

Towards Data-Informed Motion Artifact Reduction in Quantitative CT Using Piecewise Linear Interpolation

Emeric Boigné^{ID}, Dilworth Y. Parkinson^{ID}, and Matthias Ihme^{ID}

Abstract—We propose to use piecewise linear interpolation (PLI) in time to reduce motion artifacts in transmission computed tomography (CT). PLI is motivated by the natural occurrence of piecewise-linear evolution of voxel values during object motion. The method is specifically examined in the context of high-accuracy quantitative measurements that are compromised by small motions, and particularly sub-pixel motion. Compared to existing methods, the proposed approach offers three advantages: (i) the flexibility in the interpolation parameters provides a framework for joint optimization and data-informed dynamic CT, (ii) both continuous motion and sudden changes in voxel values can be represented while preserving the continuity of the interpolated solution, and (iii) the compactness of the interpolation functions reduces the increase in algorithmic cost. Total variation regularization is used with a second-order accurate discretization, and the resulting formulation is solved with the Chambolle-Pock proximal algorithm. The applicability of the method in practical cases is demonstrated using synchrotron data, with an algorithmic cost of two to four times that of equivalent static reconstruction algorithms.

Index Terms—4D computed tomography, Advanced Light Source, dynamic tomography, motion artifact, quantitative tomography, time interpolation, total variation.

I. INTRODUCTION

MOTION artifacts represent a major source of errors in quantitative computed tomography (CT) applications [1]. Standard tomographic reconstruction methods, such as the Filtered-Back Projection (FBP) algorithm or the Simultaneous Iterative Reconstruction Technique (SIRT), assume that the object remains immobile during acquisition [2]. However, motion of the sample arises in numerous tomography applications, thereby resulting in non-physical artifacts after reconstruction. Sample motion is particularly detrimental for in situ measurements, where the sample dynamics are of primary interest [3], [4], [5], [6], [7] or in medical imaging, in which patient and organ motion can compromise medical diagnostics [8], [9], [10].

Manuscript received 22 April 2022; revised 14 September 2022; accepted 12 October 2022. Date of publication 17 October 2022; date of current version 27 October 2022. This work was supported in part by the Leading Edge Aeronautics Research for NASA (LEARN) under Grant NNX15AE42A, in part by the NSF under Grant CBET-1800906, and in part by the DOE under Grant DE-SC0022222. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Se Young Chun. (Corresponding author: Emeric Boigné.)

Emeric Boigné and Matthias Ihme are with the Department of Mechanical Engineering, Stanford University, Stanford, CA 94305 USA (e-mail: eboigne@stanford.edu; mihme@stanford.edu).

Dilworth Y. Parkinson is with the Lawrence Berkeley National Laboratory, Berkeley, CA 94720 USA (e-mail: dyparkinson@lbl.gov).

Digital Object Identifier 10.1109/TCI.2022.3215096

The most direct approach to suppress motion artifacts is to increase the acquisition rate, and certain systems have been developed to achieve up to 1 kHz CT acquisition rates [11], [12], [13], [14]. For instance, the fast rotation stage at the TOMCAT beamline enabled the in situ investigation of a 3D liquid-foam flow around a sphere [15], and a burning match stick [16]. However, these fast-acquisition systems are not widely available, and the reduction in acquisition time comes at the expense of lower photon count, which may be detrimental for certain applications. Motion artifacts can also be reduced by modifying the angular acquisition scheme [17], [18], or by acquiring phase-locked or ensemble-averaged measurements [6], [19].

In contrast to modifying the acquisition, reconstruction algorithms have been developed to address motion artifacts. These methods are here classified in three categories. The first category relies on a priori knowledge of the object motion. For instance, rigid body models with up to six degrees of freedom can estimate alignment errors due to imperfect stage rotation in nanotomography [20], or represent respiratory motion in emission CT [21] and head motion in clinical CT [22], [23] using data from optical tracking systems. Motion models can also be applied during backprojection [16] or directly in the sinogram space [24], [25]. Image registration models with a large number of kinematic parameters can handle more complex motion. In particular, deformation vector fields (DVF) with B-spline interpolation has successfully reduced artifacts when scanning open-cell foams under compression [26], [27] or differentiating between motion changes and true anatomical changes, such as tumor growth [28], [29]. Data-driven models can also describe sample motion, either using reference scans [30] or convolutional neural networks [31], [32].

Even though these methods can successfully attenuate motion artifacts, they enforce local conservation of Beer-Lambert's attenuation, which may not be valid in reacting systems or multiphase applications in which mass is transported across the entire sample [7], [33]. In addition, these methods only represent invertible motions [34], which is not detrimental in most cases as inversion errors remain below the spatial resolution. However, in quantitative applications where sub-pixel motion is sufficient to compromise high-accuracy requirements, the interpolation errors due to the DVF inversion can become significant. Without prior assumption of any motion, the second category of methods instead employs regularization to penalize artifacts. Specifically, data sparsity in the solution is promoted via compressed sensing, either by allowing the solution to only take a set of fixed discrete values, or by minimizing spatial gradients. For instance, total

variation (TV) regularization promotes smoothness in the solution while preserving sharp edges, which improves the image quality, especially in cases with low photon counts and fewer angles [35], [36]. In the context of motion artifacts, TV regularization can penalize non-smooth streak artifacts, and promote similarity to artifact-free prior images [37]. Besides, by adding a gradient penalty with respect to time in the TV regularization, artifacts can be further reduced [9], [38], [39]. In these methods, the regularization term is minimized in conjunction with a data-fidelity term. However, static CT projectors cannot model motion in the sinogram, and the data-fidelity term is lower for a solution with artifacts than one without. As a result, while smooth artifact-free solutions minimize the regularization term, they do so at the cost of an increase in the data-fidelity term, and this competition overall limits the artifact reduction.

Methods in the third category address this limitation by using dynamic CT projectors that better reconstruct the sinograms of moving objects. Specifically, by modeling voxel values as a linear combination of a set of M interpolation functions, smooth artifact-free solutions can minimize both the regularization and the data-fidelity terms, thereby improving the artifact reduction. However, since not all possible time-varying solutions are spanned, the truncated decomposition will still not exactly represent the ideal solution field. In addition, because the time-decomposition increases the number of variables by a factor of M , the dimensionality of the undersampled ill-posed inverse problem is significantly increased, and the convergence towards a satisfactory solution can become challenging. Therefore, ongoing research on time-decomposition methods investigates whether truncated decompositions can result in artifact-free solutions, and whether algorithms can converge towards these solutions.

So far, piecewise constant interpolation (PCI) in time is the most examined time-decomposition choice in the CT literature. Early PCI studies primarily focused on developing local and non-local spatial-temporal regularization and only briefly analyzed the time-decomposition using fixed equidistant breakpoints [9], [38], [39]. One study then performed an extensive comparison between previous methods and a new region-based SIRT algorithm with PCI in time [40]. A follow-up article proposed a version of this algorithm that automatizes some of the underlying segmentation tasks, thereby reducing the post-processing time and algorithmic cost [41]. Another study developed a more general PCI formulation that leveraged interlaced view sampling and minimized non-convex cost functions that model Poisson photon-counting statistics, zingers, and the non-idealities that cause ring artifacts [42].

These studies have shown that PCI can model jumps in attenuation due to the sudden motion of a material phase inside or outside of a voxel [40], [41]. However, PCI fails to represent more continuous voxel changes. For instance, PCI does not accurately resolve uniform sub-pixel motion, which limits its performance in high-accuracy quantitative CT. Yet, only few CT studies have examined alternative interpolations. One study briefly considered second-order polynomials [43], while a more recent work used a Fourier-based interpolation [44].

Studies in dynamic Positron Emission Tomography (PET), with noisier data than CT, have explored additional interpolation choices [45]. For instance, the characteristic response of PET tracers was parametrized in model-based interpolation functions [46], whereas several studies used B-splines interpolation [47], [48], [49]. Furthermore, some PET studies have considered optimizing the interpolation scheme from the data itself. In particular, one study used singular value decomposition to construct a set of time-interpolation functions derived to model 4D PET data [50]. Two studies have also examined optimizing the choice of interpolation jointly with the reconstruction field during run-time, either iteratively [51] or simultaneously [52].

These PET results suggest that data-informed interpolations could work for dynamic CT. Besides, augmenting kinetic motion models with optical tracking data has already improved results in clinical CT [22], [23], while deep-learning has proved successful at directly correcting motion-corrupted reconstructions [53]. In CT time-decomposition methods, the artifact reduction already achieved by fixing an arbitrary interpolation [40], [44] could be improved by data-informed joint optimization. For instance, studies of drainage flows have tracked changes in successive radiographs to tune the time-interpolation to specific datasets using a maximum *a posteriori* expectation-maximisation (MAP-EM) algorithm [33]. This analysis paved the way towards data-informed interpolation in CT, but more work is needed to investigate the performance of these methods in different CT applications.

Here, we address this research need by using piecewise linear interpolation (PLI) in time to enable the data-informed reconstruction of single CT datasets at high accuracy. PLI is chosen because it can model both slow drifting motions and quasi-instantaneous events, as suggested by PET studies considering the more general B-spline interpolation [47], [48], [49]. The artifact reduction obtained with the proposed PLI approach is compared to a Fourier decomposition in time [44].

The specific contributions of this work are the following: (1) While previous CT work explored only fixed time interpolation [9], [38], [39], [40], [41], [42], [43], [44], the proposed PLI framework provides flexibility in the breakpoint selection. This flexibility allows to adjust the interpolation parameters to each dataset, thereby building towards data-informed joint optimization in dynamic CT. (2) Previous studies examined artifacts generated with little variations in the time-evolution [21], [23], [30], [40], [43], [44]. In contrast, by considering artifacts created from a variety of motion-progress profiles, the proposed method is evaluated on a more exhaustive range of artifacts, showing that PLI can represent both continuous and sudden motions. (3) The compactness of the PLI functions reduces the algorithmic cost associated with the CT forward and backward projections, compared to using non-compact interpolation functions [44]. (4) The artifact reduction is examined in the context of high-accuracy CT datasets that are compromised by small motions, a problem which has received little attention in the literature. In particular, PLI is motivated by its better representation of sub-pixel motion compared to PCI [9], [38], [39], [40], [41], [42]. High-accuracy reconstructions also require high number of iterations, and the

convergence of the method is characterized for different algorithms. (5) Artifact reduction is demonstrated on single CT scans as relevant to synchrotron and clinical CT, whereas most previous studies used time-series of multiple CT datasets. (6) Similar to [44], the problem is regularized via a spatio-temporal TV term and the Chambolle-Pock algorithm [54] is employed to optimize the resulting non-smooth formulation. However, we also propose a second-order accurate TV discretization, which is shown to reduce staircase artifacts compared to standard first-order discretizations [44].

The method is described in Section II and the algorithm and its implementation are provided in Section III. Section IV presents a validation of the method using numerical phantoms, and an analysis of the convergence and TV discretization schemes. Applications to several experimental synchrotron datasets are reported in Section V. Conclusions are presented in Section VI.

II. METHODS

A. Static Tomography Formulation

In transmission tomography, a penetrating radiation beam is directed onto an object to reconstruct the 3D linear attenuation field $f(x, y, z)$ from Beer-Lambert's law. To this end, a set of N_a absorption images represented by a vector \mathbf{b} are acquired while rotating the object around the z -axis along an angle θ . Although the proposed method can be applied in arbitrary 3D cone or fan beam configurations, a parallel beam is assumed here, such that the tomography problem can be decomposed along the z -axis. The sinogram \mathbf{b} provides a discrete sampling of the Radon transform Rf of the attenuation field in the x - y plane, which is defined as [2]:

$$Rf(\theta, \rho, z) = \int f(s \sin \theta + \rho \cos \theta, -s \cos \theta + \rho \sin \theta, z) ds, \quad (1)$$

where ρ is the distance of the line integral from the origin in the x - y plane. This sampling data is used to construct an approximate attenuation field \mathbf{f} that is discretized on a uniform grid of $N_x \times N_y \times N_z$ voxels, with N_i the number of voxels along the direction i . For simplicity, the 2D case with $N_z = 1$ and $N_x = N_y = N$ is considered in the following without loss of generality. In iterative reconstruction methods, the attenuation field is retrieved by considering an optimization problem in which the Radon transform is numerically approximated by a linear operator $\mathbf{A} \in \mathbb{R}^{N_a N \times N^2}$. This projection operator is defined such that $\mathbf{A}\mathbf{f}$ is a statistical estimate of the sinogram \mathbf{b} acquired experimentally [2]. Specifically, by writing \mathbf{A} in the block form $\mathbf{A}^T = (\mathbf{A}_1^T, \dots, \mathbf{A}_{N_a}^T)$, $\mathbf{A}_a \mathbf{f}$ is a statistical estimate of the experimental absorption line $\mathbf{b}_a \in \mathbb{R}^N$ acquired at the viewing angle $a \in [1, N_a]$.

