strain
An International Journal for Experimental Mechanics

Formulation of Lucas–Kanade Digital Image Correlation Algorithms for Non-contact Deformation Measurements: A Review

W. Tong

Department of Mechanical Engineering, Southern Methodist University, 3101 Dyer Street, Dallas, TX 75275-0337 USA

ABSTRACT: Digital image correlation (DIC) metrology has been increasingly used in a wide range of experimental mechanics research and applications. The DIC algorithm used so far is however limited mostly to the classic forward additive Lucas–Kanade type. In this paper, a survey is given about the formulation of other types of Lucas–Kanade DIC algorithms that have been appeared in computer vision, robotics, medical image analysis literature and so on. Concise notations consistent with the finite deformation kinematics analysis in continuum mechanics are used to describe all Lucas–Kanade DIC algorithms. An *intermediate image* is introduced as a frame of reference to clarify the so-called compositional algorithms in a two-frame DIC analysis. Explicit examples about the additive and compositional updating of deformation parameters are given for affine deformation mapping. Extensions of these algorithms to the so-called *consistent* or *symmetric* types are also presented. The equivalency of final numerical solutions using additive, compositional and inverse compositional algorithms is shown analytically for the case of affine deformation mapping. In particular, the inverse compositional algorithm for affine image subset deformation is highlighted for its superior computational efficiency. While computationally less efficient, consistent and symmetric algorithms may be more robust and less biased and their potentials in experimental mechanics applications remain to be explored. The unified formulation of these Lucas–Kanade DIC algorithms collected all together in this paper can serve as a useful guide for researchers in experimental mechanics to further evaluate the merits as well as limitations of these non-classic algorithms for image-based precision displacement measurement applications.

KEY WORDS: *digital image correlation, image alignment, image registration, motion estimation, non-contact deformation measurement, optical flow*

NOMENCLATURE

bold face symbols	Position and displacement vectors and vectored-valued functions and matrices
regular symbols	Scalars (such as greyscale values of image pixels and coordinates of subset points and their deformation parameters, and so on)
upper case symbols	Initial (template) image: forward displacement, deformation mapping or transformation
lower case symbols	Current (input) image: backward displacement, deformation mapping or transformation
subscript i	A discrete point *i* in a subset of points of an image appeared in a sum-of-squared differences (SSD) correlation coefficient
the accent '∼'over a symbol	An intermediate image between an initial image and its current image coordinates, relative displacement and its parameters using such an intermediate image as the frame of reference
The accent '–' over a symbol	A special intermediate image that is used in a symmetric analysis and is exactly half-way (i.e. 'symmetric') between an initial image and its current image coordinates, relative displacement and its parameters using such a symmetric intermediate image as the frame of reference
the accent '∧'over a symbol	Relative displacement and its parameters of an intermediate image
$G(\mathbf{X})$, $g(\mathbf{x})$	Greyscale values of points of an initial image and its current image often recorded by a digital imaging device
$\mathbf{X}$, $\mathbf{x}$	Position vectors of pixel points of an initial image and its current image
(X,Y), (x,y)	Coordinates of 2D position vectors of pixel points of an initial image and its current image
$\mathbf{U}(\mathbf{X})$, $\Delta\mathbf{U}(\mathbf{X})$	Relative (forward) displacement field and its increment between a subset of points of an initial image and the corresponding points of the current image
$\mathbf{u}(\mathbf{x})$, $\Delta\mathbf{u}(\mathbf{x})$	Relative (backward) displacement field and its increment between a subset of points of the current image and the corresponding points of its initial image

doi: 10.1111/str.12039

$G(\mathbf{x}+\mathbf{u})$, $g(\mathbf{X}+\mathbf{U})$	Numerically distorted image (from the initial image) or numerically restored image (from the current image)
C_F, C_B,…	Various sum-of-squared differences coefficients or the so-called L_2 objective similarity functions
$\mathbf{P}$, $\Delta\mathbf{P}$, $\mathbf{p}$, $\Delta\mathbf{p}$, $\widetilde{\mathbf{P}}$, $\Delta\widetilde{\mathbf{P}}$, $\widetilde{\mathbf{p}}$, $\Delta\widetilde{\mathbf{p}}$, etc.	Parameters and their increments that define the forward and backward deformation mapping, transformation matrices and displacement functions
$\mathbf{W}(\mathbf{X};\mathbf{P})$	Parameterized forward warp or deformation mapping that transforms the coordinates of a subset of points of an initial image to the coordinates of the corresponding points of the current image
$\mathbf{w}(\mathbf{x};\mathbf{p})$	Parameterized backward warp or deformation mapping that transforms the coordinates of a subset of points of the deformed image to the coordinates of the corresponding points of the initial image
U_0, V_0, U_X, U_Y, V_X, V_Y	A six-parameter set for 2D forward affine deformation mapping of a subset (displacements and their gradients for a 2D forward displacement function)
u_0, v_0, u_x, u_y, v_x, v_y	A six-parameter set for 2D backward affine deformation mapping of a subset (displacements and their gradients for a 2D backward displacement function)
$\widetilde{\mathbf{X}}, \widetilde{G}\left(\widetilde{\mathbf{X}}\right), \widetilde{g}\left(\widetilde{\mathbf{X}}\right)$	Coordinates and interpolated greyscale of an intermediate (numerically deformed or restored) image between initial and current images in a forward analysis
$\widetilde{\mathbf{x}}, \widetilde{g}(\widetilde{\mathbf{x}}), \widetilde{G}(\widetilde{\mathbf{x}})$	Coordinates and interpolated greyscale of an intermediate image (numerically restored or deformed) between initial and current images in a backward analysis
$\Phi(\hat{\mathbf{P}})$, $\varphi(\hat{\mathbf{p}})$ $\Phi\left(\widetilde{\mathbf{P}}\right)$, $\varphi\left(\widetilde{\mathbf{p}}\right)$, etc.	Prior estimated parameter matrices for linearized forward and backward deformation mappings, and they are assumed to be invertible in this work
$\Phi\left(\Delta\widetilde{\mathbf{P}}\right)$, $\Phi(\Delta\hat{\mathbf{P}})$ $\varphi\left(\Delta\widetilde{\mathbf{P}}\right)$, $\varphi(\Delta\hat{\mathbf{p}})$, etc.	Parameter matrices for incremental forward and backward deformation mappings, respectively (to be solved numerically in an iterative process)
$\Phi(\mathbf{P}+\Delta\mathbf{P})$, $\varphi(\mathbf{p}+\Delta\mathbf{p})$	The updated parameter matrices between the initial and current images in forward and backward analyses, respectively

Introduction

Displacement measurements are commonly encountered in many scientific experiments and engineering applications in general and in materials and structural testing in particular. The measurement accuracy and precision requirements are often most stringent in the latter cases as full-field strains or displacement gradients are needed in evaluating deformation and failure behaviours of materials and thermal–mechanical responses of structures under various types of loadings. In recent years, the non-contact, full-field deformation measurement metrology based on digital image correlation (DIC) has increasingly become the technique of choice in experimental mechanics [1–3]. This development trend is attributed partly to the wide availability of affordable high quality digital imaging devices and high performance computers and more importantly to the sustained research efforts on the development and implementation of DIC algorithms to meet such stringent requirements for off-line deformation measurements. Nowadays, displacements over a reasonably sized image subset can be measured within an error level of 0.01 pixel or less when one follows the recommended digital image formation, acquisition, and processing procedures in routine laboratory materials testing and structural monitoring applications [2, 4, 5]. Nevertheless, as the method is being extended in both spatial and temporal scales using various digital microscopy and high speed imaging systems and is being applied to an ever wider array of materials testing and on-line structural health monitoring applications, potential opportunities arise about further research on the formulation of various DIC algorithms to meet these new challenges involved noisy images, large heterogeneous deformation and faster and portable processing needs.

Historically, the initial development of the DIC method used in experimental solid mechanics applications follows the coherent optical interferometry techniques. Inspired in particular by both the electronic speckle pattern interferometry (ESPI) method and the area correlation method in pattern recognition, a correlation analysis instead of the fringe formation analysis was applied at first to images of laser speckle patterns as acquired in an ESPI experiment [6]. The DIC analysis algorithm was initially formulated in terms of the cross-correlation-based inverse problem of image restoration and rigid body translation of laser speckle patterns and was quickly extended to more general situations of white light incoherent digital images using least-squares correlation and affine deformation [7]. The development and applications of similar spatial-domain, image-based displacement measurement methods have also been reported by others in experimental mechanics [8, 9]. The performance of the DIC method was further enhanced (1) by formulating its algorithm as an

doi: 10.1111/str.12039

iterative cross-correlation problem and solving it by the second-order Newton–Raphson (also called Newton or full Newton) numerical solution method instead of the coarse-fine search method used previously [10] and (2) by replacing the bilinear interpolation with bi-cubic interpolation of subpixel greyscales of deformed images [11]. Vendroux and Knauss [4] further developed the DIC method by formulating its algorithm as an iterative least-squares (also called sum-of-squared differences or SSD) correlation problem with a photometric bias correction term on image greyscales. They applied the new algorithm along with the first-order approximate Newton (also called Gauss–Newton or quasi-Newton) method for submicron displacement measurements using atomic force microscopy images. Tong [5] showed that both cross-correlation and SSD correlation DIC algorithms can be represented in a normalisation form to eliminate the photometric gain and bias, and they are actually equivalent (although the first-order SSD algorithm is computationally more robust and efficient). At present, research works and application examples appeared in the experimental mechanics literature use almost exclusively either one of the DIC algorithms and associated numerical solution methods with no or only minor variations [1–3].

Image analyses similar or even identical to DIC have also been developed in parallel and extensively used in many other fields such as computer vision, robotics, pattern recognition, medical image analysis, experimental fluid mechanics and so on. These image analysis methods are instead called image registration, image alignment or image matching [12–16], optical flow computation [17, 18], object tracking or motion estimation [19–22], digital particle image velocimetry (DPIV or PIV) [23, 24] and so on. Almost all DPIV techniques use cross-correlation-based algorithms, and they locate the correlation peak coarsely first by fast Fourier transform in the frequency domain and then determine the dense displacement field vectors at subpixel accuracy by a certain peak-fit routine. On the other hand, the iterative gradient-based Lucas–Kanade SSD image registration algorithm [12] has been found to be one of the most reliable optical flow methods carried out in the spatial domain [18]. In addition to the original forward additive Lucas–Kanade algorithm, many variations in the algorithm formulation such as compositional, inverse compositional, consistent and symmetric algorithms have subsequently been developed for a wide range of applications in computer vision and medical image analysis [13–15, 21, 22, 25]. It turns out that the DIC algorithms most often used in experimental mechanics as described above are more or less equivalent to the sparse optical flow version of the original so-called *forward additive* Lucas–Kanade algorithm (which also accounts for affine deformation and photometric gain and bias by Lucas and Kanade [12]). A backward additive Lucas–Kanade DIC algorithm has only occasionally been used for analysing image sequences with growing cracks [26, 27] and has more recently been shown to have no noise-induced bias when the initial (reference) image is noise-free [28]. Many other formulations of the DIC algorithms have rarely been evaluated at all for possible materials testing and structural monitoring applications. One of the major reasons is due to different and sometimes confusing notations used in formulating those non-classic algorithms in computer vision and medial image analysis literature. In particular, a clear definition of a frame of reference for various relative displacements and deformation mappings between images is often lacking as precision deformation measurements themselves are normally not a primary concern in those applications.

This paper aims to provide a systematic survey on the formulation of existing DIC algorithms appeared in experimental mechanics, computer visions, robotics and medical image analysis literature. The purpose of this review is to introduce the non-classic algorithms commonly used in the other fields and their possible variants to the researchers and users in experimental mechanics community. In the second section, we use concise notations consistent with the finite deformation kinematics analysis in continuum mechanics to describe various Lucas–Kanade SSD DIC analysis algorithms. An *intermediate image* is introduced as a frame of reference to clarify the so-called compositional as well as consistent and symmetric DIC algorithms. It is hoped that such a unified presentation will make these non-classic Lucas–Kanade DIC algorithms more accessible to the practitioners interested in deformation measurements, so some of these algorithms may be explored as possible better alternatives for current or future deformation measurement application needs. In particular, Lucas–Kanade SSD DIC algorithms are described in details in terms of either *forward* or *backward* deformation mapping direction and in terms of either *additive* or *compositional* updating the parameters of the deformation mapping function in an iterative numerical solution process. Explicit examples about the additive and compositional updating of deformation parameters are given for affine deformation mapping. Additive and compositional updating is also extended to the so-called consistent or symmetric DIC algorithms.

As an example, we show in the third section how one can use the Gauss–Newton method (out of many numerical methods) to carry out the computational implementation of forward additive, compositional and inverse compositional algorithms by obtaining closed form solution of parameter increments of subset deformation for each algorithm. We repeat the proof on the equivalency among these algorithms analytically using the new notations proposed in this paper for the case of 1D and 2D affine warps. Examples of how to implement each algorithm are further detailed by specifying the central difference computation

doi: 10.1111/str.12039

of spatial gradients of greyscales and the linear interpolation of greyscales of images at non-integral pixel positions for one-dimensional affine deformation. In particular, the computational efficiency of the inverse compositional algorithm is pointed out. Both the advantage of using the new notations and associated schematics in describing non-classic Lucas–Kanade DIC algorithms and possible benefits of using one of these non-classic Lucas–Kanade algorithms for deformation measurement applications are discussed in the fourth section. Additional research needs for systematically assessing the relative merits and limitations of these algorithms for a given deformation measurement scenario are also briefly addressed. The paper concludes with a summary of main results of the paper in the fifth section.

A Continuum Mechanics Formulation of Digital Image Correlation Algorithms

Kinematics of image deformation and digital image correlation analysis

Consider an initial or original image with a greyscale level $G(\mathbf{X})$ and its corresponding current or deformed image with a greyscale level $g(\mathbf{x})$. A subset of n points of the initial image with position vectors $\mathbf{X}$ and the corresponding points of the current image with position vectors $\mathbf{x}$ are related by *displacement field* vectors $\mathbf{U}(\mathbf{X})$, namely, $\mathbf{x}=\mathbf{W}(\mathbf{X})=\mathbf{X}+\mathbf{U}(\mathbf{X})$. The displacements can be estimated in a DIC analysis by minimising an SSD coefficient C_F [12]

$$C_F(\mathbf{U})=\sum_{i=1}^{n}[\widetilde{g}(\mathbf{X}_i)-G(\mathbf{X}_i)]^2=\sum_{i=1}^{n}[g(\mathbf{X}_i+\mathbf{U})-G(\mathbf{X}_i)]^2. \quad (1)$$

The analysis algorithm in Equation (1) is called *forward* (deformation mapping) DIC (thus, the subscript 'F' for the SSD coefficient) as the resulting displacements $\mathbf{U}(\mathbf{X})$ map the subset of points of the initial image forward to the corresponding points of the current image. That is, the initial image is used as a frame of reference in displacement measurements of points between these two images, and the image $\widetilde{g}(\mathbf{X})$ in Equation (1) should be regarded as a copy of the current image *numerically restored* back to its initial state. As position vectors $\mathbf{X}$ are usually selected to be integer-valued (points with integral pixel coordinates of the initial image), greyscale values of the restored image $\widetilde{g}(\mathbf{X})$ are normally obtained by interpolating the spatially discrete greyscale values of the current image as $g(\mathbf{X}+\mathbf{U})$.

On the other hand, an analysis using the *backward* (deformation mapping) DIC algorithm is to estimate displacement vectors $\mathbf{u}(\mathbf{x})$ by minimising the following SSD coefficient C_B

$$C_B(\mathbf{u})=\sum_{i=1}^{n}\left[\widetilde{G}(\mathbf{x}_i)-g(\mathbf{x}_i)\right]^2=\sum_{i=1}^{n}[G(\mathbf{x}_i+\mathbf{u})-g(\mathbf{x}_i)]^2, \quad (2)$$

where $\mathbf{u}(\mathbf{x})$ are displacements that map the subset of points of the current image back to the corresponding points of the initial image, namely $\mathbf{X}=\mathbf{w}(\mathbf{x})=\mathbf{x}+\mathbf{u}(\mathbf{x})$. That is, the current image is used as a frame of reference in measuring the displacements $\mathbf{u}(\mathbf{x})$ of points in the subset between these two images. Similarly, the image $\widetilde{G}(\mathbf{x})$ is a copy of the initial image *numerically distorted* towards to the current state and is again normally obtained by interpolating the spatially discrete greyscale values of the initial image as $G(\mathbf{x}+\mathbf{u})$. Table 1 summarises the graphic definition of relative displacements and restored or distorted images in these two and other forward and backward DIC algorithms in terms of specific frames of reference.

