RESEARCH ARTICLE

3D optical flow for large CT data of materials microstructures

Tessa Nogatz[1,2] 🄳 | Claudia Redenbach[1] | Katja Schladitz[2]

[1]Department of Mathematics, Technische Universität Kaiserslautern, Kaiserslautern, Germany

[2]Department of Image Processing, Fraunhofer ITWM, Kaiserslautern, Germany

Correspondence
Tessa Nogatz, Department of Mathematics, Technische Universität Kaiserslautern, Germany.
Email: nogatz@mathematik.uni-kl.de

Funding information
MathApp

Abstract

We compute three-dimensional displacement vector fields to estimate the deformation of microstructural data sets in mechanical tests. For this, we extend the well-known optical flow by Brox et al. to three dimensions, with special focus on the discretization of nonlinear terms. We evaluate our method first by synthetically deforming foams and comparing against this ground truth and second with data sets of samples that underwent real mechanical tests. Our results are compared to those from state-of-the-art algorithms in materials science and medical image registration. By a thorough evaluation, we show that our proposed method is able to resolve the displacement best among all chosen comparison methods.

KEYWORDS
digital volume correlation, full field displacement, in situ testing, motion estimation, optical flow

1 | INTRODUCTION

The estimation of load induced motion or displacement in three-dimensional volumes is one of the core tools to characterize the behaviour of complex materials. In in situ testing, for example, a mechanical test is performed, and computed tomography (CT) images are acquired in parallel. This method gives better insight into a material's nature, especially when the behaviour is governed by the microstructure of the material. Macroscopically, the evolution of a material during the test can be described and monitored rather easily: elastic materials for instance elongate when tension is applied, or shrink when compressed. For materials like foams, however, though they seem to behave elastically at first glance, a closer look into the microstructure gives better insight into the mechanisms of failure. For instance, metal-matrix composite (MMC) foams deform in compression tests first elastically before full layers collapse. The failure of the foam is preceded by breaking struts. A change on a small scale therefore influences the stability of the whole component. A similar observation can be made for concrete: fault zones are generally preceded by a large number of microfractures. CT devices that generate images with voxel sizes in the micrometre range (μCT) are a mean to resolve this microstructural behaviour. Nevertheless, the challenge to quantify this deformation on the micrometre scale remains.

Unfortunately, this task is very complex. Even when the imaged components are manufactured in sizes of a few millimetres to centimetres, the resulting CT data sets may have a total number of up to 2048^3 voxels in 16-bit grey value range. Therefore, an algorithm is required that is not only efficient in time but also in its use of memory.

However, the size of the data sets is not the only challenge that comes with the analysis of microstructures via CT. On a macroscopic scale, the material can look very homogeneous. For instance, examining a volume rendering of a CT scan of long fibre reinforced thermoplastics (LFT) in Figure 1a, one might assume that single fibres are aligned very genuinely. Yet Figure 1b, which is a slice view of exactly the same data set, reveals that the material has layers that behave completely differently.

Another challenge is unveiled immediately, too: the quality of classical algorithms for estimating deformation fields in two or three dimensions increases if the image displays many different shapes with distinct grey values. In CT data of materials, one hardly ever observes more than a few distinct intensity levels.

In addition, estimation in medical images often benefits from the fact that the objects which undergo the motion are fully contained in the image. If, for example, a CT of the head is aligned to another one, the outer boundary is already a very powerful feature to support the registration process. Further, these features can be used to generate landmarks to support the process: either a doctor sets the landmarks intuitively manually[1] or an algorithm constructs landmarks automatically based on a clearly detectable feature.[2] In contrast, constructing landmarks is highly challenging, if not impossible, for materials samples. Figure 1 shows a typical example lacking both, an outward bound of the imaged structure and unique structural features as candidates for landmarks.

In fact, even with in situ testing, it is not guaranteed that the scanned areas before and after the material test coincide. A meaningful test for example might demand a certain thickness of the material, so that the material bears a minimal required load. If a foam or a fibre system is eventually manufactured in components with thickness of several centimetres, performing tests on few layers that cover only micrometres may not be meaningful. Therefore, to guarantee the microstructure to be properly resolved, only parts of the sample are scanned. If the material is only slightly elastic in this case, the boundaries in both scans do not coincide anymore.

In summary, an algorithm for estimating displacement has to fulfil the following requirements: first, our method must be a local one in the sense that local behaviour such as breaking struts or delamination of fibres is resolved accurately. Next, our algorithm needs to work robustly, even if the volume images do not contain very rich features. And last, the algorithm must yield a valid displacement field, even if the fields of view of the images do not coincide.

We propose to tackle these challenges by an extension of the famous optical flow of Brox et al[3] to three dimensions. Even though the approach is originally designed for 2D images acquired by camera sensors, we show that this extension is very suitable for computing displacement fields in CT images of microstructural data. We compare it to state-of-the-art methods in materials science and medical image processing and demonstrate that our method is able to outperform both.

The remainder of this paper is organized as follows. In Section 2, we give a short introduction into the state-of-the-art, not only in materials science but also in the area of medical image registration. Both areas solve the task of calculating deformation fields in 3D volumes in related yet different ways. We compare several representatives to our approach. In Section 3, we introduce the theoretical background and extend the work of Brox et al[3] to 3D. Following recent work on optical flow calculation, we slightly alter the computation of image derivatives. Section 4 deals with the numeric challenges that come along with the 3D implementation. In Section 5, we introduce four data sets that are used to test our algorithm, namely, (i) synthetically deformed foam data, (ii) CT images of metal foams deformed by compression tests, (iii) CT data of glass foams with very thin struts under compression, and (iv) CT images of fibre reinforced
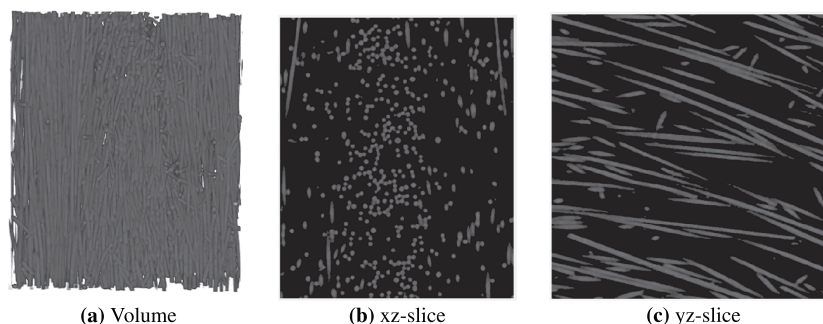

(a) Volume   **(b)** xz-slice   **(c)** yz-slice

FIGURE 1 Volume and slice view of the fibre system in a long glass fibre reinforced composite (LFT). Sample Sabic. Imaging Fraunhofer ITWM. Image size $500 \times 600 \times 500$ at voxel edge length 5 μm

polymer samples deformed by tensile tests. In Section 6, we present our results on the aforementioned data and compare them to the state-of-the-art methods described in Section 2. Finally, we conclude our work in Section 7.

2 | RELATED WORK

Before starting with a short review on algorithms dealing with motion estimation, we introduce the basic notation. We consider images $I_0 : \Omega \to \mathbb{R}^3$ and $I_1 : \Omega \to \mathbb{R}^3$, where Ω is a bounded subset of $\mathbb{R}^3$. I_0 will be the image of the unloaded specimen, while I_1 shows the loaded one. We seek a displacement $\mathbf{u} : \Omega \subset \mathbb{R}^3 \to \mathbb{R}^3$ such that ideally

$$I_0(\mathbf{x}) = I_1(\mathbf{x} + \mathbf{u}(\mathbf{x})). \tag{1}$$

Algorithms dealing with these problem in three dimensions can be mainly assorted into two groups, governed by the area of application: in materials science, the algorithms are known under the name *digital volume correlation* (DVC) and usually describe motion originating from mechanical tests. In medical image processing *medical image registration* (MIR) is used to monitor motion of inner organs or evolution of tumours. We will see that common representatives of both methods will violate the requirements postulated in the previous chapter.

The first works in DVC can be traced back to 1999 and are by BK Bay,[4] whose aim was to compute displacement fields for strain calculation based on CT data. The data were acquired during tensile tests of trabecular bone taken from animal carcasses. The original method minimizes the sum-of-squared differences (SSD) of intensities among a set of predefined subvolumes that subdivide the image domain Ω. That is, for each individual subvolume with centre $\mathbf{x}$, one computes

$$\min_{\mathbf{u}} \sum_{i=1}^{K} (I_1(\mathbf{x} + m_i + \mathbf{u}(\mathbf{x})) - I_0(\mathbf{x} + m_i))^2, \tag{2}$$

where m_i are the possible offsets within the subvolume, meaning that the sum is computed over all voxels in the subvolume. This problem is highly ill-posed. To overcome this, the initial work of Bay[4] limits the maximal admissible displacements and seeks the optimum for each subvolume and not voxelwise.

If the data sets feature outliers, such as noise or small brittle structures that detach during the test, more robust similarity terms have to be introduced. Hall[5] exchanges the SSD by normalized cross-correlation, again individually for each subvolume, and computes

$$\min_{\mathbf{u}} \frac{\sum_{i=1}^{K} I_1(\mathbf{x} + m_i + \mathbf{u}(\mathbf{x})) I_0(\mathbf{x} + m_i)}{\sqrt{\sum_{i=1}^{K} I_1(\mathbf{x} + m_i + \mathbf{u}(\mathbf{x}))^2 \sum_{i=1}^{K} I_0(\mathbf{x} + m_i)^2}}. \tag{3}$$

Note that also the zero-mean normalized cross-correlation

$$\min_{\mathbf{u}} \frac{\sum_{i=1}^{K} \left(I_1(\mathbf{x} + m_i + \mathbf{u}(\mathbf{x})) - \overline{I_1}\right) \left(I_0(\mathbf{x} + m_i) - \overline{I_0}\right)}{\sqrt{\sum_{i=1}^{K} \left(I_1(\mathbf{x} + m_i + \mathbf{u}(\mathbf{x})) - \overline{I_1}\right)^2 \sum_{i=1}^{K} \left(I_0(\mathbf{x} + m_i) - \overline{I_0}\right)^2}}, \tag{4}$$

where $\overline{I}$ denotes the mean, is widely used in DVC due to is increased robustness.