The CT reconstruction is then formulated using the penalized weighted least square (PWLS) cost function as [2]

$$\mathbf{f}^* = \arg \min_{\mathbf{f}} \left\{ \frac{1}{2} \|\mathbf{A}\mathbf{f} - \mathbf{b}\|_{\mathbf{W}}^2 + \lambda \mathcal{R}(\mathbf{f}) \right\}, \quad (2)$$

in which the first term is a data fidelity term using the \mathbf{W} -weighted L_2 -norm $\|\mathbf{p}\|_{\mathbf{W}} = \|\mathbf{W}^{1/2} \mathbf{p}\|_2$, the second term $\mathcal{R}(\mathbf{f})$

is a regularization term enforcing spatial smoothness of the solution, and λ is a trade-off parameter between the two. The weight matrix \mathbf{W} is the diagonal matrix defined with the vector $(\mathbf{A}\mathbf{1})^{-1}$ on the diagonal, where $\mathbf{1}$ is the vector full of ones in \mathbb{R}^{N^2} and the inversion is performed element-wise. The minimum of the data fidelity term corresponds to the maximum likelihood estimate (MLE) under the assumption of Gaussian noise in the measurements \mathbf{b} [2]. In low-dose tomography, this Gaussian assumption breaks down, and the Poisson statistics of photon counts have to be considered [42], which impacts the spatial resolution in motion-corrected CT [55].

B. Dynamic Tomography Formulation

If the object moves during the acquisition of duration T , then the attenuation field f also varies as a function of time t . By using a set of M linearly independent time-interpolation functions $\{\psi_k\}_{k=1}^M$ on the interval $[0, T]$, the time-varying attenuation field can be approximated by [44]

$$f(x, y, t) \approx f_M(x, y, t) = \sum_{k=1}^M \psi_k(t) \varphi_k(x, y), \quad (3)$$

in which $\varphi_1(x, y), \dots, \varphi_M(x, y)$ are the spatial weights of the time decomposition. These functions are discretized on a uniform spatial grid as weight vectors Φ_1, \dots, Φ_M , which are vertically concatenated into a single vector $\Phi \in \mathbb{R}^{MN^2}$ written as $\Phi^T = (\Phi_1^T, \dots, \Phi_M^T)$. The time interval $[0, T]$ of the acquisition is uniformly discretized such that the N_a viewing angles are acquired at a constant rate, with temporal increment $\Delta t = T/(N_a - 1)$. The attenuation field f is thus approximated by a discretized field \mathbf{f} defined by

$$\mathbf{f} = \begin{pmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_{N_a} \end{pmatrix} = \mathbf{P}_M \Phi = \begin{pmatrix} \sum_{k=1}^M \psi_k(0) \Phi_k \\ \vdots \\ \sum_{k=1}^M \psi_k(T) \Phi_k \end{pmatrix}, \quad (4)$$

where the matrix $\mathbf{P}_M \in \mathbb{R}^{N_a N^2 \times MN^2}$ represents the time interpolation operator. The CT problem is then written using a dynamic estimate $\mathbf{B}\mathbf{P}_M \Phi$ of the data \mathbf{b} with

$$\Phi^* = \arg \min_{\Phi} \left\{ \frac{1}{2} \|\mathbf{B}\mathbf{P}_M \Phi - \mathbf{b}\|_{\mathbf{W}}^2 + \lambda \mathcal{R}_t(\Phi) \right\}, \quad (5)$$

where \mathcal{R}_t is a dynamic regularization operator, and the projection matrix \mathbf{A} is written as the block diagonal matrix $\mathbf{B} \in \mathbb{R}^{N_a N \times N_a N^2}$ with the sub-matrices $(\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_{N_a})$ as diagonal blocks.

C. Piecewise Linear Interpolation in Time

Different sets of interpolation functions $\{\psi_k\}$ can be considered for the time interpolation in dynamic CT. In the following, piece-wise linear interpolation (PLI) is proposed as a natural option to suppress motion artifacts. To motivate this choice, we consider the 2D numerical phantom shown in Fig. 1, developed specifically for this study. This 2D phantom is inspired from

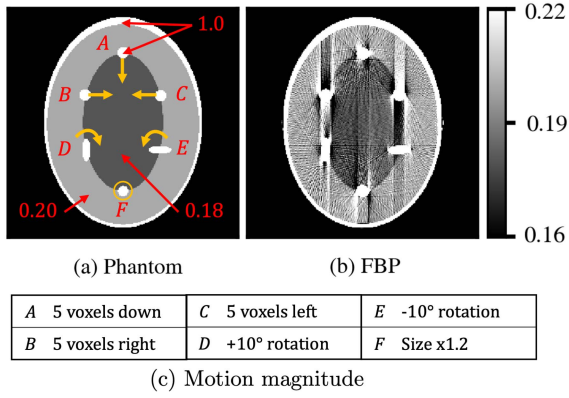


Fig. 1. (a) Numerical phantom of 250×250 voxels with translation, rotation, and scaling of six ellipses A-F, (b) FBP reconstruction with artifacts for a uniform motion, and (c) table of motion magnitude for the six ellipses. Attenuation values in AU are indicated in red on (a).

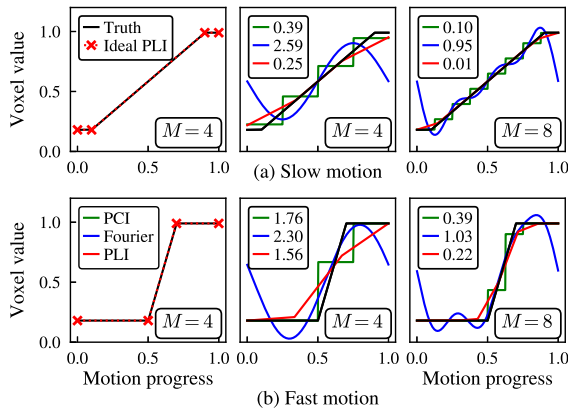


Fig. 2. Comparisons of different time interpolations to approximate the hypothetical time-varying attenuation of a single voxel during (a) slow and (b) fast motion. The number of interpolation degrees of freedom M is the same for the different approaches, and the legend indicates the mean square error of the interpolation in percentage ($100 \times \int_0^1 (f(t) - f^*(t))^2 dt$). The interpolations are performed by minimizing this value.

the Shepp-Logan phantom, and consists of a superposition of ellipses with attenuation of 1.0 arbitrary units (AU) that are translated, rotated, and scaled to simulate motion artifacts. Within the phantom, an elliptic interface separates the background at the center (0.18 AU) from its slightly more attenuating surrounding (0.20 AU). The motions of ellipses A-E are rigid, whereas ellipse F keeps a constant pixel attenuation value during scaling, thereby representing a non-rigid motion with mass addition. By considering the uniform motion of the ellipses over about 5 voxels, the results in Fig. 1 show that the 0.02 AU difference signal is significantly deteriorated by motion artifacts when using FBP reconstruction. For reference, the reconstruction shown in Fig. 1(b) corresponds to the constant linear motion later examined in Fig. 3(a).

Even though the motions considered remain relatively simple, these affine deformations replicate the motion expected in the synchrotron measurements discussed in Section V-B. Specifically, this phantom is designed to examine how

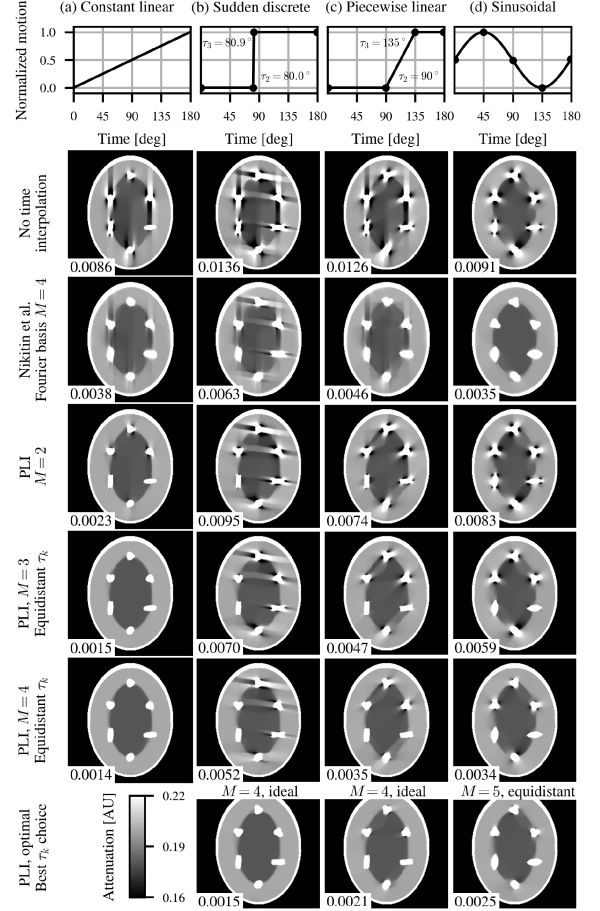


Fig. 3. Reconstructions of the 2D phantom presented in Fig. 1(a) using the CP implementation of the dynamic PLI and a Fourier-based interpolation [44], both with 10^4 iterations. The values at the bottom left of each image indicate the root mean square error within the outer ellipse between the true time average of the phantom and its estimate. The first row indicates the motion-progress evolution over the time $[0^\circ, 180^\circ]$ of the acquisition for each column. The circles on these curves indicate the location in time of the ideal breakpoints for PLI, for which the reconstructions are shown in the last row.

the reconstruction of the limited contrast of 0.02 AU is altered by artifacts resulting from the motion of elements with 1.0 AU attenuation, which directly represents the magnitude of the attenuating porous solid (1.0 /cm) and of the gas-phase signal of interest (0.02 /cm) in these high-accuracy measurements.

Let us now consider a fixed voxel in this phantom, initially located in the central phase with an attenuation value of 0.18 AU, in which one of the ellipses with attenuation of 1.0 AU moves during the CT acquisition. The attenuation values of such a voxel over a uniform motion are presented in the left column of Fig. 2 for a slow and a fast motion. Even if the motion is globally uniform, the functions that represent the evolution in voxel values are piecewise linear functions with four non-uniform breakpoints unique to each voxel, thus motivating the use of PLI in time. However, without priori knowledge of the ideal breakpoint locations, the result of an uniform PLI with M degrees of freedom is not exact, as shown in the central and

right columns of Fig. 2 in which the PLI is compared to PCI [9], [38], [39], [40], [41], [42] and Fourier interpolation [44].

Fig. 2 shows that the interpolation using harmonic functions results in large errors at the extreme of the motion, whereas PCI and PLI perform better there. However, with PCI the reconstructed solution is discontinuous, which can result in artifacts because of discontinuities in the sinogram estimate. Instead, PLI provides a continuous solution that can represent both gradual transitions and sharp gradients. Quantitative comparisons of the mean square error $\int_0^1 (f(t) - f^*(t))^2 dt$ between the interpolated voxel value $f(t)$ and its true value $f^*(t)$ are also given in the legends of Fig. 2, showing that PLI significantly reduces the error in certain cases. Finally, PCI and PLI can enable more flexible interpolation by varying not only the number of interpolation coefficients M , but also the breakpoint locations. In the following, we only consider voxel-independent breakpoints, even though voxel-dependent breakpoints would ideally be required to accurately represent motions larger than one voxel in magnitude.