The above forward deformation $\mathbf{W}(\mathbf{X})$ and backward deformation $\mathbf{w}(\mathbf{x})$ of an image subset between an initial and its current images may be represented in terms of a differentiable continuous function with respect to a set of scalar parameters $\mathbf{P}$ and $\mathbf{p}$ as [15, 22]

$$\mathbf{x}=\mathbf{W}(\mathbf{X};\mathbf{P})=\mathbf{X}+\mathbf{U}(\mathbf{X};\mathbf{P}), \quad \text{and} \quad \mathbf{X}=\mathbf{w}(\mathbf{x};\mathbf{p})=\mathbf{x}+\mathbf{u}(\mathbf{x};\mathbf{p}). \quad (3a)$$

$\mathbf{U}(\mathbf{X};\mathbf{P})$ and $\mathbf{u}(\mathbf{x};\mathbf{p})$ are the parameterized forward and backward *displacement field functions*, respectively. In the computer vision literature, $\mathbf{W}(\mathbf{X};\mathbf{P})$ and $\mathbf{w}(\mathbf{x};\mathbf{p})$ are commonly called parameterized forward and backward *warps* (by the way, an initial image and its current image are called, respectively, as 'template' and 'input' images) [15]. It is possible for some deformation functions to have the following form

$$\mathbf{W}(\mathbf{X};\mathbf{P})=\mathbf{\Phi}(\mathbf{P})\mathbf{X}, \quad \text{and} \quad \mathbf{w}(\mathbf{x};\mathbf{p})=\boldsymbol{\varphi}(\mathbf{p})\mathbf{x}, \quad (3b)$$

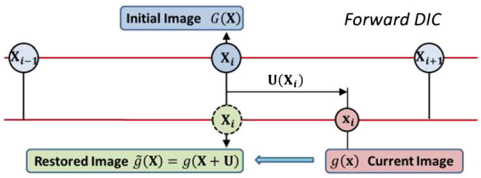
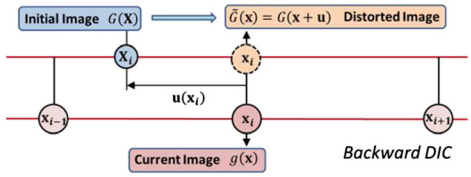
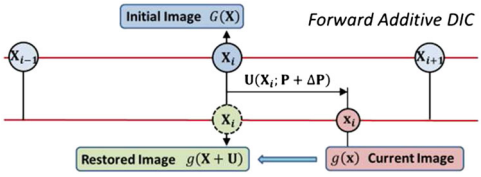
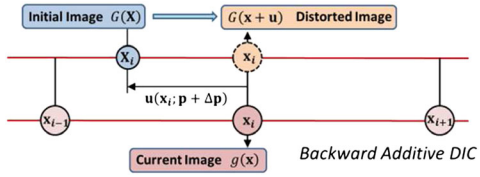
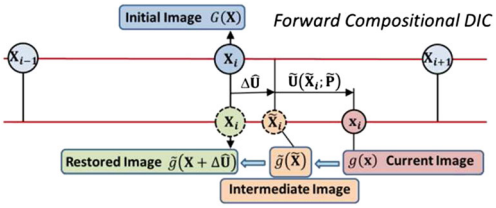
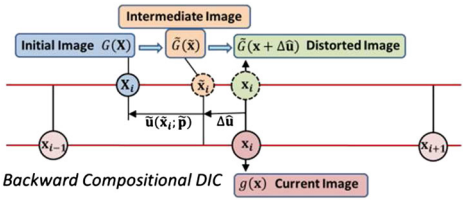
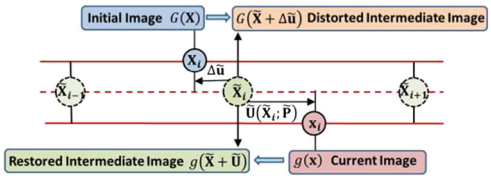
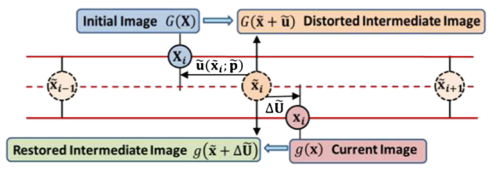
where $\mathbf{\Phi}(\mathbf{P})$ and $\boldsymbol{\varphi}(\mathbf{p})$ in Equation (3b) will be called transformation matrices or deformation parameter matrices between images [22]. In this work, it is assumed that the deformation of an image subset can always be described by a linear parameterized deformation mapping with a parameter matrix defined as in Equation (3b). In particular, a six-parameter affine displacement function is just such a deformation mapping and is often used to model a homogenous deformation field of a subset between the initial image $G(\mathbf{X})=G(X,Y)$ and the current image $g(\mathbf{x})=g(x,y)$. It has the following form for both forward and backward deformation analyses, respectively, as

$$\begin{cases} x=X+U_0+U_X X+U_Y Y, \\ y=Y+V_0+V_X X+V_Y Y, \end{cases} \quad \text{and} \quad \begin{cases} X=x+u_0+u_x x+u_y y, \\ y=y+v_0+v_x x+v_y y, \end{cases} \quad (4)$$

where parameters (U_0,V_0) and (u_0,v_0) are horizontal and vertical displacement components and parameters

© 2013 Wiley Publishing Ltd | *Strain* (2013) **49**, 313–334
doi: 10.1111/str.12039

Table 1: Summary of forward/backward and additive/compositional DIC algorithms

Forward DIC algorithms	Backwards DIC algorithms
	
$C_F(\mathbf{U}) = \sum_{i=1}^{n} [g(\mathbf{X}_i + \mathbf{U}) - G(\mathbf{X}_i)]^2$	$C_B(\mathbf{u}) = \sum_{i=1}^{n} [G(\mathbf{x}_i + \mathbf{u}) - g(\mathbf{x}_i)]^2$
	
$C_{FA}(\Delta\mathbf{P}) = \sum_{i=1}^{n} [g(\mathbf{X}_i + \mathbf{U}(\mathbf{X}_i;\mathbf{P}+\Delta\mathbf{P})) - G(\mathbf{X}_i)]^2$	$C_{BA}(\Delta\mathbf{p}) = \sum_{i=1}^{n} [G(\mathbf{x}_i + \mathbf{u}(\mathbf{x}_i;\mathbf{p}+\Delta\mathbf{p})) - g(\mathbf{x}_i)]^2$
	
$C_{FC}(\Delta\hat{\mathbf{P}}) = \sum_{i=1}^{n} \left[\tilde{g}(\mathbf{X}_i + \Delta\hat{\mathbf{U}}(\mathbf{X}_i;\Delta\hat{\mathbf{P}})) - G(\mathbf{X}_i)\right]^2$	$C_{BC}(\Delta\hat{\mathbf{p}}) = \sum_{i=1}^{n} \left[\tilde{G}(\mathbf{x}_i + \Delta\hat{\mathbf{u}}(\mathbf{x}_i;\Delta\hat{\mathbf{p}})) - g(\mathbf{x}_i)\right]^2$
Forward Inverse Compositional DIC 	*Backward Inverse Compositional DIC* 
$C_{FIC}(\Delta\tilde{\mathbf{p}}) = \sum_{i=1}^{n} \left[G\left(\tilde{\mathbf{X}}_i + \Delta\tilde{\mathbf{u}}\left(\tilde{\mathbf{X}}_i;\Delta\tilde{\mathbf{p}}\right)\right) - g\left(\tilde{\mathbf{X}}_i + \tilde{\mathbf{U}}\right)\right]^2$	$C_{BIC}\left(\Delta\tilde{\mathbf{P}}\right) = \sum_{i=1}^{n} \left[g\left(\tilde{\mathbf{x}}_i + \Delta\tilde{\mathbf{U}}\left(\tilde{\mathbf{x}}_i;\Delta\tilde{\mathbf{P}}\right)\right) - G(\tilde{\mathbf{x}}_i + \tilde{\mathbf{u}})\right]^2$

DIC, digital image correlation.
Letters with a subscripted index '*i*' enclosed in a circle are pixel coordinates of points in various images as indicated by a thin vertical line. Coordinates having indexes of '$i-1$', 'i' and '$i+1$' are points of the frame of reference with integral pixel coordinates used in an analysis. A thick vertical line with arrows at both ends indicates the pixel point on the frame of the reference where the image correlation analysis is being carried out.
An arrow with a single thick horizontal line indicates the displacement vector direction: a forward displacement with the arrow pointing to the right and a backward displacement with the arrow pointing to the left. Upper or lower case letters are, respectively, used to describe the forward or backward displacements and their parameters. An arrow with two thin horizontal lines indicates the numerically distorted (the arrow pointing to the right) or restored (the arrow pointing to the left) images from the source to target images. The displacements are, in general, non-integral in pixels.

doi: 10.1111/str.12039

(U_X,U_Y,V_X,V_Y) and (u_x,u_y,v_x,v_y) are corresponding displacement gradients, respectively. The above equations can be written in a matrix form as [15]

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1+U_X & U_Y & U_0 \\ V_X & 1+V_Y & V_0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \quad \textit{and} \quad \begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} 1+u_x & u_y & u_0 \\ v_x & 1+v_Y & v_0 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \tag{5a}$$

or in terms of homogenous coordinates as [22]

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} 1+U_X & U_Y & U_0 \\ V_X & 1+V_Y & V_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \quad \textit{and} \quad \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} = \begin{pmatrix} 1+u_x & u_y & u_0 \\ v_x & 1+v_Y & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}. \tag{5b}$$

The four terms containing displacement gradients in the affine parameter matrices above form the so-called 2D *deformation gradient tensor*. It is used extensively in defining various strain and stress measures and in formulating constitutive equations of materials in continuum mechanics [36]. The determinant of the deformation gradient tensor is the ratio of volumes (3D) or areas (2D) of a material element after and before the deformation. For any physically admissible continuous deformation, the ratio of subset areas in both initial and current images must be positive. Therefore, all physically admissible affine deformation must satisfy the condition $D=(1+U_X)(1+V_Y)-U_YV_X>0$, where D is the determinant of the 2D deformation gradient tensor. It should be pointed out that the inverse of an affine parameter matrix with $\mathbf{P}=(U_0,V_0,U_X,U_Y,V_X,V_Y)^T$ can be readily obtained, for example:

$$\mathbf{\Phi}(\mathbf{P}) = \begin{pmatrix} 1+U_X & U_Y & U_0 \\ V_X & 1+V_Y & V_0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{6a}$$

$$\mathbf{\Phi}(\mathbf{P})^{-1} = \frac{1}{D} \begin{pmatrix} 1+V_Y & -U_Y & U_YV_0-V_YU_0-U_0 \\ -V_X & 1+U_X & V_XU_0-U_XV_0-V_0 \\ 0 & 0 & (1+U_X)(1+V_Y)-U_YV_X \end{pmatrix}. \tag{6b}$$

So the condition $D>0$ for physically admissible affine deformation also guarantees that the affine parameter matrices $\mathbf{\Phi}(\mathbf{P})$ and $\boldsymbol{\varphi}(\mathbf{p})$ are invertible. As expected, $\boldsymbol{\varphi}(\mathbf{p})=\mathbf{\Phi}(\mathbf{P})^{-1}$ in this case. For rigid-body translations, $\mathbf{p}=-\mathbf{P}$. For $\mathbf{P}\to 0$, $\mathbf{\Phi}(\mathbf{P})^{-1}\approx\mathbf{\Phi}(-\mathbf{P})$ to the first-order approximation in $\mathbf{P}$ [15].

Forward and backward additive digital image correlation algorithms

When the SSD coefficient defined in either Equation (1) or (2) is a nonlinear function of the parameters of the displacement field function given in Equation (3), the numerical solution process of minimising the SSD coefficient will be an iterative one in which the parameters are updated incrementally. Even in a linear analysis, an incremental updating DIC algorithm is also useful for computing subpixel displacements with poor prior estimates [28, 29]. Consequently, the SSD coefficients given in Equation (1) or (2) may be rewritten, respectively, as

$$C_{FA}(\Delta\mathbf{P}) = \sum_{i=1}^{n} \left[g(\mathbf{X}_i+\mathbf{U}(\mathbf{X}_i;\mathbf{P}+\Delta\mathbf{P}))-G(\mathbf{X}_i)\right]^2, \tag{7}$$

and

$$C_{BA}(\Delta\mathbf{p}) = \sum_{i=1}^{n} \left[G(\mathbf{x}_i+\mathbf{u}(\mathbf{x}_i;\mathbf{p}+\Delta\mathbf{p}))-g(\mathbf{x}_i)\right]^2, \tag{8}$$

where $\mathbf{P}$ and $\mathbf{p}$ are prior estimated parameters of the forward and backward displacement fields of the image subset, and $\Delta\mathbf{P}$ and $\Delta\mathbf{p}$ are their incremental updates to be solved numerically by minimising the SSD coefficients. They are analogous to the *predictor–corrector* pairs in any iterative numerical solution algorithm. The displacement functions are simply updated as $\mathbf{U}(\mathbf{X};\mathbf{P}+\Delta\mathbf{P})$ and $\mathbf{u}(\mathbf{x};\mathbf{p}+\Delta\mathbf{p})$ at each iteration step. The numerical solution process continues until the incremental displacements $\Delta\mathbf{U}=\mathbf{U}(\mathbf{X};\mathbf{P}+\Delta\mathbf{P})-\mathbf{U}(\mathbf{X};\mathbf{P})$ and $\Delta\mathbf{u}=\mathbf{u}(\mathbf{x};\mathbf{p}+\Delta\mathbf{p})-\mathbf{u}(\mathbf{x};\mathbf{p})$ becomes vanishingly small in each analysis. Because the incremental updates of the deformation parameters are explicitly solved and then added directly to the prior estimates at each iteration step, the algorithms given in Equations (7) and (8) are called *forward additive* DIC (the classic Lucas–Kanade algorithm [12, 15]) and *backward additive* DIC, respectively. For example, the updated affine parameter matrix for the forward additive DIC algorithm is given simply as

$$\mathbf{\Phi}(\mathbf{P}+\Delta\mathbf{P}) = \begin{pmatrix} 1+U_X+\Delta U_X & U_Y+\Delta U_Y & U_0+\Delta U_0 \\ V_X+\Delta V_X & 1+V_Y+\Delta V_Y & V_0+\Delta V_0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{9}$$

Forward and backward compositional digital image correlation algorithms

A possible alternative to the additive updating DIC algorithms is the so-called compositional updating algorithms, for example [15, 22]:

$$C_{FC}(\Delta\hat{\mathbf{P}}) = \sum_{i=1}^{n} \left[\tilde{g}\left(\mathbf{X}_i+\Delta\hat{\mathbf{U}}\left(\mathbf{X}_i;\Delta\hat{\mathbf{P}}\right)\right)-G(\mathbf{X}_i)\right]^2, \tag{10}$$

$$\tilde{g}\left(\widetilde{\mathbf{X}}_i\right) = g\left(\widetilde{\mathbf{X}}_i+\widetilde{\mathbf{U}}\left(\widetilde{\mathbf{X}}_i;\widetilde{\mathbf{P}}\right)\right),$$

where $\widetilde{\mathbf{U}}\left(\widetilde{\mathbf{X}};\widetilde{\mathbf{P}}\right)$ is the prior estimated forward displacement field between an intermediate image $\tilde{g}\left(\widetilde{\mathbf{X}}\right)$ and the current

© 2013 Wiley Publishing Ltd | *Strain* (2013) **49**, 313–334
doi: 10.1111/str.12039

image $g(\mathbf{x})$, and $\Delta\hat{\mathbf{U}}(\mathbf{X};\Delta\hat{\mathbf{P}})$ is the incremental forward displacement field between the initial image $G(\mathbf{X})$ and the intermediate image $\widetilde{g}(\widetilde{\mathbf{X}})$, namely

$$\begin{aligned}&\mathbf{x}=\widetilde{\mathbf{X}}+\widetilde{\mathbf{U}}(\widetilde{\mathbf{X}};\widetilde{\mathbf{P}})=\boldsymbol{\Phi}(\widetilde{\mathbf{P}})\widetilde{\mathbf{X}},\\&\widetilde{\mathbf{X}}=\mathbf{X}+\Delta\hat{\mathbf{U}}(\mathbf{X};\Delta\hat{\mathbf{P}})=\boldsymbol{\Phi}(\Delta\hat{\mathbf{P}})\mathbf{X}.\end{aligned}\tag{11a}$$

So the forward deformation parameter matrix from the initial image to the current image can be computed as the composition of two successive warps (i.e. multiplication of their parameter matrices)

$$\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})=\boldsymbol{\Phi}(\widetilde{\mathbf{P}})\boldsymbol{\Phi}(\Delta\hat{\mathbf{P}}),\quad\text{where}\quad\mathbf{x}=\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})\mathbf{X}.\tag{11b}$$

It is noted that parameters without and with an accent of either 'tilde' or 'hat' are used to account for three different relative displacements among three image pairs (see Table 1). The way of updating deformation parameters as given in Equation (11b) is termed *compositional* in the computer vision literature [15] or composite mapping in the medical image analysis literature [16] as the estimated relative displacements $\widetilde{\mathbf{U}}$ and the increments $\Delta\hat{\mathbf{U}}$ are measured according to two different frames of reference. The algorithm given in Equation (10) will thus be called the *forward compositional* DIC algorithm. To continue the next solution step, the parameters $\widetilde{\mathbf{P}}$ are further updated to $\mathbf{P}+\Delta\mathbf{P}\rightarrow\widetilde{\mathbf{P}}+\Delta\widetilde{\mathbf{P}}$ until $\Delta\hat{\mathbf{U}}\rightarrow\mathbf{0}$ (so $\Delta\hat{\mathbf{P}}\rightarrow\mathbf{0}$ and $\widetilde{\mathbf{X}}\rightarrow\mathbf{X}$ as well at the end of the iterative solution process). As an example, the updated affine parameter matrix for the forward compositional DIC algorithm is given as

$$\begin{aligned}\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})&=\boldsymbol{\Phi}(\widetilde{\mathbf{P}})\boldsymbol{\Phi}(\Delta\hat{\mathbf{P}})\\&=\begin{pmatrix}1+\widetilde{U}_X&\widetilde{U}_Y&\widetilde{U}_0\\\widetilde{V}_X&1+\widetilde{V}_Y&\widetilde{V}_0\\0&0&1\end{pmatrix}\begin{pmatrix}1+\Delta\hat{U}_X&\Delta\hat{U}_Y&\Delta\hat{U}_0\\\Delta\hat{V}_X&1+\Delta\hat{V}_Y&\Delta\hat{V}_0\\0&0&1\end{pmatrix}.\end{aligned}\tag{12}$$

For rigid body translations with $\mathbf{P}=(U_0,V_0)$, Equations (9) and (12) are however identical. So there is no difference at all between the compositional and additive updating of rigid body translation parameters in these two algorithms.

For the completeness of the discussion, the equations for the *backward compositional* DIC algorithm are also given in the following:

$$\begin{aligned}&C_{BC}(\Delta\hat{\mathbf{p}})=\sum_{i=1}^{n}\left[\widetilde{G}(\mathbf{x}_i+\Delta\hat{\mathbf{u}}(\mathbf{x}_i;\Delta\hat{\mathbf{p}}))-g(\mathbf{x}_i)\right]^2,\\&\widetilde{G}(\widetilde{\mathbf{x}}_i)=G(\widetilde{\mathbf{x}}_i+\widetilde{\mathbf{u}}(\widetilde{\mathbf{x}}_i;\widetilde{\mathbf{p}})),\end{aligned}\tag{13}$$

where (see Table 1 for details)

$$\begin{aligned}&\mathbf{X}=\widetilde{\mathbf{x}}+\widetilde{\mathbf{u}}(\widetilde{\mathbf{x}};\widetilde{\mathbf{p}})=\boldsymbol{\varphi}(\widetilde{\mathbf{p}})\widetilde{\mathbf{x}},\quad\text{and}\\&\widetilde{\mathbf{x}}=\mathbf{x}+\Delta\hat{\mathbf{u}}(\mathbf{x};\Delta\hat{\mathbf{p}})=\boldsymbol{\varphi}(\Delta\hat{\mathbf{p}})\mathbf{x},\end{aligned}\tag{14a}$$

$$\boldsymbol{\varphi}(\mathbf{p}+\Delta\mathbf{p})=\boldsymbol{\varphi}(\widetilde{\mathbf{p}})\boldsymbol{\varphi}(\Delta\hat{\mathbf{p}}),\ \text{ so }\ \mathbf{X}=\boldsymbol{\varphi}(\mathbf{p}+\Delta\mathbf{p})\mathbf{x}.\tag{14b}$$

The backward compositional image correlation analysis estimates the parameter increments $\Delta\hat{\mathbf{p}}$ of a parameter matrix between the current image $g(\mathbf{x})$ (as a frame of reference) and an intermediate image $\widetilde{G}(\widetilde{\mathbf{x}})$ which itself is numerically distorted towards the current image $g(\mathbf{x})$ from the initial image $G(\mathbf{X})$ based on the prior estimated backward displacement field $\widetilde{\mathbf{u}}(\widetilde{\mathbf{x}};\widetilde{\mathbf{p}})$.

