susceptible to the employed learning rates with the authors recommending default values of $\varrho = 0.002$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\varepsilon = 10^{-8}$. Both the type of gradient descent algorithm and associated parameters are referred to as training hyperparameters.

2.2.5.6 Mini-batches

The true loss gradient at the current state of the model is defined by the loss of the ANN when analysing the full training dataset (referred to as the batch). However, processing the full training dataset for each iteration of training is prohibitively slow since training datasets are generally large.

ANNs typically require large training datasets to be appropriately trained such that the cardinality of the training dataset is sufficient to train the large number of weights and biases within the ANN. Furthermore, the training dataset needs to contain a wide range of variations such that the neurons are fine tuned to detect features prevalent to the underlying patterns within the data, and not identify features specific to the training dataset. This ensures good generalisability such that the ANN performs well beyond the scope of the training dataset.

As such, each iteration of training operates on a mini-batch, which is a randomly selected subset of training dataset, to improve computational efficiency. The loss gradient computed for a mini-batch gives a slightly noisy approximation of the true value for the whole batch. This alleviates the tendency of ANNs to over-fit to the training data leading to improved training stability and generalisation performance [159].

2.2.5.7 Early stopping

The definition of convergence is not necessarily when the loss function of the training dataset is minimised [160]. Supervised ANNs are typically intended to be deployed for real world applications where they are expected to maintain their level of performance when analysing unseen data. Thus, high robustness and generalisability are generally more desirable than the absolute lowest error for the training dataset.

Two subsets of the original dataset are reserved to measure generalisability: the validation dataset which is used to reflect the generalisability of the ANN during training and the testing dataset used to quantify the performance of the final trained ANN. During training the generalisability of the ANN improves as it learns the underlying relationship it is intended to model. However, as training continues, the network begins to learn the idiosyncrasies of the training dataset reducing it generalisability. This is a consequence of the difficulty in balancing the flexibility of the ANN, by virtue of its depth and number of neurons, with the complexity of the relationship it is attempting to model.

As such, training is halted once the performance of the validation dataset begins to decrease. This is referred to as early stopping and prevents the neurons from diverging from detecting features relevant to the underlying relationship, to features unique to the training dataset.

2.2.6 Feature extraction

Artificial intelligence research efforts have begun transitioning from employing predominantly machine learning based algorithms to taking advantage of the capability of ANNs to build multilevel representations of the input data. Machine learning algorithms, typically sensitive to the form of the input data presented to them, are incapable of operating directly on raw data. They rely on the process of feature engineering to manipulate data, preparing it for analysis, to improve the performance of the associated machine learning algorithm. Feature engineering requires expert knowledge of the problem domain and is typically time consuming as the approach needs to be revised, tested and refined in an iterative manner such that the resulting operations performed are beneficial to the task at hand [161].

In contrast, the ability of ANNs to perform data representation [162], with each layer building progressively higher-level abstractions of the input data, enables them to operate directly on raw data. This is most intuitively illustrated using a CNN designed to detect faces within an image. The features extracted by the neurons of the three hidden layers are illustrated in Figure 2.4.

The first hidden layer identifies low level features such as spatial frequencies, lines and edges. The second hidden layer detects combinations of these to identify more complex shapes and patterns such as eyes, noses, mouths and ears. Finally, the last hidden layer aggregates these features to identify faces within the image. As such, an image containing a face will stimulate a large portion of the neurons of the final hidden layer such that their activation functions output a strong signal. Upon receiving these strong signals, the output layer interprets these as confirmation of the presence of a face within the image.

Therefore, artificial neurons operate, in a sense, as feature extractors identifying features within the data. By building progressively higher level representations of the data, though detecting progressively more complex features, ANNs break down the data into the elements that are most relevant to the task at hand. This allows them to perform with high accuracy on raw input data making them attractive for robust and rapid deployment.

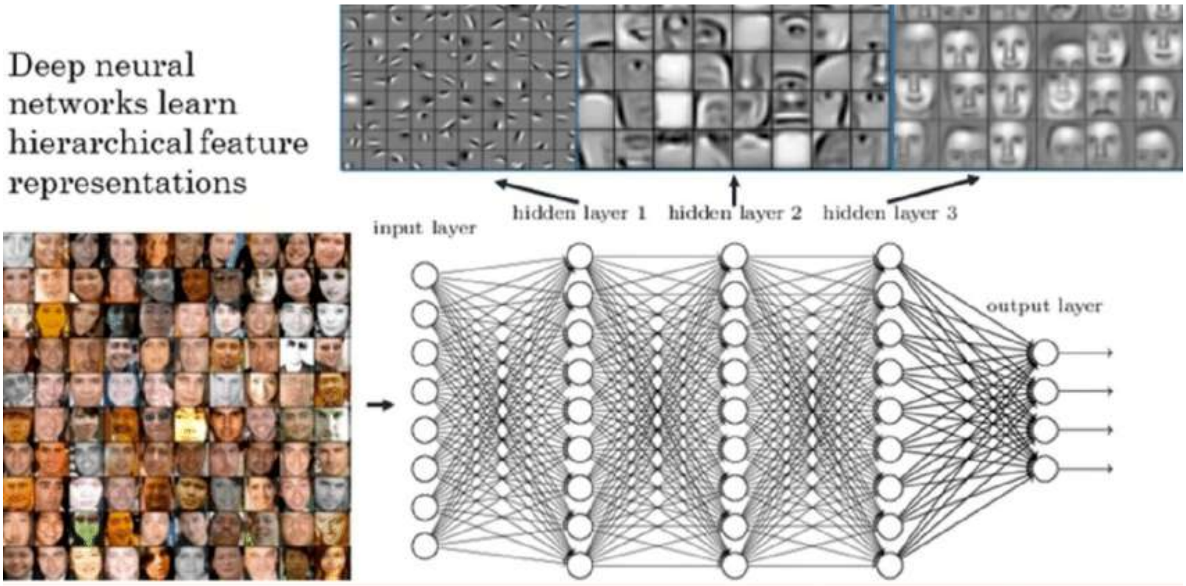


Figure 2.4: Illustration of the progressively more complex features extracted from images by a CNN [163].

The final feature extractors (features that are detected by the artificial neurons) of a trained ANN are a consequence of the features existing within the training dataset which are beneficial to the purpose of the ANN; where the loss function dictates which features are beneficial by measuring their contribution to the accuracy of the ANN. This is of significance since the features are automatically learned without the need for human intervention and without making structural assumptions about the data. This removes the need for feature engineering which is generally a time consuming and labour intensive task. This greatly speeds up the development of algorithms for specific artificial intelligence applications provided that the training dataset is adequately representative of the applicable features.

Due to the ability of ANNs to learn progressively deeper level features from input data, the process of using ANNs of three or more layers [164] is often referred to as *deep learning*. Although all ANNs used in this work fall under the category of deep learning, the term ANN is retained for clarity.

Chapter 3
Hypothesis development

This chapter documents the hypothesis of the dissertation in light of the concepts introduced in Chapter 1 and summarised in Chapter 2. It motivates the aim of the project as well as the approach taken to investigate the validity of the hypothesis. The objectives, which give rise to the hypothesis, are defined in conjunction with their intended aims as well as the tasks that they encompass. Finally, the scope and limitations of the dissertation are established.

3.1 Motivation for development of open DIC framework

Upon conducting an extensive review of the DIC technique and current state-of-the-art practices, a number of observations were made. Most notably, for the purpose of furthering the capabilities of DIC, researchers have typically begun by developing an in-house DIC algorithm from the ground up offering the capabilities required for implementation and evaluation of their proposed contribution. This is a tradition that extends back to the first application of DIC to experimental solid mechanics [30].

This approach is not without merit since in the past the majority of improvements to the technique came as a result of directly modifying and improving aspects of the correlation algorithm [9, 10, 165, 11, 15, 14]. However, the DIC technique has been refined to such a degree that the greatest contribution to error reduction in most practical applications comes as a result of factors beyond improving the core correlation algorithm itself.

Furthermore, implementation of an in-house DIC algorithm is cumbersome, especially for newcomers to the field of DIC, due to the gap in literature between the well-documented theory of DIC and its nuanced practical implementation as code. In particular, although open-source DIC algorithms with an associated documentation of their theory exist (such as Ncorr [166]), their design philosophy catering towards robustness, efficiency and user-friendliness causes the resulting code to be too complicated to serve as an educational resource.

The lack of an alternative to the traditional approach of in-house code development coupled with the gap in literature hinders the progress of DIC. More specifically, they act as barriers to newcomers to the field of DIC, intending to further its capabilities, thereby limiting the rate of development of the DIC field; particularly in terms of its adaptation to novel applications.

Despite the availability of open-source DIC algorithms (such as Ncorr [166], Digital Image Correlation engine (DICe) [167, 168], YaDIC [169] and pydic [170]), none offer the required degree of control over setting up the correlation problem. Although modifying an existing open-source code such as DICe to enable this was considered, it was decided to build an image correlation framework from the ground up, as is traditional. There are several reasons for this:

Firstly, it is desired that the code employed constitutes a straightforward implementation of image correlation free from corrective measures operating in the background; often employed by available DIC implementations to improve robustness of the process. Although such methods are highly desirable for practical applications of DIC, their presence during evaluation of the proposed ANN-based framework for DSS could obscure the potential and limitations investigated.

Secondly, although not a direct motivation in the context of the project, this dissertation is conducted under the Material Engineering research group at Stellenbosch University and its collaborators (including the Centre for Materials Engineering at the University of Cape Town) that would benefit from the creation of an in-house DIC framework. In particular, this research group has been involved in a number of projects with the Eskom Power Plant Engineering Institute including the work of van Rooyen [171], Molteno [172] and Huchzermeyer [173]. Through the use of DIC, these projects developed frameworks aiding condition monitoring in thermal power plants for the purpose of predictive maintenance.

Lastly, there is a gap in the open-source DIC landscape for a modular, versatile, straightforward and well-documented DIC code which offers full control[7] over setting up the correlation problem.

Through reviewing the current landscape of DIC research, the ongoing and future research directions (beyond the context of improving the core correlation algorithm) were noted and the required capabilities of a DIC framework viable for their implementation and validation were identified as:

- Dynamic appointment of correlation parameters on a per-subset and per-image basis for appropriate setup of the correlation problem.
- Stipulation of non-standard, custom subset sizes on a per-subset and per-image basis to account for displacement discontinuities though subset splitting [174].
- Ability to stipulate custom SFPs better suited to the deformation field such that random errors resulting from unnecessary degrees of freedom (irrelevant SFPs) can be reduced.
- Modular in that it allows for straightforward interchangeability of aspects of the DIC process such as: (i) simpler or more advanced interpolation schemes to trade higher accuracy for computation speed or vice versa; (ii) alternative methods for computation of initial displacement estimates; (iii) various correlation criteria; and (iv) different displacement transformation algorithms.
- Predominantly consistent with current state-of-the-art DIC practices ensuring performance on par with established algorithms.
- Straightforward implementation, which is well-documented, drawing a direct link between the theory of DIC and its implementation as code such that it is intuitive to understand, easy to adapt and is attractive as an educational resource.

Such a DIC framework would remove one of the barriers to dynamic correlation parameter assignment expediting research in this area. Additionally, its focus on

---

[7] In this context full control refers to the ability to stipulate the subset size, subset shape and SF on a per-subset and per-image basis.

modularity would facilitate rapid adaptation to use cases not directly catered for in its current implementation. Finally, due to its extensive documentation it would serve as an educational resource bridging the gap between the theory of DIC and its practical implementation. Moreover, it would not only offer an attractive alternative to the traditional approach of in-house code development, but additionally serve as an educational resource expediting the process of in-house DIC code development for applications that require such an approach.

Only the correlation aspect of a DIC framework is required to evaluate the applicability of ANNs for DSS, since correlation errors occur within the image domain. However, for the purpose of resolving the gaps in literature hindering the development rate of DIC, this project includes the implementation of displacement transformation and stereo matching codes. This extends the developed image correlation framework to 2D and stereo DIC such that it constitutes a full DIC framework appropriate for real world use. In particular, 2D DIC is simple and thus attractive to investigate approaches for improving the DIC method while stereo DIC is preferred for practical applications due to its ability to account for out-of-plane motion offering improved robustness. Thus, the 2D implementation lends itself to rapid development of novel approaches which, after their validation, are straightforward to extend to the stereo DIC version of the framework such that these new methods are readily available for practical use.

3.2 Motivation of ANNs for correlation parameter assignment

There are three reasons as to why the first barrier to dynamic correlation parameters assignment has remained unsolved. Firstly, the relationship between the subset size, image set properties, and resulting displacement error is complex and challenging to model. In particular, analytical formulas have been derived, based on the SF order, which relate the standard deviation of image noise and SSSIG of the subset to the resulting random errors in the computed displacements [61]. However, the derivation of these relations is for a simplistic sum of squared difference correlation criterion and relies on assumptions to reduce the mathematical complexity of the derivation. As such, they are not directly applicable to the majority of DIC instances currently in use.

ANNs have been shown to be capable of dealing with noisy, incomplete data to model the relationship between causes and effects for the purpose of estimation and prediction [175]. In particular, ANNs have been shown to be well suited to modelling complex relationships including: (i) diagnosing heart disease and prescribing the appropriate medication on par with that prescribed by a heart specialist [176]; (ii) outperforming traditional regression models for wind turbine fault detection offering improved remaining operational life prediction [23]; and (iii) improved estimation of global solar radiation relative to established regression models [177], to name a few. As such, ANNs are well suited to model complex relationships making them attractive as an approach to model the relationship between subset size, image set properties and resulting displacement error.

Secondly, since several SPQMs have been empirically shown to be related to the displacement error, no single SPQM quantifies and encapsulates all the characteristics that constitute a favourable speckle pattern. For example SSSIG, being the most widely accepted SPQM, does not take into account the randomness of the speckle pattern. Furthermore, as identified by Su et al. [178] SSSIG, being based on intensity gradient, favours fine speckles causing substantial image gradients which result in high systematic errors. ANNs offer an attractive solution to this being capable of taking in, as input, multiple SPQMs and discovering the relationship between these and the resulting displacement error.

Lastly, the lack of a method capable of spatially discerning the complexity of the underlying displacement field between an image pair independently of the displacements computed by DIC. Under-matched SFs can lead to significant systematic displacement errors while over-matched SFs increase random displacement errors. Thus, assignment of an appropriate SF order for each subset requires knowledge of the displacement field complexity in the vicinity of the query point.

The success of CNNs in the field of computer vision and DIC indicate that they are capable of assisting in this regard. In particular, recent research efforts focusing on ANN-based DIC have shown that CNNs are capable of measuring pixel-wise displacements and strains (related to the first-order displacement gradients) experienced between image pairs. Furthermore, Ma et al. [96] has shown that CNNs are capable of predicting initial estimates of the SFPs, which directly measure localised deformation, from image pairs. This suggest that CNNs are capable of measuring region-specific second-order displacement gradients directly from image pairs. Such a CNN would enable using the theoretical relations of Xu et al. [116] to predict the systematic error for a first-order SF in the presence of a second-order (quadratic) displacement field based on the subset size. This information would facilitate appropriate assignment of SF order, between first and second-order, on a per-subset and per-image basis.

Although this project investigates the viability of ANNs for both DSS and DSFS, DSS is argued to be of greater importance, and is thus focused on. More specifically, appropriate selection of SF order is guided by *a priori* knowledge of the experiment, with the specimen geometry and load applied dictating the presence of complex deformations. Stipulating a global second-order SF at most doubles the random error while mitigating the risk of exploding systematic errors. In contrast, appropriate appointment of subset size, even on a global level, is challenging due to the unknown range of subset sizes viable for an analysis due to speckle pattern quality being an abstract concept. Furthermore, the second-order SF cannot account for highly complex deformations such that the displacement error, due to both noise suppression and spatial resolution, predominantly governed by the subset size.

3.3 Aims and objectives

This project consists of three objectives. In particular, the first and second objectives align with the initial focus, seeking to develop a modular, versatile and accessible open DIC framework affording full control over setting up the correlation problem. This DIC framework serves as the foundation to develop and evaluate the ANN-based method, of the third objective, used to investigate the potential of ANNs as an approach for DSS. The objectives are defined as follows.

Objective 1: *Development of a modular, versatile and accessible open-framework for 2D*

DIC allowing independent assignment of correlation parameters on a per-subset and per-image basis. This objective, achieved in Chapter 4, aims to remove one of the barriers to dynamic correlation parameter assignment. Specific tasks include: implementation of the inverse compositional Gauss-Newton (IC-GN) optimisation algorithm to minimise the zero-mean normalised sum of squared difference (ZNSSD) correlation criterion making use of bi-cubic b-spline interpolation; modifying this implementation such that it is modular (allowing interchange and adaptation of the subprocesses of image correlation) and versatile (capable of assigning the SF order and subset size, of arbitrary shape, on a per-subset and per-image basis); implementation of 2D displacement transformation according to the pinhole camera model; documenting its implementation (as a MATLAB code) drawing direct parallels between the theory and resulting code (ensuring high accessibility of the framework); and validation of the framework to prove it performs on par with the image correlation aspect of established commercial and open-source DIC algorithms specifically in terms of noise suppression and spatial resolution.

Objective 2: *Extension of the image correlation framework of Objective 1 to stereo DIC.* This objective, achieved in Chapter 5, completes the development of an open-source framework for 2D and stereo DIC which is attractive as a starting point to develop the practical capabilities of DIC. Tasks of this objective include: extension of the image correlation framework to perform stereo matching; implementation of polynomial and linear triangulation to convert the two sets of 2D pixel displacements computed by image correlation to the 3D real world displacement of the corresponding point on the specimen; documenting these implementations in the form of MATLAB code; and validation of the resulting stereo DIC framework against established stereo DIC algorithms.

Objective 3: *Creation of an ANN to predict the random error in DIC computed displacement from purely image information and development of an ANN based DSS method.* This aims to investigate the potential of ANNs for temporal variance error prediction and DSS based on purely image information. More specifically, this investigates the ability of ANNs to model the complex interaction between subset size (and the contained speckle pattern quality), image noise, and resulting displacement error as well as the applicability of this model for DSS. This is considered in Chapter 6. Specific tasks of this objective include: development of a feed-forward ANN architecture which serves as a regression model to predict the random error of DIC computed displacements from image noise and a set of SPQMs; creation of a dataset to train and validate the ANN; implementation of this ANN in the form of a framework which appoints the smallest subset size offering random error consistent with a threshold stipulated by the operator; and validation of the proposed framework on speckle pattern characteristics consistent with and outside the scope of the training dataset to reveal the potential and limitations of the approach, respectively.

In addition, while not being a direct objective, *working towards investigating the viability of a CNN for quantifying the complexity of the underlying displacement field between image pairs for the purpose of DSFS* is also pursued. This is addressed in Section 7.4 which investigates the feasibility of CNNs to characterise the displacement field complexity in the form of second-order displacement gradients, independently of DIC computed displacements, for the purpose of assigning appropriate SF orders on a per-subset and per-image basis. The tasks involved include: development of a CNN to measure the second-order x-displacement gradient with respect to the x-direction ($\frac{\partial^2 u}{\partial x^2}$) directly from image pairs; creation of a database of varying displacement complexity (in the x-direction) and speckle pattern quality to train and validate the CNN; and assessing the viability of such an approach for DSFS relative to the traditional approach of global SF order assignment.

3.4 Central hypothesis

The traditional approach of utilising a global subset size and SF order for a DIC analysis is becoming outdated. In experimental solid mechanics the growing interest of new materials capable of complex deformation states [93] are pushing this traditional approach to its limit such that the setup of the correlation problem is typically only optimal for a portion of the region of interest (ROI) and image set. Furthermore, novel applications of DIC which analyse displacement fields dissimilar from those typical of experimental solid mechanics or rely on natural surface texture to constitute the speckle pattern (such as geotechnical engineering applications) break away from the assumptions applicable to experimental mechanics that make this traditional approach viable. To this end, the central hypothesis of the doctoral project is:

> *ANNs are a viable approach for DSS by modelling the complex relationship between subset size, image set properties and resulting random displacement error.*

3.5 Project scope and limitations

Rapid adoption of DIC, and subsequent research efforts, have expanded the technique to include several fundamentally different implementations and a wide range of use cases. This section serves to present the context of the project establishing its scope and limitations.

3.5.1 Type of DIC

Various types of DIC have been introduced over the years, such as global DIC [179], augmented Lagrangian DIC [180] and recently developed ANN-based DIC [93], which offer various advantages and disadvantages. This work focuses on local, subset-based DIC utilising the IC-GN gradient descent algorithm to solve the objective function defined by the ZNSSD correlation criterion.

Furthermore, the DIC framework employed: (i) is limited to grey-scale images (RGB images are not considered in this work [181]); (ii) does not employ the reliability-guided displacement tracking (RGDT) strategy [182]; (iii) is limited to two cameras for stereo DIC; (iv) does not implement subset splitting techniques [174]; and (v) does not use weighting of pixels of the subset to effectively reduce subset size [183].

3.5.2 DIC application

Although DIC is rapidly becoming a fundamental tool in a wide range of applications, this work focuses on the use of DIC in experimental solid mechanics applications, where most of the development of the technique has occurred. This does not prohibit the outcomes of this work from being applicable to the use of DIC in other fields, but establishes that observations and claims made in this work are directed towards the application of DIC in experimental solid mechanics, unless explicitly stated otherwise.

3.5.3 Displacement errors

The DIC error chain involves many error sources as outlined in Section 2.1. However, this work focuses solely on the errors of the correlation process resulting from the correlation problem setup; that is, errors due to the appropriateness of the SF order and subset size for the image set within the *mismatch error regime* and *ultimate error regime* as defined by Bornert et al. [184]. No effort is made to improve the image correlation process outside of increasing the flexibility and modularity of its implementation in a form consistent with established theory. Error sources not focused on in this work include, but are not limited to, those introduced during experimental setup, delamination of speckle pattern from the specimen's surface, discontinuities in the displacement field, distortions due to heat waves, distortions due to the lens system, image blur, spurious displacements due to vibrations, errors due to camera motion, and varying illumination during the experiment.

3.5.4 Synthetic and experimental image sets

Although practical applications of DIC typically involve the processing of experimental image sets, their usage to quantitatively assess the accuracy and precision of a DIC analysis is ill-advised. This is because it is challenging to design experiments for which the true underlying displacement field is known exactly. Furthermore, experimental image sets contain multiple error sources which are challenging to deconvolve [185] making it difficult to investigate the degree of change in a specific error source.

There is a trend in the DIC community to employ synthetic image sets for the validation and investigation of the proposed methods. Synthetic image sets mimic those captured of an experiment, but since they are generated by an algorithm, the true underlying displacement field is known exactly. Furthermore, the generation of synthetic image sets provides control over the error sources introduced in the DIC measurement chain. In particular, synthetic image sets can avoid introducing errors due to: variation in illumination, camera induced noise, unintended motion of the camera, delamination of the speckle pattern from the specimen surface and displacement transformation or calibration (since accuracy and precision are generally computed in pixel coordinates for synthetic image sets). Consequently, synthetic image sets are used predominantly in this work to evaluate and validate the various frameworks developed.

3.5.5 Speckle patterns

Speckle patterns are fundamental to the DIC process as their quality, density and characteristics dictate the accuracy with which displacement can be measured (as detailed in Section 2.1.2). Speckle patterns are defined as natural or applied based on whether the natural texture of the specimen's surface or paint applied to the specimen, respectively, constitutes the speckle pattern in the captured images [186].

Natural speckle patterns can have a wide array of characteristics and associated challenges. For example, in landslide monitoring vegetation, foliage, roads, shadows and landmarks create the contrast that is to be tracked [187]. However, excessive wind, changing position of the sun and other environmental factors can cause the speckle pattern to morph making accurate and precise displacement measurement challenging (as observed in analysis of the Gschliefgraben landslide [188] were localised unrealistic displacements resulted due to such effects). In this work, only data provided by the iDICs, or artificial speckle patterns that mimic the application of sprayed paint are considered. This avoids the introduction of additional error sources that could skew the potential and limitations of the investigated ANN-based approach.

Additionally, this work does not aim to provide recommendations for appropriate application of artificial speckle patterns such as the work of Dong and Pan [189] and Lionello and Cristofolini [190]. Rather, this work assumes that the speckle pattern is a fixed property of the image set for which the correlation parameters need to be adjusted for, to realise high quality displacement measurement.

3.5.6 Strain

The ability to compute the strain field experienced by a specimen from the displacement field measured via DIC is a significant reason for the rapid and widespread adoption of the DIC technique; particularly in the field of experimental solid mechanics. Although appropriate selection of the strain window size for the computation of strains from the displacement field has been identified as a key problem in the field of DIC [22], in this work, strain computation is considered a post-DIC process. As such, no attempt is made to improve the method of strain computation and no method of strain computation is implemented within the presented DIC framework.

However, it should be noted that strain is computed by differentiating the displacement field; a process that amplifies noise. As such, improving the accuracy and precision to which DIC is capable of measuring displacements (as is sought in this work) will subsequently lead to improved accuracy and precision of the computed strains.

3.5.7 ANNs

This work does not put an effort into the development of the field of ANNs but instead uses the insights and knowledge of this field to investigate its potential application in the field of DIC. As such, this project does not involve the creation of novel ANN layers or architectures, but instead employs well-established concepts of ANNs.

Chapter 4

2D DIC in MATLAB

This chapter addresses Objective 1, by developing a modular 2D DIC framework that is predominantly consistent with current state-of-the-art practices while offering full control over setting up the correlation problem. The resulting 117 line MATLAB code[8] is well-documented drawing direct links between the mathematical theory and practical implementation. The image correlation aspect of the framework is validated on image sets provided by the iDICs 2D DIC Challenge (version 1) showing that it performs on par with well-established algorithms in terms of dealing with noise, contrast changes and spatial resolution demands.

This chapter is published in the *Multidisciplinary Digital Publishing Institute (MDPI) Remote Sensing* journal under the title *A 117 Line 2D Digital Image Correlation Code Written in MATLAB* [191][9] and is presented here in its pre-review form (prior to input from the reviewers). Considerations arising during the review process are discussed in the post-submission discussion section which concludes this chapter.

Declaration by candidate:

With regards to Chapter 4, the contributions of authors and co-authors are as follows:

Name	Email	Contribution	Extent (%)
D Atkinson (candidate)	17732913 @sun.ac.za	Conceptualisation, framework development and implementation, DIC processing, analysis of DIC results, figure preparation and visualisation, writing and compilation of manuscript	90
TH Becker	thorsten.becker @uct.ac.za	Beyond supervision and reviewing contributions, assisted in conception and development of the framework	10

Signature of candidate: [DJA][10]
Date: 2022/11/07

Declaration by co-authors: The undersigned hereby confirm that:

---

[8] Code available at: github.com/SUMatEng/ADIC2D

[9] DOI link: https://doi.org/10.3390/rs12182906

[10] Declaration with signature in possession of the candidate and supervisor.

1. The declaration above accurately reflects the nature and extent of the contribution of the candidate and the co-authors to Chapter 4,
2. No other authors contributed to Chapter 4 besides those specified above, and
3. Potential conflicts of interest have been revealed to all interested parties and that the necessary arrangements have been made to use the material in Chapter 4 of this dissertation.

Signature (see Footnote 10)	**Institutional affiliation**	**Date**
[TH Becker]	UCT	2022/11/07

4.1 Nomenclature of Chapter

α	Homogeneous scaling variable
β	Window size for the Gaussian filter
c_s	Skew of sensor coordinate system (intrinsic camera parameter)
c_x, c_y	Translation applied to sensor coordinate system (intrinsic camera parameter)
C_{ZNSSD}	Zero-mean normalised sum of squared difference correlation criterion
C_{ZNCC}	Zero-mean normalised cross correlation criterion
C_{ObjFun}	Objective function of correlation
E_{proj}	Total projection error
F	Reference image
$\boldsymbol{f}$	Reference subset
$\bar{f}$	Mean light intensity of reference subset
$\tilde{f}$	Normalisation function of reference subset
∇F	Light intensity gradient of the reference image
$\nabla \boldsymbol{f}_i = \begin{bmatrix} \frac{\partial f_i}{\partial x} & \frac{\partial f_i}{\partial y} \end{bmatrix}$	Light intensity gradient of the reference subset
G	Deformed image
$\boldsymbol{g}$	Investigated subset
$\bar{g}$	Mean light intensity of investigated subset
$\tilde{g}$	Normalisation function of investigated subset
$\boldsymbol{H}$	Hessian of the optimisation equation
I	Number of pixels per-subset (counter is i)
$\boldsymbol{J}$	Jacobian of the optimisation equation
$\boldsymbol{K}$	Intrinsic camera parameters
κ_1, κ_2	Radial distortion parameters
L	Number of calibration images (counter is l)
M	Number of calibration targets per calibration image (counter is m)
MAE_u, MAE_v	Mean absolute error

$\boldsymbol{\omega}$	Function to populate a square matrix with the shape function parameters
ψ	Stopping criterion value
$\boldsymbol{P}$	Shape function parameters
$\boldsymbol{P}_{update}$	Updated shape function parameters
$\Delta\boldsymbol{P}$	Iterative improvement estimate of the shape function parameters
Q	Number of subsets per an image (counter is q)
$\boldsymbol{R}$	Rotation matrix
$RMSE_u$, $RMSE_v$	Root mean square error
ρ	Calibration plate thickness
σ^g	Standard deviation of the Gaussian function for Gaussian filtering
σ_u, σ_v	Variance (standard deviation of the absolute error)
t	Time
$\boldsymbol{T}$	Translation vector
$\boldsymbol{T}_{spec}$	Translation vector corrected for the thickness of the calibration plate
u	Displacement in the x-direction in the distorted sensor coordinate system
u^{true}	True displacement of subset in the x-direction in the distorted sensor coordinate system
u^{calc}	Calculated displacement of subset in the x-direction in the distorted sensor coordinate system
$\hat{u}_w$	Undistorted metric displacement in the x-direction in the world coordinate system
u_x, u_y, u_{xx}, u_{xy}, u_{yy}	Derivatives of the x-direction displacement
$\boldsymbol{V}$	Extrinsic camera parameters
v	Displacement in the y-direction in the distorted sensor coordinate system
v^{true}	True displacement of the subset in the y-direction in the distorted sensor coordinate system
v^{calc}	Calculated displacement of the subset in the y-direction in the distorted sensor coordinate system
$\hat{v}_w$	Undistorted metric displacement in the y-direction in the world coordinate system
v_x, v_y, v_{xx}, v_{xy}, v_{yy}	Derivatives of the y-direction displacement
$\boldsymbol{W}$	Shape function
$\frac{\partial \boldsymbol{W}_i}{\partial \boldsymbol{P}}$	Jacobian of the shape function in terms of the shape function parameters for pixel i
$\hat{\boldsymbol{x}}_w = \begin{bmatrix}\hat{x}_w & \hat{y}_w & \hat{z}_w\end{bmatrix}^T$	Ideal world coordinates
$\hat{\boldsymbol{x}}_s = \begin{bmatrix}\hat{x}_s & \hat{y}_s\end{bmatrix}^T$	Ideal sensor coordinates
$\hat{\boldsymbol{x}}_n = \begin{bmatrix}\hat{x}_n & \hat{y}_n\end{bmatrix}^T$	Normalised ideal image coordinates
$\boldsymbol{x}_n = \begin{bmatrix}x_n & y_n\end{bmatrix}^T$	Normalised distorted image coordinates
$\boldsymbol{x} = \begin{bmatrix}x & y\end{bmatrix}^T$	Distorted sensor coordinates

$\mathbf{x}^{true} = \begin{bmatrix} x^{true} & y^{true} \end{bmatrix}^T$	True location of the calibration targets in the distorted sensor coordinate system
$\mathbf{x}^{calc} = \begin{bmatrix} x^{calc} & y^{calc} \end{bmatrix}^T$	Location of the calibration targets in the distorted sensor coordinate system predicted by the camera model
$\mathbf{x}^{o} = \begin{bmatrix} x^{o} & y^{o} \end{bmatrix}^T$	Centre of reference subset in the distorted sensor coordinate system
$\hat{\mathbf{x}}^{o} = \begin{bmatrix} \hat{x}^{o} & \hat{y}^{o} \end{bmatrix}^T$	Centre of reference subset in the undistorted sensor coordinate system
$\mathbf{x}^{d} = \begin{bmatrix} x^{d} & y^{d} \end{bmatrix}^T$	Centre of investigated subset in the distorted sensor coordinate system
$\hat{\mathbf{x}}^{d} = \begin{bmatrix} \hat{x}^{d} & \hat{y}^{d} \end{bmatrix}^T$	Centre of investigated subset in the undistorted sensor coordinate system
$\mathbf{x}_i = \begin{bmatrix} x_i & y_i \end{bmatrix}^T$	i^{th} pixel position of the reference subset in the distorted sensor coordinate system
$\mathbf{x}'_i = \begin{bmatrix} x'_i & y'_i \end{bmatrix}^T$	i^{th} pixel position of the investigated subset in the distorted sensor coordinate system
$\Delta\mathbf{x}_i = \begin{bmatrix} \Delta x_i & \Delta y_i \end{bmatrix}^T$	Distance from the reference subset centre to i^{th} pixel position of reference subset
$\Delta\mathbf{x}'_i = \begin{bmatrix} \Delta x'_i & \Delta y'_i \end{bmatrix}^T$	Distance from the reference subset centre to i^{th} pixel position of investigated subset
$\hat{\mathbf{x}}^{o}_{w} = \begin{bmatrix} \hat{x}^{o}_{w} & \hat{y}^{o}_{w} \end{bmatrix}^T$	Undistorted reference subset position in the world coordinate system
$\hat{\mathbf{x}}^{d}_{w} = \begin{bmatrix} \hat{x}^{d}_{w} & \hat{y}^{d}_{w} \end{bmatrix}^T$	Undistorted investigated subset position in the world coordinate system
ξ_x, ξ_y	Scaling of metric units to pixels (intrinsic camera parameter)
ζ	Maximum distance of a pixel of the reference subset to the centre of the reference subset

4.2 Introduction

Digital image correlation (DIC) determines the displacements and deformations at multiple points spanning the surface of an object (full-field displacements and deformations) from images captured of the object. It is a full-field, non-contact, optical technique which are categorised as either interferometric or non-interferometric. The interferometric techniques, such as Electronic Speckle Pattern Interferometry and Moiré Interferometry, require a coherent light source and need to be isolated from vibrations [192]. As such, their utilisation is in the confines of a laboratory. In contrast non-interferometric techniques, DIC and the grid method, require simple incoherent light and are more robust with regards to ambient vibrations and light variations [22]. Thus non-interferometric techniques are more attractive due to their less stringent requirements and are mostly used in open literature. DIC allows for a more straightforward setup compared to the grid method as it only requires a random, irregular pattern on the surface of the object instead of a regular grid.

These advantages of DIC over other full-field, non-contact, optical techniques, along with the decreasing cost and increasing performance of digital cameras, has led to widespread use of DIC in various fields. Some applications of DIC include: (i) performing human pulse monitoring [51, 52]; (ii) analysing the stick-slip behaviour of tyre tread [193]; (iii) determining the mechanical properties of biological tissue [194, 195, 196]; (iv) in situ health monitoring of structures and components [197, 198, 199]; and (v) remote sensing applications [187, 200, 201, 202]. However, DIC has received the most attention, and thus development, for applications in experimental solid mechanics. As such this paper will predominantly focus on DIC in the context of experimental solid mechanics applications.

In the field of experimental solid mechanics measuring the displacement and deformation experienced by a specimen, as a result of an applied load, is essential to quantify its mechanical properties. As such DIC is advantageous for three reasons. Firstly, its full-field nature allows more complex constitutive equations to be used to determine more than one material property at a time using methods such as the Virtual Fields Method [203, 204, 173] and the Finite Element Model Updating Method [205]. Secondly, the non-contact nature of DIC avoids altering mechanical properties of the materials being tested such as in the case of determining the material properties of biological tissue [194, 195, 196] and hyper-elastic materials [206]. Lastly, DIC allows the specimen to be exposed to harsh environments such as high-temperature applications, while still being able to take measurements, provided the specimen is visible [47].

When DIC was first introduced by Peters and Ranson in 1982 [207] it used a simple cross-correlation criterion with a zero-order shape function (SF) and could not account for the deformation of the specimen or variations in ambient light. Between 1983 and 1989 Sutton and his colleagues improved the technique by introducing the first-order SF [9], the normalised cross-correlation criterion which is more robust against light variations [10], the Newton-Raphson (NR) optimisation method [153] and bi-cubic b-spline interpolation [11]. The two-dimensional (2D) DIC technique was extended to three dimensions (three-dimensional (3D) or stereo DIC) in 1993 by Luo et al. [208] and to digital volume correlation (DVC) in 1999 by Bay et al. [28] using X-Ray tomography-computed images.

The most significant contributions to the current state-of-the-art DIC technique, as identified by Pan [22], occurred during the 21st century. In 2000, Schreier et al. [12] proved that bi-quintic b-spline interpolation is the best interpolation method for accurate sub-pixel displacements. In the same year, Lu and Cary [13] introduced the second-order SF to account for more complex deformations. In 2004 Baker and Matthews [15] proposed the inverse compositional Gauss-Newton (IC-GN) optimisation method using the sum of squared difference correlation criterion which is more efficient than the NR method. However, Tong showed in 2005 [14] that the zero-mean normalised sum of squared difference (ZNSSD) correlation criterion is the most reliable and so Pan et al. [209] adapted the IC-GN method to use the ZNSSD criterion in 2013. Finally, Gao et al. [210] introduced the second-order SF to the IC-GN method in 2015. The IC-GN method is considered to be the state-of-the-art optimisation method because it has been shown to be theoretically equivalent to the NR method [15] while offering improved accuracy, robustness to noise and computational efficiency in practice [211].

The DIC process is complicated, comprising of several intricate elements, including: correlation, camera calibration, transformation of displacements between the device and real-world coordinates and strain computation. Successful application of DIC requires an understanding of all these elements and thus newcomers to the field need to overcome a difficult learning curve. To this end, there are several papers which give a comprehensive breakdown of the theory involved in the DIC process such as the papers by Pan et al. [209], Gao et al. [210] and Blaber et al. [166]. However, in order to contribute towards the development of DIC a deep understanding of the DIC process and its elements is required. It is incredibly time-consuming to gain this working knowledge due to a lack of publications that directly bridge the gap between the theory and its implementation in code. More specifically, papers either do not provide code that details the implementation of the theory in practice [209, 210] or the code that they provide is too complex to be beneficial as a learning resource [166].

This paper aims to bridge the gap between the theory and implementation of DIC. It does this by firstly presenting the theory for a 2D, subset based DIC framework that is predominantly consistent with current state-of-the-art practices. Thereafter the implementation of the theory of the framework as the provided 117 line MATLAB code is discussed. Lastly the correlation aspect of the code is validated using the DIC Challenge image sets documented by Reu et al. [122]. More specifically, its results are discussed in parallel with those obtained using either the commercial software package by LaVision (Davis) and the open-source software Ncorr [166] or, to results documented in the DIC Challenge paper [122], in order to draw conclusions.

The framework, referred to as the ADIC2D framework, is implemented using MATLAB because its simple syntax does not distract the reader from the mathematics of the code. Additionally its built-in functions are used to simplify the code and improve its efficiency. The code is modular, allowing readers to progressively build up their understanding of the code so that recognising the connection between the theory and code is straightforward. Moreover, this modularity allows for rapid adaption of the code thereby encouraging readers to develop the capabilities of DIC.

4.3 Framework theory

DIC consists of four processes: calibration, correlation, displacement transformation and strain computation. Calibration involves determining the parameters of the camera model which relates the location of a point on an object in the real world to the location of the corresponding point in an image taken of the object. Correlation calculates how portions of the object, captured in the image set, displace throughout the image set. Displacement transformation then uses the parameters determined by calibration to transform the pixel displacements determined by correlation to metric displacements in the real world. Finally strain computation determines the strain fields experienced by the specimen from the displacement fields.

4.3.1 Calibration

Calibration determines the parameters of the camera model. ADIC2D uses the pinhole camera model to transform the location of a point in the real world to the idealised location of the point in the image. Then a radial distortion model is used to relate the idealised location of this point to its actual, distorted location as illustrated in Figure 4.1.

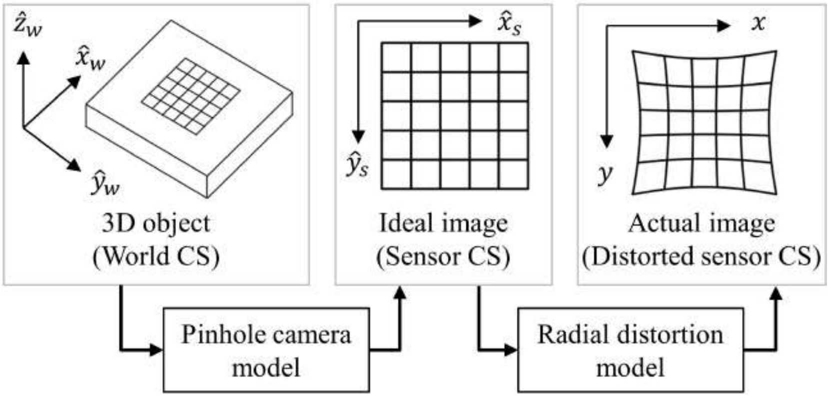


Figure 4.1: Schematic diagram illustrating how the camera model is comprised of the pinhole camera model and radial distortion model.

4.3.1.1 Homogeneous coordinates

The pinhole camera model works with homogeneous coordinates as these allow rotation, translation, scaling and perspective projection to be applied using matrix multiplication. An n-element vector, that represents a point in n-dimensional space, is converted to homogeneous coordinates by appending a scaling variable of unity to the end of the vector. Converting back from homogeneous coordinates involves dividing each element of the vector by the last element, the scaling variable, before removing the last element. Homogeneous coordinate vectors are indicated by underlining the variable name. For more information on homogeneous coordinates refer to the work of Bloomenthal and Rokne [212].

4.3.1.2 Pinhole camera model

The pinhole camera model relates the location of a point in the world coordinate system (CS) to its corresponding idealised location in the sensor CS. The 3D world CS is defined such that its x-y plane is coincident with the surface of the specimen under consideration – 2D DIC is limited to determining displacements that occur within this x-y plane. The 2D sensor CS is defined such that its x-y plane is coincident with the plane of the charge-coupled device which captures light rays incident upon its surface as an image. Let the homogeneous coordinates in the world and sensor CS be $\underline{\hat{\boldsymbol{x}}}_w = \begin{bmatrix}\hat{x}_w & \hat{y}_w & \hat{z}_w & 1\end{bmatrix}^T$ and $\underline{\hat{\boldsymbol{x}}}_s = \begin{bmatrix}\hat{x}_s & \hat{y}_s & 1\end{bmatrix}^T$, respectively. Note that the circumflex indicates that the coordinates are ideal (undistorted). The pinhole camera model is given as [213]

$$\alpha \underline{\hat{\boldsymbol{x}}}_s = \alpha \begin{bmatrix}\hat{x}_s \\ \hat{y}_s \\ 1\end{bmatrix} = \begin{bmatrix}\xi_x & c_s & c_x \\ 0 & \xi_y & c_y \\ 0 & 0 & 1\end{bmatrix} \begin{bmatrix}R_{11} & R_{12} & R_{13} & T_1 \\ R_{21} & R_{22} & R_{23} & T_2 \\ R_{31} & R_{32} & R_{33} & T_3\end{bmatrix} \begin{bmatrix}\hat{x}_w \\ \hat{y}_w \\ \hat{z}_w \\ 1\end{bmatrix} = \boldsymbol{K}\boldsymbol{V}\underline{\hat{\boldsymbol{x}}}_w \tag{4.1}$$

where matrices $\boldsymbol{V}$ and $\boldsymbol{K}$ contain the extrinsic and intrinsic camera parameters, respectively. The extrinsic camera parameters define a rotation matrix $\boldsymbol{R}$ and a translation vector $\boldsymbol{T}$ which define the position and orientation of the world CS relative to the position and orientation of the camera. Thus the extrinsic camera parameters change if the relative position or orientation between the specimen and camera change.

In contrast the intrinsic camera parameters remain unchanged because they are only dependent on the camera system. The parameters ξ_x and ξ_y perform scaling from metric units to units of pixels. This paper uses millimetres as the metric units. The parameters c_x and c_y apply translation such that the origin of the sensor CS is at the top left of the image as shown in Figure 4.1. The parameter c_s converts from an orthogonal CS to a skewed sensor CS. Here $c_s = 1$ since an orthogonal sensor CS is assumed.[11] The parameter α is an arbitrary scaling variable of the homogeneous coordinates which is factored out. For more information on the pinhole camera model refer to the work of Zhang [213] and Heikkila et al. [214].

4.3.1.3 Radial distortion model

According to Tsai [215] and Wei et al. [216] the difference between the ideal and actual image can be well accounted for by using only a radial distortion model. Radial distortion is caused by the lens system having different magnification levels depending on where the light ray passes through the lenses. The image experiences either an increase (pincushion distortion) or decrease (barrel distortion) in magnification with increasing distance from the optical axis. The radial distortion model requires that $\underline{\hat{\boldsymbol{x}}}_s$ be converted to normalised ideal image coordinates, $\hat{\boldsymbol{x}}_n = \begin{bmatrix}\hat{x}_n & \hat{y}_n\end{bmatrix}^T$, using the inverse of the intrinsic parameter matrix as

$$\hat{\boldsymbol{x}}_n = \begin{bmatrix}\hat{x}_n \\ \hat{y}_n\end{bmatrix} = \begin{bmatrix}1 & 0 & 0 \\ 0 & 1 & 0\end{bmatrix}\left(\boldsymbol{K}^{-1}\underline{\hat{\boldsymbol{x}}}_s\right) \tag{4.2}$$

This equation includes a matrix to convert from homogeneous coordinates to Cartesian coordinates. $\hat{\boldsymbol{x}}_n$ is related to the normalised, distorted image coordinates, $\boldsymbol{x}_n = \begin{bmatrix}x_n & y_n\end{bmatrix}^T$, as [213]

$$\boldsymbol{x}_n = \left(1 + \kappa_1 \hat{\boldsymbol{x}}_n^T \hat{\boldsymbol{x}}_n + \kappa_2 \left(\hat{\boldsymbol{x}}_n^T \hat{\boldsymbol{x}}_n\right)^2\right)\hat{\boldsymbol{x}}_n \tag{4.3}$$

where κ_1 and κ_2 are the unit-less radial distortion parameters that quantify the severity of the distortion. $\boldsymbol{x}_n$ is converted to distorted coordinates in the distorted sensor CS, $\boldsymbol{x} = \begin{bmatrix}x & y\end{bmatrix}^T$, as

$$\boldsymbol{x} = \begin{bmatrix}x \\ y\end{bmatrix} = \begin{bmatrix}1 & 0 & 0 \\ 0 & 1 & 0\end{bmatrix}\boldsymbol{K}\begin{bmatrix}x_n \\ y_n \\ 1\end{bmatrix} \tag{4.4}$$

4.3.1.4 Calibration process

Calibration determines the extrinsic, intrinsic and radial distortion parameters using images taken of a calibration plate. A calibration plate is an object with a flat surface having a high contrast, regular pattern which contains distinctive, point-like features called calibration targets (CTs). It is used to define a set of 3D coordinates in the world CS and a corresponding set of distorted, 2D coordinates in the distorted sensor CS.

---

[11] As noted in Section 4.8, c_s should at set as $c_s = 0$.

The 3D coordinates of these CTs in the world CS are predefined. In fact, they lie on the x-y plane of the world CS and define its position and orientation. The set of corresponding distorted, 2D coordinates in the sensor CS can be determined by locating the CTs in an image taken of the calibration plate. These two sets of 3D and 2D coordinates are used to solve for the parameters, of the camera model, which describe the relationship between the two. This is done in two steps.

The first step determines initial estimates for the extrinsic and intrinsic camera parameters using the closed form solution method proposed by Zhang [213]. The initial estimate of the radial distortion parameters are set to zero.

The second step works with two sets of CTs in the distorted sensor CS: the true CTs, $\boldsymbol{x}^{true} = \begin{bmatrix} x^{true} & y^{true} \end{bmatrix}^T$, obtained directly from the calibration images and the calculated CTs, $\boldsymbol{x}^{calc} = \begin{bmatrix} x^{calc} & y^{calc} \end{bmatrix}^T$, obtained by transforming the known CTs of the world CS to the distorted sensor CS using the camera model and the current estimate of the calibration parameters. The difference between the true and calculated CTs is quantified as the total projection error, E_{proj}, given as

$$E_{proj} = \sum_{l=1}^{L} \sum_{m=1}^{M} \left(\left(x_{lm}^{calc} - x_{lm}^{true} \right)^2 + \left(y_{lm}^{calc} - y_{lm}^{true} \right)^2 \right) \tag{4.5}$$

There are L many calibration images and M many CTs per calibration image. The second step uses iterative, non-linear, least-squares optimisation to solve for the calibration parameters which minimise E_{proj}. Note that multiple calibration images are used in order to form an over-determined system of equations. This makes the calibration process less sensitive to noise inherent in the images. For more information on the calibration process refer to the work of Zhang [213] and Heikkila et al. [214].

The last process in calibration corrects $\boldsymbol{T}$ for the thickness of the calibration plate, ρ, such that the x-y plane of the world CS is coincident with the surface of the specimen under consideration. The corrected translation vector, $\boldsymbol{T}_{spec}$, that replaces $\boldsymbol{T}$ in Equation (4.1) is determined as

$$\boldsymbol{T}_{spec} = \boldsymbol{T} - \boldsymbol{R} \begin{bmatrix} 0 \\ 0 \\ \rho \end{bmatrix} \tag{4.6}$$

where $\boldsymbol{T}$ and $\boldsymbol{R}$ are the translation vector and rotation matrix determined by the above calibration process, respectively.

4.3.2 Correlation

Correlation considers two images: a reference image, F, representing the specimen at time $t = 0$, and a deformed image, G, representing the specimen at time $t = 1$. F is broken up into subsets which are groups of neighbouring pixels. Conceptually correlation attempts to determine how a reference subset ($\boldsymbol{f}$) must displace and deform such that it matches a corresponding subset, the investigated subset ($\boldsymbol{g}$) in G. In practice, however, $\boldsymbol{f}$ remains unchanged while its pixel centre positions (hereafter referred to as pixel positions) are displaced and deformed according to $\boldsymbol{W}$, a predefined SF, resulting in the query points of the investigated subset. The investigated subset is obtained by sampling the deformed image at these query points. To better understand this some details of correlation need to be explained.

Correlation operates in the distorted sensor CS as illustrated in Figure 4.2. f's centre position, $\boldsymbol{x}^0 = \begin{bmatrix} x^0 & y^0 \end{bmatrix}^T$, has been displaced by u and v in the x- and y-direction, respectively, to obtain g's centre position, $\boldsymbol{x}^d = \begin{bmatrix} x^d & y^d \end{bmatrix}^T$. The i^{th} pixel position of f, given by $\boldsymbol{x}_i = \begin{bmatrix} x_i & y_i \end{bmatrix}^T$, is based on $\boldsymbol{x}^0$ and the distance from $\boldsymbol{x}^0$ to $\boldsymbol{x}_i$, $\Delta\boldsymbol{x}_i = \begin{bmatrix} \Delta x_i & \Delta y_i \end{bmatrix}^T$, as

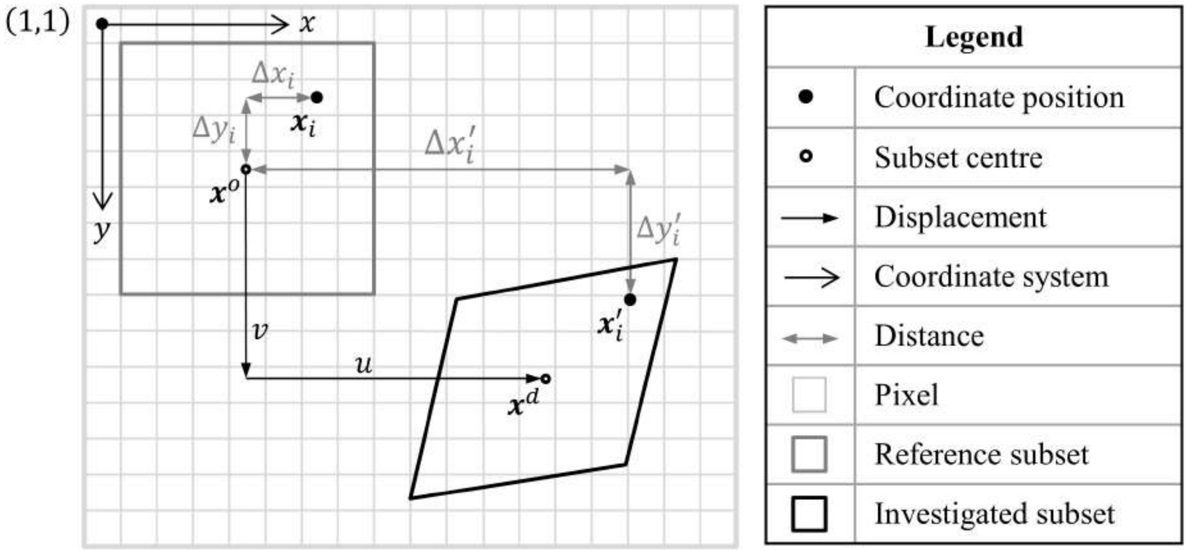


Figure 4.2: Schematic diagram illustrating how the pixel positions of the reference and investigated subsets are related to one another within the distorted sensor CS.

$$\boldsymbol{x}_i = \Delta\boldsymbol{x}_i + \boldsymbol{x}_0 = \begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} \Delta x_i \\ \Delta y_i \end{bmatrix} + \begin{bmatrix} x^0 \\ y^0 \end{bmatrix} \tag{4.7}$$

Similarly, the corresponding i^{th} query point of g, $\boldsymbol{x}'_i = \begin{bmatrix} x'_i & y'_i \end{bmatrix}^T$, is based on $\boldsymbol{x}^0$ and the distance from $\boldsymbol{x}^0$ to $\boldsymbol{x}'_i$, $\Delta\boldsymbol{x}'_i = \begin{bmatrix} \Delta x'_i & \Delta y'_i \end{bmatrix}^T$, as

$$\boldsymbol{x}'_i = \Delta\boldsymbol{x}'_i + \boldsymbol{x}^o = \begin{bmatrix} x'_i \\ y'_i \end{bmatrix} = \begin{bmatrix} \Delta x'_i \\ \Delta y'_i \end{bmatrix} + \begin{bmatrix} x^0 \\ y^0 \end{bmatrix} \tag{4.8}$$

$\Delta\boldsymbol{x}'_i$ is defined relative to $\boldsymbol{x}^o$ because $\boldsymbol{x}^d$ is unknown prior to correlation. u and v are a special case of $\Delta y'_i$ and $\Delta x'_i$ for the pixel at the centre of the investigated subset. $\Delta\boldsymbol{x}'_i$ is determined using $\boldsymbol{W}$ which modifies $\Delta\boldsymbol{x}_i$ according to a given displacement and deformation quantified by the shape function parameters (SFPs), $\boldsymbol{P}$, as

$$\Delta\boldsymbol{x}'_i = \boldsymbol{W}\left(\Delta\boldsymbol{x}_i, \boldsymbol{P}\right). \tag{4.9}$$

Each pixel of the investigated subset, g_i, is populated by sampling the light intensity of the deformed image at $\boldsymbol{x}'_i$. However, images are discrete and so interpolation must be used to obtain these light intensities of G at non-integer locations. As such, F and G are treated as functions which return the light intensity at a location in the image. For G this involves interpolation. The pixels of f and g are populated by sampling these functions as

$$f_i = F\left(\boldsymbol{x}^o + \Delta \boldsymbol{x}_i\right) \quad \text{and} \quad g_i \; = G\left(\boldsymbol{x}^o + \boldsymbol{W}\left(\Delta \boldsymbol{x}_i, \boldsymbol{P}\right)\right) \tag{4.10}$$

The similarity between $\boldsymbol{f}$ and $\boldsymbol{g}$ is quantified by the correlation criterion. Correlation aims to find the SFPs which define an investigated subset which closely matches the reference subset.

4.3.2.1 Correlation criterion

The two most popular types are the ZNSSD and zero-mean normalised cross-correlation (ZNCC) criteria which are robust against offset and scaling changes in light intensity. The ZNSSD criterion, which has a range of $\{C_{ZNSSD} \in \mathbb{R} | 0 \leq C_{ZNSSD} \leq 4\}$ where 0 indicates a perfect match, is calculated as

$$c_{ZNSSD} = \sum_{i=1}^{I} [\frac{f_i - \bar{f}}{\tilde{f}} - \frac{g_i - \bar{g}}{\tilde{g}}]^2 \tag{4.11}$$

where I is the number of pixels contained within a subset, $\bar{f} = \frac{\sum_{i=1}^{I} f_i}{I}$ and $\bar{g} = \frac{\sum_{i=1}^{I} g_i}{I}$ are the mean light intensity values, and $\tilde{f} = \sqrt{\sum_{i=1}^{I}\left(f_i - \bar{f}\right)^2}$ and $\tilde{g} = \sqrt{\sum_{i=1}^{I}\left(g_i - \bar{g}\right)^2}$ are the normalisation functions of subsets $\boldsymbol{f}$ and $\boldsymbol{g}$, respectively. Similarly, the ZNCC criterion, which has a range of $\{C_{ZNCC} \in \mathbb{R} | -1 \leq C_{ZNCC} \leq 1\}$ where 1 indicates a perfect match, is given as

$$C_{ZNCC} = \sum_{I=1}^{I} \frac{(f_i - \bar{f})(g_i - \bar{g})}{\tilde{f}\tilde{g}} \tag{4.12}$$

Pan et al. [102] proved that these two criteria are related as

$$C_{ZNCC} = 1 - \frac{C_{ZNSSD}}{2} \tag{4.13}$$

The more computationally efficient ZNSSD criterion is evaluated within ADIC2D; however it is reported as the ZNCC coefficient, using Equation (4.13), because its range is more intuitive. For more information on correlation criteria refer to the work of Pan et al. [102].

4.3.2.2 Shape function

The most common SFs are the zero ($\boldsymbol{W}^{SF0}$), first ($\boldsymbol{W}^{SF1}$) and second-order SFs ($\boldsymbol{W}^{SF2}$) expressed as [13]

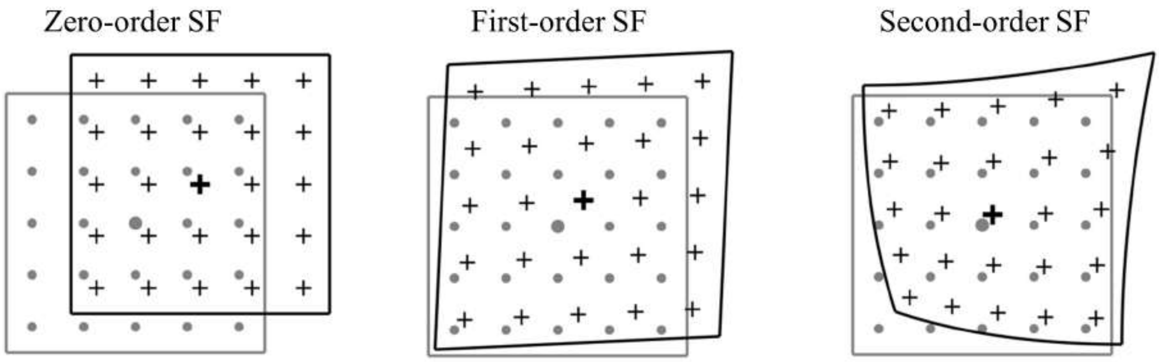
$$\boldsymbol{W}^{SF0}\left(\Delta \boldsymbol{x}_i, \boldsymbol{P}^{SF0}\right) = \begin{bmatrix} 1 & 0 & u \\ 0 & 1 & v \end{bmatrix} \begin{bmatrix} \Delta x_i \\ \Delta y_i \\ 1 \end{bmatrix}$$

$$\boldsymbol{W}^{SF1}\left(\Delta \boldsymbol{x}_i, \boldsymbol{P}^{SF1}\right) = \begin{bmatrix} 1 + u_x & u_y & u \\ v_x & 1 + v_y & v \end{bmatrix} \begin{bmatrix} \Delta x_i \\ \Delta y_i \\ 1 \end{bmatrix} \tag{4.14}$$

and

$$\boldsymbol{W}^{SF2}\left(\Delta\boldsymbol{x}_i, \boldsymbol{P}^{SF2}\right) = \begin{bmatrix} \frac{1}{2}u_{xx} & u_{xy} & \frac{1}{2}u_{yy} & 1+u_x & u_y & u \\ \frac{1}{2}v_{xx} & v_{xy} & \frac{1}{2}v_{yy} & v_x & 1+v_y & v \end{bmatrix} \begin{bmatrix} \Delta x_i^2 \\ \Delta x_i \Delta y_i \\ \Delta y_i^2 \\ \Delta x_i \\ \Delta y_i \\ 1 \end{bmatrix}$$

where u and v represent the displacement of $\boldsymbol{x}^o$ in the x- and y-directions, respectively, and their derivatives (subscript x and y) define the deformation with respect to the reference subset. Specifically, u_x, u_{xx}, v_y and v_{yy} represent elongation while u_y, v_x, u_{yy}, v_{xx}, u_{xy} and v_{xy} represent shearing of the subset. Higher order SFs, containing higher order displacement derivatives, allow for more complex deformation as shown in Figure 4.3.