In the last decade, the focus was put on improvement of the optimization. First, a distinction between *additive* and *compositional* approaches was made. When choosing DVC with SSD like norms as in Equation (2), the problem can be considered a least squares minimization problem. Because of the large number of unknowns in DVC, the optimization of the above problem is almost always done via iterative schemes. *Compositional* in this context means that after each iteration of the solution algorithm, all terms involving the displacement to be optimized are updated with the result from the iteration. This specifically involves all terms of the form $I_1(\mathbf{x} + m_i + \mathbf{u}(\mathbf{x}))$, so images evaluated at a displaced position. More detail on how and where to incorporate compositional methods can be found in Baker et al.[6] In this context, the method of Bay et al[4] can be classified as an additive Levenberg–Marquardt formulation. The inverse

compositional Gauss–Newton (IC-GN) scheme of Pan et al[7] then led to significant improvement in accuracy and speed. The advances concerning speed became also necessary, because the authors did not only consider plain displacements, but also first order displacement gradients. These add nine unknowns for each calculation point. We want to emphasize that these additional unknowns will not be considered in our algorithm. The reason for this is that first (or higher order) displacement gradients approximate rotation and shear. We for now aim for a plain displacement formulation and therefore neglect higher order gradient terms. For an example of DVC that particularly aims to incorporate higher order gradients while using an IC-GN, see Wang et al.[8] Further improvements on IC-GN, especially aiming for fast and memory efficient computations on high resolution images, were made by Pan et al.[9]

Another approach, which can be found under the name *global DVC*, is to not seek the optimal displacement for subvolumes but in terms of basis functions. Leclerc et al[10] propose to model the displacement field in terms of basis functions $\psi_1, \dots, \psi_M$. Each vector valued basis function then gets assigned a coefficient $u_1, \dots, u_M$, and the displacement is then given by

$$\mathbf{u}(\mathbf{x}) = \sum_n u_n \psi_n(\mathbf{x}), \tag{5}$$

The dependence on the subvolume is omitted, and the minimization therefore reduces to finding the scalar coefficients

$$\min_{u_1, \dots, u_n} \left(I_1(\mathbf{x} + \sum_n u_n \psi_n(\mathbf{x})) - I_0(\mathbf{x}) \right)^2. \tag{6}$$

An appropriate basis can be found in terms of finite elements (FE). If the displacement is not expected to be discontinuous, trilinear polynomials associated with 8-node cube elements can be used. The problem then reduces to calculating suitable coefficients.

Recently, Yang at al.[11] proposed to combine both local and global DVC by an augmented Lagrangian approach. In their work, which we will abbreviate with ALDVC, two subproblems are solved in an alternating manner: first, a local, IC-GN approach is used to calculate accurate subvolume displacements. Then these displacements are unified by a global, FE-based DVC step. To do so, an auxiliary variable $\hat{\mathbf{u}}$ is introduced and incorporated in the optimization functional as follows. Let the image domain Ω be separated into subvolumes such that $\bigcup_j \Omega_j = \Omega$. Then for each subvolume the classical DVC SSD assumption (2) is augmented by terms relating $\mathbf{u}_j$ to the auxiliary variable $\hat{\mathbf{u}}$, and a kinematic compatibility constraint, namely, that $\mathbf{F}_j = \nabla \hat{\mathbf{u}}\left(\mathbf{x}_j^0\right)$ and $\mathbf{x}_j = \hat{\mathbf{u}}(\mathbf{x}_j^0)$, where $\mathbf{x}_j^0$ denotes the centre point of the subvolume Ω_j. Here, the $\mathbf{F}_j$ will serve as auxiliary variables and are subject to the optimization as well $\hat{\mathbf{u}}_j$ and $\mathbf{u}_j$. Therefore, the following functional is computed such that it is minimal with respect to $\mathbf{u}_j$

$$\int_{\Omega_j} \left(I_1(\mathbf{x} + \mathbf{u}_j(\mathbf{x}) + \mathbf{F}_j\left(\mathbf{x} - \mathbf{x}_j^0\right)) - I_0(\mathbf{x}) \right)^2 + \frac{\mu}{2}\left|\hat{\mathbf{u}}_j - \mathbf{u}_j\right|^2 + \lambda\left(\hat{\mathbf{u}}_j - \mathbf{u}_j\right) + \frac{\alpha}{2}\left|\nabla\hat{\mathbf{u}}_j - \mathbf{F}_j\right|^2 + \beta_j : \left(\nabla\hat{\mathbf{u}}_j - \mathbf{F}_j\right) d\mathbf{x}. \tag{7}$$

Now the locally computed unrelated subvolumes are correlated globally by the sum over all subvolumes:

$$\sum_j \int_{\Omega_j} \left(I_1(\mathbf{x} + \mathbf{u}_j(\mathbf{x}) + \mathbf{F}_j\left(\mathbf{x} - \mathbf{x}_j^0\right)) - I_0(\mathbf{x}) \right)^2 + \frac{\mu}{2}\left|\hat{\mathbf{u}}_j - \mathbf{u}_j\right|^2 + \lambda\left(\hat{\mathbf{u}}_j - \mathbf{u}_j\right) + \frac{\alpha}{2}\left|\nabla\hat{\mathbf{u}}_j - \mathbf{F}_j\right|^2 + \beta_j : \left(\nabla\hat{\mathbf{u}}_j - \mathbf{F}_j\right) d\mathbf{x}. \tag{8}$$

Both steps are now solved in an alternating manner, where $\hat{\mathbf{u}}_j$ is kept fix in (7) and $\mathbf{u}_j$ is kept fix in (8).

Nevertheless, DVC suffers from several drawbacks. Firstly, the procedure heavily depends on the predefined maximal displacement, which might be unknown. Setting the maximal value more generously results in an increase of complexity, hence computational time. Secondly, as the algorithm is highly dependent on the predefined subvolumes, it only computes average displacements among those.[12] Also, because the computation for each subvolume happens more or less independent from each other, the resulting displacement field may not be a kinematic compatible one.[11] This can only be achieved via the global approach, where all nodes are connected via the finite element grid. But even

in the global approach the computed displacement is an average one, as the chosen finite elements again do not coincide with the voxel grid. For foams, however, the average displacement can be completely contrary to the local behaviour. For instance, locally, one will observe a breaking strut, while the whole subvolume is displaced along the loading direction. This also results in mediocre quality when deriving full field displacements: DVC does not compute displacements for each voxel but returns an average for the centre points x of the subvolumes. Therefore, when derive full field displacement, the computed values have to be interpolated to obtain information at every voxel. This argument still holds for the case of the ALDVC method, though it certainly produces a more accurate displacement field than local and global displacement individually.

The variety of algorithms used to estimate motion in medical data sets is manifold. A thorough introduction is given in Modersitzki.[13] We want to focus on a special class of algorithms, namely, *large deformation diffeomorphic mapping* (LDDMM). We refer to[14] for a general introduction. Here, we focus on the work of Mang et al.[15] The implementation of this specific algorithm is applicable to relatively large data sets and available in the open-source package CLAIRE.

The main conceptual difference to DVC in this and related approaches is not to limit the maximal displacement and the subvolumes, but to pose an additional regularization term, denoted by $\mathcal{S}$, to stabilize the computation. Another core difference is to not consider displacements, but to seek for a transformation function $\phi : \Omega \subset \mathbb{R}^3 \to \mathbb{R}^3$. Here, voxels are considered as particles, and such a transformation shall describe the position of each particle at some time t. The transformation is given via an ordinary differential equation $\partial_t \phi_t = v \circ \phi_t$, where v is the velocity of the particle, $\circ$ is the composition. At time $t = 0$ such a transformation clearly has to be the identity, $\phi_0 = id$, and we fix the endpoint to be at $t = 1$, such that $I_0 \circ \phi_1 = I_1$. Under certain conditions, the transformation is unique, see[14] for an overview of the results on this question. Note that this approach produces diffeomorphic (differentiable with a differentiable inverse) transformations. To relate the transformation to the images which undergo the sought motion, a distance measure $\mathcal{D}$ is introduced. To guarantee the previously mentioned conditions for a unique transformation, a regularization $\mathcal{S}$ on the velocities is incorporated. Overall, the deformation in this framework is computed via solving

$$\arg \min_{\phi} \mathcal{D}(I_0 \circ \phi, I_1) + \mathcal{S}(v). \tag{9}$$

Mang et al[16] measure distance by the squared $\mathcal{L}_2$-norm

$$\mathcal{D}(I_0 \circ \phi, I_1) = \frac{1}{2} \int_{\Omega} ((I_0 \circ \phi)(\mathbf{x}) - I_1(\mathbf{x}))^2 \, d\mathbf{x}, \tag{10}$$

which is the continuous equivalent of the SSD. The regularizer $\mathcal{S}$ usually consists of higher order terms. Due to the complexity of this kind of terms, we refer to the original work of Mang et al[16] for a precise formulation.

As a consequence of the use of higher order terms, the methods are highly memory consuming. Therefore, they are only suitable for data sets consisting of not more than $1\,000^3$ voxels. In addition, the formulation of the deformation as a diffeomorphic mapping produces very smooth displacements. Both restrictions are not problematic for medical image registration. Medical CT data are typically below the size limit and displacements, e.g. by breathing, are suitably smooth. In materials science, larger image sizes and discontinuous behaviour such as crack formation may pose additional challenges.

To summarize, none of the presented approaches suffices our requirements. Where DVC computes averages and cannot display small scale behaviour, MIR highly makes use of the underlying human nature of the data set and is therefore not able to display the discontinuous behaviour in our data set. We therefore decided to use an optical flow based approach and extend it to three dimensions. Optical flow has its origins in the work of Horn and Schunck[17] and Lucas and Kanade.[18] Manifold improvements and new concepts have been introduced since then. The recent development with few exceptions mainly focuses on the incorporation of machine learning. This guarantees realtime results, which are of special interest in the growing area of autonomous driving.[19,20] In our application, we neither require realtime computations, nor is there any suitable training data to use for deep learning approaches. We therefore decided to use one of the celebrated methods of the last decade, the high accuracy optical flow by Brox et al[3] and extend it to 3D. In our opinion, this algorithm balances accuracy and cost of optimization very well. It is therefore ideal to be applied to the estimation of displacement vector fields in materials testing.

Note that a modified, 3D version of the algorithm of Brox et al. was already proposed to compute displacement in lung samples induced by respiratory motion.[21] Details of this approach and differences to our method will be discussed later.

3 | LARGE DISPLACEMENT OPTICAL FLOW

In this section we extend the work of Brox et al[3] to 3D. The basis of optical flow is the assumption that each voxel in an image can be displaced to a voxel of the same grey value in the second image. Formally, the so-called constancy assumption for images $I_0, I_1 : \Omega \subset \mathbb{R}^3 \to \mathbb{R}$ can be written as

$$I_0(\mathbf{x}) = I_1(\mathbf{x} + \mathbf{u}(\mathbf{x})), \tag{11}$$

with a suitable displacement $\mathbf{u} : \Omega \subset \mathbb{R}^3 \to \mathbb{R}^3$. Note that we write $\mathbf{u}$ in bold letters for the vector field, while the components are denoted by (u, v, w). In addition, for the sake of simplicity, we omit the dependence of $\mathbf{u}$ on $\mathbf{x}$. Nevertheless, note that our aim is to compute a displacement for every position $\mathbf{x}$ in the image domain.