Inverse compositional digital image correlation algorithms

Both the prior estimated displacements and the corrective increments of displacements are in the same direction (either forward or backward) in the compositional algorithms discussed so far (see Table 1). When their deformation mapping directions are different, another variant of compositional algorithms can be formulated. The SSD coefficient for the *forward deformation mapping with inverse compositional updating* DIC algorithm (or more commonly known as the inverse compositional algorithm in the computer vision literature) is given as

$$\begin{aligned}C_{FIC}(\Delta\widetilde{\mathbf{p}})&=\sum_{i=1}^{n}\left[G\left(\widetilde{\mathbf{X}}_i+\Delta\widetilde{\mathbf{u}}\right)-\widetilde{g}\left(\widetilde{\mathbf{X}}_i\right)\right]^2\\&=\sum_{i=1}^{n}\left[G\left(\widetilde{\mathbf{X}}_i+\Delta\widetilde{\mathbf{u}}\left(\widetilde{\mathbf{X}}_i;\Delta\widetilde{\mathbf{p}}\right)\right)-g\left(\widetilde{\mathbf{X}}_i+\widetilde{\mathbf{U}}\left(\widetilde{\mathbf{X}}_i;\widetilde{\mathbf{P}}\right)\right)\right]^2,\end{aligned}\tag{15}$$

where the intermediate image $\widetilde{g}(\widetilde{\mathbf{X}})$ is used as a frame of reference for measuring both the prior estimated *forward* displacement $\widetilde{\mathbf{U}}(\widetilde{\mathbf{X}};\widetilde{\mathbf{P}})$ towards the current image $g(\mathbf{x})$ and the *backward* displacement increment $\Delta\widetilde{\mathbf{u}}(\widetilde{\mathbf{X}}_i;\Delta\widetilde{\mathbf{p}})$ towards the initial image $G(\mathbf{X})$, namely (see Table 1)

$$\begin{aligned}&\mathbf{X}=\widetilde{\mathbf{X}}+\Delta\widetilde{\mathbf{u}}(\widetilde{\mathbf{X}};\Delta\widetilde{\mathbf{p}})=\boldsymbol{\varphi}(\Delta\widetilde{\mathbf{p}})\widetilde{\mathbf{X}},\text{and}\\&\mathbf{x}=\widetilde{\mathbf{X}}+\widetilde{\mathbf{U}}(\widetilde{\mathbf{X}};\widetilde{\mathbf{P}})=\boldsymbol{\Phi}(\widetilde{\mathbf{P}})\widetilde{\mathbf{X}}.\end{aligned}\tag{16a}$$

The updated parameter matrix from the initial image to the current image is computed as (assuming the incremental parameter matrix $\boldsymbol{\varphi}(\Delta\widetilde{\mathbf{p}})$ is invertible)

$$\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})=\boldsymbol{\Phi}(\widetilde{\mathbf{P}})\boldsymbol{\varphi}(\Delta\widetilde{\mathbf{p}})^{-1},\ \text{ so }\ \mathbf{x}=\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})\mathbf{X}.\tag{16b}$$

As the inverted incremental parameter matrix $\boldsymbol{\varphi}(\Delta\widetilde{\mathbf{p}})^{-1}$ appears in Equation (16b), the algorithm defined in Equation (15) is called *inverse* compositional by Baker and Matthews [15]. It is noted that the image $\widetilde{g}(\widetilde{\mathbf{X}})=g\left(\widetilde{\mathbf{X}}+\widetilde{\mathbf{U}}\left(\widetilde{\mathbf{X}};\widetilde{\mathbf{P}}\right)\right)$ in Equation (15) should be regarded as an

doi: 10.1111/str.12039

image numerically restored from the current image towards the intermediate image frame with coordinates $\widetilde{\mathbf{X}}$, and the image $G\left(\widetilde{\mathbf{X}}+\Delta\widetilde{\mathbf{u}}\right)$ should be regarded as an image numerically distorted from the initial image towards the same intermediate image frame. The forward affine parameter matrix between the initial image and the current image for the inverse compositional DIC algorithm should be updated accordingly as

$$\begin{aligned}\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P}) &= \boldsymbol{\Phi}\left(\widetilde{\mathbf{P}}\right)\boldsymbol{\varphi}(\Delta\widetilde{\mathbf{p}})^{-1}\\ &= \begin{pmatrix} 1+\widetilde{U}_X & \widetilde{U}_Y & \widetilde{U}_0\\ \widetilde{V}_X & 1+\widetilde{V}_Y & \widetilde{V}_0\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix} 1+\Delta\widetilde{u}_X & \Delta\widetilde{u}_Y & \Delta\widetilde{u}_0\\ \Delta\widetilde{v}_X & 1+\Delta\widetilde{v}_Y & \Delta\widetilde{v}_0\\ 0 & 0 & 1\end{pmatrix}^{-1}.\end{aligned} \tag{17}$$

It is noted that the inverted incremental affine parameter matrix in Equation (17) can be obtained readily according to Equation (6b) for physically admissible affine deformation.

The backward version of the inverse compositional algorithm can also be formulated, and its SSD coefficient is given as

$$\begin{aligned}C_{BIC}\left(\Delta\widetilde{\mathbf{P}}\right) &= \sum_{i=1}^{n}\left[g\left(\widetilde{\mathbf{x}}_i+\Delta\widetilde{\mathbf{U}}\right)-\widetilde{\mathbf{G}}(\widetilde{\mathbf{x}}_i)\right]^2\\ &= \sum_{i=1}^{n}\left[g\left(\widetilde{\mathbf{x}}_i+\Delta\widetilde{\mathbf{U}}\left(\widetilde{\mathbf{x}}_i;\Delta\widetilde{\mathbf{P}}\right)\right)-G(\widetilde{\mathbf{x}}_i+\widetilde{\mathbf{u}}(\widetilde{\mathbf{x}}_i;\widetilde{\mathbf{p}}))\right]^2\end{aligned} \tag{18}$$

where

$$\begin{aligned}&\mathbf{X}=\widetilde{\mathbf{x}}+\widetilde{\mathbf{u}}(\widetilde{\mathbf{x}};\widetilde{\mathbf{p}})=\boldsymbol{\varphi}(\widetilde{\mathbf{p}})\widetilde{\mathbf{x}}, \quad \text{and}\\ &\mathbf{x}=\widetilde{\mathbf{x}}+\Delta\widetilde{\mathbf{U}}\left(\widetilde{\mathbf{x}};\Delta\widetilde{\mathbf{P}}\right)=\boldsymbol{\Phi}\left(\Delta\widetilde{\mathbf{P}}\right)\widetilde{\mathbf{x}},\end{aligned} \tag{19a}$$

and the backward deformation parameter matrix is updated as follows (assuming the incremental parameter matrix $\boldsymbol{\Phi}\left(\Delta\widetilde{\mathbf{P}}\right)$ is invertible)

$$\boldsymbol{\varphi}(\mathbf{p}+\Delta\mathbf{p})=\boldsymbol{\varphi}(\widetilde{\mathbf{p}})\boldsymbol{\Phi}\left(\Delta\widetilde{\mathbf{P}}\right)^{-1}, \quad \text{so} \quad \mathbf{X}=\boldsymbol{\varphi}(\mathbf{p}+\Delta\mathbf{p})\mathbf{x}. \tag{19b}$$

It is noted that the image $\widetilde{G}(\widetilde{\mathbf{x}})=G(\widetilde{\mathbf{x}}+\widetilde{\mathbf{u}}(\widetilde{\mathbf{x}};\widetilde{\mathbf{p}}))$ in Equation (18) should be regarded as an image numerically distorted towards the intermediate image frame $\widetilde{\mathbf{x}}$ from the initial image with the prior estimated backward displacements $\widetilde{\mathbf{u}}(\widetilde{\mathbf{x}};\widetilde{\mathbf{p}})$, and the image $g\left(\widetilde{\mathbf{x}}+\Delta\widetilde{\mathbf{U}}\left(\widetilde{\mathbf{x}};\Delta\widetilde{\mathbf{P}}\right)\right)$ is an image numerically restored towards the same intermediate image frame from the current image with forward displacement increments $\Delta\widetilde{\mathbf{U}}\left(\widetilde{\mathbf{x}};\Delta\widetilde{\mathbf{P}}\right)$. At the end of an iterative solution process using the inverse compositional algorithms, one has $\widetilde{\mathbf{X}}\to\mathbf{X}$ (integral-pixel points used in the intermediate image eventually coincide with those in the initial image within the limit of residual displacement errors) and $\widetilde{\mathbf{x}}\to\mathbf{x}$ (integral-pixel points used in the intermediate image eventually coincide with those in the current image within the limit of residual displacement errors), respectively. The intermediate image either $\widetilde{g}\left(\widetilde{\mathbf{X}}\right)$ in Equation (15) or $\widetilde{G}(\widetilde{\mathbf{x}})$ in Equation (18) is not fixed with respect to either the initial or current image during the iterative numerical solution process. The inverse compositional algorithms are thus different from the previous additive or compositional DIC algorithms given in Equations (7), (8), (10) and (13), where the fixed and integer-valued coordinates in either initial or current images are used for each point of the image subset.

Consistent digital image correlation algorithms

The so-called *consistent* (or inverse consistent) DIC algorithms appeared in the literature [14, 16, 20] can be considered as a joint forward and backward DIC analysis (Table 2). The SSD coefficient for a consistent *additive* DIC algorithm may be given by combining Equations (7) and (8) as

$$\begin{aligned}C_{CA}(\Delta\mathbf{P};\Delta\mathbf{p}) &= \sum_{i=1}^{n}\left[g(\mathbf{X}_i+\mathbf{U}(\mathbf{X}_i;\mathbf{P}+\Delta\mathbf{P}))-G(\mathbf{X}_i)\right]^2\\ &+\sum_{i=1}^{n}\left[G(\mathbf{x}_i+\mathbf{u}(\mathbf{x}_i;\mathbf{p}+\Delta\mathbf{p}))-g(\mathbf{x}_i)\right]^2,\end{aligned} \tag{20}$$

where a consistency condition is imposed for the invertible forward and backward deformation parameter matrices as ($\mathbf{I}$ is the identity or unit matrix)

$$\begin{aligned}&\boldsymbol{\varphi}(\mathbf{p})\boldsymbol{\Phi}(\mathbf{P})=\mathbf{I}, \quad \boldsymbol{\varphi}(\mathbf{p}+\Delta\mathbf{p})\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})=\mathbf{I}, \quad \text{or}\\ &\boldsymbol{\varphi}(\mathbf{p})=\boldsymbol{\Phi}(\mathbf{P})^{-1}, \quad \boldsymbol{\varphi}(\mathbf{p}+\Delta\mathbf{p})=\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})^{-1},\end{aligned} \tag{21a}$$

where

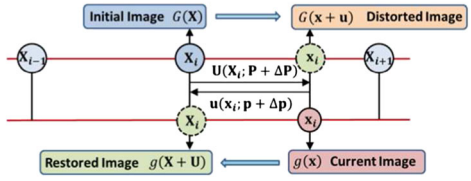
$$\begin{aligned}&\mathbf{x}=\mathbf{X}+\mathbf{U}(\mathbf{X};\mathbf{P}+\Delta\mathbf{P})=\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})\mathbf{X}, \quad \text{and}\\ &\mathbf{X}=\mathbf{x}+\mathbf{u}(\mathbf{x};\mathbf{p}+\Delta\mathbf{p})=\boldsymbol{\varphi}(\mathbf{p}+\Delta\mathbf{p})\mathbf{x}.\end{aligned} \tag{21b}$$

Parameters $\mathbf{p}$ and $\Delta\mathbf{p}$ of the backward displacements in Equation (20) are not independent but directly related to parameters $\mathbf{P}$ and $\Delta\mathbf{P}$ of the forward displacements via Equation (21a), that is $C_{CA}(\Delta\mathbf{P};\Delta\mathbf{p}(\Delta\mathbf{P}))\equiv C_{CA}(\Delta\mathbf{P})$. This notation for the SSD coefficients will be used for both consistent and symmetric algorithms in the following. For rigid-body translations, $\mathbf{p}=-\mathbf{P}$ and $\Delta\mathbf{p}=-\Delta\mathbf{P}$. For an invertible affine deformation, one can readily obtain $\mathbf{p}=(u_0,v_0,u_x,u_y,v_x,v_y)^T$ from $\mathbf{P}=(U_0,V_0,U_X,U_Y,V_X,V_Y)^T$ according to Equation (6b) as

© 2013 Wiley Publishing Ltd | *Strain* (2013) **49**, 313–334
doi: 10.1111/str.12039

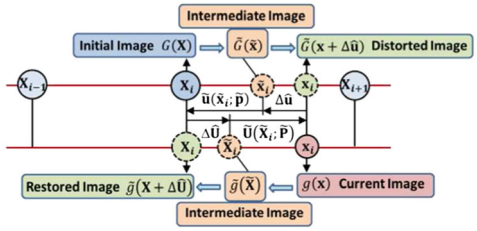
Table 2: Summary of consistent or symmetric additive and compositional DIC algorithms

Consistent Additive DIC



$$C_{CA}=\sum_{i=1}^{n}[g(\mathbf{X}_i+\mathbf{U})-G(\mathbf{X}_i)]^2+\sum_{i=1}^{n}[G(\mathbf{x}_i+\mathbf{u})-g(\mathbf{x}_i)]^2$$

Consistent Compositional DIC



$$C_{CC}=\sum_{i=1}^{n}\left[\tilde{g}(\mathbf{X}_i+\Delta\hat{\mathbf{U}})-G(\mathbf{X}_i)\right]^2+\sum_{i=1}^{n}\left[\tilde{G}(\mathbf{x}_i+\Delta\hat{\mathbf{u}})-g(\mathbf{x}_i)\right]^2$$

***Consistent Inverse Compositional DIC*⇒**

Joint Forward&Backward Inverse Compositional DIC

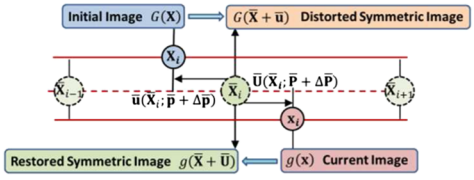
$$C_{CIC}=\sum_{i=1}^{n}\left[G\left(\tilde{\mathbf{X}}_i+\Delta\hat{\mathbf{u}}\right)-g\left(\tilde{\mathbf{X}}_i+\tilde{\mathbf{U}}\right)\right]^2+\sum_{i=1}^{n}[g(\tilde{\mathbf{x}}_i+\Delta\hat{\mathbf{U}})-G(\tilde{\mathbf{x}}_i+\tilde{\mathbf{u}})]^2$$

***Consistent Mixed Compositional DIC*⇒**

Joint Forward and Inverse Compositional DIC

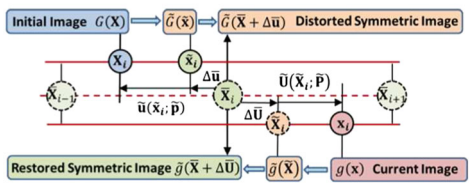
$$C_{CMC}=\sum_{i=1}^{n}\left[G\left(\tilde{\mathbf{X}}_i+\Delta\tilde{\mathbf{u}}\right)-g\left(\tilde{\mathbf{X}}_i+\mathbf{U}\right)\right]^2+\sum_{i=1}^{n}[\tilde{g}(\mathbf{X}_i+\Delta\hat{\mathbf{U}})-G(\mathbf{X}_i)]^2$$

Symmetric Additive DIC



$$C_{SA}=\sum_{i=1}^{n}\left[g(\overline{\mathbf{X}}_i+\overline{\mathbf{U}}(\overline{\mathbf{X}}_i;\overline{\mathbf{P}}+\Delta\overline{\mathbf{P}}))-G(\overline{\mathbf{X}}_i+\overline{\mathbf{u}}(\overline{\mathbf{X}}_i;\overline{\mathbf{p}}+\Delta\overline{\mathbf{p}}))\right]^2$$

Symmetric Compositional DIC



$$C_{SC}=\sum_{i=1}^{n}\left[\tilde{G}(\overline{\mathbf{X}}+\Delta\overline{\mathbf{u}}(\overline{\mathbf{X}};\Delta\hat{\mathbf{p}}))-\tilde{g}(\overline{\mathbf{X}}+\Delta\overline{\mathbf{U}}(\overline{\mathbf{X}};\Delta\hat{\mathbf{P}}))\right]^2$$

DIC, digital image correlation.

Letters with a subscripted index '*i*' enclosed in a circle are pixel coordinates of points in various images as indicated by a thin vertical line. Coordinates having indexes of '$i-1$', 'i' and '$i+1$' are points of the frame of reference with integral pixel coordinates used in an analysis. A thick vertical line with arrows at both ends indicates the pixel point on the frame of the reference where the image correlation analysis is being carried out. For consistent algorithms, the analysis is carried out at two different pixel points simultaneously (only one of them has integral-pixel coordinates though).

An arrow with a single thick horizontal line indicates the displacement vector direction: a forward displacement with the arrow pointing to the right and a backward displacement with the arrow pointing to the left. Upper or lower case letters are, respectively, used to describe the forward or backward displacements and their parameters. An arrow with two thin horizontal lines indicates the numerically distorted (the arrow pointing to the right) or restored (the arrow pointing to the left) images from the source to target images. The displacements are, in general, non-integral in pixels.