Legend	Subset outline	Pixel positions	Subset centre
Reference subset	—	•	●
Investigated subset	**—**	+	**+**

Figure 4.3: Schematic diagram illustrating the allowable deformation of a subset for various SF orders.

This enables higher order SFs to more reliably track subsets in complex displacement fields. The elements of $\boldsymbol{P}$, for each SF order, are stored as

$$\begin{aligned} \boldsymbol{P}^{SF0} &= \begin{bmatrix} u & v \end{bmatrix}^T \\ \boldsymbol{P}^{SF1} &= \begin{bmatrix} u & u_x & u_y & v & v_x & v_y \end{bmatrix}^T \\ &\text{and} \\ \boldsymbol{P}^{SF2} &= \begin{bmatrix} u & u_x & u_y & u_{xx} & u_{xy} & u_{yy} & v & v_x & v_y & v_{xx} & v_{xy} & v_{yy} \end{bmatrix}^T \end{aligned} \tag{4.15}$$

4.3.2.3 Interpolation

Interpolation determines the value at a query point ($\boldsymbol{x}_i'$) in an image by fitting an equation to the surrounding light intensity data and evaluating the equation at $\boldsymbol{x}_i'$. Polynomial interpolation and b-spline interpolation, shown in Figure 4.4 for the one-dimensional case, are the most popular types for DIC.

Polynomial interpolation fits a local polynomial equation of order n to a window of data of size $n+1$ as shown in grey in Figure 4.4(b) for cubic polynomial interpolation. The resulting interpolation equation is a piecewise polynomial where only the central portion of each local polynomial equation is used. The interpolation equation is C^0 and C^1 continuous for linear and cubic polynomial interpolation, respectively. Refer to the work of Keys [217] for more information on cubic polynomial interpolation.

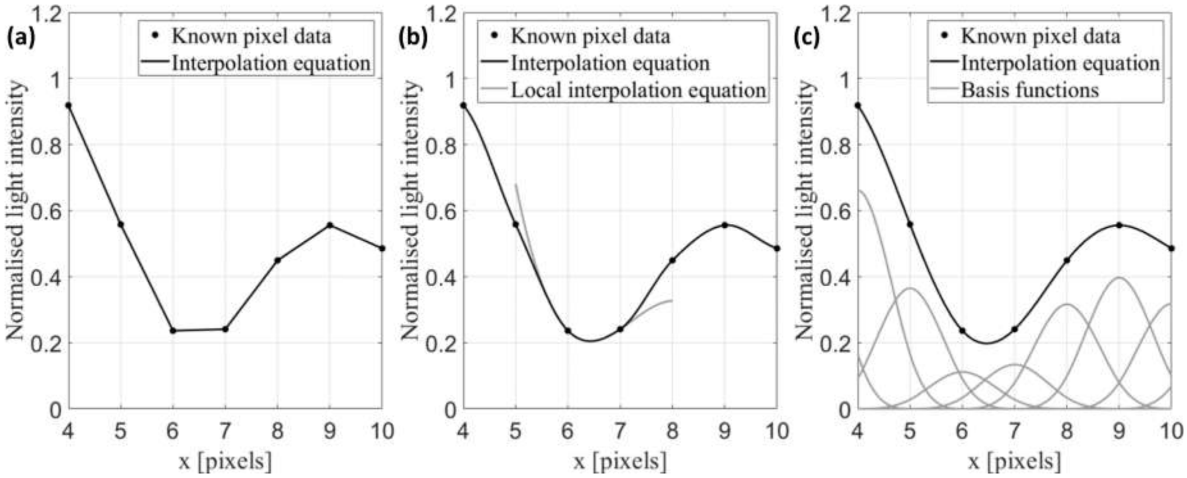


Figure 4.4: Graphical representation of the interpolation equations for: (a) linear polynomial; (b) cubic polynomial; and (c) cubic b-spline interpolation methods.

In contrast, b-spline interpolation builds up an interpolation equation from locally supported basis functions. More specifically, a basis function is defined at each data point and the coefficients of all these basis functions are determined simultaneously from the data. This is done such that the summation of the basis functions forms the interpolation equation as shown in Figure 4.4(c). For cubic b-spline the interpolation equation is C^2 continuous. Refer to the work of Hou et al. [218] for an in-depth discussion of bi-cubic b-spline interpolation.

The interpolation method should be as exact as possible in order for correlation to determine sub-pixel displacements reliably and efficiently because interpolation is the most time consuming part of correlation for iterative, sub-pixel DIC [219].

4.3.2.4 Gaussian filtering

High order interpolation methods, such as bi-cubic b-spline interpolation, are sensitive to high frequency noise contained in the images [114]. A Gaussian, low-pass filter is used to attenuate the high frequency noise of each image of the image set in order to reduce the bias of the displacement results caused by the interpolation method. Gaussian filtering convolves a 2D Gaussian point-spread function (PSF) with the image. The Gaussian function consists of a window size, β (in pixels), and standard deviation, σ^g, to determine a weighted average light intensity at each pixel position in the filtered image from a window of pixels in the unfiltered image. The Gaussian point-spread function is scaled such that the sum of itself equals 1.

Although interpolation is only required for G, all the images of the image set (including F) need to be filtered such that the light intensity patterns of the subsets, considered by the correlation criterion, are directly comparable. Despite the fact that variance of the displacement results is independent of the interpolation method, it is dependent on the image detail which is reduced by smoothing [60]. Therefore, β and σ^g should be chosen to reduce bias while not significantly increasing variance. For more information on Gaussian filtering refer to [114].

4.3.2.5 Optimisation method

The optimisation problem aims to minimise the correlation criterion (Equation (4.11)) by using the IC-GN method to iteratively solve for the optimal SFPs. An illustration of this process is shown in Figure 4.5. Substituting Equation (4.10) into Equation (4.11) results in an expression in terms of F and G being obtained. In addition, Equation (4.11) is modified to include an iterative improvement estimate, $\Delta \boldsymbol{P}$. Normally, iterative updating uses the forward additive implementation in which both $\Delta \boldsymbol{P}$ and $\boldsymbol{P}$ are applied to the investigated subset as $\boldsymbol{P} + \Delta \boldsymbol{P}$. However, for the inverse compositional implementation $\Delta \boldsymbol{P}$ is applied to the reference subset and the current estimate of $\boldsymbol{P}$ is applied to the investigated subset. Thus, the objective function is given as

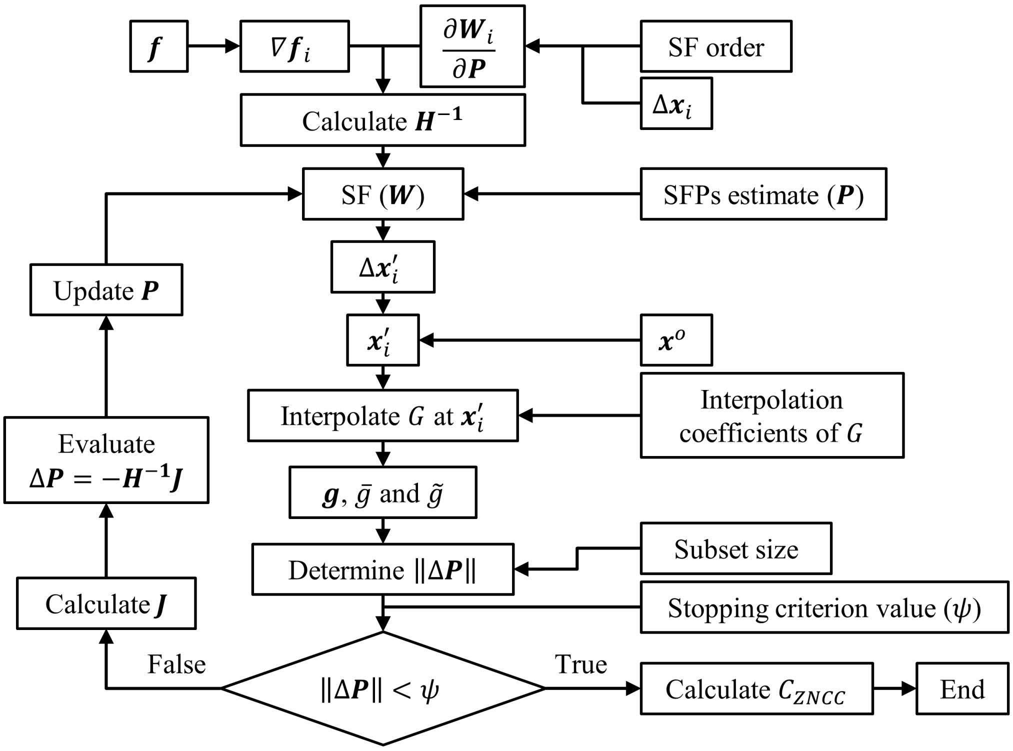


Figure 4.5: Flow diagram of ADIC2D's correlation process.

$$C_{ObjFun} = \sum_{i=1}^{I} \left[\frac{F\left(\boldsymbol{x}^o + \boldsymbol{W}\left(\Delta \boldsymbol{x}_i, \Delta \boldsymbol{P}\right)\right) - \bar{f}}{\tilde{f}} - \frac{G\left(\boldsymbol{x}^o + \boldsymbol{W}\left(\Delta \boldsymbol{x}_i, \boldsymbol{P}\right)\right) - \bar{g}}{\tilde{g}} \right]^2 \tag{4.16}$$

Taking the first-order Taylor series expansion of Equation (4.16) in terms of $\Delta \boldsymbol{P}$ gives

$$C_{ObjFun} = \sum_{i=1}^{I} \left[\frac{F\left(\boldsymbol{x}^{o} + \Delta \boldsymbol{x}_{i}\right) + \nabla f_{i} \frac{\partial \boldsymbol{W}_{i}}{\partial \boldsymbol{P}} \Delta \boldsymbol{P} - \bar{f}}{\tilde{f}} - \frac{G\left(\boldsymbol{x}^{o} + \boldsymbol{W}\left(\Delta \boldsymbol{x}_{i}, \boldsymbol{P}\right)\right) - \bar{g}}{\tilde{g}} \right]^{2} \tag{4.17}$$

where $\nabla f_i = \begin{bmatrix} \frac{\partial f_i}{\partial x} & \frac{\partial f_i}{\partial y} \end{bmatrix}$ is the light intensity gradient of f and $\frac{\partial \boldsymbol{W}_i}{\partial \boldsymbol{P}}$ is the Jacobian of the SF at each pixel position. For the zero, first and second-order SFs $\frac{\partial \boldsymbol{W}_i}{\partial \boldsymbol{P}}$ is given as [13, 15]

$$\frac{\partial \boldsymbol{W}_i^{SF0}}{\partial \boldsymbol{P}^{SF0}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\frac{\partial \boldsymbol{W}_i^{SF1}}{\partial \boldsymbol{P}^{SF1}} = \begin{bmatrix} 1 & \Delta x_i & \Delta y_i & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta x_i & \Delta y_i \end{bmatrix} \tag{4.18}$$

and

$$\frac{\partial \boldsymbol{W}_i^{SF2}}{\partial \boldsymbol{P}^{SF2}} = \begin{bmatrix} 1 & \Delta x_i & \Delta y_i & \frac{\Delta x_i^2}{2} & \Delta x_i \Delta y_i & \frac{\Delta y_i^2}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & \Delta x_i & \Delta y_i & \frac{\Delta x_i^2}{2} & \Delta x_i \Delta y_i & \frac{\Delta y_i^2}{2} \end{bmatrix}$$

Setting Equation (4.17) to zero and taking the derivative with respect to $\Delta \boldsymbol{P}$ gives the first-order, least-squares solution. Rearranging to make $\Delta \boldsymbol{P}$ the subject of the equation yields

$$\Delta \boldsymbol{P} = -\boldsymbol{H}^{-1} \sum_{i=1}^{I} \left(\nabla f_i \frac{\partial \boldsymbol{W}_i}{\partial \boldsymbol{P}} \right)^{T} \left[f_i - \bar{f} - \frac{\tilde{f}}{\tilde{g}} \left(G\left(\boldsymbol{x}^{o} + \boldsymbol{W}\left(\Delta \boldsymbol{x}_i, \boldsymbol{P}\right)\right) - \bar{g} \right) \right] \tag{4.19}$$

where $\boldsymbol{H}$ is the Hessian given by Equation (4.20) and the remaining terms, within the summation, of Equation (4.19) form the Jacobian, $\boldsymbol{J}$. $\boldsymbol{H}$ is independent of the SFPs and remains constant during iterations. Thus, Equation (4.20) can be pre-computed before iterations begin.

$$\boldsymbol{H} = \sum_{i=1}^{I} \left[\left(\nabla f_i \frac{\partial \boldsymbol{W}_i}{\partial \boldsymbol{P}} \right)^{T} \left(\nabla f_i \frac{\partial \boldsymbol{W}_i}{\partial \boldsymbol{P}} \right) \right] \tag{4.20}$$

Note that since $\Delta \boldsymbol{P}$ is applied to the reference subset, each iteration solves for a set of SFPs which if applied to the reference subset would improve the correlation criterion. However, instead of applying $\Delta \boldsymbol{P}$ to the reference subset it is used to improve the estimate of the SFPs of the investigated subset. More specifically, the updated SFPs of the investigated subset, $\boldsymbol{P}_{update}$, are obtained by composing the inverted iterative improvement, $\Delta \boldsymbol{P}$, with the current estimate, $\boldsymbol{P}$, as

$$\boldsymbol{P}_{update} = \boldsymbol{\omega}\left(\boldsymbol{P}\right) \boldsymbol{\omega}\left(\Delta \boldsymbol{P}\right)^{-1} \tag{4.21}$$

where $\boldsymbol{\omega}$ is a function which populates a square matrix with the values of the SFPs as [210]

$$\boldsymbol{\omega}^{SF0}\left(\boldsymbol{P}^{SF0}\right) = \begin{bmatrix} 1 & 0 & u \\ 0 & 1 & v \\ 0 & 0 & 1 \end{bmatrix}$$

$$\boldsymbol{\omega}^{SF1}\left(\boldsymbol{P}^{SF1}\right)=\begin{bmatrix}1+u_x & u_y & u\\ v_x & 1+v_y & v\\ 0 & 0 & 1\end{bmatrix} \tag{4.22}$$

$$\boldsymbol{\omega}^{SF2}\left(\boldsymbol{P}^{SF2}\right)=\begin{bmatrix}1+A_1 & A_2 & A_3 & A_4 & A_5 & A_6\\ A_7 & 1+A_8 & A_9 & A_{10} & A_{11} & A_{12}\\ A_{13} & A_{14} & 1+A_{15} & A_{16} & A_{17} & A_{18}\\ \frac{1}{2}u_{xx} & u_{xy} & \frac{1}{2}u_{yy} & 1+u_x & u_y & u\\ \frac{1}{2}v_{xx} & v_{xy} & \frac{1}{2}v_{yy} & v_x & 1+v_y & v\\ 0 & 0 & 0 & 0 & 0 & 1\end{bmatrix}$$

where A_1 through A_{18} are

$$\begin{array}{cc}
A_1 = 2u_x + u_x^2 + uu_{xx} & A_2 = 2uu_{xy} + 2\left(1+u_x\right)u_y\\
A_3 = u_y^2 + uu_{yy} & A_4 = 2u\left(1+u_x\right)\\
A_5 = 2uu_y & A_6 = u^2\\
A_7 = \frac{1}{2}\left(vu_{xx} + 2\left(1+u_x\right)v_x + uv_{xx}\right) & A_8 = u_yv_x + u_xv_y + vu_{xy} + uv_{xy} + v_y + u_x\\
A_9 = \frac{1}{2}\left(vu_{yy} + 2\left(1+v_y\right)u_y + uv_{yy}\right) & A_{10} = v + vu_x + uv_x\\
A_{11} = u + vu_y + uv_y & A_{12} = uv\\
A_{13} = v_x^2 + vv_{xx} & A_{14} = 2vv_{xy} + 2v_x\left(1+v_y\right)\\
A_{15} = 2v_y + v_y^2 + vv_{yy} & A_{16} = 2vv_x\\
A_{17} = 2v\left(1+v_y\right) & A_{18} = v^2
\end{array}$$

The optimisation method is computationally efficient because before iterations begin the following are computed: (i) $\boldsymbol{H}$ and its inverse; (ii) the interpolation coefficients of G; and (iii) the image gradients of F using the Prewitt gradient operator. Each iteration step involves evaluating $\boldsymbol{W}$ (Equation (4.14)) using the current estimate of $\boldsymbol{P}$ to obtain $\Delta\boldsymbol{x}'_i$, which is used by Equation (4.8) to compute $\boldsymbol{x}'_i$, interpolating G at $\boldsymbol{x}'_i$ in order to compute $\boldsymbol{g}$, $\bar{g}$ and $\tilde{g}$ and finally computing $\Delta\boldsymbol{P}$ using Equation (4.19). For each iteration $\boldsymbol{P}$ is updated using Equation (4.21). Iterations continue until the stopping criterion deems that $\boldsymbol{P}$ is a solution. The correlation coefficient is then computed using Equation (4.11) substituted into Equation (4.13) and u and v are obtained from the SFPs.

4.3.2.6 Stopping criterion

Iterations stop once the change in SFPs, $\|\Delta\boldsymbol{P}\|$, is below a specified threshold referred to as the stopping criterion value (ψ) [209]. The expressions for $\|\Delta\boldsymbol{P}\|$ for the SF orders are [210]

$$\begin{aligned}
\left\|\Delta\boldsymbol{P}^{SF0}\right\| &= \left[\Delta u^2 + \Delta v^2\right]^{0.5}\\
\left\|\Delta\boldsymbol{P}^{SF1}\right\| &= \left[\Delta u^2 + \left(\Delta u_x\zeta\right)^2 + \left(\Delta u_y\zeta\right)^2 + \Delta v^2 + \left(\Delta v_x\zeta\right)^2 + \left(\Delta v_y\zeta\right)^2\right]^{0.5}\\
\text{and } \left\|\Delta\boldsymbol{P}^{SF2}\right\| &= \Bigg[\Delta u^2 + \left(\Delta u_x\zeta\right)^2 + \left(\Delta u_y\zeta\right)^2 + \left(\frac{1}{2}\Delta u_{xx}\zeta\right)^2 + \left(\frac{1}{2}\Delta u_{xy}\zeta\right)^2 +\\
&\left(\frac{1}{2}\Delta u_{yy}\zeta\right)^2 + \Delta v^2 + \left(\Delta v_x\zeta\right)^2 + \left(\Delta v_y\zeta\right)^2 + \left(\frac{1}{2}\Delta v_{xx}\zeta\right)^2 + \left(\frac{1}{2}\Delta v_{xy}\zeta\right)^2 + \left(\frac{1}{2}\Delta v_{yy}\zeta\right)^2\Bigg]^{0.5}
\end{aligned} \tag{4.23}$$

where $\zeta = \frac{\sqrt{I}-1}{2}$ is the furthest distance from $\boldsymbol{x}^o$.

4.3.3 Displacement transformation

Displacement transformation maps u and v from the distorted sensor CS to the world CS. First, the position of the investigated subset, $\boldsymbol{x}^d$, is determined as

$$\begin{bmatrix} x^d \\ y^d \end{bmatrix} = \begin{bmatrix} x^o \\ y^o \end{bmatrix} + \begin{bmatrix} u \\ v \end{bmatrix} \tag{4.24}$$

An exact analytical solution for the inverse of Equation (4.3) does not exist because it requires determining the roots of a polynomial of degree greater than four [220]. As such distortion is removed from the reference and investigated subset positions using non-linear, least-squares optimisation.

The resulting undistorted sensor coordinates of the subset before, $\hat{\boldsymbol{x}}^o = \begin{bmatrix} \hat{x}^o & \hat{y}^o \end{bmatrix}^T$, and after deformation, $\hat{\boldsymbol{x}}^d = \begin{bmatrix} \hat{x}^d & \hat{y}^d \end{bmatrix}^T$, are transformed to the world CS using the inverse of the pinhole camera model as

$$\frac{1}{\alpha} \begin{bmatrix} \hat{x}_w \\ \hat{y}_w \\ 1 \end{bmatrix} = \left(\begin{bmatrix} \xi_x & c_s & c_x \\ 0 & \xi_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} & T_1 \\ R_{21} & R_{22} & T_2 \\ R_{31} & R_{32} & T_3 \end{bmatrix} \right)^{-1} \begin{bmatrix} \hat{x}_s \\ \hat{y}_s \\ 1 \end{bmatrix} \tag{4.25}$$

The corrected translation vector determined by Equation (4.6) is used in Equation (4.25). The resulting position of the reference, $\hat{\boldsymbol{x}}_w^o = \begin{bmatrix} \hat{x}_w^o & \hat{y}_w^o \end{bmatrix}^T$, and investigated subsets, $\hat{\boldsymbol{x}}_w^d = \begin{bmatrix} \hat{x}_w^d & \hat{y}_w^d \end{bmatrix}^T$, in the world CS are used to determine the metric displacement experienced by the subset, $\begin{bmatrix} \hat{u}_w & \hat{v}_w \end{bmatrix}^T$, as

$$\begin{bmatrix} \hat{u}_w \\ \hat{v}_w \end{bmatrix} = \begin{bmatrix} \hat{x}_w^d \\ \hat{y}_w^d \end{bmatrix} - \begin{bmatrix} \hat{x}_w^o \\ \hat{y}_w^o \end{bmatrix} \tag{4.26}$$

4.3.4 Strain computation

Strains are computed from the gradients of the displacements determined using Equation (4.26). A method of smoothing displacements before differentiation is recommended because these displacements contain noise which is amplified by differentiation. The method of point-wise least-squares proposed by Pan et al. [221] fits a planar surface to a window of displacement data using linear, least-squares optimisation with the subset of interest located at the centre of the window. The resulting equation for the planar surface is differentiated to determine the displacement gradients for the subset of interest. This is done for each subset and these displacement gradients are used to calculate the strains.

4.4 Framework implementation

The ADIC2D framework, provided in Appendix A[12], is called from the command prompt as "`ProcData=ADIC2D(FileNames, Mask, GaussFilt, StepSize, SubSize, SubShape, SFOrder, RefStrat, StopCritVal, WorldCTs, ImgCTs, rho)`" requiring input variables as defined in Table 4.1 and providing an output

---

[12] Due to page limit constraints the code is not provided within the appendices but can be accessed at: https://github.com/SUMatEng/ADIC2D

variable as a structured array containing data for each analysed image `d` and subset `q` as detailed in Table 4.2

Table 4.1: Description of the required input variables for the ADIC2D framework.

Variable	**Variable description**
`FileNames`	Cell array of character vectors containing the image file names of the image set `d`. All images need to be the same size.
`Mask`	Logical matrix, which is the same size as the images, indicating which pixels should not be analysed during correlation.
`GaussFilt`	Define the standard deviation and window size for the Gaussian filter in pixels as $[\texttt{FiltSigma}, \texttt{FiltSize}]$, respectively, where $\{\texttt{FiltSigma} \in \mathbb{R}^+ \mid \texttt{FiltSigma} > 0\}$ and $\{\texttt{FiltSize} \in \mathbb{N}\}$.
`StepSize`	Step size in pixels $\{\texttt{StepSize} \in \mathbb{N}\}$.
`SubSize`	Subset size in pixels $\{\texttt{SubSize} = 2k+1 \mid k \in \mathbb{N}\}$.
`SubShape`	Subset shape $\{\texttt{SubShape} \in \texttt{'Square'}, \texttt{'Circle'}\}$.
`SFOrder`	Dictates the SF order $\{\texttt{SFOrder} \in \mathbb{Z} \mid 0 \leq \texttt{SFOrder} \leq 2\}$.
`RefStrat`	Logical statement dictating reference image strategy (Section 4.4.2).
`StopCritVal`	Defines the stopping criterion value $\{\texttt{StopCritVal} \in \mathbb{R}^+ \mid \texttt{StopCritVal} > 0\}$.
`WorldCTs`	Location of CTs in the world CS defined according to MATLAB's `estimateCameraParameters` function.
`ImgCTs`	Location of CTs in the sensor CS defined according to MATLAB's `estimateCameraParameters` function.
`rho`	Calibration plate thickness in millimetres.

Table 4.2: Accessing the output variables for image `d` (contained in `ProcData(d)`) and subset number `q`.

Variable	**Variable description**
`ImgName`	Image name.
`ImgSize(b)`	Image size (`b` = 1 for rows and `b` = 2 for columns).
`ImgFilt(b)`	Standard deviation (`b` = 1) and window size (`b` = 2) for the Gaussian filter, respectively, in pixels.
`SubSize(q)`	Subset size in pixels.
`SubShape(q)`	Subset shape.
`SFOrder(q)`	SF order.
`Xos(b,q)`	Reference subset position in the distorted sensor CS (`b` = 1 for x^o and `b` = 2 for y^o).
`Xow(b,q)`	Reference subset position in the world CS (`b` = 1 for $\hat{x}_w^o$ and `b` = 2 for $\hat{y}_w^o$).
`P(b,q)`	SFPs (`b` = 1 for u and `b` = 7 for v).
`C(q)`	ZNCC coefficient.
`Uw(b,q)`	Displacement in the world CS (`b` = 1 for $\hat{u}_w$ and `b` = 2 for $\hat{v}_w$).
`Iter(q)`	Number of iterations until stopping criterion is satisfied (maximum of 100 iterations).
`CamParams`	Calibration parameters.

4.4.1 `ADIC2D` function

`ADIC2D` is the main function and is outlined in Table 4.3. Its purpose is to set up the DIC problem and call the appropriate subroutines. `ADIC2D` defines variables on a per-image and subset basis to allow for complete flexibility in assigning `Xos`, `SubSize`, `SubShape` and `SFOrder`. Although `ADIC2D` is capable of this, it assigns the same `SubSize`, `SubShape` and `SFOrder` to each subset (in line 8 based on the inputs) since this is the most common use case. Output variables are pre-assigned in line 8 to allow for the collection of input data used and efficient storage of computed variables. Note that the SFPs are stored in a vector `P` which corresponds to the size of the second-order SFP vector in Equation (4.15). Thus the second-order SFPs of `P`, not used by the specified SF order, remain zero.

Table 4.3: `ADIC2D` algorithm summary.

Line numbers	Task performed
Lines 2-4	Compute image names, number of images and size of the first image;
Lines 5-6	Create regularly spaced reference subset positions, `Xos`;
Line 7	Remove subsets containing invalid pixels which are defined by `Mask`;
Line 8	Pre-assign `ProcData` structure;
Line 9	Call subroutine `ImgCorr` to perform image correlation;
Line 10	Call subroutine `CSTrans` to perform transformation from the distorted sensor CS to the world CS;

`ADIC2D` calls the subroutine `ImgCorr` to perform the image correlation as presented above. `ImgCorr`'s input variables are `n` (the total number of images in the set), the pre-assigned variables in `ProcData`, `FileNames`, `RefStrat` and `StopCritVal`. The output variables are `P`, `C`, `Iter` and `StopVal` which are stored in `ProcData`. The computed SFPs are then passed to `CSTrans` to transform displacements to the world CS. `CSTrans`'s input variables are `n`, `ProcData`, `WorldCTs`, `ImgCTs` and `rho`. The output variables are `Xow`, `Uw` and MATLAB's `CamParams` (containing the intrinsic, extrinsic, and radial distortion parameters) which are stored in `ProcData`. Note that within the subroutines `ProcData` is shortened to `PD`.

The presented framework assumes a constant, regularly spaced `Xos` defined using `StepSize` and `SubSize`. Subsets which contain pixels that `Mask` indicates should not be analysed are removed.

4.4.2 Correlation implementation

Correlation is performed using five subroutines: (i) `ImgCorr`, which performs the correlation on an image bases, i.e., between F and G; (ii) `SubCorr`, which performs the correlation on a subset basis; (iii) `SFExpressions`, which defines anonymous functions based on the SF order; (iv) `SubShapeExtract`, which determines input data for `SubCorr` based on the subset shape, size and position; and (v) `PCM`, which determines initial estimates for the displacement SFPs.

`SubCorr`'s input variables are the interpolation coefficients, f_i, ∇f_i, `SubSize`, `SFOrder`, `Xos`, Δx_i, initial estimates for $\boldsymbol{P}$ and `StopCritVal`. Note that throughout Section 4.4.2 variables with subscript i refer to the full set of this variable for a subset (i.e., ∇f_i refers to $\nabla f_i \forall i \in I$). `SubCorr`'s output variables are `P`, `C`, `Iter` and `StopVal`. `SFExpressions`'s input variable is `SFOrder` with outputs as anonymous functions to compute $\boldsymbol{W}$, $\nabla f_i \frac{\partial W_i}{\partial \boldsymbol{P}}$ and $\|\boldsymbol{P}\|$. Moreover, two functions are included to compute $\boldsymbol{\omega}$ (given in Equation (4.22)) and to extract the SFPs from $\boldsymbol{\omega}$.

The framework considers two subset shapes, square and circular, which are commonly employed in subset based DIC. For circular subsets `SubSize` defines the diameter of the subset. `SubShapeExtract` is used to determine $\boldsymbol{f_i}$, $\nabla \boldsymbol{f_i}$ and $\Delta \boldsymbol{x_i}$ for a subset based on the inputs `SubSize`, `SubShape`, `Xos`, F, ∇F and `SubExtract`. ∇F is the light intensity gradient of the entire reference image and `SubExtract` is an anonymous function, defined in line 2 of `ImgCorr`, which extracts a square subset from a matrix based on the position and size of the subset. `PCM` returns u and v based on inputs F, G, `SubSize`, `Xos` (passed as 2 vectors as required by `arrayfun`) and `SubExtract`.

Furthermore, two reference strategies are considered, namely, an absolute and an incremental strategy. The absolute strategy defines the first image as F (i.e. `FileNames(1)`), whereas the incremental strategy defines the previous image as F (`FileNames(d-1)`). The incremental strategy handles large deformations between images more reliably; however, if total displacements are required, it suffers from accumulative errors. The variable `RefStrat` is set to 0 or 1 for the absolute or incremental strategy, respectively. Alternate reference strategies may be set by modifying line 8 in `ImgCorr`.

Moreover, `ADIC2D` considers the zero, first and second-order SFs, as outlined in Section 4.3.2.2. Set SFOrder to 0, 1 or 2 for the zero, first and second-order SFs, respectively.

4.4.2.1 `ImgCorr` function

`ImgCorr` uses two nested for-loops as summarised in Table 4.4. The outer loop cycles through the image set, whereas the inner loop cycles through the subsets. `ImgCorr` reads the appropriate image pairs F and G from the image set, depending on the chosen reference strategy, and filters both using MATLAB's `imgaussfilt` function. Alternate image filters can be employed by modifying line 5 and 9. Bi-cubic b-spline interpolation coefficients are computed using MATLAB's `griddedInterpolant` function. Alternate interpolation methods can be set by either modifying line 6 by replacing 'spline' with 'linear' or 'cubic', or replacing it with an alternate interpolation algorithm, such as MATLAB's `spapi` function for higher order spline interpolation. `griddedInterpolant` was used for computational efficiency.

For an incremental strategy, `Xos` is displaced using the displacement SFPs from the previous correlation run, to track the same light intensity patterns within the reference subsets. These displacement SFPs are rounded, as suggested by Zhou et al. [222], such that the pixel positions of the reference subset have integer values and avoid the need for interpolating the reference subset.

Correlation of each subset requires SFP initial estimates. For the first run `ADIC2D` uses a phase correlation method (PCM) to determine initial estimates. Subsequent correlation runs use the previous correlation run's SFPs as an initial estimate. However, PCM is used for every run in the incremental strategy, as it allows for better stability if large displacements are expected. PCM can be used between each run by replacing line 15 with line 13. Moreover, alternate initial estimate strategies can be implemented by changing line 13. The PCM algorithm is discussed in Section 4.4.2.5.

Table 4.4: `ImgCorr` algorithm summary.

Line numbers	**Task performed**
Line 2	Define SubExtract function to extract square subset data;
Line 3	**for** image number `d` $=2$ to `d` $=$ `n` , **do**
Line 4	Define G;
Line 5	Perform Gaussian filtering on G using MATLAB's `imgaussfilt` function;
Line 6	Compute interpolation coefficients using MATLAB's `griddedInterpolant` function;
Line 7	**if** first image of correlation run or `Refstrat` is incremental, **do**
Line 8	Define F;
Line 9	Perform Gaussian filtering on F using MATLAB's `imgaussfilt` function;
Line 10	Compute gradients for F (compute ∇F);
Lines 11-12	Displace `Xos` with previous image correlation run displacements (incremental strategy);
Line 13	Call subroutine `PCM` to compute initial estimates of displacement SFPs;
Line 14	**else**, **do**
Line 15	Set `P(d)` $\leftarrow$ `P(d-1)` ;
Line 16	**end if**
Line 17	Initialise temporary storage variables used to save correlation information during the inner loop;
Line 18	**for** subset number `q` $=1$ to number of subsets, **do**
Line 19	Call subroutine `SubShapeExtract`;
Line 20	Call subroutine `SubCorr`;
Line 21	**end for**
Line 22	Save correlation information to `PD` variable;
Line 23-24	Show results for image `d` correlation ;
Line 25	**end for**

The inner loop correlates each subset by using `SubShapeExtract` to determine the data for a subset while `SubCorr` uses this data to perform correlation of the subset. The loop can be implemented using parallel processing to reduce computation time by changing line 18 to a parfor-loop. However during a parfor-loop the outputs of `SubCorr` cannot be saved directly to a structure variable. It is for this reason that they are saved to the temporary storage variables (initiated in line 17) during the loop and assigned to `PD` thereafter.

4.4.2.2 `SubShapeExtract` function

`SubShapeExtract` returns the data sets of f_i, ∇f_i and $\Delta \boldsymbol{x}_i$ for a subset based on its intended shape, size and position as outlined in Table 4.5. Note that these output data sets are in the form of vertical vectors. Alternative subset shapes can be added to this function provided they produce the same output data sets.

For a square subset `SubExtract` is used to extract the appropriate elements from the input matrices (F and ∇F) which correspond to the pixels of the subset. $\Delta \boldsymbol{x}_i$ is determined in line 7 according to `SubSize`.

Table 4.5: `SubShapeExtract` **algorithm summary.**

Line numbers	**Task performed**
Line 2	**switch** `SubShape`;
Line 3	**case** `SubShape`='`Square`' , **do**
Line 4-6	Extract f_i and ∇f_i using `SubExtract`;
Line 7	Compute Δx_i using `SubSize`;
Line 8	**case** `SubShape` = '`Circle`' , **do**
Line 9-11	Extract f_i and ∇f_i using `SubExtract`;
Line 12	Compute Δx_i using `SubSize`;
Line 13	Determine mask of elements that fall within the circular subset;
Line 14-16	Use mask to extract appropriate data for circular subset;
Line 17	**end switch**

For circular subsets the same process is followed. This results in temporary data sets f_i, ∇f_i and Δx_i which correspond to a square subset of size equal to the diameter of the intended circular subset. A mask identifying which elements, of these data sets, fall within the radius of the intended circular subset is computed in line 13 using Δx_i. This mask is used to extract the appropriate elements from the temporary data sets of the square subset resulting in the appropriate data sets for the circular subset.

4.4.2.3 `SubCorr` function

`SubCorr` is at the heart of `ADIC2D` and performs the subset based correlation, as summarised in Table 4.6. It follows the theoretical framework presented in Section 4.3.2.

Table 4.6: `SubCorr` **algorithm summary.**

Line numbers	**Task performed**
Line 2	Call `SFExpressions` to assign equations dependent on the SF order;
Line 3	Compute $\nabla f_i \frac{\partial W_i}{\partial P}$;
Line 4	Compute H^{-1}, Equation (4.20);
Line 5	Compute normalisation values $\bar{f}$ and $\tilde{f}$;
Line 6	Initialise `flag` $\leftarrow 0$, `iter` $\leftarrow 0$ and $\Delta P \leftarrow 1$;
Line 7	**while** `flag`$= 0$, **do**
Line 8	Compute $\Delta x_i'$, Equation (4.14), using estimates of P;
Line 9	Compute g using interpolation coefficients;
Line 10	Compute normalisation values $\bar{g}$ and $\tilde{g}$;
Line 11	Compute $\|\Delta P\|$ using Equation (4.23);
Line 12	**if** $\|\Delta P\| <$ `StopCritVal` or `iter` > 100, **do**
Line 13	Set `flag`$\leftarrow 1$;
Line 14	Compute C, Equation (4.11) substituted into Equation (4.13);
Line 15	**else, do**
Line 16	Compute J, Summation expression of Equation (4.19);
Line 17	Compute ΔP, Equation (4.19);
Line 18	Update P, Equation (4.21);
Line 19	**end if**
Line 20	Set `iter` $\leftarrow$ `iter` $+1$;
Line 21	**end while**

4.4.2.4 `SFExpressions` function

`SFExpressions` returns five anonymous functions based on the SF order specified and is outlined in Table 4.7. `W`, defines Equation (4.14), `dFdWdP` defines $\nabla f_i \frac{\partial W_i}{\partial P}$, `SFPVec2Mat` defines Equation (4.22), `Mat2SFPVec` extracts $\boldsymbol{P}$ from `SFPVec2Mat` and `StopCrit` defines Equation (4.23). Additional SFs, such as higher order polynomials, can be added after line 20 provided they are consistent with the outputs of `SFExpressions`.

Table 4.7: `SFExpressions` **algorithm summary.**

Line numbers	**Task performed**
Line 2	**switch** `SFOrder`
Line 3-8	**case** `SFOrder`$=0$, **do** assign functions for zero-order SF;
Line 9-14	**case** `SFOrder`$=1$, **do** assign functions for first-order SF;
Line 15-20	**case** `SFOrder`$=2$, **do** assign functions for second-order SF;
Line 21	**end switch**

4.4.2.5 `PCM` function

PCM performs correlation using the zero-order SF in the frequency domain to obtain initial displacement estimates. The algorithm is summarised in Table 4.8. `PCM` is efficient; however, it is limited to integer pixel displacements and can only use square subsets. Moreover, `PCM` is only capable of determining a reliable initial estimate if the displacement is less than half of `SubSize`. For more information on PCM refer to the work of Foroosh et al. [223].

Table 4.8: `PCM` **algorithm summary.**

Line numbers	**Task performed**
Line 2	Compute normalised cross-power spectrum in the frequency domain;
Line 3	Convert back to spatial domain;
Line 4	Find index of the maximum correlation coefficient;
Line 5	Compute index vector which relates indices of the correlation coefficient matrix to the displacements they correspond to;
Line 6-7	Obtain displacements using index of the maximum correlation coefficient;

4.4.3 `CSTrans` function

`CSTrans` performs CS and displacement transformations from the distorted sensor CS to the world CS as outlined in Table 4.9. `CSTrans` uses MATLAB's image calibration toolbox to determine calibration parameters according to Section 4.3.1 which are used to perform the transformations detailed in Section 4.3.3. Note that the extrinsic calibration parameters, extracted in line 8, are based on the final set of CTs in the sensor CS (`ImgCTs(:,:,end)`). Alternate calibration algorithms may be implemented by replacing lines 13 and 14.

Table 4.9: `CSTrans` algorithm summary.

Line numbers	Task performed
Line 2	**if** calibration targets are given, **do**
Line 3	Compute calibration parameters using MATLAB's `estimateCameraParameters` function;
Line 4	**else**, **do**
Line 5	Set unit calibration parameters and pass to MATLAB's `cameraParameters` function;
Line 6	Assign CTs in the distorted sensor and world CSs using `Xos`;
Line 7	**end if**
Line 8	Extract appropriate extrinsic parameters;
Line 9	Compute T_{spec}, Equation (4.6);
Line 10	**for** image number `d`= 1 to `d=n` , **do**
Lines 11-12	Compute x_d, Equation (4.24);
Line 13-14	Compute x_w^o and x_w^d using MATLAB's `undistortPoints` and `pointsToWorld` functions;
Line 15	Compute $\hat{u}_w$ and $\hat{v}_w$, Equation (4.26);
Line 16	Save calibration parameters;
Line 17	**end for**

4.5 Validation

ADIC2D was validated using the 2D DIC Challenge image sets that were created using TexGen [224] or Fourier methods [225] as documented by Reu et al. [122]. Homoscedastic Gaussian noise was applied to each image set to simulate camera noise. As stated by Reu et al. [122]: *image noise is specified as one standard deviation of the grey level applied independently to each pixel*. The respective noise levels are listed in Table 4.10. Samples 1-3 contain rigid body translations (RBTs) to assess the performance of the ADIC2D framework in the *ultimate error regime* [184]. This type of analysis aims to highlight the errors caused by contrast and noise, in the absence of complex displacement fields, interacting with the numerical processes of correlation [122, 17]. Sample 14 contains a sinusoidal displacement field with increasing frequency. This type of analysis aims to highlight the compromise between noise suppression and spatial resolution (SR) [122].

Table 4.10: Details for the samples of the 2D DIC Challenge [122].

Name	Method	Noise	Contrast	Images	Displacement field
Sample 1	TexGen	1.5	Varying	21	Shift of 0.05 pixels in both directions per-image
Sample 2	TexGen	8	0-50	21	Shift of 0.05 pixels in both directions per-image
Sample 3	Fourier	1.5	0-200	12	Shift of 0.1 pixels in both directions per-image
Sample 14	Fourier	5	0-200	4	Sinusoid in the x-direction of increasing frequency (amplitude 0.1 pixels)

CS transformations were not performed during the validation process, by setting `WorldCTs` $= 0$, `ImageCTs` $= 0$ and `rho` $= 0$. A stopping criterion of `StopCritVal` $= 10^{-4}$, limited to 100 iterations per-subset (line 12 in `SubCorr`), was used. The Gaussian image filter was set to `FiltSize` $= 5$ as this offers the best compromise between reducing bias and avoiding increasing variance [114]. `FiltSigma` is specified on a per-sample basis.

4.5.1 Quantifying error

Bias, variance, root-mean square error (RMSE) and SR were used to quantify errors. Bias refers to the mean of the absolute error (MAE_u, MAE_v) between the correlated and true values, while variance refers to the standard deviation of the absolute error (σ_u, σ_v). These are computed as

$$MAE_u = \frac{\sum_{q=1}^{Q} |u_q^{calc} - u_q^{true}|}{Q} \qquad MAE_v = \frac{\sum_{q=1}^{Q} |v_q^{calc} - v_q^{true}|}{Q} \tag{4.27}$$

$$\sigma_u=\sqrt{\frac{\sum_{q=1}^{Q}(|u_q^{calc}-u_q^{true}|-MAE_u)^2}{Q-1}} \quad \text{and} \quad \sigma_v=\sqrt{\frac{\sum_{q=1}^{Q}(|v_q^{calc}-v_q^{true}|-MAE_v)^2}{Q-1}} \tag{4.28}$$

where u_q^{calc} and v_q^{calc} are the correlated, u_q^{true} and v_q^{true} the true displacements in the x- and y-direction, respectively and Q is total number of subsets. Bornert et al. [184] introduced a RMSE which summarises the full-field displacement errors as a single number calculated as

$$RMSE_u=\sqrt{\frac{\sum_{q=1}^{Q}\left(u_q^{calc}-u_q^{true}\right)^2}{Q}} \quad \text{and} \quad RMSE_v=\sqrt{\frac{\sum_{q=1}^{Q}\left(v_q^{calc}-v_q^{true}\right)^2}{Q}} \tag{4.29}$$

Strain bias, variance and RMSE are calculated in the same way. SR is defined as the highest frequency of a sinusoidal displacement field at which the code is capable of capturing the peak displacements and strains within 95% and 90% of the true values, respectively [122]. SR is reported as the period such that lower values indicate better performance across all error metrics.

4.5.2 Samples 1-3

Samples 1-3 were correlated using ADIC2D, Blaber's Ncorr (version 1.2) and LaVision's DaVis (version 8.4). Ncorr was used as it is well-established [226, 227] and its correlation process is similar in theory to ADIC2D with the exception that it uses bi-quintic b-spline interpolation and the reliability-guided displacement tracking (RGDT) strategy proposed by Pan [182]. DaVis uses bi-sextic b-spline interpolation and was included to compare ADIC2D to a commercial software package.

ADIC2D was used by setting `SubShape='Circle'`, `StepSize=5` pixels, `SFOrder=1` and `SubSize` to 21, 41 and 81 pixels. A `FiltSigma` of 0.4, 0.8 and 0.6 was used for Samples 1, 2 and 3, respectively. Subsets of size 21, 41 and 81 pixels had 5000, 4500 and 3650 subsets per-image, respectively.

The following procedure is used to determine the error metrics for each sample on a per-algorithm and per-subset basis: (i) the displacement errors in the x- and y-direction were computed for each subset; (ii) these displacement errors were used to determine the error metrics in each direction for all subsets throughout the image set; and (iii) Pythagoras' theorem was used to determine the magnitude of each error metric, in pixels (pix), which is reported in Tables 4.11-4.13 for Samples 1-3, respectively.

Table 4.11: Sample 1 error analysis reported at $\times 10^{-3}$ pix.

Code	Subset size 21			Subset size 41			Subset size 81		
	Bias	Var[13]	RMSE	Bias	Var	RMSE	Bias	Var	RMSE
ADIC2D	2.32	2.59	3.48	1.08	0.88	1.4	0.634	0.481	0.796
Ncorr	2.31	2.63	3.5	0.995	0.822	1.29	0.504	0.402	0.645
DaVis	1.34	1.05	1.7	0.652	0.497	0.819	0.317	0.246	0.401

Table 4.12: Sample 2 error analysis reported at $\times 10^{-3}$ pix.

Code	Subset size 21			Subset size 41			Subset size 81		
	Bias	Var[13]	RMSE	Bias	Var	RMSE	Bias	Var	RMSE
ADIC2D	3.49	4.24	5.5	1.16	0.917	1.48	0.519	0.391	0.65
Ncorr	5.59	5.96	8.17	2.05	1.71	2.67	0.956	0.759	1.22
DaVis	1.84	1.57	2.42	0.781	0.599	0.985	0.377	0.281	0.47

Table 4.13: Sample 3 error analysis reported at $\times 10^{-3}$ pix.

Code	Subset size 21			Subset size 41			Subset size 81		
	Bias	Var[13]	RMSE	Bias	Var	RMSE	Bias	Var	RMSE
ADIC2D	9.75	8.17	12.7	3.77	2.9	4.76	1.87	1.4	2.34
Ncorr	9.18	7.58	11.9	3.97	3.05	5.01	2.36	1.7	2.91
DaVis	5.16	4.02	6.54	2.47	1.9	3.12	1.29	1	1.64

Sample 1 reflects robustness to contrast changes, Sample 2 reflects robustness to higher noise content with a limited contrast range and Sample 3 reflects effects due to the interpolation method used.

ADIC2D's performance for Sample 1 is similar to that of Ncorr (up to 26% higher error). It is noted that ADIC2D does not use the RGDT strategy and therefore is somewhat more susceptible to contrast changes compared to Ncorr.

The improved performance of ADIC2D over Ncorr for Sample 2 (up to 48% improvement) is reasoned to be due to the SF order used to obtain an initial estimate of the SFPs. ADIC2D uses the zero-order SF resulting in reliable estimates of the SFPs. In contrast, Ncorr uses an overmatched first-order SF which causes estimates to take on spurious values, due to noise, causing correlation to proceed along an unfavourable solution path. Moreover, DaVis removes displacement results with poor correlation coefficients thereby not providing a true reflection of the overall error for Sample 2.

Sample 3 highlights the effect of lower order b-spline interpolation for smaller subsets. ADIC2D uses lower order b-spline interpolation, in comparison to Ncorr

---

[13] Shorthand for variance

and DaVis, resulting in less accurate matching for smaller subsets (ADIC2D for `SubSize`=21 has a 7% higher RMSE relative to Ncorr).

4.5.3 Sample 14

ADIC2D was used for Sample 14 by setting `SubShape='Square'`, `StepSize`=5 pixels, `SFOrder`=1, `FiltSigma`=0.4 and `SubSize` to 25, 31, 51 and 71 pixels. A window of 9 measurement points was used to compute strain data. Table 4.14 shows the displacement and strain results in the x-direction, for the last image of the set, that were analysed using the MATLAB code provided by the DIC Challenge [122]. Codes A and G published in [122], which exhibit the best noise suppression (variance) and SR, respectively, are included for comparison. Subsets of size 25, 31, 51 and 71 pixels had 43700, 43700, 42600 and 40600 subsets per image.

Table 4.14: Sample 14 error analysis in the x-direction for last image of the image set [122].

Code/ subset size	**Displacement**				**Strain**			
	RMSE (pix)	**Var**[13] (pix)	**Max Bias** (pix)	**SR** (pix)	**RMSE** (pix/pix)	**Var** (pix/pix)	**Max Bias** (pix/pix)	**SR** (pix)
Code G	0.010	0.010	0.012	100	453	429	923	74
ADIC2D 25	0.018	0.017	0.015	1629	598	406	1488	173
ADIC2D 31	0.014	0.013	0.017	160	600	335	1674	182
ADIC2D 51	0.014	0.007	0.033	257	839	193	2720	233
ADIC2D 71	0.022	0.005	0.059	354	1255	125	4412	294
Code A	0.022	0.005	0.056	716	1131	172	3399	410

ADIC2D is capable of dealing with high frequency displacement fields. For a subset size of 71 pixels ADIC2D performs similarly to code A (within 0.1% difference) with the exception of an improved SR (51%) and higher maximum bias (5%). As the subset size decreases so does the RMSE, bias and SR while variance increases. For `SubSize`=25 pixels, the error metrics increase (except strain SR as illustrated in Figure 4.6(b)), indicating a limitation of ADIC2D with regards to noise suppression and SR for smaller subset sizes (as shown in Figure 4.6(a)). The strain SR does not increase because strain experiences more spatial filtering than displacement for the reasons outlined in the DIC Challenge paper [122]. Although ADIC2D cannot achieve results similar to code G, the results in Table 4.14 indicate that the noise suppression and SR are within the range of established DIC codes evaluated in the DIC Challenge paper [122].

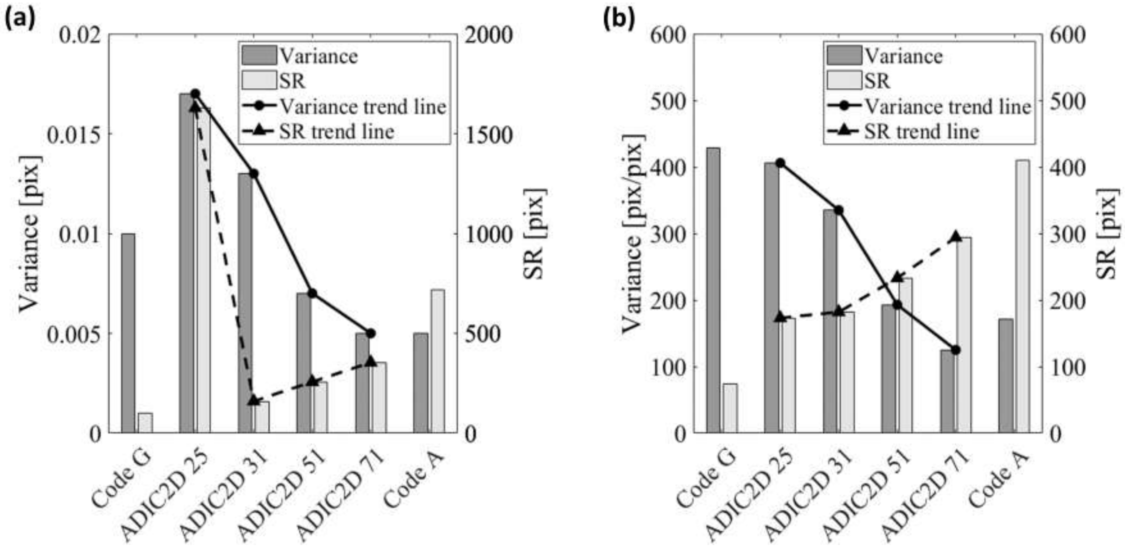


Figure 4.6: Comparison of the noise suppression (variance) and SR for various subset sizes in terms of: (a) displacement; and (b) strain results.

4.6 Discussion

The code was designed with modularity in mind. Firstly, it is modular in that each main task is performed by a separate subroutine such that the reader can progressively build up their understanding of the overall code by considering individual subroutines. This is particularly evident for the correlation subroutines which separates aspects of the correlation process such that the logistics of preparing data for correlation (`ImgCorr`), the core correlation operations (`SubCorr`), the effect of different SF orders on correlation (`SFExpressions`), how data is prepared for different subset shapes (`SubShapeExtract`) and determining initial estimates of the SFPs (`PCM`) can be considered separately.

Secondly, the code allows for changing of the SF order, subset shape, interpolation method and Gaussian filtering parameters. Although the effect of these on the displacement and strain results is well documented [12, 114, 115], this code allows the reader to easily investigate the effect of these in a practical manner.

Thirdly, the code is modular in that it allows the subset size, subset shape and SF order to be assigned on a per-subset and per-image basis. Traditionally DIC makes use of a single subset size, subset shape and SF order for all subsets across all images. However, there has been a growing interest in the field of DIC to create algorithms which adaptively assign these parameters such that they are the most appropriate for the displacement and speckle pattern that the subset is attempting to track resulting in more reliable displacements being computed. The modularity of ADIC2D means it is straightforward to couple it with such an adaptive strategy.

In order to keep the code simple two aspects were neglected that would have otherwise made the correlation aspect of ADIC2D consistent with the current state-of-the-art as identified by Pan [22]. Firstly, ADIC2D makes use of bi-cubic b-spline interpolation as opposed to the recommended bi-quintic b-spline interpolation. As stated in the work of Bornert et al. [184] the errors in the *ultimate error regime* are reduced by increasing the degree of the interpolation method particularly for smaller subsets. This is reflected in Table 4.13 which shows that, although the error metrics of ADIC2D are better than that of Ncorr for larger subsets, the opposite is true for the subset size of 21 pixels.

Secondly, ADIC2D does not use the RGDT strategy. While ADIC2D uses the optimal SFPs of a subset for the previous image pair as an initial estimate of the SFPs for the current image pair, RGDT only does this for the subset with the best correlation coefficient for the previous image pair. It then uses the SFPs of this subset, for the current image pair, as initial estimates to correlate its neighbouring subsets. It then repeatedly identifies the subset with the best correlation coefficient, which has neighbouring subsets which have not yet been correlated, and uses its SFPs to correlate its neighbouring subsets. This is repeated until the all the subsets have been correlated for current image pair.

Thus ADIC2D is susceptible to propagating spurious SFPs of a subset through the image set which the RGDT strategy would have avoided. The effect of this is reflected in the results of Table 4.11 which shows how ADIC2D struggles to perform as consistently as Ncorr in the presence of contrast changes in the image set.

Despite this, ADIC2D performs on par with established DIC algorithms. More specifically: (i) it is capable of dealing with contrast changes as shown in Table 4.11; (ii) it handles high levels of noise within the images sufficiently well as reflected in the results of Table 4.12; (iii) although displacement results of smaller subsets suffer due to its lower order bi-cubic b-spline interpolation, its interpolation method is sufficient achieving results similar to Ncorr as show in Table 4.13; and (iv) it has noise suppression and SR characteristics that fall within the range of those reported for established DIC algorithms as shown in Figure 4.6.

Thus ADIC2D can be considered sufficiently reliable for use in the field of experimental solid mechanics. However, ADIC2D is not limited to this field since its modularity means it can be easily adapted for various applications and specific use cases. Furthermore, validation of ADIC2D coupled with its modularity not only makes it attractive as a learning resource but also as a starting point to develop the capabilities of DIC.

4.7 Conclusion

This paper presents the theory of a 2D, subset based DIC framework (ADIC2D) that is predominantly consistent with current state-of-the-art techniques, and illustrates its numerical implementation in 117 lines of MATLAB code. ADIC2D allows for complete flexibility in assigning correlation attributes on a per-image and per-subset basis. ADIC2D includes: Gaussian image filtering parameters; square or circular subset shape selection; zero, first and second-order SFs; reference image strategy selection; interpolation method flexibility; image calibration using MATLAB's image calibration toolbox and displacement transformation. Moreover, the presented code is modular. Sections of the framework can readily be changed enabling the reader to gain a better understanding of DIC as well as to contribute to the development of new DIC algorithm capabilities. Validation of ADIC2D shows that it performs on par with established DIC algorithms.

4.8 Post-submission discussion

Following the submission of the manuscript, areas of improvement and insights were identified during the peer review process. Despite the framework allowing for the choice of circular or square subsets, the effect of this choice on the displacement accuracy was not discussed. Although the effect of the subset shape is subtle, it is worth considering since the displacement computed for a subset is the average of that experienced by the pixels contained within the subset. For complex displacement fields the displacements experienced by pixels located farther from the subset's centre are less representative of that occurring at the subset's centre. Consequently, circular subsets have become favoured since their pixels are evenly distributed in a radially symmetric manner about the subset centre. Nevertheless, many DIC algorithms still employ square subsets since this trade-off is inconsequential in most applications and square subsets are convenient from a mathematical and programming perspective. A similar discussion was included within the manuscript for completeness.

Validation of the proposed DIC framework focuses on presenting numerical results in terms of displacement errors achieved relative to well-established DIC algorithms. Although this is useful to quantitatively establish the validity of the framework, a visual representation of the computed displacements would be beneficial in aiding intuitive understanding of how the framework performs. To this end a visual representation of the displacements computed by the framework for Sample 14 of the 2D DIC Challenge, as shown in Figure 4.7, was included in the manuscript post-review. This illustrates how increasing the subset size from 25 to 51 pixels greatly improves noise suppression in the computed displacements.