PLI has already been employed in CT as a technique for the reduction of large datasets [56]. By representing data from about 100 successive CT acquisitions with a PLI model using only 10 breakpoints, this previous study demonstrated data denoising and compression by a factor of about 10. However, PLI was only applied *a posteriori* on datasets reconstructed through static formulations similar to (2), such that the temporal resolution was limited to the acquisition time of an entire rotation and motion artifacts were not directly addressed. In contrast, PLI is here performed during reconstruction by solving (5) with P_M chosen by selecting the M interpolation functions ψ_k from a linearly independent family of compact hat functions with values in $[0, 1]$. Specifically, the time interval of the acquisition $[0, T]$ is discretized using a set of M breakpoints $\{\tau_i\}$ with $M \geq 2$ such that

$$\begin{cases} \tau_1 = 0 \\ \tau_i < \tau_{i+1}, \quad 1 \leq i \leq M-1 \\ \tau_M = T \end{cases} \quad (6)$$

At any time $t \in [0, T]$, the continuous attenuation field $f(x, y, t)$ is then estimated by its discretization \mathbf{f}_t constructed by linear interpolation. Specifically, using the index k defined such that $\tau_k \leq t < \tau_{k+1}$, \mathbf{f}_t is linearly interpolated between the two closest attenuation fields using weights w_t as

$$\begin{aligned} \mathbf{f}_t &= \frac{\tau_{k+1} - t}{\tau_{k+1} - \tau_k} \Phi_k + \frac{t - \tau_k}{\tau_{k+1} - \tau_k} \Phi_{k+1} \\ &= (1 - w_t) \Phi_k + w_t \Phi_{k+1}. \end{aligned} \quad (7)$$

The spatial weight Φ_k corresponds to the attenuation field \mathbf{f}_t at the time of the k^{th} breakpoint $t = \tau_k$. This interpolation results in a bidiagonal structure of the matrix \mathbf{P}_M .

D. Joint Interpolation Optimization Using Variable Breakpoints

In the previous section, PLI was considered with constant and equidistant breakpoints τ_k . Instead of remaining fixed, these breakpoints can be varied such that the interpolation is adapted to

the specific sample motion of individual dataset. The following general formulation is thus considered

$$(\Phi^*, \mathbf{P}^*) = \arg \min_{\Phi, \mathbf{P} \in \mathcal{P}_M} \left\{ \frac{1}{2} \|\mathbf{B}\mathbf{P}\Phi - \mathbf{b}\|_{\mathbf{W}}^2 + \lambda \mathcal{R}_t(\Phi) \right\}, \quad (8)$$

in which both the solution, and the interpolation operator are jointly optimized. Such joint formulations have been considered in the dynamic PET literature, with different attempts to find optimal interpolations [50], [51], [52]. In CT, one study tracked changes in the total attenuation of successive radiographs to tune the location of time-interpolation breakpoints [33]. However, one limitation of this formulation, is that the cost function is not jointly convex in (Φ, \mathbf{P}) , which makes this formulation challenging to optimize simultaneously [52].

The simplest approach to this problem is to construct a best-guess projector \mathbf{P} , and solve the convex minimization problem from (5), as done in past studies with different algorithms keeping a fixed projector [33], [50]. The present analysis instead compares the results obtained by solving (5) with different choices of breakpoints, such that the projector is optimized *a posteriori*. Specifically, in the case of numerical phantoms with a known solution, this analysis is performed quantitatively and an optimal set of breakpoints is constructed.

This optimization approach represents a first step towards solving (8). In particular, optimizing the breakpoints *a posteriori* provides a lower bound of the extent of artifact reduction achievable by joint optimization methods. With real datasets no quantitative errors can be computed, and insights from the phantom study are instead used to select the breakpoints. Different data-informed heuristics can guide the interpolation choice, including the visual or automated identification of specific artifacts [10], [32], the tracking of sudden changes in the sinogram [33], or the limitation of M and the number of coefficients by minimizing L_0 norms.

III. ALGORITHM IMPLEMENTATION

A. Total Variation Regularization

In this work, TV regularization is used. The regularization term $\mathcal{R}_t(\mathbf{f})$ in the cost functions defined in (5) and (8) is set to the space-time TV of the spatial weights Φ . For a continuous field $f(x, y, t)$, the TV is often written using the $L_{2,1}$ norm as [9], [44], [57]

$$R_t(f) = \iint \int_{t=0}^T \sqrt{(\partial_x f)^2 + (\partial_y f)^2 + \mu (\partial_t f)^2} dt dx dy, \quad (9)$$

where μ is a time-regularization parameter. Other spatio-temporal regularizations have been proposed, including TV with separation between the space and time terms [38], or non-local patch-based regularization that is less sensitive to noise [39]. In the dynamic case, using (9) to evaluate the discrete TV regularization with time increments $\Delta t = T/(N_a - 1)$ increases the number of operations by a factor of $O(N_a)$ compared to the static case, which is prohibitive in practice. Instead, in the present work, a simplified regularization term is used, which

only increases the algorithmic cost by a factor of $O(M)$. Specifically, for the field $f(x, y, t)$ discretized as a vector $\mathbf{f} = \mathbf{P}_M \Phi$ as defined previously, the TV regularization is computed as

$$\mathcal{R}_t(\Phi) = \sum_{p=1}^{N^2} \sum_{k=1}^M \frac{\sqrt{\sum_{d=1}^{N_d} [\mathbf{D}_d \Phi]_{p,k}^2}}{M} = \sum_{p=1}^{N^2} \sum_{k=1}^M \frac{[\mathbf{g}(\mathbf{D}\Phi)]_{p,k}}{M}, \quad (10)$$

in which the summations are performed over every pixel p and every breakpoint k . The N_d square matrices \mathbf{D}_d are finite-difference discretizations of the gradient operators ∂_x, ∂_y , and ∂_t along x, y and t , which depend on the TV scheme considered. The matrix $\mathbf{D} \in \mathbb{R}^{N_d M N^2 \times M N^2}$ is the vertical concatenation of these matrices $\mathbf{D} = (\mathbf{D}_1^T, \mathbf{D}_2^T, \dots, \mathbf{D}_{N_d}^T)^T$, and the function \mathbf{g} returns the vector in $\mathbb{R}^{M N^2}$ of the local gradient L_2 -norms, such that $\mathcal{R}_t(\Phi) = \|\mathbf{g}(\mathbf{D}\Phi)\|_1/M$ using the standard L_1 -norm for vectors. The matrices \mathbf{D}_d are characterized for different TV discretizations in Section IV-E. For static algorithms, there is only one static image field, and \mathcal{R} from (2) is defined equal to \mathcal{R}_t in (10) with $M = 1$ and $\mu = 0$ by convention.

Note that if two PLI breakpoints are located close to one another, the time-gradient between the two corresponding fields Φ_k and Φ_{k+1} will increase significantly. However, following the hypothesis that motion is as likely to occur smoothly or suddenly, our implementation chooses to equally penalize between successive fields Φ_k , independently of the breakpoint separation. Therefore, in the matrices \mathbf{D}_d used to discretize the gradient operator over time, the coefficients are multiplied by the time-separation between the breakpoints, such that the time-difference terms appearing under the square root in (10) are equal to $\mu(\Phi_{p,k+1} - \Phi_{p,k})^2$ for the upwind discretization. Besides, to simplify notations, the time-regularization parameter is incorporated as $\sqrt{\mu}$ within the matrices \mathbf{D}_d used to discretize the gradient operator over time.

B. Subgradient Descent Optimization

In order to solve (2) and (5), different optimization algorithms are considered. Because total variation (TV) regularization is used, the objective function is not everywhere differentiable and standard smooth optimization algorithms such as gradient descent approaches may not converge [58]. However, these algorithms are easy to implement and remain valuable for development and prototyping. For this reason, the subgradient descent (SD) algorithm is considered first. Iterations of the SD algorithm for (2) are computed as

$$\mathbf{f}^{n+1} = \mathbf{f}^n - \gamma_n \mathbf{Q} (\mathbf{A}^T \mathbf{W} (\mathbf{A} \mathbf{f}^n - \mathbf{b}) + \lambda \mathbf{h}), \quad (11)$$

in which the index n is the iteration number, γ_n is the step size, \mathbf{Q} is a preconditioner matrix, and \mathbf{h} is a subgradient of \mathcal{R} at \mathbf{f}^n . To retrieve the SIRT algorithm in the case $\gamma_n = 1$ and $\lambda = 0$ for which the objective function is smooth, the preconditioner \mathbf{Q} is defined as the diagonal matrix with the vector $(\mathbf{A}^T \mathbf{1})^{-1}$ on the diagonal, where the inversion is performed element-wise [59]. The SD update for solving the dynamic problem from (5) is

similarly written as

$$\Phi^{n+1} = \Phi^n - \gamma_n \mathbf{Q}_M (\mathbf{C}_M^T \mathbf{W} (\mathbf{C}_M \Phi^n - \mathbf{b}) + \lambda/M \mathbf{L}_M \mathbf{h}), \quad (12)$$

where the matrix \mathbf{C}_M is defined by $\mathbf{C}_M = \mathbf{B} \mathbf{P}_M$, the diagonal preconditioner \mathbf{Q}_M is prescribed by its diagonal $(\mathbf{C}_M^T \mathbf{1})^{-1}$, \mathbf{h} is a subgradient of \mathcal{R}_t at $\mathbf{P}_M \Phi^n$, and \mathbf{L}_M is an added preconditioner prescribed for the regularization. The preconditioner $\mathbf{L}_M \in \mathbb{R}^{M N^2 \times M N^2}$ is defined as the block diagonal matrix with matrices $M \lambda_k \mathbf{I}_{N^2}$ on the diagonal for k from 1 to M with $\lambda_k = (\tau_{k+1} - \tau_{k-1})/(2T)$, where by convention $\tau_0 = 0$ and $\tau_{M+1} = T$. This preconditioner is added to enforce a comparable level of regularization to every Φ_k for any arbitrary set of breakpoints.

Although the above SD update is easy to implement, its convergence remains limited compared to the guaranteed $\mathcal{O}(1/n)$ convergence rate of GD for smooth objective functions. In particular, if the step size is kept constant, convergence with SD is only guaranteed at a $\mathcal{O}(1/\sqrt{n})$ rate and not necessarily towards the optimal value [58]. However, with a proper reduction in step size, convergence towards the optimal can be guaranteed. In practice, SD often converges faster than the theoretical rate before reaching a plateau [58]. In this work, SD is employed both using a constant step size $\gamma_n = 1$, and a decaying step size for which $\gamma_n = 1$ is kept for $n < 100$, while the reduction $\gamma_n = (100/n)^{0.5}$ is used for $n \geq 100$. The series of decaying step sizes is thus square summable but not summable, which guarantees convergence towards the optimal with a rate of at least $\mathcal{O}(1/\sqrt{n})$ [58].