Illustrations for the consistent inverse and mixed compositional algorithms are omitted as they are basically the combination of either forward and backward inverse compositional algorithms or forward compositional and backward inverse compositional algorithms shown in Table 1.

$$\begin{pmatrix} u_x & u_y & u_0 \\ v_x & v_y & v_0 \end{pmatrix}=\frac{1}{(1+U_X)(1+V_Y)-U_YV_X}\begin{pmatrix} 1+V_Y & -U_Y & U_YV_0-V_YU_0-U_0 \\ -V_X & 1+U_X & V_XU_0-U_XV_0-V_0 \end{pmatrix}. \tag{22}$$

The SSD coefficient for a consistent *compositional* DIC algorithm may be given by combining Equations (10) and (13)

$$C_{CC}(\Delta\hat{\mathbf{P}};\Delta\hat{\mathbf{p}})=\sum_{i=1}^{n}\left[\tilde{g}(\mathbf{X}_i+\Delta\hat{\mathbf{U}}(\mathbf{X}_i;\Delta\hat{\mathbf{P}}))-G(\mathbf{X}_i)\right]^2+\sum_{i=1}^{n}\left[\tilde{G}(\mathbf{x}_i+\Delta\hat{\mathbf{u}}(\mathbf{x}_i;\Delta\hat{\mathbf{p}}))-g(\mathbf{x}_i)\right]^2, \tag{23}$$

where the intermediate images $\tilde{g}(\tilde{\mathbf{X}})=g(\tilde{\mathbf{X}}_i+\tilde{\mathbf{U}}(\tilde{\mathbf{X}}_i;\tilde{\mathbf{P}}))$ and $\tilde{G}(\tilde{\mathbf{x}})=G(\tilde{\mathbf{x}}_i+\tilde{\mathbf{u}}(\tilde{\mathbf{x}}_i;\tilde{\mathbf{p}}))$ are numerically restored and distorted images based on the prior estimated forward and backward displacements, respectively, see Equations (11a), (11b), (14a) and (14b) about details of their definitions and compositional updating. To make this algorithm consistent, deformation parameter matrices and their increments are again assumed to be invertible and

doi: 10.1111/str.12039

$$\boldsymbol{\varphi}(\mathbf{p}+\Delta\mathbf{p})\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})=\mathbf{I},$$
$$\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})=\boldsymbol{\Phi}(\tilde{\mathbf{P}})\boldsymbol{\Phi}\left(\Delta\hat{\mathbf{P}}\right),\quad \boldsymbol{\varphi}(\mathbf{p}+\Delta\mathbf{p})=\boldsymbol{\varphi}(\tilde{\mathbf{p}})\boldsymbol{\varphi}(\Delta\hat{\mathbf{p}}), \tag{24a}$$

$$\boldsymbol{\varphi}(\tilde{\mathbf{p}})=\boldsymbol{\Phi}\left(\tilde{\mathbf{P}}\right)^{-1},\ \text{and}\ \ \boldsymbol{\varphi}(\Delta\hat{\mathbf{p}})=\boldsymbol{\Phi}\left(\tilde{\mathbf{P}}\right)\boldsymbol{\Phi}\left(\Delta\hat{\mathbf{P}}\right)^{-1}\boldsymbol{\Phi}\left(\tilde{\mathbf{P}}\right)^{-1}. \tag{24b}$$

The SSD coefficient for a consistent *inverse* compositional DIC algorithm may be given by combining both forward (Equation 15) and backward (Equation 18) inverse compositional algorithms

$$C_{CIC}\left(\Delta\hat{\mathbf{P}};\Delta\hat{\mathbf{p}}\right)=\sum_{i=1}^{n}\left[G\left(\tilde{\mathbf{X}}_i+\Delta\,\hat{\mathbf{u}}\left(\tilde{\mathbf{X}}_i;\Delta\hat{\mathbf{p}}\right)\right)-\mathbf{g}\left(\tilde{\mathbf{X}}_i+\tilde{\mathbf{U}}\right)\right]^2 +\sum_{i=1}^{n}\left[g\left(\tilde{\mathbf{x}}_i+\Delta\,\hat{\mathbf{U}}\left(\tilde{\mathbf{x}}_i;\Delta\hat{\mathbf{P}}\right)\right)-G(\tilde{\mathbf{x}}_i+\tilde{\mathbf{u}})\right]^2, \tag{25}$$

where $\tilde{\mathbf{X}}$ are coordinates of the intermediate image in the forward analysis (first term) and $\tilde{\mathbf{x}}$ are coordinates of the intermediate image in the backward analysis (the second term). However, the incremental displacements $\Delta\hat{\mathbf{u}}\left(\hat{\mathbf{X}};\Delta\hat{\mathbf{p}}\right)$ and $\Delta\hat{\mathbf{U}}\left(\tilde{\mathbf{x}};\Delta\hat{\mathbf{P}}\right)$ in each analysis are in the opposite directions to $\tilde{\mathbf{U}}\left(\tilde{\mathbf{X}};\tilde{\mathbf{P}}\right)$ and $\tilde{\mathbf{u}}(\tilde{\mathbf{x}};\tilde{\mathbf{p}})$, respectively. The consistent condition should be imposed on both forward and backward deformation parameter matrices after compositional updating, namely (see Equations (16b) and (19b))

$$\begin{aligned}&\boldsymbol{\varphi}(\mathbf{p}+\Delta\mathbf{p})\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})=\mathbf{I},\\&\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})=\boldsymbol{\Phi}(\tilde{\mathbf{P}})\boldsymbol{\varphi}(\Delta\hat{\mathbf{p}})^{-1},\\&\boldsymbol{\varphi}(\mathbf{p}+\Delta\mathbf{p})=\boldsymbol{\varphi}(\tilde{\mathbf{p}})\boldsymbol{\Phi}\left(\Delta\hat{\mathbf{P}}\right)^{-1},\end{aligned} \tag{26a}$$

so $\tilde{\mathbf{p}}$ and $\Delta\hat{\mathbf{P}}$ are related to $\tilde{\mathbf{P}}$ and $\Delta\hat{\mathbf{p}}$ via

$$\boldsymbol{\varphi}(\tilde{\mathbf{p}})=\boldsymbol{\Phi}\left(\tilde{\mathbf{P}}\right)^{-1},\quad \text{or}\ \ \boldsymbol{\Phi}\left(\Delta\hat{\mathbf{P}}\right)=\boldsymbol{\Phi}\left(\tilde{\mathbf{P}}\right)\boldsymbol{\varphi}(\Delta\hat{\mathbf{p}})^{-1}\boldsymbol{\Phi}\left(\tilde{\mathbf{P}}\right)^{-1}. \tag{26b}$$

Finally, a consistent *mixed* compositional DIC algorithm may also be considered and its SSD coefficient is given by combining both forward compositional (Equation 10) and forward inverse compositional (Equation 18) algorithms

$$C_{CMC}\left(\Delta\hat{\mathbf{P}};\Delta\hat{\mathbf{p}}\right)=\sum_{i=1}^{n}\left[\tilde{g}\left(\mathbf{X}_i+\Delta\hat{\mathbf{U}}\left(\mathbf{X}_i;\Delta\hat{\mathbf{P}}\right)\right)-G(\mathbf{X}_i)\right]^2 +\sum_{i=1}^{n}\left[G\left(\tilde{\mathbf{X}}_i+\Delta\tilde{\mathbf{u}}\left(\tilde{\mathbf{X}}_i;\Delta\hat{\mathbf{p}}\right)\right)-g\left(\tilde{\mathbf{X}}_i+\tilde{\mathbf{U}}\right)\right]^2, \tag{27}$$

with a consistent condition imposed on both forward deformation parameter matrices after compositional and inverse compositional updating as

$$\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})=\boldsymbol{\Phi}\left(\tilde{\mathbf{P}}\right)\boldsymbol{\varphi}(\Delta\hat{\mathbf{p}})^{-1}=\boldsymbol{\Phi}\left(\tilde{\mathbf{P}}\right)\boldsymbol{\Phi}\left(\Delta\hat{\mathbf{P}}\right), \tag{28a}$$

where $\tilde{\mathbf{P}}$ is the same for both terms in Equation (27) and $\Delta\hat{\mathbf{p}}$ is related to $\Delta\hat{\mathbf{P}}$ via

$$\boldsymbol{\varphi}(\Delta\hat{\mathbf{p}})=\boldsymbol{\Phi}\left(\Delta\hat{\mathbf{P}}\right)^{-1}. \tag{28b}$$

Symmetric digital image correlation algorithms

The so-called *symmetric* image correlation algorithms have also been proposed in the literature [13, 16, 30]. The SSD coefficient for a symmetric additive DIC algorithm has the following form

$$C_{SA}\left(\Delta\overline{\mathbf{P}};\Delta\overline{\mathbf{p}}\right)=\sum_{i=1}^{n}\left[g\left(\overline{\mathbf{X}}_i+\overline{\mathbf{U}}\left(\overline{\mathbf{X}}_i;\overline{\mathbf{P}}+\Delta\overline{\mathbf{P}}\right)\right)-G\left(\overline{\mathbf{X}}_i+\overline{\mathbf{u}}\left(\overline{\mathbf{X}}_i;\overline{\mathbf{p}}+\Delta\overline{\mathbf{p}}\right)\right)\right]^2, \tag{29}$$

where $g\left(\overline{\mathbf{X}}+\overline{\mathbf{U}}\right)$ is a numerically restored image from the current image, $G\left(\overline{\mathbf{X}}+\overline{\mathbf{u}}\right)$ is a numerically distorted image from the initial image and $\overline{\mathbf{X}}$ are integer-valued coordinates associated with an intermediate image (see Table 2). In addition, the following conditions are imposed on the relative forward and backward deformation parameter matrices as

$$\boldsymbol{\varphi}(\overline{\mathbf{p}})\boldsymbol{\Phi}\left(\overline{\mathbf{P}}\right)=\mathbf{I},\quad \boldsymbol{\varphi}(\overline{\mathbf{p}}+\Delta\overline{\mathbf{p}})\boldsymbol{\Phi}\left(\overline{\mathbf{P}}+\Delta\overline{\mathbf{P}}\right)=\mathbf{I},\quad \text{or} \tag{30a}$$
$$\boldsymbol{\varphi}(\overline{\mathbf{p}})=\boldsymbol{\Phi}\left(\overline{\mathbf{P}}\right)^{-1},\quad \boldsymbol{\varphi}(\overline{\mathbf{p}}+\Delta\overline{\mathbf{p}})=\boldsymbol{\Phi}\left(\overline{\mathbf{P}}+\Delta\overline{\mathbf{P}}\right)^{-1},$$

where

$$\mathbf{x}=\overline{\mathbf{X}}+\overline{\mathbf{U}}\left(\overline{\mathbf{X}};\overline{\mathbf{P}}+\Delta\overline{\mathbf{P}}\right)=\boldsymbol{\Phi}\left(\overline{\mathbf{P}}+\Delta\overline{\mathbf{P}}\right)\overline{\mathbf{X}}, \tag{30b}$$
$$\mathbf{X}=\overline{\mathbf{X}}+\overline{\mathbf{u}}\left(\overline{\mathbf{X}};\overline{\mathbf{p}}+\Delta\overline{\mathbf{p}}\right)=\boldsymbol{\varphi}(\overline{\mathbf{p}}+\Delta\overline{\mathbf{p}})\overline{\mathbf{X}},$$

$$\begin{aligned}\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})&=\boldsymbol{\Phi}\left(\overline{\mathbf{P}}+\Delta\overline{\mathbf{P}}\right)\boldsymbol{\varphi}(\overline{\mathbf{p}}+\Delta\overline{\mathbf{p}})^{-1}\\&=\boldsymbol{\Phi}\left(\overline{\mathbf{P}}+\Delta\overline{\mathbf{P}}\ \right)^2,\ \ \text{so}\ \ \mathbf{x}=\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})\mathbf{X}.\end{aligned} \tag{30c}$$

Images $g\left(\overline{\mathbf{X}}+\overline{\mathbf{U}}\right)$ and $G\left(\overline{\mathbf{X}}+\overline{\mathbf{u}}\right)$ in the analysis are thus exactly 'half-way' between the initial and current images, and the coordinates $\overline{\mathbf{X}}$ belong to a special fixed intermediate image frame, namely, the symmetric image frame. Parameters $\overline{\mathbf{p}}$ and $\Delta\overline{\mathbf{p}}$ of the backward displacement field in Equation (29) are again not independent but directly related to parameters $\overline{\mathbf{P}}$ and $\Delta\overline{\mathbf{P}}$ of the forward displacement field via Equation (30a). For rigid-body translations, $\overline{\mathbf{p}}=-\overline{\mathbf{P}}$ and $\Delta\overline{\mathbf{p}}=-\Delta\overline{\mathbf{P}}$, so $\mathbf{P}=2\overline{\mathbf{P}}$ and $\Delta\mathbf{P}=2\Delta\overline{\mathbf{P}}$. According to Equation (22), one can analytically obtain parameters of the backward affine deformation $\overline{\mathbf{p}}=\left(\overline{u}_0,\overline{v}_0,\overline{u}_x,\overline{u}_y,\overline{v}_x,\overline{v}_y\right)^T$ from the parameters of the forward affine deformation $\overline{\mathbf{P}}=\left(\overline{U}_0,\overline{V}_0,\overline{U}_X,\overline{U}_Y,\overline{V}_X,\overline{V}_Y\right)^T$. It is noted that even though the deformation parameters are updated *additively* in the iterative solution process per Equation (30b), the resulting forward transformation $\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})$ between the initial and current images is the composition of two successive (and identical) warps as per Equation (30c). Furthermore, such a deformation parameter matrix is now applied over a set of non-integral pixel points $\mathbf{X}$ of the initial image (as the coordinates of the corresponding

© 2013 Wiley Publishing Ltd | *Strain* (2013) **49**, 313–334
doi: 10.1111/str.12039

points $\overline{\mathbf{X}}$ of the intermediate image are integer-valued in the analysis, see Equations 29 and 30b).

Similarly, the SSD coefficient for a symmetric compositional DIC algorithm has the following form

$$C_{SC}(\Delta\hat{\mathbf{P}};\Delta\hat{\mathbf{p}}) = \sum_{i=1}^{n}\left[\widetilde{\mathbf{G}}(\overline{\mathbf{X}}+\Delta\hat{\mathbf{u}}(\overline{\mathbf{X}};\Delta\hat{\mathbf{p}})) - \widetilde{\mathbf{g}}(\overline{\mathbf{X}}+\Delta\hat{\mathbf{U}}(\overline{\mathbf{X}};\Delta\hat{\mathbf{P}}))\right]^2, \tag{31}$$

where $\widetilde{G}(\overline{\mathbf{X}}+\Delta\hat{\mathbf{u}})$ is a numerically distorted image from one intermediate image $\widetilde{G}(\widetilde{\mathbf{x}})$ towards the symmetric image frame with the coordinates $\overline{\mathbf{X}}$ and $\widetilde{g}(\overline{\mathbf{X}}+\Delta\hat{\mathbf{U}})$ is a numerically restored image from another intermediate image $\widetilde{g}(\tilde{\mathbf{X}})$ towards the same symmetric image frame. Furthermore, the intermediate image $\widetilde{G}(\widetilde{\mathbf{x}}) = G(\widetilde{\mathbf{x}}+\tilde{\mathbf{u}}(\widetilde{\mathbf{x}};\widetilde{\mathbf{p}}))$ is numerically distorted from the initial image $G(\mathbf{X})$ with displacements $\tilde{\mathbf{u}}(\widetilde{\mathbf{x}};\widetilde{\mathbf{p}})$, and the intermediate image $\widetilde{g}(\tilde{\mathbf{X}}) = g(\tilde{\mathbf{X}}+\tilde{\mathbf{U}}(\tilde{\mathbf{X}};\tilde{\mathbf{P}}))$ is numerically restored from the current image $g(\mathbf{x})$ with displacements $\tilde{\mathbf{U}}(\tilde{\mathbf{X}};\tilde{\mathbf{P}})$ (see Table 2). The relative displacements among these six images are summarised in the following:

$$\begin{aligned}&\mathbf{x} = \tilde{\mathbf{X}}+\tilde{\mathbf{U}}(\tilde{\mathbf{X}};\tilde{\mathbf{P}}) = \boldsymbol{\Phi}(\tilde{\mathbf{P}})\tilde{\mathbf{X}},\\ &\tilde{\mathbf{X}} = \overline{\mathbf{X}}+\Delta\hat{\mathbf{U}}(\overline{\mathbf{X}};\Delta\hat{\mathbf{P}}) = \boldsymbol{\Phi}(\Delta\hat{\mathbf{P}})\overline{\mathbf{X}},\end{aligned} \tag{32a}$$

$$\mathbf{x} = \boldsymbol{\Phi}(\overline{\mathbf{P}}+\Delta\overline{\mathbf{P}})\overline{\mathbf{X}}, \quad \boldsymbol{\Phi}(\overline{\mathbf{P}}+\Delta\overline{\mathbf{P}}) = \boldsymbol{\Phi}(\tilde{\mathbf{P}})\boldsymbol{\Phi}(\Delta\hat{\mathbf{P}}), \tag{32b}$$

$$\begin{aligned}&\mathbf{X} = \widetilde{\mathbf{x}}+\tilde{\mathbf{u}}(\widetilde{\mathbf{x}};\widetilde{\mathbf{p}}) = \boldsymbol{\varphi}(\widetilde{\mathbf{p}})\widetilde{\mathbf{x}},\\ &\widetilde{\mathbf{x}} = \overline{\mathbf{X}}+\Delta\hat{\mathbf{u}}(\overline{\mathbf{X}};\Delta\hat{\mathbf{p}}) = \boldsymbol{\varphi}(\Delta\hat{\mathbf{p}})\widetilde{\mathbf{X}},\end{aligned} \tag{32c}$$

$$\mathbf{X} = \boldsymbol{\varphi}(\overline{\mathbf{p}}+\Delta\overline{\mathbf{p}})\widetilde{\mathbf{X}}, \quad \boldsymbol{\varphi}(\overline{\mathbf{p}}+\Delta\overline{\mathbf{p}}) = \boldsymbol{\varphi}(\widetilde{\mathbf{p}})\boldsymbol{\varphi}(\Delta\hat{\mathbf{p}}). \tag{32d}$$

Again, the symmetric condition similar to Equation (30a) is imposed on $\boldsymbol{\Phi}(\overline{\mathbf{P}}+\Delta\overline{\mathbf{P}})$ and $\boldsymbol{\varphi}(\overline{\mathbf{p}}+\Delta\overline{\mathbf{p}})$, and the forward deformation parameter matrix $\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})$ between the initial and current images is computed per Equation (30c). Consequently, parameters $\tilde{\mathbf{p}}$ and $\Delta\hat{\mathbf{p}}$ of the backward displacements and their increments are again not independent but directly related to parameters $\tilde{\mathbf{P}}$ and $\Delta\hat{\mathbf{P}}$ of the forward displacements and their increments as

$$\boldsymbol{\varphi}(\hat{\mathbf{p}}) = \boldsymbol{\Phi}(\tilde{\mathbf{P}})^{-1}, \quad \boldsymbol{\varphi}(\Delta\hat{\mathbf{p}}) = \boldsymbol{\Phi}(\tilde{\mathbf{P}})\boldsymbol{\Phi}(\Delta\hat{\mathbf{P}})^{-1}\boldsymbol{\Phi}(\tilde{\mathbf{P}})^{-1}. \tag{33}$$

Computational Implementation of Digital Image Correlation Algorithms: An Example

As the formulation of various DIC algorithms is presented for the first time using the new but very concise notations in this paper, we show in this section how some of these algorithms may be implemented computationally by specifying a particular *numerical method* (Gauss–Newton), *subset deformation model* (1D and 2D affine), *spatial gradient* computation routine (central difference), and *subpixel interpolation* scheme (linear). Just like there are many types of DIC algorithms available, there are many choices for each of these aspects in a DIC analysis as well depending on one's requirement for efficiency, robustness and accuracy. Although a detailed comparative evaluation of their relative merits is beyond the scope of this paper, it is hoped that the examples shown in this section can provide as a starting point for anyone interested in carrying out such a study in the future.

Numerical solutions using the Gauss–Newton gradient decent method

The correlation coefficients of Lucas–Kanade DIC algorithms described above all have a similar form in terms of the SSD in greyscales between two digital images at the same frame of reference. The displacement parameters appeared in these coefficients can be solved iteratively by a nonlinear gradient-decent method for minimising these SSD coefficients. In this section, the so-called Gauss–Newton or quasi-Newton method will be applied to the additive and compositional DIC algorithms listed in Table 1; as it is shown, such a method has an overall excellent performance [15]. For the sake of brevity, detailed derivations of the closed form solutions will be provided only for the forward versions of the iterative algorithms list in Table 1. Results for the backward versions as well as for consistent and symmetric versions can be obtained in a similar way.