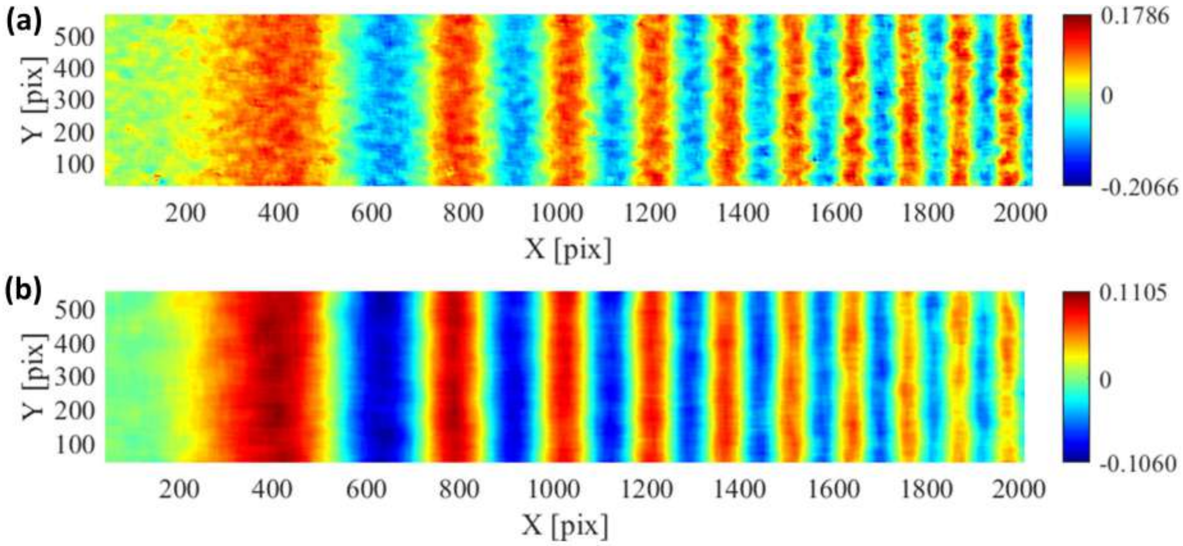


Figure 4.7: Comparison of the x-displacement for Sample 14 of the 2D DIC Challenge for a subset size of: (a) 25 pixels; and (b) 51 pixels.

Following the publication of the paper, the author was contacted by a fellow postgraduate student who was making use of the framework to investigate methods of reducing the impact of heat waves on DIC measurements. They pointed out that the image gradient computed via MATLAB's `imagradient` function needs to be divided by six such that it is normalised. A new repository[14] for the corrected version of the framework was created accordingly (such that the original repository remains consistent with the published paper).

Lastly, post-publication the author realised two typographical errors in the paper. c_s of Equation (4.1) should be set as c_s=0 instead of c_s=1. Additionally, line 12 of function `SubShapeExtract` should read "`"[dX,dY]=meshgrid(-(SubSize-1)/2:(SubSize-1)/2,-(SubSize-1)/2:(SubSize1)/2);"` instead of "`[dX,dY]=meshgrid(-(SubSize-1)/2:(SubSize-1)/2;`". A correction paper was published accordingly to inform all interested readers as well as apologise for any inconvenience caused.

---

[14] The updated framework can be found at: https://github.com/SUMatEng/ADIC2D_Updated

Chapter 5
Stereo DIC in MATLAB

This chapter extends the image correlation framework developed in Chapter 4 to an open-source stereo DIC framework, in the form of a 202 line MATLAB code[15], to address Objective 2.

This chapter is published in the *MDPI Applied Sciences* journal [228][16] under the title *Stereo Digital Image Correlation in MATLAB* and has been presented here in its pre-review form. A discussion of post-review considerations concludes this chapter. This work, along with the work of Chapter 4, has been presented on at 15th International Conference on Advances in Experimental Mechanics hosted by the British Society for Strain Measurement in 2021[17].

Declaration by candidate:

With regards to Chapter 5, the contributions of authors and co-authors are as follows:

Name	Email	Contribution	Extent (%)
D Atkinson (candidate)	17732913 @sun.ac.za	Conceptualisation, framework development and implementation, DIC processing, analysis of DIC results, figure preparation and visualisation, writing and compilation of manuscript	90
TH Becker	thorsten.becker @uct.ac.za	Beyond supervision and reviewing contributions, assisted in conception and development of the framework	10

Signature of candidate: [DJA] (see Footnote 10 on page 40)
Date: 2022/11/07

Declaration by co-authors: The undersigned hereby confirm that:

1. The declaration above accurately reflects the nature and extent of the contribution of the candidate and the co-authors to Chapter 5,

2. No other authors contributed to Chapter 5 besides those specified above, and

---

[15] Code available at: https://github.com/SUMatEng/ADIC3D

[16] DOI link: https://doi.org/10.3390/app11114904

[17] Link to the extended abstract: https://www.bssm.org/media/3958/2d-and-stereo-digital-image-correlation-written-in-matlab.pdf

3. Potential conflicts of interest have been revealed to all interested parties and that the necessary arrangements have been made to use the material in Chapter 5 of this dissertation.

Signature (see Footnote 10)	**Institutional affiliation**	**Date**
[TH Becker]	UCT	2022/11/07

5.1 Nomenclature

a_1 through a_6	Affine transformation parameters
α_j	Homogeneous scaling variable of the j^{th} camera
$\boldsymbol{B}$	Fundamental matrix
$\boldsymbol{B}_1$	Fundamental matrix after translation of polynomial triangulation method
$\boldsymbol{B}_2$	Fundamental matrix after rotation of polynomial triangulation method
β	Gaussian filtering window size
c_{x_j} and c_{y_j}	Translation applied to sensor coordinate system of the j^{th} camera
c_{s_j}	Skew of ideal sensor coordinate system of the j^{th} camera
C_{ZNSSD}	Zero-mean normalised sum of squared difference correlation criterion
C_{ZNCC}	Zero-mean normalised cross correlation criterion
C_{ObjFun}	Objective function of correspondence problem
D	Geometric error cost function
δ	Function to determine the Euclidean distance between two points
$\underline{\boldsymbol{e}}_{s_j}=\begin{bmatrix} e^1_{s_j} & e^2_{s_j} & e^3_{s_j} \end{bmatrix}^T$	Epipole of the j^{th} camera
E_{proj}	Total projection error
E_k^{dist}	Squared error distance of k^{th} keypoint
F	Reference image
$\boldsymbol{f}$	Reference subset
f_i	Light intensity value of pixel i of reference subset
$\bar{f}$	Mean light intensity value of reference subset
$\tilde{f}$	Normalisation function of reference subset
$\nabla f_i=\begin{bmatrix} \frac{\partial f_i}{\partial x} & \frac{\partial f_i}{\partial y} \end{bmatrix}$	Light intensity gradient of reference subset for pixel i
G	Deformed image
$\boldsymbol{g}$	Investigated subset
g_i	Light intensity value of pixel i of investigated subset
$\bar{g}$	Mean light intensity value of investigated subset
$\tilde{g}$	Normalisation function of investigated subset
$\boldsymbol{H}$	Hessian of optimisation equation
I	Number of pixels within a subset

$\boldsymbol{I}$	3×3 identity matrix
i	Subscript indicating that a variable is associated with the i^{th} pixel of a subset
$\boldsymbol{J}$	Jacobian of optimisation equation
j	Subscript indicating that a variable is associated with the j^{th} camera
$\boldsymbol{K}_j$	Intrinsic parameter matrix of the j^{th} camera
K	Number of keypoints under consideration
κ_j^1 and κ_j^2	Radial distortion parameters of the j^{th} camera
L	Number of calibration targets within a calibration image series
$\boldsymbol{\lambda}_j$	Epipolar line of the j^{th} camera
$\boldsymbol{M}$	3×3 matrix
$\boldsymbol{m}$	3×1 vector
$[\boldsymbol{m}]_\times$	Skew symmetric matrix of $\boldsymbol{m}$
MAE	Mean of the absolute error
$\hat{\mu}_{w_q}^{true}$	True displacement magnitude
$\hat{\mu}_{w_q}^{calc}$	Calculated displacement magnitude
$\boldsymbol{\Omega}$	Homography applied to projection matrices
$\boldsymbol{\omega}$	Function to populate a square matrix with the shape function parameters
$\boldsymbol{P}$	Shape function parameters
$\Delta\boldsymbol{P}$	Iterative improvement of shape function parameters
$\boldsymbol{P}_{update}$	Updated shape function parameters
$\|\|\Delta\boldsymbol{P}\|\|$	Stopping criterion
Φ	Function representing conversion of coordinate from projective space to Euclidean space
Ψ	Function representing the triangulation methods
$\varphi_1 - \varphi_4$	Elements of the fundamental matrix for the polynomial triangulation method
$\boldsymbol{Q}_j$	Projection matrix of the j^{th} camera
$\boldsymbol{Q}_{c_j}$	Projection matrix of the j^{th} camera in canonical form
$\boldsymbol{q}_{c_j}^n$	The n^{th} row of the canonical projection matrix of the j^{th} camera
Q	Number of subset pairs per image pair
$\boldsymbol{R}_j$	Rotation matrix for pinhole camera model of the j^{th} camera
$\boldsymbol{R}_j^{pm}$	Rotation matrix of the j^{th} camera for the polynomial triangulation method
$\boldsymbol{S}$	Matrix for the linear triangulation method
σ^g	Standard deviation of Gaussian function
σ	Standard deviation of the absolute error
$\boldsymbol{T}_j$	Translation vector for pinhole camera model of the j^{th} camera
$\boldsymbol{T}_j^{pm}$	Translation matrix of the j^{th} camera for polynomial triangulation method

t	Variable for parametrization of epipolar lines
τ	Squared error distance threshold
u	Displacement in x-direction in the distorted sensor coordinate system
$u_x, u_{xx}, u_y, u_{yy}, u_{xy}$	Derivatives of the x-direction displacement
$\hat{u}_w$	Displacement in x-direction in the world coordinate system
$\boldsymbol{V}_j$	Extrinsic parameter matrix of the j^{th} camera
v	Displacement in y-direction in the distorted sensor coordinate system
$v_x, v_{xx}, v_y, v_{yy}, v_{xy}$	Derivatives of the y-direction displacement
$\hat{v}_w$	Displacement in y-direction in the world coordinate system
$\boldsymbol{W}$	Shape function
$\frac{\partial \boldsymbol{W}_i}{\partial \boldsymbol{P}}$	Jacobian of the shape function, in terms of the shape function parameters, for pixel i
$\hat{w}_w$	Displacement in z-direction in the world coordinate system
$\hat{\boldsymbol{x}}_w = \begin{bmatrix} \hat{x}_w & \hat{y}_w & \hat{z}_w \end{bmatrix}^T$	Ideal coordinate in the world coordinate system
$\hat{\boldsymbol{x}}_{s_j} = \begin{bmatrix} \hat{x}_{s_j} & \hat{y}_{s_j} \end{bmatrix}^T$	Ideal coordinate in the ideal sensor coordinate system of the j^{th} camera
$\hat{\boldsymbol{x}}_{n_j} = \begin{bmatrix} \hat{x}_{n_j} & \hat{y}_{n_j} \end{bmatrix}^T$	Normalised ideal image coordinates of the j^{th} camera
$\boldsymbol{x}_{n_j} = \begin{bmatrix} x_{n_j} & y_{n_j} \end{bmatrix}^T$	Normalised distorted image coordinates of the j^{th} camera
$\boldsymbol{x}_{s_j} = \begin{bmatrix} x_{s_j} & y_{s_j} \end{bmatrix}^T$	Coordinate in the distorted sensor coordinate system of the j^{th} camera
$\boldsymbol{x}_l^{true} = \begin{bmatrix} x_l^{true} & y_l^{true} \end{bmatrix}^T$	True location of l^{th} calibration target in distorted sensor coordinate system
$\boldsymbol{x}_l^{calc} = \begin{bmatrix} x_l^{calc} & y_l^{calc} \end{bmatrix}^T$	Location of l^{th} calibration target in distorted sensor coordinate system predicted by the camera model
$\boldsymbol{x}_i = \begin{bmatrix} x_i & y_i \end{bmatrix}^T$	Pixel position of the i^{th} pixel of the reference subset in distorted sensor coordinate system
$\boldsymbol{x}^o = \begin{bmatrix} x^o & y^o \end{bmatrix}^T$	Centre of reference subset in distorted sensor coordinate system
$\Delta\boldsymbol{x}_i = \begin{bmatrix} \Delta x_i & \Delta y_i \end{bmatrix}^T$	Distance from the reference subset centre to i^{th} pixel position of reference subset
$\Delta\boldsymbol{x}'_i = \begin{bmatrix} \Delta x'_i & \Delta y'_i \end{bmatrix}^T$	Distance from the reference subset centre to i^{th} pixel position of investigated subset
$\boldsymbol{x}'_i = \begin{bmatrix} x'_i & y'_i \end{bmatrix}^T$	i^{th} pixel position of the investigated subset in the distorted sensor coordinate system
$\boldsymbol{x}^o_{s_j} = \begin{bmatrix} x^o_{s_j} & y^o_{s_j} \end{bmatrix}^T$	Reference subset position in distorted sensor coordinate system of the j^{th} camera
$\boldsymbol{x}^d_{s_j} = \begin{bmatrix} x^d_{s_j} & y^d_{s_j} \end{bmatrix}^T$	Investigated subset position in the distorted sensor coordinate system of the j^{th} camera

$\check{x}^o_{s_j} = \left[\check{x}^o_{s_j} \;\; \check{y}^o_{s_j}\right]^T$	Measured position of the reference subset in the ideal sensor coordinate system of the j^{th} camera
$\check{x}^d_{s_j} = \left[\check{x}^d_{s_j} \;\; \check{y}^d_{s_j}\right]^T$	Measured position of the investigated subset in the ideal sensor coordinate system of the j^{th} camera
$\check{x}_{s_j} = \left[\check{x}_{s_j} \;\; \check{y}_{s_j}\right]^T$	Measured coordinate in the ideal sensor coordinate system of the j^{th} camera
$\hat{x}^o_w = \left[\hat{x}^o_w \;\; \hat{y}^o_w \;\; \hat{z}^o_w\right]^T$	Position of reference subset pair in the world coordinate system
$\hat{x}^d_w = \left[\hat{x}^d_w \;\; \hat{y}^d_w \;\; \hat{z}^d_w\right]^T$	Position of investigated subset pair in the world coordinate system
$x^k_{s_j} = \left[x^k_{s_j} \;\; y^k_{s_j}\right]^T$	k^{th} keypoint location in the first image of the image series of the j^{th} camera
ξ_{x_j} and ξ_{y_j}	Scaling of metric units to pixels for the j^{th} camera
ζ	Maximum distance, along a single axis, from the centre of the reference subset to a pixel in the reference subset

5.2 Introduction

Digital image correlation (DIC) is an optical metrology technique capable of determining full-field displacements and deformations experienced by a body from images captured of the body's surface. DIC, being a non-interferometric technique, is more robust to vibrations and ambient light variations than interferometric techniques, such as Electronic Speckle Pattern Interferometry and Moiré interferometry, allowing it to be used outside the confines of a laboratory. Additionally, its non-contact, full-field nature, its ability to function at a range of scales (from scanning electron microscopy images [229, 230] to satellite images [231]), and rapid improvement of cameras has resulted in it becoming one of the most favoured techniques in experimental solid mechanics.

Although DIC has received the most attention and thus development within experimental solid mechanics, it has found widespread use across many fields. Some applications of DIC include: (i) investigating mechanical properties of biological tissues [196, 195, 54, 194, 232]; (ii) deflection measurement and structural health monitoring of bridges [233, 234, 235, 107, 236, 237, 238]; (iii) investigating cracks in masonry structures [239, 240, 241]; (iv) landslide monitoring [242, 243, 187, 244]; (v) determining material properties of woven fabrics (such as cotton) [245, 246]; vibrational analysis of components [247, 248, 249]; and (vii) in situ damage detection of components and structures [250, 197, 199], to name a few.

Following the introduction of two-dimensional (2D) DIC in 1982 by Peters and Ranson [207], it was extended to three dimensions (stereo DIC) by Luo et al. [208] in 1993, and to digital volume correlation (DVC) by Bay et al. [28] in 1999. The correlation aspect of DIC has undergone significant refinement over the years. Current state-of-the-art DIC algorithms, as identified by Pan [22], make use of: (i) bi-quintic b-spline interpolation which was shown by Schreier et al. [12] to produce the most accurate sub-pixel displacements; (ii) the first and second-order shape functions (SFs) proposed by Peters et al. [9] and Lu and Carry [13] respectively; (iii) the zero-mean normalised sum of squared difference (ZNSSD) correlation criterion suggested by Tong [14]; (iv) Gaussian pre-filtering of images to reduce bias in the displacements caused by high frequency noise in the images [114]; and (v) the inverse compositional Gauss-Newton (IC-GN) optimisation method introduced by Baker and Matthews [15]. Although the IC-GN optimisation method is theoretically equivalent to the Newton-Raphson (NR) method [15], in practice it has greater efficiency, accuracy and robustness to noise [211] making it preferable.

Although DIC is a complex process, the papers proposed by Pan et al. [209], Gao et al. [210], Solav et al. [251] and Blaber et al. [166] provide an in-depth explanation of the theory of the DIC process, which coupled with the Good Practices Guide for DIC [8], enables newcomers to the field of DIC to gain the necessary knowledge to effectively apply it in established use cases. However, in order for a newcomer to contribute to the field of DIC, by developing new algorithms which improve its capabilities or adapting it for new applications, requires gaining a deep understanding of DIC. Gaining such a deep understanding is cumbersome due to the lack of resources which directly bridge the gap between the theory of DIC and its coded implementation. More specifically, although some publications release code that is consistent with the theory presented [209, 210, 251], these codes focus on robustness and ease of use which, despite making them more suitable for real world applications, results in a complex code which is ineffective as a learning resource. This lack of resources acts as a barrier to newcomers, intending to further the capabilities of DIC, thereby limiting the development rate of the field.

It is for this reason that the authors published a paper bridging the gap between the theory of a 2D DIC framework (ADIC2D) and its practical implementation as a modular 117 line MATLAB code [191]. However, 2D DIC, being limited to determining in-plane displacements is susceptible to errors in the presence of out-of-plane motion, which is generally unavoidable especially outside the confines of a laboratory [109]. As such, stereo DIC has become favoured since it accounts for out-of-plane motion allowing for more reliable use across a broader range of applications and fields. This is supported by the significantly larger number of publications of stereo DIC relative to 2D DIC.

This paper builds upon the work of the preceding paper by extending ADIC2D to stereo DIC with the aim of bridging the gap between the theory and implementation of stereo DIC. This is done by presenting the theory of a subset-based, stereo DIC framework that is predominantly consistent with current state-of-the-art techniques. How this theory is implemented as a modular 202 line MATLAB code is then discussed. These theory and implementation sections focus on aspects of stereo DIC which differ from that of ADIC2D. Thereafter the framework is validated using image sets provided by the stereo DIC Challenge [123]. More specifically, ADIC3D's results are compared to those of LaVision's StrainMaster software and Digital Image Correlation engine (DICe) [167] in order to make observations about its capabilities. Finally, the proposed framework's modularity, limitations and overall performance is discussed.

The framework, referred to as ADIC3D, was developed in MATLAB since its simple syntax lets the reader focus on the core mathematics of the code. Furthermore, MATLAB's efficient built-in functions are leveraged to simplify the code and reduce its computation time. The code is designed to be modular such that the link between the theory and code clear as well as enabling readers to systematically develop an understanding of the code. Additionally, this modularity allows the code to be easily altered urging readers to further the capabilities of DIC.

5.3 Framework theory

Stereo DIC is fundamentally an extension of 2D DIC which relies upon two, or more, cameras simultaneously capturing images of the specimen (where specimen refers to the body being tracked) from different perspectives in order to determine its in-plane and out-of-plane displacements. As such, there are aspects of stereo DIC which are identical to those of 2D DIC; namely calibration and correlation. Stereo DIC relies upon the epipolar geometry to determine in-plane and out-of-plane displacements in the real world from in-plane displacements determined within images. Section 5.3.5 discusses how calibration and correlation are employed by stereo DIC based on the epipolar geometry and how triangulation methods are used to perform displacements transformation.

Although stereo DIC can employ more than two cameras [251], ADIC3D is limited to the conventional two camera implementation. Variables that need to be defined on a per camera basis have a subscript j to indicate that they are associated with the j^{th} camera (i.e., camera 1 and 2). Within this chapter an image series is defined as the collection of images captured by a single camera and an image set refers to the two corresponding image series. Thus, an image pair refers to the two images, one from each image series, which were captured simultaneously.

5.3.1 Homogeneous coordinates

Calibration and triangulation use homogeneous coordinates [212], to represent coordinates in projective space, such that affine and projective transformations can be implemented through matrix multiplication. Appending a scaling variable of unity to a n dimensional vector in Euclidean space converts it to a $n+1$ dimensional vector in projective space. Converting back to Euclidean space, represented as function Φ, entails dividing the vector's elements by the scaling variable before eliminating it to reduce the vector to n dimensions. Underlined variables denote homogenous coordinate vectors.

5.3.2 Calibration

Calibration, explained in terms of a single camera since each camera is calibrated separately, determines the parameters of the camera model which involves the four coordinate systems (CSs) depicted in Figure 5.1: (i) the three-dimensional (3D) world CS having arbitrary orientation and position; (ii) the 3D camera CS with its origin at the aperture of the camera and its z-axis aligned with the optical axis; (iii) the 2D ideal sensor CS which is coincident with the plane of the charge-coupled device (CCD) representing the image in the absence of distortion; and (iv) the 2D distorted sensor CS which is coincident with the ideal sensor CS but represents the actual image by accounting for radial distortion. No symbols have been assigned to the camera CS since it is not directly involved in DIC.

The camera model uses the pinhole camera model to project a 3D coordinate of a point on the specimen in the world CS to its corresponding 2D location in the ideal sensor CS before the radial distortion model determines its location in the distorted sensor CS.

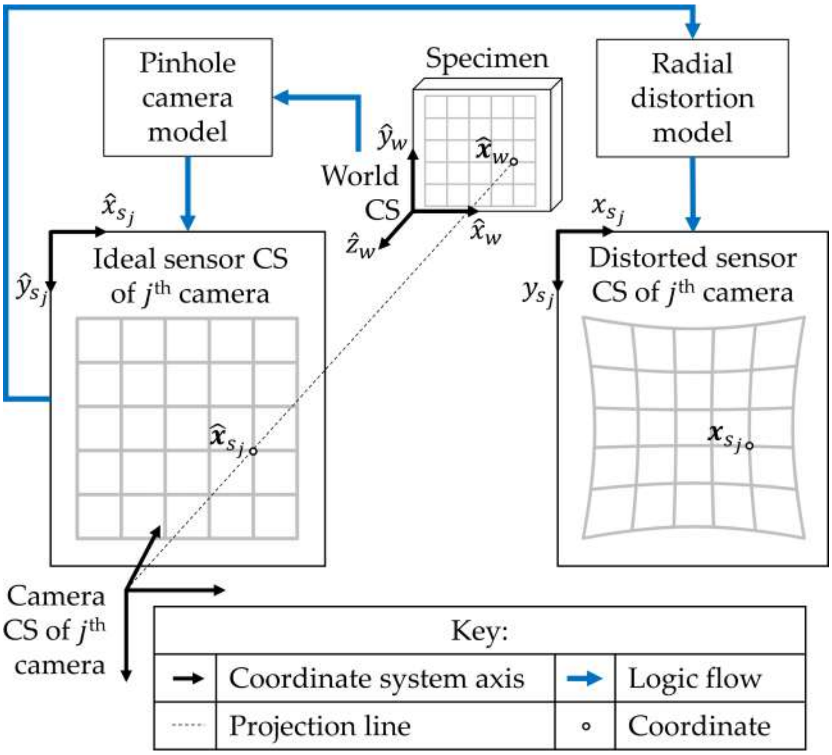


Figure 5.1: Schematic diagram of coordinate systems employed by the camera model.

5.3.2.1 Pinhole camera model

The projection matrix $\boldsymbol{Q}_j$, consisting of intrinsic ($\boldsymbol{K}_j$) and extrinsic ($\boldsymbol{V}_j$) parameter matrices, relates a coordinate in the world CS, $\underline{\hat{\boldsymbol{x}}}_w = \begin{bmatrix} \hat{x}_w & \hat{y}_w & \hat{z}_w 1 \end{bmatrix}^T$, to a coordinate in the ideal sensor CS, $\underline{\hat{\boldsymbol{x}}}_{s_j} = \begin{bmatrix} \hat{x}_{s_j} & \hat{y}_{s_j} & \hat{z}_{s_j} & 1 \end{bmatrix}^T$, as

$$\alpha_j \underline{\hat{\boldsymbol{x}}}_{s_j} = \boldsymbol{Q}_j \underline{\hat{\boldsymbol{x}}}_w = \boldsymbol{K}_j \boldsymbol{V}_j \underline{\hat{\boldsymbol{x}}}_w = \begin{bmatrix} \xi_{x_j} & c_{s_j} & c_{x_j} \\ 0 & \xi_{y_j} & c_{y_j} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_j^{11} & R_j^{12} & R_j^{13} & T_j^1 \\ R_j^{21} & R_j^{22} & R_j^{23} & T_j^2 \\ R_j^{31} & R_j^{32} & R_j^{33} & T_j^3 \end{bmatrix} \underline{\hat{\boldsymbol{x}}}_w \qquad (5.1)$$

Here α_j is the arbitrary scaling variable of the homogenous coordinates. Firstly, the rotation matrix, $\boldsymbol{R}_j$, and translation vector, $\boldsymbol{T}_j$, transform $\underline{\hat{\boldsymbol{x}}}_w$ to the camera CS. Thereafter this 3D coordinate is projected to the ideal sensor CS using $\boldsymbol{K}_j$ which performs four operations: (i) it projects the 3D coordinate to a 2D coordinate on the plane of the CCD; (ii) scales the coordinate from metric units to units of pixels using ratios ξ_{x_j} and ξ_{y_j} in the x- and y-directions respectively; (iii) applies a translation, c_{x_j} and c_{y_j} in the x- and y-directions respectively, placing the origin of the ideal sensor CS at the top left of the image; and (iv) converts from the orthogonal camera CS to a skew ideal sensor CS using c_{s_j}. In this work c_{s_j}=0 since an orthogonal ideal sensor CS is assumed.

$\boldsymbol{K}_j$ is only dependent on the camera system and does not change if the camera position changes. In contrast, $\boldsymbol{V}_j$ depends on the relative position and orientation between the camera and the world CS.

5.3.2.2 Radial distortion model

The radial distortion model is sufficient to account for distortion in the images caused by imperfections of the lens system [216]. First the normalised, ideal image coordinates, $\hat{\boldsymbol{x}}_{n_j} = \begin{bmatrix} \hat{x}_{n_j} & \hat{y}_{n_j} \end{bmatrix}^T$, are computed as

$$\hat{\boldsymbol{x}}_{n_j} = \begin{bmatrix} \hat{x}_{n_j} \\ \hat{y}_{n_j} \end{bmatrix} = \Phi\left(\boldsymbol{K}_j^{-1} \hat{\boldsymbol{x}}_{s_j}\right) \tag{5.2}$$

Thereafter, the unit-less radial distortion parameters, κ_j^1 and κ_j^2, are used to compute the normalised, distorted image coordinates, $\boldsymbol{x}_{n_j} = \begin{bmatrix} x_{n_j} & y_{n_j} \end{bmatrix}^T$, as [213]

$$\boldsymbol{x}_{n_j} = \left(1 + \kappa_j^1 \hat{\boldsymbol{x}}_{n_j}^T \hat{\boldsymbol{x}}_{n_j} + \kappa_j^2 \left(\hat{\boldsymbol{x}}_{n_j}^T \hat{\boldsymbol{x}}_{n_j}\right)^2\right) \hat{\boldsymbol{x}}_{n_j} \tag{5.3}$$

Finally, the distorted coordinates in the distorted sensor CS, $\boldsymbol{x}_{s_j} = \begin{bmatrix} x_{s_j} & y_{s_j} \end{bmatrix}^T$, are obtained as

$$\boldsymbol{x}_{s_j} = \begin{bmatrix} x_{s_j} \\ y_{s_j} \end{bmatrix} = \Phi\left(\boldsymbol{K}_j \begin{bmatrix} x_{n_j} \\ y_{n_j} \\ 1 \end{bmatrix}\right) \tag{5.4}$$

5.3.2.3 Determining calibration parameters

Prior to performing calibration, a calibration image series must be captured of the calibration plate. The calibration plate is a planar object containing distinct, point-like features, called calibration targets (CTs), at known 3D locations which define the position and orientation of the world CS. A corresponding set of 2D coordinates in the distorted sensor CS are obtained by locating the CTs in the calibration images. These sets of 3D and 2D coordinates are used to solve for the parameters, of the camera model, which describe the relationship between the two. Using at least 50 calibration images is recommended since this inverse problem is sensitive to noise in the 3D and 2D coordinates [22, 252].

Calibration is performed using two steps. Firstly, initial estimates for $\boldsymbol{K}_j$ and $\boldsymbol{V}_j$ are obtained using the method proposed by Zhang [213] while κ_j^1 and κ_j^2 are set to zero. Secondly, non-linear, least-squares optimisation is used to optimise all the parameters by minimising the total projection error, E_{proj}, given as

$$E_{proj} = \sum_{l=1}^{L} \left(\left(x_l^{calc} - x_l^{true}\right)^2 + \left(y_l^{calc} - y_l^{true}\right)^2\right) \tag{5.5}$$

Here $\boldsymbol{x}_l^{true} = \begin{bmatrix} x_l^{true} & y_l^{true} \end{bmatrix}^T$ and $\boldsymbol{x}_l^{calc} = \begin{bmatrix} x_l^{calc} & y_l^{calc} \end{bmatrix}^T$ are the locations of the l^{th} CT in the distorted sensor CS determined from the calibration image series and predicted by the camera model respectively. There is a total of L many CTs visible within the calibration image series.

5.3.3 Correlation

ADIC3D uses correlation to track a subset, a cluster of pixels, between images. The specimen's surface requires an isotropic, random pattern with high information content such that subsets are tracked reliably. Correlation operates on two images: a reference image, F, representing the specimen in its relaxed state and a deformed image, G, representing the specimen after experiencing displacement and/or deformation. Correlation determines how a subset of the reference image, called a reference subset ($\boldsymbol{f}$), must displace and deform such that it matches a corresponding subset in the deformed image, called an investigated subset ($\boldsymbol{g}$). This is referred to as the correspondence problem which is mathematically defined as the objective function. For a more detailed discussion on the correspondence problem the reader is referred to the authors' previous paper [191].

Note that correlation operates solely within the distorted sensor CS and is unconcerned with which camera the distorted sensor CS belongs to. Thus, in Sections 5.3.3 and 5.4.2 the subscript s_j is dropped from $\boldsymbol{x}_{s_j}$ (i.e., $\boldsymbol{x}$) to simplify the mathematics.

To understand how the objective function is comprised of the shape function (SF), interpolation method and correlation criterion first consider that the i^{th} pixel position of $\boldsymbol{f}$, $\boldsymbol{x}_i = \begin{bmatrix} x_i & y_i \end{bmatrix}^T$, is represented by the reference subset centre position, $\boldsymbol{x}^o = \begin{bmatrix} x^o & y^o \end{bmatrix}^T$, and the distance from $\boldsymbol{x}^o$ to $\boldsymbol{x}_i$, $\Delta\boldsymbol{x}_i = \begin{bmatrix} \Delta x_i & \Delta y_i \end{bmatrix}^T$, as

$$\boldsymbol{x}_i = \Delta\boldsymbol{x}_i + \boldsymbol{x}^o \tag{5.6}$$

The SF, $\boldsymbol{W}$, determines how the pixel positions of $\boldsymbol{f}$ displace as a result of the displacement and deformation of $\boldsymbol{f}$ which is quantified by the shape function parameters (SFPs), $\boldsymbol{P}$. The SF does this by modifying $\Delta\boldsymbol{x}_i$ as

$$\Delta\boldsymbol{x}'_i = \begin{bmatrix} \Delta x'_i \\ \Delta y'_i \end{bmatrix} = \boldsymbol{W}(\Delta\boldsymbol{x}_i, \boldsymbol{P}) \tag{5.7}$$

The pixel positions of $\boldsymbol{f}$ after displacement and deformation, $\boldsymbol{x}'_i = \begin{bmatrix} x'_i & y'_i \end{bmatrix}^T$, (referred to as query points) are determined as

$$\boldsymbol{x}'_i = \Delta\boldsymbol{x}'_i + \boldsymbol{x}^o \tag{5.8}$$

Thus $\Delta\boldsymbol{x}'_i$ represents the distance from the centre of $\boldsymbol{f}$ to $\boldsymbol{x}'_i$. The light intensity values of G are interpolated at $\boldsymbol{x}'_i$ to obtain $\boldsymbol{g}$. Thus, $\boldsymbol{g}$ represents a portion of G at a location described by the displacement SFPs with the deformation defined by the SFPs having been removed. The correlation criterion quantifies the degree of similarity between $\boldsymbol{f}$ and $\boldsymbol{g}$.

5.3.3.1 Shape function

The main SFs are the zero ($\boldsymbol{W}^{SF0}$), first ($\boldsymbol{W}^{SF1}$) and second-order SFs ($\boldsymbol{W}^{SF2}$) given as [13]

$$\begin{aligned}
\boldsymbol{W}^{SF0}\left(\Delta\boldsymbol{x}_i,\boldsymbol{P}^{SF0}\right)&=\begin{bmatrix}1 & 0 & u\\ 0 & 1 & v\end{bmatrix}\begin{bmatrix}\Delta x_i\\ \Delta y_i\\ 1\end{bmatrix}\\
\boldsymbol{W}^{SF1}\left(\Delta\boldsymbol{x}_i,\boldsymbol{P}^{SF1}\right)&=\begin{bmatrix}1+u_x & u_y & u\\ v_x & 1+v_y & v\end{bmatrix}\begin{bmatrix}\Delta x_i\\ \Delta y_i\\ 1\end{bmatrix}\\
\text{and}\quad \boldsymbol{W}^{SF2}\left(\Delta\boldsymbol{x}_i,\boldsymbol{P}^{SF2}\right)&=\begin{bmatrix}\frac{1}{2}u_{xx} & u_{xy} & \frac{1}{2}u_{yy} & 1+u_x & u_y & u\\ \frac{1}{2}v_{xx} & v_{xy} & \frac{1}{2}v_{yy} & v_x & 1+v_y & v\end{bmatrix}\begin{bmatrix}\Delta x_i^2\\ \Delta x_i\Delta y_i\\ \Delta y_i^2\\ \Delta x_i\\ \Delta y_i\\ 1\end{bmatrix}
\end{aligned}\tag{5.9}$$

Their sets of SFPs are given as

$$\begin{aligned}
\boldsymbol{P}^{SF0}&=\begin{bmatrix}u & v\end{bmatrix}^T\\
\boldsymbol{P}^{SF1}&=\begin{bmatrix}u & u_x & u_y & v & v_x & v_y\end{bmatrix}^T\\
\text{and}\quad \boldsymbol{P}^{SF2}&=\begin{bmatrix}u & u_x & u_y & u_{xx} & u_{xy} & u_{yy} & v & v_x & v_y & v_{xx} & v_{xy} & v_{yy}\end{bmatrix}^T
\end{aligned}\tag{5.10}$$

Here u and v represent the displacement of $\boldsymbol{f}$ in the x- and y-directions respectively. Furthermore, u_x, u_{xx}, v_y and v_{yy} define stretching of $\boldsymbol{f}$ whereas u_y, v_x, u_{yy}, v_{xx}, u_{xy} and v_{xy} define shearing. Although higher order SFs more reliably determine displacements in complex displacement fields by accounting for more complex deformation, the DIC process is only concerned with the displacements determined by correlation.

5.3.3.2 Interpolation

ADIC3D employs the bi-cubic b-spline interpolation method detailed in the work of Hou et al. [218]. Letting F and G represent interpolation functions, the light intensity values of pixel i of the reference, f_i, and investigated subsets, g_i, are interpolated as

$$\begin{aligned}
f_i&=F(\boldsymbol{x}^o+\Delta\boldsymbol{x}_i)\\
\text{and}\quad g_i&=G(\boldsymbol{x}^o+\boldsymbol{W}(\Delta\boldsymbol{x}_i,\boldsymbol{P}))
\end{aligned}\tag{5.11}$$

Images are filtered using a Gaussian low-pass filter [114], since otherwise the sensitivity of bi-cubic b-spline interpolation to high frequency noise would lead to bias in the displacements. The Gaussian filtering parameters, being the window size (β) and the standard deviation of the Gaussian function (σ^g), should be chosen to reduce bias without increasing variance.

5.3.3.3 Correlation criterion

The objective function employs the more computationally efficient ZNSSD correlation criterion (C_{ZNSSD}), having a range of $\{C_{ZNSSD}\in\mathbb{R}|0\leq C_{ZNSSD}\leq 4\}$ where smaller values indicate higher similarity, given as

$$c_{ZNSSD}=\sum_{i=1}^{I}\left[\frac{f_i-\bar{f}}{\tilde{f}}-\frac{g_i-\bar{g}}{\tilde{g}}\right]^2 \tag{5.12}$$

$$\bar{f}=\frac{\sum_{i=1}^{I}f_i}{I} \qquad \bar{g}=\frac{\sum_{i=1}^{I}g_i}{I} \tag{5.13}$$

$$\tilde{f}=\sqrt{\sum_{i=1}^{I}(f_i-\bar{f})^2} \quad \text{and} \quad \tilde{g}=\sqrt{\sum_{i=1}^{I}(g_i-\bar{g})^2} \tag{5.14}$$

Here there are I many pixels per-subset. The correlation coefficient is reported as the zero-mean normalised cross-correlation (ZNCC), C_{ZNCC}, since its range, $\{C_{ZNCC}\in\mathbb{R}|-1\leq C_{ZNCC}\leq 1\}$ where larger values indicate higher similarity, is more intuitive. These correlation criteria are related as [102]

$$C_{ZNCC}=1-\frac{C_{ZNSSD}}{2} \tag{5.15}$$

5.3.3.4 Objective function

The objective function, based on Equation (5.12), is given as

$$C_{ObjFun}=\sum_{i=1}^{I}\left[\frac{f_i-\bar{f}}{\tilde{f}}-\frac{G(\boldsymbol{x}^o+\boldsymbol{W}(\Delta\boldsymbol{x}_i,\boldsymbol{P}))-\bar{g}}{\tilde{g}}\right]^2 \tag{5.16}$$

Therefore, correlation seeks the SFPs that define $\boldsymbol{g}$ to contain a light intensity pattern similar to $\boldsymbol{f}$ thereby optimising C_{ZNSSD}. However, the IC-GN optimisation method requires modifying the objective function in order to derive an iterative optimisation equation from it. More specifically, the current estimate of the SFPs, $\boldsymbol{P}$, is applied to the investigated subset while the iterative improvement of the SFPs, $\Delta\boldsymbol{P}$, is applied to the reference subset as

$$C_{ObjFun}=\sum_{i=1}^{I}\left[\frac{F(\boldsymbol{x}^o+\boldsymbol{W}(\Delta\boldsymbol{x}_i,\Delta\boldsymbol{P}))-\bar{f}}{\tilde{f}}-\frac{G(\boldsymbol{x}^o+\boldsymbol{W}(\Delta\boldsymbol{x}_i,\boldsymbol{P}))-\bar{g}}{\tilde{g}}\right]^2 \tag{5.17}$$

5.3.3.5 Optimisation equation

The optimisation equation derived from Equation (5.17), derivation detailed in [191], is given as

$$\Delta\boldsymbol{P}=-\boldsymbol{H}^{-1}\sum_{i=1}^{I}\left(\nabla f_i\frac{\partial\boldsymbol{W}_i}{\partial\boldsymbol{P}}\right)^T\left[f_i-\bar{f}-\frac{\tilde{f}}{\tilde{g}}(G(\boldsymbol{x}^o+\boldsymbol{W}(\Delta\boldsymbol{x}_i,\boldsymbol{P}))-\bar{g})\right] \tag{5.18}$$

Here $\boldsymbol{H}$ is the Hessian, defined in Equation (5.19), and the remaining terms (within the summation) form the Jacobian, $\boldsymbol{J}$. $\boldsymbol{H}$, being independent of $\boldsymbol{P}$, is precomputed prior to beginning iterations.

$$\boldsymbol{H}=\sum_{i=1}^{I}\left[\left(\nabla f_i\frac{\partial\boldsymbol{W}_i}{\partial\boldsymbol{P}}\right)^T\left(\nabla f_i\frac{\partial\boldsymbol{W}_i}{\partial\boldsymbol{P}}\right)\right] \tag{5.19}$$

$\nabla \boldsymbol{f}_i=\begin{bmatrix}\frac{\partial f_i}{\partial x} & \frac{\partial f_i}{\partial y}\end{bmatrix}$ is the light intensity gradient of $\boldsymbol{f}$ at pixel i. $\frac{\partial\boldsymbol{W}_i}{\partial\boldsymbol{P}}$ is the Jacobian of the SF for pixel i and based on the SF order is given as

$$\begin{gathered}
\frac{\partial \boldsymbol{W}_i^{SF0}}{\partial \boldsymbol{P}^{SF0}}=\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\
\frac{\partial \boldsymbol{W}_i^{SF1}}{\partial \boldsymbol{P}^{SF1}}=\begin{bmatrix} 1 & \Delta x_i & \Delta y_i & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta x_i & \Delta y_i \end{bmatrix} \\
\text{and} \\
\frac{\partial \boldsymbol{W}_i^{SF2}}{\partial \boldsymbol{P}^{SF2}}=\begin{bmatrix} 1 & \Delta x_i & \Delta y_i & \frac{\Delta x_i^2}{2} & \Delta x_i \Delta y_i & \frac{\Delta y_i^2}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & \Delta x_i & \Delta y_i & \frac{\Delta x_i^2}{2} & \Delta x_i \Delta y_i & \frac{\Delta y_i^2}{2} \end{bmatrix}
\end{gathered} \tag{5.20}$$

Although, each iteration of Equation (5.18) attempts to determine how f should displace and deform (according to $\Delta \boldsymbol{P}$) such that it matches g, f is not displaced or deformed in practice. Instead $\Delta \boldsymbol{P}$ is used to determine the updated SFPs, $\boldsymbol{P}_{update}$, which serves as the current estimate in the next iteration. The stopping criterion deems when further iterations are redundant.

5.3.3.6 Updating the SFPs

$\boldsymbol{P}_{update}$ is obtained by composing the inverse of $\Delta \boldsymbol{P}$ with $\boldsymbol{P}$ as

$$\boldsymbol{P}_{update}=\boldsymbol{\omega}(\boldsymbol{P})\boldsymbol{\omega}(\Delta \boldsymbol{P})^{-1} \tag{5.21}$$

Here $\boldsymbol{\omega}$, being dependent on the SF order, uses the SFPs to populate a square matrix as [13]

$$\begin{gathered}
\boldsymbol{\omega}^{SF0}\left(\boldsymbol{P}^{SF0}\right)=\begin{bmatrix} 1 & 0 & u \\ 0 & 1 & v \\ 0 & 0 & 1 \end{bmatrix} \\
\boldsymbol{\omega}^{SF1}\left(\boldsymbol{P}^{SF1}\right)=\begin{bmatrix} 1+u_x & u_y & u \\ v_x & 1+v_y & v \\ 0 & 0 & 1 \end{bmatrix} \\
\boldsymbol{\omega}^{SF2}\left(\boldsymbol{P}^{SF2}\right)=\begin{bmatrix} 1+A_1 & A_2 & A_3 & A_4 & A_5 & A_6 \\ A_7 & 1+A_8 & A_9 & A_{10} & A_{11} & A_{12} \\ A_{13} & A_{14} & 1+A_{15} & A_{16} & A_{17} & A_{18} \\ \frac{1}{2}u_{xx} & u_{xy} & \frac{1}{2}u_{yy} & 1+u_x & u_y & u \\ \frac{1}{2}v_{xx} & v_{xy} & \frac{1}{2}v_{yy} & v_x & 1+v_y & v \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}
\end{gathered} \tag{5.22}$$

where

$$\begin{array}{cc}
A_1=2u_x+u_x^2+uu_{xx} & A_2=2uu_{xy}+2(1+u_x)u_y \\
A_3=u_y^2+uu_{yy} & A_4=2u(1+u_x) \\
A_5=2uu_y & A_6=u^2 \\
A_7=\frac{1}{2}(vu_{xx}+2(1+u_x)v_x+uv_{xx}) & A_8=u_yv_x+u_xv_y+vu_{xy}+uv_{xy}+v_y+u_x \\
A_9=\frac{1}{2}(vu_{yy}+2(1+v_y)u_y+uv_{yy}) & A_{10}=v+vu_x+uv_x \\
A_{11}=u+vu_y+uv_y & A_{12}=uv \\
A_{13}=v_x^2+vv_{xx} & A_{14}=2vv_{xy}+2v_x(1+v_y) \\
A_{15}=2v_y+v_y^2+vv_{yy} & A_{16}=2vv_x \\
A_{17}=2v(1+v_y) & A_{18}=v^2
\end{array}$$

5.3.3.7 Stopping criterion

The stopping criterion, $\|\Delta\boldsymbol{P}\|$, determines the magnitude of the change in the SFPs between iterations, based on $\Delta\boldsymbol{P}$, as [210]

$$\left\|\Delta\boldsymbol{P}^{SF0}\right\|=\left[\Delta u^2+\Delta v^2\right]^{0.5}$$
$$\left\|\Delta\boldsymbol{P}^{SF1}\right\|=\left[\Delta u^2+(\Delta u_x\zeta)^2+\left(\Delta u_y\zeta\right)^2+\Delta v^2+(\Delta v_x\zeta)^2+\left(\Delta v_y\zeta\right)^2\right]^{0.5}$$
$$\text{and }\left\|\Delta\boldsymbol{P}^{SF2}\right\|=\Bigg[\Delta u^2+(\Delta u_x\zeta)^2+\left(\Delta u_y\zeta\right)^2+\left(\frac{1}{2}\Delta u_{xx}\zeta\right)^2+\left(\frac{1}{2}\Delta u_{xy}\zeta\right)^2+ \tag{5.23}$$
$$\left(\frac{1}{2}\Delta u_{yy}\zeta\right)^2+\Delta v^2+(\Delta v_x\zeta)^2+\left(\Delta v_y\zeta\right)^2+\left(\frac{1}{2}\Delta v_{xx}\zeta\right)^2+\left(\frac{1}{2}\Delta v_{xy}\zeta\right)^2+\left(\frac{1}{2}\Delta v_{yy}\zeta\right)^2\Bigg]^{0.5}$$

Here ζ is the largest value of $\Delta\boldsymbol{x}_i$ or $\Delta\boldsymbol{y}_i$ for the subset. Iterations cease once $\|\Delta\boldsymbol{P}\|$ falls below a predefined value called the stopping criterion value.

5.3.4 Epipolar geometry

The epipolar geometry is the projective geometry between the ideal sensor CSs, i.e., between camera 1 and 2. Consider the following definitions aided by Figure 5.2: (i) the baseline is formed between the two origins of the camera CSs; (ii); the epipolar plane is formed between the 3D coordinate and the baseline (iii) the epipolar lines, $\underline{\boldsymbol{\lambda}}_i$, are the intersection of the epipolar plane and the x-y plane of the ideal sensor CSs; and (iv) the epipoles, given as $\underline{\boldsymbol{e}}_{s_j}=\left[e^1_{s_j}\ \ e^2_{s_j}\ \ e^3_{s_j}\right]^T$ in homogeneous coordinates, are located at the intersection of the baseline and the x-y plane of the ideal sensor CSs.

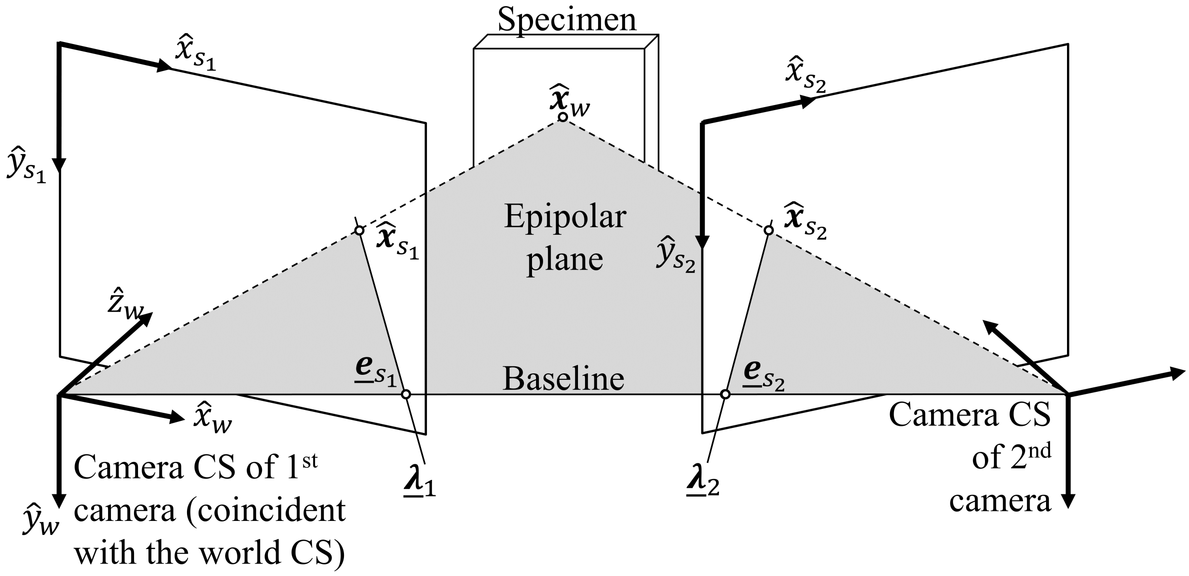


Figure 5.2: Schematic illustration of the epipolar geometry.

For 2D coordinates in the ideal sensor CSs of the first, $\underline{\hat{x}}_{s_1}$, and second cameras, $\underline{\hat{x}}_{s_2}$, to correspond to the same $\hat{x}_w$ requires that these coordinates satisfy the epipolar constraint defined as

$$\underline{\hat{x}}_{s_2}^T B \underline{\hat{x}}_{s_1} = 0 \tag{5.24}$$

More specifically, since the fundamental matrix, $\boldsymbol{B}$, maps a point in one ideal sensor CS to its corresponding epipolar line in the other ideal sensor CS, the 2D coordinates must lie on the epipolar lines of their respective ideal sensor CSs. $\boldsymbol{B}$ encapsulates the intrinsic and extrinsic relation between the ideal sensor CSs.

Computing $\boldsymbol{B}$ requires applying a homography, $\boldsymbol{\Omega}$, to the projection matrices of the cameras to transform them to canonical form ($\boldsymbol{Q}_{c_j}$), where the world CS coincides with the camera CS of the first camera, as [253]

$$\boldsymbol{Q}_{c_1} = \boldsymbol{Q}_1 \boldsymbol{\Omega} = \left[\boldsymbol{I} | \boldsymbol{0}\right] \qquad \boldsymbol{Q}_{c_2} = \boldsymbol{Q}_2 \boldsymbol{\Omega} = \left[\boldsymbol{M} | \boldsymbol{m}\right] \tag{5.25}$$

where

$$\boldsymbol{\Omega} = \begin{bmatrix} Q_1^{11} & Q_1^{12} & Q_1^{13} & Q_1^{14} \\ Q_1^{21} & Q_1^{22} & Q_1^{23} & Q_1^{24} \\ Q_1^{31} & Q_1^{32} & Q_1^{33} & Q_1^{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} \tag{5.26}$$

Here $\boldsymbol{I}$ is a 3×3 identity matrix. $\boldsymbol{Q}_{c_2}$, which consists of a 3×3 matrix $\boldsymbol{M}$ and a 3×1 vector $\boldsymbol{m}$, transforms a coordinate from the camera CS of the first camera to its location in the ideal sensor CS of the second camera. $\boldsymbol{B}$ is given as

$$\boldsymbol{B} = \left[\boldsymbol{m}\right]_\times \boldsymbol{M} \tag{5.27}$$

where $\left[\boldsymbol{m}\right]_\times$ is the skew-symmetric matrix of $\boldsymbol{m} = \left[m^1 \quad m^2 \quad m^3\right]^T$ defined as

$$[\boldsymbol{m}]_x = \begin{bmatrix} 0 & -m^3 & m^2 \\ m^3 & 0 & -m^1 \\ -m^2 & m^1 & 0 \end{bmatrix} \tag{5.28}$$

5.3.5 Stereo DIC overview

The core operations of stereo DIC are subset matching, calibration and displacement transformation. For a point on the specimen subset matching determines the two sets of in-plane, pixel displacements it experiences in each distorted sensor CS which, along with the calibration parameters determined by the calibration process, are used by displacement transformation to determine the in-plane and out-of-plane (thus 3D) metric displacements it experiences in the world CS. Note that with regards to subset matching and displacement transformation the term in-plane refers to the x-y plane of the distorted sensor CSs and world CS respectively.

Subset matching is composed of stereo and temporal matching which utilise correlation differently. Firstly, subset matching breaks up the first image of the first image series (FIS1) into subsets. Secondly, stereo matching determines the location of the corresponding subsets in the first image of the second image series (FIS2). Such subsets, one from each image of an image pair which correspond to the same point on the specimen, are termed a subset pair. Lastly, temporal matching, processing each image series separately, determines the displacements experienced by these subsets through their respective image series. This is achieved by sequentially stepping through an image series, starting at the second image, and at each step performing correlation by treating the current image as the deformed image while the reference image is dictated by the reference image strategy as discussed in Section 5.3.6.1. Thus, stereo matching is unique to stereo DIC while temporal matching is identical to that of 2D DIC.

The calibration method is identical to that of 2D DIC. However, in order to determine the epipolar geometry from the calibration parameters requires that the two calibration image series, used to calibrate the cameras separately, form part of the same calibration image set.

Displacement transformation, illustrated in Figure 5.3, steps through image pairs of the image set similarly to temporal matching. More specifically, considering a single reference subset pair with known locations, $x^o_{s_j}$, it uses the in-plane displacements determined by temporal matching (u_j and v_j in the x- and y-directions), between the reference images (of the reference image pair) and the deformed images (of the deformed image pair), to determine the locations of the investigated subset pair, $x^d_{s_j}$.

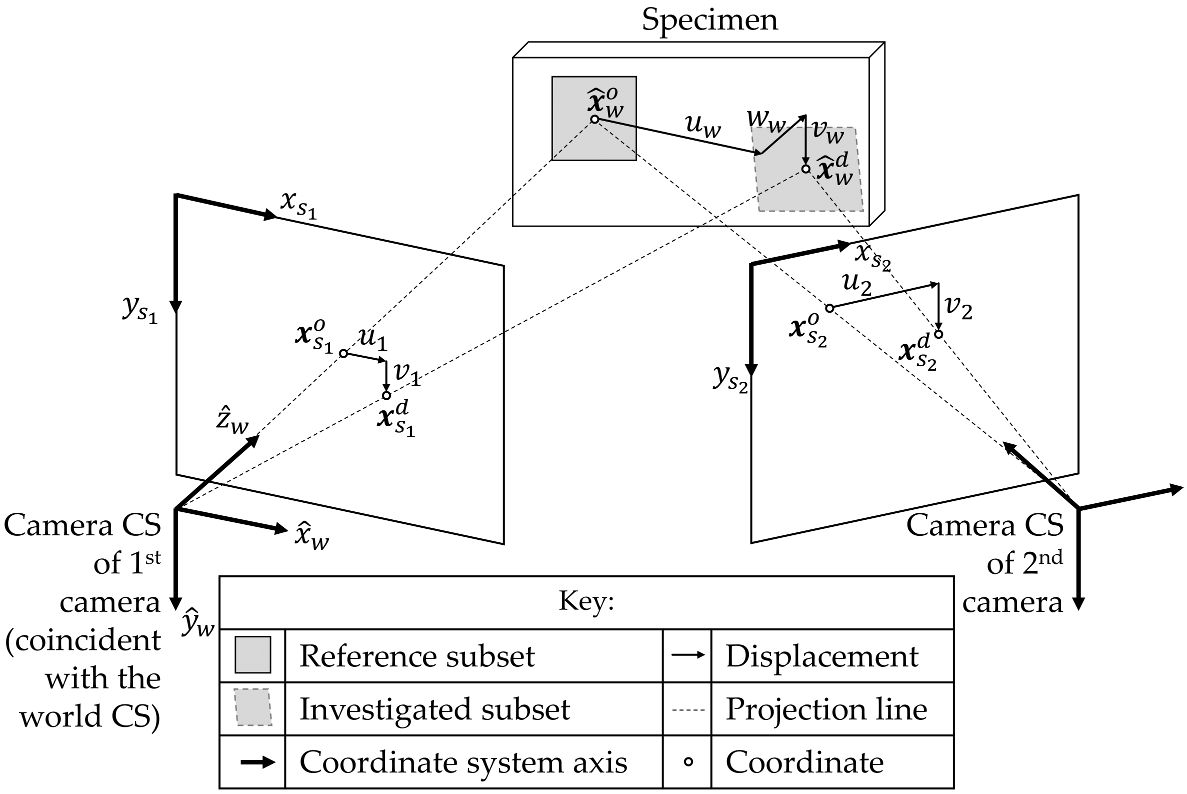


Figure 5.3: Schematic illustration of the displacement transformation process. Note that a projection line represents the combined process of distortion correction and triangulation.

In contrast to 2D DIC, distortion correction alone is insufficient to determine the ideal coordinate of a subset in the ideal sensor CS. More specifically, discretisation of light to form an image introduces noise in the images [254] and displacements measured by correlation, which are used to determine the locations of subsets, inherently contain some noise. Thus, correcting $x^o_{s_j}$ and $x^d_{s_j}$ for radial distortion results in the measured coordinates of the reference, $\check{x}^o_{s_j}$, and investigated subset pair, $\check{x}^d_{s_j}$, in the ideal sensor CS respectively. Generally, these measured coordinates of a subset pair, represented as $\check{\boldsymbol{x}}_{s_j}$, do not satisfy the epipolar constraint of Equation (5.24) since their back-projection rays do not intersect. Thus, triangulation, which determines the 3D coordinate in the world CS, $\hat{\boldsymbol{x}}_w$, from $\check{\boldsymbol{x}}_{s_j}$, is performed in two steps. First the polynomial triangulation method, detailed in Section 5.3.7, determines the ideal coordinates of a subset pair, $\hat{\boldsymbol{x}}_{s_j}$, which lie as close to $\check{\boldsymbol{x}}_{s_j}$ as possible while satisfying the epipolar constraint. Thereafter the linear triangulation method, detailed in Section 5.3.8, determines $\hat{\boldsymbol{x}}_w$ to which $\hat{\boldsymbol{x}}_{s_j}$ correspond as the intersection of their back-projected rays. Performing triangulation on $\check{\boldsymbol{x}}^o_{s_j}$ and $\check{\boldsymbol{x}}^d_{s_j}$ results in $\hat{\boldsymbol{x}}^o_w$ and $\hat{\boldsymbol{x}}^d_w$ respectively.

Finally, $\hat{\boldsymbol{x}}^o_w$ is subtracted from $\hat{\boldsymbol{x}}^d_w$ to determine the displacement experienced by the point on the specimen, in the x-, y- and z-directions represented as $\hat{u}_w$, $\hat{v}_w$ and $\hat{w}_w$ respectively, between the time stamps at which the reference and deformed image pairs were captured. Note that subscript w indicates coordinates or displacements in the world CS whereas variable w indicates displacement in the z-direction. This is done for all subset pairs across all image pair combinations, analysed by temporal matching, to determine the full-field metric displacements experienced by the specimen throughout the image set.

5.3.6 Subset matching

The optical flow determined by temporal and stereo matching is a result of the displacement and/or deformation experienced by the specimen (over time) and the perspective change between the cameras respectively. They differ predominantly in the method used to obtain SFP initial estimates.

5.3.6.1 Temporal matching

There are two reference image strategies for temporal matching: absolute and incremental. The absolute and incremental strategies define the reference image as either the first image of the image series or the previous image, relative to the current deformed image, respectively. The incremental strategy more reliably tracks more severe deformations between images but incurs accumulative errors when determining the displacement relative to the first image.

Temporal matching uses the phase correlation method (PCM) to determine initial estimates for the displacements between two subsets. PCM efficiently computes the correlation coefficients, in the frequency domain, for a range of integer displacements, of up to half the subset size in each direction, between the two subsets. The integer displacements with the best correlation coefficient are identified and used as initial estimates for the displacement SFPs while the deformation SFPs are set to zero. Consult the work of Foroosh et al. [223] for a discussion on PCM. For the absolute strategy PCM is only used for the first correlation run. Thereafter the SFPs of the previous correlation run, of the same image series, are used as an initial estimate for the SFPs of the current correlation run. The incremental strategy uses PCM for each correlation run.