C. Primal-Dual Optimization

Alternatively to gradient descent methods, splitting methods, such as the Douglas-Rachford splitting, the alternating direction method of multipliers (ADMM), or the primal-dual proximal algorithm from Chambolle and Pock (CP) [54] can accelerate convergence for non-smooth cost functions. With the non-smooth TV regularization considered, artifact reduction is hindered by the limited convergence of the SD algorithm. Therefore, the better-converging CP algorithm is preferred, which guarantees a $\mathcal{O}(1/n)$ decrease in the primal-dual gap. The CP algorithm uses proximal operators to accelerate the convergence where the objective function is non-smooth. A proximal operator provides a vector close to the input \mathbf{y} that lowers the value of a function F as defined by

$$\text{prox}_{\nu F}^{\Sigma}(\mathbf{y}) = \arg \min_{\mathbf{z}} \left\{ F(\mathbf{z}) + \frac{1}{2\nu} \|\mathbf{y} - \mathbf{z}\|_{\Sigma^{-1}}^2 \right\}, \quad (13)$$

where $\|\cdot\|_{\Sigma^{-1}}$ is the L_2 norm weighted by a matrix Σ^{-1} , and ν is a trade-off parameter between function minimization and proximity to the argument. To match the CP formulation [54], the cost function from (2) is rewritten as $F_1(\mathbf{A}\mathbf{f}) + F_2(\mathbf{D}\mathbf{f})$, where $F_1(\mathbf{p}) = \frac{1}{2} \|\mathbf{p} - \mathbf{b}\|_{\mathbf{W}}^2$, and $F_2(\mathbf{p}) = \lambda \|\mathbf{g}(\mathbf{p})\|_1/M$ where $M = 1$ is used by convention for the static case. The update rule for the static CP algorithm is

thus computed as [44], [54]

$$\mathbf{y}_1^{n+1} = \text{prox}_{F_1^*}^{\Sigma_1}(\mathbf{y}_1^n + \Sigma_1 \mathbf{A} \tilde{\mathbf{f}}^n) = \frac{\mathbf{y}_1^n + \Sigma_1(\mathbf{A} \tilde{\mathbf{f}}^n - \mathbf{b})}{1 + \Sigma_1 \mathbf{W}^{-1} \mathbf{1}} \quad (14a)$$

$$\mathbf{y}_2^{n+1} = \text{prox}_{F_2^*}^{\Sigma_2}(\mathbf{y}_2^n + \Sigma_2 \mathbf{D} \tilde{\mathbf{f}}^n) = \frac{\lambda(\mathbf{y}_2^n + \Sigma_2 \mathbf{D} \tilde{\mathbf{f}}^n)}{\max(\lambda, \mathbf{g}(\mathbf{y}_2^n + \Sigma_2 \mathbf{D} \tilde{\mathbf{f}}^n))} \quad (14b)$$

$$\mathbf{f}^{n+1} = \mathbf{f}^n - \mathbf{T}_1 \mathbf{A}^T \mathbf{y}_1^{n+1} - \mathbf{T}_1 \mathbf{D}^T \mathbf{y}_2^{n+1} \quad (14c)$$

$$\tilde{\mathbf{f}}^{n+1} = \mathbf{f}^{n+1} + \kappa(\mathbf{f}^{n+1} - \mathbf{f}^n) \quad (14d)$$

in which the divisions indicated by fractions and the maximum are performed element-wise, F^* denotes the Fenchel conjugate of function F , \mathbf{y}_1 and \mathbf{y}_2 are the dual variables, $\tilde{\mathbf{f}}$ is the over-relaxation in the primal variable controlled by the extrapolation factor κ , and the primal and dual preconditioning matrices Σ_1 , Σ_2 , and \mathbf{T}_1 are computed as proposed by Chambolle and Pock with $\alpha = 1$ using notation from their article [60]. With these definitions, $\Sigma_1 = \mathbf{W}$ such that the denominator in the update of \mathbf{y}_1^{n+1} simplifies to a scalar division by two. The preconditioners Σ_2 , and \mathbf{T}_1 depend on the TV scheme used. For the explicit expressions of Σ_2 and \mathbf{T}_1 , the reader is referred to the open-source code available online. In this work, κ is kept equal to 1. Details on the derivation of the CP algorithm can be found in previous articles [44], [54], [60].

In the case of the PLI objective of the dynamic problem from (5), the update from the CP algorithm reads

$$\mathbf{y}_1^{n+1} = \text{prox}_{F_1^*}^{\Sigma_3}(\mathbf{y}_1^n + \Sigma_3 \mathbf{C}_M \tilde{\Phi}^n) = \frac{\mathbf{y}_1^n + \Sigma_3(\mathbf{C}_M \tilde{\Phi}^n - \mathbf{b})}{1 + \Sigma_3 \mathbf{W}^{-1} \mathbf{1}} \quad (15a)$$

$$\mathbf{y}_2^{n+1} = \text{prox}_{F_2^*}^{\Sigma_4}(\mathbf{y}_2^n + \Sigma_4 \mathbf{D} \tilde{\Phi}^n) = \frac{\tilde{\lambda}(\mathbf{y}_2^n + \Sigma_4 \mathbf{D} \tilde{\Phi}^n)}{\max(\tilde{\lambda}, \mathbf{g}(\mathbf{y}_2^n + \Sigma_4 \mathbf{D} \tilde{\Phi}^n))} \quad (15b)$$

$$\Phi^{n+1} = \Phi^n - \mathbf{T}_2 \mathbf{C}_M^T \mathbf{y}_1^{n+1} - \mathbf{T}_2 \mathbf{L}_M \mathbf{D}^T \mathbf{y}_2^{n+1} \quad (15c)$$

$$\tilde{\Phi}^{n+1} = \Phi^{n+1} + \kappa(\Phi^{n+1} - \Phi^n) \quad (15d)$$

in which the divisions indicated by fractions and the maximum are evaluated element-wise, and $\tilde{\lambda} = \lambda/M$. The primal and dual preconditioning matrices $\Sigma_3 = \mathbf{W}$, Σ_4 , and \mathbf{T}_2 are again computed as proposed by Chambolle and Pock [60].

In the following, the static SD and CP algorithms, corresponding to the updates given by (11) and (14) respectively, are initialized using a zero solution field. In contrast, the two PLI algorithms corresponding to the updates given by (12) and (15) are initialized by the solution obtained after 200 iterations of the corresponding static algorithms.

D. Implementation and Scaling

One key benefit of using the linear time decomposition from (3) is that the current algorithm can leverage already optimized implementations of the forward and backward CT projections,

TABLE I
AVERAGE TIMES FOR ONE ITERATION OF THE PROPOSED ALGORITHMS
(NORMALIZED BY THE 7.2 S OF ONE SD ITERATION, $4 \times \text{BIN}$, $M = 1$)

N_z, N (bin)	132, 640 (4)				264, 1280 (2)			
M	1	2	4	8	1	2	4	8
SD	1.0	2.7	3.8	6.2	7.0	18	22	51
CP	1.3	3.9	4.8	8.3	8.7	24	36	68

represented by the operators \mathbf{A} and \mathbf{A}^T . Therefore, the proposed algorithm is implemented at a modular level in Python to facilitate its implementation in already optimized lower-level CT libraries such as TomoPy [61], TIGRE [62], Savu [63], CIL [64] or ODL [65]. The current implementation is using the CUDA routines of the ASTRA Toolbox for forward and backward projections [66].

Compared to previously proposed approaches using non-compact interpolation functions, PLI reduces the number of floating-point operations. Specifically, because of the bidiagonal structure of \mathbf{P}_M , the algorithmic cost associated with the CT projection at every iteration of the dynamic algorithm is only twice that of the equivalent static algorithm for all values of M , compared to an increase by a factor $\mathcal{O}(M)$ when using non-compact interpolation functions such as harmonic functions [44]. Indeed, each spatial weight Φ_k is only projected at the angles acquired between times τ_{k-1} and τ_{k+1} , instead of over the entire half-rotation. However, the global algorithmic cost associated with computing updates for the regularization term is still increased by a factor of M compared to the equivalent static algorithm, since a TV update is computed for each spatial weight Φ_k at every iteration.

The code was implemented to be compatible with both CPU and GPU. The multithreading available in Numpy was used for the CPU implementation, while the GPU version was developed using the CUDA convolutions and operations from PyTorch [67]. The TV routines for 4D data on CPU and GPU are provided in the package PyTV-4D (<https://github.com/eboigne/PyTV-4D>). Note that the choice of using separate Python libraries for the CT and TV calculations comes at the cost of an added transfer between host and device per iteration.

To accommodate large datasets, the implementation is directly parallelized along the z -direction by performing the operations successively on chunks of multiple slices at every iteration. For a parallel beam, the CT problem is purely separable along the z -direction and no specific considerations are required for this parallelization. In contrast, computing the TV update for slices z_1 to z_2 requires access to the data from slices $z_1 - 1$ to $z_2 + 1$, and this one slice overlap is accounted for in the implementation.

For a fixed number of angles N_a , the code execution time is expected to scale in $\mathcal{O}(N_z M N^2)$. To confirm this scaling, Table I reports the time of one iteration of the SD and CP algorithms for varying data size and values of M . These calculations are done with the second dataset discussed in Section V-B which is of size $N_z \times N \times N_a$. It is reconstructed into a volume of size $N_z \times N \times N$, while keeping $N_a = 2101$ constant for all binnings. $M = 1$ correspond to the static algorithms given by

(11) and (14). The calculations are performed using single precision on a node equipped with a GeForce GTX TITAN X GPU and an Intel Xeon E5-2697 v2 CPU with 24 cores. The results show that the code scales reasonably well, with both algorithms performing better than the asymptotic scaling suggest in most cases. However, the CP algorithm with $M = 2$ is almost twice slower than expected, showing the limitations of a high-level Python wrapper. The present implementation can be improved by using a compiled language to accelerate the code, as done in the implementations of other time-decomposition methods [44]. Alternatively, as explored in previous CT studies [42], [68], the convergence may be accelerated by binning down the data and up-scaling it during the iterations, although this approach was not tested here.

IV. VALIDATION

A. Ideal Phantom Results

In order to validate the proposed algorithm and to evaluate the capacities of the method to reduce motion artifacts, a numerical phantom analysis is first performed. The focus of this analysis is to quantitatively retrieve the limited contrast in the linear attenuation from the phantom shown in Fig. 1, while considering artifacts from the motion of highly attenuating elements. A range of different time histories of the sample motion is analyzed to further investigate the robustness of the method to different types of motions. Indeed, analytical studies of motion artifacts clearly showed that artifacts are heavily dependent on the time history of the motion, with characteristic patterns for continuous motion [69] and quasi-instantaneous events [70]. In the following, the time variable is expressed in units of degrees corresponding to the value of the rotating angle, such that the first half-rotation is acquired over the time interval $[0^\circ, 180^\circ]$.

The phantom images considered are squares of 250 voxels per side, which are projected with a parallel beam on a detector of 250 pixels over 200 equidistant angles covering a half-rotation of 180° . To construct the sinograms, the 2.0.0 version of the TomoPhantom toolbox is employed [71]. The use of the analytical Radon transform in TomoPhantom ensures that the constructed sinograms are free of aliasing errors arising from the discretization scheme employed in the tomography projector **A** [72]. Specifically, the analytical sinogram is constructed by evaluating the Radon integrals at the central pixel location, assuming a zero-width pixel and accounting for sub-pixel motion. TomoPhantom also constructs a time-series of the corresponding discretized phantom, which is however implemented without sub-pixel interpolation. Therefore, in order to create a time-series phantom that accounts for sub-pixel motion, the phantoms are generated in TomoPhantom from over-sampling by a factor of eight in the number of pixels, and then down-sampling using a 2D bilinear interpolation. The construction of the analytical sinogram **b** and the reconstruction results are not affected by this procedure. Indeed, the phantom time-series is only used to evaluate the RMS error between the reconstructed phantoms and the ideal solutions, which barely change when over-sampling by a factor of more than eight.

In a first analysis, four different time-history profiles of the motions are considered. Represented in Fig. 3, these motions describe either (a) continuous motion, (b) instantaneous motion event, (c) more intermittent motion, or (d) periodic motion. The sinograms reconstructed with the PLI method proposed in this work are compared with two reference algorithms: one that does not involve any time interpolation, and one time-interpolation method based on a Fourier decomposition [44]. The regularization parameters used in the PLI and Fourier-based implementations are chosen as $\lambda = 2^{-4}$ and $\mu = 2^{-2}$, and both algorithms are run for 10^4 iterations. The implementation of the Fourier-based method is the one made available publicly in the original article [44], in which the initial guess is the FBP reconstruction.

Time-averaged reconstructed solution fields \bar{f} are presented in Fig. 3, along with the root mean square errors to the ideal time-averaged phantom evaluated over the distribution of pixels contained within the outer ellipse. The static reconstructions highlight the large impact of the time history on the resulting artifacts. With a continuous motion (a), some elements feature only limited artifacts (ellipses A and E, as labeled in Fig. 1), while with a sudden motion (b), sharp visible streaks align along the angle corresponding to the time of the discrete event. This direct relation between the streak alignment and the time of the discrete event has been investigated in past analytical work [70], and is used in the present study to tune the breakpoints of the PLI.