Using a first-order Taylor expansion of the numerically restored image $g(\mathbf{X}+\mathbf{U}(\mathbf{X};\mathbf{P}+\Delta\mathbf{P}))$ in the forward additive SSD coefficient of Equation (7) about the parameter increments $\Delta\mathbf{P}$, one has

$$\begin{aligned}g(\mathbf{X}+\mathbf{U}(\mathbf{X};\mathbf{P}+\Delta\mathbf{P})) &\approx g(\mathbf{X}+\mathbf{U}(\mathbf{X};\mathbf{P}))\\ &\quad+\frac{\partial g(\mathbf{X}+\mathbf{U}(\mathbf{X};\mathbf{P}))}{\partial\mathbf{P}}\Delta\mathbf{P}\\ &= g(\mathbf{X}+\mathbf{U}(\mathbf{X};\mathbf{P}))+\mathbf{J}(\mathbf{X};\mathbf{P})\Delta\mathbf{P},\end{aligned} \tag{34a}$$

where the Jacobian vector $\mathbf{J}(\mathbf{X};\mathbf{P})$ at each point of the image subset is computed at $\Delta\mathbf{P}\approx 0$ from the current image $g(\mathbf{x}) = g(x,y)$ using the chain rule as

$$\begin{aligned}\mathbf{J}(\mathbf{X};\mathbf{P}) &\equiv \frac{\partial g}{\partial\mathbf{P}} = \frac{\partial g}{\partial\mathbf{x}}\frac{\partial\mathbf{x}(\mathbf{X};\mathbf{P})}{\partial\mathbf{P}} = \frac{\partial g}{\partial\mathbf{x}}\frac{\partial\mathbf{U}(\mathbf{X};\mathbf{P})}{\partial\mathbf{P}}\\ &= g_{\mathbf{x}}(\mathbf{X}+\mathbf{U})\mathbf{U}_{\mathbf{P}}(\mathbf{X};\mathbf{P}), \quad g_{\mathbf{x}}(\mathbf{X}+\mathbf{U}) = (g_x, g_y).\end{aligned} \tag{34b}$$

As an example, $\mathbf{P} = (U_0, V_0, U_X, U_Y, V_X, V_Y)^T$ and $\Delta\mathbf{P} = (\Delta U_0, \Delta V_0, \Delta U_X, \Delta U_Y, \Delta V_X, \Delta V_Y)^T$ for the forward affine deformation as given by Equation (4). So gradients of the forward displacement function $\mathbf{U}_{\mathbf{P}}(\mathbf{X};\mathbf{P})$ and the components of the corresponding Jacobian vector $\mathbf{J}(\mathbf{X};\mathbf{P})$ are respectively

doi: 10.1111/str.12039

$$\mathbf{U_P}(\mathbf{X};\mathbf{P}) = \mathbf{U_P}(\mathbf{X}) = \begin{pmatrix} 1 & 0 & X & 0 & Y & 0 \\ 0 & 1 & 0 & X & 0 & Y \end{pmatrix}, \tag{35a}$$

$$\begin{aligned}\mathbf{J}(\mathbf{X};\mathbf{P}) &= g_{\mathbf{x}}(\mathbf{X}+\mathbf{U})\mathbf{U_P}(\mathbf{X};\mathbf{P}) \\ &= (g_x, g_y)\begin{pmatrix} 1 & 0 & X & 0 & Y & 0 \\ 0 & 1 & 0 & X & 0 & Y \end{pmatrix} \\ &= (g_x, g_y, Xg_x, Xg_y, Yg_x, Yg_y).\end{aligned} \tag{35b}$$

Minimization of the approximate forward additive SSD coefficient (differentiating the SSD coefficient with respect to $\Delta\mathbf{P}$ and setting the result to zero) leads to a closed form solution:

$$\begin{aligned}&\frac{\partial C_{FA}(\Delta\mathbf{P})}{\partial(\Delta\mathbf{P})} = 0: \\ &\Delta\mathbf{P} = \mathbf{H}^{-1}\sum_{i=1}^{n}[G(\mathbf{X}_i) - g(\mathbf{X}_i + \mathbf{U}(\mathbf{X}_i;\mathbf{P}))]\mathbf{J}(\mathbf{X}_i;\mathbf{P})^T, \\ &\mathbf{H}(\mathbf{P}) = \sum_{i=1}^{n}\mathbf{J}(\mathbf{X}_i;\mathbf{P})^T\mathbf{J}(\mathbf{X};\mathbf{P}),\end{aligned} \tag{36}$$

where $\mathbf{H}(\mathbf{P})$ is the Hessian matrix depending on the prior estimated displacement parameters $\mathbf{P}$. Both greyscales of the restored image $g(\mathbf{X}+\mathbf{U})$ and their gradients $g_{\mathbf{x}}(\mathbf{X}+\mathbf{U})$ in Equations (35b) and (36) have to be evaluated at the non-integral pixel locations $\mathbf{X}+\mathbf{U}$ by a bilinear or other higher-order interpolation scheme.

Similarly, a first-order Taylor expansion of the numerically restored image $\widetilde{g}\big(\mathbf{X}+\Delta\hat{\mathbf{U}}(\mathbf{X};\Delta\hat{\mathbf{P}})\big)$ in the forward compositional SSD coefficient of Equation (10) about the parameter increments $\Delta\hat{\mathbf{P}}$ leads to

$$\begin{aligned}\widetilde{g}\big(\mathbf{X}+\Delta\hat{\mathbf{U}}(\mathbf{X};\Delta\hat{\mathbf{P}})\big) &\approx \widetilde{g}(\mathbf{X}) + \frac{\partial\widetilde{g}\big(\mathbf{X}+\Delta\hat{\mathbf{U}}(\mathbf{X};\Delta\hat{\mathbf{P}})\big)}{\partial(\Delta\hat{\mathbf{P}})}\Delta\hat{\mathbf{P}} \\ &= g\Big(\mathbf{X}+\tilde{\mathbf{U}}\Big(\mathbf{X};\tilde{\mathbf{P}}\Big)\Big) + \tilde{\mathbf{J}}\Big(\mathbf{X};\tilde{\mathbf{P}}\Big)\Delta\hat{\mathbf{P}},\end{aligned} \tag{37a}$$

where the Jacobian vector $\tilde{\mathbf{J}}\Big(\mathbf{X};\tilde{\mathbf{P}}\Big)$ is computed at $\Delta\hat{\mathbf{P}} \approx \mathbf{0}$ using the chain rule as

$$\begin{aligned}\tilde{\mathbf{J}}\Big(\mathbf{X};\tilde{\mathbf{P}}\Big) &\equiv \frac{\partial\widetilde{g}}{\partial(\Delta\hat{\mathbf{P}})} = \frac{\partial\widetilde{g}}{\partial\widetilde{\mathbf{X}}}\frac{\partial\widetilde{\mathbf{X}}}{\partial(\Delta\hat{\mathbf{P}})} = \frac{\partial g}{\partial\mathbf{x}}\frac{\partial\mathbf{x}}{\partial\widetilde{\mathbf{X}}}\frac{\partial\widetilde{\mathbf{X}}}{\partial(\Delta\hat{\mathbf{P}})} \\ &= \frac{\partial g}{\partial\mathbf{x}}\Big(\mathbf{I} + \frac{\partial\widetilde{\mathbf{U}}}{\partial\widetilde{\mathbf{X}}}\Big)\frac{\partial\Delta\hat{\mathbf{U}}}{\partial(\Delta\hat{\mathbf{P}})},\end{aligned} \tag{37b}$$

or

$$\begin{aligned}\tilde{\mathbf{J}}\Big(\mathbf{X};\tilde{\mathbf{P}}\Big) &= \left\{\frac{\partial g\Big(\mathbf{X}+\tilde{\mathbf{U}}\Big(\mathbf{X};\tilde{\mathbf{P}}\Big)\Big)}{\partial\mathbf{x}}\left[\mathbf{I} + \frac{\partial\tilde{\mathbf{U}}\Big(\mathbf{X};\tilde{\mathbf{P}}\Big)}{\partial\widetilde{\mathbf{X}}}\right]\right\}\frac{\partial[\Delta\hat{\mathbf{U}}(\mathbf{X};\Delta\hat{\mathbf{P}})]}{\partial(\Delta\hat{\mathbf{P}})} \\ &= \Big\{g_{\mathbf{x}}\Big(\mathbf{X}+\tilde{\mathbf{U}}\Big)\Big(\mathbf{I}+\tilde{\mathbf{U}}_{\tilde{\mathbf{X}}}\Big)\Big\}\Delta\hat{\mathbf{U}}_{\Delta\hat{\mathbf{P}}}(\mathbf{X}).\end{aligned} \tag{37c}$$

For the affine deformation given by Equation (4), one has

$$\mathbf{I} + \tilde{\mathbf{U}}_{\tilde{\mathbf{X}}} = \mathbf{I} + \frac{\partial\tilde{\mathbf{U}}\Big(\mathbf{X};\tilde{\mathbf{P}}\Big)}{\partial\widetilde{\mathbf{X}}} = \begin{pmatrix} 1+\tilde{U}_X & \tilde{U}_Y \\ \tilde{V}_X & 1+\tilde{V}_Y \end{pmatrix}, \tag{38a}$$

$$\begin{aligned}&\Delta\hat{\mathbf{U}}_{\Delta\hat{\mathbf{P}}} = \frac{\partial\Delta\hat{\mathbf{U}}(\mathbf{X};\Delta\hat{\mathbf{P}})}{\partial(\Delta\hat{\mathbf{P}})} = \begin{pmatrix} 1 & 0 & X & 0 & Y & 0 \\ 0 & 1 & 0 & X & 0 & Y \end{pmatrix}, \\ &\tilde{\mathbf{J}}\Big(\mathbf{X};\tilde{\mathbf{P}}\Big) = (g_x, g_y)\begin{pmatrix} 1+\tilde{U}_X & \tilde{U}_Y \\ \tilde{V}_X & 1+\tilde{V}_Y \end{pmatrix}\begin{pmatrix} 1 & 0 & X & 0 & Y & 0 \\ 0 & 1 & 0 & X & 0 & Y \end{pmatrix}.\end{aligned} \tag{38b}$$

Minimization of the approximate forward compositional SSD coefficient (differentiating the SSD coefficient with respect to $\Delta\hat{\mathbf{P}}$ and setting the result to zero) leads to a closed form solution:

$$\begin{aligned}&\frac{\partial C_{FC}(\Delta\hat{\mathbf{P}})}{\partial(\Delta\hat{\mathbf{P}})} = 0: \\ &\Delta\hat{\mathbf{P}} = \tilde{\mathbf{H}}^{-1}\sum_{i=1}^{n}\big[G(\mathbf{X}_i) - g\big(\mathbf{X}_i + \tilde{\mathbf{U}}(\mathbf{X}_i;\tilde{\mathbf{P}})\big)\big]\tilde{\mathbf{J}}(\mathbf{X}_i;\tilde{\mathbf{P}})^T, \\ &\tilde{\mathbf{H}}(\tilde{\mathbf{P}}) = \sum_{i=1}^{n}\tilde{\mathbf{J}}(\mathbf{X}_i;\tilde{\mathbf{P}})^T\tilde{\mathbf{J}}(\mathbf{X}_i;\tilde{\mathbf{P}}),\end{aligned} \tag{39}$$

where $\tilde{\mathbf{H}}(\tilde{\mathbf{P}})$ is the Hessian matrix depending also on the prior estimated displacement parameters $\tilde{\mathbf{P}}$. Both greyscales of the restored image $g(\mathbf{X}+\tilde{\mathbf{U}})$ and their gradients $g_{\mathbf{x}}(\mathbf{X}+\tilde{\mathbf{U}})$ in Equations (38b) and (39) have to be evaluated at the non-integral pixel locations $\mathbf{X}+\tilde{\mathbf{U}}$ by a bilinear or other higher-order interpolation scheme. It is important to note that the only difference between the forward additive and compositional algorithms is the extra term $\mathbf{I}+\tilde{\mathbf{U}}_{\tilde{\mathbf{X}}}$ in the Jacobian vector given in Equation (37c) as compared with that given in Equation (34b).

Finally, a first-order Taylor expansion of the numerically distorted image $G\big(\tilde{\mathbf{X}}+\Delta\tilde{\mathbf{u}}(\tilde{\mathbf{X}};\Delta\tilde{\mathbf{p}})\big)$ in the forward inverse compositional SSD coefficient of Equation (15) about the parameter increments $\Delta\tilde{\mathbf{p}}$ leads to

$$\begin{aligned}G\big(\tilde{\mathbf{X}}+\Delta\tilde{\mathbf{u}}\big(\tilde{\mathbf{X}};\Delta\tilde{\mathbf{p}}\big)\big) &\approx G\big(\tilde{\mathbf{X}}\big) + \frac{\partial G\big(\tilde{\mathbf{X}}+\Delta\tilde{\mathbf{u}}\big(\tilde{\mathbf{X}};\Delta\tilde{\mathbf{p}}\big)\big)}{\partial(\Delta\tilde{\mathbf{p}})}\Delta\tilde{\mathbf{p}} \\ &= G\big(\tilde{\mathbf{X}}\big) + \mathbf{J}_0\big(\tilde{\mathbf{X}}\big)\Delta\tilde{\mathbf{p}},\end{aligned} \tag{40a}$$

where the Jacobian vector $\mathbf{J}_0\big(\tilde{\mathbf{X}}\big)$ is computed at $\Delta\tilde{\mathbf{p}} \approx \mathbf{0}$ using the chain rule as

$$\begin{aligned}\mathbf{J}_0\big(\tilde{\mathbf{X}}\big) &\equiv \frac{\partial G\big(\tilde{\mathbf{X}}+\Delta\tilde{\mathbf{u}}\big(\tilde{\mathbf{X}};\Delta\tilde{\mathbf{p}}\big)\big)}{\partial(\Delta\tilde{\mathbf{p}})} = \frac{\partial G\big(\tilde{\mathbf{X}}\big)}{\partial\mathbf{X}}\frac{\partial\mathbf{X}}{\partial(\Delta\tilde{\mathbf{p}})} \\ &= \frac{\partial G\big(\tilde{\mathbf{X}}\big)}{\partial\mathbf{X}}\frac{\partial\big[\Delta\tilde{\mathbf{u}}\big(\tilde{\mathbf{X}};\Delta\tilde{\mathbf{p}}\big)\big]}{\partial(\Delta\tilde{\mathbf{p}})} = G_{\mathbf{X}}\big(\tilde{\mathbf{X}}\big)\Delta\tilde{\mathbf{u}}_{\Delta\tilde{\mathbf{p}}}\big(\tilde{\mathbf{X}};\Delta\tilde{\mathbf{p}}\big).\end{aligned} \tag{40b}$$

doi: 10.1111/str.12039

For the affine deformation given by Equation (4), one has

$$\Delta\widetilde{\mathbf{u}}_{\Delta\widetilde{\mathbf{p}}}\left(\widetilde{\mathbf{X}};\Delta\widetilde{\mathbf{p}}\right)=\frac{\partial\left[\Delta\widetilde{\mathbf{u}}\left(\widetilde{\mathbf{X}};\Delta\widetilde{\mathbf{p}}\right)\right]}{\partial(\Delta\widetilde{\mathbf{p}})}=\begin{pmatrix}1&0&\widetilde{X}&0&\widetilde{Y}&0\\0&1&0&\widetilde{X}&0&\widetilde{Y}\end{pmatrix}, \tag{41a}$$

$$\begin{aligned}\mathbf{J}_0\left(\widetilde{\mathbf{X}}\right)&=G_{\mathbf{X}}\left(\widetilde{\mathbf{X}}\right)\Delta\widetilde{\mathbf{u}}_{\Delta\widetilde{\mathbf{p}}}\left(\widetilde{\mathbf{X}};\Delta\widetilde{\mathbf{p}}\right)\\&=(G_X,G_Y)\begin{pmatrix}1&0&\widetilde{X}&0&\widetilde{Y}&0\\0&1&0&\widetilde{X}&0&\widetilde{Y}\end{pmatrix}\\&=\left(G_X,G_Y,\widetilde{X}G_X,\widetilde{X}G_Y,\widetilde{Y}G_X,\widetilde{Y}G_Y\right).\end{aligned} \tag{41b}$$

Minimization of the approximate forward inverse compositional SSD coefficient (differentiating the SSD coefficient with respect to $\Delta\widetilde{\mathbf{p}}$ and setting the result to zero) leads to a closed form solution:

$$\begin{aligned}&\frac{\partial C_{FIC}(\Delta\widetilde{\mathbf{p}})}{\partial(\Delta\widetilde{\mathbf{p}})}=0:\\&\Delta\widetilde{\mathbf{p}}=-\mathbf{H}_0^{-1}\sum_{i=1}^{n}\left[G\left(\widetilde{\mathbf{X}}_i\right)-g\left(\widetilde{\mathbf{X}}_i+\widetilde{\mathbf{U}}\left(\widetilde{\mathbf{X}}_i;\widetilde{\mathbf{P}}\right)\right)\right]\mathbf{J}_0\left(\widetilde{\mathbf{X}}_i\right)^T,\\&\mathbf{H}_0=\sum_{i=1}^{n}\mathbf{J}_0\left(\widetilde{\mathbf{X}}_i\right)^T\mathbf{J}_0\left(\widetilde{\mathbf{X}}_i\right),\end{aligned} \tag{42}$$

where $\mathbf{H}_0$ is the Hessian matrix. The subscript '0' used for the Jacobian vector and Hessian matrix emphasises the fact that both of them are independent of the prior estimated displacement parameters $\widetilde{\mathbf{P}}$. Only greyscales of the restored image $g\left(\widetilde{\mathbf{X}}+\widetilde{\mathbf{U}}\right)$ in Equation (42) have to be evaluated at the non-integral pixel locations $\widetilde{\mathbf{X}}+\widetilde{\mathbf{U}}$ by a bilinear or other higher-order interpolation scheme. The difference between the inverse compositional and forward additive algorithms is that the Jacobian vector as given in Equation (40b) is computed from the greyscale gradients of the initial image instead from those of the current image as given in Equation (34b). Because the SSD coefficients of Lucas–Kanade DIC algorithms all have the similar mathematical form as described in the second section and summarised in Tables 1 and 2, a similar closed form solution for the parameter increments should also be readily obtained for any other algorithms discussed in this study.