Brox et al[3] solve the optical flow equation as a variational problem; therefore, the above equation enters via

$$E_{\text{Data}}(\mathbf{u}) = \int_\Omega \Psi(|I_0(\mathbf{x}) - I_1(\mathbf{x} + \mathbf{u})|) \, d\mathbf{x}, \tag{12}$$

where $| \cdot |$ is the Euclidean norm. The function $\Psi(s) = \sqrt{s^2 + \epsilon^2}, \epsilon = 0.0001$ smooths the norms to simplify the computation of optimality conditions. This cost function alone is not sufficient to compute a valid displacement field, as the problem is ill-posed. The classical way to overcome this problem is to pose a regularization constraint on the vector field. Brox et al. propose to use the so-called smoothness constraint (or total variation regularization)

$$E_{\text{Smooth}}(\mathbf{u}) = \int_\Omega \Psi(|\nabla u| + |\nabla v| + |\nabla w|) \, d\mathbf{x}. \tag{13}$$

The novelty of Brox et al[3] is a third penalty on differences in gradients of both images. The original purpose of the so-called gradient constancy assumption was to enhance robustness with respect to changes in illumination. We noticed that in our applications, this term puts special emphasis on features, mainly on prominent edges. It connects the gradients of the original image and the deformed image by

$$E_{\text{Grad}}(\mathbf{u}) = \int_\Omega \Psi(|\nabla I_0(\mathbf{x}) - \nabla I_1(\mathbf{x} + \mathbf{u})|) \, d\mathbf{x}. \tag{14}$$

The improvement caused by this additional term can be seen in Section 6.1. The variational problem now consists of finding a solution $\mathbf{u} = (u, v, w)$, such that

$$E_{\text{total}}(\mathbf{u}) = E_{\text{Data}}(\mathbf{u}) + \lambda E_{\text{Grad}}(\mathbf{u}) + \mu E_{\text{Smooth}}(\mathbf{u}) \tag{15}$$

is minimal. λ and μ are regularization parameters and are chosen problem dependent, but larger than 0.

We now shortly come back to the already mentioned 3D extension of optical flow by Hermann et al.[21] In their approach, E_{Grad} and E_{Smooth} are used as here, while E_{Data} is replaced by a census cost transform. The use of this transform within optical flow in two dimensions was also investigated in Hafner et al[22] by one of the authors of Brox et al.[3] There, the census transform does not replace the data term but is directly related to the gradient constancy by its continuous limit. Moreover, the census transform is shown to perform better than the classical gradient constancy assumption only in case of strong illumination changes. It can therefore be considered as a *more robust* version. The price for this increased robustness nevertheless is the loss of information. The census cost transform is a binary measure

and therefore loses parts of already sparse information of our data. These two reasons, the loss of information and the fact that CT data usually does not suffer from strong illumination changes, caused us to consider the 3D extension of the original work, consisting of the constancy assumption, the gradient constancy assumption and the smoothness constraint.

As in Brox et al,[3] the optimality conditions are computed componentwise for (u, v, w) by Euler–Lagrange equations

$$
\begin{aligned}
0 = \ & \Psi'(I_d^2)I_d I_x \\
& + \lambda \Psi'(I_{xd}^2 + I_{yd}^2 + I_{zd}^2)(I_{xx}I_{xd} + I_{xy}I_{yd} + I_{xz}I_{zd}) \\
& + \mu \mathrm{div}(\Psi'(|\nabla u| + |\nabla v| + |\nabla w|)\nabla u)
\end{aligned}
\tag{16}
$$

$$
\begin{aligned}
0 = \ & \Psi'(I_d^2)I_d I_y \\
& + \lambda \Psi'(I_{xd}^2 + I_{yd}^2 + I_{zd}^2)(I_{xy}I_{xd} + I_{yy}I_{yd} + I_{yz}I_{zd}) \\
& + \mu \mathrm{div}(\Psi'(|\nabla u| + |\nabla v| + |\nabla w|)\nabla v)
\end{aligned}
\tag{17}
$$

$$
\begin{aligned}
0 = \ & \Psi'(I_d^2)I_d I_z \\
& + \lambda \Psi'(I_{xd}^2 + I_{yd}^2 + I_{zd}^2)(I_{xz}I_{xd} + I_{yz}I_{yd} + I_{zz}I_{zd}) \\
& + \mu \mathrm{div}(\Psi'(|\nabla u| + |\nabla v| + |\nabla w|)\nabla w)
\end{aligned}
\tag{18}
$$

where

$$
\begin{aligned}
I_x &= \partial_x I_1(\mathbf{x} + \mathbf{u}) & I_{xx} &= \partial_{xx} I_1(\mathbf{x} + \mathbf{u}) \\
I_y &= \partial_y I_1(\mathbf{x} + \mathbf{u}) & I_{xy} &= \partial_{xy} I_1(\mathbf{x} + \mathbf{u}) \\
I_z &= \partial_z I_1(\mathbf{x} + \mathbf{u}) & I_{xz} &= \partial_{xz} I_1(\mathbf{x} + \mathbf{u}) \\
I_d &= I_0(x) - I_1(\mathbf{x} + \mathbf{u}) & I_{yy} &= \partial_{yy} I_1(\mathbf{x} + \mathbf{u}) \\
I_{xd} &= \partial_x I_d & I_{yz} &= \partial_{yz} I_1(\mathbf{x} + \mathbf{u}) \\
I_{yd} &= \partial_y I_d & I_{zz} &= \partial_{zz} I_1(\mathbf{x} + \mathbf{u}) \\
I_{zd} &= \partial_z I_d.
\end{aligned}
\tag{19}
$$

To solve this equation, a fixed point iteration is derived. The goal is that, after linearization and simplification, we derive an update scheme of the form $u^{k+1} = u^k + f(u^k)$, where f consists roughly of the terms in (16). As a start, all terms that depend on u, v or w directly or via subindex d in image derivatives are equipped with an index $k+1$, and all remaining terms with an index k, here shown exemplarily for u:

$$
\begin{aligned}
0 = \ & \Psi'((I_d^{k+1})^2)I_d^{k+1} I_x^k \\
& + \lambda \Psi'\left((I_{xd}^{k+1})^2 + (I_{yd}^{k+1})^2 + (I_{zd}^{k+1})^2 \right) \\
& \left(I_{xx}^k I_{xd}^{k+1} + I_{xy}^k I_{yd}^{k+1} + I_{xz}^k I_{zd}^{k+1} \right) \\
& + \mu \mathrm{div}\left(\Psi'(|\nabla u^{k+1}| + |\nabla v^{k+1}| + |\nabla w^{k+1}|)\nabla u^{k+1} \right)
\end{aligned}
\tag{20}
$$

Yet we remain with the nonconvex and nonlinear terms, namely, those involving I with a subindex, and Ψ'. The nonlinearities concering I are removed via linearization with a first order Taylor expansion. To do so, the iterates are split into a known part and an unknown update $u^{k+1} = u^k + du^k$, $v^{k+1} = v^k + dv^k$ and $w^{k+1} = w^k + dw^k$. The Taylor expansion then reads

$$
\begin{aligned}
I_d^{k+1} &\approx I_d^k + I_x^k du^k + I_y^k dv^k + I_z^k dw^k \\
I_{xd}^{k+1} &\approx I_{xd}^k + I_{xx}^k du^k + I_{xy}^k dv^k + I_{xz}^k dw^k, \\
I_{yd}^{k+1} &\approx I_{yd}^k + I_{xy}^k du^k + I_{yy}^k dv^k + I_{yz}^k dw^k \\
I_{zd}^{k+1} &\approx I_{zd}^k + I_{xz}^k du^k + I_{yz}^k dv^k + I_{zz}^k dw^k
\end{aligned}
\tag{21}
$$

where all terms involving I are now evaluated for the current known displacement (u^k, v^k, w^k). The remaining nonlinearity is now removed with a similar step: all terms dependent on u, v or w that occur inside Ψ' are equipped with an additional index ℓ, the others with the index $\ell + 1$.

In addition, we slightly change the computation of gradients compared to the original contribution. Following Wedel et al,[23] we use a blended version of the derivative. That is, we compute all terms involving image gradients via

$$
\nabla I = (1 - \beta) \nabla I_1(x + \mathbf{u}) + \beta \nabla I_0(x).
\tag{22}
$$

The choice of β is problem dependent, we decided to fix it to a value of $\beta = 0.5$. The final inner iteration is now achieved by collecting all terms $du^{k,\ell}$, $dv^{k,\ell}$ and $dw^{k,\ell}$ on one side. This results in a 3-by-3-matrix equation for each $\mathbf{u}$, which we solve by successive overrelaxation (SOR).[24]

4 | NUMERICS

Brox et al[3] propose to use a coarse-to-fine strategy to improve the performance of the algorithm. Coarse-to-fine means to solve the optimization problem on a coarser scale, upscale this solution, and use the solution as a starting value for the optimization on the next finer scale. This improves the algorithm in two ways. First, being a more or less linear approximation, optical flow is only able to compute small displacements around one voxel. The computation on different scales can extend this range significantly. Second, due to the downscaling step, large scale details in images are matched first, as they appear on coarser scales. After refinement, small details are matched on finer levels.

Brox et al. precompute the image pyramids. A level of the pyramid is calculated by a Gaussian smoothing of the previous, finer level, and a downscaling of the smoothed image by a fixed factor. In the original work, a factor close to 1 is chosen, which causes a very smooth transition from one level to the next. This classical construction of image pyramids proceeds from fine to coarse as one starts with the original image and computes the smoothed, coarsened steps level by level. It is therefore necessary to store all intermediate steps.

Our data sets are too large to do so. We thus multiply the standard deviation of the original filter kernel for Gaussian smoothing by the number of remaining levels in each iteration. We smooth with these large filter kernels only when we need the images on that scale. Also, we use fewer levels than proposed in the original work as we only expect smaller displacements on similar scales.

For discretizing terms of the type $(\Psi'(|\nabla u|^2 + |\nabla v|^2 + |\nabla w|^2)\nabla u)$, the original work chose Brox diffusivity.[25] Denoting $\Psi_S = (\Psi'(|\nabla u|^2 + |\nabla v|^2 + |\nabla w|^2))$, we extend this to three dimensions at voxel position (ℓ, m, n) via

$$
\begin{aligned}
\operatorname{div} &\left(\Psi_S \nabla(u^k + du^k)\right) \\
&\approx \Psi_{\ell+1/2,m,n}^S \left(u_{\ell+1,m,n}^k + du_{\ell+1,m,n}^k - u_{\ell,m,n}^\ell - du_{\ell,m,n}^k\right) \\
&\quad - \Psi_{\ell-1/2,m,n}^S \left(u_{\ell,m,n}^k + du_{\ell,m,n}^k - u_{\ell-1,m,n}^k - du_{\ell-1,m,n}^k\right) \\
&\quad + \Psi_{\ell,m+1/2,n}^S \left(u_{\ell,m+1,n}^k + du_{\ell,m+1,n}^k - u_{\ell,m,n}^k - du_{\ell,m,n}^k\right) \\
&\quad - \Psi_{\ell,m-1/2,n}^S \left(u_{\ell,m,n}^k + du_{\ell,m,n}^k - u_{\ell,m-1,n}^k - du_{\ell,m-1,n}^k\right) \\
&\quad + \Psi_{\ell,m,n+1/2}^S \left(u_{\ell,m,n+1}^k + du_{\ell,m,n+1}^k - u_{\ell,m,n}^k - du_{\ell,m,n}^k\right) \\
&\quad - \Psi_{\ell,m,n-1/2}^S \left(u_{\ell,m,n}^k + du_{\ell,m,n}^k - u_{\ell,m,n-1}^k - du_{\ell,m,n-1}^k\right),
\end{aligned}
\tag{23}
$$

where exemplarily, we denote

$$\Psi^S_{\ell+1/2,m,n} = \left((u_{\ell+1,m,n} - u_{\ell,m,n})^2 \right.$$
$$+ \left(\frac{1}{2}\left(\frac{u_{\ell+1,m+1,n} - u_{\ell+1,m-1,n}}{2} + \frac{u_{\ell,m+1,n} - u_{\ell,m-1,n}}{2} \right) \right)^2$$
$$\left. + \left(\frac{1}{2}\left(\frac{u_{\ell+1,m,n+1} - u_{\ell+1,m,n-1}}{2} + \frac{u_{\ell,m,n+1} - u_{\ell,m,n-1}}{2} \right) \right)^2 \right)^{1/2}, \tag{24}$$

and note that the terms $\Psi^S_{\ell-1/2,m,n}$, $\Psi^S_{\ell,m+1/2,n}$, $\Psi^S_{\ell,m-1/2,n}$, $\Psi^S_{\ell,m,n+1/2}$ and $\Psi^S_{\ell,m,n-1/2}$ are computed similarly. For all other derivatives, we use central difference schemes.