5.3.6.2 Stereo matching

Determining reliable out-of-plane displacements in the world CS requires an angle between the optical axes of the cameras of between 15°C and 35°C [110, 105].

As such, stereo matching uses the second-order SF since subsets experience complex deformation between the FIS1 and FIS2 due to this perspective change [22].

Traditionally, template matching, which utilises image rectification, was used to determine SFP initial estimates for stereo matching [255]. However, image rectification is susceptible to errors in the calibration parameters, can cause aliasing in the rectified images and interpolation during image rectification can lead to errors in the determined SFPs [121].

As such, feature matching methods are becoming favoured [22]. ADIC3D's feature matching method is based on that proposed by Zhou et al. [256]. It uses the scale-invariant feature transform (SIFT) feature matching algorithm [257] to identify corresponding keypoints between the images (where a keypoint is a unique feature in an image) and the affine transformation mapping model to mathematically relate the coordinates of these keypoint pairs between the images.

The SIFT algorithm is used because it is robust to translation, rotation, image scaling, variation in illumination, affine transformation and moderate perspective change. SIFT identifies keypoints that are scale invariant and computes a descriptor, in the form of a 128-dimensional vector, for each keypoint based on the light intensity gradient information in the vicinity of the keypoint.

For each keypoint of the FIS1 the Euclidean distance between its descriptor and that of every keypoint of the FIS2 is determined. The two keypoints of the FIS2 with the smallest descriptor distances are identified. If the ratio of the smallest to second smallest descriptor distance is less than 0.8 the keypoint of the FIS2 with the smallest descriptor distance is designated as the matching keypoint of the keypoint of the FIS1 [258]. Otherwise, this keypoint of the FIS1 is discarded.

The affine transformation relates a keypoint pair as

$$\begin{bmatrix} x_{s_2}^k \\ y_{s_2}^k \end{bmatrix} = \begin{bmatrix} 1+a_1 & a_2 & a_3 \\ a_4 & 1+a_5 & a_6 \end{bmatrix} \begin{bmatrix} x_{s_1}^k \\ y_{s_1}^k \\ 1 \end{bmatrix} \tag{5.29}$$

Here $\boldsymbol{x}_{s_1}^k = \begin{bmatrix} x_{s_1}^k & y_{s_1}^k \end{bmatrix}^T$ and $\boldsymbol{x}_{s_2}^k = \begin{bmatrix} x_{s_2}^k & y_{s_2}^k \end{bmatrix}^T$ are the locations of the k^{th} keypoint in the FIS1 and FIS2 respectively while a_1 through a_6 are the affine transformation parameters. Given three or more keypoint pairs the affine transformation parameters can be solved either as a system of linear equations or through linear least-squares respectively.

The K many $\boldsymbol{x}_{s_1}^k$ that are nearest to the subset's centre position are used to determine the affine transformation parameters of the subset. However, since SIFT often returns false keypoint pairs, the m-estimator sample consensus (MSAC) method [258] is used to remove these false keypoint pairs when determining the affine transformation parameters. False keypoint pairs are identified as those which have a squared error distance greater than a predefined threshold, τ. The squared error distance of the k^{th} keypoint pair, E_k^{dist}, is calculated as

$$E_k^{dist} = \left(x_{s_2}^k - (1+a_1)x_{s_1}^k - a_2 y_{s_1}^k - a_3\right)^2 + \left(y_{s_2}^k - a_4 x_{s_1}^k - (1+a_5) y_{s_1}^k - a_6\right)^2 \tag{5.30}$$

ADIC3D uses $K=20$, $\tau=1$ pixel^2 and a confidence, specifying the probability that MSAC finds the maximum number of inliers, of 99.5%. Second-order SFP initial estimates are determined from the affine transformation parameters and the subset position in the FIS1, $\boldsymbol{x}_{s_1}\begin{bmatrix}x_{s_1} & y_{s_1}\end{bmatrix}^T$, as

$$\begin{aligned} u&=a_1x_{s_1}+a_2y_{s_1}+a_3 & v&=a_4x_{s_1}+a_5y_{s_1}+a_6 \\ u_x&=a_1 & v_x&=a_4 \\ u_y&=a_2 & v_y&=a_5 \\ u_{xx}&=0 & v_{xx}&=0 \\ u_{xy}&=0 & v_{xy}&=0 \\ u_{yy}&=0 & v_{yy}&=0 \end{aligned} \tag{5.31}$$

Once correlation has optimised these SFPs the corresponding subset position in the FIS2, $\boldsymbol{x}_{s_2}=\begin{bmatrix}x_{s_2} & y_{s_2}\end{bmatrix}^T$, is determined as

$$\begin{bmatrix}x_{s_2}\\y_{s_2}\end{bmatrix}=\begin{bmatrix}x_{s_1}\\y_{s_1}\end{bmatrix}+\begin{bmatrix}u\\v\end{bmatrix} \tag{5.32}$$

5.3.7 Polynomial triangulation method

The polynomial triangulation method, proposed by Hartley and Sturm [254], determines $\hat{\boldsymbol{x}}_{s_j}$ by minimising the geometric error cost function, D, given as

$$D=\delta(\check{\boldsymbol{x}}_{s_1},\hat{\boldsymbol{x}}_{s_1})^2+\delta(\check{\boldsymbol{x}}_{s_2},\hat{\boldsymbol{x}}_{s_2})^2 \text{ subject to } \underline{\hat{\boldsymbol{x}}}_{s_2}{}^T\boldsymbol{B}\underline{\hat{\boldsymbol{x}}}_{s_1}=0 \tag{5.33}$$

Here function δ determines the Euclidean distance between two points. Determining $\hat{\boldsymbol{x}}_{s_j}$ by minimising the distance between the coordinates $\hat{\boldsymbol{x}}_{s_j}$ and $\check{\boldsymbol{x}}_{s_j}$ within the ideal sensor CSs ensures that the polynomial triangulation method is projective-invariant. Projective-invariance implies that the projective frame within which $\hat{\boldsymbol{x}}_w$ is defined does not affect the values of $\hat{\boldsymbol{x}}_{s_j}$ [253]. The polynomial method assumes that that the calibration parameters are known with greater accuracy than the measured coordinates [254]. Furthermore, it is assumed that neither $\hat{\boldsymbol{x}}_{s_j}$ nor $\check{\boldsymbol{x}}_{s_j}$ coincides with an epipole since this would lead to an unfavourable solution for the 3D coordinate.

It is known that a pair of ideal coordinates which satisfy the epipolar constraint lie on a pair of corresponding epipolar lines. Additionally, the epipolar plane can only rotate about the baseline since the camera CSs, and thus epipoles, are fixed. Thus, there is a family of epipolar lines upon which the idea coordinates can lie. Furthermore, the projective-invariance of the polynomial triangulation method allows applying a separate rigid transformation to each ideal sensor CS in order to parameterise the family of epipolar lines in terms of a single variable t. As such, homogeneous coordinates are utilised and Equation (5.33) becomes

$$D=\delta\left(\underline{\check{\boldsymbol{x}}}_{s_1},\underline{\boldsymbol{\lambda}}_1(t)\right)^2+\delta\left(\underline{\check{\boldsymbol{x}}}_{s_2},\underline{\boldsymbol{\lambda}}_2(t)\right)^2 \tag{5.34}$$

where $\underline{\boldsymbol{\lambda}}_j(t)$ is a vector representing an epipolar line in the ideal sensor CS as a function of t while $\delta\left(\underline{\check{\boldsymbol{x}}}_{s_j},\underline{\boldsymbol{\lambda}}_j(t)\right)$ represents the distance from $\underline{\check{\boldsymbol{x}}}_{s_j}$ to its orthogonal projection $\underline{\boldsymbol{\lambda}}_j(t)$. The rigid transformations first apply a translation matrix, $\boldsymbol{T}_j^{pm}$, to each ideal sensor CS translating $\underline{\check{\boldsymbol{x}}}_{s_j}$ to the origin of its CS.

$$\boldsymbol{T}_1^{pm}=\begin{bmatrix}1 & 0 & -\check{x}_{s_1}\\0 & 1 & -\check{y}_{s_1}\\0 & 0 & 1\end{bmatrix} \quad \text{and} \quad \boldsymbol{T}_2^{pm}=\begin{bmatrix}1 & 0 & -\check{x}_{s_2}\\0 & 1 & -\check{y}_{s_2}\\0 & 0 & 1\end{bmatrix} \tag{5.35}$$

The original fundamental matrix, $\boldsymbol{B}$, determined using Equation (5.27), needs to be updated as

$$\boldsymbol{B}_1 = \boldsymbol{T}_2^{pm-T} \boldsymbol{B} \boldsymbol{T}_1^{pm-1} \tag{5.36}$$

The epipoles $\underline{\boldsymbol{e}}_{s_1}$ and $\underline{\boldsymbol{e}}_{s_2}$ of the first and second cameras are computed, using singular value decomposition, as the right and left null-space of $\boldsymbol{B}_1$ respectively. Thereafter they are normalised such that $\left(e^1_{s_1}\right)^2 + \left(e^2_{s_1}\right)^2 = 1$ and $\left(e^1_{s_2}\right)^2 + \left(e^2_{s_2}\right)^2 = 1$.

The second step of the rigid transformation applies rotations $\boldsymbol{R}_1^{pm}$ and $\boldsymbol{R}_2^{pm}$ to the ideal sensor CSs of the first and second cameras respectively in order to place their epipoles on the x-axes of their respective CSs as $\boldsymbol{R}_1^{pm} \underline{\boldsymbol{e}}_{s_1} = \begin{bmatrix} 1 & 0 & e^3_{s_1} \end{bmatrix}^T$ and $\boldsymbol{R}_2^{pm} \underline{\boldsymbol{e}}_{s_2} = \begin{bmatrix} 1 & 0 & e^3_{s_2} \end{bmatrix}^T$.

$$\boldsymbol{R}_1^{pm} = \begin{bmatrix} e^1_{s_1} & e^2_{s_1} & 0 \\ -e^2_{s_1} & e^1_{s_1} & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \boldsymbol{R}_2^{pm} = \begin{bmatrix} e^1_{s_2} & e^2_{s_2} & 0 \\ -e^2_{s_2} & e^1_{s_2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.37}$$

The fundamental matrix is updated as

$$\boldsymbol{B}_2 = \boldsymbol{R}_2^{pm} \boldsymbol{B}_1 \boldsymbol{R}_1^{pmT} \tag{5.38}$$

Since $\boldsymbol{B}_2 \begin{bmatrix} 1 & 0 & e^3_{s_1} \end{bmatrix}^T = \begin{bmatrix} 1 & 0 & e^3_{s_2} \end{bmatrix} \boldsymbol{B}_2 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$ the elements of the fundamental matrix are represented as

$$\boldsymbol{B}_2 = \begin{bmatrix} e^3_{s_1} e^3_{s_2} \varphi_4 & -e^3_{s_2} \varphi_3 & -e^3_{s_2} \varphi_4 \\ -e^3_{s_1} \varphi_2 & \varphi_1 & \varphi_2 \\ -e^3_{s_1} \varphi_4 & \varphi_3 & \varphi_4 \end{bmatrix} \tag{5.39}$$

An epipolar line of the first camera, $\underline{\boldsymbol{\lambda}}_1(t)$, passing through point $\begin{bmatrix} 0 & t & 1 \end{bmatrix}^T$ on the y-axis of the ideal sensor CS is represented as

$$\underline{\boldsymbol{\lambda}}_1(t) = \begin{bmatrix} 0 & t & 1 \end{bmatrix}^T \times \begin{bmatrix} 1 & 0 & e^3_{s_1} \end{bmatrix}^T = \begin{bmatrix} t e^3_{s_1} & 1 & -t \end{bmatrix}^T \tag{5.40}$$

The corresponding epipolar line of the second camera, $\underline{\boldsymbol{\lambda}}_2(t)$, is computed as

$$\underline{\boldsymbol{\lambda}}_2(t) = \boldsymbol{B}_2 \begin{bmatrix} 0 & t & 1 \end{bmatrix}^T = \begin{bmatrix} -e^3_{s_2}(\varphi_3 t + \varphi_4) & \varphi_1 t + \varphi_2 & \varphi_3 t + \varphi_4 \end{bmatrix}^T \tag{5.41}$$

Thus, $\delta\left(\underline{\check{\boldsymbol{x}}}_{s_1}, \underline{\boldsymbol{\lambda}}_1(t)\right)^2 = \frac{t^2}{1+\left(t e^3_{s_1}\right)^2}$ and $\delta\left(\underline{\check{\boldsymbol{x}}}_{s_2}, \underline{\boldsymbol{\lambda}}_2(t)\right)^2 = \frac{(\varphi_3 t + \varphi_4)^2}{(\varphi_1 t + \varphi_2)^2 + {e^3_{s_2}}^2 (\varphi_3 t + \varphi_4)^2}$ and so Equation (5.34) has the form

$$D(t) = \frac{t^2}{1+\left(t e^3_{S_1}\right)^2} + \frac{(\varphi_3 t + \varphi_4)^2}{(\varphi_1 t + \varphi_2)^2 + {e^3_{S_2}}^2 (\varphi_3 t + \varphi_4)^2} \tag{5.42}$$

Finding the optimal value of t involves taking the derivative Equation (5.42) in terms of t, collecting terms over a common denominator and equating the resulting numerator to 0 as

$$\begin{aligned} & t\left((\varphi_1 t + \varphi_2)^2 + {e^3_{s_2}}^2 (\varphi_3 t + \varphi_4)^2\right)^2 \\ & -(\varphi_1 \varphi_4 - \varphi_2 \varphi_3)\left(1 + \left(t e^3_{s_1}\right)^2\right)^2 (\varphi_1 t + \varphi_2)(\varphi_3 t + \varphi_4) = 0 \end{aligned} \tag{5.43}$$

The six roots of Equation (5.43) are solved for and Equation (5.42) is evaluated at the real parts of these roots to determine the value of t which corresponds to a minimum. The asymptotic value of Equation (5.42) should also be evaluated as $t_{\rightarrow\infty}$ which corresponds to an epipolar line in the first image of $e^3_{s_1}\check{x}_{s_1}=1$ [253].

The optimal t is used to determine $\hat{\boldsymbol{x}}_{s_1}$ and $\hat{\boldsymbol{x}}_{s_2}$ as

$$\hat{\boldsymbol{x}}_{s_1}=\Phi\left(\boldsymbol{T}_1^{pm-1}\boldsymbol{R}_1^{pmT}\begin{bmatrix} t^2e^3_{s_1} \\ t \\ \left(te^3_{s_1}\right)^2+1 \end{bmatrix}\right) \tag{5.44}$$

$$\text{and } \hat{\boldsymbol{x}}_{s_2}=\Phi\left(\boldsymbol{T}_2^{pm-1}\boldsymbol{R}_2^{pm}\begin{bmatrix} e^3_{s_2}(\varphi_3t+\varphi_4)^2 \\ -(\varphi_1t+\varphi_2)(\varphi_3t+\varphi_4) \\ (\varphi_1t+\varphi_2)^2+{e^3_{s_2}}^2(\varphi_3t+\varphi_4)^2 \end{bmatrix}\right) \tag{5.45}$$

Note that the inverse of the rigid transformations are applied during Equations (5.44) and (5.45) such that $\hat{\boldsymbol{x}}_{s_1}$ and $\hat{\boldsymbol{x}}_{s_2}$ are in the original ideal sensor CSs.

5.3.8 Linear triangulation method

Denoting the n^{th} row of $\boldsymbol{Q}_{c_j}$ as $\boldsymbol{q}^n_{c_j}$ the projection of $\hat{\boldsymbol{x}}_w$ to $\hat{\boldsymbol{x}}_{s_1}$ is described by the following three equations

$$\alpha_1\hat{x}_{s_1}={\boldsymbol{q}^1_{c_1}}^T\hat{\boldsymbol{x}}_w \qquad \alpha_1\hat{y}_{s_1}={\boldsymbol{q}^2_{c_1}}^T\hat{\boldsymbol{x}}_w \qquad \text{and} \qquad \alpha_1={\boldsymbol{q}^3_{c_1}}^T\hat{\boldsymbol{x}}_w \tag{5.46}$$

Similarly, for $\hat{x}_{s_2}$ this is given as

$$\alpha_2\hat{x}_{s_2}={\boldsymbol{q}^1_{c_2}}^T\hat{\boldsymbol{x}}_w \qquad \alpha_2\hat{y}_{s_2}={\boldsymbol{q}^2_{c_2}}^T\hat{\boldsymbol{x}}_w \qquad \text{and} \qquad \alpha_2={\boldsymbol{q}^3_{c_2}}^T\hat{\boldsymbol{x}}_w \tag{5.47}$$

Using the third relation of Equations (5.46) and (5.47) to remove the scaling variables, α_1 and α_2, the following relation is obtained.

$$\begin{bmatrix} \hat{x}_{s_1}{\boldsymbol{q}^3_{c_1}}^T-{\boldsymbol{q}^1_{c_1}}^T \\ \hat{y}_{s_1}{\boldsymbol{q}^3_{c_1}}^T-{\boldsymbol{q}^2_{c_1}}^T \\ \hat{x}_{s_2}{\boldsymbol{q}^3_{c_2}}^T-{\boldsymbol{q}^1_{c_2}}^T \\ \hat{y}_{s_2}{\boldsymbol{q}^3_{c_2}}^T-{\boldsymbol{q}^2_{c_2}}^T \end{bmatrix}\hat{\boldsymbol{x}}_w=\boldsymbol{S}\hat{\boldsymbol{x}}_w=0 \tag{5.48}$$

$\hat{\boldsymbol{x}}_w$ is computed, using singular value decomposition, as the unit eigenvector of $\boldsymbol{S}^T\boldsymbol{S}$ associated with the smallest eigenvalue [253].

5.3.9 Displacement transformation

Consider a single subset pair and its two sets of in-plane displacements determined by temporal matching between the reference and deformed image pairs under consideration. First $\boldsymbol{x}^d_{s_j}=\begin{bmatrix} x^d_{s_j} & y^d_{s_j}\end{bmatrix}^T$ is calculated based on $\boldsymbol{x}^o_{s_j}=\begin{bmatrix} x^o_{s_j} & y^o_{s_j}\end{bmatrix}^T$ as

$$\begin{bmatrix} x^d_{s_j} \\ y^d_{s_j} \end{bmatrix}=\begin{bmatrix} x^o_{s_j} \\ y^o_{s_j} \end{bmatrix}+\begin{bmatrix} u_j \\ v_j \end{bmatrix} \tag{5.49}$$

Thereafter $\boldsymbol{x}^o_{s_j}$ and $\boldsymbol{x}^d_{s_j}$ are undistorted to obtain $\check{\boldsymbol{x}}^o_{s_j}=\begin{bmatrix} \check{x}^o_{s_j} & \check{y}^o_{s_j}\end{bmatrix}^T$ and $\check{\boldsymbol{x}}^d_{s_j}=\begin{bmatrix} \check{x}^d_{s_j} & \check{y}^d_{s_j}\end{bmatrix}^T$ respectively. This is performed using non-linear, least-squares optimisation since an exact solution for the inverse of Equation (5.3) does not exist because it requires determining the roots of a polynomial of degree greater than four [220]. Triangulation, represented by function Ψ, determines the 3D coordinates in the world CS to which the reference subset pair, $\hat{x}^o_w = \begin{bmatrix} \hat{x}^o_w & \hat{y}^o_w & \hat{z}^o_w \end{bmatrix}^T$, and investigated subset pair, $\hat{x}^d_w = \begin{bmatrix} \hat{x}^d_w & \hat{y}^d_w & \hat{z}^d_w \end{bmatrix}^T$, correspond as

$$\hat{x}^o_w = \Psi\left(\check{x}^o_{s_1}, \check{x}^o_{s_2}\right) \qquad \text{and} \qquad \hat{x}^d_w = \Psi\left(\check{x}^d_{s_1}, \check{x}^d_{s_2}\right) \tag{5.50}$$

Finally, the displacements $\hat{u}_w$, $\hat{v}_w$ and $\hat{w}_w$ are determined as

$$\begin{bmatrix} \hat{u}_w \\ \hat{v}_w \\ \hat{w}_w \end{bmatrix} = \begin{bmatrix} \hat{x}^d_w \\ \hat{y}^d_w \\ \hat{z}^d_w \end{bmatrix} - \begin{bmatrix} \hat{x}^o_w \\ \hat{y}^o_w \\ \hat{z}^o_w \end{bmatrix} \tag{5.51}$$

5.4 Implementation

ADIC3D's framework, presented in Appendix A[18], is invoked from MATLAB's command prompt as "`ProcData=ADIC3D(FileNames1,FileNames2, Mask, GaussFilt, StepSize, SubSize, SubShape, SFOrder, RefStrat, StopCritVal, WorldCTs, ImgCTs);`" with the input variables defined in Table 5.1.

Table 5.1: Required input variables of the ADIC3D framework.

Variable	**Variable description**
`FileNames1`	Cell array of character vectors containing the image file names of the first image series. All images need to be the same size.
`FileNames2`	Cell array of character vectors containing the image file names of the second image series. All images need to be the same size.
`Mask`	Logical matrix, which is the same size as the images, indicating which pixels should not be analysed during correlation.
`GaussFilt`	Define the standard deviation and window size for the Gaussian filter in pixels as [`FiltSigma`,`FiltSize`] respectively where $\{\texttt{FiltSigma} \in \mathbb{R}^+ \mid \texttt{FiltSigma} > 0\}$ and $\{\texttt{FiltSize} \in \mathbb{N}\}$.
`StepSize`	Step size in pixels $\{\texttt{StepSize} \in \mathbb{N}\}$.
`SubSize`	Subset size in pixels $\{\texttt{SubSize} = 2k+1 \mid k \in \mathbb{N}\}$.
`SubShape`	Subset shape $\{\texttt{SubShape} \in \texttt{'Square'}, \texttt{'Circle'}\}$.
`SFOrder`	Dictates the SF order $\{\texttt{SFOrder} \in \mathbb{Z} \mid 0 \leq \texttt{SFOrder} \leq 2\}$.
`RefStrat`	Logical statement dictating reference image strategy (Section 5.3.6.1).
`StopCritVal`	Defines the stopping criterion value $\{\texttt{StopCritVal} \in \mathbb{R} \mid \texttt{StopCritVal} > 0\}$.
`WorldCTs`	Location of CTs in the world CS defined according to MATLAB's `estimateCameraParameters` function.
`ImgCTs`	Location of CTs in the distorted sensor CS defined according to MATLAB's `estimateCameraParameters` function.

ADIC3D returns a structured array `ResultData` containing four main fields: structured variable `Stereo` holding stereo matching data as detailed in Table 5.2; structured arrays `ProcData1` and `ProcData2`, detailed in Table 5.3, holding the processing data for temporal matching of subsets of the first and second image

---

[18] The codes are not included in appendices within this dissertation due to page limits. The code can be accessed at: github.com/SUMatEng/ADIC3D

series respectively; and structured array `DispTrans` containing displacement transformation data detailed in Table 5.4.

Table 5.2: Accessing the stereo matching data contained in `ResultData.Stereo` for subset `q`.

`P(b,q)`	SFPs (`b`=1 for u and `b`=7 for v).
`C(q)`	ZNCC coefficient.
`Iter(q)`	Number of iterations until stopping criterion is satisfied (maximum of 100 iterations).
`StopVal(q)`	Final stopping criterion value for subset q.

Table 5.3: Accessing the processing data for temporal matching (`ResultData.ProcData1(d)` and `ResultData.ProcData2(d)`) for image `d` and subset number `q`.

Variable	**Variable description**
`ImgName`	Deformed image name.
`ImgSize(b)`	Image size (`b`=1 for rows and `b`=2 for columns).
`ImgFilt(b)`	Standard deviation (`b`=1) and window size (`b`=2) for the Gaussian filter respectively in pixels.
`SubSize(q)`	Subset size in pixels.
`SubShape(q,:)`	Subset shape.
`SFOrder(q)`	SF order.
`Xos(b,q)`	Reference subset position in the distorted sensor CS of the relevant camera (`b`=1 for x^o and `b`=2 for y^o).
`P(b,q)`	SFPs (`b`=1 for u and `b`=7 for v).
`C(q)`	ZNCC coefficient.
`Iter(q)`	Number of iterations until stopping criterion is satisfied (maximum of 100 iterations).
`StopVal(q)`	Final stopping criterion value.

Table 5.4: Accessing the displacement transformation data for image `d` (contained in `ResultData.DispTrans(d)`) for subset `q`.

Variable	**Variable description**
`Xow(b,q)`	Reference subset position in the world CS (`b`=1 for $\hat{x}^o_w$, `b`=2 for $\hat{y}^o_w$ and `b`=3 for $\hat{z}^o_w$).
`Uw(b,q)`	Displacement in the world CS (`b`=1 for $\hat{u}_w$, `b`=2 for $\hat{v}_w$ and `b`=3 for $\hat{w}_w$).
`CamParams`	Calibration parameters (extrinsic, intrinsic and radial distortion parameters).

Refer to Appendix A.1 for brief discussion of utilising parallel processing to reduce computation time of ADIC3D.

5.4.1 `ADIC3D` function

The main function `ADIC3D`, outlined in Table 5.5, sets up the DIC problem and calls the appropriate subroutines. The output variables are preassigned in lines 9-12 to save input data and efficiently store computed data. `ADIC3D` allows subsets to be analysed in a flexible manner by assigning `Xos`, `SubSize`, `SubShape` and `SFOrder` on a per-subset and per-image basis. However, despite being capable of this, `ADIC3D` assigns the same `SubSize`, `SubShape` and `SFOrder` to each subset as is conventional for DIC. Subsets containing invalid pixels, identified by `Mask`, are eliminated.

Table 5.5: `ADIC3D` algorithm summary.

Line Numbers	Task Performed
Lines 2–3	Compute image names of both image series;
Lines 4-5	Compute number of images and size of the first image in first image series;
Lines 6-7	Create regularly spaced reference subset positions, `Xos`;
Line 8	Remove subsets containing invalid pixels which are defined by `Mask`;
Line 9-12	Pre-assign `ResultData` structure array;
Line 13	Call subroutine `StereoMatch` to perform stereo matching;
Line 15	Call subroutine `ImgCorr` to perform temporal matching of first image series;
Line 17	Call subroutine `ImgCorr` to perform temporal matching of second image series;
Line 18	Call subroutine `CSTrans` to perform displacement transformation from the distorted sensor CSs to the world CS;

The subroutine `StereoMatch` is called to perform stereo matching. `StereoMatch` has input variables `n` (the total number of images in each image series), `ResultData`, `FileNames1`, `FileNames2` and `StopCritVal`. It returns the subset positions in the FIS2 stored as `ProcData2(d).Xos` and stereo matching correlation results `P`, `C`, `Iter` and `StopVal` stored in `Stereo`.

Temporal matching is performed on each image series separately in lines 15 and 17 using the `ImgCorr` subroutine. `ImgCorr`'s inputs are `n`, `ProcData1` or `ProcData2`, `FileNames1` or `FileNames2`, `RefStrat` and `StopCritVal`. It returns `P`, `C`, `Iter` and `StopVal` for each subset throughout the image series stored in `ProcData1` or `ProcData2` for the first and second image series respectively.

Finally, subroutine `CSTrans` determines coordinates and displacements in the world CS based on inputs `n`, `ResultData`, `WorldCTs`, `ImgCTs` and `RefStrat`. `CSTrans` returns `Xow`, `Uw` and `CamParams` stored in `DispTrans`. Note that within the subroutines `ResultData` is shortened to `RD` and `ProcData` is shortened to `PD` where appropriate.

5.4.2 Correlation implementation

Both stereo and temporal matching make use of correlation, detailed in Section 5.3.3, which is implemented using the three subroutines: `SubShapeExtract`, `SFExpressions` and `SubCorr`. These subroutines are discussed briefly since they are identical to those used in ADIC2D [191].

SubShapeExtract, detailed in Table 5.6, determines f_i, $\nabla \boldsymbol{f}_i$ and $\Delta \boldsymbol{x}_i$ of a subset based on inputs `SubSize`, `SubShape`, `Xos`, F, ∇F and `SubExtract`. Note that variables with subscript i refer to the full set of this variable for the subset (i.e., f_i refers to $f_i \forall i \in I$) throughout Section 5.4.2. ∇F is the light intensity gradient of the reference image whereas `SubExtract` is an anonymous function which extracts a square subset from a matrix based on the size and position of the subset. For `SubShape` defined as `'Circle'`, `SubSize` specifies the diameter.

Table 5.6: `SubShapeExtract` algorithm summary.

Line Numbers	**Task Performed**
Line 2	**switch** `SubShape`;
Line 3	**case** `SubShape`=`'Square'`, **do**
Line 4-6	Extract f_i and ∇f_i using `SubExtract`;
Line 7	Compute Δx_i using `SubSize`;
Line 8	**case** `SubShape`=`Circle`, **do**
Line 9-11	Extract f_i and ∇f_i using `SubExtract`;
Line 12	Compute Δx_i using `SubSize`;
Line 13	Determine mask of elements that fall within the circular subset;
Line 14-16	Use mask to extract appropriate data for circular subset;
Line 17	**end switch**

`SFExpressions` returns anonymous functions for expressions that are dependent on the SF order, as outlined in Table 5.7. These anonymous functions are as follows: `W` defines Equation (5.9); `dFdWdP` defines $\nabla f_i \frac{\partial W_i}{\partial P}$; `SFPVec2Mat` defines Equation (5.22); `Mat2SFPVec` extracts the SFPs from `SFPVec2Mat`; and `StopCrit` defines Equation (5.23). ADIC3D allows for the zero, first or second-order SFs to be used by setting `SFOrder` to 0, 1 or 2 respectively.

Table 5.7: `SFExpressions` algorithm summary.

Line Numbers	**Task Performed**
Line 2	**switch** `SFOrder`
Line 3–8	**case** `SFOrder`=0, **do** assign functions for zero-order SF;
Line 9–14	**case** `SFOrder`=1, **do** assign functions for first-order SF;
Line 15–20	**case** `SFOrder`=2, **do** assign functions for second-order SF;
Line 21	**end switch**

`SubCorr` performs the core operations of correlation for a subset as outlined in Table 5.8. Its inputs are the interpolation coefficients of the deformed image, f_i, ∇f_i, `SubSize`, `SFOrder`, `Xos`, Δx_i, initial estimates for `P` and `StopCritVal`. `SubCorr` returns `P`, `C`, `Iter` and `StopVal`. The size of vector `P` is consistent with that of the second-order SF. Thus, the elements of $\boldsymbol{P}^{SF2}$, not relevant to the specified SF order, remain zero.

`SubShapeExtract` is called prior to calling `SubCorr` in order to determine data used by `SubCorr` while `SFExpressions` is called within `SubCorr` to define the appropriate anonymous functions on a subset basis. Thus, the effect of the subset shape and SF order as well as the core mathematical operations of correlation can be considered separately.

Table 5.8: `SubCorr` **algorithm summary.**

Line Numbers	**Task Performed**
Line 2	Call `SFExpressions` to assign equations dependent on the SF order;
Line 3	Compute $\nabla f_i \frac{\partial W_i}{\partial P}$;
Line 4	Compute $\boldsymbol{H}^{-1}$, Equation (5.19);
Line 5	Compute $\bar{f}$ and $\tilde{f}$;
Line 6	Initialise `flag`$\leftarrow$0, `Iter`$\leftarrow$1 and $\Delta\boldsymbol{P}\leftarrow$1;
Line 7	**while** "`flag`"=0, **do**
Line 8	Compute $\Delta \boldsymbol{x}_i'$, Equation (5.7), using estimates of $\boldsymbol{P}$;
Line 9	Compute $\boldsymbol{g}$ using interpolation coefficients;
Line 10	Compute $\bar{g}$ and $\tilde{g}$;
Line 11	Compute $\|\Delta\boldsymbol{P}\|$ using Equation (5.23);
Line 12	**if** $\|\Delta\boldsymbol{P}\|<$`StopCritVal` or `Iter`$\geq$100, **do**
Line 13	Set `flag`$\leftarrow$1;
Line 14	Compute `C`, Equation (5.12) substituted into Equation (5.15);
Line 15	**else, do**
Line 16	Compute $\boldsymbol{J}$, Summation expression of Equation (5.18);
Line 17	Compute $\Delta\boldsymbol{P}$, Equation (5.18);
Line 18	Update $\boldsymbol{P}$, Equation (5.21);
Line 19	**end if**
Line 20	Set `Iter`$\leftarrow$`Iter`+1;
Line 21	**end while**

5.4.3 Stereo matching implementation

Excluding the subroutines used to perform correlation, stereo matching is implemented using two additional subroutines: `FeatureMatch` and `StereoMatch`. `FeatureMatch` returns SFP initial estimates, determined according to the feature matching method outlined in Section 5.3.6.2, based on inputs `ProcData1`, `d` (the image to be analysed), `F`, `G` and `SubExtract`. `StereoMatch` refines these SFPs through correlation and uses them to determine the subset positions in the FIS2.

5.4.3.1 `StereoMatch` function

`StereoMatch`, detailed in Table 5.9, loads the FIS1 and FIS2 as `F` and `G`, respectively, and performs Gaussian filtering on these using MATLAB's `imgaussfilt` function. Lines 4 and 6 can be changed to use alternative filtering methods. Bi-cubic b-spline interpolation coefficients are determined for the deformed image using MATLAB's `griddedInterpolant` function. In line 7 '`spline`' can be replaced with '`linear`' or '`cubic`' for bi-linear or bi-cubic polynomial interpolation respectively. Moreover, line 7 can be replaced with an alternative interpolation method such as MATLAB's `spapi` function. Line 9 of `SubCorr` may need to be updated if alternative interpolation methods are employed.

Lines 11 to 16 cycles through each subset of the FIS1, for which `FeatureMatch` in line 9 determined SFP initial estimates, performing correlation using subroutines `SubShapeExtract` and `SubCorr`. These displacement SFPs are used to determine the reference subset positions in the second image series.

Subset pairs which achieve a ZNCC coefficient below 0.6, or which fall outside the bounds of the FIS2, are assumed to have failed stereo matching. These subsets are determined in lines 21-22 and their subset positions in both image series are set to NaN (not a number) in lines 23-26 such that they are not analysed during temporal matching or displacement transformation.

Table 5.9: `StereoMatch` **algorithm summary.**

Line Numbers	**Task Performed**
Line 2	Define `SubExtract` function to extract square subset data;
Line 3-6	Load the FIS1 and FIS2 and perform Gaussian filtering on them;
Line 7	Compute interpolation coefficients of `G` using MATLAB's `griddedInterpolant` function;
Line 9	Call subroutine `FeatureMatch` to determine SFP initial estimates;
Line 10	Initialise temporary storage variables used to save correlation information during the for loop;
Line 11	**for** subset number `q`=1 to number of subsets, **do**
Line 12	**if** `FeatureMatch` determined SFP initial estimates for subset `q`, **do**
/ Line 13	Call subroutine `SubShapeExtract`;
Line 14	Call subroutine `SubCorr`;
Line 15	**end if**
Line 16	**end for**
Line 17	Save correlation information to `RD.Stereo` variable;
Line 18	**for** image number `d`=1 to `d`=`n`, **do**
Line 19	Compute the subset positions in the FIS2 using Equation (5.32);
Line 20	**end for**
Line 21-22	Determine subsets which fail stereo matching and subsets which pass;
Line 23-26	Set the subset position of subsets which fail stereo matching to NaNs throughout both image series;
Line 27	Display information for stereo matching;

`StereoMatch` returns `ResultData` with the subset positions in the FIS2 assigned to `ProcData2(d).Xos`; the correlation results for stereo matching (`P`, `C`, `Iter`, `StopVal`) are assigned to `Stereo`.

5.4.3.2 `FeatureMatch` function

`FeatureMatch`, outlined in Table 5.10, implements SIFT feature matching using codes available from the VLFeat open-source library of computer vision algorithms. More specifically, VLFeat's `vl_sift` function is used to determine the keypoint locations and descriptors of `F` and `G` in lines 3-4. The keypoints of `F` which fall within the perimeter of at least one subset (square subsets equivalent in size to that specified for the subset under consideration are used) are identified in line 5. Only these keypoints of `F` are retained, in line 6, in order to reduce the computation time of VLFeat's `vl_ubcmatch` function in determining matching keypoint pairs between `F` and `G` in lines 7-8.

For each subset, the 20 closest keypoints to its centre position are identified in line 9. The for loop of line 13 cycles through the subsets using MSAC to determine reliable affine transformation parameters which are used by Equation (5.31) to compute the corresponding second-order SFPs.

MSAC is implemented using MATLAB's `ransac` function. Two anonymous functions are used in the call to `ransac`. `RansacModel`, defined in line 10, determines the affine transformation parameters from the keypoint locations by solving Equation (5.29). `RansacError`, defined in line 11, evaluates Equation (5.30) to determine each keypoint's squared error distance (for keypoints identified to be relevant to the subset). The fifth argument of `ransac` sets τ=1 pixel^2 and the final argument specifies the confidence that the maximum number of inliers are determined. The fifth and sixth arguments of `ransac`, as well as the number of nearest keypoints used for each subset (line 9), can be altered to suit the reader's needs. The try statement of line 14 catches instances where `ransac` fails to identify three keypoint inliers. As such, the SFP initial estimates of these subsets, stored in `P`, remain NaN as initialised on line 12.

Table 5.10: `FeatureMatch` **algorithm summary.**

Line Numbers	**Task Performed**
Line 2	**if** VLFeat library is not setup, then return an error;
Line 3-4	Compute keypoint locations and descriptors for `F` and `G` using `vl_sift`;
Line 5	Determine vector `KptsInVacinity` identifying the keypoints of `F` which fall within the perimeter of the square subsets, equivalent in size to that specified for the subset under consideration, of `F`;
Line 6	Eliminate keypoints (and their associated descriptors) of `F` which do not fall within the perimeter of any of the subsets;
Line 7-8	Determine matching keypoints using `vl_ubcmatch`;
Line 9	Determine the 20 nearest keypoints for each subset stored in matrix `relevantKpts`;
Line 10-11	Define anonymous functions `RansacModel` and `RansacError` for determining affine transformation parameters (of Equation (5.29)) and evaluating Equation (5.30) respectively;
Line 12	Initialise `P` as NaNs;
Line 13	**for** subset number `q`=1 to number of subsets, **do**
Line 14	**try**
Line 15	Use MSAC to determine affine transformation parameters for subset `q` from its relevant keypoints;
Line 16	Convert affine transformation parameters to second-order SFPs using Equation (5.31);
Line 17	**end try**
Line 18	**end for**
Line 19	Display information for SIFT feature matching;

5.4.4 Temporal matching implementation

Excluding the subroutines used to perform correlation, temporal matching is implemented using `PCM` and `ImgCorr`. ADIC3D's implementation of temporal matching is discussed briefly since it is almost identical to that of ADIC2D [191] with the exception that `PCM` and `ImgCorr` are modified to avoid analysing subsets which fail stereo matching.

5.4.4.1 `PCM` function

`PCM` determines an initial estimate of the displacement SFPs of a subset based on inputs `F`, `G`, `SubSize`, the subsets position and `SubExtract` as outlined in Table 5.11. Subset positions are passed as separate x and y positions as required by

MATLAB's `arrayfun` function which is used to call `PCM`. The if statement of line 2 avoids analysing subsets which contain NaN positions and returns NaN displacements in this case.

Table 5.11: `PCM` algorithm summary.

Line Numbers	**Task Performed**
Line 2	**if** subset positions do not contain NaNs, **do**
Line 3	Compute normalised cross-power spectrum in the frequency domain;
Line 4	Convert back to spatial domain;
Line 5	Find index of the maximum correlation coefficient;
Line 6	Compute index vector which relates indices of the correlation coefficient matrix to the displacements they correspond to;
Line 7–8	Obtain displacements using index of the maximum correlation coefficient;
Line 9	**else**, **do**
Line 10	Set output displacements to NaN;
Line 11	**end if**

5.4.4.2 `ImgCorr` function

`ImgCorr`, outlined in Table 5.12, makes use of a nested for loop to perform temporal matching. The outer loop cycles through the images of an image series, starting at the second image, preparing data for correlation. This includes: loading the appropriate reference and deformed images; filtering the images; determining the interpolation coefficients of G; determining the reference image gradient; shifting the reference subset positions by the displacement SFPs determined in the previous correlation run (these displacement SFPs are rounded down such that the reference subset need not be interpolated [222]); and using `PCM` to determine initial estimates of the SFPs for the current correlation run. In the case of the absolute reference image strategy the reference subset positions are not shifted and `PCM` is only used to obtain an initial estimate of the displacement SFPs for the first correlation run. For subsequent iterations the SFPs of the previous correlation run are used as initial estimates.

The inner for loop cycles through the subsets calling `SubShapeExtract` and `SubCorr` to perform correlation of the subsets. The if statement of line 19 ensures only subset which pass stereo matching are analysed within the for loop.

Table 5.12: `ImgCorr` **algorithm summary.**

Line Numbers	**Task Performed**
Line 2	Define `SubExtract` function to extract square subset data;
Line 3	**for** image number `d`=2 to `d`=`n`, **do**
Line 4	Define G;
Line 5	Perform Gaussian filtering on G using MATLAB's `imgaussfilt` function;
Line 6	Compute interpolation coefficients of G using MATLAB's `griddedInterpolant` function;
Line 7	**if** first image of correlation run **or** `RefStrat`=1 , `do`
Line 8	Define F;
Line 9	Perform Gaussian filtering on F using MATLAB's `imgaussfilt` function;
Line 10	Compute gradients for F (compute ∇F);
Lines 11–12	Displace `Xos` with previous image correlation run displacement SFPs (incremental strategy);
Line 13	Call subroutine `PCM` to compute initial estimates of displacement SFPs;
Line 14	**else**, **do**
Line 15	Set `P(d)`←`P(d-1)` ;
Line 16	**end if**
Line 17	Initialise temporary storage variables used to save correlation information during the inner loop;
Line 18	**for** subset number `q`=1 to number of subsets, **do**
Line 19	**if** subset `q` passed stereo matching, **do**
Line 20	Call subroutine `SubShapeExtract`;
Line 21	Call subroutine `SubCorr`;
Line 22	**end if**
Line 23	**end for**
Line 24	Save correlation information to `PD` variable;
Line 25–26	Display results for image `d` correlation;
Line 27	**end for**

5.4.5 Displacement transformation implementation

Displacement transformation is implemented using subroutines `CSTrans` and `Triangulation`.

5.4.5.1 `CSTrans` function

`CSTrans`, outlined in Table 5.13, uses MATLAB's `estimateCameraParameters` function to determine the calibration parameters according to Section 5.3.2 and the fundamental matrix according to Section 5.3.4. The canonical form of the projection matrices are used such that the world CS is aligned with the camera CS of the first camera.

Line 6-18 cycles through image pairs of the image set performing displacement transformation according to Section 5.3.9. Line 9 identifies subsets which do not contain NaN values in either their position or SFPs (subsets which pass stereo matching) which are to be analysed during displacement transformation. MATLAB's `undistortPoints` function is used to remove distortion from the subset positions while the subroutine `Triangulation` uses polynomial and linear triangulation to determine the 3D coordinate in the world CS corresponding to a subset pair. The if statement of line 10 avoids recomputing the reference subset positions for the absolute reference image strategy.

Table 5.13: `CSTrans` **algorithm summary.**

Line Numbers	**Task Performed**
Line 2	Compute calibration parameters using MATLAB's `estimateCameraParameters` function;
Line 3-4	Compute canonical projection matrices of first and second cameras;
Line 5	Load the fundamental matrix;
Line 6	**for** image number `d`=1 to `d`=`n` , **do**
Line 7-8	Compute $\boldsymbol{x}^d_{s_1}$ and $\boldsymbol{x}^d_{s_2}$, Equation (5.49);
Line 9	Determine indices of subsets which do not contain NaNs in either $\boldsymbol{x}^d_{s_1}$ or $\boldsymbol{x}^d_{s_2}$;
Line 10	**if** first image pair of image set or `RefStrat`=1, **do**
Line 11	Compute $\hat{\boldsymbol{x}}^o_w$ (for subsets identified in line 9) using the `Triangulation` subroutine and MATLAB's `undistortPoints` function;
Line 12	**else**
Line 13	Assign $\hat{\boldsymbol{x}}^o_w$ of previous image pair to current image pair;
Line 14	**end if**
Line 15	Compute $\hat{\boldsymbol{x}}^d_w$ (for subsets identified in line 9), using the `Triangulation` subroutine and MATLAB's `undistortPoints` function, and subtract $\hat{\boldsymbol{x}}^o_w$ from it to determine $\hat{u}_w$, $\hat{v}_w$ and $\hat{w}_w$ using Equation (5.51);
Line 16	Save calibration parameters;
Line 18	**end for**

`CSTrans` is susceptible to particular issues which can cause a fatal error. Refer to Appendix A.2 for a discussion of a function which improves the robustness of `CSTrans`.

5.4.5.2 `Triangulation` function

`Triangulation`, outlined in Table 5.14, cycles through the subset pairs determining their ideal coordinates in the ideal sensor CSs using the polynomial triangulation method according to Section 5.3.7, in lines 3-20 (based on the code of Lourakis [259]), before performing linear triangulation according to Section 5.3.8, in lines 21-22, to determine their 3D coordinate in the world CS.

Table 5.14: `Triangulation` algorithm summary.

Line Numbers	Task Performed
Line 2	**for** subset pair number q=1 to number of subsets, **do**
Lines 3-4	Determine translation matrices of Equation (5.35);
Line 5	Apply translation matrices to fundamental matrix according to Equation 5.36);
Lines 6-8	Compute epipoles and normalise them;
Lines 9-10	Compute rotation matrices according to Equation (5.37);
Line 11	Apply rotation matrices to fundamental matrix according to Equation (5.38);
Lines 12-13	Compute polynomial coefficients of Equation (5.43);
Lines 14-15	Compute real roots of the polynomial;
Line 16	Evaluate polynomial of Equation (5.42) at these roots;
Line 17	Determine root corresponding to minimum of Equation (5.42);
Lines 18-20	Compute ideal subset positions according to Equations (5.44) and 5.45);
Lines 21-22	Determine 3D position of noiseless subset positions using linear triangulation method;
Line 23	**end for**

5.5 Validation

The temporal matching aspect of ADIC3D has been shown to be sufficiently robust to contrast changes and noise while offering a compromise between noise suppression and spatial resolution consistent with that of established DIC algorithms [191]. Thus, Samples 1 and 2 of the stereo DIC Challenge [123], exhibiting rigid body translations which avoid introducing errors during temporal matching due to complex displacement fields, are used to assess the remaining aspects of ADIC3D in a quantitative manner. Sample 5, exhibiting complex displacement fields, assess ADIC3D in a qualitative manner.

ADIC3D was used by setting `StopCritVal`$=10^{-4}$ (limited to 100 iterations per-subset), `FiltSize`=0, `RefStrat`=0, `SubShape`=`'Square'`, `SFOrder`=1, `SubSize`=61 and `StepSize`=5. A larger subset size was used to reduce errors during temporal matching. Increasing the subset size beyond 61 pixels produced only marginal improvements.

Additionally, DICe (Version 2.0-beta.16) [167] and LaVision's StrainMaster (version 1.3) were used to analyse the samples. DICe is well-established [168, 111] and its modularity enables it to use the same subset positions in the FIS1 as ADIC3D prior to stereo matching. DICe was run using the same parameters as ADIC3D. StrainMaster, a streamlined version of LaVision's DaVis software, was used to analyse Samples 1 and 2. Its streamlined nature prevents modifying some correlation parameters such that it employed circular subsets, a stopping criterion of 0.01 and a maximum of 200 iterations. Although this prohibits its results from being directly comparable to ADIC3D, it is included to reflect the capabilities of a commercial code. All codes used identical masks for each sample.

For ADIC3D subsets achieving a ZNCC criterion below 0.8 for stereo matching were eliminated while DICe and StrainMaster have their own methods of eliminating subsets which fail stereo matching. The final sets of valid subset pairs for ADIC3D and DICe were determined in two steps such that the results of the two codes are directly comparable.

Firstly, a preliminary set of valid subset pairs is determined for each code as the subsets which achieved a ZNCC criterion above 0.8 for temporal matching. Additionally, for Samples 1 and 2 of DICe the Grubbs's test [260] was used to remove outliers, in terms of displacement error, to reduce outliers occurring in its preliminary set. Secondly, the final set of valid subset pairs for both codes was determined as the intersection of these preliminary sets based on the subset positions in the FIS1.

Successive images of all samples exhibit a jump in displacement such that PCM failed to determine SFP initial estimates. As such, subroutine `FeatureMatch` is used to obtain SFP initial estimates during temporal matching by inserting "`[PD(d).P]=FeatureMatch(PD,d,F,G,SubExtract);`" after line 16 in subroutine `ImgCorr`. DICe also used feature matching to determine SFP initial estimates.

The provided calibration image sets, of dot-grid style calibration plates, were used by each code to perform calibration. ADIC3D used the `STEP1_CalcDLTparameters` function of MultiDIC, an open-source stereo DIC software proposed by Solav et al. [251], to locate the CTs in the distorted sensor CSs. All experimental image sets were captured using the FLIR Grasshopper 2 model Gras-50S5M cameras with (Edmund Optics) lenses having a focal length of 35 mm.

5.5.1 Sample 1 and 2

Sample 1 captures images of a complex specimen geometry, consisting of a flat base plate with protruding 3D features, as it is displaced in increments of 10 mm in the x- and z-directions according to Table A.2 in Appendix A.3. The specimen is displaced by an Aerotech nano-positioning stage (model ANT130-160-XY) with a high-precision linear encoder and position feedback control such that the true displacements are known with high certainty. Sample 2 simulates the experiment of Sample 1 using the synthetic image generator documented by Balcaen et al. [252] to minimize error sources in the image set.

The world CS of each DIC code differs from that within which the displacements were imposed. Thus, for each code Pythagoras's theorem is used to compute the true, $\mu_{w_q}^{true}$, and calculated, $\mu_{w_q}^{calc}$, displacement magnitudes of each subset q. The resulting errors in displacement magnitude are quantified as bias, computed as the mean of the absolute error (MAE), and variance, computed as the standard deviation of the absolute error (σ).

$$MAE = \frac{\sum_{q=1}^{Q} \left| \hat{\mu}_{w_q}^{\text{calc}} - \hat{\mu}_{w_q}^{\text{true}} \right|}{Q} \qquad \sigma = \sqrt{\frac{\sum_{q=1}^{Q} \left(\left| \hat{\mu}_{w_q}^{\text{calc}} - \hat{\mu}_{w_q}^{\text{true}} \right| - MAE \right)^2}{Q-1}} \tag{5.52}$$

Here Q is the number of valid subset pairs in the image pair analysed. For Samples 1 and 2 ADIC3D and DICe had 71715 and 66850 valid subsets, respectively, while StrainMaster had 112021 and 108085 valid subsets respectively. Tables 5.15 and 16 report errors metrics for Samples 1 and 2, respectively, for steps 3, 7 and 15 which correspond to displacement extremes.

Table 5.15: Displacement error metrics for Sample 1 reported at $\times 10^{-3}$ mm.

Step	ADIC3D		DICe		StrainMaster	
	Bias	Variance	Bias	Variance	Bias	Variance
3	47.2	9.17	37.1	23	55.5	14.1
7	44.2	7.65	41.1	21.6	50.2	13
15	61.6	13.4	57.1	33.5	77.4	20.1

Table 5.16: Displacement error metrics for Sample 2 reported at $\times 10^{-3}$ mm.

Step	ADIC3D		DICe		StrainMaster	
	Bias	Variance	Bias	Variance	Bias	Variance
3	1.55	2.48	5.14	4.59	11.3	5.89
7	1.4	2.16	4.48	4.4	3.07	6.47
15	2.35	3.52	9.92	6.86	7.03	21.2

The improved error metrics of Sample 2 relative to Sample 1 for each code, with the exception of StrainMaster's variance for step 15, reflects the impact of the reduced error sources of the synthetic image set. For Sample 1 ADIC3D performs on par with DICe (up to 27% higher bias) and StrainMaster (at least 12% lower bias). In contrast, for Sample 2 ADIC3D shows improved error metrics relative to DICe and StrainMaster (at least 69% and 54% lower bias respectively). StrainMaster's higher bias and variance are a result of it analysing more subsets in the vicinity of protruding features, which are challenging to track, which are removed for ADIC3D and DICe.

5.5.2 Sample 5

Sample 5 captures images of an ASTM tensile specimen made of Aluminium 2024-T6 as it is loaded in tension. The displacements computed by ADIC3D and DICe are directly comparable since both align their world CS with the camera CS of the first camera.

Figure 5.4 shows the $\hat{u}_w$ displacement field of image pair 1000 superimposed on the FIS1. The mean and maximum of the percentage difference of the $\hat{u}_w$, $\hat{v}_w$ and $\hat{w}_w$ displacements, between ADIC3D and DICe, are reported in Table 5.17 for selected image pairs.

Table 5.17: Percentage difference of computed displacements between ADIC3D and DICe.

Image pair	250		500		750		1000	
Metric	Mean	Max	Mean	Max	Mean	Max	Mean	Max
$\hat{u}_w$	0.0342	0.131	0.0354	0.113	0.0373	0.118	0.0374	0.122
$\hat{v}_w$	0.374	2.19	0.6	2.57	0.875	2.97	1.04	3.84
$\hat{w}_w$	0.093	0.443	0.193	0.744	0.291	0.926	0.345	1.02

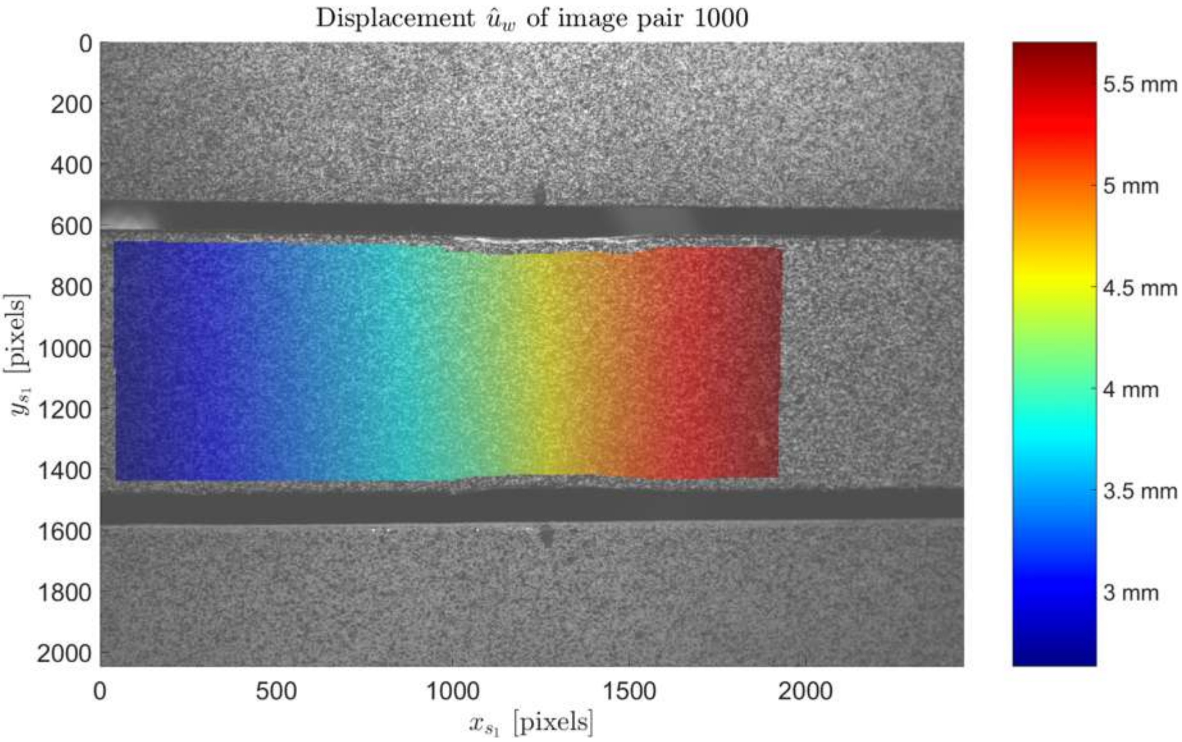


Figure 5.4: Displacement in the x-direction in the world CS superimposed on the FIS1 of the ASTM tensile specimen [123].

The cameras being positioned apart horizontally in the x-direction is reasoned to cause the larger percentage difference of $\hat{v}_w$ and $\hat{z}_w$ relative to $\hat{u}_w$ which is consistent with the findings of Balcaen et al. [105]. The low percentage difference (remaining below 2% mean and 4% maximum) indicates that ADIC3D performs consistently with DICe.

5.6 Discussion

ADIC3D's modularity is threefold. Firstly, main tasks are performed by separate subroutines, closely linked to the mathematical theory presented, allowing readers to progressively build up their understanding of ADIC3D. For instance, correlation is separated into three subroutines enabling the reader to consider: (i) how data is prepared based on the subset shape (`SubShapeExtract`); (ii) the SF order's influence on expressions for variables used during correlation (`SFExpressions`); and (iii) the core operations of correlation (`SubCorr`) separately. Furthermore, how subroutines `ImgCorr` and `StereoMatch` show how correlation is utilised differently by temporal and stereo matching.

Secondly, the subset shape, SF order, interpolation method and Gaussian filtering parameters can be readily changed. Although the influence of these on displacement results is well documented in experimental solid mechanics applications [12, 114, 115], this allows investigating their influence in novel applications to fine-tune ADIC3D's setup.

Thirdly, the SF order, subset size and subset shape can be changed on a per-subset and per-image basis enabling straightforward integration with adaptive strategies.

Adaptive strategies, which are gaining interest in the field of DIC, attempt to assign optimal parameters to each subset such that the resulting displacements are, theoretically, as reliable as possible and independent of the user.

In order for the code to retain its simplicity four aspects, identified by Pan [22] as state-of-the-art requirements, were omitted. Firstly, calibration does not use bundle adjustment, proposed by Furukawa and Ponce [261], to refine the calibration parameters . Additionally, Vo's method [262] of combing the frontal image plane and DIC to more accurately locate the CTs isn't implemented. Secondly, although the incremental reference image strategy is included, ADIC3D does not automatically employ it if the ZNCC coefficient falls below a certain threshold as recommended [263]. Thirdly, ADIC3D uses bi-cubic b-spline interpolation whereas bi-quintic b-spline interpolation is recommend [12] since increasing the degree of the interpolation method reduces the errors in the "ultimate error regime" [184] especially for smaller subsets.

Lastly, ADIC3D does not employ Pan's [182] RGDT strategy. ADIC3D's temporal matching uses the SFPs of each subset of the previous correlation run as initial estimates in the current correlation run. In contrast, RGDT only does this for the subset achieving the best ZNCC in the previous correlation run. This subset's optimised SFPs are used as an initial estimate to correlate its neighbouring subsets. Of the correlated subsets, having uncorrelated neighbours, the one with the best ZNCC has its SFPs used as an initial estimate to correlate its neighbours. This is repeated to correlate all subsets in the current correlation run. As noted with ADIC2D [191], this exclusion makes ADIC3D susceptible to propagating spurious SFPs of a subset throughout the image series.

Furthermore, RGDT is excluded such that ADIC3D is path-independent (where each subset is correlated independently of its neighbours). Thus, this avoids the need to specify seed points and the issue path-dependent methods have, with complex specimen geometries, of transferring SFPs across subsets which experience different deformation modes. This also allows ADIC3D to leverage MATLAB's parfor loop to utilise parallel processing to reduce computation time (as discussed in Appendix A.1).