Equivalence of numerical solutions by additive, compositional and inverse compositional algorithms

Here, analytical proofs on the equivalence of the numerical solutions by forward additive, compositional and inverse compositional algorithms are first presented for the case of one-dimensional affine deformation $\mathbf{P}=(U_0,U_X)^T$. Written in the matrix form for this case, the closed form solution Equation (36) for forward additive algorithm becomes

$$\begin{pmatrix}\Delta U_0\\\Delta U_X\end{pmatrix}=\mathbf{H}^{-1}\begin{pmatrix}\sum_{i=1}^{n}[G-g]g_x\\\sum_{i=1}^{n}[G-g]Xg_x\end{pmatrix}\approx\begin{pmatrix}\frac{\sum_{i=1}^{n}[G-g]g_x}{\sum_{i=1}^{n}(g_x)^2}\\\frac{\sum_{i=1}^{n}[G-g]Xg_x}{\sum_{i=1}^{n}(Xg_x)^2}\end{pmatrix}, \tag{43a}$$

$$\text{where }\mathbf{H}=\mathbf{H}(\mathbf{P})=\begin{pmatrix}\sum_{i=1}^{n}(g_x)^2&\sum_{i=1}^{n}X(g_x)^2\\\sum_{i=1}^{n}X(g_x)^2&\sum_{i=1}^{n}(Xg_x)^2\end{pmatrix}. \tag{43b}$$

Similarly, the closed form solution, Equation (39), for compositional algorithm becomes

$$\begin{aligned}\begin{pmatrix}\Delta\hat{U}_0\\\Delta\hat{U}_X\end{pmatrix}&=\mathbf{H}^{-1}\left(1+\widetilde{U}_X\right)\begin{pmatrix}\sum_{i=1}^{n}[G-g]g_x\\\sum_{i=1}^{n}[G-g]Xg_x\end{pmatrix}\\&\approx\frac{1}{\left(1+\widetilde{U}_X\right)}\begin{pmatrix}\frac{\sum_{i=1}^{n}[G-g]g_x}{\sum_{i=1}^{n}(g_x)^2}\\\frac{\sum_{i=1}^{n}[G-g]Xg_x}{\sum_{i=1}^{n}(Xg_x)^2}\end{pmatrix}\\&=\frac{1}{\left(1+\widetilde{U}_X\right)}\begin{pmatrix}\Delta U_0\\\Delta U_X\end{pmatrix},\end{aligned} \tag{43c}$$

$$\text{where from Eq.(38b) and Eq.(39)}:\quad\widetilde{\mathbf{H}}=\widetilde{\mathbf{H}}\left(\widetilde{\mathbf{P}}\right)=\left(1+\widetilde{U}_X\right)^2\begin{pmatrix}\sum_{i=1}^{n}(g_x)^2&\sum_{i=1}^{n}X(g_x)^2\\\sum_{i=1}^{n}X(g_x)^2&\sum_{i=1}^{n}(Xg_x)^2\end{pmatrix}, \tag{43d}$$

Finally, the closed form solution Equation (42) for inverse compositional algorithm becomes

$$\begin{aligned}\begin{pmatrix}\Delta\widetilde{u}_0\\\Delta\widetilde{u}_X\end{pmatrix}&=-\mathbf{H}_0^{-1}\begin{pmatrix}\sum_{i=1}^{n}[G-g]G_X\\\sum_{i=1}^{n}[G-g]\widetilde{X}G_X\end{pmatrix}\approx-\begin{pmatrix}\frac{\sum_{i=1}^{n}[G-g]G_X}{\sum_{i=1}^{n}(G_X)^2}\\\frac{\sum_{i=1}^{n}[G-g]\widetilde{X}G_X}{\sum_{i=1}^{n}\left(\widetilde{X}G_X\right)^2}\end{pmatrix}\\&\approx\frac{1}{\left(1+\widetilde{U}_X\right)}\begin{pmatrix}-\Delta U_0\\-\Delta U_X\end{pmatrix},\end{aligned} \tag{43e}$$

doi: 10.1111/str.12039

$$\textit{where}\ \ \mathbf{H}_0 = \begin{pmatrix} \sum_{i=1}^{n}(G_X)^2 & \sum_{i=1}^{n}\widetilde{X}(G_X)^2 \\ \sum_{i=1}^{n}X\,(G_X)^2 & \sum_{i=1}^{n}\left(\widetilde{X}G_X\right)^2 \end{pmatrix} \approx \left(1+\widetilde{U}_X\right)^2 \begin{pmatrix} \sum_{i=1}^{n}(g_x)^2 & \sum_{i=1}^{n}X(g_x)^2 \\ \sum_{i=1}^{n}X(g_x)^2 & \sum_{i=1}^{n}(Xg_x)^2 \end{pmatrix}. \tag{43f}$$

The diagonal approximation are made to the inverse of the Hessian matrices above and such an approximation is very good when the origin of coordinates either X or $\widetilde{X}$ is set at the geometrical centre of the image subset [29]. In addition, $G(\widetilde{X}) \approx g(\widetilde{X}+\widetilde{U})$ with $\Delta\widetilde{\mathbf{p}} \to \mathbf{0}$ is assumed at the end of the iterative numerical solution process, so one has

$$G_X = \frac{\partial G(\widetilde{X})}{\partial X} \approx \frac{\partial g}{\partial x}\frac{\partial x}{\partial X} = \frac{\partial g}{\partial x}\frac{\partial x}{\partial \widetilde{X}} = g_x\left(1+\widetilde{U}_X\right) \tag{44}$$

for the inverse compositional algorithm given in Equations (43e) and (43f).

The forward composite mapping of Equation (12) for the one-dimensional case with its parameter increments given in Equation (43b) becomes

$$\mathbf{\Phi}\left(\widetilde{\mathbf{P}}\right)\mathbf{\Phi}\left(\Delta\hat{\mathbf{P}}\right) = \begin{pmatrix} 1+\widetilde{U}_X & \widetilde{U}_0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1+\Delta\hat{U}_X & \Delta\hat{U}_0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1+\widetilde{U}_X & \widetilde{U}_0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1+\dfrac{\Delta U_X}{1+\widetilde{U}_X} & \dfrac{\Delta U_0}{1+\widetilde{U}_X} \\ 0 & 1 \end{pmatrix}, \tag{44a}$$

or

$$\mathbf{\Phi}\left(\widetilde{\mathbf{P}}\right)\mathbf{\Phi}\left(\Delta\hat{\mathbf{P}}\right) = \begin{pmatrix} 1+\widetilde{U}_X+\Delta U_X & \widetilde{U}_0+\Delta U_0 \\ 0 & 1 \end{pmatrix}, \tag{44b}$$

which is identical to that of Equation (9) in the one-dimensional case (with $\widetilde{U}_X = U_X$). In other words, parameters obtained for the one-dimensional affine deformation by the forward compositional algorithm are equivalent to those obtained by the forward additive algorithm. Similarly, the inverse composite mapping of Equation (17) for the one-dimensional case with its parameter increments given in Equation (43e) becomes

$$\mathbf{\Phi}\left(\widetilde{\mathbf{P}}\right)\boldsymbol{\varphi}(\Delta\widetilde{\mathbf{p}}) = \begin{pmatrix} 1+\widetilde{U}_X & \widetilde{U}_0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1+\widetilde{u}_X & \Delta\widetilde{u}_0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1+\widetilde{U}_X & \widetilde{U}_0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1-\dfrac{\Delta U_X}{1+\widetilde{U}_X} & -\dfrac{\Delta U_0}{1+\widetilde{U}_X} \\ 0 & 1 \end{pmatrix}^{-1}, \tag{45a}$$

or (recall $\boldsymbol{\varphi}(\Delta\widetilde{\mathbf{p}})^{-1} = \boldsymbol{\varphi}(-\Delta\widetilde{\mathbf{p}})$ for $\Delta\widetilde{\mathbf{p}} \to \mathbf{0}$ [15])

$$\mathbf{\Phi}\left(\widetilde{\mathbf{P}}\right)\boldsymbol{\varphi}(\Delta\widetilde{\mathbf{p}}) \approx \begin{pmatrix} 1+\widetilde{U}_X & \widetilde{U}_0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1+\dfrac{\Delta U_X}{1+\widetilde{U}_X} & \dfrac{\Delta U_0}{1+\widetilde{U}_X} \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1+\widetilde{U}_X+\Delta U_X & \widetilde{U}_0+\Delta U_0 \\ 0 & 1 \end{pmatrix}, \tag{45b}$$

which is again identical to that of Equation (9) in the one-dimensional case (with $\widetilde{U}_X = U_X$). In other words, parameters obtained for the one-dimensional affine deformation by the forward inverse compositional algorithm are also equivalent to those obtained by the forward additive algorithm.

Modified forward additive and compositional algorithms

A modified version of the forward additive algorithm given in Equation (7) with its closed form numerical solution given in Equation (36) has been proposed by Hager and Belhumeur [21]. The modified algorithm has been called as 'inverse additive' by Baker and Matthews [15]. The basic premise of the approach by Hager and Belhumeur [21] is that (1) the prior estimated deformation parameters $\mathbf{P}$ are sufficiently close to the final solution, so the numerically restored image $g(\mathbf{X}+\mathbf{U})$ is very close to the initial image $G(\mathbf{X})$ and (2) the parameterized deformation function has certain properties, so the greyscale gradients and the Jacobian vector may be computed very differently:

$$g(\mathbf{x}(\mathbf{X};\mathbf{P})) = g(\mathbf{X}+\mathbf{U}(\mathbf{X};\mathbf{P})) \approx G(\mathbf{X}),$$
$$\frac{\partial g}{\partial \mathbf{X}} = \frac{\partial g}{\partial \mathbf{x}}\frac{\partial \mathbf{x}}{\partial \mathbf{X}} = \frac{\partial g}{\partial \mathbf{x}}\left(\mathbf{I}+\frac{\partial \mathbf{U}(\mathbf{X};\mathbf{P})}{\partial \mathbf{X}}\right) \approx \frac{\partial G(\mathbf{X})}{\partial \mathbf{X}}, \tag{46a}$$

$$\text{so}\ \ g_{\mathbf{x}}(\mathbf{X}+\mathbf{U}) = \frac{\partial g}{\partial \mathbf{x}} \approx \frac{\partial G(\mathbf{X})}{\partial \mathbf{X}}\left(\mathbf{I}+\frac{\partial \mathbf{U}(\mathbf{X};\mathbf{P})}{\partial \mathbf{X}}\right)^{-1} = G_{\mathbf{X}}(\mathbf{X})[\mathbf{I}+\mathbf{U}_{\mathbf{X}}(\mathbf{X};\mathbf{P})]^{-1}, \tag{46b}$$

$$\mathbf{J}(\mathbf{X};\mathbf{P}) = G_{\mathbf{X}}(\mathbf{X})[\mathbf{I}+\mathbf{U}_{\mathbf{X}}(\mathbf{X};\mathbf{P})]^{-1}\mathbf{U}_{\mathbf{P}}(\mathbf{X};\mathbf{P}) = (G_X, G_Y)\begin{pmatrix} 1+U_X & U_Y \\ V_X & 1+V_Y \end{pmatrix}^{-1}\begin{pmatrix} 1 & 0 & X & 0 & Y & 0 \\ 0 & 1 & 0 & X & 0 & Y \end{pmatrix}. \tag{46c}$$

Following the matrix form given in Baker and Matthews [15], the Jacobian vector above can be re-written as

© 2013 Wiley Publishing Ltd | *Strain* (2013) **49**, 313–334
doi: 10.1111/str.12039

$$\mathbf{J}(\mathbf{X};\mathbf{P}) = (G_X, G_Y)\begin{pmatrix} 1 & 0 & X & 0 & Y & 0 \\ 0 & 1 & 0 & X & 0 & Y \end{pmatrix}\begin{pmatrix} 1+V_Y & -U_Y & 0 & 0 & 0 & 0 \\ -V_X & 1+U_X & 0 & 0 & 0 & 0 \\ 0 & 0 & 1+V_Y & -U_Y & 0 & 0 \\ 0 & 0 & -V_X & 1+U_X & 0 & 0 \\ 0 & 0 & 0 & 0 & 1+V_Y & -U_Y \\ 0 & 0 & 0 & 0 & -V_X & 1+U_X \end{pmatrix}/D. \tag{47}$$

where $D=(1+U_X)(1+V_Y)-U_YV_X>0$. So both the Jacobian vector and Hessian matrix can be re-written as

$$\begin{aligned}\mathbf{J}(\mathbf{X};\mathbf{P}) &= \mathbf{J}_0(\mathbf{X})\boldsymbol{\Gamma}(\mathbf{P}),\ \ \mathbf{H}(\mathbf{P}) = \sum_{i=1}^{n}\mathbf{J}(\mathbf{X}_i;\mathbf{P})^T\mathbf{J}(\mathbf{X};\mathbf{P}) \\ &= \boldsymbol{\Gamma}(\mathbf{P})^T\left[\sum_{i=1}^{n}\mathbf{J}_0(\mathbf{X})^T\mathbf{J}_0(\mathbf{X})\right]\boldsymbol{\Gamma}(\mathbf{P}) = \boldsymbol{\Gamma}(\mathbf{P})^T\mathbf{H}_0\boldsymbol{\Gamma}(\mathbf{P}),\end{aligned} \tag{48}$$

where $\mathbf{H}(\mathbf{P})$ is the Hessian matrix depending on the prior estimated displacement parameters $\mathbf{P}$. Minimization of the approximate forward additive SSD coefficient (differentiating the SSD coefficient with respect to $\Delta\mathbf{P}$ and setting the result to zero) leads to a closed form solution:

$$\begin{aligned}\Delta\mathbf{P} &= \mathbf{H}^{-1}\sum_{i=1}^{n}[G(\mathbf{X}_i) - g(\mathbf{X}_i+\mathbf{U}(\mathbf{X}_i;\mathbf{P}))]\mathbf{J}(\mathbf{X}_i;\mathbf{P})^T \\ &= \boldsymbol{\Gamma}(\mathbf{P})^{-1}\mathbf{H}_0^{-1}\sum_{i=1}^{n}[G(\mathbf{X}_i) - g(\mathbf{X}_i+\mathbf{U})]\mathbf{J}_0(\mathbf{X}_i)^T,\end{aligned} \tag{49a}$$

$$\text{or}\quad \Delta\mathbf{P} = -\boldsymbol{\Gamma}(\mathbf{P})^{-1}\Delta\widetilde{\mathbf{p}}, \tag{49b}$$

where $\Delta\widetilde{\mathbf{p}}$ is given by Equation (42) and $\Gamma(\mathbf{P})^{-1}$ is a 6-by-6 matrix as [15]

$$\boldsymbol{\Gamma}(\mathbf{P})^{-1} = \begin{pmatrix} 1+U_X & U_Y & 0 & 0 & 0 & 0 \\ V_X & 1+V_Y & 0 & 0 & 0 & 0 \\ 0 & 0 & 1+U_X & U_Y & 0 & 0 \\ 0 & 0 & V_X & 1+V_Y & 0 & 0 \\ 0 & 0 & 0 & 0 & 1+U_X & U_Y \\ 0 & 0 & 0 & 0 & V_X & 1+V_Y \end{pmatrix}.$$

The key assumption about the modified additive algorithm is that the Jacobian vector can be changed from the form of Equation (46c) into the form of Equation (47), and the matrix ($\mathbf{P}$) is invertible [21]. It turns out that affine deformation mapping considered here satisfies just such conditions. Furthermore, one can prove through some algebraic operations that the parameter increments $\Delta\mathbf{P}$ given in Equation (49b) is the same as $\Delta\mathbf{P}$ computed from Equation (17) (noting $\mathbf{P}=\widetilde{\mathbf{P}}$ and $\boldsymbol{\varphi}(\Delta\widetilde{\mathbf{p}})^{-1}=\boldsymbol{\varphi}(-\Delta\widetilde{\mathbf{p}})$ for $\Delta\widetilde{\mathbf{p}}\to\mathbf{0}$). That is, the final numerical solutions obtained by the modified forward additive and inverse compositional algorithms are equivalent for (2D) affine deformation mapping.

The forward compositional algorithm given in Equation (10) with its closed form numerical solution given in Equation (39) can also be modified in a similar way. Again, it is assumed that that the prior estimated deformation parameters $\widetilde{\mathbf{P}}$ are sufficiently close to the final solution, so the numerically restored image $\widetilde{g}(\mathbf{X}+\Delta\mathbf{U})$ is very close to the initial image $G(\mathbf{X})$, namely (with $\Delta\hat{\mathbf{P}}\to\mathbf{0}$)

$$\begin{aligned}&\widetilde{g}(\mathbf{X}+\Delta\mathbf{U}) = g\left(\mathbf{x}\left(\widetilde{\mathbf{X}}(\mathbf{X};\Delta\hat{\mathbf{P}});\widetilde{\mathbf{P}}\right)\right)\approx G(\mathbf{X}),\qquad \text{so} \\ &\frac{\partial g}{\partial\mathbf{X}} = \frac{\partial g}{\partial\mathbf{x}}\frac{\partial\mathbf{x}}{\partial\widetilde{\mathbf{X}}}\frac{\partial\widetilde{\mathbf{X}}}{\partial\mathbf{X}} = \frac{\partial g}{\partial\mathbf{x}}\left(\mathbf{I}+\frac{\partial\widetilde{\mathbf{U}}}{\partial\mathbf{X}}\right)\approx\frac{\partial G(\mathbf{X})}{\partial\mathbf{X}}.\end{aligned} \tag{51}$$

The Jacobian vector given in Equation (38b) will thus be replaced by the one given in Equation (41b), noting $\widetilde{\mathbf{X}}$ in Equation (41b) is also very close to $\mathbf{X}$ in Equation (38b) for $\Delta\hat{\mathbf{P}}\to\mathbf{0}$. Consequently, the numerical solution given in Equation (39) becomes

$$\begin{aligned}\Delta\hat{\mathbf{P}} &= \widetilde{\mathbf{H}}^{-1}\sum_{i=1}^{n}\left[G(\mathbf{X}_i) - g\left(\mathbf{X}_i+\widetilde{\mathbf{U}}\left(\mathbf{X}_i\widetilde{\mathbf{P}}\right)\right)\right]\widetilde{\mathbf{J}}\left(\mathbf{X}_i\widetilde{\mathbf{P}}\right)^T \\ &= \mathbf{H}_0^{-1}\sum_{i=1}^{n}\left[G(\mathbf{X}_i) - g\left(\mathbf{X}_i+\widetilde{\mathbf{U}}\right)\right]\mathbf{J}_0(\mathbf{X}_i)^T = -\Delta\widetilde{\mathbf{p}}.\end{aligned} \tag{52}$$

One can prove that $\boldsymbol{\Phi}(\Delta\hat{\mathbf{P}}) = \boldsymbol{\varphi}(\Delta\widetilde{\mathbf{p}})^{-1}$ for $\Delta\hat{\mathbf{P}} = -\Delta\widetilde{\mathbf{p}}$ (noting again $\boldsymbol{\varphi}(\Delta\widetilde{\mathbf{p}})^{-1}=\boldsymbol{\varphi}(-\Delta\widetilde{\mathbf{p}})$ for $\Delta\widetilde{\mathbf{p}}\to\mathbf{0}$). As the updated parameters for forward compositional and inverse compositional algorithms are given by Equations (12) and (17), respectively, the final numerical solutions of the deformation parameters obtained by the forward compositional and inverse compositional algorithms are equivalent for (2D) affine deformation mapping.

doi: 10.1111/str.12039

Comparison of the computational cost

Closed form solutions for the parameter increments of the affine displacement field function of an image subset have been obtained for forward additive, compositional and inverse compositional DIC algorithms using the Gauss–Newton gradient decent method, as shown in Equations (36), (39) and (42). Assuming that the image pair are of sufficient high quality and the prior estimate of the displacement parameters is within the radius of convergence of the gradient decent method, the repeated use of the closed form solution after parameter updating at each iteration step is expected to produce the correct displacement parameters within the limit of image noises and numerical approximation errors [4, 29, 31]. The major differences in the numerical solutions of DIC algorithms include the computation of the Jacobian vector (for example, see Equations (34b), (37c) and (40b)) and the subsequent formation of the corresponding Hessian matrix, whether only one image or both images are numerically modified in the correlation analysis, and how the displacement parameters are updated from their prior estimates with the new increments. The major computational cost associated with the numerical solution will be the re-evaluation of image greyscales and especially their spatial gradients at the non-integral pixel positions at each iteration step. As the greyscale gradients are re-calculated, so the inverse of the Hessian matrix in the closed form solution has to be updated as well, which is computationally very expensive [15]. Tables 3 and 4 list the required image greyscales, and their spatial gradients appeared in a closed form solution of each algorithm by the Gauss–Newton gradient decent method. Their explicit computations in terms of image greyscales at only integral pixel locations with the case of 1D affine deformation and linear interpolation are given in Table 5 for all forward and backward algorithms.