With $M = 2$, PLI achieves significant artifact reduction in the case of the linear motion of type (a). This result is relatively unexpected: PLI with two breakpoints is not able to accurately represent the time-evolution of all voxels, because of the partial volume effect and the extent of the motion over 5 voxels. Yet, the delocalized streaks are still significantly reduced and remnant artifacts are localized in the vicinity of the moving elements. For the other time histories (b)-(d), PLI with $M = 2$ primarily removes the streaks aligned along the vertical direction of the first angle of projection, but the extent of the overall artifact reduction remains limited.

As M is increased to 3 and 4, the artifact reduction becomes more significant in all cases (a)-(d). For motion (a), significant artifact reduction is achieved with PLI for $M \geq 3$, whereas the results with $M = 4$ are comparable to using a Fourier-based interpolation for motions (b) and (c). However, by considering four non-equidistant PLI breakpoints and instead fixing them to their ideal values, the PLI algorithm retrieves solutions that are almost entirely free of artifacts in cases (b) and (c). Finally, as expected, the Fourier-based method does particularly well for a periodic motion (d), even though promising results are also achieved with PLI and $M = 5$.

This phantom study is ideal as there is no noise in the sinogram data, and the motion considered are monotonic despite being diverse. Under these ideal conditions, the results demonstrate the potential of the proposed algorithm to converge towards artifact-free solutions. In particular, this analysis shows that the flexibility in the PLI breakpoints is a major advantage over fixed interpolations when attempting to model a variety of motions. However, we note that both PLI and the Fourier-based methods

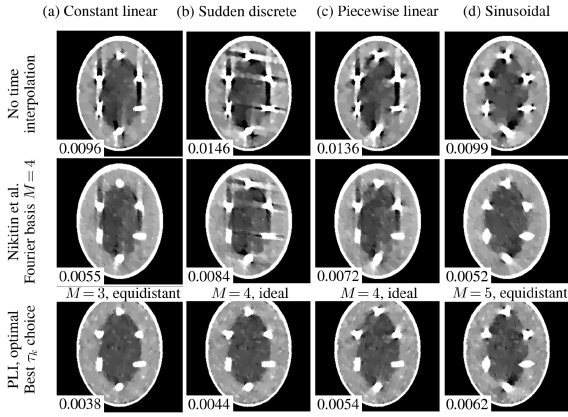


Fig. 4. Reconstructions of a noisy sinogram of the 2D phantom presented in Fig. 1(a) using the CP implementation of the dynamic PLI and a Fourier-based interpolation [44], both with 10^4 iterations. The grayscale, root mean square error values, and ideal PLI breakpoints are defined as in Fig. 3.

produce results with a broadening of the solid elements, which is attributed to the failure of the truncated time-decomposition to exactly represent the underlying evolution for all voxels. Even though this broadening is relatively over-represented in Fig. 3 by the narrow grayscale window in attenuation values, it remains a limitation of these methods. Finally, even though the motion of ellipse F is non-rigid, these 2D phantoms do not account for the unique types of motion artifacts that can be generated in 3D. Evaluating PLI on 3D motion is investigated with the synchrotron datasets discussed in Section V.

B. Impact of Realistic-Data Irregularities

Additional phantom results are discussed in this section to analyze how robust the method is to the presence of noise, and how it performs when the motion is non-monotonic or spatially-varying. First, Fig. 4 shows results similar to Fig. 3 when pixel-noise is added to the sinogram data. The noise added is sampled from a mean-zero Gaussian distribution with standard deviation of 1.0, assuming of a voxel size of 1.0. These results show that the performances of both PLI and the Fourier interpolation are similarly reduced, with slightly more remnant artifacts present in the reconstructions.

To illustrate the time-resolving capabilities of the PLI method, Fig. 5 compares the solution fields Φ_k reconstructed with PLI and $M = 4$ at the different breakpoints in the case of the sudden motion from Fig. 4(b). The results show that the streak artifacts are entirely eliminated at all times with the ideal breakpoints, whereas they dominate the solution fields in the case of equidistant breakpoints. The broadening of the solid elements is visible, but it remains clear that the algorithm retrieves the rotations of ellipses D and E.

Fig. 6 shows results examining how PLI performs when more realistic motions are considered. In the case of (a) a single sudden motion with vibrations, the artifact correction is reduced, and the performance gap between PLI and the Fourier-based method is cut down. In the second case (b) considering two sudden motions occurring at different times, the artifact reduction with $M = 4$

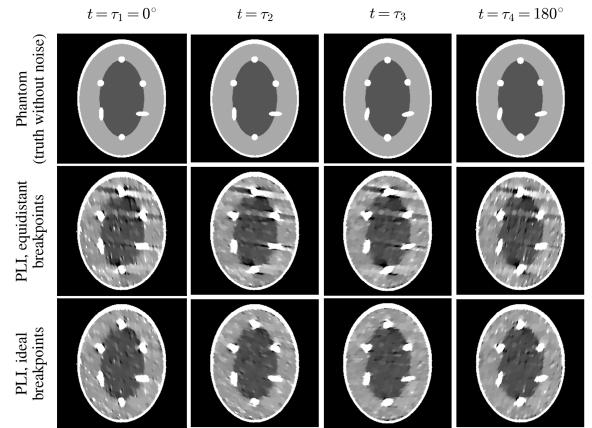


Fig. 5. Time-resolved results of the spatial weights Φ_k obtained with the PLI algorithm for $M = 4$, compared to the original phantom for the sudden discrete motion shown in Fig. 3(b). For the phantom and PLI with ideal breakpoints $\tau_2 = 81.0^\circ$ and $\tau_3 = 81.9^\circ$, whereas for PLI with equidistant breakpoints $\tau_2 = 60^\circ$ and $\tau_3 = 120^\circ$. The grayscale and the noisy sinogram used are the same as in Fig. 4.

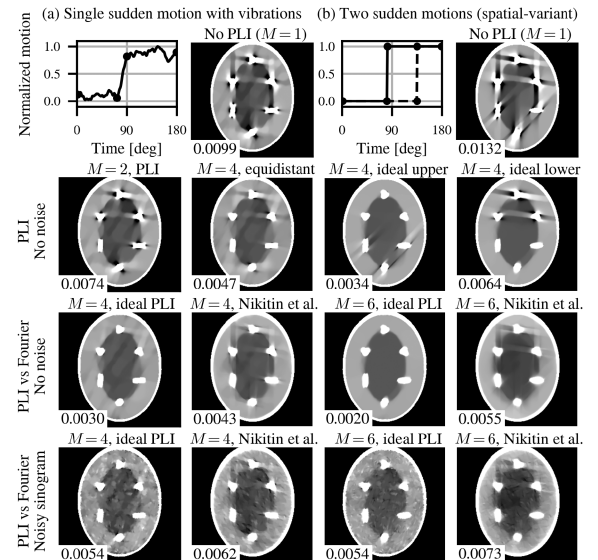


Fig. 6. Reconstructions of two complex motions of the 2D phantom presented in Fig. 1(a) using the CP implementation of the dynamic PLI and a Fourier-based interpolation [44], both with 10^4 iterations. The time histories for both motions are indicated in the first line plot. In the two left columns (a), a sudden motion occurs near 81° along with random vibrations, while in the two right columns (b), the motion is spatially variant: the upper half of the phantom suddenly moves at 81° , while the lower half moves at 135° . The grayscale levels, the noisy sinogram, and the root mean square error values are defined as in Fig. 4. The ideal PLI breakpoints in the first case (a) for $M = 4$ are $[0^\circ, 72^\circ, 90^\circ, 180^\circ]$, while in the second case (b) for $M = 6$ they are $[0^\circ, 81^\circ, 81.9^\circ, 135^\circ, 135.9^\circ, 180^\circ]$.

remains limited. This trend is consistent with the fact that PLI with $M = 4$ and global voxel-independent breakpoints fail to capture such spatially-varying motions. Specifically, the artifacts corresponding to the first and second motions can be reduced independently with $M = 4$. However, reducing artifacts from both motions simultaneously requires at least $M = 6$ global breakpoints.

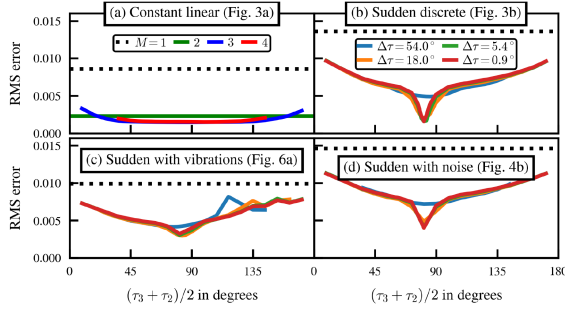


Fig. 7. Optimization of the PLI breakpoints using the root mean square error to the true time-averaged phantom for some of the results shown in Figs. 3, 4, and 6. In (a) M is increased with equidistant breakpoints for $M \geq 3$, while in (b)-(d) the two inner breakpoints are varied for $M = 4$. In (a), for $M = 3$ the abscissa is equal to the second-breakpoint τ_2 . For $M = 4$, the results are plotted along the mid-point between the two inner breakpoints $(\tau_3 + \tau_2)/2$ for diverse distances between these breakpoints $\Delta\tau = \tau_3 - \tau_2$. The first and last breakpoints remain fixed: $\tau_1 = 0^\circ$ and $\tau_M = 180^\circ$.

These results indicate that although the performance of PLI will suffer from irregularities in more realistic data, the flexible-breakpoint PLI method still arguably outperforms the fixed Fourier interpolation. Results in the case of spatially-varying motion also indicate that adding breakpoints could improve the artifact reduction. In particular, using voxel-specific breakpoints could improve future results, where coarse spatial parametrization would capture local motion while limiting the increase in the number of parameters, similar to the coarse B-spline interpolations used to represent DVFs [26], [27], [28], [29].

C. Analysis of Optimal Breakpoints

As shown in Figs. 3, 4, and 6, the improved performance of PLI over a fixed interpolation is largely due to the flexibility to select an ideal set of breakpoints. To quantitatively analyze the impact of the selection of breakpoints on the reconstruction quality, Fig. 7 reports the RMS error to the ideal phantom for different motions and sets of breakpoint. In the case of (a) a constant linear motion, the quality of the reconstruction is rather independent to the choice of M . In addition, as long as the inner breakpoints for $M \geq 3$ are not too close to the beginning or the end of the acquisition, the performance does not suffer from varying the breakpoints.

However, in the case of (b) a more sudden motion, there is a clear optimal selection of breakpoints. Specifically, it is necessary to have two breakpoints close to the time of the discontinuity, with a sharp reduction in the PLI performance if the center between these breakpoints is only placed a few degrees away from the characteristic angle of the motion. In contrast, the performance is rather insensitive to the separation between these two inner breakpoints, with only limited performance changes for separations $\Delta\tau < 30^\circ$. The addition of (c) vibrations to the motion reduces the sensitivity to the breakpoint location as the motion-progress is more continuous, while (d) pixel-noise barely impact these performance curves.

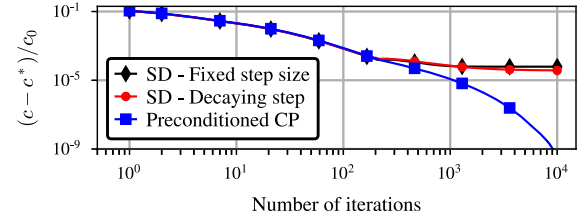


Fig. 8. Convergence of the normalized cost function from the dynamic CT formulation of (5) for different optimization algorithms. These results corresponds to the phantom shown in Fig. 3(a) with $M = 2$.