Additionally, although not a start-of-the-art requirement, ADIC3D's feature matching method can be improved. Sample 1 and 2 highlight its limitation as it determines invalid SFP initial estimates for subsets in the vicinity of protruding features, that experience a large perspective change. Iniyan et al. [264] recommend using the DeepFlow [265] feature matching algorithm combined with the homography transform and MSAC.

Despite these omissions, ADIC3D's performance is on par with established codes. More specifically, Samples 1 and 2 show that ADIC3D determines displacement magnitudes in the world CS with bias and variance consistent with that of DICe and StrainMaster while Sample 5 shows that ADIC3D determines displacement vectors within 4% maximum difference of those determined by DICe. Although Samples 1 and 2 reveal limitations of ADIC3D's stereo matching, it performs sufficiently well in the presence of a complex specimen geometry.

This, coupled with ADIC2D's validation of temporal matching, and thus the correlation method, of ADIC3D indicates that ADIC3D performs sufficiently well for practical use in the field of experimental solid mechanics. Moreover, due to its modularity ADIC3D can be readily adapted to a wide range of applications across many fields. Furthermore, validation of this modular framework makes it both attractive as an education resource and as a basis to further the capabilities of DIC.

5.7 Conclusion

The theory of a subset-based, stereo DIC framework, that is predominantly consistent with current state-of-the-art techniques, and its implementation as a modular 202 line MATLAB code is presented. This framework is an extension of the previously presented ADIC2D framework and as such is also comprised of square or circular subset shape selection, assigning of Gaussian filtering parameters, altering of the interpolation method, the zero, first and second-order SFs, reference image strategy selection, the Phase Correlation Method to determine SFP initial estimates for temporal matching and calibration through using MATLAB's image calibration toolbox. ADIC3D, being capable of performing stereo DIC to determine both in-plane and out-of-plane displacements in the world CS, is additionally comprised of stereo matching, SIFT feature matching to determine SFP initial estimates for stereo matching and displacement transformation using the polynomial and linear triangulation methods. Furthermore, ADIC3D allows for assignment of SF order, subset shape and subset size on a per-image and per-subset basis enabling straightforward integration with adaptive strategies. The framework is modular enabling the reader to gain a deeper understanding of DIC as well as encouraging readers to adapt ADIC3D for new applications. Validation of the full framework coupled with its modularity makes it appealing not only as an educational resource, but also as a starting point to develop the capabilities of DIC.

5.8 Post-submission discussion

A number of insights raised during the review process where outside the scope of the paper and thus were not included in the post-review manuscript. However, they are worth considering here due to their implications.

The calibration process employed is simplistic relative to more advanced methods used in high precision optics. In addition to the limitations of the calibration process highlighted in Section 5.6, the calibration process could be improved in two distinct ways.

Firstly, many publications propose methods of more accurately locating the calibration targets within the image. Traditional calibration generally makes use of the intersection of black and white squares of a checkerboard pattern or circles aligned in a grid pattern as calibration targets. The precision to which these calibration targets can be located within an image directly affects the precision of the obtained calibration parameters. Datta et al. [266] propose using the initial calibrated camera parameters to remove aberrations due to distortions and projections to more accurately locate the calibration targets. Zhu et al. [267] focused on using polarisation filters to reduce high intensity reflected light within calibration images such that these calibration targets could be more precisely located by sub-pixel search algorithms.

In 2020, Chen and Pan [268] documented a camera calibration method, which is utilised by select commercial DIC software, that uses a synthetically generated random speckle pattern as the calibration pattern applied to the calibration plate. Using the synthetic speckle pattern as the reference image and the calibration image under consideration as the deformed image, DIC is used to locate subsets of the synthetic speckle pattern within the calibration images. Treating subsets of the speckle pattern, which contain unique speckle patterns of high information content, as calibration targets enables DIC to locate these to higher accuracy and precision than is obtainable for traditional checkerboard and circular calibration patterns. Furthermore, this approach enables the determination of more calibration targets per calibration image relative to traditional calibration patterns. As a result, this technique offers considerably more accurate and precise calibration parameters relative to existing techniques.

Secondly, more advanced camera models exist which provide a more complete description of real camera optics. In particular, by accounting for a greater variety of lens distortions within the camera model Stein [269], Cluas and Fitzgibbon [270] and Jin and Yang [271] have improved the precision of camera calibration and the resulting accuracy of the measured displacements.

Furthermore, fundamentally different camera models exist such as the generic camera model which remove constraints arising from the assumption of a pinhole camera model. Calibration based on the generic camera model maps image pixels to light rays captured by the camera system [272]. Such methods are capable of accounting for high-frequency distortions which is advantageous for high-precision optical measurement [273]. Subsequently, the ability of the generic camera model to provide a more complete description of the camera optics is acknowledged in the post-review manuscript and references to papers discussing its theory are provided for interested readers.

Despite the availability of such advanced calibration techniques, the purpose of this paper was to detail the inner workings of stereo DIC. Thus, the inclusion of a discussion of the camera model and calibration process was to establish their role in mapping the coordinates between the two sensor coordinate systems of the camera pair. The simplistic, yet well-established and widely used in DIC, pinhole camera model accounting for radial distortion combined with the calibration process based on the work of Zhang [213] and Heikkila and Silven [214] is appropriate for the purposes of this paper.

It was pointed out that the capabilities of DIC and dense optical flow methods are similar and a request was made for the inclusion of a discussion with regards to this. Although such a discussion was omitted as it fell outside the scope of the paper, it is worth considering in light of the rapid advancement of dense optical flow methods through the use of ANNs and the subsequent interest in ANN-based DIC that has recently emerged.

The use of dense optical flow for applications typically dominated by DIC has been investigated by a number of researchers. Hartmann et al. [274] investigated various traditional dense optical flow methods for experimental solid mechanics and showed that the Pyramid Horn–Schunck global method has poor accuracy. Although they showed that the the Pyramid Lucas–Kanade algorithm achieved similar accuracy to DIC, this is reasoned to be due to their similar basis (the DIC framework of this project being based on the Lucas–Kanade optical flow algorithm at its core). Similarly, an extensive analysis of various ANN-based dense optical flow methods conducted by Bouktache et al. [25] revealed that in their current form they are inappropriate for experimental solid mechanics application due to their low accuracy for sub-pixel displacements.

Although both dense optical flow and DIC are concerned with estimating displacement present within images, their differences in design limit the applicability of dense optical flow methods for traditional DIC applications. More specifically, dense optical flow is typically concerned with reliably tracking large displacements due to RBT of objects within the image. In contrast, DIC focuses on accurately tracking sub-pixel displacements as a result of deformation of a specimen. Additionally, dense optical flow typically employs dynamically assigned query points corresponding to high information content areas of the image whereas DIC makes use of a dense grid of regularly spaced query points to determine an evenly distributed displacement field. These differences in design limit the accuracy and precision achievable by both traditional and ANN-based dense optical flow methods in experimental solid mechanics applications.

Although these results show the potential of ANN-based DIC for experimental solid mechanics applications, significant work is required before such methods can be recommended. More specifically, the metrological characteristics of DIC are well established through decades of error analysis, and this knowledge is widely applicable across the various DIC algorithms. In contrast, ANN-based DIC is relatively new with limited analysis of its metrological performance and each network is likely to have fundamentally differing metrological characteristics due to their performance and generalisability being dependent on their architectural design, training dataset[19] and training decisions. Furthermore, ANN-based DIC methods do not impose regularisation, as done for local and global DIC, such that the continuity within the computed displacement field is not guaranteed. For this reason derivation of these displacement fields to obtain strain fields is ill-advised as noted by Yang et al. [93] who proposed the need to develop ANN-based DIC to directly determine strain from image pairs. Thus, despite dense optical flow methods showing great potential in traditional DIC applications, significant work is still necessary to establish their metrological characteristics before they can be recommended for practical use.

---

[19] The limited generalisability of certain ANN-based DIC methods are discussed in Section 1.4.

Chapter 6

ANN for DSS and error prediction

This chapter addresses the third objective, developing an ANN for random displacement error prediction and DSS based on purely image information. The DIC framework developed within Chapters 4 and 5 is employed to generate the database for training the ANN as well as to validate the subset size selection framework. Validation is performed on synthetic images falling within and outside the scope of the training dataset which reveal the potential and current limitations of the approach, respectively.

The coded implementation of artificial neural network based dynamic subset selection (ANNDSS) can be found at: https://github.com/SUMatEng/ANNDSS. This repository also provides an example file for running the framework on the validation image sets developed for this chapter.

This chapter has been submitted to the Springer's *Experimental Mechanics* journal under the title of *Artificial Neural Network for Dynamic Subset Selection and Displacement Error Prediction*. Although reviewer feedback has not yet been received, a post-submission discussion is included which expands on the discussion included within the paper by focusing on providing evidence for stated claims.

Declaration by candidate:

With regards to Chapter 6, the contributions of authors and co-authors are as follows:

Name	**Email**	**Contribution**	**Extent (%)**
D Atkinson (candidate)	17732913 @sun.ac.za	Conceptualisation, generation of datasets for training, testing and validation, training of the ANN, framework development and implementation, DIC processing, analysis of DIC results, figure preparation and visualisation, writing and compilation of manuscript	90
TH Becker	thorsten.becker @uct.ac.za	Supervision and reviewing, assisted in conception of framework	7
M Neaves	melzvanrooyen @sun.ac.za	Supervision and reviewing	3

Signature of candidate: [DJA] (see Footnote 10)
Date: 2022/11/07
Declaration by co-authors: The undersigned hereby confirm that:

1. The declaration above accurately reflects the nature and extent of the contribution of the candidate and the co-authors to Chapter 6,
2. No other authors contributed to Chapter 6 besides those specified above, and
3. Potential conflicts of interest have been revealed to all interested parties and that the necessary arrangements have been made to use the material in Chapter 6 of this dissertation.

Signature (see Footnote 10)	**Institutional affiliation**	**Date**
[TH Becker]	UCT	2022/11/07
[M Neaves]	SU	2022/11/07

6.1 Nomenclature

b	Counter for grey level values
β	Bit depth
d	Counter for number of datapoints of the database
η	Subset intensity variation
f	Reference subset
f^{th}	Binary threshold of reference subset
I	Speckle function
I_s	Intensity at the centre of the s^{th} speckle
I^{min}	Minimum light intensity value
I^{max}	Maximum light intensity value
i, j, k and l	Arbitrary counter variables
Λ	Intensity variation ratio
M	Subset size
M^{height}	Height of speckle image
M^{width}	Width of speckle image
ME_x	Mean error in the x-direction
ME_y	Mean error in the y-direction
M_1	Length, in pixels, of speckle pattern in the x-direction
M_2	Length, in pixels, of speckle pattern in the y-direction
M'_{max}	Maximum allowable subset size of artificial neural network based dynamic subset selection (ANNDSS)
M'	Subset size investigated by ANNDSS
N	Number of query points (counter is n)
ω	Period of sinusoidal function
p_b	Number of pixels within a speckle pattern having grey level value of b
ψ	Standard deviation of Gaussian noise
Q	Number of query points (counter is q)
R_s	Radius of the s^{th} speckle

R^{min}	Minimum speckle radius
R^{max}	Maximum speckle radius
S	Number of speckles within a speckle function (counter is s)
SE	Subset entropy
σ_x	Displacement error standard deviation (DESD) in the x-direction
σ_y	DESD in the y-direction
σ_d^{pred}	Predicted output of artificial neural network (ANN)
σ_d^{true}	True output of ANN
$\Delta\sigma_d$	Prediction error of ANN
σ	Directionless DESD
σ^{thresh}	DESD threshold value
SSE	Subset Shannon entropy
SSE^{im}	Subset Shannon entropy of the whole reference image
SSE^{sub}	Subset Shannon entropy of a reference subset
$SSSIG_x$	Sum of square of subset intensity gradient (SSSIG) in the x-direction
$SSSIG_y$	SSSIG in the y-direction
T_x	Magnitude of displacement imposed in the x-direction
T_y	Magnitude of displacement imposed in the y-direction
u	Displacement field in the x-direction
u^{RBT}	Displacement field in the x-direction for rigid body translation (RBT)
u^{sin}	Displacement field in the x-direction for sinusoidal displacement
Δu_n	Displacement error in the x-direction for the n^{th} query point
u_n^{calc}	Displacement in the x-direction for the n^{th} query point calculated by DIC
u_n^{true}	True displacement in the x-direction for the n^{th} query point
v	Displacement field in the y-direction
v^{RBT}	Displacement field in the y-direction for RBT
v^{sin}	Displacement field in the y-direction for sinusoidal displacement
Δv_n	Displacement error in the y-direction for the n^{th} query point
v_n^{calc}	Displacement in the y-direction for the n^{th} query point calculated by DIC
v_n^{true}	True displacement in the y-direction for the n^{th} query point
x and y	Image coordinates
x_s and y_s	Location of the s^{th} speckle within the image
X^{norm}	Normalised value of ANN input parameter for the database
X	Value of an arbitrary ANN input parameter
X^{min}	Minimum value of an arbitrary ANN input parameter for the database
X^{max}	Maximum value of an arbitrary ANN input parameter for the database

6.2 Introduction

Digital image correlation (DIC) can determine the full-field displacement and deformation experienced by a specimen from images captured of its surface. Advantages of DIC include its non-contact nature, ability to operate at a range of scales (from microscopy images [229, 230] to satellite imagery [231]) and full-field nature, which enables more complex material behaviour to be investigated [275, 276, 237].

This has led to its widespread adoption beyond the field of experimental solid mechanics [277, 195, 187].

The widespread adoption of DIC has prompted extensive investigation of its error sources so that methods could be developed to mitigate these, enabling the most accurate and reliable displacements to be achieved [104, 278, 12, 115]. For example, the iDICs provides a Good Practices Guide [8], part of which aims to better identify and limit errors in DIC. Despite this, the appropriate selection of DIC parameters for a given analysis remains a cumbersome task, typically achieved through trial-and-error. In particular, selecting the size of the subset is a complex problem that involves a trade-off between suppression of image noise and the ability to accurately track displacement fields with substantial gradients, termed spatial resolution (SR) [122].

Local DIC typically determines the displacement at a query point, which dictates the location of the centre of the subset, as the average of that experienced by pixels within the subset. The subset needs to be sufficiently large to incorporate enough pixels to average out the effect of, for example, image noise to minimise random errors in the displacements. At the same time, the subset needs to be small enough to avoid incorporating pixels too far from the query point, which experience incongruent displacements, to minimise systematic errors [115]. As such, subset size selection presents a trade-off to minimise the displacement error: on the one hand, maximising the subset size reduces error due to the image quality; on the other hand, reducing the subset size allows for better matching of the underlying displacement data. An optimal subset size exists, which limits both, resulting in the lowest overall displacement error. This optimal subset size is sought in most practical applications.

Although traditional DIC uses the same subset size for each query point, the speckle pattern quality and displacement field complexity typically vary spatially within an image set, suggesting a location-specific optimal subset size. Therefore, methods for determining the appropriate global subset size for an image set [279, 280, 281] have limited applicability, since the accuracy of the displacement will be dictated by the region of the image with the poorest speckle pattern quality. Methods have been proposed to weight pixels in the subset so that pixels near the query point have a higher weight, and therefore influence, than those further away from the query point [282, 183, 283]. A parameter controlling the distribution of the weights is optimised during the correlation process to essentially reduce the subset size in the presence of high displacement gradients, thereby improving the results. However, these methods cannot increase the size of the subset to improve noise suppression and require alteration of the DIC algorithm to optimise the weighting parameter.

An ideal DSS method, which dynamically appoints appropriate subset sizes for each query point, should operate as a precursor to the correlation process, allowing straightforward implementation with any DIC algorithm capable of assigning the subset size of each query point independently. Such a DSS method needs to predict appropriate subset sizes purely based on image information, since displacement field information is only available post DIC analysis.

Several methods for DSS have been proposed based solely on image quality. Kang et al. [284] proposed capturing multiple reference images and performing correlation between each unique reference image pair at the query point for a range of subset sizes. The size of the subset at which the computed displacement settles is taken as the optimal size of the subset. The appropriate subset size of each query point is taken as the mean plus one standard deviation of the settling subset sizes across all unique reference image pairs. Other authors have proposed speckle pattern quality metrics (SPQMs) (such as the Shannon entropy of the subset by Liu et al. [59], the sum of the intensity variation of the subset by Zhang et al. [20], subset entropy by Yaofeng and Pang [19] and the intensity variation ratio by Hassan et al. [21]) to quantify the quality of the speckle pattern, which is shown to be related to the resulting displacement errors, and propose determining the appropriate subset size as the smallest subset size that satisfies a threshold value. Although the respective authors show that these methods are capable of determining an optimal subset size, they lack corroboration from other authors and have not been widely adopted. More importantly, these methods are empirically based and do not allude to the displacement error to be expected in the results.

Through a theoretical error analysis, Pan et al. [60] derived a relationship between image noise, a proposed SPQM termed the sum of square of subset intensity gradient (SSSIG) and the resulting DESD. An appropriate subset size is determined as that which has sufficiently large SSSIG for the noise level of the image to return a DESD value below a threshold stipulated by the user. This relationship is derived for a basic sum of the squared difference correlation criterion and the zero-order shape function (SF). Several assumptions are made to reduce the complexity of its theoretical derivation, for example, the intensity gradient of the speckle pattern is much larger than that of image noise. As such, it is not directly applicable to most DIC instances currently in use. Additionally, this theoretical derivation tends to underpredict the DESD since it does not account for factors arising due to the practical implementation of the DIC algorithm such as the early stopping typical of DIC algorithms.

Evidently, solving the problem of DSS purely from image information is a complex task that lacks a universally accepted approach. Furthermore, since several different SPQMs are related to the resulting displacement error, an indisputable SPQM that fully encapsulates and quantifies the characteristics that constitute a favourable speckle pattern has not been established. The relationship between the speckle pattern within a subset and its resulting displacement error cannot be known exactly without such a metric.

ANNs can identify patterns in high-dimensional input data without imposing fixed relationships by creating multiple-level representations of the input data. Consequently, ANNs have been successfully utilised in the field of DIC for crack detection [285, 81], displacement computation [25], strain computation directly from images (independent of the displacement field) [93] as well as real-time stereo-DIC [94]. However, to our knowledge ANNs have not been investigated for either DSS or displacement error prediction.

This work postulates that an ANN could aid DSS by incorporating multiple SPQMs of a subset, as well as image noise, to predict its resulting DESD for various subset sizes at a query point. DSS would be based on image quality, and the ANN will inherently favour noise suppression over SR. However, it is hypothesised that the smallest subset size that offers an acceptable DESD, indicating noise suppression capacity, will offer a favourable trade-off between noise suppression and SR for up to moderate displacement gradients (common in experimental solid mechanics applications). For this reason, the paper investigates the use of ANNs for DSS. The numerical complexity in determining DESD can be overcome to include current state-of-the-art correlation algorithms and SF orders without simplifying analytical complexity. We note that the purpose of this work is not to solve the problem of subset size selection, but to investigate the potential of ANNs for displacement error prediction and DSS.

6.3 Background

In this work, it suffices to understand the role of the correlation process in determining the displacement field experienced by a grid of query points, spanning the region of interest (ROI), between the reference and deformed images representing the specimen in its undeformed and loaded states, respectively. For a detailed discussion of DIC refer to the publication of Atkinson and Becker [191] which documents the modular, open-source 2D local DIC framework used throughout this work. We also highlight the stereo adaption of the open-source 2D local DIC framework [228].

The displacement of each query point is computed by utilising a subset, which is a cluster of pixels forming a speckle pattern, centred at the query point. Unless otherwise stated, a speckle pattern refers to that contained within a subset. For each query point the correlation process optimises the parameters of the SF, which defines how the speckle pattern of the reference image can displace and deform, so that it matches the corresponding speckle pattern of the deformed image.

Throughout this work, correlation employs the first-order SF, square subsets, no image filtering, and a stopping criterion of 10^{-4} (limited to 100 iterations per subset). Furthermore, the scope of this work is limited to 2D local DIC for experimental solid mechanics applications, experiencing at most moderate displacement gradients, wherein the speckle pattern characteristics are consistent with application of spayed paint.

6.3.1 Synthetic image generation

Although advanced methods of synthetic image generation that accurately mimic the imaging process of real cameras exist, such as those of Sur et al. and Orteu et al. [286, 224], a more simplified approach was opted for due to the large number of synthetic images required to develop training database for the ANN (Section 6.4.3). Speckle functions were created according to the equation proposed by Pan et al. [287], given by

$$I(x,y)=\sum_{s=1}^{S} I_s \exp\left(\frac{-\left((x-x_s)^2+(y-y_s)^2\right)}{R_s^2}\right) \tag{6.1}$$

For the speckle function, I, there are S many speckles where the s^{th} speckle has a light intensity of I_s at its centre, located at (x_s,y_s), with a speckle radius of R_s. The speckle parameters are randomly assigned within their respective ranges as

$$\left\{ I_s \in \mathbb{R} \,|\, I^{min} < I_s < I^{max} \right\} \tag{6.2}$$

$$\left\{ x_s \in \mathbb{R} \,|\, 1 < x_s < M^{width} \right\} \tag{6.3}$$

$$\left\{ y_s \in \mathbb{R} \,|\, 1 < y_s < M^{height} \right\} \tag{6.4}$$

$$\text{and} \quad \left\{ R_s \in \mathbb{R} \,|\, R^{min} < R_s < R^{max} \right\} \tag{6.5}$$

Here M^{width} and M^{height} are the width and height of the desired synthetic image. Super-sampling of $I(x,y)$, as suggested by Orteu et al. [224], was used to mimic how charge-coupled devices (CCDs) convert continuous light into discrete pixels. A 10-by-10 grid of sampling points, evenly distributed within the boundaries of the pixel, has random displacements applied to their locations up to a magnitude of half the spacing between them. Equation (6.1) is sampled at these locations and the average is taken as the light intensity of the pixel.

The speckle function is super-sampled to obtain raw images. In this work, raw images and quantised images store the light intensity of each pixel as a double (double-precision floating-point number) and β-bit integer, respectively. A raw reference image is obtained by super-sampling the speckle function as $I(x,y)$. A corresponding raw deformed image, experiencing a displacement field of $u(x,y)$ and $v(x,y)$ relative to the raw reference image in the x- and y-direction, respectively, is obtained as $I(x-u(x,y), y-v(x,y))$. These raw images are converted to quantised images (hereafter referred to simply as images) by applying homoscedastic Gaussian noise, with a mean of zero and its severity dictated by its standard deviation (ψ), followed by quantisation into 8-bit images.

The two displacement fields used in this work are rigid body translation (RBT) given as

$$u^{RBT}(x,y,T_x) = T_x \tag{6.6}$$

and

$$v^{RBT}(x,y,T_y) = T_y \tag{6.7}$$

as well as sinusoidal displacement in the x-direction with no displacement in the y-direction as

$$u^{sin}(x,y,T_x) = sin\left(\frac{x}{\omega}\right) T_x \tag{6.8}$$

and

$$v^{sin}(x,y,T_y) = 0 \tag{6.9}$$

where, ω dictates the frequency of the sinusoid. Here T_x and T_y dictate the magnitude of the displacement imposed in the x- and y-direction, respectively. In this work, speckle pattern parameters refer to the variables of Equation (6.1), while speckle pattern characteristics refer to the attributes of the speckle pattern that are due to the speckle function used to generate synthetic images or the method used to apply the speckle patterns to the specimen for experimental images.

6.3.2 Displacement errors

Each subset correlated has an associated displacement error in each direction, represented as Δu_n and Δv_n in the in the x- and y-direction, respectively, for the n^{th} subset, computed as

$$\Delta u_n = u_n^{calc} - u_n^{true} \tag{6.10}$$

and

$$\Delta v_n = v_n^{calc} - v_n^{true} \tag{6.11}$$

where, u_n^{calc} and v_n^{calc} are the calculated, and u_n^{true} and v_n^{true} the true displacements in the x- and y-direction, respectively. The systematic error, reflecting accuracy, of a set of N displacement errors is quantified by the mean error (ME), represented as ME_x and ME_y in the x- and y-direction, respectively, as

$$ME_x = \frac{\sum_{n=1}^{N} \Delta u_n}{N} \tag{6.12}$$

and

$$ME_y = \frac{\sum_{n=1}^{N} \Delta v_n}{N} \tag{6.13}$$

The random error, reflecting precision, is quantified by the DESD, represented as σ_x and σ_y in the x- and y-direction, respectively, as

$$\sigma_x = \sqrt{\frac{N\sum_{n=1}^{N} \Delta u_n^2 - \left(\sum_{n=1}^{N} \Delta u_n\right)^2}{N(N-1)}} \tag{6.14}$$

and

$$\sigma_y = \sqrt{\frac{N\sum_{n=1}^{N} \Delta v_n^2 - \left(\sum_{n=1}^{N} \Delta v_n\right)^2}{N(N-1)}} \tag{6.15}$$

Note that in the context of the proposed ANN, DESDs are represented as σ regardless of direction since the ANN is unconcerned with the direction in which it predicts the DESD. Traditionally, the ME and DESD are computed over the set of displacement errors associated with the subsets analysed by DIC to quantify the accuracy and precision of the analysis for the image set. Although this approach is employed in Section 6.5 to validate the proposed artificial neural network based dynamic subset selection (ANNDSS) method, an alternative approach is used to quantify the noise suppression capacity of the speckle pattern.

Due to the random nature of image noise, each instance of applying noise of constant ψ to the same raw image results in a unique speckle pattern, and thus unique bias in the computed displacements, for the same query point and subset size. Thus, correlating the same subset size at the same query point N times, such that the quantised image pair analysed each time is obtained by reapplying noise of ψ to the same raw image pair, results in a set of N many Δu_n and Δv_n which fully encapsulate the influence of image noise in these directions. The DESD of these sets of Δu_n and Δv_n quantifies the noise suppression capacity, in the respective directions, of the speckle pattern of the raw image, contained within the subset, in the presence of noise of standard deviation ψ. With the exception of the σ_x and σ_y results of Section 6.5 (computed across the subsets analysed), DESD is computed according to this approach in the remainder of this article.

A Monte Carlo analysis was used to compute the DESD over a set of up to N=600 displacement errors and found that the DESD settles sufficiently within N=150, as illustrated in Figure 6.1.

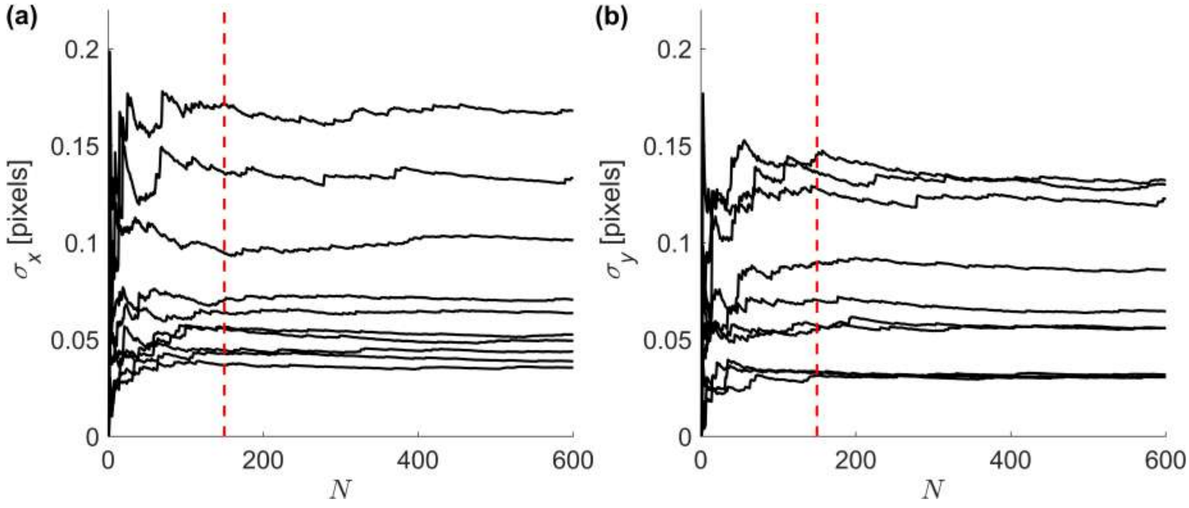


Figure 6.1: Monte Carlo analysis investigating the stability of σ_x (a) and σ_y (b) as a function of the number of displacement errors, N, in the set. Each line represents the trend for a data point randomly selected from the database created in Section 6.4.3.

Quantifying the noise suppression capability of real speckle patterns using this approach is not feasible since a set of N reference and deformed images, each exhibiting the exact same underlying displacement field, are not typically available for experimental image sets. As such, an ANN is utilised to predict the DESD from SPQMs.

6.3.3 SPQMs

The following SPQMs serve as inputs for the ANN. For more information refer to their respective publications. Pan et al. [60] proposed the SSSIG, represented as $SSSIG_x$ and $SSSIG_y$ in the x- and y-direction, respectively, is given as

$$SSSIG_x = \sum_{i=1}^{M}\sum_{j=1}^{M}\left(\frac{f(i+1,j)-f(i-1,j)}{2}\right)^2 \tag{6.16}$$

and

$$SSSIG_y = \sum_{i=1}^{M}\sum_{j=1}^{M}\left(\frac{f(i,j+1)-f(i,j-1)}{2}\right)^2 \tag{6.17}$$

where, M is the length of the square subset f of the reference image. These are related to theoretical lower limit of the DESD as

$$\sigma_x \approx \frac{\psi}{\sqrt{SSSIG_x}} \tag{6.18}$$

and

$$\sigma_y \approx \frac{\psi}{\sqrt{SSSIG_y}} \tag{6.19}$$

Yaofeng and Pang [19] proposed subset entropy (SE), shown to be related to the DESD, which measures the average of the absolute deviations in intensity of the pixels relative to their neighbours normalised by the image bit depth (β) as

$$SE = \frac{\sum_{i=1}^{M}\sum_{j=1}^{M}\left(\sum_{k=-1}^{1}\sum_{l=-1}^{1}|f(i,j)-f(i+k,j+l)|\right)}{2^{\beta}M^{2}} \tag{6.20}$$

However, Zhang et al. [20] noted that since SE relays the average, and not the total, information content of a subset it has limited potential for subset size selection. As such, they proposed the subset intensity variation (η) as a single metric which reflects the intensity gradient information contained within a speckle pattern as

$$\eta = \sum_{i=1}^{M}\sum_{j=1}^{M}\left(\sum_{k=-1}^{1}\sum_{l=-1}^{1}|f(i,j)-f(i+k,j+l)|\right) \tag{6.21}$$

Liu et al. [59] proposed subset Shannon entropy (SSE) to measure the information content within a speckle pattern, of size M_1 by M_2 pixels in the x- and y-direction, respectively, computed as

$$SSE = -\sum_{b=0}^{2^{\beta}-1}\frac{p_b}{M_1\times M_2}\log_2\left(\frac{p_b}{M_1\times M_2}\right) \tag{6.22}$$

where p_b is the number of pixels within the speckle pattern having a grey level of b. Their method of selecting the subset size relied on the ratio of the SSE of the subset (SSE^{sub}) to that of the image (SSE^{im}) and as such both are passed as inputs to the ANN. Note that for SSE^{sub} the size of the speckle pattern is given as $M_1 = M_2 = M$ while for SSE^{im} it simplifies as $M_1 = M^{width}$ and $M_2 = M^{height}$.

Hassan et al. [21] proposed the intensity variation ratio (Λ) as the ratio between the length of the edges of the speckles within a subset and the perimeter of the subset. First a binary threshold image (f^{th}) is created, where values of one and zero indicate pixels associated with speckles and background, respectively, using Otsu's method [288] which is implemented using MATLAB's `imbinarize` function . Then Λ is computed as

$$\Lambda = \frac{\sum_{i=2}^{M}\sum_{j=1}^{M}|f^{th}(i,j)-f^{th}(i-1,j)| + \sum_{i=1}^{M}\sum_{j=2}^{M}|f^{th}(i,j)-f^{th}(i,j-1)|}{4M} \tag{6.23}$$

6.4 Methodology

The presented ANNDSS method uses an ANN as an error prediction tool for the purpose of DSS. We note that the ANN's architecture, database creation and training method are documented before detailing its implementation within ANNDSS.

6.4.1 DSS based on noise suppression

The displacement errors of the correlation process are primarily caused by image noise and SR effects, reflected as random and systematic errors, respectively, the severity of which is governed by the subset size. Consider an image set that undergoes sinusoidal displacement according to Equations (6.8-6.9), with T_x=0.5 and ω=25, generated according to Section 6.3.1. Figures 6.2(a-b) illustrate the ME_x and σ_x as a function of the subset size for three query points. Figure 6.2(c) plots DESD in both directions as a function of SSSIG for a range of query points and subset sizes analysed. Note that throughout this article all plots of SSSIG vs. DESD show $SSSIG_x$ vs. σ_x and $SSSIG_y$ vs. σ_y on the same axis since their strong correlation intuitively illustrates the trend in noise suppression with increasing speckle pattern information for an image set.

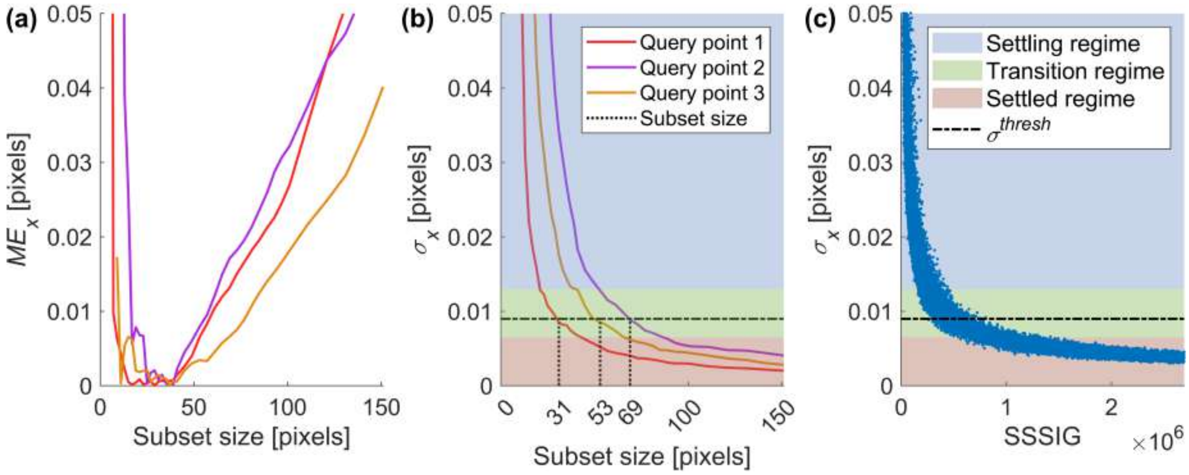


Figure 6.2: The trend in (a) ME_x and (b) σ_x with increasing subset size for three query points for a sinusoidal displacement field. Furthermore (b) illustrates how ANNDSS appoints the smallest subsets for each query point which satisfy σ^{thresh}. (c) Trend in $SSSIG_x$ vs. σ_x and $SSSIG_y$ vs. σ_y plotted on the same axis for a range of query points and subset sizes analysed. The settling, transition and settled regime are highlighted in (b) and (c).

The ME settles abruptly with subset size as the contained speckle pattern becomes distinct enough to be uniquely tracked[20]. Thereafter, the ME depends on SR as correlation attempts to determine accurate displacements in the presence of displacement gradients, which cause the speckle pattern to deform between images. Unless the deformation complexity prescribed by the SF can fully account for that experienced by the speckle pattern, spurious displacements will result. Thus, as the subset size increases, it incorporates pixels further from the query point, which experience increasingly incongruent displacements, increasing the ME due to SR effects.

Image noise causes the speckle patterns of the reference and deformed images to diverge from one another such that the correlation process computes spurious displacements in an effort to improve the match between them, as it tracks both the underlying speckle pattern and image noise. Thus, increasing the subset size, thereby increasing the contained underlying speckle pattern information, reduces the DESD as image noise is averaged out.

---

[20] When the ME settles it takes on small values because positive and negative errors cancel each other out in Equation (6.12).

The optimal subset size is that which offers the lowest overall error by ensuring a good compromise between ME and DESD. Although DSS methods exist which seek this optimal subset size (for example by applying a weighting to the pixels of a subset to effectively reduce its size to improve the ME [183, 282, 283, 289]), they have several drawbacks due to their dependence on DIC computed displacements to account for the impact of the subset size on the ME. Firstly, their reliance on DIC computed displacements and iterative nature makes then time consuming. Secondly, they require a good initial guess of the subset sizes such that the DIC computed displacements are reliable enough to guide the iterative optimisation of these subset sizes. Lastly, they require modifying the DIC algorithm to be implemented.

These drawbacks are similarly applicable to adaptive global DIC methods which based on DIC computed displacements iteratively refine the mesh, by increasing its density (h-adaptive [290]) or element orders (p-adaptive [117]), to reduce MEs. As such, the proposed DSS method is based on considering only the DESD such that it is independent of DIC computed displacements and can operate as a pre-process to any compatible DIC algorithm.

The trend in DESD with increasing speckle pattern information, regardless of whether it is measured in terms of subset size or SSSIG, can be divided into three regimes as shown in Figures 6.2(b-c): (i) a "settling" regime where the noise averaging effect causes the DESD to initially decrease rapidly as the increased speckle pattern information greatly aids noise suppression; (ii) a "transition" regime where there is a decreasing rate of improvement of the DESD with increasing speckle pattern information as the speckle pattern begins to contain enough information to sufficiently suppress image noise; and (iii) a "settled" regime experiencing marginal rate of improvement in DESD as additional speckle pattern information becomes redundant. Hence, it can be argued that for each query point the appropriate subset sizes are those with associated DESD values in the transition regime since they offer sufficient noise suppression while further increase in subset size, into the settled regime, offers only marginal improvement in noise suppression at a potentially high cost to SR.

The transition regime occurs over a narrow band of DESD values, particularly evident in Figure 6.2(c), since the speckle pattern characteristics and image noise are generally consistent across an image set. As such, ANNDSS consists of stipulating a DESD threshold, σ^{thresh}, within the transition regime and for each query point assigning the smallest subset size for which the ANN predicts a σ_x and σ_y satisfying this σ^{thresh}.

It should be noted that by not considering the displacement signal directly this approach cannot guarantee the subset size is optimal. More specifically, for the conservative σ^{thresh} shown in Figure 6.2(b-c) the subset size of 69 pixels for query point 2 would cause a significant ME despite the reasonably low DESD. However, it is hypothesised that *for an appropriate* σ^{thresh} *this approach offers a favourable compromise between noise suppression and SR for up to moderate displacement gradients since the subset is as small as possible, thus limiting errors due to SR, while offering sufficient noise suppression*.

In fact, the choice of σ^{thresh} dictates the compromise between noise suppression and spatial resolution; where a lower or higher σ^{thresh} will favour noise suppression or spatial resolution, respectively. This flexibility in the choice of σ^{thresh} is beneficial since, as noted in the Good Practices Guide [8], the preference towards noise suppression and spatial resolution is application dependent. Moreover, the expected displacement field complexity is generally known *a priori* for experimental solid mechanics applications since the specimen geometry is designed to exhibit a particular displacement field to investigate specific material properties. Furthermore, the gradient of the settled regime, which is steeper for image sets of poor speckle pattern quality, indicates the benefit of selecting a stricter σ^{thresh}. Therefore, all the necessary information is available for the appropriate selection of σ^{thresh} which affords a favourable compromise between noise suppression and spatial resolution for the analysis.

For example, knowing the displacement field of Figure 6.2 is complex, an appropriate σ^{thresh} would be at the beginning of the transition regime which for query point 2 suggests a subset size of around 53 pixels which offers a favourable compromise between ME and DESD.

6.4.2 Proposed ANN

ANNs are made up of layers of artificial neurons. Each artificial neuron determines the weighted sum of its inputs before adding a bias, applying a non-linear function and passing the result on to the next layer of artificial neurons. There are three types of layers: (i) an input layer which receives the data to be processed; (ii) the hidden layers which process the data; and (iii) the output layer, which takes the processed information from the hidden layer and formats it to the expected output of the ANN. The layers of the ANN, called the architecture, need to be designed around the intended task of the ANN. For a more in-depth discussion of ANNs, refer to Khamparia and Singh [291] as well as Basheer and Hajmeer [292].

MATLAB's `fitrnet` function (ver. 2021a) was used to create a fully-connected, feed-forward ANN to develop a regression model that predicts the DESD of a subset in a single direction from its SPQMs and image noise associated with that direction. In particular, the ANN takes as inputs the SSSIG, SE, η, SSE^{im}, SSE^{sub}, Λ and ψ. Note that either $SSSIG_x$ or $SSSIG_y$ are passed as inputs to the ANN depending on weather σ_x or σ_y is to be predicted. The ANN's architecture, which through trial-and-error was found to give favourable results, consists of 50 artificial neurons in the input layer, 4 hidden layers of 30, 30, 30 and 10 artificial neurons, respectively, and an output layer consisting of one artificial neuron. The tanh activation function was used for each neuron.

To improve the reliability and speed of training, the ANN operates on normalised inputs and outputs falling between -1 and 1. Each parameter, X, is normalised to this range, X^{norm}, as

$$X^{norm} = \frac{X - X^{min}}{X^{max} - X^{min}} \times 2 - 1 \tag{6.24}$$

where X^{min} and X^{max} are the minimum and maximum values of X in the database.

6.4.3 Database creation

Each datapoint within the database consists of a set of inputs to the ANN and the corresponding output. As much randomisation was introduced during database creation to mitigate bias since the scope of the database greatly influences the capabilities and generalisability of an ANN [292]. To ensure diversity in the resulting speckle functions used to generated image sets (Section 6.3.1), I^{min} and I^{max} were randomised between 0.1 and 1, respectively, while R^{min} and R^{max} of each speckle function were computed as

$$a = \texttt{rand}([0,1]) \times 60 + 6 \tag{6.25}$$

$$c = \texttt{rand}([0,1]) \times a \times 0.5 \tag{6.26}$$

$$R^{min} = a - c \tag{6.27}$$

$$\text{and} \quad R^{max} = a + c \tag{6.28}$$

where $\texttt{rand}([0,1])$ is MATLAB's pseudo random number generator returning a result between 0 and 1. Thus, R_s can take on values between 3 and 99 pixels; a minimum speckle size of 3 pixels was chosen to avoid aliasing [293]. Note that the intensity of each speckle function was capped at I^{max} to ensure variety in terms of contrast. To further increase the diversity of the resulting speckle functions the number of speckles, S, within each function definition were randomised to allow for coverage between 40% and 90% of the pattern.

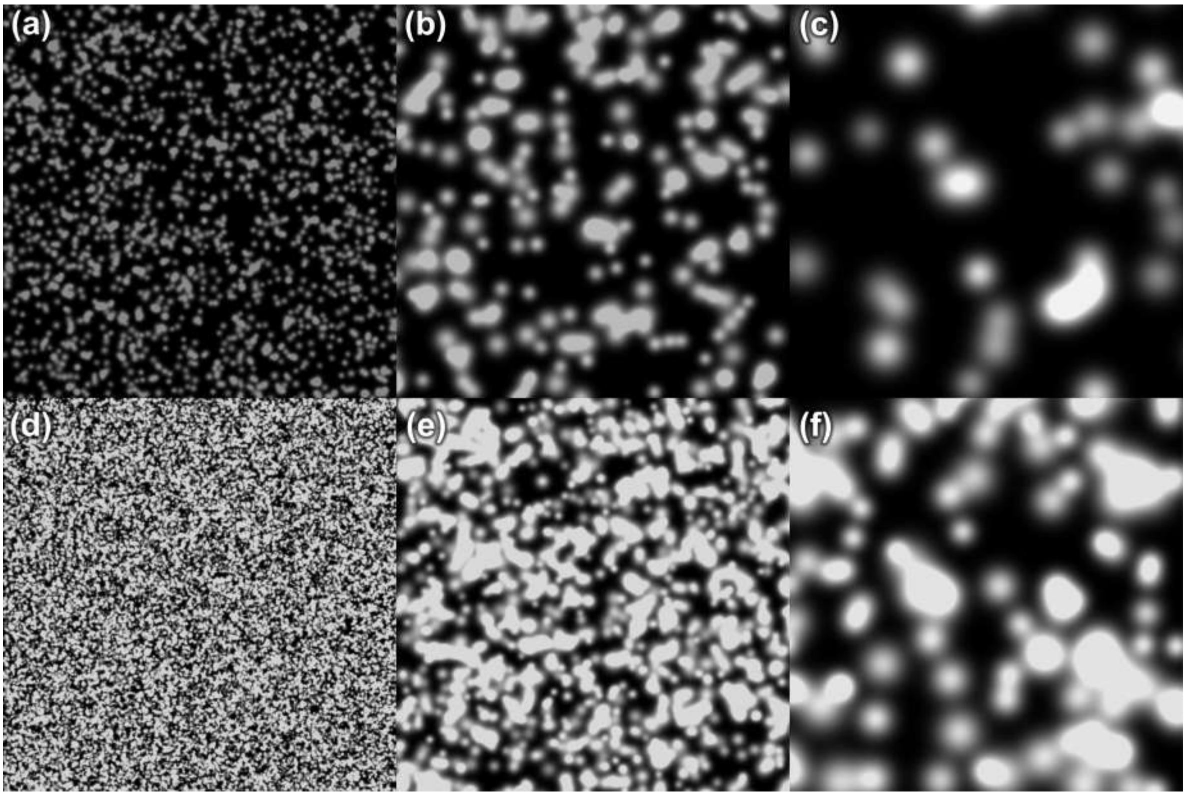


Figure 6.3: Diversity of the resulting speckle functions due to variability of speckle function parameters as documented in Table 6.1.

Using this method, 120 unique speckle functions were created, a sample of which is illustrated in Figure 6.3 with their speckle function parameters documented in Table 6.1. For each speckle function, a raw reference and deformed image of 2001-by-2001 pixels, were created by super-sampling the speckle functions as $I(x,y)$ and $I(x-u(x,y), y-v(x,y))$, respectively. Although DESD computed according to Section 6.3.2

reflects noise suppression independent of SR effects as observed in Figure 6.2(b), the RBT displacement field of Equations (6.6-6.7) was utilised to ensure the absence of displacement gradients thus eliminating any potential influence of SR on DESD. Due to DESD being independent of the imposed displacement as shown in Figure 6.4(a), corresponding to the findings of Shao et al. [294], deformed images were created with an imposed displacement of 0.5 pixels in each direction (T_x=0.5 and T_y=0.5).

Table 6.1: Details of speckle functions illustrated in Figure 6.3.

Parameters	a	b	c	d	e	f
S	21681	2744	694	216594	10335	1089
R^{min}	3.69	14.4	9.78	3.42	11.6	29
R^{max}	15.6	46.3	93.7	6.36	29.3	85.7
I^{min}	0.372	0.6	0.446	0.303	0.473	0.775
I^{max}	0.541	0.735	0.953	0.832	0.856	0.893
Coverage	43.3%	54.5%	43%	89%	87.9%	73.8%

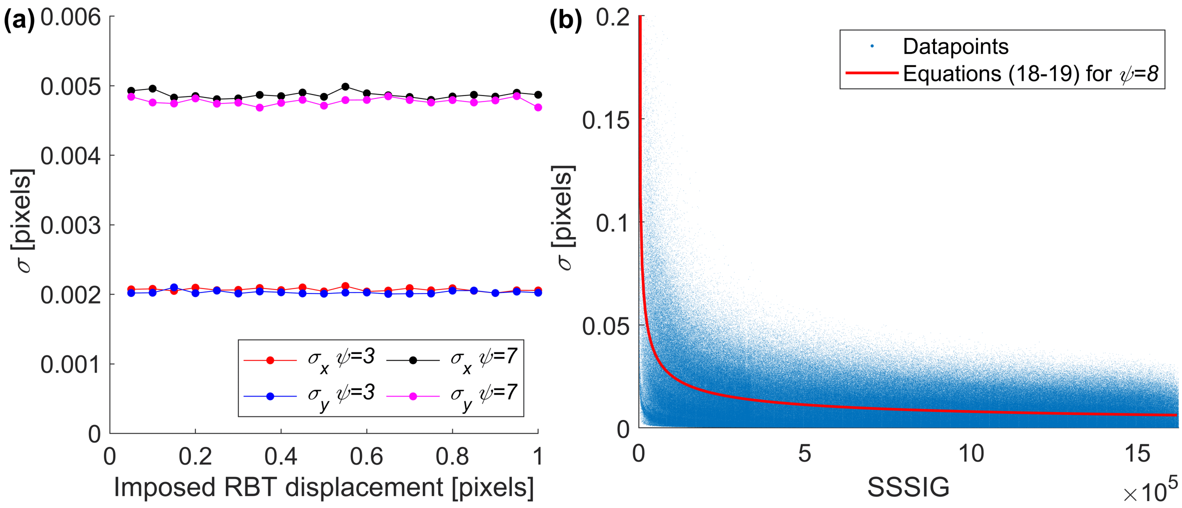


Figure 6.4: (a) σ_x and σ_y as a function of the imposed RBT displacement magnitude. (b) σ_x and σ_y of the database plotted on the same axis as a function of SSSIG. The theoretical lower limit of σ predicted by Equations (6.18-6.19) is included (in red) reflecting how true values diverge from those predicted due to practical effects of DIC implementation.

Each speckle pattern analysed has two associated datapoints; one for σ_x and one for σ_y. The inputs and outputs of the datapoint are determined by computing the SPQMs of Equations (6.16-6.17) and (6.20-6.23), image noise and noise suppression capacity in each direction (DESD). As such, for each of the 120 raw image sets, 6 batches of 8-bit quantised images, each batch containing 150 image sets of the same ψ of noise reapplied to the raw images, were created such that the noise suppression capacity of the speckle patterns can be quantified as DESD, resulting in 108 000 image sets. The standard deviation of the image noise applied to each batch was randomised within the range of $\{\psi \in \mathbb{R} | 0 \leq \psi \leq 8\}$. Each SPQM of a data point was taken as its mean within the batch of image sets since the SPQMs of each data point varied between the image sets of the same batch.

For each batch of image sets, the speckle patterns analysed were dictated by randomly placing 81 query points and for each query point appointing 18 subsets of random size within the range of $\{M{\in}\mathbb{N}|5{\leq}M{\leq}601\}$. Initial displacement estimates were randomised between -0.25 and 0.25 pixels.

Out of a potential 2 099 520 datapoints 1 661 732 were successfully correlated, while others failed due to insufficient speckle pattern information. However, datapoints were eliminated if σ_x or σ_y were above 0.2 pixels (since a $\sigma{>}0.2$ is typically undesirable), or if their SSSIG value in either direction was greater than $1.6{\times}10^6$ (since the DESD was found to settle by this value, as observed in Figure 6.4(b)). This resulted in 1 368 246 datapoints used to train, test and validate the ANN.

6.4.4 Training the ANN

The database was divided into training, testing and validation datasets in a ratio of 7:1:2, respectively. Supervised learning is used to train the ANN on a training dataset to optimise its artificial neuron weights and biases, dictating its behaviour. Training involves iteratively passing the inputs of a batch of datapoints, from the training dataset, through the ANN, which predicts σ values according to its current neuron weightings and biases. The d^{th} datapoint of this batch has a prediction error, $\Delta\sigma_d$, between the DESD predicted by the ANN, σ_d^{pred}, and the true DESD, σ_d^{true}, computed as

$$\Delta\sigma_d{=}\sigma_d^{pred}-\sigma_d^{true} \tag{6.29}$$

The gradients of the activation functions of the artificial neurons are used according to the chain rule to backpropagate these prediction errors through the ANN to determine how the weights and biases of the artificial neurons should change to improve the overall accuracy of the ANN.

Training was carried out using the `fitrnet` function of MATLAB using the limited-memory Broyden-Fletcher-Goldfarb-Shanno (LBFGS) quasi-Newton loss minimisation algorithm [295] to minimise loss which is quantified as the mean square of prediction errors. At each iteration, the mean square of prediction errors of the testing dataset is computed alongside with the training dataset. Iterations cease once the mean square of prediction errors of the testing dataset has not improved for 6 consecutive iterations. Subsequently, after 426 iterations, the ANN had a loss for the training and testing datasets of $1.2896{\times}10^{-3}$ and $1.2591{\times}10^{-3}$, respectively. The validation dataset is used in Section 6.5.1 to investigate the performance of the trained ANN for data not used during training.

6.4.5 ANNDSS implementation

Prior to the ANNDSS process, the query points are placed, according to the step size, throughout an area defined to be half the maximum allowable subset size, M'_{max}, from each edge of the ROI. Provided the ROI is placed appropriately, this ensures that the subset sizes appointed by ANNDSS do not incorporate pixels outside the limits of the image or the perimeter under consideration. If ψ of image noise is unknown, it is computed according to Immerkær's method [296].

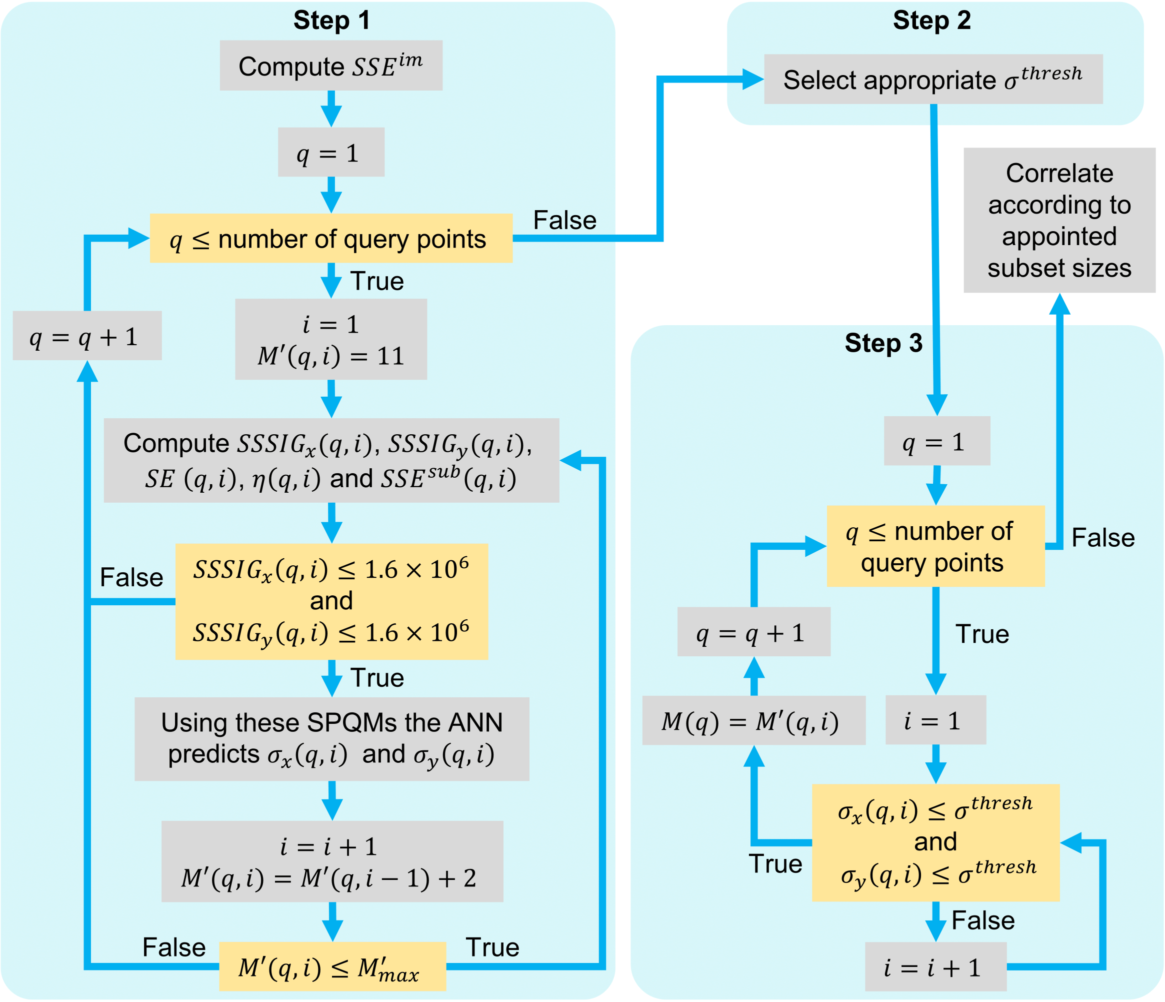


Figure 6.5: Flow diagram illustrating the process of the proposed ANNDSS method.

Thereafter the ANNDSS process, consisting of three steps as illustrated in Figure 6.5, proceeds as follows:

Step 1: Utilisation of the ANN to predict the relationship between the subset size and DESD in each direction for each query point. Once the SSE^{im} has been determined according to Equation (6.22), the SPQMs of Equations (6.16-6.17) and (6.20-6.23) are computed across a range of subsets sizes at each query point. For the q^{th} query point the i^{th} investigated subset size is represented as $M'(q,i)$. The range of subset sizes investigated for each query point starts at 11 pixels and increases by 2 pixels until either $SSSIG_x(q,i)$ or $SSSIG_y(q,i)$ exceed 1.6×10^6 (the maximum on which the ANN was trained) or $M'(q,i)>M'_{max}$. Using the associated SPQMs and ψ the ANN predicts the DESD in the x- and y-direction, represented as $\sigma_x(q,i)$ and $\sigma_y(q,i)$, respectively, for each valid subset size across all query points.

Step 2: Selection of σ^{thresh}. The $SSSIG_x(q,i)$ vs. $\sigma_x(q,i)$ and $SSSIG_y(q,i)$ vs. $\sigma_y(q,i)$ are plotted on the same axis, illustrating the trend in noise suppression, to facilitate appropriate selection of the σ^{thresh}. The DESD is represented as a function of the SSSIG since the strong correlation intuitively illustrates the trend in noise suppression, enabling straightforward identification of the settling, transition and settled regimes. If the SSSIG vs. DESD relation lacks an identifiable settled regime M'_{max} is too small and should be increased.

Step 3: Appoint the subset size of each query point, $M(q)$, as the smallest subset size for which $\sigma_x(q,i)$ and $\sigma_y(q,i)$ are below the σ^{thresh} stipulated. Thereafter, conventional correlation is performed using these appointed subset sizes.

If too high a percentage of the appointed subsets are of size equivalent to M'_{max}, then M'_{max} should be increased since it is insufficient for some of the query points to satisfy the σ^{thresh} stipulated. The coded implementation of the ANNDSS framework can be accessed on GitHub at `https://github.com/SUMatEng/ANNDSS`.

6.5 Validation

The performance of the ANN is first quantitatively evaluated as a DESD prediction tool. Subsequently, the performance of ANNDSS is quantitatively evaluated on five synthetic image sets (Cases 1-5) and qualitatively on an experimental image set (Case 6).

Table 6.2: Details of the validation image sets analysed.

Case	**1**	**2**	**3**	**4**	**5**	**6**
Generation method	Section 6.3.1	Section 6.3.1	TexGen [224]	FFT shift [225]	Binning [225]	Real image set [123]
Imposed displacement Equations	6.6-6.7	6.8-6.9	6.6-6.7	6.6-6.7	6.6-6.7	NA
T_x and T_y increment	0.1	0.1	0.05	0.1	0.1	NA
ψ	4	4	8	8	0.66	1.67
Maximum subset size	121	121	161	191	121	241
Contrast range	0-255	0-255	0-50	0-50	0-50	0-255
N	59 290	57 760	103 680	42 284	30 459	17 840

The details of Cases 1-6 are documented in Table 6.2. Cases 1-2 are generated according to Section 6.3.1, having spatially varying speckle radius, light intensity, and percentage coverage, while Samples 2, 4 and 7 of the 2D DIC Challenge (version 1.0) were analysed as Cases 3-5. Case 3 was generated using TexGen software [224], while Cases 4 and 5 use a fast Fourier transform (FFT) [225] and binning [225], respectively, to create deformed images from reference images that capture real speckle patterns. Sample 6 of the Stereo DIC Challenge [123], a 2D experimental image set of a tensile sample undergoing extension, is analysed as Case 6. Thus, Cases 1-2 and 3-6 investigate performance for speckle pattern characteristics within and outside the scope of the training dataset (hereafter referred to as "training scope"), respectively.