After these discretizations, (20) does not contain any nonlinearities anymore. The optimality conditions can be put into a linear system and are solved for $du^{k,l+1}$, $dv^{k,l+1}$, $dw^{k,l+1}$ via successive overrelaxation.

Currently, we use our MATLAB implementation based on the 2D implementation of Katerina Fragkiadaki.[1] Our implementation can be found online[2] and runs under the AGPL license. Runtime and memory can surely be more efficiently balanced in a plain C++-implementation. We nevertheless decided for MATLAB as it provides—more or less platform-independent—efficient algorithms for all filtering and voxelwise operations on the data sets. MATLAB usually comes with a built-in method to compile C++-code as long as a compiler is available; therefore, only a short rebuild of one single file is necessary to run our code. Optimization, such as parallelization of the iterative schemes or a complete C++-implementation, will be part of future research.

The inner iterations of the successive overrelaxation are implemented in C++ and connected to the MATLAB foreground by the Mex Api. This code is not parallelized. To our knowledge, there are no 3D extensions of, for example, the Red-Black-SOR, that has been used to parallelize the 2D version of the algorithm.[26]

We conclude this section with a pseudo code description of our algorithm.

Algorithm 1 3D optical flow

Require: Images I_0, I_1, parameters L, σ, d, N, M, λ, μ

Ensure: Displacement u, v, w

  1: **for** $l = L, L-1, \dots 1$ **do**

  2:     Filter images with Gaussian filter with parameter σL and resize by factor d^l

  3:     **for** *outer* $j = 1, 2, \dots, N$ **do**

  4:         Compute image gradients according to (19)

  5:         **for** *inner* $i = 1, 2, \dots, M$ **do**

  6:             Calculate terms involving Ψ and solve via SOR to get $du^{j,i+1}, dv^{j,i+1}, dw^{j,i+1}$

  7:         **end for**

  8:         Update $u^{j+1} = u^j + du^M$, $v^{j+1} = v^j + dv^M$, $w^{j+1} = w^j + dw^M$

  9:     **end for**

10:     Upscale u^N, v^N, w^N

11: **end for**

---

5 | DATA SETS

Our main interest was to find an algorithm that calculates displacement fields in in situ testing of materials with behaviour on a microstructural scale. Therefore, our evaluation only involves data sets of this kind, in detail three foam structures and one long fibre reinforced thermoplastic (LFT).

---

[1]https://www.cs.cmu.edu/katef/LDOF.html.

[2]https://gitlab.itwm.fraunhofer.de/kuschnerus/3dof.

5.1 | Synthetically deformed foam

The first data set is given by synthetically deformed CT images of a ceramic foam. The original image is of size $630 \times 630 \times 230$ voxels at a voxel edge length of 70.88 μm. The chosen displacement field follows Losch et al[27] and is defined by

$$
\begin{aligned}
u(x,y,z) &= \begin{cases} z \cdot 0.005 \cdot K & \text{if } z < D/2, \\ (z - D/2) \cdot 0.005 \cdot K & \text{if } z > D/2 \end{cases} \\
v(x,y,z) &= 0 \\
w(x,y,z) &= -0.2 - \frac{K - 0.2}{\sqrt{0.5 \cdot e^{-0.04 \cdot (z - D/2)}}},
\end{aligned}
\tag{25}
$$

where K is the maximal displacement and chosen as $K = 10$, and $D = 230$ is the image size in z-direction. The displacement field exhibits two difficulties: the deformation in z-direction is nonlinear and mimics a compression. The deformation in x-direction is discontinuous mimicking struts breaking due to load. Both effects can be seen in Figure 2.

Due to the known ground truth displacement field, this data set will also serve as benchmark to evaluate the improvement achieved by the additional data term E_{Data}.

(a) Volume rendering

(b) xy-slice

(c) xy-slice compressed

(d) xz-slice with discontinuity
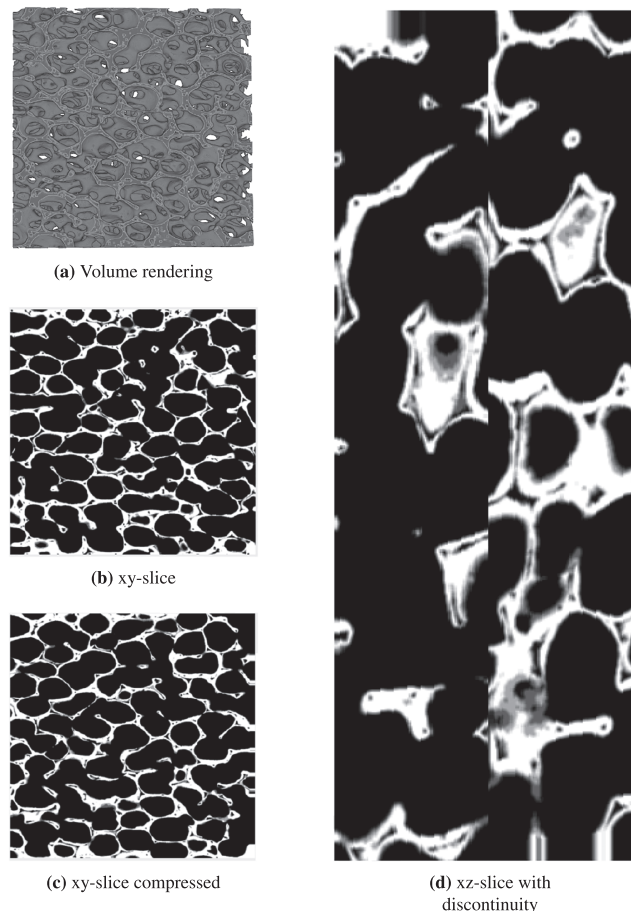
FIGURE 2 Volume and slice view of the synthetically compressed ceramic foam. Note the discontinuity in the middle layer in (d). Sample Foseco. Imaging Fraunhofer IZFP. Image size $630 \times 630 \times 230$ at voxel edge length 70.88 μm

5.2 | Metal matrix composite foam

For the second data set, we choose CT images of an MMC foam acquired during a compression test. The foam consists of highly alloyed stainless Cr-Mn-Ni TRIP-steel reinforced by magnesia partially stabilised zirconia (Mg-PSZ)-particles.[28] MMC foams are used as lightweight materials. Yet they need to be mechanically sufficiently stable. Their load-bearing capacity can be investigated in in situ tests. In the experiment described in Berek et al,[28] the foam was compressed by 2 %, 10 % and 16 %. The image size is $420 \times 420 \times 480$ voxels at a voxel edge length of 30 μm.

Figure 3a shows a slight corruption in the background caused by the high X-ray absorption of metals. We therefore pre-process the images: The whole image is binarized by Otsu's method, and the resulting binary image is used as a mask for the original image. Figure 3b shows the result. Note that these images are brightened to make the background corruption visible.

Metal foams feature rather large struts and do not fall in the category of foams whose failure is preceded by breaking struts. They are nevertheless a very suitable example to test the power of our algorithm. In fact, one can observe three stages of deformation during compression: linear elasticity, plastic collapse and densification. Hopefully, our algorithm is able to map all three kinds of motion. The stages of deformation can be seen in Figure 4.

5.3 | White glass foam

The third foam that we use to evaluate our method consists of recycled float glass. The CT images are acquired during an in situ compression test. In detail, we investigate the volumes at strain levels 1% and 3.8%. More information about the testing procedure can be found in Hubálková et al.[29] The image size is $450 \times 450 \times 400$ voxels at a voxel edge
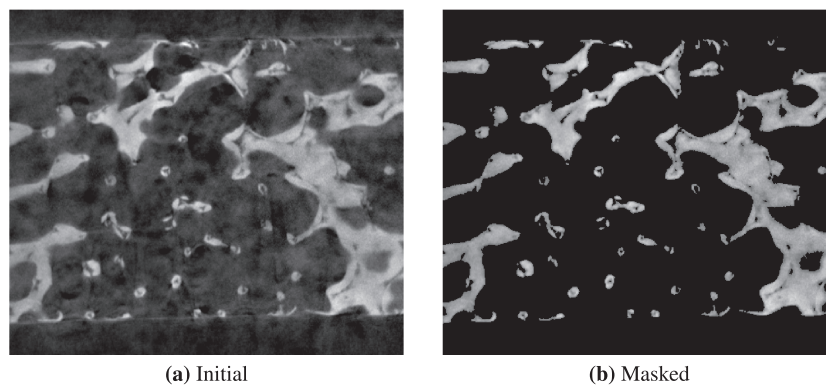

(a) Initial   **(b)** Masked

FIGURE 3 MMC foam: Result of the background noise removal

(a) Unloaded   **(b)** 10% compression, linear elastic and early plastic deformation   **(c)** 16% compression, plastic deformation and early densification
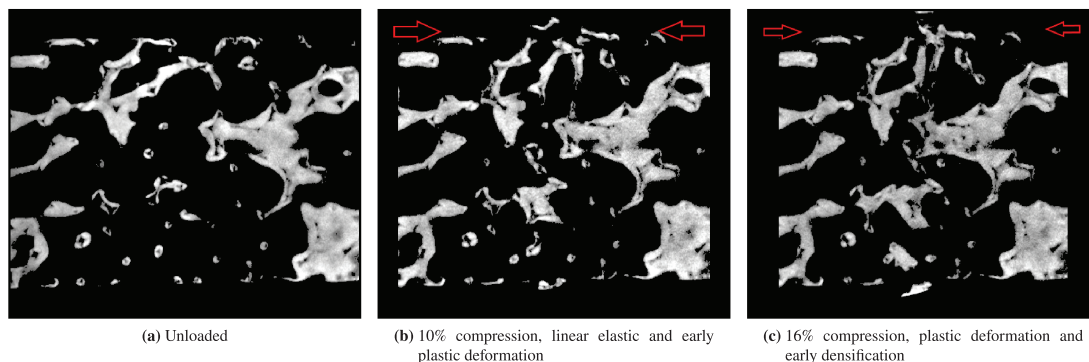
FIGURE 4 yz-slices showing the collision behaviour of the MMC foam. Red arrows indicate compression direction. Sample, in situ testing, and imaging: TU Bergakademie Freiberg. Image size $420 \times 420 \times 480$ at voxel edge length 30 μm

(a) Unloaded, xz-slice **(b)** 1% compression, xz-slice **(c)** 3, 8% compression, xz-slice