D. Optimization Algorithm and Convergence

In most CT applications, a few hundred iterations are sufficient to reach satisfactory convergence. In contrast, more iterations are necessary for quantitative CT results that guarantee errors below 1%. In addition, the use of TV regularization results in a non-smooth objective function for which standard gradient descent methods may fail to converge. However, despite the recognized importance of regularization in reducing motion artifacts [44], only few studies report on the convergence of the methods that are proposed. In this study, the maximum artifact reduction is achieved using the CP algorithm with at least 10^4 iterations, whereas only a limited artifact reduction is observed with the SD algorithm.

The convergence of the cost function c normalized with respect to its initial value c_0 and optimal value c^* is presented in Fig. 8 over 10^4 iterations for SD with fixed and decaying step size, and for the CP algorithm. The optimal value c^* corresponds to the value achieved with the CP algorithm after 10^5 iterations. These results confirm that SD reaches a plateau in which the algorithm oscillates due to a too large step size, whereas CP converges with a rate of almost $\mathcal{O}(1/n^2)$. With SD, reducing the step size slightly improves the convergence. Note that $c = c_0$ only when the number of iterations is equal to 0 which does not appear in this log-log graph, and explains why all values shown are below 1.

Fig. 9 shows a comparison of the reconstructions of the time-averaged solution field \bar{f} obtained after 10^4 iterations for the different algorithms. The results prove that, although the SD convergence only plateaus after about 3000 iterations, the better convergence of CP is directly related to a further reduction in streak artifacts. In particular, the red arrows in Fig. 9 identify regions where artifacts visible in the SD results are eliminated with CP. In addition, even though reducing the step size with SD eliminates the checkerboard artifacts visible in Fig. 9, it does not provide the artifact reduction achieved by the CP algorithm. For the CP algorithm, the results achieved with 10^5 iterations do not yield any visible changes in the solution field compared to 10^4 iterations. However, visible changes are observed when comparing the solutions reconstructed with less than 10^4 iterations. In particular, the results achieved with CP after 500 iterations resemble those achieved with SD after 10^4 iterations.

Therefore, this analysis indicates that algorithm convergence should be carefully examined when evaluating artifact reduction, particularly for high-accuracy reconstructions. In the present

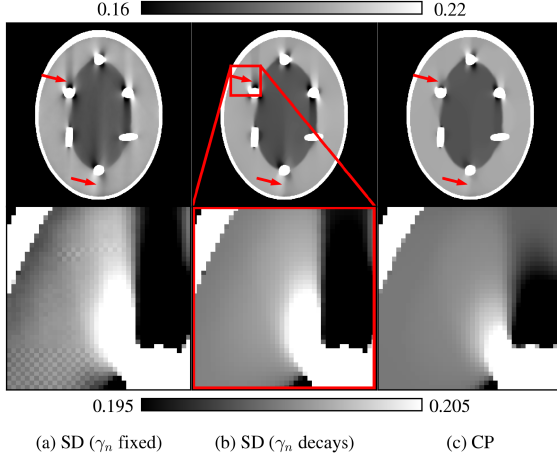


Fig. 9. Comparisons of the reconstructions obtained with the different algorithms after 10^4 iterations. The phantom case is the same as considered in Fig. 8, and corresponds to the motion shown in Fig. 3(a) with $M = 2$. The second row is a zoom into the results from the first row. The two colorbars indicate the attenuation in AU for the first and second rows.

work, the focus is on high-accuracy measurements and high number of iterations is required. In addition, although gradient descent approaches such as SD are faster to implement and convenient for prototyping, the present results confirm that gradient descent methods fail to converge when non-smooth TV objectives are considered. As a result, these methods should be avoided when high-accuracy reconstructions are desired. The better-performing CP algorithm with 10^4 iterations is thus used in the remainder of this study.

E. TV Discretization Scheme

In this work, TV regularization is used to drive the convergence towards artifact-free solutions. One limitation of TV regularization is the “staircase” effect, in which over-regularized images are recovered with undesirable regions of piecewise-constant values. This effect appears in the results obtained in this work, and a new method is considered to reduce their impact. Several approaches have been proposed to minimize the staircase effect, such as using higher order derivatives [2], [57], a general form of the total variation [73], or non-local regularization [39], [74]. Alternatively, the solution can be regularized by explicitly minimizing motion artifacts metrics, such as image entropy and positivity that quantify the magnitude of artifacts [10].

To reduce staircase artifacts in this study, a different approach is taken. Instead of considering higher-order derivatives in the definition of the TV [2], [57], higher-order approximations of the squares of the first-order derivatives in (9) are considered. This approach is motivated by recognizing that although the standard TV defined by (10) is invariant to rotation, its discretization using finite differences may not be. Indeed, the spatial gradients are typically evaluated using a first-order finite difference scheme such as

$$(\partial_x f)_{i,j} = f_{i+1,j} - f_{i,j}, \quad (\partial_y f)_{i,j} = f_{i,j+1} - f_{i,j}, \quad (16)$$

TABLE II
NUMERICAL TV SCHEMES CONSIDERED IN THIS WORK, ILLUSTRATED FOR THE 2D CONTINUOUS FIELD $f(x, y)$ DISCRETIZED AS $f_{i,j}$

TV scheme	Discretization of $(\partial_x f)_{i,j}^2$	Order
Upwind	$(f_{i+1,j} - f_{i,j})^2$	1
Downwind	$(f_{i,j} - f_{i-1,j})^2$	1
Central	$\frac{1}{4} (f_{i+1,j} - f_{i-1,j})^2$	2
Hybrid	$\frac{1}{2} (f_{i+1,j} - f_{i,j})^2 + \frac{1}{2} (f_{i,j} - f_{i-1,j})^2$	2

in which the indexes i and j of the 2D discrete field $f_{i,j}$ are along the x and y -axis of the 2D continuous field f . This discretization can be interpreted as the upwind discretization used to numerically solve partial differential equations with unit step size. This scheme is commonly used in TV calculations because it provides satisfactory results in most applications while remaining simple to implement [2]. However, the TV discretized using this scheme is not rotation-invariant in the following sense: at an image invariant by rotation \mathbf{f} , a subgradient \mathbf{h} of the TV function \mathcal{R} from (10) is not itself invariant by rotation. Instead, second-order approximations of $(\partial_x f)_{i,j}^2$ and $(\partial_y f)_{i,j}^2$ result in a discretized TV function satisfying this property.

In the literature, the idea of considering higher order neighborhood has been discussed (Chapter 1 of [2]), and a few studies have used schemes with better than first-order accuracy [36], [75], [76]. However, only limited research has considered how higher-order discretization can minimize staircase artifacts, particularly in the context of dynamic CT. Four different TV discretization schemes are here compared, including a new hybrid discretization that is proposed to reduce staircase artifacts. The discretization of $(\partial_x f)_{i,j}^2$ for these four schemes are explicitly defined in Table II. The discretization of $(\partial_y f)_{i,j}^2$ is similarly defined by permuting the i and j indices. Extensions to the 3D and 4D cases are also performed by extension along the z and t -axis.

Fig. 10 compares the reconstructions retrieved with the different TV discretizations after 10^4 iterations of the CP algorithm. With all schemes, the sharp solid edges are preserved. However, the upwind and downwind schemes both feature staircase artifacts, whereas these artifacts are absent with the proposed hybrid discretization scheme. The central scheme is also second-order accurate, but the difference between direct neighbor voxels is never computed in this scheme, which results in the checker-board pattern visible in the results. Because of the key role of TV regularization in penalizing streak artifacts and nudging the convergence towards smooth reconstructions, overall better results are achieved with the hybrid scheme. Therefore, this hybrid scheme is used in all the results shown in this study except those of Fig. 10.

V. APPLICATION TO EXPERIMENTAL DATA

A. Application to a Bone Tissue Experiment

Synchrotron measurements of the wettability of bone tissue scaffolds acquired at the SYMREP beamline from Elettra have been shared on TomoBank [77], [78]. This dataset is used here, and consists of a 180° tomography acquisition of a scaffold

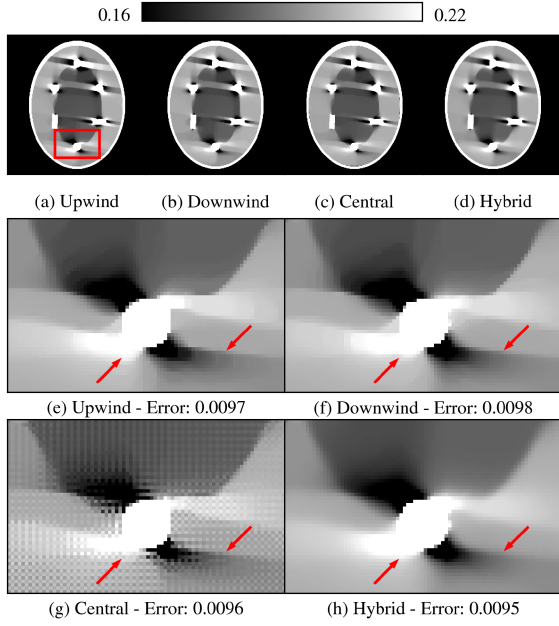


Fig. 10. Comparisons of the reconstructions obtained after 10^4 iterations of the different TV discretization schemes. The phantom case is the sudden discrete motion shown in Fig. 3(b) with $M = 2$ and the CP algorithm. Images (e)-(h) are zoom into the red rectangle shown in (a), and the error values specified in the captions indicate the root mean square error within the outer ellipse between the true time average of the phantom and its estimate.

embedded with liquid water. The wettability of the bio-structure is such that air bubbles form within the scaffolds. Over the course of the acquisition, some of these air bubbles move, shrink, or coalesce, thus resulting in artifacts in the form of incomplete bubble contours, as well as streaks aligned with parts of the moving scaffolds and bubbles.

These artifacts are visible in Fig. 11 showing the reconstructions obtained with the static CP algorithm, a Fourier-based dynamic reconstruction [44], and the PLI interpolation with $M = 2$ and $M = 4$. Without time interpolation, artificial streaks deteriorate the attenuation field within the water, which would otherwise be homogeneous. The vertical alignment of these streaks suggests a continuous drift of the solid, similar to what is shown in Fig. 3(a). With the dynamic PLI algorithm and $M = 2$, these streaks are largely eliminated, especially far away from the moving scaffold or bubbles. The bubble contours are more sharply defined, although some artifacts are still present. However, switching from $M = 2$ to $M = 4$ with PLI barely changes the results. This trend is consistent with the numerical results from Fig. 3(a), and is explained by the fact that a uniform motion is equally modeled with $M = 2$ and $M = 4$. Varying the breakpoints with $M = 4$ did not visually improve the results with this dataset. With the Fourier-based reconstruction, the vertical streaks are slightly less attenuated than with PLI. However, the Fourier-based reconstruction eliminates small artifacts near the scaffold struts that are not very affected by PLI. It is not clear what these artifacts are due to, and overall qualitative comparisons suggest that both methods arguably perform similarly on this dataset. Note that compared to the phantom analysis

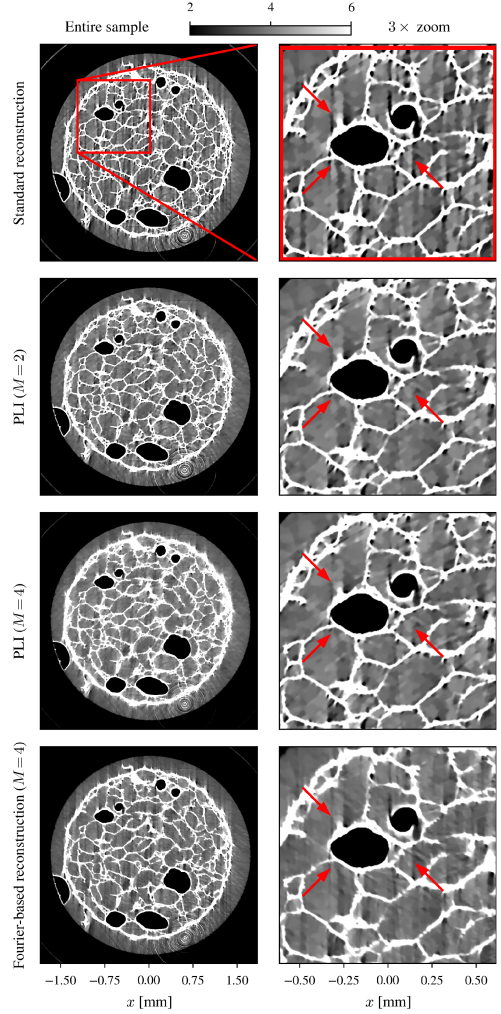


Fig. 11. Algorithm comparisons on bone scaffold data [77] for 10^4 iterations with $\lambda = 2^1$ and $\mu = 2^{-4}$. For the case $M = 4$, the breakpoints are equidistant. The colorbar indicates the linear attenuation in cm^{-1} . For the Fourier-based reconstruction [44]: $\lambda = 2^1$ and $\mu = 2^{-3}$.

in which similar regularization parameters λ and ν were used, different values are necessary with these larger datasets in order to retrieve comparable levels of regularization in the images, which is attributed to the use of two different implementations.