In addition to the subset sizes appointed by ANNDSS, each case is correlated for a range of global subset sizes to reflect the best results achievable for a traditional trial-and-error approach to subset size selection. ANNDSS and the results of the global analysis of each case are directly comparable, since both employ the same query points and M'_{max} is set to be equivalent to the largest global subset size analysed as documented in Table 6.2. The largest global subset size analysed was that at which the DESD had settled to a reasonable degree. Cases 1-5 and 6 are analysed using a step size of 5 and 15 pixels, respectively. The ROI was set to the whole image for Cases 1-5 while for Case 6 it was set to the perimeter of the specimen.

6.5.1 ANN validation

The DESD predicted by the ANN for the validation dataset are shown in Figure 6.6(a). The mean and standard deviation of the associated prediction errors, computed according to Equation (6.29), quantify the trained ANN's accuracy and precision, respectively, as documented in Table 6.3 along with those of Equations (6.18-6.19).

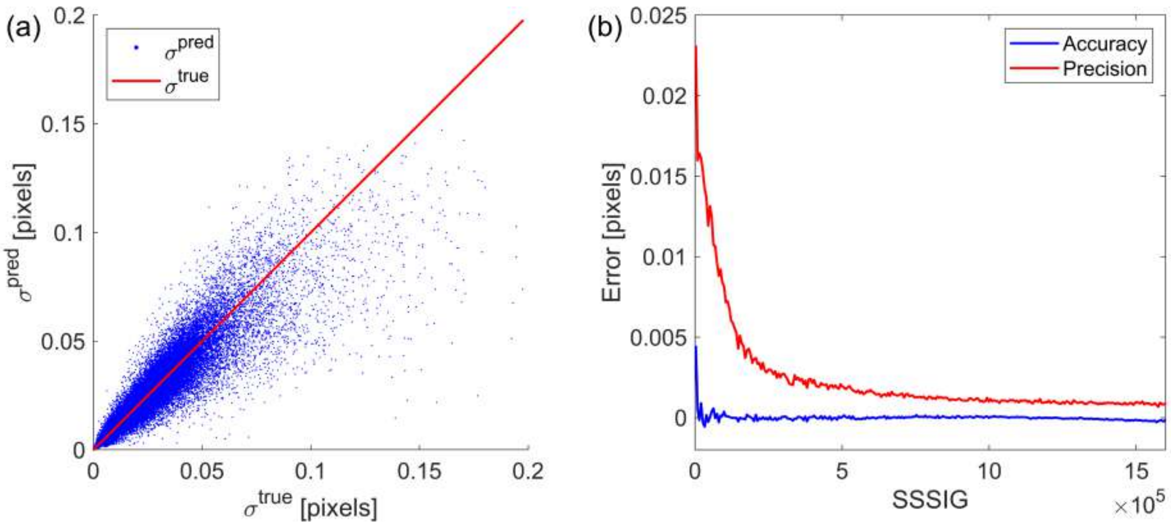


Figure 6.6: (a) The correspondence between σ^{true} (red line) and σ^{pred} (blue dots) of the validation dataset. (b) Moving mean and standard deviation of $\Delta\sigma_d$ as a function of SSSIG.

Although the ANN is an order of magnitude more accurate and precise than the theoretically derived Equations (6.18-6.19), its precision is dependent on SSSIG as illustrated in Figure 6.6(b). This is due to the combination of this being the region of the SSSIG vs. DESD relation which is most volatile and the lack of valid datapoints in this region available to train the ANN as observed in Figure 6.4(b).

Table 6.3: Accuracy and precision for DESD prediction of ANNDSS and Equations (6.18-6.19), of Pan et al. [60], for the validation dataset [10^{-2} pixels].

Method	Accuracy	Precision
ANNDSS	0.135	0.362
Equations (6.18-6.19)	1.21	1.45

6.5.2 Synthetic validation

The σ^{thresh} dictates the compromise between noise suppression and SR as discussed in Section 6.4.5. Furthermore, more lenient σ^{thresh} correspond to smaller SSSIG values while the ANN's precision was shown in Section 6.5.1 to decrease with the SSSIG value. As such, two σ^{thresh} values, corresponding to the beginning and end of the transition regime of the SSSIG vs. predicted DESD curve, are employed for Cases 1-6 to investigate the influence of the stipulated σ^{thresh} on the performance of ANNDSS.

Table 6.4 documents, for each analysis, the σ^{thresh}, the corresponding σ_x and σ_y for the subset sizes appointed by ANNDSS and the percentage discrepancy between σ^{thresh} and σ_x or σ_y treating the latter as the true values. Furthermore, it documents the σ_x and σ_y for a global analysis of subset size equivalent to the mean, rounded to the nearest odd integer, appointed by ANNDSS as well as the percentage improvement in σ_x and σ_y offered by ANNDSS relative to this global analysis.

Table 6.4: The DESD results of ANNDSS (with percentage discrepancy relative to the σ^{thresh} stipulated treating σ_x and σ_y as the true values) and a global analysis for subset sizes equivalent to the mean appointed by ANNDSS (with percentage improvement in DESD of ANNDSS relative to the global analysis) for Cases 1-5 [10^{-2} pixels].

Case	σ^{thresh}	σ_x				σ_y			
		ANNDSS		Mean		ANNDSS		Mean	
		Pixels	%[21]	Pixels	%[22]	Pixels	%[21]	Pixels	%[22]
1	1.8	1.57	-14.6	7.92	80.2	1.67	-7.78	16.1	89.6
	0.9	0.847	-6.26	1.22	30.6	0.918	1.96	1.64	44
2	2.1	1.89	-11.1	2.05	7.8	1.58	-32.9	1.92	17.7
	1	2.52	60.3	1.73	-45.7	0.919	-8.81	1.01	9.01
3	8	6.02	-32.9	5.96	-1.01	5.68	-40.8	5.69	0.176
	4	3.29	-21.6	3.25	-1.23	2.95	-35.6	2.93	-0.683
4	24	1.55	-1448	1.55	0	1.7	-1312	1.72	1.16
	19.5	0.902	-2062	0.891	-1.23	0.981	-1888	0.974	-0.719
5	0.8	0.445	-79.8	0.428	-3.97	0.481	-66.3	0.455	-5.71
	0.4	0.237	-68.8	0.232	-2.16	0.281	-42.3	0.282	0.35

Although σ_x and σ_y are computed over the set of displacements errors associated with the subsets analysed, in the absence of SR effects (which is valid for all cases, except Case 2 in the x-direction), this approximates the random error due to image noise. Thus, the percentage discrepancy between the σ_x and σ_y returned by the ANNDSS relative to the σ^{thresh} stipulated, indicates the ANNDSS's (and by extension the ANN's) accuracy.

ANNDSS is most accurate for Cases 1-2 having at most 14.6% and 32.9% discrepancy, respectively. The higher discrepancy of a real speckle pattern of low noise

---

[21] Negative percentage discrepancy indicates that the σ^{thresh} stipulated has been satisfied.

[22] Positive values indicate improved DESD of ANNDSS relative to the global analysis.

(Case 5 of at most 79.8%) relative to a synthetic speckle pattern of high noise (Case 3 within 40.8%) indicates that ANNDSS's, and thus the ANN's, accuracy deteriorates for real speckle patterns. The sharper speckle boundary of real speckle patterns relative to those generated by Section 6.3.1 causes the ANN to underestimate their noise suppression capacity.

Case 4's discrepancy of up to 20 times (2062%) is due to the combination of a real speckle pattern, high image noise (higher than is typical for machine vision cameras [122]) and low contrast (outside the range the ANN was trained on). Despite this, ANNDSS still appoints appropriate subset sizes for Case 4 as discussed at the end of this section. Thus, ANNDSS's accuracy is dependent on whether the speckle pattern characteristics fall within the training scope.

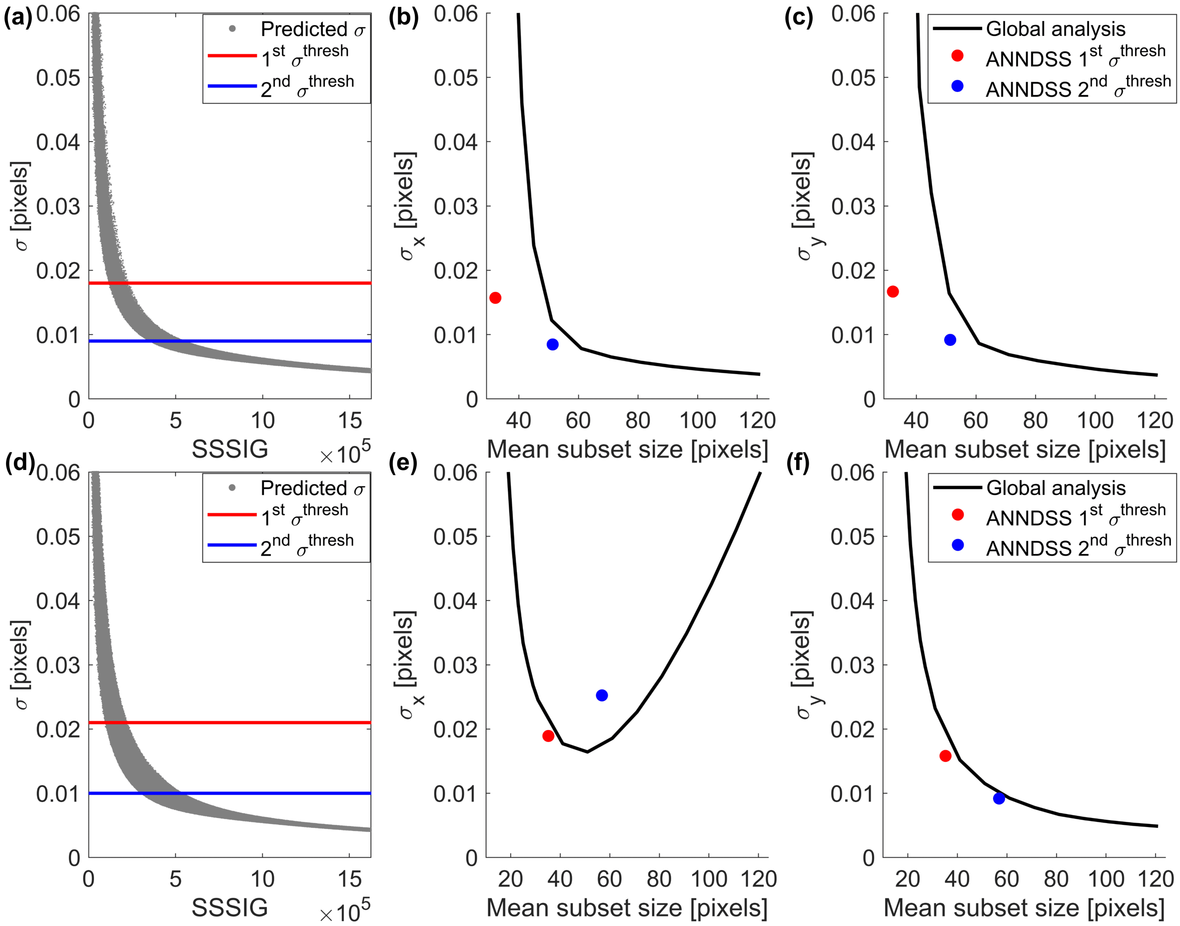


Figure 6.7: The two σ^{thresh} plotted relative to the DESD predicted by the ANN for Case 1 (a) and Case 2 (d). The σ_x of ANNDSS and the global analysis for Case 1 (b) and Case 2(e). The σ_y of ANNDSS and the global analysis for Case 1 (c) and Case 2 (f).

The DESD results of Cases 1-2, plotted in Figure 6.7 along with a plot of the SSSIG vs. predicted DESD, illustrate the potential of ANNDSS. In particular, the percentage coverage of the speckle pattern of Case 1, illustrated in Figure 6.8(a), increases along the x-direction, while the speckle radius decreases along the y-direction. ANNDSS compensates for this by appointing decreasing subset sizes along these directions, as shown in Figure 6.8(b), so that it returns improved DESD results (by at least 30.6%) relative to the global approach at the same mean subset size. Furthermore, we note that for a global subset size of 61 pixels, σ_y is decidedly larger than σ_x indicating that the noise suppression capacity of the speckle pattern is anisotropic. However, the ANNDSS's directive to appoint the subset size based on the direction in which noise suppression is predicted to be poorest results in greater consistency between σ_x and σ_y.

For Case 2, shown in Figure 6.8(d), the global analysis, where σ_x increases for subsets smaller and larger than 51 pixels due to noise suppression and SR, respectively, emphasizes the necessity of appropriate subset size selection to realise accurate and reliable displacements. The ability of ANNDSS to assign subset sizes that offer a favourable compromise between noise suppression and SR is illustrated by the DESD results of the first σ^{thresh} which, in addition to offering improved DESD relative to a global analysis for the same mean subset size, are on par with the best results achievable in the x-direction for a global analysis. The results of the second σ^{thresh} confirm the anticipated tendency of ANNDSS to favour noise suppression over SR.

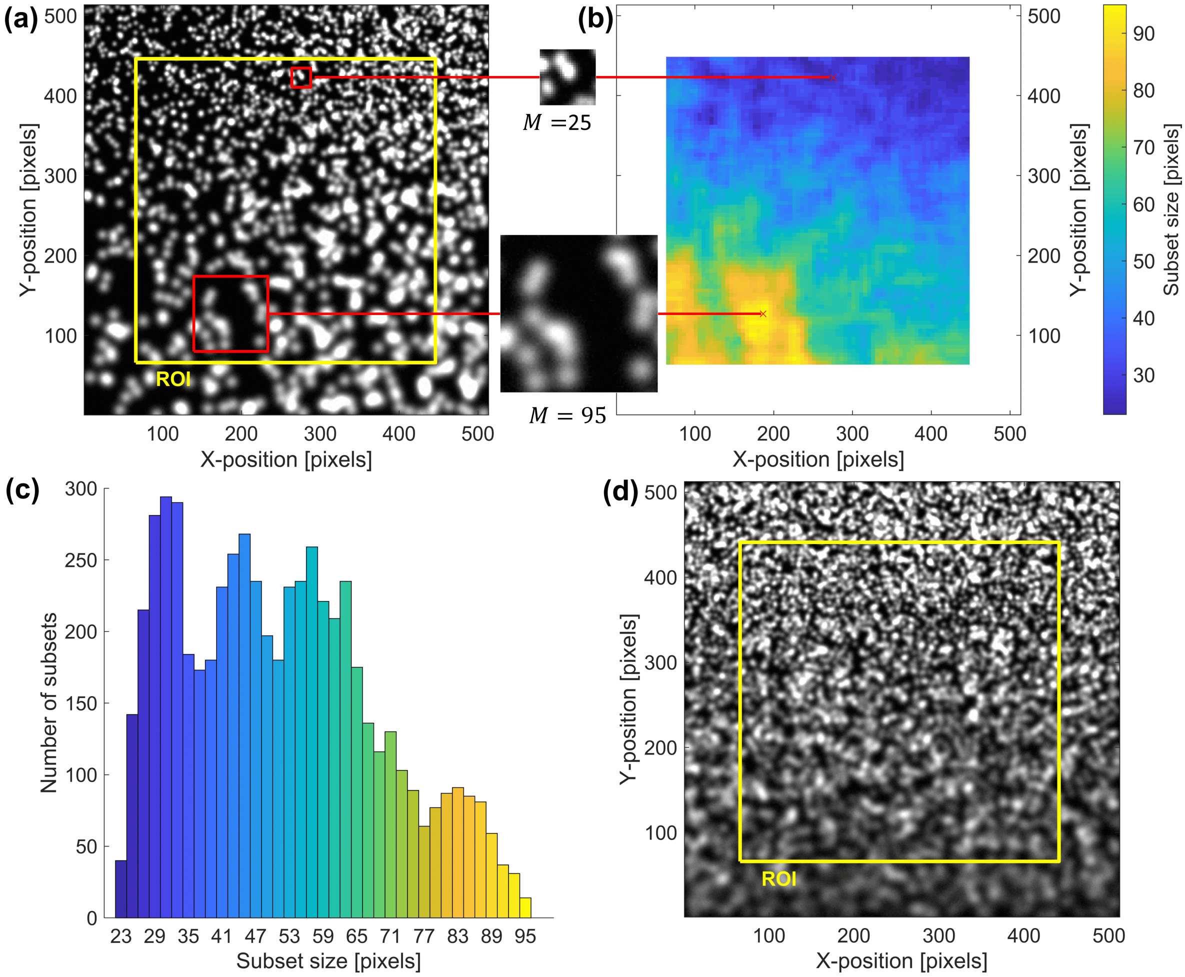


Figure 6.8: Synthetic image sets of varying speckle pattern quality for Case 1 (a) and Case 2 (d). For Case 1 the subset sizes appointed by ANNDSS for a σ^{thresh}=0.009 are shown as a heatmap in (b) and a histogram in (c).

The DESD results of Cases 3-5 are essentially identical in the x- and y-directions. As such, the σ_x results are shown in Figure 6.9 while those of σ_y are presented in Figures B.1-B.3 of Appendix B, along with the SSSIG vs. predicted DESD, for Cases 3-5.

Figure 6.9, as well as Table 6.4, show that the DESD results of ANNDSS for Cases 3-5 are on par with that of the global analysis (within 5.7%) for the same mean subset size. The spatially consistent speckle pattern density and quality of Cases 3 and 5 limit the benefit of ANNDSS. Although ANNDSS logically assigns the subset sizes for Case 4 according to the varying speckle pattern density, as illustrated in Figure 6.10, the noise suppression is not improved to the degree anticipated by the ANN due to its low accuracy for this image set. However, despite ANNDSS's low accuracy for Case 4 (discrepancy of up to 2062%) and consequential inability to offer improved DESD relative to the global analysis for the same mean subset size, ANNDSS remains capable of appointing subset sizes for the first and second σ^{thresh} consistent with beginning and end of the transition regime of the subset size vs. global DESD curve as desired.

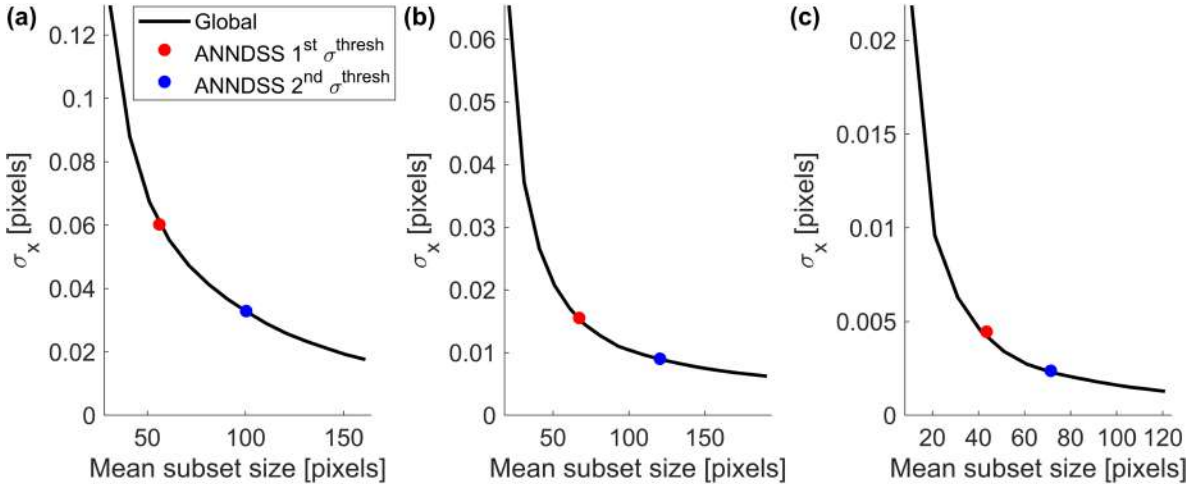


Figure 6.9: The σ_x of ANNDSS and the global analysis for Case 3 (a), Case 4 (b) and Case 5 (c).

More generally, Figures 6.7 and 6.9 show that for the first and second σ^{thresh} ANNDSS returns DESD results on par with those of the global analysis consistent with the beginning and end of the transition regime of the subset size vs. global DESD curve; albeit at a smaller average subset size for Cases 1-2 having speckle pattern characteristics within the training scope. This illustrates the ability of ANNDSS to appoint subset sizes offering noise suppression consistent with the region of the trend in DESD at which σ^{thresh} is stipulated while limiting the potential cost to SR regardless of whether speckle pattern characteristics fall within the training scope. Furthermore, these results show that the dependency of the ANN's precision on SSSIG does not hinder the capability of ANNDSS to assign subset sizes offering noise suppression consistent with the region of the SSSIG vs. DESD curve at which the σ^{thresh} was stipulated; provided that the σ^{thresh} is stipulated within the transition regime.

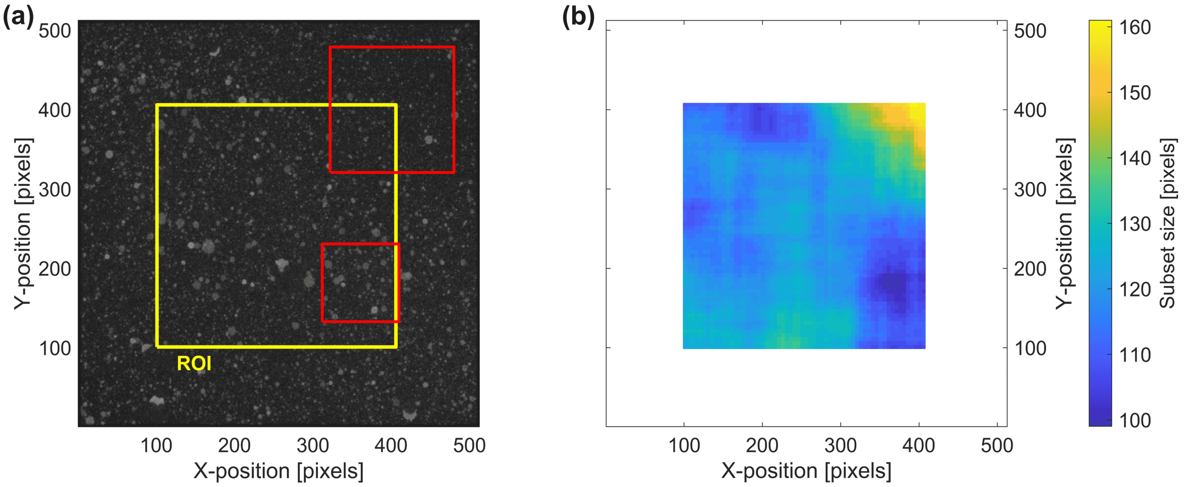


Figure 6.10: (a) Reference image of Case 4 with the largest and smallest subset sizes outlined in red. (b) Distribution of subset sizes appointed by ANNDSS. The contrast of (a) is tripled such that the speckle pattern is visible.

6.5.3 Experimental validation

SR effects are mitigated by correlating only the first four deformed images such that the first-order SF fully accounts for their deformation. The first and second σ^{thresh} are set as 0.003 and 0.012, respectively, as shown in Figure 6.11(a). As the true underlying displacement field is unknown, the correlation results are reported as a DESD relative to the displacements computed for a subset size of 241 pixels, as shown in Figures 6.11(b-c). This gives a good indication of the DESD trend since the smallest SSSIG for a subset size of 241 pixels is 2.96×10^6 which is well above the value at which the DESD is observed, in Figure 6.4(b), to settle for ψ=8 (whereas for Case 6, ψ=1.67 according to [296]).

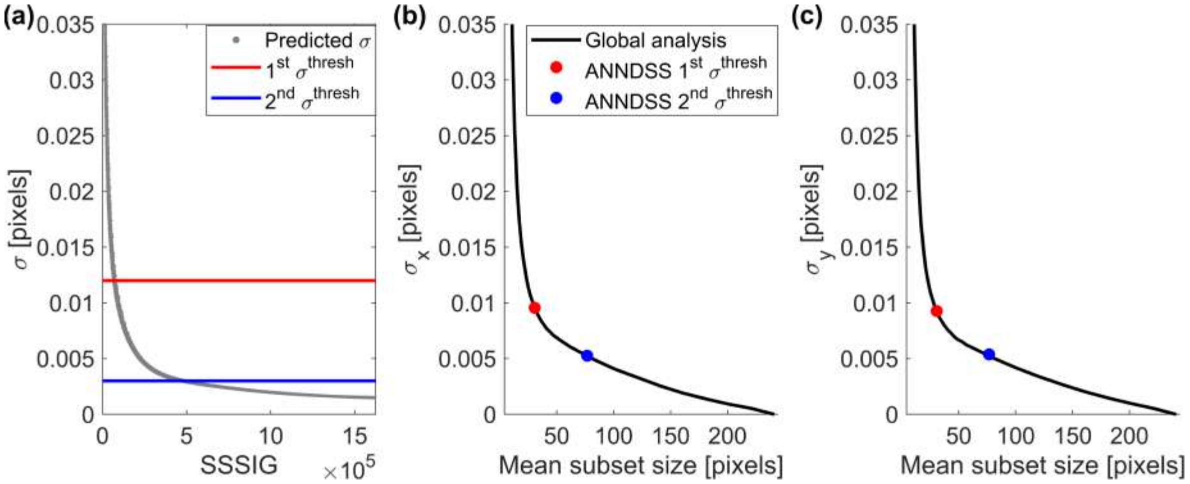


Figure 6.11: The results of Case 6 illustrated as: (a) the two σ^{thresh} plotted relative to the DESD predicted by the ANN; (b) the relative σ_x for the global analysis and ANNDSS; and (c) the relative σ_y for the global analysis and ANNDSS.

The relative DESD of ANNDSS fall on par with those of the global analysis because of constant speckle pattern density. In particular, for the first and second σ^{thresh}, ANNDSS returns relative DESD results at a point of the subset size vs. relative DESD curve of the global analysis, which correspond to the beginning and end of the transition regime. This illustrates the ability of ANNDSS to appoint subset sizes that offer sufficient noise suppression while limiting the potential cost to SR for experimental image sets, for σ^{thresh} within the transition regime of the SSSIG vs. DESD curve.

6.6 Discussion

Speckle pattern characteristics within the training scope indicate the potential of ANNs as an approach to the prediction of displacement errors and DSS purely based on image information. First, the incorporation of multiple SPQMs and the accounting for the complete error chain of the correlation process enables the ANN to predict the DESD of a speckle pattern to an order of magnitude more accurately and precisely than the widely accepted, theoretically derived equations of Pan et al. [60]. Furthermore, this approach of DESD prediction via ANNs can be extended to any DIC algorithm and shape function without requiring simplification of the numerical complexity; while transfer learning [297] is attractive as a method to retrain the existing ANN for alternative DIC implementations.

Secondly, the high accuracy of the ANN enables ANNDSS to dynamically assign the subset sizes of each query point, based on the local speckle pattern quality, to offer noise suppression capacity consistent with the σ^{thresh} stipulated (Cases 1-2). This is advantageous since: (i) for speckle patterns of varying density ANNDSS returns improved noise suppression relative to a global analysis for the same mean subset size; (ii) knowing the contribution of image noise to random errors enables the reliability of the computed displacement field to be inferred despite the true displacements being unknown, giving confidence to results; and (iii) through stipulation of σ^{thresh} the user dictates this contribution of image noise to random errors.

Cases 3-5 motivate expanding the scope of speckle pattern characteristics upon which the ANN is trained to improve the accuracy to which the subset sizes assigned by ANNDSS match the σ^{thresh} stipulated. Despite this, ANNDSS is shown to successfully assign appropriate subset sizes for speckle pattern characteristics outside the training scope; even in the presence of low contrast and high noise content (Cases 3 and 4).

Moreover, Cases 1-6 show that regardless of whether the speckle pattern characteristics are within the training scope, the precision of the ANN is sufficient to reliably predict the trend in noise suppression. Subsequently, for the first and second σ^{thresh}, ANNDSS returns DESD results on par with those of the global analysis which are consistent with the beginning and end of the transition regime of the subset size vs. global DESD curve. This illustrates the ability of ANNDSS to assign subset sizes that return DESD results offering noise suppression consistent with the region within the transition regime of the SSSIG vs. DESD curve at which the σ^{thresh} was stipulated. Furthermore, Case 2 confirms, as hypothesised, that the smallest subset size offering sufficient noise suppression, consistent with the transition regime, offers a favourable compromise between noise suppression and SR for up to moderate displacement gradients, although with a tendency to favour the former. Therefore, regardless of whether the characteristics of the speckle pattern fall within the limited training scope, ANNDSS is able to assign subset sizes that offer a favourable compromise between noise suppression and SR provided the σ^{thresh} selected is appropriate.

Thus, although ANNDSS, in its current implementation, cannot be recommended for practical deployment as a DSS method since its success in this regard is limited to speckle pattern characteristics falling within the training scope, it is attractive for practical use as a tool to indicate what range of global subset sizes are appropriate for an analysis such that they offer a favourable compromise between noise suppression and SR for up to moderate displacement gradients.

The ability of ANNDSS to offer a favourable compromise between noise suppression and SR is expected to break down for complex displacement fields. The use of ANNDSS as a precursor to the rotated Gaussian weighted digital image correlation (RGW-DIC) algorithm [289] is an attractive solution to this. More specifically, the weakness of ANNDSS to favour noise suppression is negated by the ability of the RGW-DIC algorithm to effectively reduce the appointed subset size to account for severe heterogeneous displacement gradients, while the weakness of the RGW-DIC algorithm in that it is incapable of increasing the subset size to improve noise suppression is negated by utilising a strict σ^{thresh} for ANNDSS. Such an approach would offer a favourable compromise between noise suppression and SR for severe displacement gradients which are independent of the user provided the σ^{thresh} stipulated is sufficiently strict.

Although ANNDSS was developed for 2D DIC, in its current form it should be straightforward to extend to stereo DIC by simply operating separately on the image sets associated with each camera.

6.7 Conclusion

This work investigates the potential of utilising an ANN as an approach to displacement error prediction and DSS purely based on image information. In particular, image sets containing speckle pattern characteristics consistent with that the ANN was trained on reveal the potential of the proposed ANNDSS method in that:

i The ANN accurately and precisely predicts the DESD of a subset from its SPQMs and standard deviation of image noise;

ii Subsequently the ANN accurately and reliably predicts the trend in noise suppression for an analysis which along with knowledge of the severity of the displacement field complexity, generally known *a priori* in experimental solid mechanics applications, affords all the necessary information to stipulate a DESD threshold appropriate for the analysis;

iii ANNDSS dynamically appoints the smallest subset size for each query point, based on the local speckle pattern, that offers noise suppression satisfying this DESD threshold;

iv The method offers improved noise suppression relative to the best results for the traditional trial-and-error approach to subset size selection for the same mean subset size;

v Knowledge of the contribution of image noise to random error enables the quality of the computed displacement field to be inferred; and

vi Provided the DESD threshold is appropriate, ANNDSS appoints subset sizes offering a favourable compromise between noise suppression and SR for up to moderate displacement gradients.

Although the generalisability of the ANN and thus potential of ANNDSS is limited for speckle pattern characteristics outside the scope that the ANN was trained on, ANNDSS remains capable of offering a favourable compromise between noise suppression and SR for up to moderate displacement gradients. Furthermore, ANNDSS can be readily implemented with any 2D DIC algorithm capable of assigning the subset size of each query point independently while extension to stereo DIC should be straightforward.

Thus, this novel approach to DSS purely based on image information through utilising an ANN as an error prediction tool is an attractive alternative to the traditional trial-and-error approach of subset size selection.

6.8 Post-submission discussion

This section presents a discussion of the presented work based on discoveries made, literature published post-submission and insights gained during the peer review process.

It should be noted that during analysis of the accuracy and precision to which the Equations (6.18-6.19) of Pan et al. [60] predict the DESD of the validation dataset, the SSSIG values were incorrectly scaled. It was discovered that alternative version of these equations have been proposed in literature. In particular, Equations (6.18-6.19) correspond with the work of Pan et al. [60] while Equations (6.30-6.31), presented below, correspond with the work of Wang and Pan [61].

$$\sigma_u \approx \frac{\sqrt{2}\psi}{\sqrt{SSSIG_x}} \tag{6.30}$$

$$\sigma_v \approx \frac{\sqrt{2}\psi}{\sqrt{SSSIG_y}} \tag{6.31}$$

As such, the prediction accuracy and precision of both versions of these equations were analysed according to the approach of Section 6.5.1 using correctly scaled SSSIG values. The results are presented in Table 6.5 along with the original results of the ANN ($\text{ANN}^{\text{Original}}$). It can be seen that Equations (6.30-6.31) have improved accuracy and precision relative to Equations (6.18-6.19). Despite this, $\text{ANN}^{\text{Original}}$ offers higher accuracy and precision, outperforming both versions of these equations. As such, an analysis was conducted to determine to what extent incorporating additional SPQMs contributed to this.

More specifically, additional entries in Table 6.5 present prediction accuracy and precision results for ANNs which use different combinations of SPQMs as inputs to reflect the impact of these SPQMs on the ANNs ability to predict DESD. In addition to standard deviation of image noise: (i) $\text{ANN}^{\text{Retrain}}$ takes as input the same SPQMs as $\text{ANN}^{\text{Original}}$ to illustrate the training stability of the proposed ANN; (ii) $\text{ANN}^{\text{SSSIG}}$

Table 6.5: Accuracy and precision for DESD prediction of theoretical relations and ANNs using various SPQMs as inputs.

Method	Accuracy	Precision
Equations (6.18-6.19)	0.6483	0.8539
Equations (6.30-6.31)	0.4443	0.7201
$\text{ANN}^{\text{Original}}$	0.1354	0.3362
$\text{ANN}^{\text{Retrain}}$	0.1366	0.3362
$\text{ANN}^{\text{SSSIG}}$	0.3006	0.4423
$\text{ANN}^{\text{NoSSSIG}}$	0.1915	0.3879

uses SSSIG as the only SPQM; (iii) $\text{ANN}^{\text{NoSSSIG}}$ employs all SPQMs with the exception of SSSIG, that is all SPQMs which have only been shown through empirical observation to be related to the displacement error; and (iv) ANN^{IVR} using IVR as the only SPQM. More specifically, additional entries in Table 6.5 present prediction accuracy and precision results for ANNs which use different combinations of SPQMs as inputs to reflect the impact of these SPQMs on the ANNs ability to predict DESD. In addition to standard deviation of image noise: (i) $\text{ANN}^{\text{Retrain}}$ takes as input the same SPQMs as $\text{ANN}^{\text{Original}}$ to illustrate the training stability of the proposed ANN; (ii) $\text{ANN}^{\text{SSSIG}}$ uses SSSIG as the only SPQM; and (iii) $\text{ANN}^{\text{NoSSSIG}}$ employs all SPQMs with the exception of SSSIG, that is all SPQMs which have only been shown through empirical observation to be related to the displacement error. The development of these ANNs is identical to that of $\text{ANN}^{\text{Original}}$ using the same architecture, training approach as well as data points within the training, testing and validation datasets as detailed in Section 6.4.4. Several observations can be made from these results.

Firstly, $\text{ANN}^{\text{Retrain}}$ performs on par with $\text{ANN}^{\text{Original}}$ indicating that good training stability is achieved through the approach taken to train the ANN.

Secondly, regardless of which combination of SPQMs are used, each results in an ANN which outperforms the theoretically derived equations of Wang and Pan [61] as a result of the ANN accounting for the full error chain of the correlation process within the *ultimate error regime* [122]. These theoretically derived equations rely on the assumption that both the contribution of image noise to the intensity gradient and the error introduced by the interpolation method are negligible. These assumptions are violated by the datasets since a large portion of image pairs contain significant noise. Furthermore, image correlation employs the bi-cubic b-spline interpolation method without image pre-filtering which is prone to appreciable interpolation errors [12]. As a result, $\text{ANN}^{\text{SSSIG}}$ predicts the DESD with improved accuracy and precision (percentage improvement of 32.3 % and 38.6 %, respectively) relative to Equations (6.30-6.31).

Thirdly, the inclusion of SPQMs which have not been theoretically related to the displacement error greatly improve the accuracy and precision of the ANN. Moreover, $\text{ANN}^{\text{NoSSSIG}}$ models the relationship between SPQMs, which have only through empirical observation been related to the displacement error, and DESD with greater precision and accuracy than the theoretically derived equations.

Lastly, as hypothesised, a single SPQM is insufficient to capture all the characteristics that make a speckle pattern favourable for DIC. This is further supported by the discovery of PIB [118] which shows that the speckle pattern impacts systematic errors in addition to random errors.

Feedback received from the peer review process of the article suggested that the proposed ANNDSS method should be compared to the performance of the DSS methods associated with the SPQMs used as inputs to the ANN. To this end, Cases 1-5 were correlated for subset sizes appointed by the SPQM based DSS methods of SSSIG, subset intensity variation, subset Shannon entropy and intensity variation ratio. For SSSIG, subset sizes were assigned as the smallest which offers a $SSSIG_x$ and $SSSIG_y$ value of at least 10^5 as recommended [60]. The subset Shannon entropy based DSS method [59] adjusts the subset sizes of each query point until SSE^{sub} is within $(SSE^{im}-0.1) \leq SSE^{sub} \leq (SSE^{im}+0.1)$.

The DSS methods of subset intensity variation and intensity variation ratio assign subset sizes such that they contain approximately 4 speckles; motivated by empirical observations and the recommendation of Hassan et al. [298]. For the intensity variation ratio this translates to appointing the smallest subset which offers a Λ value of 1.5 [21].

The threshold subset intensity variation value is designed to adjust to the speckle pattern within the ROI. More specifically, the threshold image is summed to determine the area occupied by speckles within the ROI as $A^{speckles}$. Canny edge detection [299] [23] is performed on the threshold image to determine speckle edges; the sum of which returns the total edge length of the speckles within the ROI as $L^{speckles}$. The number of speckles within the ROI, S^{η}, is computed as

$$S^{\eta} = \frac{{L^{speckles}}^2}{4\pi A^{speckles}} \tag{6.32}$$

The threshold subset intensity variation value, η^{thresh}, is computed as

$$\eta^{thresh} = \frac{s^{\eta}}{S^{\eta}} \eta^{ROI} \tag{6.33}$$

where $s^{\eta} = 4$ as the desired number of speckles within a subset and η^{ROI} is the subset intensity variation over the ROI. For each query point the subset is assigned as the smallest which satisfies η^{thresh}.

Cases 1-5 were correlated according to the subset sizes assigned by each SPQM based DSS method using the same query points and maximum subset size as ANNDSS such that the results are directly comparable. The DESD, computed according to Equations (6.14-6.15), and error range (represented as Δu^{range} and Δv^{range} in the x- and y-direction, respectively) were computed across all subsets of each correlation analysis and are reported in Table 6.6 along with those of ANNDSS (consistent with those reported in Table 6.4, repeated here for convenience).

The relatively higher DESD results of SSSIG for Case 1 are due to the method appointing small subsets, which contain a single speckle, for a handful of query points. The resulting spurious displacement measurements for these query points increase the DESD results.

---

[23] Note that Zhang et al. [20] do not recommend an edge detection algorithm. Canny edge detection [299] was chosen since using two thresholds to detect weak and strong edges makes it more robust to noise present in the image.

Table 6.6: The resulting DESD and error ranges (represented as Δu^{range} and Δv^{range} in the x- and y-direction, respectively) of the SPQM based DSS methods and ANNDSS [10^{-2} pixels].

Case		1	2	3	4	5
SSSIG	σ_x	5.38	1.99	21.4	0.811	0.239
	Δu^{range}	724	19.4	1495	6.44	1.63
	σ_y	5.34	1.86	10.3	0.897	0.278
	Δv^{range}	622	19.5	719	6.12	1.99
Intensity variation ratio	σ_x	0.746	2.22	-	4.22	0.589
	Δu^{range}	7.06	34.7	-	49.4	11.4
	σ_y	0.798	1.01	-	4.55	0.67
	Δv^{range}	6.96	26.9	-	53.5	13
Subset Shannon entropy	σ_x	1.72	3.63	-	-	0.131
	Δu^{range}	62.8	30.4	-	-	0.932
	σ_y	1.93	0.88	-	-	0.146
	Δv^{range}	117	15.6	-	-	1.08
Subset intensity variation	σ_x	0.999	1.55	-	6.7	0.269
	Δu^{range}	13.2	19.3	-	95.8	1.87
	σ_y	1.12	1.32	-	7.08	0.313
	Δv^{range}	22.4	17.3	-	132	2.14
ANNDSS 1^{st} σ^{thresh}	σ_x	1.57	1.89	6.02	1.55	0.445
	Δu^{range}	15.8	17.9	46.3	11.6	4.26
	σ_y	1.67	1.58	5.68	1.7	0.481
	Δv^{range}	17.8	16.2	47.1	14.3	4.29
ANNDSS 2^{nd} σ^{thresh}	σ_x	0.847	2.52	3.29	0.902	0.237
	Δu^{range}	7.09	30.1	26	7.64	1.79
	σ_y	0.918	0.919	2.95	0.481	0.281
	Δv^{range}	7.15	8.52	22.6	6.86	2.2

Undersized subsets, containing insufficient speckle pattern information, are assigned for intensity variation ratio (Case 3), subset Shannon entropy (Case 3 and 4) and subset intensity variation (Case 3) such that correlation becomes unstable and fails. It is reasoned that the low contrast of these speckles causes the high image noise to contribute significantly to the computed SSE^{sub} values, such that the DSS method interprets image noise as speckle information, leading to the assignment of undersized subsets. For intensity variation ratio and subset intensity variation it is reasoned that the low contrast and high noise content reduces the accuracy to which Otsu's method [288] can identify speckle edges decreasing the reliability of the intensity variation ratio and subset intensity variation DSS methods for these images. This is argued to be the reason that the DESD results of intensity variation ratio and subset intensity variation are the highest across the DSS methods for Case 4 (at least 148% higher).

In contrast, the inclusion of multiple SPQMs as inputs to the ANN enables reliable prediction of the DESD such that ANNDSS avoids appointing undersized subsets thereby preventing unstable correlation. Furthermore, the SSSIG vs predicted DESD trend facilitates informed selection of an appropriate σ^{thresh} for the image set under consideration. Consequently, ANNDSS performs consistently across Cases 1-5 offering DESD results in line with the best achieved by existing SPQM based DSS methods.

Upon submission of the article it was believed to be the first attempt to use ANNs to relate the quality of a speckle pattern to the resulting error in the DIC computed displacements; however, as recently as October 2022 Kwon et al. [97] proposed a method using a CNN to predict mean error from speckle patterns. This method differs greatly from the approach taken here in that: (i) through the use of a CNN it operates on speckle patterns directly with the intention of avoiding the need for SPQMs; (ii) it focuses on the global speckle pattern quality and as such intends to compete with global SPQMs such as mean intensity gradient [280], subset Shannon entropy (SSE) [59], mean subset fluctuation [279] and standard deviation of grey intensities within each speckle [300]; and (iii) it computes displacement errors across an image set, instead of the approach taken in this work (discussed in Section 6.3.2), such that SFAB is present within the errors which cannot be accounted for since only the speckle pattern is used as input. Although the method is designed to function for a range of input image sizes by compressing the input image to 50 by 50 pixels before processing them, this approach is ill-advised as a great amount of speckle pattern information is lost during this process and it leads to aliasing of speckles. As such, it performs poorly for smaller images of size 150 by 150 pixels, which is typically larger than an optimal subset size for experimental solid mechanics applications. Thus, despite this work showing the potential of CNNs for quantifying the quality of speckle patterns directly from images, it does not offer an approach for DSS.

Chapter 7
Discussion

The core chapters of this dissertation have been presented as self-contained sections; consisting of an introduction, literature review, methodology, validation, discussion, and conclusion. The purpose of this section is to expand on the discussions of these core chapters in relation to the objectives and motivations outlined in Chapter 3. In particular, this section expands on topics discussed previously by interpreting the findings in terms of their broader implications within the fields of DIC and experimental solid mechanics, as well as identifying limitations of the proposed frameworks in their current form. Thereafter, a preliminary investigation of the feasibility of ANNs for DSFS is presented motivating recommendations for future research directions.

7.1 Open-framework for image correlation

Chapter 4 documents the approach taken to develop a modular, open image correlation framework offering full control over setting up the correlation problem. Although the concept of offering control beyond that of stipulating a global subset size and SF order is not unique to this project, with DICe [167] offering control of the subset size and shape on a per-subset basis, the degree of control offered by the proposed framework exceeds that available in the current open-source DIC landscape, to the authors knowledge.

In particular, the proposed method allows for assignment of the subset location, size, shape, and SF order on a per-subset and per-image basis. Additionally, the framework allows custom SFs and subset sizes to be defined and assigned in this manner. Thus, not only does the framework allow full control over setting up the correlation problem from the standard options of zero, first or second-order SFs and square or circular subsets, but additionally specialised options, which can be stipulated as necessary. This completely removes the second barrier towards dynamic appointment of appropriate correlation parameters for local DIC identified in Section 1.3, enabling flexible adaption of the correlation setup to the requirements of any analysis.

For example, consider a specimen that exhibits a crack. Correlation fails for subsets which contain the crack path as the assumption of a continuous displacement field, required by the continuous SFs, is violated. Subset splitting [301] and segmentation [302] methods have been proposed to account for this, which break up the original subset so that the portion of the subset analysed contains a continuous displacement field enabling accurate displacement measurements in the region of the crack path. Such procedures are straightforward to implement within the proposed framework through the ability to define custom subset shapes.

The flexibility of the image correlation framework extends beyond setting up the correlation problem, with each main task being performed by a separate function, allowing these aspects to be readily interchanged for alternative methods of accomplishing these tasks. This is advantageous to individuals investigating novel approaches of performing various aspects of correlation, since the unproven method can be readily implemented within the framework. Validation is straightforward through comparing the resulting displacement errors of the modified algorithm to those of the original framework, that has been validated against established DIC algorithms as documented within the article [191] published in association with Chapter 4 (which is available through open access).

Furthermore, this article summarises the theory of the image correlation process, detailing its implementation as the proposed open framework, drawing direct links between the mathematical theory and coded implementation. Documentation of the framework coupled with the simple syntax of the code, which avoids obscuring the mathematical operations performed, makes the framework attractive as an educational resource; bridging the gap between the well-established theory of DIC and its nuanced practical implementation. This is of significance since historically this gap in literature has made it such that newcomers to the field of DIC needed to overcome a difficult learning curve to gain the working knowledge of DIC necessary to contribute to its development. Such an obstacle has arguably limited the potential rate of development of DIC.

Although the limitations of the framework are a consequence of the intentional compromises made during the design phase, it is worth discussing their implications. In addition to deviating from the current state-of-the-art practices of DIC as identified by Pan [22], by employing a simpler interpolation method and not utilising the RGDT strategy, the framework makes use of PCM [223] for initial displacement estimation.

PCM is limited in that it cannot track displacements greater than half the subset size in each direction, only accounts for RBT and only functions with square subsets. Generally these limitations are of little concern; however, in the presence of complex deformation or large displacements, PCM can break down leading to the failure of the correlation process which relies on these initial displacement estimates. As such, the robustness of the framework would greatly benefit from an improved method of initial displacement estimation.

The RGDT strategy [182] in conjunction with PCM is a viable solution to this. It uses subsets for which PCM was successful (returning high correlation coefficients) to seed the initial displacement estimates of their neighbours. Alternatively, in extreme cases the SIFT feature tracking algorithm offers an attractive solution that is robust to affine transformation, rotation, image scaling and variations in illumination. This is straightforward to implement as was shown in Section 5.5 to account for the jump in displacements between the sparse image sets of Samples 1, 2 and 5 of the Stereo DIC Challenge [123].

Although not investigated, it is probable that the proposed image correlation framework is prone to errors for displacement fields containing appreciable rotation [192]; a typical limitation of many DIC algorithms. Rotation is not uncommon for experimental solid mechanics applications with three-point bending flexural tests [303] used to determine the elastic modulus of the material in response to bending and flexural stress-strain. This limitation can readily be addressed through the modularity of the framework, enabling straightforward implementation of the SF proposed by Wu et al. [304] that accounts for rotation. Furthermore, as previously discussed, the SIFT algorithm can be used to ensure reliable initial displacement estimates in the presence of severe rotations.

As noted by de Deus Filho et al. [305], despite the image correlation algorithm being open-source, its implementation within a proprietary programming language limits its accessibility due to the cost of acquiring a MATLAB licence. This limitation was accepted as a compromise to be able to leverage MATLAB's simple syntax to avoid obscuring the mathematics implemented within the code. Furthermore, despite intending to improve the accessibility of DIC, the primary focus of the framework is to expedite efforts towards furthering the capabilities of DIC; which is predominantly undertaken within research institutions that typically have access to MATLAB licenses.

Thus, the majority of the limitations of the correlation framework are a consequence of the desire to keep the framework free of corrective measures such that these do not mask the limitations of the ANN-based framework developed. However, these limitations are straightforward to resolve due to the flexibility and modularity inherent to the framework. Furthermore, the framework: (i) bridges the gap in the open-source DIC landscape for an image correlation algorithm offering full control over setting up the correlation problem; (ii) offers modularity while remaining predominantly consistent with current state-of-the-art practices, reflected in it performing on par with established DIC algorithms; (iii) is attractive as a starting point to develop the capabilities of DIC removing the need, in many cases, for the traditional approach of developing an in-house DIC algorithm to validate novel contributions to the field; and (iv) serves as an educational resource bridging the gap between the theory of DIC and its implementation, as an intuitive code, alleviating the difficult learning curve faced by newcomers to the field of DIC.

7.2 Framework for 2D and stereo DIC

Chapters 4 and 5 extend the image correlation framework into a fully-functioning, open-framework for 2D and stereo DIC. This is done by documenting the theory of camera calibration (implemented using MATLAB's `estimateCameraParameters` function), displacement transformation (for 2D and stereo DIC) and stereo matching; the implementation of the latter two is detailed to draw direct links between the mathematical theory and code.

ADIC3D was shown to perform on the same level as the established DIC algorithms DICe and LaVision's StrainMaster software. Despite the 2D displacement transformation code not being validated directly, it is based on a well-established theory. Additionally, it uses MATLAB's camera calibration algorithm `estimateCameraParameters`, the performance of which is validated through the results of ADIC3D. Thus, this framework is suitable for practical use in the field of experimental solid mechanics. Additionally, their implementation within 117 and 202 lines of intuitive code, for ADIC2D and ADIC3D, respectively, combined with the modularity and flexibility of their image correlation implementation enables them to be readily adapted to a wide range of DIC applications outside the field of experimental solid mechanics.

This is advantageous because many novel applications of DIC are not directly addressed by currently available DIC algorithms. For example, remote sensing applications make use of DIC to measure the movement of the Earth's surface from satellite imagery [306] and light detection and range (LIDAR) data [188]. However, the displacement accuracy of these methods is limited as they rely on simplistic approaches to displacement transformation; which assume a constant metres-per-pixel ratio across the images. Additionally, they use crude corrective measures to account for distortions present within the image due to the lens system and atmospheric factors. This DIC framework offers an attractive starting point for investigating methods of improved displacement transformation to obtain more accurate displacement measurements for such applications.

Although not a concern for most DIC applications, the SIFT feature matching algorithm used to determine initial displacement estimates for stereo matching breaks down on the ridges and perimeters of protruding features, as shown in Figure 7.1. This is highlighted by how StrainMaster is capable of reliably correlating at least 56% more subsets than ADIC3D for Samples 1 and 2 of the Stereo DIC Challenge [123] (as reported in Section 5.5.1). As noted in Section 5.6, DeepFlow [265] is an attractive alternative to this. Moreover, with the recent introduction of openly available ANN-based DIC algorithms such as DisplacementNet [93], DeepFlow can be employed to track the larger displacements occurring between the different views of the specimen, while DisplacementNet can refine these displacements for more reliable initial estimates.

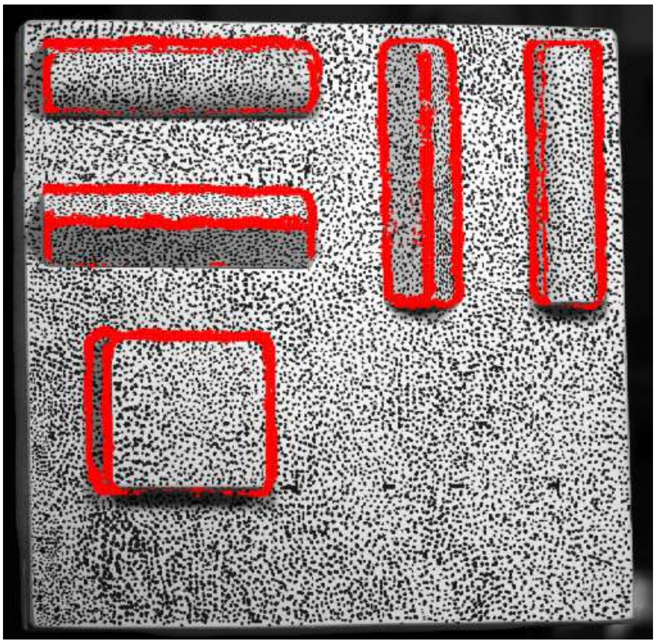


Figure 7.1: Illustration of the subsets which failed during stereo matching of Sample 2 of the Stereo DIC Challenge (indicated in red) due to poor initial estimates of SFPs [123].

The current stereo displacement transformation framework operates on the basis of two cameras viewing the specimen as is typical for stereo DIC. However, the need for multicamera stereo DIC has been identified for applications where the structure to be analysed is too large to be captured with sufficient SR by a single camera pair [307, 308]; as well as applications where the specimens surface is non-planar, such as cylindrical specimens [309], requiring several camera pairs to view the full surface. In such situations, multiple camera pairs are required to measure the full area of interest through image stitching to create a distributed field of view.

Furthermore, multicamera stereo DIC that captures the same area of the specimen with more than two cameras has been shown to obtain more accurate displacement measurements [310]. The availability of redundant data from additional cameras limits the influence of error sources, such as errors due to the stereo angle (such errors observable in Table 5.17).

The MutliDIC, an open-source toolbox for multi-camera DIC, developed in MATLAB by Solav et al. [251], offers an attractive solution to both these needs. Furthermore, MultiDIC offers bundle adjustment, a state-of-the-art requirement not met by the implemented displacement transformation algorithms. Utilising this toolbox as a wrapper for the image correlation framework requires straightforward conversion of the data it produces into the format expected by MultiDIC.

7.3 ANN for error prediction and DSS

Chapter 6 details the first reported use, to the author's knowledge, of ANNs for either DIC displacement random error prediction or DSS. Due to the relatively recent resurgence of ANNs in DIC, the novelty of the approach taken, and the limited understanding of the inner functionality of ANNs, this section is broken up into three main discussions. These review the broader implications of this work in terms of error prediction in DIC, generalisability concerns of ANNs, and the proposed ANN-based DSS method.

7.3.1 ANN as an error prediction tool

Assessing and quantifying the metrological performance of DIC has proven to be a challenging task due to the plethora of error sources within the DIC error chain and their complex interaction, as summarised in Section 2.1. The fact that PIB has only as recently as 2019 been identified as an error source despite decades of efforts towards metrological performance analysis of DIC puts the complexity of such an undertaking into perspective. At the same time, effective use of DIC requires a good understanding of the DIC error chain such that aspects under the control of the DIC practitioner can be conducted in such a way as to reduce the error sources introduced. This enables the most accurate and precise displacements to be realised.

This work indicates, in a broad sense, that ANNs have the potential to model aspects of the complex DIC error chain through their ability to detangle complex relationships from representative data. This is illustrated in Table 6.5 by how an ANN can relate SPQMs, other than SSSIG, to the DESD, despite this relation being at most observed through empirical observation. This is powerful, since empirical observations are typically the basis upon which the DIC error chain is investigated, and the influence of an error source is identified.

The approach of utilising ANNs to build tools for error prediction is particularly attractive with the availability of synthetic image generation methods, which can control the magnitude of various error sources present in the generated image set. Balcaen et al. [252] developed such a method which was used to empirically observe the influence of error sources introduced by image noise, lighting variations, stereo angle, aliasing of speckles, and motion blur [105]. Employing this method, a database of image sets containing such error sources, of varying magnitude, can be created. Performing DIC on these image sets would reflect their impact on the resulting displacement error. Treating the parameters controlling the severity of these error sources as inputs and the resulting displacement error as the output, an ANN could be trained to model the relationship between the two.

Utilising an ANN to model this relationship, which forms a portion of the DIC error chain, would be valuable for several reasons. Firstly, the ANN can be used as an initial step to detangle the impact of error sources and their combination on the resulting measured displacement field quality. Secondly, more in-depth knowledge of this enables the development of more informed guidelines and standards, such as the iDICs Good Practices Guide [8], to advise DIC practitioners on how to most effectively setup and conduct a DIC analysis to limit the error sources introduced such that higher quality displacement fields can be obtained. Lastly, the availability of such an error prediction tool would greatly aid research and investigation conducted through DIC by predicting the quality of the displacement field computed. This would give confidence in the results and conclusions drawn, or indicate the need for an improved experimental design (expediting the trial-and-error approach typically employed for the latter).

7.3.2 ANN generalisability

Before investigating ANNs as an approach to model the complex DIC error chain, the generalisability concerns typical of ANNs should be considered. Such generalisability concerns arise for the ANN of Chapter 6 where the performance diminishes for speckle pattern characteristics outside the training scope, as can be seen in Table 6.4. This is not as a result of over-fitting, with the loss value of the training and testing datasets being on par with one another, but is rather caused by design decisions made during ANN conception and database creation.

Firstly, the intensity variation ratio [21] is reasoned to add bias to the random errors predicted by the ANN. The intensity variation ratio relies upon Otsu's method [288] to classify pixels as belonging to either speckles or the background. The work of Alexander et al. [311] has shown that the ability of such thresholding methods to accurately determine the edges of speckles is subject to how appropriate the threshold value is. Additionally, lower contrast speckles, which do not cross the threshold value, will fail to be identified by such methods. Thus, the accuracy to which Otsu's method determines the speckle boundaries varies based on the speckle pattern characteristics thereby decreasing the generalisability of the ANN for speckle pattern characteristics outside the training scope.

Secondly, the synthetic speckle patterns generated by the method of Pan [287] have unique properties. It is well established that the systematic error of DIC computed displacements typically has a sinusoidal trend with a period of 1 pixel [17] due to interpolation errors [312]. However, Reu [225] identified that DIC performed on speckle patterns generated by Pan's method results in a systematic displacement errors which are largely independent of the sub-pixel displacement. This is illustrated in Figure 7.2 relative to experimentally acquired images which are shifted by the FFT method (consistent with Case 4 of Section 6.5.2). This is reasoned to be related to the notably smoother speckle boundaries produced by Pan's method relative to the speckle patterns of experimentally acquired images. Smoother speckle boundaries reduce the frequency content of the speckle pattern, which leads to smaller interpolation errors and more accurate displacement measurements as observed by Schreier et al. [12]. As such, speckle patterns produced by this method have unique speckle pattern characteristics that differ from those typical of DIC in experimental solid mechanics and are expected to affect the generalisability of the ANN.