(d) Unloaded, yz-slice **(e)** 1% compression, yz-slice **(f)** 3, 8% compression, yz-slice
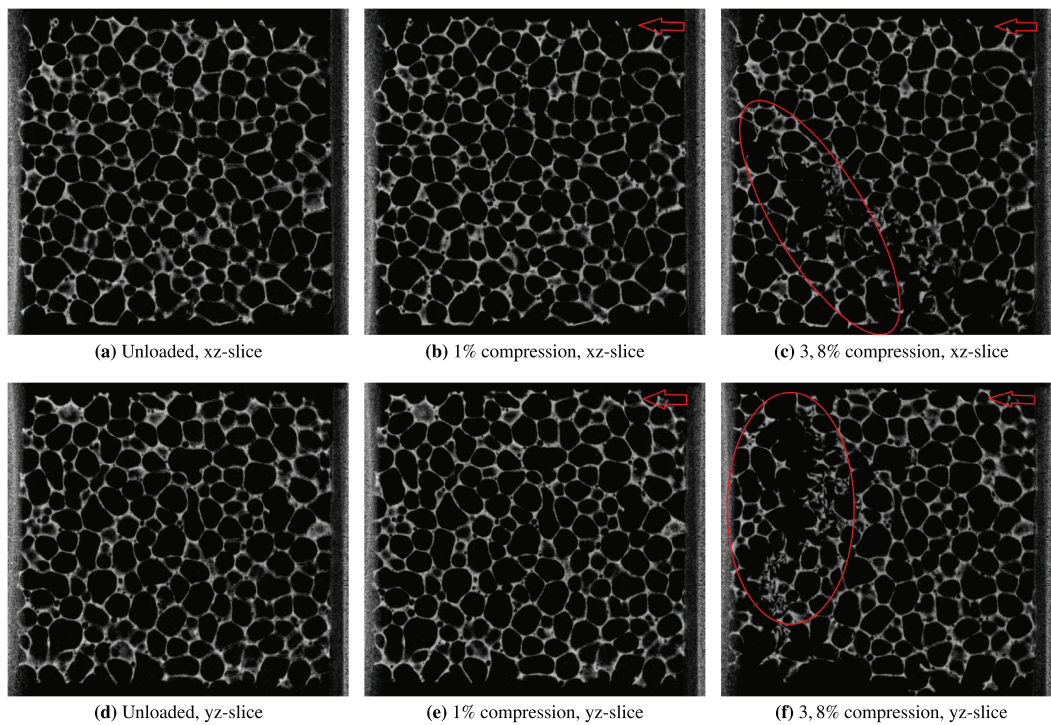
FIGURE 5 Slices of the compressed white glass foam. Observe that the first loading stage in panels (b) and (e) hardly influences the structural integrity. The second loading stage however introduces a diagonal crack visible in the lower left corner in panel (c) and a crack orthogonal to pressure direction visible in panel (f). The samples are compressed in z-direction, which is indicated by red arrows. Sample, in situ testing and imaging and sample by TU Bergakademie Freiberg

length of 40 μm. Figure 5 shows the same slice at the different loading steps. While images (a) to (c) show the evolution of a crack orthogonal to the compression direction, we can also observe diagonal cracks in images (d) to (f).

5.4 | Long fibre reinforced thermoplastic

As an example of a non-foam material, we consider an LFT sample. LFTs can be produced inexpensively and provide favourable properties with respect to stiffness and impact strength[30] and are therefore a popular replacement for metals in automotive industry. In order to fully exploit their potential it is crucial to characterize their material behaviour under load reliably. In situ testing is the method of choice to accomplish this task as the LFT's macroscopic behaviour is known to be driven by the microstructure, in particular the fibre component. In situ testing enables better understanding of the relationship of microstructural geometric and macroscopic physical properties by direct observation of microstructural changes during loading.

We use a μCT image of the central part of a tensile test specimen of size $1162 \times 769 \times 1481$ voxels at a voxel edge length of 5 μm. A volume rendering and a slice view illustrating the spatially sparse microstructure can be found in Figure 1.

6 | RESULTS AND DISCUSSION

We compare our method to several state-of-the-art algorithms in the area of motion estimation. Our method will be abbreviated with *3DOF*, which is short for **3D O**ptical **F**low, not to be confused with *Degrees of Freedom*. The parameters for our method for the different data sets can be found in Table 1. The first one to compare is the already mentioned LDDMM implementation CLAIRE[1][15] with parameters in Table 2. We chose this method as it is not only able

---

[1]https://github.com/andreasmang/claire.

TABLE 1 Parameter choice for our algorithms

| | 3DOF | | | | | | | |
Data set	λ	μ	M	N	W	L	d	σ
Synth. Def. Foam	1	20	15	15	3	10	0.9	$\frac{1}{\sqrt{2d}}$
MMC 10%	10	10	20	30	3	15	0.9	$\frac{1}{\sqrt{2d}}$
MMC 16%	10	10	20	30	3	15	0.9	$\frac{1}{\sqrt{2d}}$
Glass Foam 1%	20	5	20	30	3	15	0.9	$\frac{1}{\sqrt{2d}}$
Glass Foam 3.8%	20	5	20	30	3	15	0.9	$\frac{1}{\sqrt{2d}}$
LFT	50	50	10	5	3	15	0.98	$\frac{1}{\sqrt{2d}}$

Note: λ and μ are regularization parameters; M is the number of outer iterations; N is the number of inner iterations; W is the number of warps; L number of levels; d is the downscaling factor; and σ is the smoothing factor.

TABLE 2 Parameter choice for CLAIRE

| | CLAIRE | |
Data set	N	n_t
Synth. Def. Foam	40	16
MMC 10%	20	8
MMC 16%	20	8
Glass Foam 1%	20	8
Glass Foam 3.8%	20	8
LFT	20	8

Note: N is the maximal number of iterations; n_t is the time stepping.

TABLE 3 Parameter choice for DVC-based methods

| | Local DVC | Global DVC | | ALDVC | |
Data set	S_V	S_{FEM}	α	S_{FEM}	S_V
Synth. Def. Foam	40	10	2	10	40
MMC 10%	40	20	2	10	40
MMC 16%	40	10	2	10	40
Glass Foam 1%	40	10	2	10	40
Glass Foam 3.8%	40	10	2	10	40
LFT	40	120	2	40	120

Note: S_V is the subvolume size for IC-GN based local DVC; S_{FEM} is the subset size for global FEM based DVC; α is the regularization parameter. Note that for ALDVC, the used parameters are the recommended minimal choice of parameters by the original authors.

to map full field displacements in the same manner as we do but is also able to compute large deformation. We use two other classical DVC methods: one is a local IC-GN based method[2], the other one a global finite element method[3], namely, the above mentioned hybrid ALDVC2 method. The parameters for all DVC-based methods can be found in Table 3. Note that for methods two and three, we use the same implementation as ALDVC. Therefore, a more detailed description of the implementation can be found in Yang at al.[11] For all displacement fields, we computed the root mean squared error (RMSE)

---

[2]https://github.com/FranckLab/ALDVC.

[3]https://github.com/jyang526843/FE_Global_DVC.

$$\left(\frac{1}{N} \sum_x \left(I_{\text{warped}}(x) - I_0(x) \right)^2 \right)^{\frac{1}{2}}, \tag{26}$$

where I_{warped} is I_1 deformed by the computed displacement field $\mathbf{u}$, and N is the number of voxels. The results are summarized in Table 4. To compute the warped image, an interpolation step is required—we use MATLABs inhouse trilinear interpolation. To upscale the DVC such that it is possible to compute a warped image, we also used trilinear interpolation. The data sets used here are smaller than usual in materials science, see Section 1 for details. This restriction is required to allow for the comparison with CLAIRE. Note, however, that the 3DOF approach has proven to successfully compute displacement fields in data sets of size 1800^3 voxels already.

6.1 | Synthetically deformed foam

The RMSE values for this data set already demonstrate the superior performance of our algorithm. Figure 6 shows a slice view of the residuals of the data sets at a critical position. Here, residual refers to the voxelwise absolute grey value difference of the original image and the warped image. The view exhibits the main difficulty of the data set, namely, the discontinuity. All algorithms struggle to compute a perfect residual here, 3DOF showing the lowest error. The computation time for this data set was about 10 h.

TABLE 4 RMSE (root mean squared error/difference) of warped and original data sets for our application cases

Data set	3DOF	CLAIRE	IC-GN	FEM	ALDVC
Synth. Def. Foam	**0.0876**	0.1732	0.1265	0.1343	0.1069
MMC 10%	**0.0110**	0.1245	0.0225	0.0202	0.0224
MMC 16%	**0.0101**	0.1467	0.0173	0.0163	0.0172
Glass Foam 1%	**0.0124**	0.0627	0.0164	0.0147	0.0146
Glass Foam 3.8%	**0.0182**	0.0767	0.0237	0.0234	0.0224
LFT	**0.0073**	0.2930	0.0277	0.0283	0.0277

Note: Best performing methods highlighted in bold.

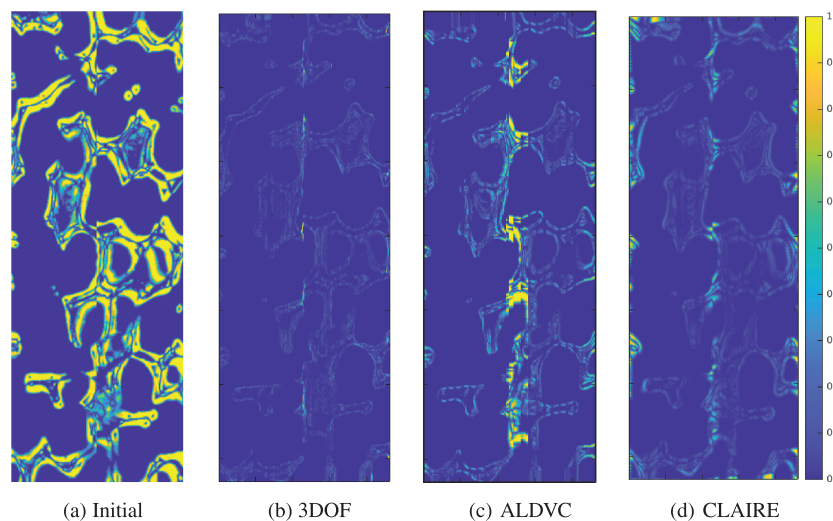(a) Initial (b) 3DOF (c) ALDVC (d) CLAIRE

FIGURE 6 xz-slices of residuals of the synthetically deformed ceramic foam. Yellow colour in the residual indicates the largest possible error, and dark blue indicates a perfect match between both data sets. The scale for all the data sets can be found on the very right of the figure

TABLE 5   Angular error measures of all methods for the synthetically deformed foam

Measure	3DOF	CLAIRE	IC-GN	FEM	ALDVC
AAE	**2.29°**	15.1°	2.34°	2.305°	2.31°
AEE	**1.60**	24.14	2.57	1.99	2.05

Yet we want to put emphasis on another evaluation method. As we synthetically deformed the foam, we have a known displacement field, which we can compare to the computed version. This can be done via the average angular error (AAE)

$$\frac{1}{N}\sum_{\mathbf{x}}\arccos\left(\frac{1+\mathbf{u}\cdot\mathbf{u}_{GT}}{\sqrt{1+u^2+v^2+w^2}\sqrt{1+u_{GT}^2+v_{GT}^2+w_{GT}^2}}\right) \tag{27}$$

and the average endpoint error (AEE)

$$\frac{1}{N}\sum_{\mathbf{x}}\left(\sqrt{(u-u_{GT})^2+(v-v_{GT})^2+(w-w_{GT})^2}\right). \tag{28}$$

where u_{GT}, v_{GT}, w_{GT} are the groundtruth components of the displacement vector field. Table 5 contains AAE and AEE, for all methods. Again, our method 3DOF clearly yields the lowest values. AAE and AEE are well-known measures in 2D Optical Flow,[31] and values in the range achieved here by 3DOF are considered to prove good performance already in two dimensions. The outstanding performance can also be observed visually. Figure 7 shows slices of the displacement field in all coordinate directions of our method, ALDVC as best performing DVC-based method and CLAIRE. Especially the discontinuity in u is computed best by our algorithm. Note that for the displacement fields in u and v direction, the colour axis for CLAIRE had to be modified due to large outliers.