B. Application to Gas Measurements Inside a Ceramic Foam

The proposed method is applied to a second set of data from in situ experiments performed at the 8.3.2 beamline of the Advanced Light Source (ALS) at the Lawrence Berkeley National Laboratory (LBNL). These measurements investigate the topology of flames stabilized inside open-cell foams of silicon carbide. The quantity of interest is the X-ray attenuation of the gas-phase within the solid matrix pores, which is augmented using Krypton, a high-attenuation inert gas. The measured attenuation signal in the pores is proportional to the gas-phase density ρ , which is inversely related to the temperature T according to the ideal gas law $T \propto 1/\rho \propto 1/f$. The temperature can therefore be retrieved from the attenuation f , even

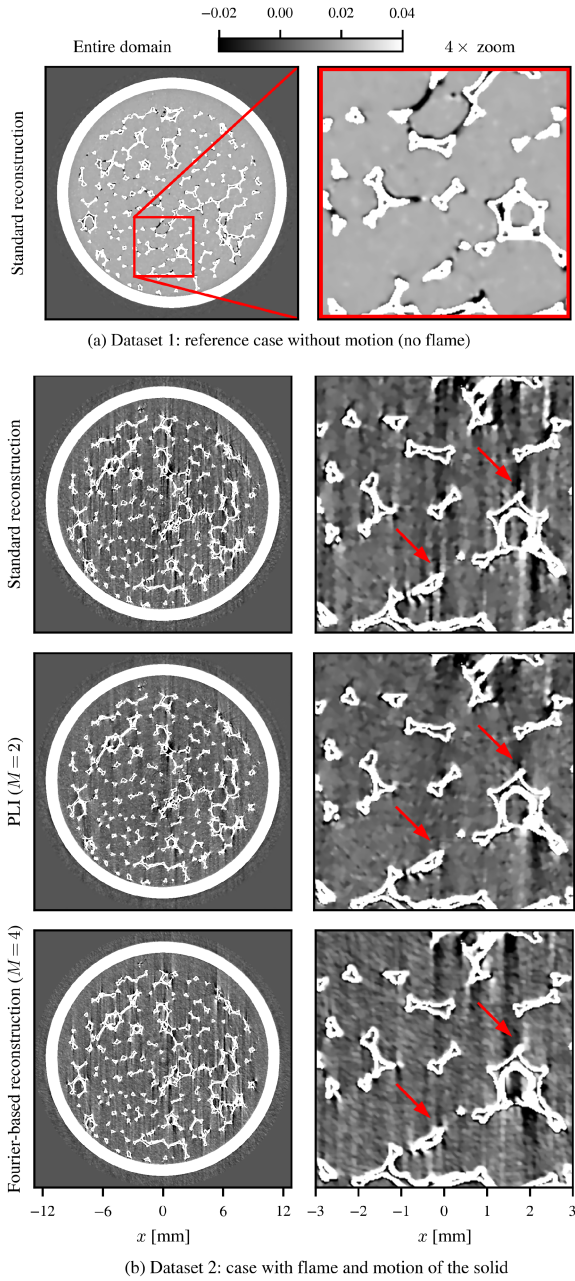


Fig. 12. Algorithm comparisons for the interstitial flame datasets acquired at the ALS. Reconstructions are performed using 10^4 iterations with $\lambda = 2^{-6}$ and $\mu = 2^{-5}$. The colorbar indicates the linear attenuation in cm^{-1} . For the Fourier-based reconstruction [44]: $\lambda = 2^{-6}$ and $\mu = 2^{-3}$.

though the signal remains of relatively low magnitude. These experiments therefore require high-accuracy quantitative measurements to characterize the flame structure within the pores of ceramic foams [79], [80]. However, under the thermal load of the flame, the foam expands and slowly drifts, thereby resulting in motion artifacts that compromise the gas-phase temperature measurements. In addition, sudden motion of the foam can occur because of strut buckling and cracking during the acquisition. The presence of both continuous and sudden motions in these experiments motivated the development of the proposed flexible PLI formulation.

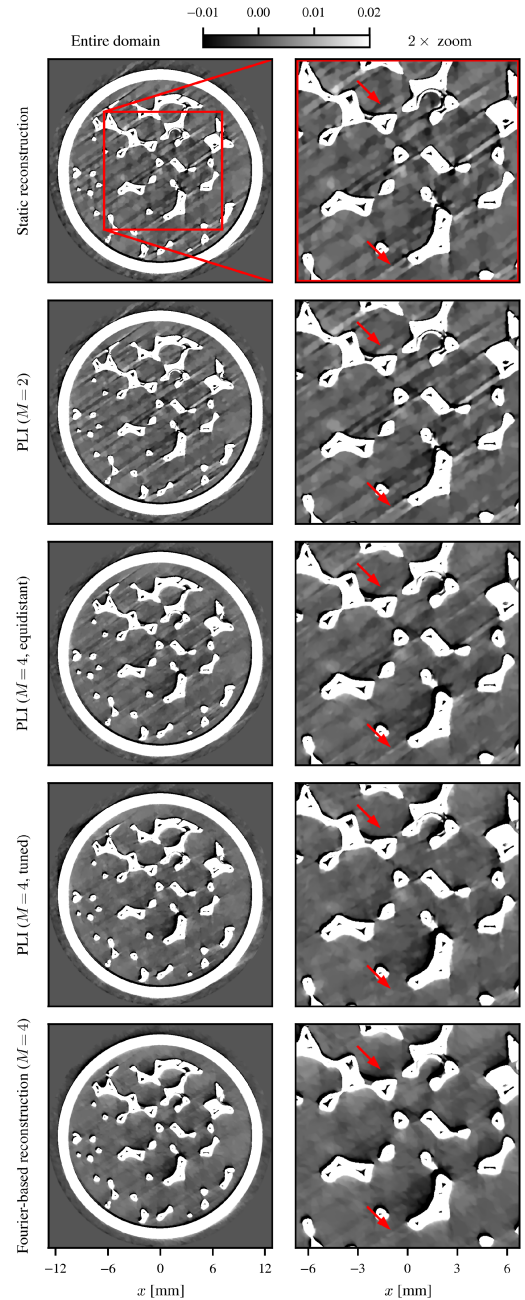


Fig. 13. Algorithm comparisons for the third interstitial flame dataset acquired at the ALS. Reconstructions are performed using 10^4 iterations with $\lambda = 2^{-4}$ and $\mu = 2^{-5}$. For $M = 4$, the breakpoints are either equidistant $[0^\circ, 60^\circ, 120^\circ, 180^\circ]$ or manually tuned $[0^\circ, 124^\circ, 125^\circ, 180^\circ]$. The colorbar indicates the linear attenuation in cm^{-1} . For the Fourier-based reconstruction [44]: $\lambda = 2^{-4}$ and $\mu = 2^{-3}$.

Figs. 12 and 13 compare the artifact reduction obtained with Fourier-based reconstruction [44] and the proposed PLI method. Three datasets are reported, in which a methane-air mixture containing 50% Kr per volume is flown through different ceramic foams placed inside a quartz tube. Each dataset consists of 2101 images acquired at a rate of 25 fps during a 180° rotation of the sample.

The first dataset, shown in Fig. 12(a) is a reference dataset with a non-ignited mixture. In this case, no motion artifacts are

visible, as there is no flame within the static foam. The constant attenuation value of 0.021 cm^{-1} in the gas-phase corresponds to the Kr-attenuation at a temperature of 300 K.

Fig. 12(b) shows results from a second CT scan acquired while the mixture is ignited and a flame is stabilized inside the ceramic foam. The gas-phase attenuation is significantly lower because of the higher gas temperature T as per $f \propto 1/T$, and motion artifacts due to the moving solid foam are observed. Similar to the scaffold data shown in Fig. 11, the vertical alignment of the streaks suggests a continuous solid motion. Comparing successive CT scans indicates that the amplitude of this motion is lower than one voxel ($10 \text{ }\mu\text{m}$) per scan, which is however sufficient to compromise temperature measurements.

Whereas with the bone scaffold dataset, PLI and Fourier interpolation performed similarly, the results with this ALS dataset show that a significantly higher artifact reduction is achieved with PLI. Specifically, the impact of the artifacts on the gas-phase measurements in the middle of the pores is significantly reduced with PLI. Analogous to the results shown in Fig. 11, the PLI reconstructions with $M = 4$ and equidistant breakpoints barely differ from the case $M = 2$, and are thus not shown. These results confirm that PLI with $M = 2$ is particularly effective and robust at eliminating streaks that are aligned with the direction of the first angle of projection.

The third dataset shown in Fig. 13 features major streaks aligned at a specific angle, resembling the artifacts from Fig. 3(b) for a sudden motion. As supported by analytical results [70], this streak alignment indicates that the solid suddenly moved by less than a voxel when the angle 125° was acquired. The artifact reduction achieved using PLI for $M = 2$ and $M = 4$ with equidistant breakpoints remains limited. However, by setting the PLI breakpoints right before and after the sample motion, the artifacts are largely reduced. Specifically, in coherence with the breakpoint analysis of Fig. 7, the artifact reduction is barely sensitive to the breakpoint separation, whereas it is very dependent on the mean value between the two inner breakpoints, which is tuned by visually comparing results obtained with 1° increment. The data-informed breakpoint selection demonstrated with numerical phantom in Fig. 3(b) can thus also work with real datasets.

The Fourier interpolation performs similarly to PLI in this third dataset, although certain streaks that are still observed in the Fourier results are largely reduced with PLI (upper arrow in Fig. 13). Overall, these comparisons show that PLI performs better on the second dataset but similarly on the third dataset.

VI. CONCLUSIONS AND DISCUSSION

A new approach for reducing motion artifacts using Piecewise Linear Interpolation (PLI) in time is proposed, validated with numerical phantoms, and demonstrated on datasets from two synchrotron experiments. The method relies on decomposing the solution in time using M breakpoints to model the time variations of each voxel and reduce motion artifacts. The implementation leverages the compactness of the PLI functions to reduce the algorithmic cost, and the code used to produce results from this article is shared online (<https://github.com/eboigne/>

PyRAMID-CT). A second-order accurate discretization of the TV regularization is also proposed to reduce staircase artifacts, representing a potential alternative to TV implementations using higher order derivatives [2], [57].

PLI appears particularly natural for sub-pixel motion as variations in voxel values are piecewise-linear during uniform motion because of the partial volume effect. With ideal phantoms, our analysis shows that by optimizing the PLI breakpoints, the method can eliminate artifacts created from a variety of motions, outperforming a Fourier interpolation approach from the literature [44]. The breakpoint optimization is done *a posteriori* on a per-case basis, which may be limiting for certain applications, but illustrates the extent of artifact reduction that could be achieved by jointly optimizing for the interpolation basis. Interestingly, even when insufficient breakpoints are used to exactly model the evolution of all voxel values, the artifacts can still be greatly attenuated and localized near the moving objects.