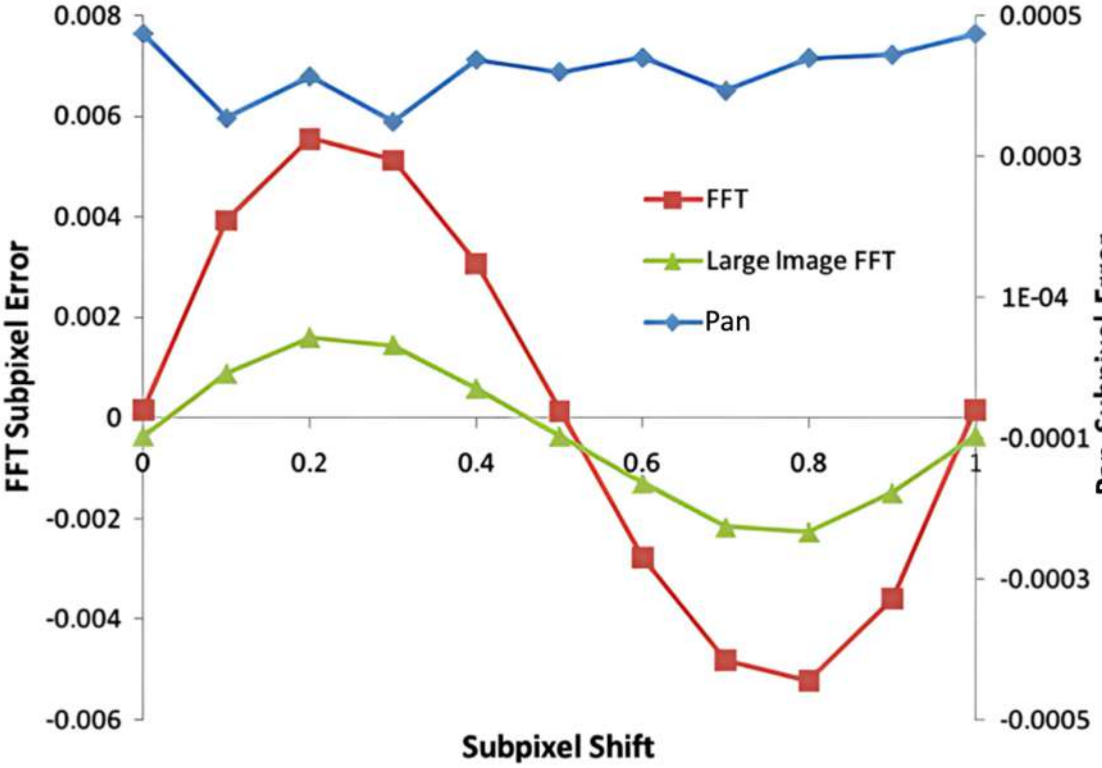


Figure 7.2: Illustration of the systematic errors of synthetic images generated by the method of Pan [287] and FFT [225].

Thirdly, although it is not observed to limit the functionality of ANNDSS, the data points are unevenly distributed within the training dataset, with regard to their spread over DESD values. Moreover, there are not many data points corresponding to higher DESD and lower SSSIG values since correlation for such subsets is unstable. ANNs are known to be sensitive to how evenly spread the training data is in terms of the input and output variables [313]. Otherwise, the loss function, which guides the training of the ANN, will favour reducing the prediction error of the data points with the most common output value since this will give the biggest reduction in the loss value. This is assumed to be the reason for the percentage discrepancy between σ^{thresh} and the actual DESD, in Section 6.5.2, being larger for the first threshold relative to the second threshold across Cases 1, 3 and 5; the first threshold being closer to the beginning of the settling regime having a lower SSSIG value. Cases 2 and 4 do not conform to this trend, due to the imposed displacements being higher than zero-order and the poor quality of the image sets, respectively.

Lastly, although not directly evaluated, the relationship that the ANN has modelled is assumed to be specific to the DIC algorithm that was used to generate the training dataset. It is well established that different DIC algorithms offer differing metrological performance due to the alternative methods that they employ within the correlation process [17]. In particular, the interpolation method greatly impacts the resulting displacement error [12]. It follows that the approach for developing an ANN for the purpose of random displacement error prediction will result in an ANN which caters towards predicting random error of the specific DIC algorithm on which it was trained.

Generalisability concerns as a result of the scope of the training dataset are not uncommon for applications of ANNs within the field of DIC. As highlighted in Section 1.4, DisplacementNet (of Deep DIC [93]) has poor performance for RBT while StrainNet [25] cannot handle displacements larger than a pixel due to these respective displacement categories being excluded from their training dataset. Training scope limitations are challenging to resolve due to the difficulty of obtaining, either through synehtetic image generation or experimental acquisition, the wide variety of displacements, lighting conditions, speckle patterns and noise that DIC algorithms are required to handle in practice.

Transfer learning offers an attractive solution to the generalisability concerns associated with the latter three mentioned aspects; that is, due to synthetic image generation method employed, uneven spread in datapoints and catering towards the DIC algorithm it is trained on. Transfer learning is the process of retraining the ANN to fine tune its weights and biases for the relations present within an alternative dataset. This is an attractive prospect as it enables improving the performance of an ANN for a specific use case by ensuring data relevant to this application is within the training database used for transfer learning. With regards to this specific ANN application, significantly fewer data points are presumed to be required for transfer learning since the ANN already has knowledge of the general trend in the random error; which remains predominantly consistent in form regardless of the speckle pattern characteristics or DIC algorithm. Thus, the generalisability of the ANN can be greatly improved while adapting it to a different DIC algorithm by: (i) using an alternative synthetic image generation method which generates images more representative of those encountered in experimental solid mechanics applications; (ii) employing the DIC algorithm of choice to determine the displacement errors of the dataset; and (iii) incorporating more data points associated with high DESD and low SSSIG values. In this way, the ANN can be adapted to any DIC algorithm without requiring simplification of the numerical complexity.

Furthermore, this offers an attractive means to investigate the potential of ANNDSS for naturally occurring speckle patterns that are reasoned to have differing speckle pattern characteristics. This would be advantageous for situations where application of sprayed speckles is infeasible due to the scale of the specimens, as in the case of DIC performed on microscope [314] and satellite imagery [306], or environmental conditions, such as long-term structural health monitoring of bridges [235] where painted speckle patterns are likely to fade reducing the quality of the computed displacements.

7.3.3 ANN for DSS

Despite the advantages of the ANNDSS method having been briefly mentioned, it is worth reiterating them in terms of their impact on the field of DIC, particularly in relation to currently available DSS methods. The DSS methods proposed, in association with the SPQMs employed by ANNDSS, assign subset sizes as the smallest which satisfies a threshold value for their respective SPQMs.

With the exception of SSSIG, these thresholds are obscurely related to the displacement error and are either: (i) stipulated by the user which is cumbersome and arguably the reason for the limited adoption of these approaches [19]; (ii) suggested based on empirical observation for a limited scope of testing cases, affording little confidence for use outside this scope; or (iii) based on the assumption that four speckles within a subset is ideal [20], which is shown by the work of Pan et al. [60] and Fayad [118] to be an oversimplification.

Consequently, the ANNDSS method is attractive in offering a streamlined alternative to existing DSS methods and rephrasing the obscure question of *what global subset size is appropriate for an analysis* to *what amount of random displacement error within the computed displacements is acceptable*. Furthermore, by presenting the trend in the predicted random error, the practitioner can make an informed decision as to what random error threshold is appropriate; enabling them to adjust this value to control the compromise between noise suppression and SR based on their *a priori* knowledge of the complexity of the underlying displacement field. This ability to infer and control the quality of the measured displacement field with regard to noise suppression, prior to the initiation of the DIC analysis, is invaluable for practical use of DIC.

Beyond generalisability concerns, the biggest limitation of the proposed approach going forward is that it is assumed to break down in the presence of complex deformations, as it will favour noise suppression over SR. This is undesirable since, for complex deformations, the first-order SF struggles to account for the underlying deformation such that its systematic error is significantly larger than that of the second-order SF as documented by Yu and Pan [315]. However, they also discovered that for an identical subset, the random errors of the second-order SF are at most double that of the first-order SF. This was later proven via theoretical derivation by Wang and Pan [61].

Consequently, it is highly recommended that the current approach be adopted to train an ANN as a random error prediction tool for the second-order SF. Employing this ANN in the ANNDSS method to appoint the smallest subset size that offers sufficient noise suppression for the second-order SF will offer a favourable compromise between noise suppression and SR for even complex deformation fields; relative to what is achievable for the standard options of SFs.

Additional recommendations include using the displacement transformation method to transform the metric of the predicted random displacement errors from pixels to millimeters (or inches, as desired) so that it is more intuitive. Furthermore, instead of appointing a square subset size, the random error in the x- and y-directions can be used to adjust the subset size in these directions independently. The resulting rectangular or oval subset would offer consistent noise suppression in both directions so that the SR of either direction is not restricted by the limited noise suppression of the opposing direction.

7.4 Towards DSFS

As noted by Pan [22] the difficulty in DSFS is a result of the difficulty of quantifying the displacement field complexity in the vicinity of the query point. However, recent developments in ANN-based DIC, being capable of pixel-wise displacement and strain measurement between image pairs, suggest that CNNs offer an attractive solution to this problem. This section briefly discusses an approach taken to investigate the potential of CNNs for DSFS and aims to lay foundation for future work.

7.4.1 Background

DSFS is attractive for two reasons. Firstly, there are many applications of DIC that lead to poor quality image sets which increase the influence of noise in the image set, such that doubling of the random error can lead to significantly poorer quality displacement measurements. One particular example is the use of high-speed cameras, which trade acquisition speed for image quality, to investigate non-quasi-static behaviour of materials [33, 34, 35].

Secondly, testing done within the field of experimental solid mechanics typically exposes the specimen under consideration to forces of increasing magnitude. Thus, the specimen initially undergoes a linear elastic deformation, which is typically well accounted for by the first-order SF, followed by plastic deformation, which requires higher order SFs to be accurately measured. Furthermore, based on the geometry of the specimens, these plastic deformations occur within a localised region corresponding to stress concentrations. The second-order SF is typically only relevant for a limited portion of the subsets in the tail end of the image set. Increasing the random error in the measured displacements due to the need, of a limited number of subsets, for the higher order SF is a subpar compromise. A framework for appropriate selection of the SF order on a per-subset and per-image basis is attractive in such scenarios.

Xu et al. [116] derived equations for the SFAB induced systematic errors, associated with the first-order SF in the presence of second-order displacement field, based on the second-order displacement gradients at the subset centre. These systematic errors in the x-direction (ME_x) and y-direction (ME_y) are given as

$$ME_x = \frac{M-1}{12}\left(\frac{M-1}{2}+1\right)\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right) \tag{7.1}$$

$$ME_y = \frac{M-1}{12}\left(\frac{M-1}{2}+1\right)\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right) \tag{7.2}$$

These relations show how the second-order displacement gradient, not accounted for by the first-order SF, dictate the severity of the systematic error. More specifically, the systematic error in the x-displacement (u) increases as its second-order derivatives in the x-direction ($\frac{\partial^2 u}{\partial x^2}$) and y-direction ($\frac{\partial^2 u}{\partial y^2}$) increase. Similarly, the systematic error in the y-displacement (v) increases as its second-order derivatives in the x-direction ($\frac{\partial^2 v}{\partial x^2}$) and y-direction ($\frac{\partial^2 v}{\partial y^2}$) increase. Furthermore, this relation quantifies how increasing the subset size (M), such that it incorporates pixels farther from the query point experiencing increasing incongruent displacements, increases the systematic errors.

This highlights how an appropriate combination of subset size and SF is required for the displacement field complexity in the vicinity of the query point such that SFAB errors can be mitigated. In fact, for the first-order SF, these systematic errors increase linearly relative to the second-order displacement gradients and quadratically relative to the subset size.

Although Equations (7.1) and (7.2) offer an attractive approach to DSFS between first and second-order SFs, they have not been used in practice due to the difficulty in measuring the second-order displacement gradient.

7.4.2 CNN development

It was found that the method of Sur et al. [286] produces speckle patterns that are more representative of those typical within experimental solid mechanics applications relative to the method employed in Chapter 6. As such, it was employed to generate a database of image pairs for training and validation of an ANN to quantify second-order displacement gradient in the x-direction directly from image pairs. With the developed CNN intending to serve as a proof of concept, a limited scope of training data was created for time considerations. In particular, only the speckle size and percentage coverage were varied randomly while other parameters of [286] were kept constant as shown in Table 7.1.

Table 7.1: Parameters used to generate synthetic speckle pattern images.

Parameter	Speckle size	PSF[24]	Percentage coverage	Contrast
Value	0.5 - 1	1	30% - 60%	0.6

These parameters ensure high speckle pattern information content within the generated images, as shown in Figure 7.3, such that a smaller input image can be used for the CNN; reducing the number of weights and biases of the CNN and accelerating training. An exponential distribution of speckle sizes was employed with single precision computation, which was shown to be sufficient for 8-bit images [286]. Each image set consists of a reference and deformed image of size 31 by 31 pixels.

Figure 7.3: Example images of the database of image pairs generated.

---

[24] The point-spread function (PSF) controls how smooth the transition in light intensity is at the speckle boundaries.

Deformed images were created using a sinusoidal displacement field given as

$$\begin{aligned} u(x,y) &= 0.5 \times \sin\left(\frac{2\pi}{\vartheta}x - \varpi\right) \\ v(x,y) &= 0 \end{aligned} \tag{7.3}$$

where, ϑ (period) and ϖ (shift) were randomised within the ranges $\{\vartheta \in \mathbb{R} | 80 \leq \vartheta \leq 700\}$ and $\{\varpi \in \mathbb{R} | -350 \leq \varpi \leq 350\}$, respectively. The second-order displacement gradients are given as

$$\begin{aligned} \frac{\partial^2 u}{\partial x^2} &= -2\left(\frac{\pi}{\vartheta}\right)^2 \times \sin\left(\frac{2\pi}{\vartheta}x - \varpi\right) \\ \frac{\partial^2 u}{\partial y^2} &= 0 \\ \frac{\partial^2 v}{\partial x^2} &= 0 \\ \frac{\partial^2 v}{\partial y^2} &= 0 \end{aligned} \tag{7.4}$$

Random homoscedastic Gaussian noise of standard deviation, $\psi = 1$, was applied to these image sets. Finally the images are quantified as 8-bit images. A total of 46140 image sets were generated which were divided in a ratio of 0.8:0.1:0.1 for the training, validating and testing datasets, respectively.

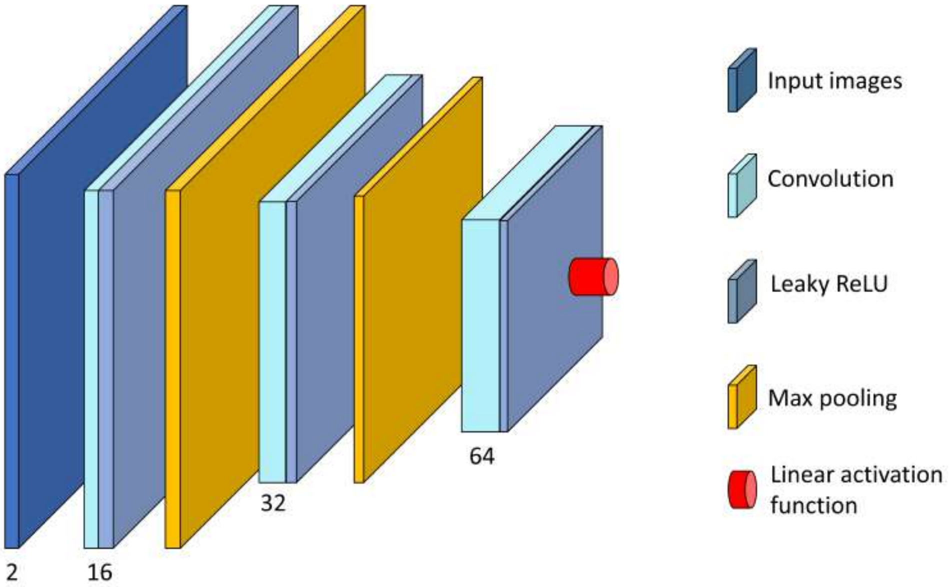


Figure 7.4: Illustration of the architecture of the CNN for DSFS. The numbers under the layers indicate the depth of the associated feature maps.

The architecture of the CNN, developed through trial-and-error, consists of 3 pairs of convolution and Leaky ReLU activation layers with max pooling performed between each pair. Convolution activation functions have a filter size of 3 by 3 pixels with depth increasing in multiples of 16, as indicated in Figure 7.4, and a stride of

1. This preserves the size of the input feature map while doubling its depth. Max pooling with a filter size of 3 by 3 pixels and stride of 2 is used to reduce the dimensionality of the feature maps to reduce the number of weights and biases that need to be trained. Thus, max pooling reduces the dimensionality of the feature maps while convolution layers increase the depth of the feature maps, progressively building higher level representations of the data such that deep features can be detected. Batch normalisation was performed between convolutional and Leaky ReLU activation layers to mitigate the possibility of vanishing gradients and accelerate training [316]. The output layer consists of a single neuron with a linear activation function which is fully connected to all the elements of the final feature maps.

The network constitutes a CNN to perform regression, taking in an image pair of size 31 by 31 pixels and predicting $\frac{\partial^2 u}{\partial x^2}$ occurring at the centre.

The CNN was trained on the training dataset using the half mean square error loss function, that is Equation (2.6) divided by two, and the Adam optimisation algorithm [157] with default training parameters. After 190 epochs, the loss of the training and validation datasets were 5.2×10^{-4} and 7.2×10^{-4}, respectively. This enabled the CNN to predict the second-order x-displacement gradients of the testing dataset with an accuracy and precision of 6.6×10^{-4} and 0.01, respectively.

7.4.3 Artificial neural network based dynamic shape function selection (ANNDSFS) implementation

Prior to the artificial neural network based dynamic shape function selection (ANNDSFS) method, the query points are placed in a regular grid spanning the ROI according to the `ADIC2D` or `ADIC3D` function; since the method is compatible with both 2D and 3D DIC. Thereafter, ANNDSFS consists of three steps, illustrated in Figure 7.5, as follows:

Step 1: Determining the second-order displacement gradients using the proposed CNN. The process begins with the first image pair of the image set (d=1). For each query point (q) a window of 31 by 31 pixels, centred at the query point, is extracted from the reference and deformed images under consideration which are processed by the CNN to predict the second-order x-displacement gradient ($\frac{\partial^2 u}{\partial x^2}$). Equation (7.1) predicts the associated systematic error ($ME_x(d,q)$) based on the subset size. This process is repeated for each image pair of the DIC analysis building up matrices of the predicted systematic error.

Step 2: Stipulation of the systematic error threshold (ME_x^{thresh}). A plot of the predicted second-order x-displacement gradient vs. the predicted systematic error is presented to the DIC practitioner such that an appropriate systematic error threshold can be chosen.

Step 3: Assigning the appropriate SF orders. By looping through the deformed images the appropriate SF order across all query points is appointed as: `ProcData(d).SFOrder(q)=1` if $ME_x(d,q)$ is below ME_x^{thresh} since this indicates that the first-order SF is appropriate for the stipulated threshold; or `ProcData(d).SFOrder(q)=2` if $ME_x(d,q) > ME_x^{thresh}$ since the first-order SF would cause too high of a systematic error.

Thereafter, conventional correlation is performed using the appointed SF orders. This process enables ANNDSFS to adjust the assigned SF orders on a per-subset and per-image basis according to the spatially and temporally changing displacement field complexity.

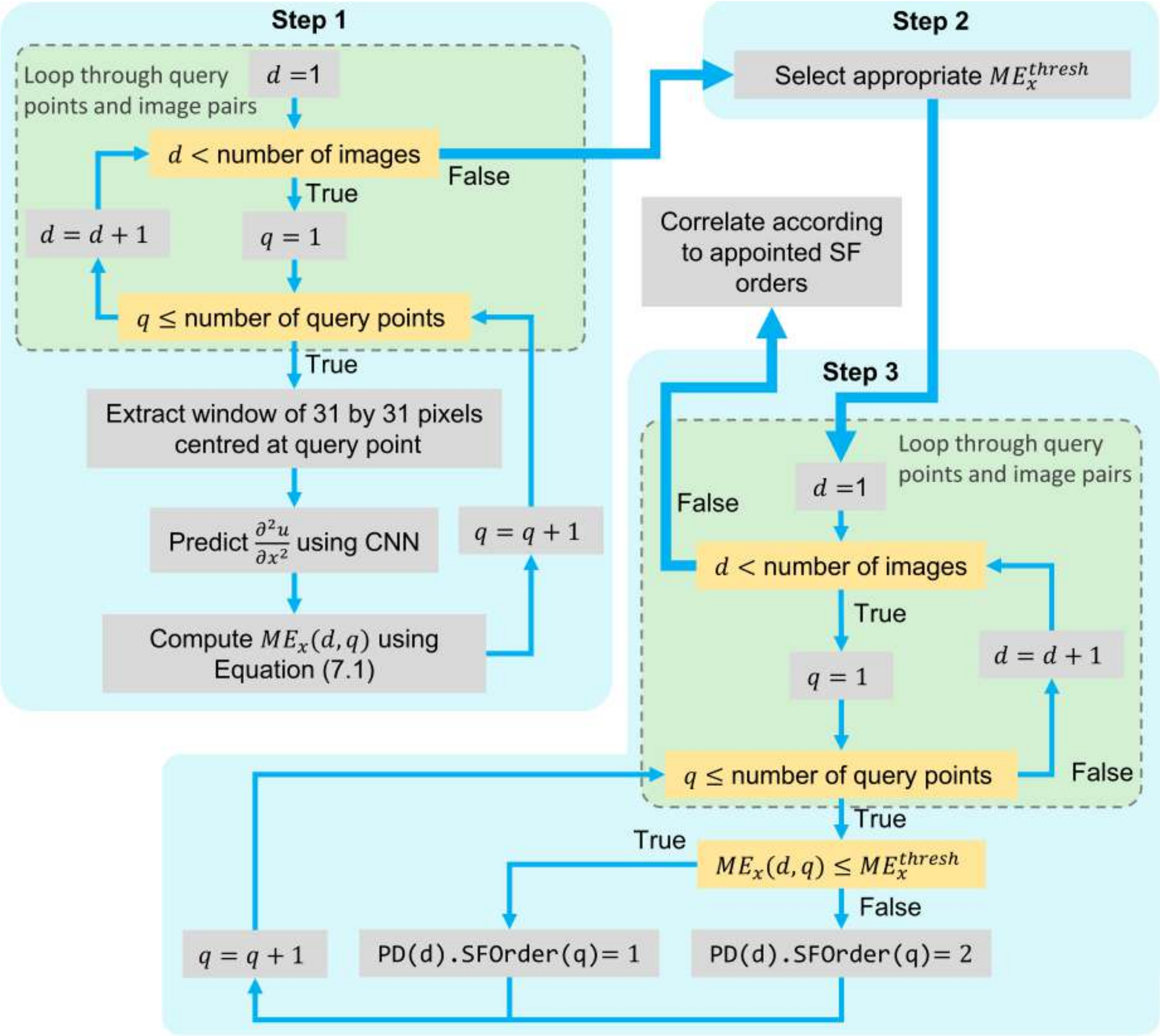


Figure 7.5: Flow diagram illustrating the process of the proposed ANNDSFS method.

7.4.4 Performance of ANNDSFS

The performance of ANNDSFS is assessed on a test image set generated by the method of Sur et al. [286] using the parameters of Table 7.2. The images are created to be of size 1400 by 250 pixels. The deformed image has an imposed sinusoidal displacement of increasing frequency (decreasing period) in the x-direction (period decreasing from 500 to 160 pixels) by setting ϑ of Equation (7.3) to

Table 7.2: Parameters used to generate the test image set.

Parameter	Speckle size	PSF[24]	Percentage coverage	Contrast
Value	0.917	1	30% - 60%	0.6

$$\vartheta = 500 - 360\frac{x}{1400} \tag{7.5}$$

The test image set is analysed by four cases corresponding to four different configurations of setting up the correlation problem as detailed in Table 7.3. Case 1 and 2, using first and second-order SFs, respectively, to analyse the image set for a traditional approach of globally assigned correlation parameters. Case 3 makes use of ANNDSFS to appoint independent SF orders for each query point based on the method outlined in Section 7.4.3. Case 4 uses ANNDSS (of Chapter 6) to appoint independent subset sizes while ANNDSFS appoints independent SF orders. Moreover, ANNDSS is performed as a preprocess to ANNDSFS, which then predicts the ME_x of each query point based on its subset sizes and predicted $\frac{\partial^2 u}{\partial x^2}$ such that an appropriate SF order is assigned. All cases use the same subset positions and do not employ image filtering.

Table 7.3: Correlation parameters used to analyse the test image set.

Case	Step size	Subset size	SF order
1	5	31	1
2	5	31	2
3	5	31	ANNDSFS
4	5	ANNDSS	ANNDSFS

The trend in DESD predicted by ANNDSS is shown in Figure 7.6(a). A threshold of σ^{thresh}=0.01, corresponding to the end to the transition regime, was chosen to ensure sufficient noise suppression. The trend in the systematic error is shown in Figure 7.6(b) for Cases 3 and 4. This figure shows how using Equation (7.1) enables ANNDSFS to predict ME_x based on $\frac{\partial^2 u}{\partial x^2}$ and the changing subset sizes for Case 4. A threshold of ME_x^{thresh}=0.01 is selected, for both Case 3 and 4, such that the random and systematic errors are on par with one another.

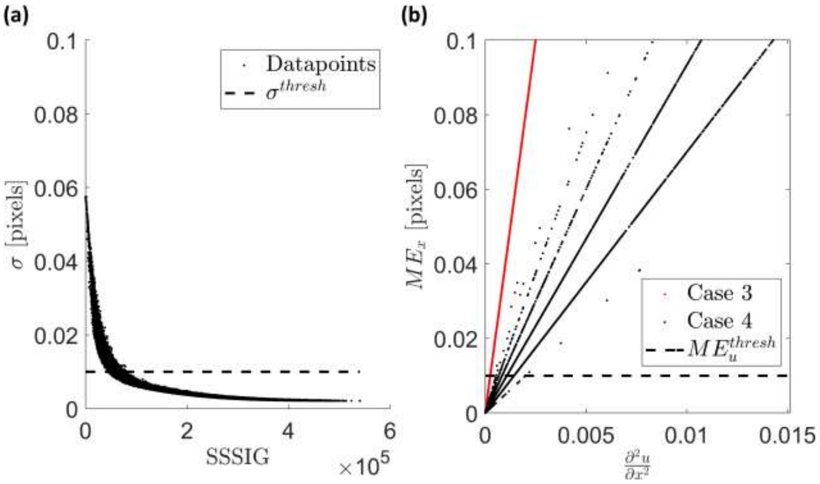


Figure 7.6: (a) Random error trend predicted by ANNDSS. (b) Systematic error trend predicted by ANNDSFS for Case 3 (red) and Case 4 (black).

The speckle pattern of the reference image is shown in Figure 7.7(a). Figure 7.7(b) shows how the subset sizes assigned by ANNDSS adjust according to the local speckle pattern quality. Due to the high information content of the speckle pattern, most of the subset sizes assigned are significantly smaller than 31 pixels as was used for Cases 1-3.

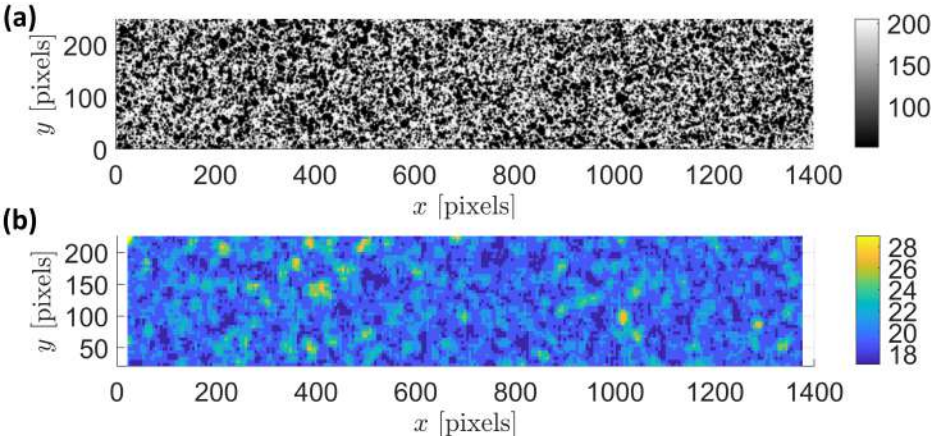


Figure 7.7: (a) Reference image of the test image set. (b) Subset sizes assigned by ANNDSS for Case 4.

The true displacement field in the x-direction is shown in Figure 7.8(a) illustrating how the displacement frequency increases in the x-direction. The SF orders assigned by ANNDSFS are shown in Figures 7.8(b) and 7.8(c) for Case 3 and 4, respectively, indicating how ANNDSFS assigns first and second-order SFs for regions of the image subject to lower and higher frequency displacements, respectively.

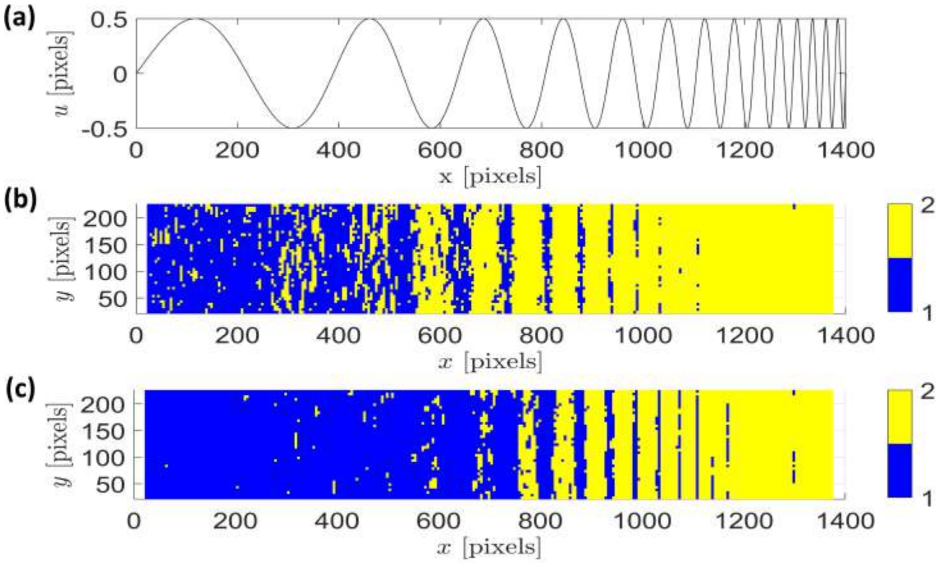


Figure 7.8: (a) Imposed displacement in the x-direction. (b) SF orders assigned by ANNDSFS for Case 3. (c) SF orders assigned by ANNDSFS for Case 4.

Note that the ANNDSFS method appoints first and second-order SF orders in bands corresponding to the regions of imposed displacement field that have low and high second-order displacement derivatives, respectively. Furthermore, the SF orders assigned differ between Cases 3 and 4 due to the different subset sizes. The smaller subset sizes of Case 4 enable the first-order SF to accurately track the underlying displacement field for higher frequencies than that of Case 3 (observable for x<750).

The test image set is correlated for each case using the approach of Section 6.3.2. More specifically, each subset is correlated 150 times with noise of ψ=1 reapplied to the noise-free image each time. Subtracting the computed x-direction displacement from the true value results in a set of 150 errors for each subset. The mean and standard deviation of these errors is taken, for each subset, to determine its systematic and random errors, respectively. Thereafter, these error metrics are averaged across subsets of the same x position to determine the trend in systematic (ME_x) and random errors (σ_x) in the x-direction as illustrated Figures 7.9(a-d) for Cases 1-4.

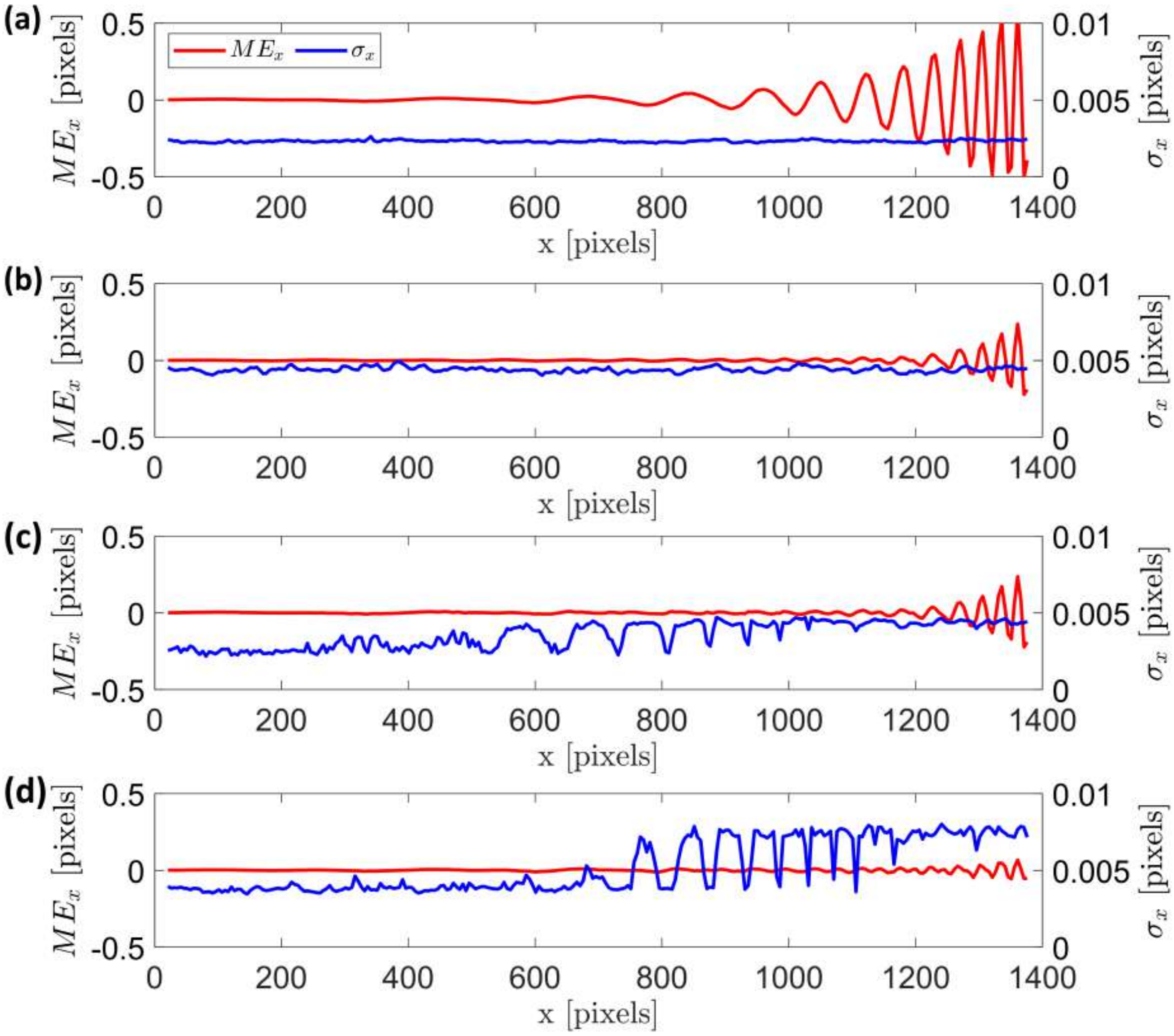


Figure 7.9: Systematic error in the x-direction for: (a) first-order SF (Case 1); (b) second-order SF (Case 2); (c) SFs assigned by ANNDSFS (Case 3); and (d) subset sizes assigned by ANNDSS and SFs assigned by ANNDSFS (Case 4).

Cases 1 and 2 illustrate the well-established metrological characteristics of the

SF. In particular, as the frequency of the displacement increases in the x-direction the systematic error increases as the SF becomes incapable of accounting for the underlying deformation. Although the second-order SF is capable of accurately tracking the displacement for higher frequencies relative to the first-order SF, this comes at the cost of doubling the random errors due to the second-order SF having double the SFPs relative to the first-order.

Case 3 shows the benefit of the DSFS method. Moreover, ANNDSFS assigns first and second-order SFs for regions of negligible and appreciable second-order displacement gradients, respectively. As such, it offers improved random errors relative to Case 2 (specifically for x<500) and improved systematic errors relative to Case 1 (for x>800). Thus, optimal SF orders are assigned independently for the query points such that the quality of the displacement field is not dictated by regions of complex deformation.

Case 4 illustrates two points. Firstly, the importance of appropriate subset size selection with regards to systematic errors. Although the smaller subset sizes of Case 4 marginally increase the random error, they offer significantly improved systematic errors as they enable the second-order SF to more accurately account for the deformation experienced by these smaller subset sizes. This occurs because the sinusoidal displacement field is of higher order than the second-order SF. Thus, a smaller subset size improves its ability to approximate the underlying deformation. In practice, displacement fields are often more complex than the second-order SF can directly account for. Thus, the hypothesis of Chapter 6 in that *the smallest subset size offering sufficient noise suppression, consistent with the transition regime, offers a favourable compromise between noise suppression and SR* is presumably relevant to any SF which is of lower order than the complexity of the underlying displacement field. Furthermore, this hypothesis offers a viable and attractive approach for appropriate subset size selection to offer a favourable compromise between noise suppression and SR.

Secondly, the benefit of dynamic correlation parameter assignment relative to the tradition of global correlation parameter assignment. Case 2 is a good example of an initial choice of correlation parameters for the traditional trial-and-error approach to global correlation parameter selection. The second-order SF is appropriate due to the complexity of the displacement field and a global subset size of 31 pixels is a good choice as it ensures sufficient speckle pattern information within each subset. This is supported by Figure 7.7(b) where regions of poor speckle pattern quality require a subset size of 28 pixels to ensure random errors below σ_x=0.01 pixels. However, the quality of the resulting displacement field is dictated by the regions of the image set of poor speckle pattern quality and high displacement field complexity, leading to subpar results relative to Case 4.

In contrast, the proposed ANN-based method for dynamic correlation parameter assignment appoints an appropriate subset size (ANNDSS) and SF order (ANNDSFS) combination for each query point independently. Assigning these parameters based on the speckle pattern quality and displacement field complexity in the vicinity of the query point, enables high quality displacement measurements for each query point, relative to what is achievable (for the standard options of square subsets and first or second-order SF). This is the reason for the dips in the random error curves of Cases 3 and 4 for x>750. These dips correspond to ANNDSFS assigning first-order SFs for regions within the high frequency displacement field which correspond to negligible second-order displacement gradients; thus reducing random errors whilst remaining capable of accurately tracking the displacement.

7.4.5 Discussion of ANNDSFS

The results indicate that CNNs are a viable approach for DSFS, between first and second-order SFs, capable of quantifying the complexity of the underlying displacement field. In particular, they are able to predict the second-order displacement gradient which, using the equations of Xu et al. [116], can be used to predict the systematic error for SFAB based on the subset size. Selection of a systematic error threshold offers an intuitive approach for DSFS while presenting a plot of the predicted systematic error facilitates an informed selection of the threshold. Furthermore, this CNN is capable of DSFS between each image pair such that appropriate SF orders can be assigned temporally as as well as spatially.

The proposed ANNDSFS method is capable of adjusting to location specific subset sizes such that it can be combined with ANNDSS. Coupling ANNDSFS with ANNDSS offers an attractive approach to dynamic correlation parameter assignment. Correlation parameter selection is typically challenging because of the complex interactions which dictate the metrological characteristics of the correlation process, and the difficulty in quantifying speckle pattern quality and displacement field complexity in relation to the correlation parameters. In contrast, stipulation of intuitive random and systematic error thresholds enables: (i) ANNDSS to assign appropriate subset sizes spatially; and (ii) ANNDSFS to assign appropriate SF orders spatially and temporally, based on these subset sizes. Furthermore, the method predicts the trend in these error metrics for the image set enabling an informed decision to be made with regards to these error thresholds. Such an approach enables appropriately setting up a DIC analysis on a per-subset and per-image basis such that high quality displacement measurements (relative to what is achievable) can be realised across the image set regardless of localised low quality speckle patterns, and/or localised or time dependent displacement field complexity. Furthermore, these methods are based on purely image information enabling them to operate as a pre-process to any DIC algorithm capable of assigning correlation parameters on a per-subset and per-image basis. This motivates the need to extend the ANNDSS method of Chapter 6 to predict random displacement errors associated with the second-order SF; such that appropriate subset sizes can be appointed in regions for which ANNDSFS assigns the second-order SF.

The generalisability of the CNN is limited in its current form due to the limited scope of its training dataset[25]. As such the scope of the training dataset should be improved by including: (i) a greater variety of displacement fields; (ii) discontinuous displacement fields such that the method can function in the vicinity of a crack (since second-order SFs are usually required at a crack tip due to its stress concentration effect); (iii) greater variation in the speckle size, speckle coverage, percentage coverage of the speckle pattern, contrast, standard deviation of noise and PSF values; and (iv) contrast changes between image pairs.

Additionally, the architecture of the CNN should be revised to improve upon the design limitations of the current implementation. Firstly, the current CNN only

---

[25] This serves as a preliminary investigation into the use of CNNs for DSFS.

considers $\frac{\partial^2 u}{\partial x^2}$ whereas $\frac{\partial^2 u}{\partial y^2}$, $\frac{\partial^2 v}{\partial x^2}$ and $\frac{\partial^2 v}{\partial y^2}$ are required for fully informed DSFS. Secondly, the proposed CNN operates on a small window of 31 by 31 pixels, which is assumed to break down for sparse speckle patterns.

More specifically, it is reasonable to assume that CNN based displacement and displacement gradient measurement (which consist of direction and magnitude) from scalar intensity information is an ill-posed problem similar in nature to that of DIC (as discussed in Section 2.1.2). The CNNs accuracy will decrease in the presence of sparse speckle patterns as fewer features (speckle boundaries) fall within the window processed, such that fewer feature detectors of the CNN become activated which are meant to guide its prediction. Although using a larger window could alleviate this, inclusion of displacements occurring farther from the centre of the window will reduce the accuracy of the predictions in severe displacement fields.

Future work should adopt the use of an encoder-decoder CNN similar to the approach of Yang et al. [93] for displacement and strain measurement. The encoder aspect consists of a series of convolution layers which process the image creating feature maps of decreasing dimensionality and increasing depth. This enables the extraction of higher level features from the sparse speckle data, which are stored within these feature maps. Thereafter, the decoder, consisting of a series of transposed convolution layers, reverses the process increasing the dimensionality of the features maps while reducing their dimensionality to recover the high resolution second-order displacement gradients.

This CNN architecture is powerful because it can be designed to operate on large window sizes, thus ensuring sufficient speckle pattern information is present for robust prediction, and affords pixel-wise prediction of the four second-order displacement gradients. This pixel-wise displacement gradient information is attractive since it can allow for more informed assignment of SF orders by considering the highest second-order displacement gradients present within the subset, not only those occurring at its centre.

Additionally, this second-order displacement gradient information is potentially attractive for weighted subset based DIC methods which traditionally apply an axis [283] or line symmetric [289] weight distribution to pixels of the subset. The weighting of the pixel controls the relative influence of its associated image gradient in guiding the iterative updates applied to the SFPs during the correlation process. Thus, by reducing the weight of pixels farther from the query point which, are assumed in a simplistic sense, to be experiencing incongruent motion relative to the query point this method reduces SFAB. However, the availability of pixel-wise second-order displacement gradient information would enable informed pixel-wise weight application, reducing the weighting of pixels which experience more severe second-order displacement gradients. The severity of the second-order displacement gradient could be used to determine an informed initial estimate of the weighting parameters which are typically optimised during correlation alongside the SFPs.

7.5 Future work

This work shows that ANNs offer an attractive approach to dynamic correlation parameter assignment by: (i) rephrasing the obscure question of *what global* *correlation parameters are optimal* to *what degree of random and systematic errors in the computed displacement is appropriate*; (ii) providing error prediction which allows the user to infer the quality of the computed displacement field; and (iii) operating as a pre-process to DIC offering DSS and DSFS based on purely image information. In addition to the objective-specific recommendations for future work provided in the previous sections, this section outlines additional recommendations for the use of ANNs in the context of streamlining dynamic correlation parameter assignment for local DIC.

Progress within the field of machine learning over the last decade has shown that CNNs are a powerful tool for extracting data from images. Thus, leveraging the capabilities of CNNs to determine speckle pattern quality directly from an image is an attractive prospect. This is supported by the recent (October 2022) work of Kwon et al. [97] which predicts the average displacement error of a DIC analysis from the speckle pattern of the reference image. However, this first attempt at using CNNs for assessment of speckle pattern quality focuses on quantifying the quality of the global speckle pattern and as such is not applicable to DSS. As such, it is recommended that future work in this area focuses on the development of CNNs to predict the random displacement error associated with a subset directly from its contained speckle pattern.

This prospect is attractive since this would bypass the need to rely on SPQMs to quantify the quality of the speckle patterns. Although SPQMs are helpful in guiding the selection of subset sizes as shown in Chapter 6, they do not capture all characteristics of a speckle pattern that make them favourable for DIC. A CNN can automatically discover what characteristics of a speckle pattern make it favourable for DIC through building progressively higher level feature detectors guided by the loss function. Choosing between systematic or random errors for the output of the CNN will enforce these feature detectors to identify features relevant to PIB or noise suppression, respectively. This offers an attractive approach to quantify all characteristics of a speckle pattern that make it favourable, or detrimental, for DIC. Furthermore, SPQMs are computationally expensive, leading to long run times for the ANNDSS framework. In contrast, CNNs are computationally efficient, enabling rapid evaluation. It is for this reason that ANNDSFS is ten times more computationally efficient than ANNDSS.

A concern of such an approach is that CNNs typically require the input image to be of specific dimensions as the fully connected output layer expects the feature maps to be of a specific size. The approach taken by Kwon et al. [97] to compress the input image to the size required by the CNN is ill advised as speckle pattern information will be lost and can lead to aliasing of speckles which are likely to bias the predictions of the ANN. However, recent work has shown that CNNs can be designed such that they are immune to the size of the input image through the use of spatial pyramidal pooling. Additionally, spatial pyramidal pooling has been shown to be robust to deformations of objects within the image [317]. The use of such pooling techniques would enable the creation of a single CNN which can directly determine the random displacement error associated with a speckle pattern for a wide range of subset sizes.

Coupling such a CNN based DSS method with the CNN based DSFS method recommended in Section 7.4.5 would offer an attractive approach to dynamic assignment of correlation parameters. In particular, these methods would enable quantifying the speckle pattern quality and displacement field complexity directly from images which can be related, through theoretical derivations or ANN models, to the resulting errors in the DIC computed displacements. This would facilitate dynamic assignment of correlation parameters, guided by knowledge of their impact of the metrological performance of the correlation process, on a per-subset and per-image basis.

Generalisability concerns are not unique to the proposed ANN. Many ANN-based DIC methods break down for particular types of displacement fields common within DIC applications. In particular, StrainNet [25] cannot track displacements larger than a pixel while Deep DIC [93] performs poorly for simple RBT. Although the solution to this is obvious through improving the scope of training dataset, this is not straightforward in practice. Over the last four decades, the use of DIC has been extended to a broad range of applications each introducing unique image set properties. Generating a database of image sets which encompass such a broad range of image set properties is challenging both in terms of development of associated algorithms and computational resources.

This suggests the need for a collective effort to generate, and possibly capture through well designed experiments, a well-vetted database of image sets which contain a broad range of speckle patterns and displacement fields typical of the widespread applications of DIC. Moreover, speckle patterns should include both natural and applied speckle patterns while displacement fields should include discontinuities, stress concentrations, RBT, rotation, high strain, out-of-plane motion and vibrations, to name a few. The availability of such a database would facilitate the development of more robust ANN-based DIC techniques. Furthermore, it would expedite research directed towards the use of ANNs to improve traditional local DIC. This is argued to be a future area of focus within the DIC community with recent efforts (in October of 2022) focusing on quantifying the quality of global speckle patterns [97] and determination of initial estimates of SFPs [96] directly from images using CNNs.

Additionally, a subset of these image sets can be retained for performance assessment purposes similar to the function of the ImageNet Large Scale Visual Recognition Competition [66] and DIC Challenge image sets[26] [122, 123]. Validation of these methods would be beneficial for several reasons including: (i) revealing the true capabilities of these methods for real world applications across the range of challenges traditionally faced by DIC; (ii) identify specific applications for which they struggle (for example large variations in illumination between image pairs not yet investigated for ANN-based DIC) such that methods can be designed to account for these; (iii) investigating the metrological characteristics of these methods; (iv) relating their metrological characteristics to architectural design and training decisions to inform future research in this area of best practices and areas requiring improvement; (v) knowledge of the metrological characteristics of these methods would enable the development of corrective measures to mitigate their shortcomings as well as the development of guidelines for their effective use.

---

[26] The DIC Challenge image sets are not applicable to this purpose due to their limited range of speckle pattern characteristics and displacement fields.

Such a database is not only attractive to expedite research in this area, but is arguably necessary to take these ideas from the conception phase to deployment; where they can be confidently recommended for practical use.

Chapter 8
Conclusion

This project was motivated by three objectives as detailed in Section 3.3. A modular, open-source DIC framework, for 2D and stereo applications, was developed offering full control over setting up the correlation problem on a per-subset and per-image basis. This framework was subsequently used to develop and validate an ANN to model the relationship between SPQMs, image noise and the resulting random displacement error to facilitate DSS. Furthermore, initial investigations were conducted into the use of a CNN to spatially quantify the displacement field complexity between an image pair for the purpose of DSFS. The results indicate the potential of ANNs for dynamic assignment of correlation parameters while the limitations of the current implementation guides future research directions.

This work shows, through Chapters 4-6, that ANNs are a viable approach for DSS from purely image information by modelling the complex relationship between subset size, image set properties and resulting random displacement error as hypothesised in Section 3.4.

Additionally, Section 7.4 reveals that ANNs are feasible as a tool for quantifying displacement field complexity between image pairs offering a viable tool for DSFS.

Key contributions and future work recommendations are outlined below:

- A modular, open-source 2D DIC framework is developed and implemented in 117 lines of MATLAB code. This DIC framework, being predominantly consistent with current state-of-the-art practices, performs on par with established open-source and commercial DIC algorithms. In addition to this framework being modular in terms of allowing straightforward adaptation and interchange of the DIC subprocesses, it offers full control over setting up the correlation problem by enabling assignment of the subset size, subset shape and SF on a per-subset and per-image basis; thereby addressing Objective 1. This removes one of the barriers to dynamic assignment of correlation parameters which is a unique contribution of this work.

- The DIC framework is extended to stereo DIC which is implemented in 202 lines of MATLAB code to address Objective 2. This bridges the gap in literature between the well-documented theory of DIC and its nuanced practical implementation as code, by directly drawing links between the two to document the framework's implementation. This removes the difficult learning curve faced by newcomers to the field of DIC who intend to contribute to its development which is a unique contribution of this work. Furthermore, this documentation combined with the framework's validated performance and modular implementation make it attractive as a starting point to further the capabilities of both 2D and stereo DIC.

- The DIC framework is used to develop and validate a feed-forward ANN to model the relationship between the speckle pattern quality (contained within a subset), standard deviation of image noise and resulting DESD, in line with Objective 3. This can be seen as a unique contribution to the field of DIC. This ANN predicts DESD more accurately and precisely than theoretical derivations, for speckle patterns consistent with the training scope, by accounting for all error sources involved. As such, this approach should be applicable to any DIC algorithm, regardless of DIC subprocesses, without requiring simplification of the mathematical complexity.

- Employing the ANN as an error prediction tool, a framework for DSS is developed to appoint subset sizes based on local speckle pattern quality to address Objective 3. Validation conducted on speckle pattern characteristics consistent with the training scope indicates that this framework: (i) appoints each subset's size as the smallest which offers noise suppression consistent with the DESD threshold stipulated; (ii) offers improved DESD results relative to the traditional trial-and-error approach of global subset size selection, for the same mean subset size, in the presence of varying speckle pattern quality; (iii) offers a favourable compromise between noise suppression and spatial resolution for up to moderate displacement gradients for stipulation of an appropriate DESD threshold, which is guided by the predicted trend in noise suppression and *a priori* knowledge of the expected displacement field complexity; (iv) rephrases the obscure question of *what global subset size is appropriate for an analysis* to *what amount of random error within the computed displacements is acceptable*; and (iv) knowledge of the contribution of image noise to the DESD enables the inference of the quality of the computed displacement field. Validation on speckle pattern characteristics outside the training scope indicates the generalisability limitations of this approach in its current form, motivating the need to broaden the scope of its training dataset. DSS from purely image information through the use of an ANN, such that it can appropriately set up subset sizes prior to (and independently of) the DIC process, is a unique contribution of this work.

- Investigation of the viability of CNNs for DSFS is initiated. This shows that CNNs are capable of measuring region-specific displacement complexity (in the form of second-order displacement gradients) with sufficient accuracy and precision to facilitate DSFS, for the limited scope of displacements and speckle patterns investigated. ANN-based DSFS from purely image information is a unique contribution of this work, despite the limited scope of the DSFS method developed.

- This work shows that ANNs are a viable approach for dynamic correlation parameter assignment from purely image information. In particular, ANNs are capable of modelling the complex interaction between subset size, image set properties and resulting random displacement error facilitating DSS. Furthermore, CNNs are capable of spatially quantifying the underlying displacement field complexity (in the form of second-order displacement gradients) guiding DSFS. Such an approach to dynamic correlation parameter assignment can act as a pre-process to DIC, setting up the correlation problem through intuitive selection of informed error thresholds.

- Future research endeavours towards dynamic correlation parameter assignment are highly recommended to leverage the capabilities of CNNs. Furthermore, this work indicates that such efforts should focus on development of a training dataset covering a broad range of speckle pattern characteristics and displacement fields to ensure generalisability, and thus practical applicability, of developed methods.

Bibliography

1 Hummel, R. E. (2004). *Understanding materials science*. 2nd. New York, NY: Springer New York.

2 Rockett, A. (2008). *The materials science of semiconductors*. Boston, MA: Springer US.

3 Pham, M. H. et al. (2015). "Material science in cervical total disc replacement". In: *BioMed Research International* 2015 719123, pp. 1–9.

4 Klinkenberg, E.-D. and Neumann, H.-G. (2014). "Surface Design for Medical Implants". In: *Contributions to Plasma Physics* 54 2, pp. 131–137.

5 Ballarino, A et al. (1996). "Design of 12.5 kA current leads for the Large Hadron Collider using high temperature superconductor material". In: *16th International Cryogenic Engineering Conference and International Cryogenic Materials Conference*. Kitakyushu, Japan, pp. 1143–1146.

6 Burns, H., Whitaker, A., and Linton, R. (1989). "Atomic oxygen resistant protective coatings for the hubble space telescope solar array in low earth orbit". In: *Surface and Coatings Technology* 39-40, pp. 627–636.

7 Fedorov, P. (2020). "Nanotechnology and material science". In: *Nanosystems: Physics, Chemistry, Mathematics* 11 3, pp. 314–315.

8 Jones, E. and Iadicola, M. (2018). "A good practices guide for digital image correlation". In: *International Digital Image Correlation Society* 10, pp. 1–110.

9 Peters, W. H. et al. (1983). "Application of digital correlation methods to rigid body mechanics". In: *Optical Engineering* 22 6, pp. 738–742.

10 Chu, T. C., Ranson, W. F., and Sutton, M. A. (1985). "Applications of digital-image-correlation techniques to experimental mechanics". In: *Experimental Mechanics* 25 3, pp. 232–244.

11 Bruck, H. A. et al. (1989). "Digital image correlation using newton-raphson method of partial differential correction". In: *Experimental Mechanics* 29 3, pp. 261–267.

12 Schreier, H. W., Braasch, J. R., and Sutton, M. A. (2000). "Systematic errors in digital image correlation caused by intensity interpolation". In: *Optical Engineering* 39 11, pp. 2915–2921.

13 Lu, H. and Cary, P. D. (2000). "Deformation measurements by digital image correlation: implementation of a second-order displacement gradient". In: *Experimental Mechanics* 40 4, pp. 393–400.

14 Tong, W. (2005). "An evaluation of digital image correlation criteria for strain mapping applications". In: *Strain* 41 4, pp. 167–175.

15 Baker, S. and Matthews, I. (2004). "Lucas-kanade 20 years on: a unifying framework". In: *International journal of computer vision* 56 3, pp. 221–255.

16 Barranger, Y. et al. (2010). "Digital image correlation accuracy: Influence of kind of speckle and recording setup". In: *EPJ Web of Conferences* 6, p. 31002.

17 Amiot, F. et al. (2013). "Assessment of digital image correlation measurement accuracy in the ultimate error regime: Main results of a collaborative benchmark". In: *Strain* 49 6, pp. 483–496.

18 Titkov, V. V. and Panin, S. V. (2019). “Measurement affecting errors in digital image correlation”. In: *IOP Conference Series: Materials Science and Engineering* 511 1, p. 012018.

19 Yaofeng, S. and Pang, J. H. (2007). “Study of optimal subset size in digital image correlation of speckle pattern images”. In: *Optics and Lasers in Engineering* 45 9, pp. 967–974.

20 Zhang, W., Zhou, R., and Zou, Y. (2017). “Self-adaptive and bidirectional dynamic subset selection algorithm for digital image correlation”. In: *Journal of Information Processing Systems* 13 2, pp. 305–320.

21 Hassan, G. M. et al. (2016b). “Digital image correlation with dynamic subset selection”. In: *Optics and Lasers in Engineering* 84, pp. 1–9.

22 Pan, B. (2018). “Digital image correlation for surface deformation measurement: historical developments, recent advances and future goals”. In: *Measurement Science and Technology* 29 8, p. 082001.

23 Schlechtingen, M. and Ferreira Santos, I. (2011). “Comparative analysis of neural network and regression based condition monitoring approaches for wind turbine fault detection”. In: *Mechanical Systems and Signal Processing* 25 5, pp. 1849–1875.

24 Yoo, H.-J. (2015). “Deep convolution neural networks in computer vision: A review”. In: *IEIE Transactions on Smart Processing and Computing* 4 1, pp. 35–43.

25 Boukhtache, S. et al. (2021). “When deep learning meets digital image correlation”. In: *Optics and Lasers in Engineering* 136, p. 106308.

26 Min, H.-G. et al. (2020). “Strain measurement during tensile testing using deep learning-based digital image correlation”. In: *Measurement Science and Technology* 31 1, p. 015014.

27 Wang, B. and Pan, B. (2016). “Subset-based local vs. finite element-based global digital image correlation: A comparison study”. In: *Theoretical and Applied Mechanics Letters* 6 5, pp. 200–208.

28 Bay, B. K. et al. (1999). “Digital volume correlation: three-dimensional strain mapping using x-ray tomography”. In: *Experimental Mechanics* 39 3, pp. 217–226.

29 Anuta, P. E. (1970). “Spatial registration of multispectral and multitemporal digital imagery using fast Fourier transform techniques”. In: *IEEE Transactions on Geoscience Electronics* 8 4, pp. 353–368.

30 Sutton, M. et al. (1983). “Determination of displacements using an improved digital correlation method”. In: *Image and Vision Computing* 1 3, pp. 133–139.

31 Read, J. M. (1996). “Fatigue cracking of bituminous paving mixtures”. PhD thesis. University of Nottingham.

32 Kang, J et al. (2007). “Digital image correlation studies for microscopic strain distribution and damage in dual phase steels”. In: *Scripta Materialia* 56 11, pp. 999–1002.

33 Flores, M. et al. (2017). “High-speed 3D digital image correlation of low-velocity impacts on composite plates”. In: *Composites Part B: Engineering* 131, pp. 153–164.

34 Li, E., Tieu, A., and Yuen, W. (2003). "Application of digital image correlation technique to dynamic measurement of the velocity field in the deformation zone in cold rolling". In: *Optics and Lasers in Engineering* 39 4, pp. 479–488.

35 Gao, X. et al. (2020). "High-speed 3D digital image correlation for rolling deformation of a tire sidewall and measuring dynamic contact patch length". In: *Applied Optics* 59 5, p. 1313.

36 Dickinson, A. S. et al. (2011). "Experimental validation of a finite element model of the proximal femur using digital image correlation and a composite bone model". In: *Journal of Biomechanical Engineering* 133 1, p. 014504.

37 Seo, Y. et al. (2004). "A study of crack-tip deformation and crack growth in asphalt concrete using fracture mechanics". In: *Technical Sessions of the Journal of the Association of Asphalt Paving Technologists.* Baton Rouge, Louisiana: Association of Asphalt Paving Technologists (AAPT), pp. 697–730.