At points of discontinuity, most methods including ours still struggle. This can be seen in the difference images of Figure 8. In Figure 8a, the initial difference image is shown. Here, the magenta colour corresponds to the solid component that is only apparent in the original, undeformed foam, whereas the green colour corresponds to the solid component of the synthetically deformed foam. White marks pixels where both components overlay. A favourable result would therefore be a difference image that consists only of white colour. Our algorithm nearly achieves this, see Figure 8b. Note that in this case the magenta colour displays the initial foam deformed by the computed displacement field. The same holds for Figure 8c,d, which show the difference image resulting from CLAIRE. Compared to this performance, our algorithm performs much better; however, the zoom shows the inability of the algorithm to eventually *break* the strut.

We also used this synthetic example for a small study on how the parameter λ influences the quality of the estimation of the displacement vector field. In Figure 9, we can see that a careful choice of λ results in a minimum in the RSME and, most important, is a drastic improvement over setting $\lambda = 0$, so not including the difference in gradients at all. Note that nonetheless the choice of the parameters μ and λ is highly problem dependent. To our knowledge, there are no rules for calculating the optimal parameters, they can only be found by testing on, for example, downscaled versions of the images to reduce computation time.

6.2 | MMC foam

Table 4 again shows that we outperform the state-of-the-art methods with respect to the RMSE. The computation time for each image pair was about 9 h. Note that from now on, we will only focus on the analysis of our results and renounce further comparisons to CLAIRE and local and FE-based DVC, but show comparisons to ALDVC, as it performed best among the comparison methods.

Our interest also lies in the displacement field. As the data set was generated during a compression test, the computed displacement should reflect the behaviour during the test. Figure 10 shows that our algorithm again describes the
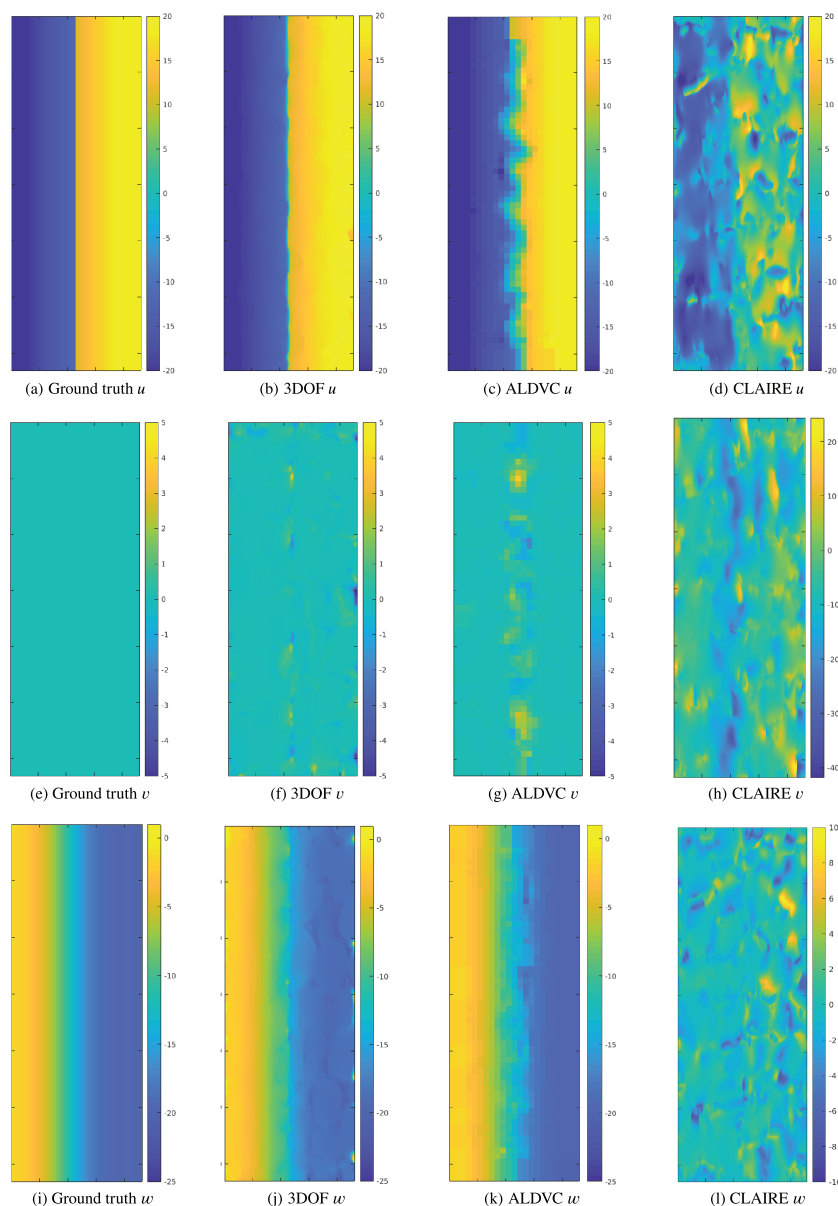
FIGURE 7 Slices of the ground truth and computed displacement fields for the synthetically deformed ceramic foam. Yellow colour indicates movement along the horizontal axis, and blue refers to the opposite direction

increasing compression best. We can see a very detailed edge where material densifies in the middle. ALDVC resolves this behaviour rather roughly. The displacement field between 10% and 16% presents a plausible discontinuity along the failure. Note that we only focus on w, the displacement field in z-direction, as it is along compression direction and therefore the most interesting one.

However, the difference images in Figure 11, composed in the same way as in Section 6.1, again show the difficulty of brittle and plastic behaviour. We can clearly see that the result on those parts yet needs to be improved in future work.

6.3 | White glass foam

This third data set is the main representative of data sets of special interest for us. Its spatial content is very sparse, as it consists of very thin struts, and is fractured rather immediate between 1% and 3.8% of loading. Figure 12 shows the displacement of the glass foam from 1% to 3.8%. We did not include the displacement field of the first loading stage here,

(a) No warping   **(b)** 3DOF with zoom   **(c)** ALDVC   **(d)** CLAIRE
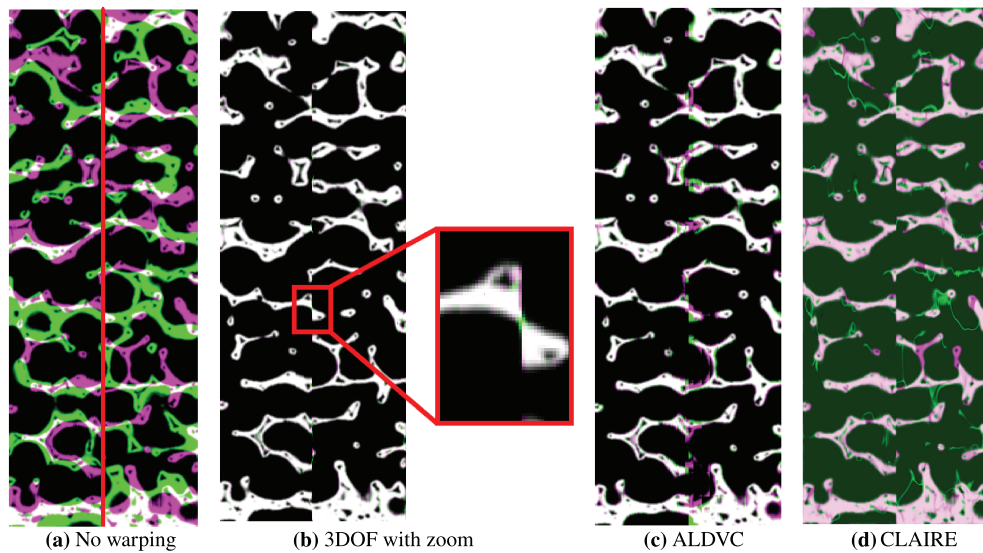
FIGURE 8   Difference images of the synthetically deformed foam. White: pixels in the solid component in original and deformed image. Magenta: pixels only in the solid component of the initial foam in panel (a) and deformed by the computed displacement fields in panels (b)–(d). Green: pixels only in the solid component of the synthetically deformed foam. In panel (a), we also marked the discontinuity by a red line. To not distort the results, we forgo this marker in the computed images. The zoom in panel (b) shows that the displacement field does not fully capture discontinuous behaviour. 3DOF did not break the strut
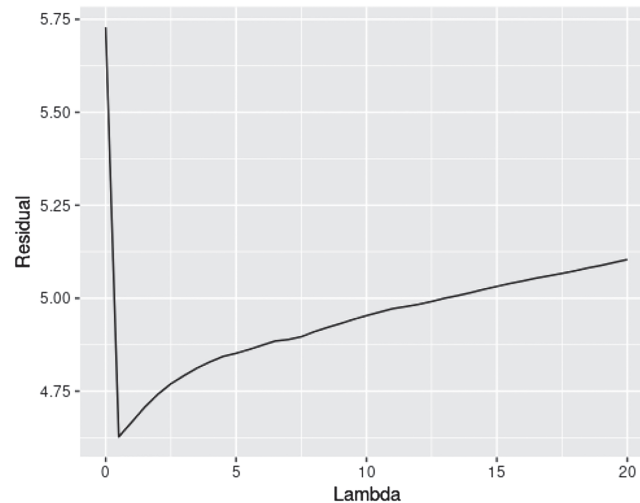
FIGURE 9   Influence of the parameter λ. The residual plot shows the improvement over $\lambda = 0$ and also that an optimal value can be found at $\lambda = 1$