As shown in Table 3, the computational costs of forward and backward algorithms of the same kind (additive, compositional or inverse compositional) are exactly the same, assuming the same subpixel interpolation scheme is used. The computational costs of the additive and compositional algorithms are similar as well as both the image greyscales and their gradients are re-computed at each iteration step after the parameter updating. However, both forward and backward inverse compositional algorithms will incur much less computational cost as the spatial gradients of image greyscales are computed at the integral pixel positions and remain the same for the entire iterative solution process (that is, they need to be pre-computed only once for constructing the Jacobian vector and Hessian

Table 3: Gradient calculation and subpixel interpolation in forward and backward DIC algorithms

Algorithm name	Coordinates (integer pixel)	Greyscale spatial gradient and parameter updating	Subpixel interpolation ($\mathbf{X}+\mathbf{U}=\Phi(\mathbf{P})\mathbf{X}$, etc.)
Forward additive	$\mathbf{X}$	$g_{\mathbf{X}}(\mathbf{X}+\mathbf{U})$, $\mathbf{P}\rightarrow\mathbf{P}+\Delta\mathbf{P}$	$g(\mathbf{X}+\mathbf{U})$, $g_{\mathbf{X}}(\mathbf{X}+\mathbf{U})$
Backward additive	$\mathbf{x}$	$G_{\mathbf{X}}(\mathbf{x}+\mathbf{u})$, $\mathbf{p}\rightarrow\mathbf{p}+\Delta\mathbf{p}$	$G(\mathbf{x}+\mathbf{u})$, $G_{\mathbf{X}}(\mathbf{x}+\mathbf{u})$
Forward compositional	$\mathbf{X}$	$\widetilde{g}_{\widetilde{\mathbf{X}}}(\mathbf{X})=g_{\mathbf{x}}\left(\mathbf{X}+\widetilde{\mathbf{U}}\right)\left(\mathbf{I}+\frac{\partial\widetilde{\mathbf{U}}}{\partial\widetilde{\mathbf{X}}}\right)$, $\Phi\left(\widetilde{\mathbf{P}}\right)\Phi(\Delta\hat{\mathbf{P}})\rightarrow\Phi(\mathbf{P}+\Delta\mathbf{P})$	$g\left(\mathbf{X}+\widetilde{\mathbf{U}}\right), g_{\mathbf{x}}\left(\mathbf{X}+\widetilde{\mathbf{U}}\right)$, $\widetilde{g}(\mathbf{X})\equiv g\left(\mathbf{X}+\widetilde{\mathbf{U}}\right)$
Backward compositional	$\mathbf{x}$	$\widetilde{G}_{\widetilde{\mathbf{X}}}(\mathbf{x})=G_{\mathbf{X}}(\mathbf{x}+\widetilde{\mathbf{u}})\left(\mathbf{I}+\frac{\partial\widetilde{\mathbf{u}}}{\partial\widetilde{\mathbf{x}}}\right)$, $\boldsymbol{\varphi}(\widetilde{\mathbf{p}})\boldsymbol{\varphi}(\Delta\hat{\mathbf{p}})\rightarrow\boldsymbol{\varphi}(\mathbf{p}+\Delta\mathbf{p})$	$G(\mathbf{x}+\widetilde{\mathbf{u}}), G_{\mathbf{X}}(\mathbf{x}+\widetilde{\mathbf{u}})$, $\widetilde{G}(\mathbf{x})\equiv G(\mathbf{x}+\widetilde{\mathbf{u}})$
Forward inverse compositional	$\widetilde{\mathbf{X}}$	$G_{\mathbf{X}}\left(\widetilde{\mathbf{X}}\right)^{*}$, $\Phi\left(\widetilde{\mathbf{P}}\right)\boldsymbol{\varphi}(\Delta\widetilde{\mathbf{p}})^{-1}\rightarrow\Phi(\mathbf{P}+\Delta\mathbf{P})$	$g\left(\widetilde{\mathbf{X}}+\widetilde{\mathbf{U}}\right)$
Backward inverse compositional	$\widetilde{\mathbf{x}}$	$g_{\mathbf{x}}(\widetilde{\mathbf{x}})^{*}$, $\boldsymbol{\varphi}(\widetilde{\mathbf{p}})\Phi\left(\Delta\widetilde{\mathbf{P}}\right)^{-1}\rightarrow\boldsymbol{\varphi}(\mathbf{p}+\Delta\mathbf{p})$	$G(\widetilde{\mathbf{x}}+\widetilde{\mathbf{u}})$

DIC, digital image correlation.
*Evaluated at the same integral pixel positions and thus computed only once.
Unless otherwise noted, the coordinate symbols listed in the second column are integer-valued pixel points. The spatial gradients of greyscales needed for completing gradient-decent Gauss–Newton numerical solutions are listed in the third column. The updating methods for increments of deformation parameters are given also in the third column. Images that need to be numerically distorted or restored with subpixel interpolation of greyscales and their spatial gradients that need to be computed are listed in the fourth column.

doi: 10.1111/str.12039

Table 4: Greyscale gradient calculation and subpixel interpolation in consistent and symmetric DIC algorithms

Algorithm name	Coordinates (integer pixel)	Greyscale spatial gradient and parameter matrix relation	Subpixel interpolation ($\mathbf{X}+\mathbf{U}=\boldsymbol{\Phi}(\mathbf{P})\mathbf{X}$, etc.)
Consistent additive	$\mathbf{X}$ $\widetilde{\mathbf{x}}=\boldsymbol{\Phi}\left(\widetilde{\mathbf{P}}\right)\widetilde{\mathbf{X}}^{*}$	$g_{\mathbf{x}}(\mathbf{X}+\mathbf{U}), G_{\mathbf{X}}(\mathbf{x}+\mathbf{u})$, $\boldsymbol{\varphi}(\mathbf{p}+\Delta\mathbf{p})=\boldsymbol{\Phi}(\mathbf{P}+\Delta\mathbf{P})^{-1}$	$g(\mathbf{X}+\mathbf{U}), g_{\mathbf{x}}(\mathbf{X}+\mathbf{U})$, $g(\mathbf{x}), G(\mathbf{x}+\mathbf{u}), G_{\mathbf{X}}(\mathbf{x}+\mathbf{u})$
Consistent compositional	$\mathbf{X}$ $\widetilde{\mathbf{x}}=\boldsymbol{\Phi}\left(\widetilde{\mathbf{P}}\right)\widetilde{\mathbf{X}}^{*}$	$g_{\mathbf{x}}\left(\mathbf{X}+\widetilde{\mathbf{U}}\right)\left(\mathbf{I}+\dfrac{\partial\widetilde{\mathbf{U}}}{\partial\widetilde{\mathbf{X}}}\right)$, $G_{\mathbf{X}}(\mathbf{x}+\widetilde{\mathbf{u}})\left(\mathbf{I}+\dfrac{\partial\widetilde{\mathbf{u}}}{\partial\widetilde{\mathbf{x}}}\right)$, $\boldsymbol{\varphi}(\Delta\hat{\mathbf{p}})=\boldsymbol{\Phi}(\hat{\mathbf{P}})\boldsymbol{\Phi}(\Delta\hat{\mathbf{P}})^{-1}\boldsymbol{\Phi}\left(\widetilde{\mathbf{P}}\right)^{-1}$	$g\left(\mathbf{X}+\widetilde{\mathbf{U}}\right), g_{\mathbf{x}}\left(\mathbf{X}+\widetilde{\mathbf{U}}\right)$, $g(\mathbf{x}), G(\mathbf{x}+\widetilde{\mathbf{u}}), G_{\mathbf{X}}(\mathbf{x}+\widetilde{\mathbf{u}})$
Consistent inverse compositional	$\widetilde{\mathbf{X}}$ $\widetilde{\mathbf{x}}=\boldsymbol{\Phi}\left(\widetilde{\mathbf{P}}\right)\widetilde{\mathbf{X}}^{*}$	$G_{\mathbf{X}}\left(\widetilde{\mathbf{X}}\right)^{**}, g_{\mathbf{x}}(\widetilde{\mathbf{x}})$, $\boldsymbol{\varphi}(\Delta\hat{\mathbf{p}})=\boldsymbol{\Phi}\left(\widetilde{\mathbf{P}}\right)^{-1}\boldsymbol{\Phi}\left(\Delta\widetilde{\mathbf{P}}\right)^{-1}\boldsymbol{\Phi}\left(\widetilde{\mathbf{P}}\right)$	$g\left(\widetilde{\mathbf{X}}+\widetilde{\mathbf{U}}\right), G(\widetilde{\mathbf{x}}+\widetilde{\mathbf{u}})$, $g(\widetilde{\mathbf{x}}), g_{\mathbf{x}}(\widetilde{\mathbf{x}})$
Consistent mixed compositional	$\mathbf{X}$ $\widetilde{\mathbf{X}}\approx\mathbf{X}$	$g_{\mathbf{x}}\left(\mathbf{X}+\hat{\mathbf{U}}\right)\left(\mathbf{I}+\dfrac{\partial\widetilde{\mathbf{U}}}{\partial\widetilde{\mathbf{X}}}\right), \quad G_{\mathbf{X}}\left(\widetilde{\mathbf{X}}\right)^{**}$, $\boldsymbol{\varphi}(\Delta\hat{\mathbf{p}})=\boldsymbol{\Phi}\left(\Delta\hat{\mathbf{P}}\right)^{-1}$	$g\left(\mathbf{X}+\hat{\mathbf{U}}\right), g_{\mathbf{x}}\left(\mathbf{X}+\hat{\mathbf{U}}\right)$
Symmetric additive	$\overline{\mathbf{X}}$	$g_{\mathbf{x}}(\overline{\mathbf{X}}+\overline{\mathbf{U}}), G_{\mathbf{X}}(\overline{\mathbf{X}}+\overline{\mathbf{u}})$, $\boldsymbol{\varphi}(\overline{\mathbf{p}}+\Delta\overline{\mathbf{p}})=\boldsymbol{\Phi}(\overline{\mathbf{P}}+\Delta\overline{\mathbf{P}})^{-1}$	$g(\overline{\mathbf{X}}+\overline{\mathbf{U}}), g_{\mathbf{x}}(\overline{\mathbf{X}}+\overline{\mathbf{U}})$ $G(\overline{\mathbf{X}}+\overline{\mathbf{u}}), G_{\mathbf{X}}(\overline{\mathbf{X}}+\overline{\mathbf{u}})$
Symmetric compositional	$\overline{\mathbf{X}}$	$g_{\mathbf{x}}\left(\overline{\mathbf{X}}+\widetilde{\mathbf{U}}\right)\left(\mathbf{I}+\dfrac{\partial\widetilde{\mathbf{U}}}{\partial\widetilde{\mathbf{X}}}\right)$, $G_{\mathbf{X}}(\overline{\mathbf{X}}+\widetilde{\mathbf{u}})\left(\mathbf{I}+\dfrac{\partial\widetilde{\mathbf{u}}}{\partial\widetilde{\mathbf{x}}}\right)$, $\boldsymbol{\varphi}(\Delta\hat{\mathbf{p}})=\boldsymbol{\Phi}\left(\widetilde{\mathbf{P}}\right)\boldsymbol{\Phi}(\Delta\hat{\mathbf{P}})^{-1}\boldsymbol{\Phi}\left(\widetilde{\mathbf{P}}\right)^{-1}$	$g\left(\overline{\mathbf{X}}+\widetilde{\mathbf{U}}\right), g_{\mathbf{x}}\left(\overline{\mathbf{X}}+\widetilde{\mathbf{U}}\right)$ $G(\overline{\mathbf{X}}+\widetilde{\mathbf{u}}), G_{\mathbf{X}}(\overline{\mathbf{X}}+\widetilde{\mathbf{u}})$

DIC, digital image correlation.
*Non-integral pixel positions. **Evaluated at the same integral pixel positions and thus computed only once.
Unless otherwise noted, the coordinate symbols listed in the second column are integer-valued pixel points. The spatial gradients of greyscales needed for completing gradient-decent Gauss–Newton numerical solutions are listed in the third column. The relations between increments of deformation parameters in two different deformation parameter matrices are given in the third column. Images that need to be numerically distorted or restored with subpixel interpolation of greyscales and their spatial gradients that need to be computed are listed in the fourth column.

matrix, [15]). When analysing a sequence of images using the same initial image as the reference, both the Jacobian vector and Hessian matrix are unchanged so the computational cost is further reduced in comparison with the additive and compositional algorithms. One possible significant saving on the computational cost for additive and compositional algorithms is when the prior estimate of the deformation parameters are close enough to the final convergent solution so the Hessian matrix in the closed-form solutions no longer needs updating for a given subpixel interpolation scheme used. As shown in Table 5, the greyscale gradients computed from a central difference formula will remain constant when the maximum absolute increment in displacements is too small to change the integral pixel coordinates Ω used in the gradient calculations. Nevertheless, tracking such changes is rather cumbersome in comparison with the inverse compositional algorithms where the Hessian matrix always remains constant.

The computational costs are generally higher for consistent and symmetric algorithms, and the computational cost savings of a constant Hessian matrix in inverse compositional algorithms are no longer available, see Table 4. Again, the consistent additive and compositional algorithms incur about the same amount of the computational cost consisting of subpixel interpolation of greyscales of three images and spatial gradient vectors of greyscales of two images. The consistent inverse compositional algorithm is only slightly

doi: 10.1111/str.12039

Table 5: Forward and backward DIC algorithms for 1D affine deformation with linear interpolation

Algorithm name	Integer pixel (fractional)	Greyscale spatial gradient	Subpixel interpolation (updating with $\mathbf{U}, \mathbf{u}, \widetilde{\mathbf{U}}, \widetilde{\mathbf{u}}$)
Forward additive	$X, \xi, \Omega (0 \le \rho < 1)$	$2g_x(X+U) = g(\Omega+1) - g(\Omega-1), \Omega$ $-0.5 < X+U \le \Omega+0.5$	$g(X+U) = g(\xi) + g_\xi \rho, g_\xi = g(\xi+1) - g(\xi), \xi$ $= X + U - \rho$
Backward additive	x, ξ, Ω $(0 \le \rho < 1)$	$2G_X(x+u) = G(\Omega+1) - G(\Omega-1),$ $\Omega - 0.5 < x+u \le \Omega+0.5$	$G(x+u) = G(\xi) + G_\xi \rho,$ $G_\xi = G(\xi+1) - G(\xi),$ $\xi = x + u - \rho$
Forward compositional	X, ξ, Ω $(0 \le \rho < 1)$	$2g_x(X+\widetilde{U}) = g(\Omega+1) - g(\Omega-1),$ $\Omega - 0.5 < X+\widetilde{U} \le \Omega+0.5$	$g(X+\widetilde{U}) = g(\xi) + g_\xi \rho,$ $g_\xi(x) = g(\xi+1) - g(\xi),$ $\xi = X + \widetilde{U} - \rho$
Backward compositional	x, ξ, Ω $(0 \le \rho < 1)$	$2G_X(x+\widetilde{u}) = G(\Omega+1) - G(\Omega-1),$ $\Omega - 0.5 < x+\widetilde{u} \le \Omega+0.5$	$G(x+\widetilde{u}) = G(\xi) + G_\xi \rho,$ $G_\xi = G(\xi+1) - G(\xi),$ $\xi = x + \widetilde{u} - \rho$
Forward inverse compositional	$\widetilde{X}, \xi, \Omega$ $(0 \le \rho < 1)$	$2G_X(\widetilde{X}) = G(\Omega+1) - G(\Omega-1)^*,$ $\Omega \equiv \widetilde{X}$	$g(\widetilde{X}+\widetilde{U}) = g(\xi) + g_\xi \rho,$ $g_\xi(x) = g(\xi+1) - g(\xi),$ $\xi = \widetilde{X} + \widetilde{U} - \rho$
Backward inverse compositional	$\tilde{x}, \xi, \Omega$ $(0 \le \rho < 1)$	$2g_x(\tilde{x}) = g(\Omega+1) - g(\Omega-1)^*,$ $\Omega \equiv \widetilde{X}$	$G(\tilde{x}+\widetilde{u}) = G(\xi) + G_\xi \rho,$ $G_\xi = G(\xi+1) - G(\xi),$ $\xi = \tilde{x} + \widetilde{u} - \rho$

DIC, digital image correlation.
*Evaluated at the same integer pixels and thus computed only once.
For the case of one-dimensional affine deformation between two images, explicit expressions are given for computing the spatial gradients of image greyscales (third column) using a Prewitt gradient operator and image greyscales (fourth column) at non-integral pixel point locations using linear interpolation. The integral pixel coordinates used in the spatial gradient calculation formula and in the linear interpolation of image greyscales themselves may differ by one pixel.

more computationally efficient as the spatial gradient vector of only one image is needed. The so-called consistent mixed compositional algorithm appears to be more efficient as its computational cost is about the same as that of a normal additive or compositional algorithm given in Table 3: it needs subpixel interpolation of greyscales of only one image and computation of spatial gradient vectors of the same and only image. Both symmetric additive and compositional algorithms are found to be of the same computational cost, and it is higher than consistent inverse and mixed compositional algorithms (see Table 4). Again, the only one possible significant saving on the computational cost for consistent and symmetric algorithms is when the prior estimate of the deformation parameters are close enough to the final convergent solution, so the Hessian matrix in each closed-form solution no longer needs updating in the subsequent iteration steps.

Discussion

The deformation measurement based on DIC is essentially solving an inverse nonlinear optimization problem from a pair of digital images. Any practical DIC algorithm should thus be best stated explicitly as an iteratively numerical solution to the problem by *incrementally updating* the deformation parameters as given first by Lucas and Kanade [12]. For non-rigid deformation such as affine deformation considered in this work, one of the three possible methods (additive, compositional or inverse compositional) of incremental updating of the deformation mapping parameters should also be explicitly described in the formulation of a DIC algorithm [15] (see also Tables 1 and 2 in this paper). However, a closer attention has not been paid in the literature to the important fact that the parameterized displacement field or deformation mapping should be concisely defined with respect to a particular frame of reference or coordinates. For example, the presentation of the compositional image alignments by Baker and Matthews [15] is rather confusing as they use the deformation or warp parameters (vectors) $\mathbf{p}$ and $\Delta\mathbf{p}$ in their detailed discussion of the algorithms (for example, see their Equations (12)–(30a)). One may simply ask, if $\Delta\mathbf{p}$ are the increments of $\mathbf{p}$ as their notation implies, then why is the updating of $\mathbf{p}$ cannot be simply done as a vector addition $\mathbf{p} + \Delta\mathbf{p}$ here? As shown in Equations (10)–(12) in this paper, in fact, these two sets of warp parameters are measured against two different frames of reference, and they are better represented as different vectors with respect to

doi: 10.1111/str.12039

two different bases (as $\widetilde{\mathbf{P}}$ and $\Delta\hat{\mathbf{P}}$ in this paper), and the updating of the parameters $\mathbf{P}+\Delta\mathbf{P}$ for the deformation between the initial and current images thus has to be done by multiplication of their parameter matrices (i.e. via 'composing'). We also make a very careful distinction in this paper between the increments $\Delta\mathbf{P}$ in the resulting composed warp parameters $\mathbf{P}+\Delta\mathbf{P}$ and $\Delta\hat{\mathbf{P}}$, that is, they are not the same as well (see Equation (11b) in this paper).