38 Chehab, G. R., Seo, Y., and Kim, Y. R. (2007). "Viscoelastoplastic damage characterization of asphalt–aggregate mixtures using digital image correlation". In: *International Journal of Geomechanics* 7 2, pp. 111–118.

39 Kim, Y. R. and Little, D. N. (1990). "One-Dimensional Constitutive Modeling of Asphalt Concrete". In: *Journal of Engineering Mechanics* 116 4, pp. 751–772.

40 Kim, R et al. (2003). "Characterization of asphalt concrete in uniaxial tension using a viscoelastoplastic continuum damage model". In: *Journal of the Association of Asphalt Paving Technologists* 72, pp. 315–355.

41 ASTM E8/E8M (2016). *Tension Testing of Metallic Materials.*

42 ASTM E132 (2004). *Standard test method for Poisson's ratio at room temprature.*

43 Grédiac, M. et al. (2006). "The Virtual Fields Method for Extracting Constitutive Parameters From Full-Field Measurements: a Review". In: *Strain* 42 4, pp. 233–253.

44 Alkayem, N. F. et al. (2018). "Structural damage detection using finite element model updating with evolutionary algorithms: a survey". In: *Neural Computing and Applications* 30 2, pp. 389–411.

45 Lee, T.-I., Kim, M. S., and Kim, T.-S. (2016). "Contact-free thermal expansion measurement of very soft elastomers using digital image correlation". In: *Polymer Testing* 51, pp. 181–189.

46 Bolivar, J. et al. (2017). "Evaluation of multiple stress corrosion crack interactions by in-situ Digital Image Correlation". In: *Corrosion Science* 128, pp. 120–129.

47 van Rooyen, M. and Becker, T. H. (2018). "High-temperature tensile property measurements using digital image correlation over a non-uniform temperature field". In: *The Journal of Strain Analysis for Engineering Design* 53 3, pp. 117–129.

48 Take, W. A. (2015). "Thirty-Sixth Canadian Geotechnical Colloquium: Advances in visualization of geotechnical processes through digital image correlation". In: *Canadian Geotechnical Journal* 52 9, pp. 1199–1220.

49 Bartelmo, S. et al. (2016). "Digital Image Correlation Techniques for Prestressed Concrete Tie Quality Control". In: *2016 Joint Rail Conference.* American Society of Mechanical Engineers, V001T01A035.

50 Ahmadi Soleimani, S., Konstantinidis, D., and Balomenos, G. P. (2022). "Nondestructive Assessment of Elastomeric Bridge Bearings Using 3D Digital Image Correlation". In: *Journal of Structural Engineering* 148 1, p. 04021233.

51 Shao, X. et al. (2016). "Real-time 3D digital image correlation method and its application in human pulse monitoring". In: *Applied Optics* 55 4, pp. 696–704.

52 Lin, D. et al. (2018). "Detection of multipoint pulse waves and dynamic 3D pulse shape of the radial artery based on binocular vision theory". In: *Computer Methods and Programs in Biomedicine* 155, pp. 61–73.

53 Maghsoudi-Ganjeh, M. et al. (2021). "Developing a Lung Model in the Age of COVID-19: A Digital Image Correlation and Inverse Finite Element Analysis Framework". In: *Frontiers in Bioengineering and Biotechnology* 9, p. 684778.

54 Libertiaux, V., Pascon, F., and Cescotto, S. (2011). "Experimental verification of brain tissue incompressibility using digital image correlation". In: *Journal of the Mechanical Behavior of Biomedical Materials* 4 7, pp. 1177–1185.

55 Palanca, M. et al. (2018). "Full-field strain distribution in multi-vertebra spine segments: An in vitro application of digital image correlation". In: *Medical Engineering & Physics* 52, pp. 76–83.

56 Blaysat, B. et al. (2020). "Towards criteria characterizing the metrological performance of full-field measurement techniques: Application to the comparison between local and global versions of DIC". In: *Experimental Mechanics* 60 3, pp. 393–407.

57 Irie, K. et al. (2008). "A Technique for Evaluation of CCD Video-Camera Noise". In: *IEEE Transactions on Circuits and Systems for Video Technology* 18 2, pp. 280–284.

58 Reibel, Y. et al. (2003). "CCD or CMOS camera noise characterisation". In: *The European Physical Journal Applied Physics* 21 1, pp. 75–80.

59 Liu, X. Y. et al. (2020). "A self-adaptive selection of subset size method in digital image correlation based on shannon entropy". In: *IEEE Access* 8, pp. 184822–184833.

60 Pan, B. et al. (2008). "Study on subset size selection in digital image correlation for speckle patterns". In: *Optics Express* 16 10, pp. 7037–7048.

61 Wang, B. and Pan, B. (2015). "Random Errors in Digital Image Correlation Due to Matched or Overmatched Shape Functions". In: *Experimental Mechanics* 55 9, pp. 1717–1727.

62 Sadiq, R., Rodriguez, M. J., and Mian, H. R. (2019). "Empirical models to predict disinfection by-products (DBPs) in drinking water: An updated review". In: *Encyclopedia of Environmental Health*. Ed. by J. Nriagu. 2nd. Elsevier, pp. 324–338.

63 Antunes, E. et al. (2021). "Application of biochar for emerging contaminant mitigation". In: *Advances in Chemical Pollution, Environmental Management and Protection*. Vol. 7, pp. 65–91.

64 Fukushima, K. (1980). "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position". In: *Biological Cybernetics* 36 4, pp. 193–202.

65 Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017). "ImageNet classification with deep convolutional neural networks". In: *Communications of the ACM*. Ed. by F Pereira et al. Vol. 60. 6. Curran Associates, Inc., pp. 84–90.

66 Russakovsky, O. et al. (2015). "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision* 115 3, pp. 211–252.

67 Hassaballah, M. and Awad, A. I. (2020). *Deep Learning in Computer Vision*. Ed. by M. Hassaballah and A. I. Awad. CRC Press.

68 Guo, T. et al. (2017). "Simple convolutional neural network on image classification". In: *IEEE 2nd International Conference on Big Data Analysis (ICBDA)*. IEEE, pp. 721–724.

69 Sultana, F., Sufian, A., and Dutta, P. (2020). "A Review of Object Detection Models Based on Convolutional Neural Network". In: *Advances in Intelligent Systems and Computing*. Vol. 1157, pp. 1–16.

70 Guo, Y. et al. (2018). "A review of semantic segmentation using deep neural networks". In: *International Journal of Multimedia Information Retrieval* 7 2, pp. 87–93.

71 Cao, F., Yao, K., and Liang, J. (2020). "Deconvolutional neural network for image super-resolution". In: *Neural Networks* 132, pp. 394–404.

72 Liang, Y. et al. (2016). "Incorporating image priors with deep convolutional neural networks for image super-resolution". In: *Neurocomputing* 194, pp. 340–347.

73 Nazeri, K., Ng, E., and Ebrahimi, M. (2018). "Image colorization using generative adversarial networks". In: *Articulated Motion and Deformable Objects: 10th International Conference*. Vol. 10. Springer International Publishing, pp. 85–94.

74 An, J., Kpeyiton, K. G., and Shi, Q. (2020). "Grayscale images colorization with convolutional neural networks". In: *Soft Computing* 24 7, pp. 4751–4758.

75 Jing, Y. et al. (2020). "Neural Style Transfer: A Review". In: *IEEE Transactions on Visualization and Computer Graphics* 26 11, pp. 3365–3385.

76 Sangkloy, P. et al. (2017). "Scribbler: controlling deep Image synthesis with sketch and color". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. IEEE, pp. 5400–5409.

77 Schlemper, J. et al. (2018). "A Deep Cascade of Convolutional Neural Networks for Dynamic MR Image Reconstruction". In: *IEEE Transactions on Medical Imaging* 37 2, pp. 491–503.

78 Li, Q. et al. (2014). "Medical image classification with convolutional neural network". In: *2014 13th International Conference on Control Automation Robotics & Vision (ICARCV)*. IEEE, pp. 844–848.

79 Ni, J. et al. (2020). "A Survey on Theories and Applications for Self-Driving Cars Based on Deep Learning Methods". In: *Applied Sciences* 10 8, p. 2749.

80 Naranjo-Torres, J. et al. (2020). "A Review of Convolutional Neural Network Applied to Fruit Image Processing". In: *Applied Sciences* 10 10, p. 3443.

81 Strohmann, T. et al. (2021). "Automatic detection of fatigue crack paths using digital image correlation and convolutional neural networks". In: *Fatigue and Fracture of Engineering Materials and Structures* 44 5, pp. 1336–1348.

82 Wu, Z. et al. (2022). "A Metallic Fracture Estimation Method Using Digital Image Correlation". In: *Processes* 10 8, p. 1599.

83 Meruane, V. et al. (2021). "A Deep Learning Framework for Damage Assessment of Composite Sandwich Structures". In: *Shock and Vibration* 2021. Ed. by C. Sbarufatti, pp. 1–12.

84 Wang, Y. et al. (2022). “Digital image correlation (DIC) based damage detection for CFRP laminates by using machine learning based image semantic segmentation”. In: *International Journal of Mechanical Sciences* 230, p. 107529.

85 Pitter, M., See, C. W., and Somekh, M. (2001a). “Subpixel microscopic deformation analysis using correlation and artificial neural networks”. In: *Optics Express* 8 6, pp. 322–327.

86 Reed, R. A. (2010). *Comparison of Subpixel Phase Correlation Methods for Image Registration*. Tech. rep. Arnold Engineering Development Center Arnold Air Force Base, pp. 1–13.

87 Pitter, M., See, C., and Somekh, M. (2001b). “Fast subpixel digital image correlation using artificial neural networks”. In: *Proceedings 2001 International Conference on Image Processing (Cat. No.01CH37205)*. Vol. 2. IEEE, pp. 901–904.

88 Liu, X. and Tan, Q. (2010). “Subpixel in-plane displacement measurement using digital image correlation and artificial neural networks”. In: *2010 Symposium on Photonics and Optoelectronics*. IEEE, pp. 1–4.

89 Zhai, M. et al. (2020). “Optical Flow Estimation Using Dual Self-Attention Pyramid Networks”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 30 10, pp. 3663–3674.

90 Hui, T.-W., Tang, X., and Loy, C. C. (2018). “LiteFlowNet: A Lightweight Convolutional Neural Network for Optical Flow Estimation”. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 8981–8989.

91 Sun, D. et al. (2018). “PWC-Net: CNNs for Optical Flow Using Pyramid, Warping, and Cost Volume”. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 8934–8943.

92 Ma, C., Ren, Q., and Zhao, J. (2021). “Optical-numerical method based on a convolutional neural network for full-field subpixel displacement measurements”. In: *Optics Express* 29 6, p. 9137.

93 Yang, R. et al. (2022). “Deep DIC: Deep learning-based digital image correlation for end-to-end displacement and strain measurement”. In: *Journal of Materials Processing Technology* 302, p. 117474.

94 Wang, G., Zhang, L., and Yao, X. (2022). “StrainNet-3D: Real-time and robust 3-dimensional speckle image correlation using deep learning”. In: *Optics and Lasers in Engineering* 158, p. 107184.

95 Duan, X. et al. (2023). “Digital image correlation based on convolutional neural networks”. In: *Optics and Lasers in Engineering* 160, p. 107234.

96 Ma, X. et al. (2022). “Convolutional neural network based displacement gradients estimation for a full-parameter initial value guess of digital image correlation”. In: *Optics Continuum* 1 10, p. 2195.

97 Kwon, T.-H. et al. (2023). “Assessment of Speckle-Pattern Quality using Deep-Learning-Based CNN”. In: *Experimental Mechanics* 63 1, pp. 163–176.

98 Widenhorn, R. et al. (2002). “Temperature dependence of dark current in a CCD”. In: *Sensors and Camera Systems for Scientific, Industrial, and Digital Photography Applications III*. Ed. by M. M. Blouke, J. Canosa, and N. Sampat. SPIE, pp. 193–201.

99 Beenakker, C. W. J. and Patra, M. (1999). “Photon shot noise”. In: *Modern Physics Letters B* 13 11, pp. 337–347.

100 Davenport, J. J. et al. (2015). “Noise analysis for CCD-based ultraviolet and visible spectrophotometry”. In: *Applied Optics* 54 27, p. 8135.

101 Balcaen, R. et al. (2019). “Impact of motion blur on stereo-digital image correlation with the focus on a drone-carried stereo rig”. In: *Strain* 55 1, e12300.

102 Pan, B., Xie, H., and Wang, Z. (2010). “Equivalence of digital image correlation criteria for pattern matching”. In: *Applied Optics* 49 28, pp. 5501–5509.

103 Poncelet, M. and Leclerc, H. (2015). “A digital image correlation algorithm with light reflection compensation”. In: *Experimental Mechanics* 55 7, pp. 1317–1327.

104 Lava, P. et al. (2013). “Impact of lens distortions on strain measurements obtained with 2D digital image correlation”. In: *Optics and Lasers in Engineering* 51 5, pp. 576–584.

105 Balcaen, R. et al. (2017b). “Stereo-DIC Uncertainty Quantification based on Simulated Images”. In: *Experimental Mechanics* 57 6, pp. 939–951.

106 Ma, S., Pang, J., and Ma, Q. (2012). “The systematic error in digital image correlation induced by self-heating of a digital camera”. In: *Measurement Science and Technology* 23 2, p. 025403.

107 Yoneyama, S. and Ueda, H. (2012). “Bridge deflection measurement using digital image correlation with camera movement correction”. In: *Materials Transactions*. Vol. 53. 2, pp. 285–290.

108 Jones, E. and Reu, P. (2018). “Distortion of Digital Image Correlation (DIC) Displacements and Strains from Heat Waves”. In: *Experimental Mechanics* 58 7, pp. 1133–1156.

109 Sutton, M. A. et al. (2008b). “The effect of out-of-plane motion on 2D and 3D digital image correlation measurements”. In: *Optics and Lasers in Engineering* 46 10, pp. 746–757.

110 Reu, P. (2013). “Stereo-rig Design: Stereo-Angle Selection - Part 4”. In: *Experimental Techniques* 37 2, pp. 1–2.

111 Lehoucq, R. B., Reu, P. L., and Turner, D. Z. (2017). “The effect of the ill-posed problem on quantitative error assessment in digital image correlation”. In: *Experimental Mechanics* 61 3, pp. 609–621.

112 Wagne, B., Roux, S., and Hild, F. (2002). “Spectral approach to displacement evaluation from image analysis”. In: *The European Physical Journal Applied Physics* 17 3, pp. 247–252.

113 Wang, Z. Y. et al. (2007). “Statistical Analysis of the Effect of Intensity Pattern Noise on the Displacement Measurement Precision of Digital Image Correlation Using Self-correlated Images”. In: *Experimental Mechanics* 47 5, pp. 701–707.

114 Zhou, Y. et al. (2015). “Image pre-filtering for measurement error reduction in digital image correlation”. In: *Optics and Lasers in Engineering* 65, pp. 46–56.

115 Schreier, H. W. and Sutton, M. A. (2002). “Systematic errors in digital image correlation due to undermatched subset shape functions”. In: *Experimental Mechanics* 42 3, pp. 303–310.

116 Xu, X., Su, Y., and Zhang, Q. (2017). “Theoretical estimation of systematic errors in local deformation measurements using digital image correlation”. In: *Optics and Lasers in Engineering* 88, pp. 265–279.

117 Wittevrongel, L. et al. (2014). “A Self Adaptive Global Digital Image Correlation Algorithm”. In: *Experimental Mechanics* April 2014, pp. 361–378.

118 Fayad, S. S., Seidl, D. T., and Reu, P. L. (2020). “Spatial DIC errors due to pattern-induced bias and grey level discretization”. In: *Experimental Mechanics* 60 2, pp. 249–263.

119 Sur, F., Blaysat, B., and Grédiac, M. (2022). “Which pattern for a low pattern-induced bias?” In: *Conference Proceedings of the Society for Experimental Mechanics Series*, pp. 103–105.

120 Sur, F., Blaysat, B., and Grédiac, M. (2021). “On biases in displacement estimation for image registration, with a focus on photomechanics”. In: *Journal of Mathematical Imaging and Vision* 63 7, pp. 777–806.

121 Sutton, M. A., Orteu, J.-J., and Schreier, H. (2009). *Image correlation for shape, motion and deformation measurements*. Boston, MA: Springer US, pp. 1–321.

122 Reu, P. L. et al. (2018b). “DIC challenge: Developing images and guidelines for evaluating accuracy and resolution of 2D analyses”. In: *Experimental Mechanics* 58 7, pp. 1067–1099.

123 Reu, P. et al. (2018a). *Stereo DIC challenge dataset: DIC challenge web page*.

124 Reu, P. L. et al. (2022). “DIC Challenge 2.0: Developing Images and Guidelines for Evaluating Accuracy and Resolution of 2D Analyses: Focus on the Metrological Efficiency Indicator”. In: *Experimental Mechanics* 62 4, pp. 639–654.

125 McCulloch, W. S. and Pitts, W. (1943). “A logical calculus of the ideas immanent in nervous activity”. In: *The Bulletin of Mathematical Biophysics* 5 4, pp. 115–133.

126 Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). “Learning representations by back-propagating errors”. In: *Nature* 323 6088, pp. 533–536.

127 Hebb, D. O. (1949). *The organization of behavior; a neuropsychological theory.* Oxford, England: Wiley, pp. xix, 335–xix, 335.

128 Rosenblatt, F. (1958). “The perceptron: A probabilistic model for information storage and organization in the brain.” In: *Psychological Review* 65 6, pp. 386–408.

129 Gurney, K. (2018). *An Introduction to Neural Networks*. CRC Press.

130 Glorot, X. and Bengio, Y. (2010). “Understanding the difficulty of training deep feedforward neural networks”. In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256.

131 Bengio, Y. (2009). “Learning Deep Architectures for AI”. In: *Foundations and Trends® in Machine Learning* 2 1, pp. 1–127.

132 Maas, A. L., Hannun, A. Y., and Ng, A. Y. (2013). “Rectifier nonlinearities improve neural network acoustic models”. In: *Proceedings of the 30th International Conference on Machine Learning*, p. 3.

133 Yamashita, R. et al. (2018). “Convolutional neural networks: an overview and application in radiology”. In: *Insights into Imaging* 9 4, pp. 611–629.

134 Albawi, S., Mohammed, T. A., and Al-Zawi, S. (2017). “Understanding of a convolutional neural network”. In: *2017 International Conference on Engineering and Technology (ICET)*. IEEE, pp. 1–6.

135 Milosevic, N. et al. (2020). *Introduction to convolutional neural networks: with image classification using PyTorch*. Berkeley, CA: Apress.

136 Cybenko, G. (1989). “Approximation by superpositions of a sigmoidal function”. In: *Mathematics of Control, Signals, and Systems* 2 4, pp. 303–314.

137 Ba, L. J. and Caruana, R. (2014). “Do Deep Nets Really Need to be Deep?” In: *Advances in Neural Information Processing Systems*. Vol. 27. Curran Associates, Inc.

138 Bebis, G. and Georgiopoulos, M. (1994). “Feed-forward neural networks”. In: *IEEE Potentials* 13 4, pp. 27–31.

139 Oyedotun, O. K., Ismaeil, K. A., and Aouada, D. (2022). “Why Is Everyone Training Very Deep Neural Network With Skip Connections?” In: *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15.

140 He, K. et al. (2015a). “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification”. In: *2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE, pp. 1026–1034.

141 Srivastava, R. K., Greff, K., and Schmidhuber, J. (2015). “Training very deep networks”. In: *Advances in Neural Information Processing Systems*. Vol. 28, pp. 1–9.

142 Benardos, P. and Vosniakos, G.-C. (2007). “Optimizing feedforward artificial neural network architecture”. In: *Engineering Applications of Artificial Intelligence* 20 3, pp. 365–382.

143 Ma, L. and Khorasani, K. (2003). “A new strategy for adaptively constructing multilayer feedforward neural networks”. In: *Neurocomputing* 51, pp. 361–385.

144 Fahlman, S. E. and Lebiere, C (1990). “The cascade-correlation learning architecture”. In: *Advances in Neural Information Processing*. Vol. 2.

145 Bebis, G., Georgiopoulos, M., and Kasparis, T. (1997). “Coupling weight elimination with genetic algorithms to reduce network size and preserve generalization”. In: *Neurocomputing* 17 3-4, pp. 167–194.

146 Benardos, P. and Vosniakos, G. (2002). “Prediction of surface roughness in CNC face milling using neural networks and Taguchi’s design of experiments”. In: *Robotics and Computer-Integrated Manufacturing* 18 5-6, pp. 343–354.

147 Ross, P. J. (1996). *Taguchi Techniques for Quality Engineering: Loss Function, Orthogonal Experiments, Parameter and Tolerance Design*. McGraw-Hill.

148 Zadeh, L. (1965). “Fuzzy sets”. In: *Information and Control* 8 3, pp. 338–353.

149 Leski, J. and Czogała, E. (1999). “A new artificial neural network based fuzzy inference system with moving consequents in if–then rules and selected applications”. In: *Fuzzy Sets and Systems* 108 3, pp. 289–297.

150 Mohammad, R. M., Thabtah, F., and McCluskey, L. (2016). “An Improved Self-Structuring Neural Network”. In: *Trends and Applications in Knowledge Discovery and Data Mining: PAKDD 2016 Workshops*. Springer, pp. 35–47.

151 Jesus, R. J. et al. (2021). “Effect of Initial Configuration of Weights on Training and Function of Artificial Neural Networks”. In: *Mathematics* 9 18, p. 2246.

152 Goodfellow, I., Benigo, Y., and Courville Aaron (2016). *Deep Learning (Adaptive Computation and Machine Learning series)*. The MIT Press.

153 Sutton, R. S. (1986). “Two problems with backpropagation and other steepest-descent learning procedures for networks”. In: *Proc. 8th annual conf. cognitive science society*, pp. 823–832.

154 Du, S. S. et al. (2017). "Gradient Descent Can Take Exponential Time to Escape Saddle Points". In: *Advances in Neural Information Processing Systems*. Ed. by I Guyon et al. Vol. 30. Curran Associates, Inc.

155 Dauphin, Y. N. et al. (2014). "Identifying and attacking the saddle point problem in high-dimensional non-convex optimization". In: *Advances in Neural Information Processing Systems*. Vol. 4, pp. 1–14.

156 Qian, N. (1999). "On the momentum term in gradient descent learning algorithms". In: *Neural Networks* 12 1, pp. 145–151.

157 Kingma, D. P. and Ba, J. L. (2015). "Adam: A method for stochastic optimization". In: *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.

158 Ruder, S. (2016). "An overview of gradient descent optimization algorithms". In: *arXiv preprint arXiv:1609.04747*.

159 Masters, D. and Luschi, C. (2018). "Revisiting Small Batch Training for Deep Neural Networks". In: *CoRR* abs/1804.0.

160 Li, M., Soltanolkotabi, M., and Oymak, S. (2020). "Gradient Descent with Early Stopping is Provably Robust to Label Noise for Overparameterized Neural Networks". In: *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics* 108, pp. 4313–4324.

161 Nargesian, F. et al. (2017). "Learning Feature Engineering for Classification". In: *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*. Vol. 0. California: International Joint Conferences on Artificial Intelligence Organization, pp. 2529–2535.

162 Bengio, Y., Courville, A., and Vincent, P. (2013). "Representation Learning: A Review and New Perspectives". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35 8, pp. 1798–1828.

163 Raphael, A. et al. (2020). "Neural Network Recognition of Marine Benthos and Corals". In: *Diversity* 12 1, p. 29.

164 Hinton, G. E., Osindero, S., and Teh, Y.-W. (2006). "A Fast Learning Algorithm for Deep Belief Nets". In: *Neural Computation* 18 7, pp. 1527–1554.

165 Sutton, M. et al. (1986). "Application of an optimized digital correlation method to planar deformation analysis". In: *Image and Vision Computing* 4 3, pp. 143–150.

166 Blaber, J., Adair, B., and Antoniou, A. (2015). "Ncorr: open-source 2D digital image correlation matlab software". In: *Experimental Mechanics* 55 6, pp. 1105–1122.

167 Turner, D. (2015). *Digital image correlation engine (DICe) reference manual*.

168 Turner, D. Z. (2016). "An overview of the gradient-based local DIC formulation for motion estimation in DICe". In: *Sandia Report, SAND2016-7360 R* Dic, pp. 1–6.

169 Seghir, R., Witz, J., and Coudert, S. (2014). *YaDICs - Digital Image Correlation 2/3D software*.

170 Andre, D. (2020). *PyDIC*.

171 van Rooyen, M. (2020). "Thermal power plant steel creep deformation measurement using digital image correlation". Doctoral Dissertation. Stellenbosch University, pp. 1–185.

172 Molteno, M. (2017). "Measuring fracture properties using digital image and volume correlation: decomposing the J-integral for mixed-mode parameters". PhD thesis, pp. 1–168.

173 Huchzermeyer, R. (2017). "Measuring mechanical properties using digital image correlation: extracting tensile and fracture properties from a single sample". PhD thesis. Master's thesis, Stellenbosch University, Stellenbosch.

174 Hassan, G. M. (2021). "Deformation measurement in the presence of discontinuities with digital image correlation: A review". In: *Optics and Lasers in Engineering* 137, p. 106394.

175 Rafiq, M., Bugmann, G, and Easterbrook, D. (2001). "Neural network design for engineering applications". In: *Computers & Structures* 79 17, pp. 1541–1552.

176 Hannan, S. A., Manza, R. R., and Ramteke, R. J. (2010). "Generalized Regression Neural Network and Radial Basis Function for Heart Disease Diagnosis". In: *International Journal of Computer Applications* 7 13, pp. 7–13.

177 Kumar, R., Aggarwal, R., and Sharma, J. (2015). "Comparison of regression and artificial neural network models for estimation of global solar radiations". In: *Renewable and Sustainable Energy Reviews* 52, pp. 1294–1299.

178 Su, Y. et al. (2016). "Quality assessment of speckle patterns for DIC by consideration of both systematic errors and random errors". In: *Optics and Lasers in Engineering* 86, pp. 132–142.

179 Beeck, J. van et al. (2014). "Quantification of Three-Dimensional Surface Deformation using Global Digital Image Correlation". In: *Experimental Mechanics* 54 4, pp. 557–570.

180 Yang, J. and Bhattacharya, K. (2019a). "Augmented Lagrangian Digital Image Correlation". In: *Experimental Mechanics* 59 2, pp. 187–205.

181 Curt, J. et al. (2020). "Optimal digital color image correlation". In: *Optics and Lasers in Engineering* 127, p. 105896.

182 Pan, B. (2009). "Reliability-guided digital image correlation for image deformation measurement". In: *Applied Optics* 48 8, pp. 1535–1542.

183 Huang, J. et al. (2013). "Digital image correlation with self-adaptive gaussian windows". In: *Experimental Mechanics* 53 3, pp. 505–512.

184 Bornert, M. et al. (2009). "Assessment of digital image correlation measurement errors: Methodology and results". In: *Experimental Mechanics* 49 3, pp. 353–370.

185 Reu, P. L. (2016). "A Realistic Error Budget for Two Dimension Digital Image Correlation". In: *Conference Proceedings of the Society for Experimental Mechanics Series*. Vol. 3, pp. 189–193.

186 Dong, Y. and Pan, B. (2017a). "A review of speckle pattern fabrication and assessment for digital image correlation". In: *Experimental Mechanics* 57 8, pp. 1161–1181.

187 Bickel, V. T., Manconi, A., and Amann, F. (2018). "Quantitative assessment of digital image correlation methods to detect and monitor surface displacements of large slope instabilities". In: *Remote Sensing* 10 6, p. 865.

188 Aydin, A., Eker, R., and Fuchs, H. (2017). "Lidar Data Analysis With Digital Image Correlation (DIC) In Obtaining Landslide Displacement Fields: A

Case Of Gschliefgraben Landslide-Austria". In: *The Online Journal of Science and Technology* 7 4, pp. 121–127.

189 Dong, Y. and Pan, B. (2017b). "A review of speckle pattern fabrication and assessment for digital image correlation". In: *Experimental Mechanics* 57 8, pp. 1161–1181.

190 Lionello, G. and Cristofolini, L. (2014). "A practical approach to optimizing the preparation of speckle patterns for digital-image correlation". In: *Measurement Science and Technology* 25 10, p. 107001.

191 Atkinson, D. and Becker, T. (2020). "A 117 line 2D digital image correlation code written in matlab". In: *Remote Sensing* 12 18, p. 2906.

192 Pan, B. et al. (2009b). "Two-dimensional digital image correlation for in-plane displacement and strain measurement: a review". In: *Measurement Science and Technology* 20 6, p. 062001.

193 Tuononen, A. J. (2014). "Digital Image Correlation to analyse stick-slip behaviour of tyre tread block". In: *Tribology International* 69, pp. 70–76.

194 Sutton, M. A. et al. (2008a). "Strain field measurements on mouse carotid arteries using microscopic three-dimensional digital image correlation". In: *Journal of Biomedical Materials Research Part A: An Official Journal of The Society for Biomaterials, The Japanese Society for Biomaterials, and The Australian Society for Biomaterials and the Korean Society for Biomaterials* 84 1, pp. 178–190.

195 Palanca, M., Tozzi, G., and Cristofolini, L. (2016). "The use of digital image correlation in the biomechanical area: A review". In: *International Biomechanics* 3 1, pp. 1–21.

196 Zhang, D. and Arola, D. D. (2004). "Applications of digital image correlation to biological tissues". In: *Journal of Biomedical Optics* 9 4, pp. 691–700.

197 Winkler, J. and Hendy, C. (2017). "Improved structural health monitoring of London's docklands light railway bridges using digital image correlation". In: *Structural Engineering International* 27 3, pp. 435–440.

198 Reagan, D., Sabato, A., and Niezrecki, C. (2018). "Feasibility of using digital image correlation for unmanned aerial vehicle structural health monitoring of bridges". In: *Structural Health Monitoring* 17 5, pp. 1056–1072.

199 LeBlanc, B. et al. (2013). "Damage detection and full surface characterization of a wind turbine blade using three-dimensional digital image correlation". In: *Structural Health Monitoring* 12 5-6, pp. 430–439.

200 Walter, T. R. et al. (2013). "Volcanic eruption monitoring by thermal image correlation: Pixel offsets show episodic dome growth of the Colima volcano". In: *Journal of Geophysical Research: Solid Earth* 118 4, pp. 1408–1419.

201 Leprince, S. et al. (2007). "Automatic and precise orthorectification, coregistration, and subpixel correlation of satellite images, application to ground deformation measurements". In: *IEEE Transactions on Geoscience and Remote Sensing* 45 6, pp. 1529–1558.

202 Heid, T. and Kääb, A. (2012). "Evaluation of existing image matching methods for deriving glacier surface displacements globally from optical satellite imagery". In: *Remote Sensing of Environment* 118, pp. 339–355.

203 Wang, P., Pierron, F., and Thomsen, O. T. (2013). "Identification of material parameters of PVC foams using digital image correlation and the virtual fields method". In: *Experimental Mechanics* 53 6, pp. 1001–1015.

204 Grédiac, M., Pierron, F., and Surrel, Y. (1999). "Novel procedure for complete in-plane composite characterization using a single T-shaped specimen". In: *Experimental Mechanics* 39 2, pp. 142–149.

205 Leclerc, H. et al. (2009). "Integrated digital image correlation for the identification of mechanical properties". In: *International Conference on Computer Vision/Computer Graphics Collaboration Techniques and Applications*. Ed. by A. Gagalowic and W. Philips. Heidelberg, Berlin: Springer, pp. 161–171.

206 Chevalier, L. et al. (2001). "Digital image correlation used to analyze the multiaxial behavior of rubber-like materials". In: *European Journal of Mechanics - A/Solids* 20 2, pp. 169–187.

207 Peters, W. H. and Ranson, W. F. (1982). "Digital imaging techniques in experimental stress analysis". In: *Optical Engineering* 21 3, pp. 427–431.

208 Luo, P. F. et al. (1993). "Accurate measurement of three-dimensional deformations in deformable and rigid bodies using computer vision". In: *Experimental Mechanics* 33 2, pp. 123–132.

209 Pan, B., Li, K., and Tong, W. (2013). "Fast, robust and accurate digital image correlation calculation without redundant computations". In: *Experimental Mechanics* 53 7, pp. 1277–1289.

210 Gao, Y. et al. (2015). "High-efficiency and high-accuracy digital image correlation for three-dimensional measurement". In: *Optics and Lasers in Engineering* 65, pp. 73–80.

211 Pan, B. and Wang, B. (2016). "Digital image correlation with enhanced accuracy and efficiency: a comparison of two subpixel registration algorithms". In: *Experimental Mechanics* 56 8, pp. 1395–1409.

212 Bloomenthal, J. and Rokne, J. (1994). "Homogeneous coordinates". In: *The Visual Computer* 11 1, pp. 15–26.

213 Zhang, Z. (2000). "A flexible new technique for camera calibration". In: *IEEE Transactions on pattern analysis and machine intelligence* 22 11, pp. 1330–1334.

214 Heikkila, J. and Silven, O. (1997). "Four-step camera calibration procedure with implicit image correction". In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Ed. by D. Plummer and I. Tonvick. Washington, D.C., United States of America: IEEE Computer Society, pp. 1106–1112.

215 Tsai, R. Y. (1987). "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses". In: *IEEE Journal on Robotics and Automation* 3 4, pp. 323–344.

216 Wei, G. Q. and De Ma, S. (1994). "Implicit and explicit camera calibration: theory and experiments". In: *IEEE Transactions on pattern analysis and machine intelligence* 16 5, pp. 469–480.

217 Keys, R. G. (1981). "Cubic convolution interpolation for digital image processing". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 29 6, pp. 1153–1160.

218 Hou, H. S. and Andrews, H. C. (1978). "Cubic splines for image interpolation and digital filtering". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 26 6, pp. 508–517.

219 Pan, Z. et al. (2016). "Performance of global look-up table strategy in digital image correlation with cubic B-spline interpolation and bicubic interpolation". In: *Theoretical and Applied Mechanics Letters* 6 3, pp. 126–130.

220 Żołądek, H. (2000). "The topological proof of abel-ruffini theorem". In: *Topological Methods in Nonlinear Analysis* 16 2, pp. 253–265.

221 Pan, B. et al. (2009a). "Digital image correlation using iterative least squares and pointwise least squares for displacement field and strain field measurements". In: *Optics and Lasers in Engineering* 47 7-8, pp. 865–874.

222 Zhou, Y., Sun, C., and Chen, J. (2014). "Adaptive subset offset for systematic error reduction in incremental digital image correlation". In: *Optics and Lasers in Engineering* 55, pp. 5–11.

223 Foroosh, H., Zerubia, J. B., and Berthod, M. (2002). "Extension of phase correlation to subpixel registration". In: *IEEE Transactions on Image Processing* 11 3, pp. 188–200.

224 Orteu, J.-J. et al. (2006). "A speckle texture image generator". In: *Speckle06: Speckles, From Grains to Flowers*. Ed. by P. Slangen and C. Cerruti. SPIE, pp. 104–109.

225 Reu, P. L. (2011). "Experimental and numerical methods for exact subpixel shifting". In: *Experimental Mechanics* 51 4, pp. 443–452.

226 Harilal, R. and Ramji, M. (2014). "Adaptation of open source 2D DIC software ncorr for solid mechanics applications". In: *9th International Symposium on Advanced Science and Technology in Experimental Mechanics*. New Delhi, India.

227 Lunt, D. et al. (2020). "Comparison of sub-grain scale digital image correlation calculated using commercial and open-source software packages". In: *Materials Characterization* 163, p. 110271.

228 Atkinson, D. and Becker, T. H. (2021). "Stereo digital image correlation in matlab". In: *Applied Sciences* 11 11, p. 4904.

229 Kammers, A. D. and Daly, S. (2013). "Self-assembled nanoparticle surface patterning for improved digital image correlation in a scanning electron microscope". In: *Experimental Mechanics* 53 8, pp. 1333–1341.

230 Jin, H., Lu, W. Y., and Korellis, J. (2008). "Micro-scale deformation measurement using the digital image correlation technique and scanning electron microscope imaging". In: *Journal of Strain Analysis for Engineering Design* 43 8, pp. 719–728.

231 Rosu, A. M. et al. (2015). "Measurement of ground displacement from optical satellite image correlation using the free open-source software micmac". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 100, pp. 48–59.

232 Genovese, K. et al. (2013). "An improved panoramic digital image correlation method for vascular strain analysis and material characterization". In: *Journal of the Mechanical Behavior of Biomedical Materials* 27, pp. 132–142.

233 Masc, C. M. et al. (2015). "Field monitoring of a bridge using digital image correlation". In: *Proceedings of the Institution of Civil Engineers: Bridge Engineering*. Vol. 168. 1. Thomas Telford Services Ltd, pp. 3–12.

234 Yoneyama, S. et al. (2007). "Bridge deflection measurement using digital image correlation". In: *Experimental Techniques* 31 1, pp. 34–40.

235 Winkler, J., Specialist, C., and Hansen, M. D. (2018). "Innovative long-term monitoring of the great belt bridge expansion joint using digital image correlation". In: *Structural Engineering International* 28 3, pp. 347–352.

236 Sousa, P. J. et al. (2019). "Experimental measurement of bridge deflection using digital image correlation". In: *Procedia Structural Integrity*. Vol. 17. Elsevier, pp. 806–811.

237 Ngeljaratan, L. and Moustafa, M. A. (2020). "Structural health monitoring and seismic response assessment of bridge structures using target-tracking digital image correlation". In: *Engineering Structures* 213, p. 110551.

238 Tian, L. and Pan, B. (2016). "Remote bridge deflection measurement using an advanced video deflectometer and actively illuminated LED targets". In: *Sensors (Switzerland)* 16 9, p. 1344.

239 Tung, S. H., Shih, M. H., and Sung, W. P. (2008). "Development of digital image correlation method to analyse crack variations of masonry wall". In: *Sadhana - Academy Proceedings in Engineering Sciences* 33 6, pp. 767–779.

240 Nghiem, H. L., Al Heib, M., and Emeriault, F. (2015). "Method based on digital image correlation for damage assessment in masonry structures". In: *Engineering Structures* 86, pp. 1–15.

241 Ghorbani, R., Matta, F., and Sutton, M. A. (2015). "Full-field deformation measurement and crack mapping on confined masonry walls using digital image correlation". In: *Experimental Mechanics* 55 1, pp. 227–243.

242 Mazzanti, P., Caporossi, P., and Muzi, R. (2020). "Sliding time master digital image correlation analyses of cubesat images for landslide monitoring: The Rattlesnake Hills landslide (USA)". In: *Remote Sensing* 12 4, p. 592.

243 Caporossi, P., Mazzanti, P., and Bozzano, F. (2018). "Digital image correlation (DIC) analysis of the 3 December 2013 Montescaglioso landslide (Basilicata, southern Italy): Results from a multi-dataset investigation". In: *ISPRS International Journal of Geo-Information* 7 9, p. 372.

244 Daehne, A. and Corsini, A. (2013). "Kinematics of active earthflows revealed by digital image correlation and DEM subtraction techniques applied to multi-temporal LiDAR data". In: *Earth Surface Processes and Landforms* 38 6, pp. 640–654.

245 Dridi, S., Morestin, F., and Dogui, A. (2012). "Use of digital image correlation to analyse the shearing deformation in woven fabric". In: *Experimental Techniques* 36 5, pp. 46–52.

246 Hursa, A., Rolich, T., and Ražić, S. E. (2009). "Determining pseudo Poisson's ratio of woven fabric with a digital image correlation method". In: *Textile Research Journal* 79 17, pp. 1588–1598.

247 Helfrick, M. N. et al. (2011). "3D digital image correlation methods for full-field vibration measurement". In: *Mechanical Systems and Signal Processing* 25 3, pp. 917–927.

248 Yu, L. and Pan, B. (2017). "Single-camera high-speed stereo-digital image correlation for full-field vibration measurement". In: *Mechanical Systems and Signal Processing* 94, pp. 374–383.

249 Beberniss, T. J. and Ehrhardt, D. A. (2017). "High-speed 3D digital image correlation vibration measurement: Recent advancements and noted limitations". In: *Mechanical Systems and Signal Processing* 86, pp. 35–48.

250 Rajaram, S. et al. (2017). "Full-field deformation measurements during seismic loading of masonry buildings". In: *Structural Control and Health Monitoring* 24 4, e1903.

251 Solav, D. et al. (2018). "MultiDIC: An Open-Source Toolbox for Multi-View 3D Digital Image Correlation". In: *IEEE Access* 6, pp. 30520–30535.

252 Balcaen, R. et al. (2017a). "Stereo-DIC Calibration and Speckle Image Generator Based on FE Formulations". In: *Experimental Mechanics* 57 5, pp. 703–718.

253 Hartley, R. and Zisserman, A. (2004). *Multiple View Geometry in Computer Vision*. 2nd. Cambridge University press, pp. 310–324.

254 Hartley, R. I. and Sturm, P. (1997). "Triangulation". In: *Computer Vision and Image Understanding* 68 2, pp. 146–157.

255 Garcia, D., Orteu, J. J., and Penazzi, L. (2002). "A combined temporal tracking and stereo-correlation technique for accurate measurement of 3D displacements: Application to sheet metal forming". In: *Journal of Materials Processing Technology* 125, pp. 736–742.

256 Zhou, Y., Pan, B., and Chen, Y. Q. (2012). "Large deformation measurement using digital image correlation: a fully automated approach". In: *Applied Optics* 51 31, pp. 6674–7683.

257 Lowe, D. G. (2004). "Distinctive image features from scale-invariant keypoints". In: *International Journal of Computer Vision* 60 2, pp. 91–110.

258 Torr, P. H. and Zisserman, A. (2000). "MLESAC: A new robust estimator with application to estimating image geometry". In: *Computer Vision and Image Understanding* 78 1, pp. 138–156.

259 Lourakis, M. (2018). *Stereo triangulation.*

260 Grubbs, F. E. (1969). "Procedures for detecting outlying observations in samples". In: *Technometrics* 11 1, pp. 1–21.

261 Furukawa, Y. and Ponce, J. (2009). "Accurate camera calibration from multi-view stereo and bundle adjustment". In: *International Journal of Computer Vision* 84 3, pp. 257–268.

262 Vo, M. (2011). "Advanced geometric camera calibration for machine vision". In: *Optical Engineering* 50 11, p. 110503.

263 Pan, B., Dafang, W., and Yong, X. (2012). "Incremental calculation for large deformation measurement using reliability-guided digital image correlation". In: *Optics and Lasers in Engineering* 50 4, pp. 586–592.

264 Thiruselvam, N. I. and Subramanian, S. J. (2019). "Feature-assisted stereo correlation". In: *Strain* 55 5, e12315.

265 Weinzaepfel, P. et al. (2013). "DeepFlow: Large displacement optical flow with deep matching". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1385–1392.

266 Datta, A., Kim, J.-S., and Kanade, T. (2009). "Accurate camera calibration using iterative refinement of control points". In: *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*. IEEE, pp. 1201–1208.

267 Zhu, Z. et al. (2019). "Camera calibration method based on optimal polarization angle". In: *Optics and Lasers in Engineering* 112, pp. 128–135.

268 Chen, B. and Pan, B. (2020). "Camera calibration using synthetic random speckle pattern and digital image correlation". In: *Optics and Lasers in Engineering* 126, p. 105919.

269 Stein, G. (1997). "Lens distortion calibration using point correspondences". In: *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE Comput. Soc, pp. 602–608.

270 Claus, D. and Fitzgibbon, A. (2005). "A Rational Function Lens Distortion Model for General Cameras". In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. Vol. 1. IEEE, pp. 213–219.

271 Jin, D. and Yang, Y. (2019). "Using distortion correction to improve the precision of camera calibration". In: *Optical Review* 26 2, pp. 269–277.

272 Rosebrock, D. and Wahl, F. M. (2012). "Complete Generic Camera Calibration and Modeling Using Spline Surfaces". In: *11th Asian Conference on Computer Vision (ACCV 2012)*. Daejeon, Korea: Springer, pp. 487–498.

273 Reh, T. et al. (2014). "Improving the Generic Camera Calibration technique by an extended model of calibration display". In: *Journal of the European Optical Society: Rapid Publications* 9, p. 14044.

274 Hartmann, C. et al. (2018). "Implementation and evaluation of optical flow methods for two-dimensional deformation measurement in comparison to digital image correlation". In: *Optics and Lasers in Engineering* 107, pp. 127–141.

275 Cunha, F. G., Santos, T. G., and Xavier, J. (2021). "In situ monitoring of additive manufacturing using digital image correlation: A review". In: *Materials* 14 6, p. 1511.

276 Tong, J. (2018). "Full-field characterisation of crack tip deformation and fatigue crack growth using digital image correlation—a review". In: *Fatigue and Fracture of Engineering Materials and Structures* 41 9, pp. 1855–1869.

277 Yoon, S. et al. (2021). "Digital image correlation in dental materials and related research: A review". In: *Dental Materials* 37 5, pp. 758–771.

278 Wang, Y. Q. et al. (2009b). "Quantitative error assessment in pattern matching: Effects of intensity pattern noise, interpolation, strain and image contrast on motion measurements". In: *Strain* 45 2, pp. 160–178.

279 Hua, T. et al. (2011). "Evaluation of the quality of a speckle pattern in the digital image correlation method by mean subset fluctuation". In: *Optics and Laser Technology* 43 1, pp. 9–13.

280 Pan, B., Lu, Z., and Xie, H. (2010). "Mean intensity gradient: An effective global parameter for quality assessment of the speckle patterns used in digital image correlation". In: *Optics and Lasers in Engineering* 48 4, pp. 469–477.

281 Lane, C., Burguete, R. L., and Shterenlikht, A. (2008). "An objective criterion for the selection of an optimum DIC pattern and subset size". In: *Proceedings of the SEM XIth international congress and exposition on experimental and applied mechanics*. Vol. 2, pp. 1–9.

282 Wang, M. et al. (2009a). "A weighting window applied to the digital image correlation method". In: *Optics and Laser Technology* 41 2, pp. 154–158.

283 Yuan, Y. et al. (2014). "Accurate displacement measurement via a self-adaptive digital image correlation method based on a weighted ZNSSD criterion". In: *Optics and Lasers in Engineering* 52 1, pp. 75–85.

284 Kang, M. S. and An, Y. K. (2020). "Adaptive subset-based digital image correlation for fatigue crack evaluation". In: *Applied Sciences* 10 10, p. 3574.

285 Słoński, M. and Tekieli, M. (2020). "2D digital image correlation and region-based convolutional neural network in monitoring and evaluation of surface cracks in concrete structural elements". In: *Materials* 13 16, p. 3527.

286 Sur, F., Blaysat, B., and Grédiac, M. (2018). "Rendering deformed speckle images with a boolean model". In: *Journal of Mathematical Imaging and Vision* 60 5, pp. 634–650.

287 Pan, B. et al. (2006). "Performance of sub-pixel registration algorithms in digital image correlation". In: *Measurement Science and Technology* 17 6, p. 1615.

288 Otsu, N. (1979). "A threshold selection method from gray-level histograms". In: *IEEE Transactions on Systems, Man, and Cybernetics* 9 1, pp. 62–66.

289 Ye, X. and Zhao, J. (2022). "Adaptive rotated gaussian weighted digital image correlation (RGW-DIC) for heterogeneous deformation measurement". In: *Experimental Mechanics* 62 2, pp. 271–286.

290 Yang, J. and Bhattacharya, K. (2019b). "Fast Adaptive Global Digital Image Correlation". In: *Conference Proceedings of the Society for Experimental Mechanics Series*. Vol. 12, pp. 69–73.

291 Khamparia, A. and Singh, K. M. (2019). "A systematic review on deep learning architectures and applications". In: *Expert Systems* 36 3, e12400.

292 Basheer, I. and Hajmeer, M (2000). "Artificial neural networks: Fundamentals, computing, design, and application". In: *Journal of Microbiological Methods* 43 1, pp. 3–31.

293 Reu, P. (2014). "All about speckles: Speckle size measurement". In: *Experimental Techniques* 38 6, pp. 1–2.

294 Shao, X., Dai, X., and He, X. (2015). "Noise robustness and parallel computation of the inverse compositional gauss–newton algorithm in digital image correlation". In: *Optics and Lasers in Engineering* 71, pp. 9–19.

295 Nocedal, J and Wright, S (2006). *Numerical optimization*. New York: Springer.

296 Immerkaer, J. (1996). "Fast noise variance estimation". In: *Computer Vision and Image Understanding* 64 2, pp. 300–302.

297 Zhuang, F. et al. (2021). "A comprehensive survey on transfer learning". In: *Proceedings of the IEEE* 109 1, pp. 43–76.

298 Hassan, G. M. et al. (2016a). "A comparative study of techniques of distant reconstruction of displacement and strain fields using the DISTRESS simulator". In: *Optik* 127 23, pp. 11288–11305.

299 Canny, J. (1986). "A computational approach to edge detection". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-8 6, pp. 679–698.

300 Park, J. et al. (2017). "Assessment of speckle-pattern quality in digital image correlation based on gray intensity and speckle morphology". In: *Optics and Lasers in Engineering* 91, pp. 62–72.

301 Poissant, J. and Barthelat, F. (2010). "A Novel "Subset Splitting" Procedure for Digital Image Correlation on Discontinuous Displacement Fields". In: *Experimental Mechanics* 50 3, pp. 353–364.

302 Hassan, G. M. (2019). "Digital Image Correlation for discontinuous displacement measurement using subset segmentation". In: *Optics and Lasers in Engineering* 115, pp. 208–216.

303 Ramos, T. et al. (2015). "Comparison Between Finite Element Method Simulation, Digital Image Correlation and Strain Gauges Measurements in a 3-Point Bending Flexural Test". In: *Procedia Engineering* 114, pp. 232–239.

304 Wu, R., Qian, H., and Zhang, D. (2016). "Robust full-field measurement considering rotation using digital image correlation". In: *Measurement Science and Technology* 27 10, p. 105002.

305 de Deus Filho, J. C. A., da Silva Nunes, L. C., and Xavier, J. M. C. (2022). "iCorrVision-2D: An integrated python-based open-source Digital Image Correlation software for in-plane measurements (Part 1)". In: *SoftwareX* 19, p. 101131.

306 Guerriero, L. et al. (2020). "Digital Image Correlation of Google Earth Images for Earth's Surface Displacement Estimation". In: *Remote Sensing* 12 21, p. 3518.

307 Malesa, M. et al. (2015). "Application of Multi-camera DIC System for Measurements of Industrial Structures". In: *Procedia Engineering* 114, pp. 453–460.

308 Dai, Y. and Li, H. (2022). "Multi-Camera Digital Image Correlation in Deformation Measurement of Civil Components with Large Slenderness Ratio and Large Curvature". In: *Materials* 15 18, p. 6281.

309 Li, H. et al. (2022). "Application of multi-camera digital image correlation in the stability study of the long timber column with the circular cross-section under axial compression". In: *BioResources* 17 1, pp. 1717–1728.

310 Chovan, C. P. et al. (2013). *Improving Material Property Measurement by Using Multi-Camera Digital Image Correlation*. Tech. rep. SAE Technical Paper.

311 Alexander, T. L., Harvey, J. E., and Weeks, A. R. (1994). "Average speckle size as a function of intensity threshold level: comparison of experimental measurements with theory". In: *Applied Optics* 33 35, p. 8240.

312 Pan, B. et al. (7375). "On errors of digital image correlation due to speckle patterns". In: *ICEM 2008: International Conference on Experimental Mechanics*. Ed. by X. He, H. Xie, and Y. Kang. SPIE, pp. 1157–1163.

313 Gyori, N. G. et al. (2022). "Training data distribution significantly impacts the estimation of tissue microstructure with machine learning". In: *Magnetic Resonance in Medicine* 87 2, pp. 932–947.

314 Zhang, D., Luo, M., and Arola, D. (2006). "Displacement/strain measurements using an optical microscope and digital image correlation". In: *Optical Engineering* 45 3, p. 033605.

315 Yu, L. and Pan, B. (2015). "The errors in digital image correlation due to overmatched shape functions". In: *Measurement Science and Technology* 26 4, p. 045202.

316 Ioffe, S. and Szegedy, C. (2015). "Batch normalization: Accelerating deep network training by reducing internal covariate shift". In: *32nd International Conference on Machine Learning, ICML 2015*. Vol. 1, pp. 448–456.

317 He, K. et al. (2015b). "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37 9, pp. 1904–1916.

Appendix A
Chapter 5 appendices

A.1 Utilising parallel processing for ADIC3D

Parallel processing can be used to reduce ADIC3D's execution time by making use of MATLAB's `parfor` loops. More specifically, the for loops in line 11 of `StereoMatch`, 13 of `FeatureMatch` and 18 of `ImgCorr` can be replaced by `parfor` loops. It is for this reason that data within these for loops are saved to temporary storage variables, initiated prior to the for loops, and assigned to structured arrays thereafter. Note that `parfor` loops cause ADIC3D to require more random-access memory.

A.2 `UndistortPasser` function

Requiring `CSTrans` to analyse a large number of subsets causes MATLAB's `undistortPoints` function to require a large amount of random-access memory. If it requires more memory than is available this causes a fatal error. This is avoided by replacing calls to undistortPoints with calls to subroutine `UndistortPasser`, detailed in Table A.1, which has inputs `Xos` and the calibration parameters and returns the undistorted subset positions. UndistortPasser passes batches of 100 subsets to `undistortPoints` in order to avoid this fatal error. Note that this also reduces the computation time of lines 11 and 15 of `CSTrans`.

Table A.1: `UndistortPasser` **algorithm summary.**

Line Numbers	**Task Performed**
Line 2-3	Determine how many times the number of subsets is divisible by 100 and save as `iterations`;
Line 4	**for** q=1 to iterations, **do**
Line 5	Undistort a batch of 100 subset positions using MATLAB's `undistortPoints` function;
Line 6	**end for**
Line 7-9	Undistort the remaining subsets, not processed in the for loop, using `undistortPoints`;

A.3 Imposed displacements of Sample 1.

Table A.2: Mean and standard deviation of the imposed displacements measured by the stage's optical encoder for Sample 1 [123].

Step	$\hat{u}_w$ **mean (mm)**	$\hat{u}_w$ **STD (nm)**	$\hat{w}_w$ **mean (mm)**	$\hat{w}_w$ **STD (nm)**
1	0	7.01	0	6.76
2	0	7.69	10	6.16
3	0	6.30	20	6.21
4	0	7.67	-10	6.12
5	0	6.74	-20	6.33
6	-10	4.91	0	6.83
7	-20	5.71	0	7.27
8	10	6.53	0	6.79
9	20	5.69	0	7.37
10	-10	5.99	-10	4.57
11	-20	14.65	-20	25.19
12	10	7.65	10	6.43
13	20	6.10	20	6.54
14	-10	5.70	10	6.08
15	-20	5.14	20	6.45
16	10	6.29	-10	5.01
17	20	5.99	-20	6.07
18	0	6.36	0	7.59

Appendix B
Additional ANNDSS results

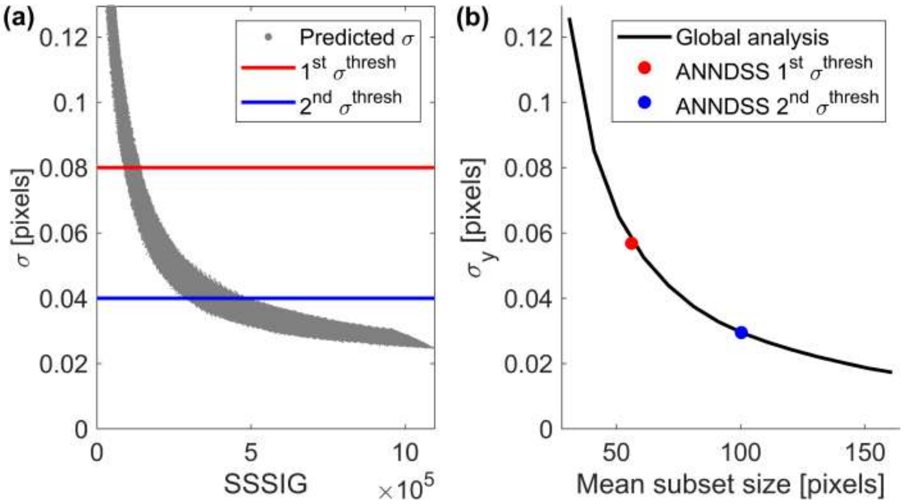


Figure B.1: Case 3 results: (a) The two σ^{thresh} plotted relative to the DESD predicted by the ANN; and (b) the σ_y for the global analysis and ANNDSS.

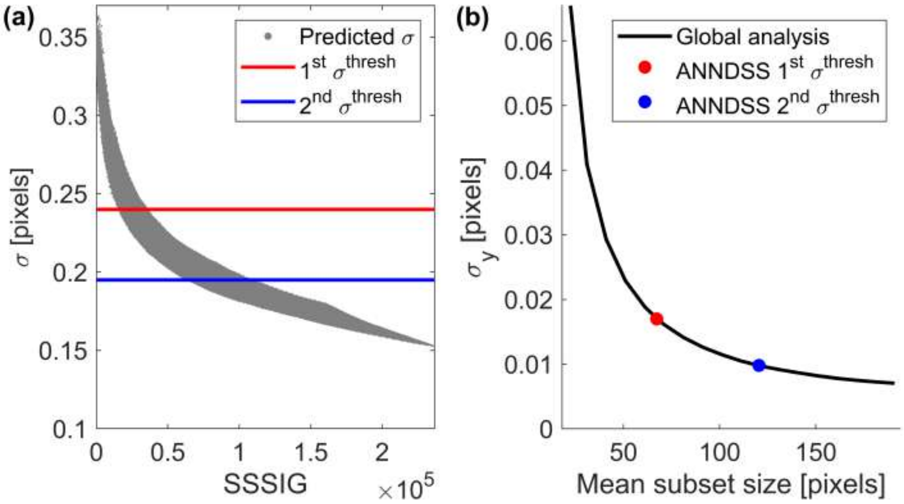


Figure B.2: Case 4 results: (a) The two σ^{thresh} plotted relative to the DESD predicted by the ANN; and (b) the σ_y for the global analysis and ANNDSS. The y-axis range of (a) was adapted such that the similarity in trend between the plots can be observed.

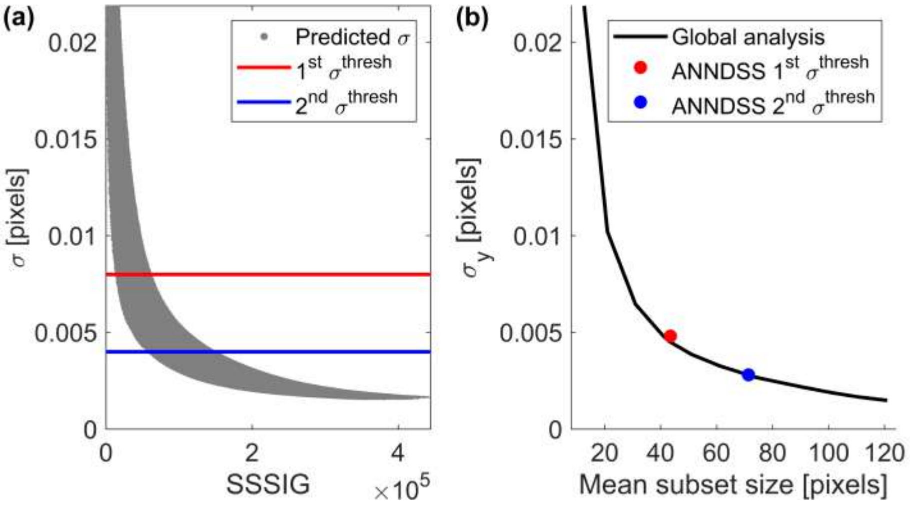


Figure B.3: Case 5 results: (a) The two σ^{thresh} plotted relative to the DESD predicted by the ANN; and (b) the σ_y for the global analysis and ANNDSS.