as it does not contain any large damage behaviour. The qualitative result of the displacement field perfectly maps the behaviour described in Hubálková et al.[29] This example also shows remarkably the difference between DVC-based methods like ALDVC and our approach: we compute a true voxel-based displacement field that is not gained by interpolation as in DVC. The computation time for each image pair was about 10 h. The accurate outcomes of this approach are apparent over all directions, but are particularly prominent in Figure 12c,f,i,l. These images show the fault zones in loading direction and are therefore of special interest. Here, we see that our approach calculates a strut-wise displacement, where we gain the impression that ALDVC computes only an average displacement (per subvolume). Nevertheless, the difference images in Figure 13 again shows the weakness of our method. Especially in this case, where the motion between 1% and 3.8% is rather large, the areas suffering from brittle fracture cannot be matched sufficiently, but still better than in ALDVC. And again, the improvement in these regions is postponed to future work.
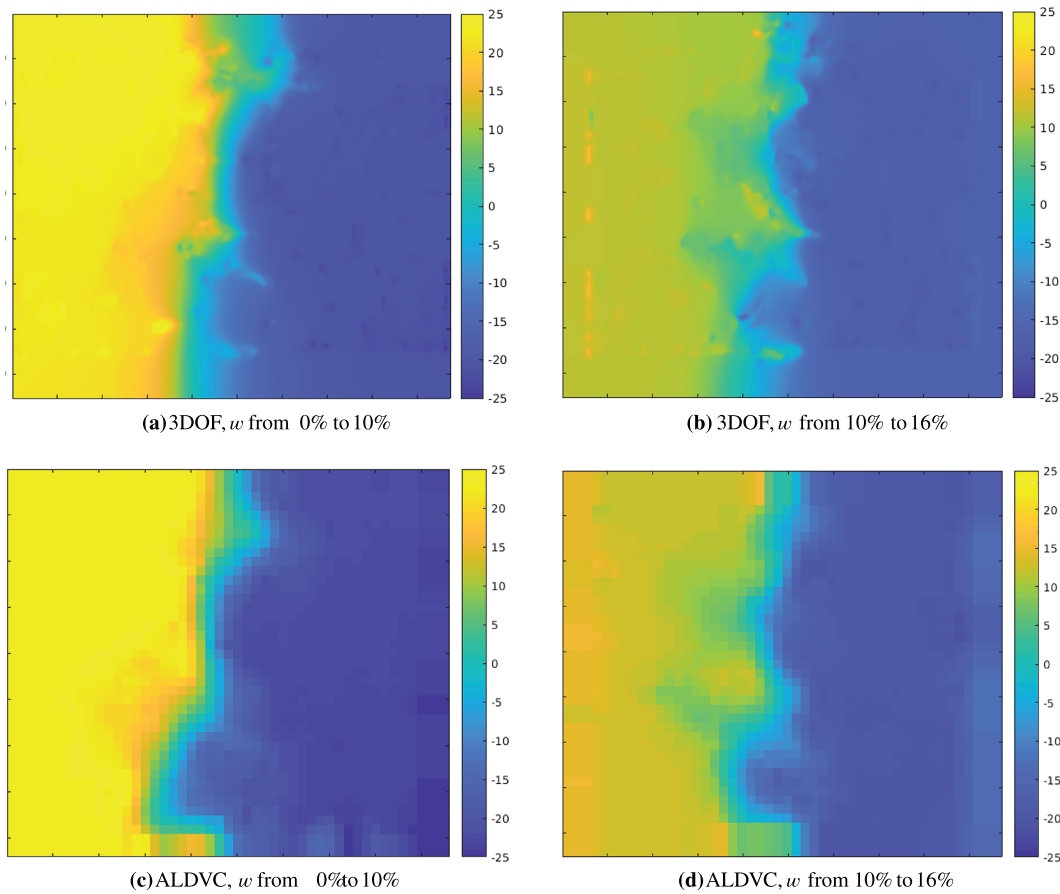
FIGURE 10 xz-slices of displacement field of the MMC foam. Again, yellow colour indicates movement along the horizontal axis, and blue refers to the opposite direction. The computed displacement field reflects the expected behaviour during the experiment
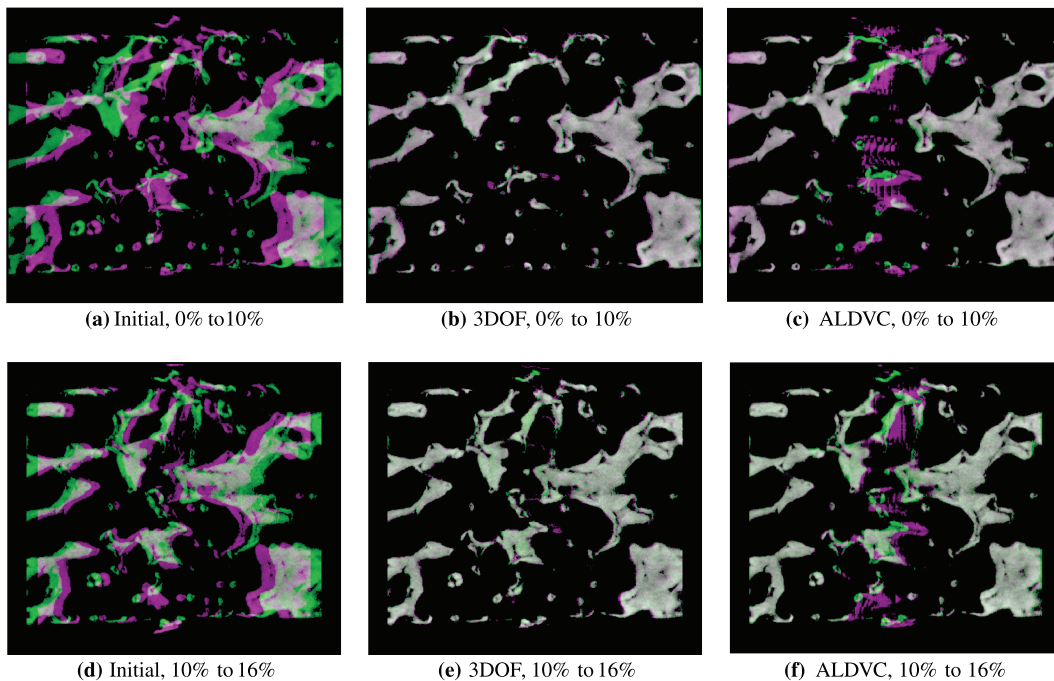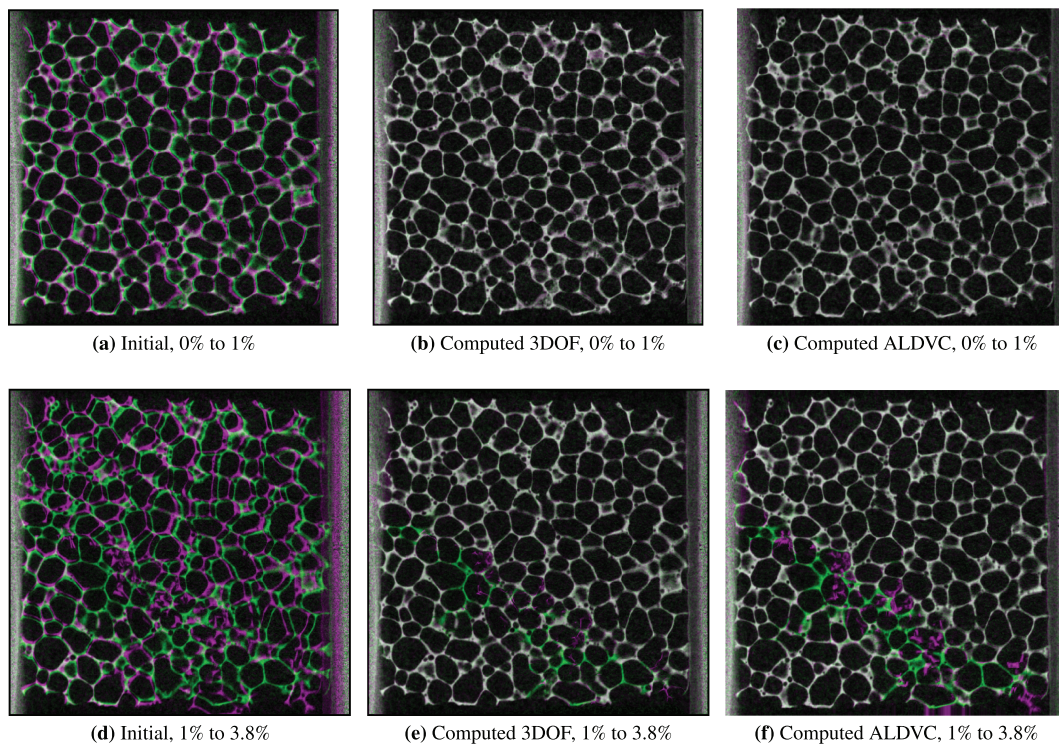

FIGURE 11 xz-slices of the difference image of MMC foam. White: pixels in the solid component in original and deformed image. Green: pixels only in the solid component of the initial foam in panels (a) and (d) and deformed by the computed displacement fields in panels (a), (c), (e) and (f). Magenta: pixels only in the solid component of the deformed foam. Especially brittly fractured parts are difficult to match, as we can see in magenta in panels (b), (c), (e) and (f)
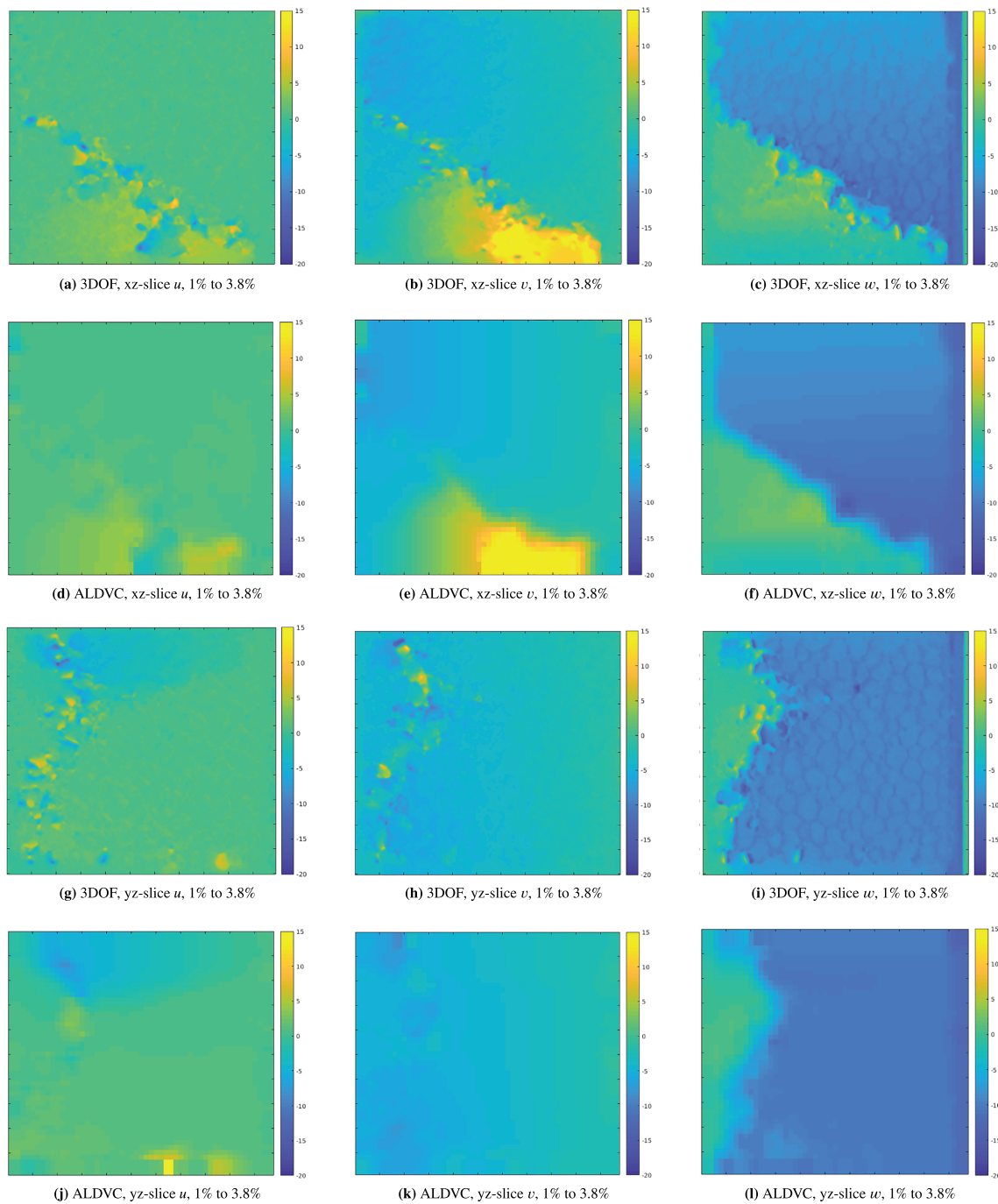
FIGURE 12 Slices of the displacement field of the Glass Foam. The discontinuity along the previously observed fault zones can be well observed in the computed displacement fields in *z*-direction in (c) and (i). Both the averaging nature of the DVC-based displacement field and the fine details of 3DOF can be observed very well