The use of confusing notations is not only confined to the computer vision literature. For example, Zhou and Goodson [32] used $I_1(\mathbf{r})$ and $I_2(\mathbf{r})$ to designate the initial and current images and then stated their relations as $I_2(\mathbf{r}) = I_1[\mathbf{r} - \mathbf{U}(\mathbf{r})]$ and $I_1(\mathbf{r}) = I_2[\mathbf{r} + \mathbf{U}(\mathbf{r})]$, where $\mathbf{r} = \mathbf{X} = (X, Y)$ and $\mathbf{U}(\mathbf{r})$ is the forward displacement field of the initial image towards the current image. While there is nothing wrong with these two relations, one must understand that the pixel point $\mathbf{r}$ in $I_2(\mathbf{r})$ is physically different from the pixel point $\mathbf{r}$ in $I_1(\mathbf{r})$. Consequently, the simultaneous use of both $I_1(\mathbf{r})$ and $I_2(\mathbf{r})$ in their formulation of the DIC algorithm (see Sections 2.2 and 2.3 of their paper) is strictly incorrect. Similarly, the illustrations shown in Figures 1 and 2 by Avants *et al*. [16] for their symmetric and consistent cross-correlation image registration algorithms are rather confusing as the same pixel point $\mathbf{x}$ is used for both initial and current images as well. The approach used in this work to define each displacement field and deformation mapping concisely with respect to each specific frame of reference (integral-pixel coordinate system) is thus recommended.

As mentioned in the Introduction, DIC algorithms used so far in experimental mechanics is predominantly the forward additive type. One reason is that it is very natural to use the Lagrangian description of total motion and deformation based on the reference and deformed configurations in continuum mechanics [36]. Another reason is that any initial boundary value problem in continuum mechanics is often solved in the forward direction due to the dictate of thermodynamics and application needs. However, an image correlation analysis for deformation measurements does not have to be in the forward direction only as it involves only the kinematics of a continuum material element. It is thus not surprising that the image correlation analysis can be much more flexible in terms of either the total deformation or the incremental deformation mapping directions. As shown here for physically admissible affine deformation, one can always compose and/or invert multiple such affine parameter matrices to obtain the total forward (Lagrangian) transformation matrix for an image pair in the end. DIC algorithms other than the forward additive type may be more advantageous in some deformation measurement applications. For example, a backward additive DIC algorithm is used by Tong [26] and Tao *et al*. [27] to analyse the deformation field around a growing crack. The obvious advantage of such an approach is to avoid the use of a subset with a discontinuous displacement field as the subset based on the current image will have the cracked path and the free boundaries clearly defined. Another benefit of using a backward additive analysis is the reduced bias and variability in the displacement measurements when the initial image at the onset of a test is virtually noise-free [28]. It is often the case that one can record multiple image frames of a stationary sample before the onset of the experiment to obtain the frame-averaged initial image with little or much reduced random Gaussian white noises [31, 33]. In addition, the inverse compositional algorithm is shown to be much more computationally efferent [15], and it will be especially useful for analysing a set of images such as recorded sequentially during an experiment. Displacement measurements of better quality may also be obtained by using the forward version of the inverse compositional algorithm for the case of a noise-free initial image and noisy current images. As shown in Tables 3 and 5, the spatial greyscale gradient $G_{\mathbf{X}}(\widetilde{\mathbf{X}})$ will be noise-free and thus more accurate, similar to the situation in a backward algorithm [28]. In other words, although the final numerical solutions of the DIC algorithms are all equivalent, they are not identical in terms of bias and variability levels in the measured displacements using noisy digital images.

The (inverse) consistent and symmetric DIC algorithms have been proposed in the computer vision and medical image analysis literature for dealing with challenging image registration problems often involved images with low signal-to-noise ratios and very large heterogeneous deformation. They have so far seldom been used in experimental mechanics. One drawback of these algorithms is that the computational efficiency of inverse compositional algorithms is no longer available (see the third section and Table 4), although they are not much more costly than the additive or compositional algorithms. They may be nevertheless advantageous for some selected applications. It is well known that the success of a gradient-decent-based method such as the Gauss–Newton method in solving the minimization problem of an SSD coefficient depends on the quality of the initial guess. That is, the initial guess has to be within the radius of convergence for the gradient decent method to converge to the correct solution. The symmetric DICs would have the radius of convergence twice as large as the ones by other algorithms. A symmetric DIC may also be a natural choice for 3D surface deformation measurement applications using a dual-camera stereo vision system where the middle view (by a third camera) between a left camera and a right camera does not exist [1, 34]. Although all DIC algorithms discussed so far is for two-frame analysis applications, the consistent and symmetric formulations of the DIC algorithms can also be extended to a multi-frame DIC analysis. For example, the intermediate image in a symmetric DIC algorithm will no longer be a virtual one but can be an actual

doi: 10.1111/str.12039

middle image in a three-frame analysis. Both forward and backward inverse compositional algorithms may be used to achieve much better computational efficiency than the forward additive algorithm given by Cofaru *et al*. [35]. Although there is an implicitly imposed condition on the affine deformation parameters for physically admissible continuous deformation, the classic forward additive DIC algorithms do not enforce such a requirement explicitly at all. There may exist occasions when the image quality is poor, image deformation is large and the prior estimate of the deformation parameters fall outside the radius of convergence, so the resulting affine parameters may converge to a wrong solution that violates the condition for physically admissible affine deformation. The invertible affine deformation mapping condition is automatically embedded in all inverse consistent DIC algorithms so the resulting affine deformation parameters will more likely be physically admissible with such additional constraints. Additional research into this topic using actual examples of digital images will be warranted to verify if these algorithms will indeed perform better under such circumstances.

The focus in this study has been on the mathematical formulation of DIC algorithms with a particular emphasis on the total and incremental deformation mapping directions and their concise definitions with respect to the frame of reference (integral-pixel coordinate systems). An example of using the gradient-decent Gauss–Newton numerical solution method has also been discussed in some details. There are many other additional aspects of a DIC algorithm that will affect its computational efficiency and measurement quality. All SSD coefficients for the DIC algorithms discussed in this study (see Tables 1 and 2) have been defined in terms of as-recorded greyscales of initial and current images. One can use the normalised greyscales of these images to make the algorithms more robust against the photometric gain and bias that may exist in original greyscale images [1, 3, 5]. The use of the so-called zero-mean normalised sum-of-squared differences will incur some additional computational costs but will make the DIC analysis more robust. As shown in Tables 3 and 4, the actual implementation of a DIC algorithm needs also to specify how the image greyscales at non-integral pixel locations and their spatial gradients at both integral and non-integral pixel locations should be computed. This is related to the selection of a subpixel interpolation scheme (such as the example of linear interpolation given in Table 5). It is interesting to note that the computation of the spatial gradients of the image greyscales may not need to be strictly following the interpolation scheme used for the image greyscales. For example, it is rather common in the digital image processing literature to use a digital gradient filter or operator to approximate the spatial gradients of the image greyscales [19]. As shown in Table 5, the example of a central difference gradient (the so-called Prewitt operator instead of Robert operator in 1D) is used for the spatial gradient of the image greyscales at both integral and non-integral pixel locations. The effect of a particular interpolation scheme and spatial gradient operator used in the analysis on the accuracy and precision of displacement measurements will have to be evaluated case-by-case in combination with the specific algorithm and numerical solution method selected.

Another important aspect of a DIC algorithm is the choice of the deformation mapping types. In this study, only the physically admissible affine deformation mapping has been used because it has been widely used in many existing DIC algorithms, and its parameter matrix is invertible. The invertible parameter matrix (or the deformation mapping parameters form a group) is an important requirement in inverse compositional, consistent and symmetric DIC algorithms [15, 16]. A homography from projective space is another commonly used invertible transformation [15, 22]. As it is stated in the Introduction, applications in solid mechanics often have more stringent requirements on the accuracy and precision of displacement field measurements. The use of a larger image subset is often favoured in a DIC analysis because a larger n (the total number of pixels used) effectively reduces the variability of the displacement and displacement-gradient measurements due to image noises [29, 31]. The affine deformation mapping will not be a good approximation at all for such a large image subset undergoing non-affine deformation and the mismatch between the actual deformation and assumed affine deformation will induce significant systematic biases in the deformation measurements. Higher-order non-affine (and non-groupwise, in general) deformation mapping has to be used in these circumstances to obtain the displacement measurements over a large image subset at reduced bias and variability levels. Finally, all DIC algorithms discussed here are of a local type that uses a deformation mapping over a small image subset centred at a selected image pixel point, and the analysis repeats over many individual points with associated subsets to obtain the displacement field over the entire domain of interest in images. The formulation of various DIC algorithms presented in this study can, in principle, be readily applied to a global DIC analysis algorithm where a fixed finite-element mesh is prescribed over an entire domain of interest in images and nodal displacements as deformation parameters of all nodes of the finite-element mesh are solved together. In fact, the case of the so-called piecewise affine deformation mappings considered by Baker and Matthews [25] is basically a global analysis using linear triangular finite elements. It is interesting to know that the piecewise affine deformation mappings do not form a group, although Baker and Matthews [25] was able to extend nevertheless the inverse compositional algorithm to such non-group mappings to the first-order approximation in deformation parameters. Further research is needed to assess

doi: 10.1111/str.12039

if inverse compositional and other advanced algorithms discussed here can also be extended to more general types of non-groupwise deformation mappings in a local analysis and to other types of finite elements such as four-node bilinear elements and higher-order elements in a global analysis while maintaining the high measurement quality on the deformation parameters.

Summary

A forward additive DIC analysis, which is most commonly used in experimental mechanics, is to estimate the increments of forward deformation parameters of an image subset by minimising their SSD of greyscales between an image *numerically restored* from the current image and the initial image. A backward additive DIC analysis is to estimate the increments of backward deformation parameters of the image subset by minimising their SSD of greyscales between an image *numerically distorted* from the initial image and current image.

A compositional DIC analysis is to estimate the increments of deformation parameters of the image subset by minimising their SSD of greyscales between a pair of images that are *numerically restored* (from the current image) and *numerically distorted* (from the initial image), respectively, towards an intermediate image frame. The intermediate image itself is between the initial and current images and is moved incrementally in the iterative solution process towards either the initial or the current image. Among the two types of compositional algorithms, the inverse compositional version is shown to be computationally most efficient of all algorithms considered here while maintaining the same or even superior accuracy and precision in displacement measurements.

A consistent DIC analysis is a joint forward and backward analysis in which both forward deformation and backward deformation are made to be consistent (that is, their transformation or parameter matrices are the inverse of each other). A symmetric DIC analysis is to estimate the increments of deformation parameters of the image subset by minimising their SSD of greyscales between a pair of images that are respectively *numerically restored* (from the current image) and *numerically distorted* (from the initial image) towards a fixed image frame. The fixed image is chosen to be the symmetric image that is exactly half-way in terms of relative deformation between the initial and current images. While both consistent and symmetric algorithms are in general computationally more costly, they may be more robust and less biased for some materials testing and structural monitoring applications.

The equivalency among the algorithms has been proved but only within the limit of first-order numerical approximations and within the measurement errors due to image noises. Among the remaining questions in comparing various algorithms collected in this paper are (a) whether or not they achieve the same level of accuracy for a given image pair and (b) what is the applicability of each algorithm for a particular deformation measurement situation. The general answer to the first question appears to be no: for example, the backward additive algorithm has no noise-induced bias in displacements in comparison with the forward additive algorithm if one uses a noise-free initial image [28]. The answer would be specific to each individual algorithm and its implementation. For example, as far as the author knows, there is no study on comparing the displacement measurement accuracy between the forward additive and inverse compositional algorithms. The second question appears to a little more challenging: it is well known in the computer vision community that although the inverse compositional algorithm is fastest (that is, it is ideal for real-time applications) but is less flexible in terms of subset deformation models (being limited to groupwise deformation such as affine and homographic warps only [15]). Several attempts to extend the algorithm to higher-order non-groupwise deformation fields (such as a quadric or cubic polynomial deformation field) have so far not been successful. The use of central difference for spatial gradients of greyscales and linear interpolation for subpixel greyscale computations are only given as an example about how each algorithm may be implemented for a one-dimensional case. Just like the use of the Gauss–Newton numerical method, they are used as means to provide an introductory step to anyone who is interested in implementing and evaluating any of the algorithms collected in this paper. The effect of particular choices of spatial gradient routine and subpixel interpolation scheme on the measurement errors using a particular numerical method in a particular type of DIC algorithms requires an in-depth theoretical analysis, numerical verification and experimental validation. Hopefully, this work can serve as a useful guide for interested researchers to take a closer look at these non-classic DIC algorithms and help to develop new robust and efficient implementations for accurate deformation measurements in experimental mechanics applications.

REFERENCES

1. Sutton, M. A., Orteu, J. J. and Schreier, H. W. (2009) Image Correlation for Shape, Motion and Deformation Measurements. Springer: New York.
2. Bornert, M., Brémand, F., Doumalin, P., *et al*. (2009) Assessment of digital image correlation measurement errors: methodology and results. *Exp. Mech.* **49**, 353–370.
3. Pan, B., Qian, K. M., Xie, H. M. and Asundi, A. (2009) Two-dimensional digital image correlation for in-plane displacement and strain measurement: a review. *Meas. Sc. Technol.* **20**, 062001.
4. Vendroux, G. and Knauss, W. G. (1998) Submicron deformation field measurements: part 2. Improved digital image correlation. *Exp. Mech.* **38**, 86–92.
5. Tong, W. (2005) An evaluation of digital image correlation criteria for strain mapping applications. *Strain* **41**, 167–175.

doi: 10.1111/str.12039

6. Peters, W. H. and Ranson, W. F. (1982) Digital imaging techniques in experimental stress analysis. *Opt. Eng.* **21**, 427–431.
7. Peters, W. H., Ranson, W. F., Sutton, M. A., Chu, T. C. and Anderson, J. (1983) Application of digital correlation methods to rigid body mechanics. *Opt. Eng.* **22**, 738–742.
8. James, M. R., Morris, W. L. and Cox, B. N. (1990) A high accuracy automated strain-field mapper. *Exp. Mech.* **30**, 60–67.
9. Franke, E. A., Wenzel, D. J. and Davison, D. L. (1991) Measurement of microdisplacements by machine vision photogrammetry. *Rev. Sci. Instrum.* **62**, 1270–1279.
10. Sutton, M. A., Cheng, M. Q., Peters, W. H., Chao, Y. J. and McNeill, S. R. (1986) Application of an optimized digital image correlation method to planar deformation analysis. *Image Vis. Comput.* **4**, 143–150.
11. Bruck, H. A., McNeill, S. R., Sutton, M. A. and Peters, W. H. (1989) Digital image correlation using Newton–Raphson method of partial differential correction. *Exp. Mech.* **29**, 261–267.
12. Lucas, B. D. and Kanade, T. (1981) An iterative image registration technique with an application to stereo vision. *Proceedings of the 7th International Joint Conference on Artificial Intelligence* (*IJCAI '81*), 674–679.
13. Davis, C. Q. and Freeman, D. M. (1998) Statistics of subpixel registration algorithms based on spatiotemporal gradients or block matching. *Opt. Eng.* **37**, 1290–1298.
14. Christensen, G. E. and Johnson, H. J. (2001) Consistent image registration. *IEEE Tran. Medical Imaging* **20**, 568–582.
15. Baker, S. and Matthews, I. (2004) Lucas-Kanade 20 years on: a unifying framework. *Int. J. Computer Vision* **56**, 221–255.
16. Avants, B. B., Epstein, C. L., Grossman, M. and Gee, J. C. (2008) Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain. *Med. Image Anal.* **12**, 26–41.
17. Horn, B. K. P. and Schunk, B. G. (1981) Determining optical flow. *Artif. Intell.* **17**, 185–204.
18. Barron, J. L., Fleet, D. J. and Beauchemin, S. S. (1994) Performance of optical flow techniques. *Int. J. Computer Vision* **12**, 43–77.
19. Jain, R., Kasturi, R. and Schunck, B. G. (1995) Machine Vision. McGraw-Hill, New York.
20. Zhang, Z. (1997) Motion and structure from two perspective views: from essential parameters to Euclidean motion via fundamental matrix. *J Opt. Soc, of America A* **14**, 2938–2950.
21. Hager, G. D. and Belhumeur, P. N. (1998) Efficient region tracking with parametric models of geometry and illumination. *IEEE Trans. Patt. Ana. Mach. Intell.* **20**, 1025–1039.
22. Shum, H.-Y. and Szeliski, R. (2000) Construction of panoramic image mosaics with global and local alignment. *Int. J.Computer Vision* **16**, 63–84.
23. Westerweel, J. (1997) Fundamentals of digital particle image velocimetry. *Meas. Sci. Technol.* **8**, 1379–1392.
24. Stanislas, M., Okamoto, K. and Kahler, C. (2003) Main results of the first international PIV challenge. *Meas. Sci. Technol.* **14**, R63–R89.
25. Baker, S. and Matthews, I. (2001) Equivalence and efficiency of image alignment algorithms. *Proceedings of the IEEE Conf. Computer Vision and Pattern Recognition*, vol. **1**, 1090–1097.
26. Tong, W. (2004) An adaptive backward image correlation technique for deformation mapping of a growing crack in thin sheets. *Exp. Tech.* **28**, 63–67.
27. Tao, H., Zavattieri, P. D., Hector, Jr. L. G. and Tong, W. (2010) Mode I fracture at spot welds in dual-phase steel: an application of reverse digital image correlation. *Exp. Mech.* **50**, 1199–1212.
28. Tong, W. (2011) Subpixel image registration with reduced bias. *Opt. Lett.* **36**, 763–765.
29. Tong, W., Yao, H. and Xuan, Y. (2011) An improved error evaluation in one-dimensional deformation measurements by linear digital image correlation. *Exp. Mech.* **51**, 1019–1031.
30. Birchfield, S. (1997) Derivation of Kanade-Lucas-Tomasi tracking equation. http://www.ces.clemson.edu/~stb/klt/birchfield-klt-derivation.pdf.
31. Wang, Y. Q., Sutton, M. A., Bruck, H. A. and Schreier, H. W. (2009) Quantitative error assessment in pattern matching: effects of intensity pattern noise, interpolation, strain and image contrast on motion measurements. *Strain* **45**, 160–178.
32. Zhou, P. and Goodson, K. E. (2001) Subpixel displacement and deformation gradient measurement using digital image speckle correlation (DISC). *Opt. Eng.* **40**, 1613–1620.
33. Smith, B. W., Li, X. and Tong, W. (1998) Error assessment for strain mapping by digital image correlation. *Exp. Tech.* **22**, 19–21.
34. Tong, W. (2004) Plastic strain mapping of bent sheets by image correlation. *Exp. Mech.* **44**, 502–511.
35. Cofaru, C., Philips, W. and Van Paepegem, W. (2012) A three-frame digital image correlation (DIC) method for the measurement of small displacements and strains. *Meas. Sci. Technol.* **23**, 105406.
36. Asaro, R. J. and Lubarda, V. A. (2006) Mechanics of Solids and Materials. Cambridge University Press, New York.

doi: 10.1111/str.12039