6.4 | Long fibre reinforced thermoplastic

Figure 14 shows the residuals for the LFT data set. The algorithm managed to match the individual fibres properly, and to significantly reduce the overall RMSE. Note that the initial RMSE is already much lower than for all data sets we discussed earlier. This can also be seen in Figure 14a, where the maximal absolute difference value can be found at ≈0.2, where for other data sets it was close to 1, see, for example, Figure 6. This means that the expected maximal displacement is also much lower than for the foam samples we already investigated. Nevertheless, the residual of our method in Figure 14b shows a significant improvement and a lower residual than ALDVC in Figure 14c. Therefore, our

FIGURE 13 xz-slices of the difference image of the glass foam. White: pixels in the solid component in original and deformed image. Green: pixels only in the solid component of the initial foam in panels (a) and (d) and deformed by the computed displacement fields in panels (b), (c), (e), and (f). Magenta: pixels only in the solid component of the deformed foam. Observe the green component in the lower left corner of panels (e), and (f). Although the displacement field displays plausible behaviour, the fault zone is not mapped correctly
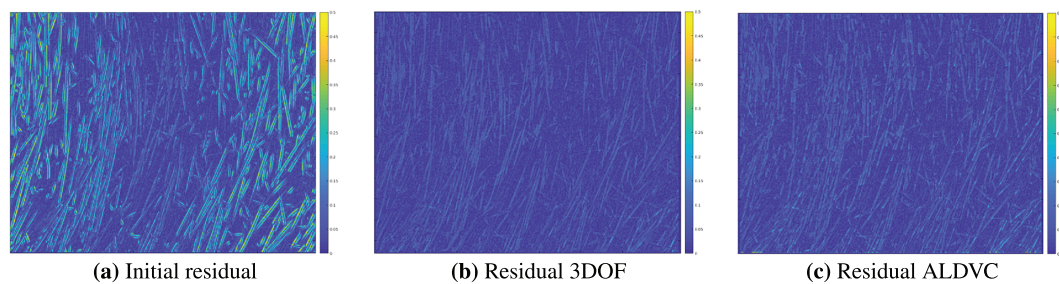
FIGURE 14 LFT: slice view of the residuals

proposed method does not only compute large displacement of foams in a very exact way, but also smaller displacement of fibres. The computation time for this data set took about 30 h.

7 | CONCLUSION

Estimating motion in CT data of microstructural materials poses very specific challenges. We proposed to extend a well-known, two dimensional method to 3D and showed that most of our requirements are fulfilled. The method estimates a displacement field for every voxel, based on intensities and intensity gradients and is regularized by the well-known total variation regularization. We provided a detailed description on how to tackle the problem numerically, including a description on how to discretize the three-dimensional differential operators. Our algorithm was tested on manifold examples, including an artificially deformed foam for exact benchmarking. The deformation of this foam may not be mechanically accurate, but it provided a heavy discontinuity which our algorithm resolves best. Further

examples included compression of foams, and tension of a long fibre reinforced thermoplastic. We showed that DVC-based algorithms struggle to accurately compute displacement in spatially very sparse areas, whereas our approach produces very accurate results. Especially for glass foams, which are a textbook example of this category, our algorithm displays highly local behaviour and can therefore help to identify early stage failure of single struts. Classical and modern DVC approaches only gave a rough idea on how and where the fracture arises. Therefore our method is much more suitable to map early stage behaviour of material failure.

Our algorithm does not only generate satisfactory results in regions that exhibit discontinuous behaviour, it can also be applied to large data sets. The LFT data set served as example, and again, we were able to map the deformation of the sparse content very precisely. Nevertheless, our algorithm has issues to address. Brittle and plastic motion is still not matched to full mechanic accuracy, as the difference images showed. Though the displacement field is plausible, the residual showed that the computed deformation is not capable of completely *breaking* the material. This topic will be tackled in future research which will also address the efficiency and parallelization of our solution algorithms. In total, we can conclude that the requirements posed in Section 1 were met more successfully by 3DOF than by existing methods: Our method computes local displacement fields for spatially thin contents, even when the field of views do not coincide.

DATA AVAILABILITY STATEMENT

Parts of the data that support the findings of this study, namely, the synthethically deformed foam, are openly available at https://gitlab.itwm.fraunhofer.de/kuschnerus/3dof. The remainder of the data is not available to the public.

ORCID

Tessa Nogatz https://orcid.org/0000-0001-9466-688X

REFERENCES

[1] K. Rohr, H. S. Stiehl, R. Sprengel, T. M. Buzug, J. Weese, M. H. Kuhn, *IEEE T-MI* **2001**, *20*(6), 526.

[2] T. Polzin, J. Rühaak, R. Werner, J. Strehlow, S. Heldmann, H. Handels, J. Modersitzki, *in 5th pia 2013*, (Eds: M. Brown, M. de Bruijne, K. Ding, B. van Ginneken, A. Kiraly, J.-M. Kuhnigk, J. McClelland, K. Mori, J. Reinhardt), CreateSpace, Nagoya, Japan **2013**, 85–96.

[3] T. Brox, A. Bruhn, N. Papenberg, J. Weickert, *in Computer vision—ECCV 2004*, (Eds: T. Pajdla, J. Matas), Springer Berlin Heidelberg, Prague, Czech Republic **2004**, 25–36.

[4] B. K. Bay, T. S. Smith, D. P. Fyhrie, M. Saad, *Exp. Mech.* **1999**, *39*(3), 217.

[5] S. A. Hall, *ALERT Geomaterials Doctoral Summer Sch.* **2012**, *2012*, 69.

[6] S. Baker, I. Matthews, *Int. J. Comput. Vis.* **2004**, *56*(1), 221.

[7] B. Pan, B. Wang, D. Wu, G. Lubineau, *Opt. Lasers Eng.* **2014**, *58*, 126. https://www.sciencedirect.com/science/article/pii/S0143816614000529

[8] B. Wang, B. Pan, *Meas. Sci. Technol.* **2017**, *28*(10), 105007.

[9] B. Pan, B. Wang, *Meas. Sci. Technol.* **2017**, *28*(10), 105007. https://doi.org/10.1088/1361-6501/aa7e8a

[10] H. Leclerc, J.-N. Périé, S. Roux, F. Hild, *Exp. Mech.* **2011**, *51*(4), 479.

[11] J. Yang, L. Hazlett, A. K. Landauer, C. Franck, *Exp. Mech.* **2020**, *60*, 1205.

[12] H. Leclerc, J.-N. Périé, F. Hild, S. Roux, *Mech. Ind.* **2012**, *13*(6), 361.

[13] J. Modersitzki, *Numerical methods for image registration*, Oxford University Press on Demand **2004**.

[14] T. Polzin. **2018**, Large deformation diffeomorphic metric mappings: Theory, numerics, and applications, Ph.D. Thesis.

[15] A. Mang, A. Gholami, C. Davatzikos, G. Biros, *SIAM J. Sci. Comput.* **2019**, *41*(5), C548.

[16] A. Mang, G. Biros, *SIAM J. Imaging Sci.* **2016**, *9*(3), 1154.

[17] B. K. P. Horn, B. G. Schunck, *rtif. Intell.* **1981**, *17*(1), 185.

[18] B. D. Lucas, T. Kanade, *in Proceedings of the 7th international joint conference on artificial intelligence - volume 2*, Morgan Kaufmann Publishers Inc. **1981**, 674–679.

[19] J. Cheng, Y.-H. Tsai, S. Wang, M.-H. Yang, *in 2017 iccv*, (Eds: L. O'Conner), IEE CPS **2017**, 686–695.

[20] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, T. Brox, *in Proceedings of the ieee cvpr 2017*, (Eds: L. O'Conner), IEE CPS **2017**, 2462–2470.

[21] S. Hermann, R. Werner,In R. Klette, M. Rivera, S. Satoh (eds.), *in Psivt 2013*, Springer **2013**, 23–35.

[22] D. Hafner, O. Demetz, J. Weickert, *in Scale space and variational methods in computer vision*, (Eds: A. Kuijper, K. Bredies, T. Pock, H. Bischof), Springer Berlin Heidelberg **2013**, 210–221.

[23] A. Wedel, T. Pock, C. Zach, H. Bischof, D. Cremers, *in Statistical and geometrical approaches to visual motion analysis*, (Eds: D. Cremers, B. Rosenhahn, A. L. Yuille, F. R. Schmidt), Springer **2009**, 23–45.

[24] M. Hanke-Bourgeois, *Grundlagen der numerischen mathematik und des wissenschaftlichen rechnens*, Springer **2009**.

[25] T. Brox. **2005**, Von Pixeln zu Regionen: Partielle Differentialgleichungen in der Bildanalyse, Ph.D. Thesis.

[26] P. Gwosdek, A. Bruhn, J. Weickert, *in 13th international fall workshop on vision, modeling, and visualization, VMV 2008, konstanz, germany, october 8-10, 2008*, (Eds: O. Deussen, D. A. Keim, D. Saupe), Aka GmbH **2008**, 253–262.

[27] K. Losch, K. Schladitz, U. Ballaschk, H. Berek, C. G. Aneziris, *Image Anal. Stereology* **2014**, *33*(2), 131.

[28] H. Berek, U. Ballaschk, C. G. Aneziris, K. Losch, K. Schladitz, *Mater. Charact.* **2015**, *107*, 139.

[29] J. Hubálková, C. Voigt, A. Schmidt, K. Moritz, C. G. Aneziris, *Adv. Eng. Mater.* **2017**, *19*(9), 1700286.

[30] H. Ning, N. Lu, A. A. Hassen, K. Chawla, M. Selim, S. Pillay, *Int. Mater. Rev.* **2020**, *65*(3), 164.

[31] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, R. Szeliski, *Int. J. Comput. Vision* **2011**, *92*(1), 1.