The method also achieves large artifact reduction in synchrotron data, notably enabling quantitative gas-phase experiments compromised by sub-pixel motion. In these experiments, the performance achieved is slightly lower than with ideal phantoms, but comparable to what is obtained with non-ideal phantoms modeling noise and irregularities in the motion. In contrast, when applied to the less quantitative measurements of a bone scaffold dataset featuring larger motion, the PLI method performs similarly to Fourier interpolation. PLI therefore appears particularly appropriate for high-accuracy quantitative CT applications that are compromised by small motions. For such high-accuracy applications, our convergence study indicates that a large number of iterations is required, whereas 500 to 1000 iterations will likely be sufficient for less quantitative applications.

In addition, our analysis of a variety of motion profiles highlights the enhanced performance achieved by considering a flexible interpolation instead of a fixed one [44]. The parametric flexibility of the proposed PLI formulation is a first step in this direction, demonstrating that data-informed joint optimization of the interpolation could work in CT, similarly to that proposed in dynamic PET [51], [52]. In the future, the choice of interpolation could be guided by data-driven approaches, such as the identification of characteristic artifacts by convolutional neural networks [32], [81]. The time decomposition could also be varied spatially using coarse local parametrizations, to model more complex motions than with the global voxel-independent breakpoints considered in the present study.

Finally, while most artifact-reduction methods are evaluated on time-series of multiple CT rotations, the proposed PLI method is demonstrated to work on single CT datasets. This result may be particularly relevant to synchrotron CT with limited beamtime access prohibiting extensive dynamic studies, and clinical CT for which minimal X-ray exposure is critical. In clinical CT, PLI is likely most promising for head and breast CT or gated chest CT with limited motion, whereas a Fourier time-interpolation would be more suited for non-gated chest CT with periodic motion. In particular, even though time-decomposition approaches remain expensive compared to direct reconstruction algorithms (FBP, FDK), their cost scales linearly with M and

they are only a few times slower than iterative methods (SIRT, CGLS, MLEM), and may thus soon be relevant to clinical CT.

ACKNOWLEDGMENT

The authors would like to thank Alastair MacDowell, Harold Barnard, Douglas Taube, and Lee Yang from LBNL for their help performing measurements at the ALS synchrotron.

REFERENCES

- [1] E. Maire and P. J. Withers, "Quantitative X-ray tomography," *Int. Mater. Rev.*, vol. 59, no. 1, pp. 1–43, 2014.
- [2] J. A. Fessler, "Image reconstruction: Algorithms and analysis," Accessed: Feb. 02, 2022. [Online]. Available: <https://web.eecs.umich.edu/fessler/book>
- [3] X. Liu et al., "Four dimensional visualization of highly transient fuel sprays by microsecond quantitative X-ray tomography," *Appl. Phys. Lett.*, vol. 94, 2009, Art. no. 084101.
- [4] H. A. Bale et al., "Real-time quantitative imaging of failure events in materials under load at temperatures above 1,600°C," *Nature Mater.*, vol. 12, no. 1, pp. 40–46, 2012.
- [5] K. J. Harry, D. T. Hallinan, D. Y. Parkinson, A. A. MacDowell, and N. P. Balsara, "Detection of subsurface structures underneath dendrites formed on cycled lithium metal electrodes," *Nature Mater.*, vol. 13, no. 1, pp. 69–73, 2014.
- [6] A. Tekawade, B. A. Sforzo, K. E. Matusik, K. Fezzaa, A. L. Kastengren, and C. F. Powell, "Time-resolved 3D imaging of two-phase fluid flow inside a steel fuel injector using synchrotron X-ray tomography," *Sci. Rep.*, vol. 10, 2020, Art. no. 8674.
- [7] E. Boigné, N. R. Bennett, A. Wang, and M. Ihme, "Structural analysis of biomass pyrolysis and oxidation using in-situ X-ray computed tomography," *Combustion Flame*, vol. 235, 2022, Art. no. 111737.
- [8] W. P. Segars, D. S. Lalush, and M. W. T. Benjamin, "A realistic spline-based dynamic heart phantom," *IEEE Trans. Nucl. Sci.*, vol. 46, no. 3, pp. 503–506, Jun. 1999.
- [9] L. Ritschl, S. Sawall, M. Knaup, A. Hess, and M. Kachelrie, "Iterative 4D cardiac micro-CT image reconstruction using an adaptive spatio-temporal sparsity prior," *Phys. Med. Biol.*, vol. 57, no. 6, pp. 1517–1525, 2012.
- [10] C. Rohkohl, H. Bruder, K. Stierstorfer, and T. Flohr, "Improving best-phase image quality in cardiac CT by motion correction with MAM optimization," *Med. Phys.*, vol. 40, 2013, Art. no. 031901.
- [11] R. Mokso, F. Marone, and M. Stampanoni, "Real time tomography at the Swiss Light Source," in *Proc. AIP Conf.*, 2010, vol. 1234, pp. 87–90.
- [12] S. Moser, S. Nau, M. Salk, and K. Thoma, "In situ flash X-ray high-speed computed tomography for the quantitative analysis of highly dynamic processes," *Meas. Sci. Technol.*, vol. 25, 2014, Art. no. 025009.
- [13] M. Wallenstein et al., "Qualitative and quantitative insights into multiphase flow in ceramic sponges using X-ray computed tomography," *Chem. Eng. Sci.*, vol. 138, pp. 118–127, 2015.
- [14] S. Yoon, S. A. Mäkiharju, J. A. Fessler, and S. L. Ceccio, "Image reconstruction for limited-angle electron beam X-ray computed tomography with energy-integrating detectors for multiphase flows," *IEEE Trans. Comput. Imag.*, vol. 4, no. 1, pp. 112–124, Mar. 2018.
- [15] C. Raufaste, B. Dollet, K. Mader, S. Santucci, and R. Mokso, "Three-dimensional foam flow resolved by fast X-ray tomographic microscopy," *Europhys. Lett.*, vol. 111, 2015, Art. no. 38004.
- [16] A. Ruhlandt, M. Töpperwien, M. Krenkel, R. Mokso, and T. Salditt, "Four dimensional material movies: High speed phase-contrast tomography by backprojection along dynamically curved paths," *Sci. Rep.*, vol. 7, 2017, Art. no. 6487.
- [17] C. Badea, L. W. Hedlund, and G. A. Johnson, "Micro-CT with respiratory and cardiac gating," *Med. Phys.*, vol. 31, no. 12, pp. 3324–3329, 2004.
- [18] M. N. D'eurydice, C. H. Arns, J. Y. Arns, and R. T. Armstrong, "Dynamic imaging of multiphase flow through porous media using 4D cumulative reconstruction," *J. Microsc.*, vol. 272, no. 1, pp. 12–24, 2018.
- [19] B. Münch, "Spatiotemporal computed tomography of dynamic processes," *Opt. Eng.*, vol. 50, 2011, Art. no. 123201.
- [20] D. Gürsoy et al., "Rapid alignment of nanotomography data using joint iterative reconstruction and reprojection," *Sci. Rep.*, vol. 7, 2017, Art. no. 11818.
- [21] B. Feng et al., "Estimation of the rigid-body motion from three-dimensional images using a generalized center-of-mass points approach," *IEEE Trans. Nucl. Sci.*, vol. 53, no. 5, pp. 2712–2718, Oct. 2006.
- [22] T. Sun, J.-H. Kim, R. Fulton, and J. Nuyts, "An iterative projection-based motion estimation and compensation scheme for head X-ray CT," *Med. Phys.*, vol. 43, no. 10, pp. 5705–5716, 2016.
- [23] S. Jang, S. Kim, M. Kim, and J. B. Ra, "Head motion correction based on filtered backprojection for X-ray CT imaging," *Med. Phys.*, vol. 45, no. 2, pp. 589–604, 2018.
- [24] W. Lu and T. R. Mackie, "Tomographic motion detection and correction directly in Sinogram space," *Phys. Med. Biol.*, vol. 47, no. 8, pp. 1267–1284, 2002.
- [25] R. Mooser, F. Forsberg, E. Hack, G. Székely, and U. Sennhauser, "Estimation of affine transformations directly from tomographic projections in two and three dimensions," *Mach. Vis. Appl.*, vol. 24, no. 2, pp. 419–434, 2013.
- [26] V. Van Nieuwenhove, J. De Beenhouwer, J. Vlassenbroeck, M. Brennan, and J. Sijbers, "MoVIT: A tomographic reconstruction framework for 4D-CT," *Opt. Exp.*, vol. 25, 2017, Art. no. 19236.
- [27] T. De Schryver et al., "Motion compensated micro-CT reconstruction for in-situ analysis of dynamic processes," *Sci. Rep.*, vol. 8, 2018, Art. no. 7655.
- [28] H. Dang, A. S. Wang, M. S. Sussman, J. H. Siewerdsen, and J. W. Stayman, "DPIRPLE: A joint estimation framework for deformable registration and penalized-likelihood CT image reconstruction using prior images," *Phys. Med. Biol.*, vol. 59, no. 17, pp. 4799–4826, 2014.
- [29] A. Biguri, M. Dosanjh, S. Hancock, and M. Soleimani, "A general method for motion compensation in X-ray computed tomography," *Phys. Med. Biol.*, vol. 62, no. 16, pp. 6532–6549, 2017.
- [30] V. Van Nieuwenhove, J. De Beenhouwer, T. De Schryver, L. Van Hoorebeke, and J. Sijbers, "Data-driven affine deformation estimation and correction in cone beam computed tomography," *IEEE Trans. Image Process.*, vol. 26, no. 3, pp. 1441–1451, Mar. 2017.
- [31] X. Huang, Y. Zhang, L. Chen, and J. Wang, "U-Net-based deformation vector field estimation for motion-compensated 4D-CBCT reconstruction," *Med. Phys.*, vol. 47, no. 7, pp. 3000–3012, 2020.
- [32] M. Bührer et al., "Deep learning based classification of dynamic processes in time-resolved X-ray tomographic microscopy," *Sci. Rep.*, vol. 11, 2021, Art. no. 24174.
- [33] G. R. Myers, M. Geleta, A. M. Kingston, B. Recur, and A. P. Sheppard, "Improving dynamic tomography, through maximum a posteriori estimation," in *Proc. SPIE Develop. X-Ray Tomogr. IX*, 2014, vol. 9212, pp. 270–278.
- [34] S. Y. Chun and J. A. Fessler, "Joint image reconstruction and nonrigid motion estimation with a simple penalty that encourages local invertibility," *Proc. SPIE*, vol. 7258, 2009, Art. no. 72580U.
- [35] X. Jia, Y. Lou, R. Li, W. Y. Song, and S. B. Jiang, "GPU-based fast cone beam CT reconstruction from undersampled and noisy projection data via total variation," *Med. Phys.*, vol. 37, no. 4, pp. 1757–1760, 2010.
- [36] Z. Tian, X. Jia, K. Yuan, T. Pan, and S. B. Jiang, "Low-dose CT reconstruction via edge-preserving total variation regularization," *Phys. Med. Biol.*, vol. 56, no. 18, pp. 5949–5967, 2011.
- [37] G.-H. Chen, J. Tang, and S. Leng, "Prior image constrained compressed sensing (PICCS): A method to accurately reconstruct dynamic CT images from highly undersampled projection data sets," *Med. Phys.*, vol. 35, no. 2, pp. 660–663, 2008.
- [38] H. Wu, A. Maier, R. Fahrigr, and J. Hornegger, "Spatial-temporal total variation regularization (STTVR) for 4D-CT reconstruction," *Proc. SPIE*, vol. 8313, 2012, Art. no. 83133J.
- [39] D. Kazantsev et al., "4D-CT reconstruction with unified spatial-temporal patch-based regularization," *Inverse Problems Imag.*, vol. 9, no. 2, pp. 447–467, 2015.
- [40] G. Van Eyndhoven et al., "An iterative CT reconstruction algorithm for fast fluid flow imaging," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4446–4458, Nov. 2015.
- [41] M. Bührer, H. Xu, J. Eller, J. Sijbers, M. Stampanoni, and F. Marone, "Unveiling water dynamics in fuel cells from time-resolved tomographic microscopy data," *Sci. Rep.*, vol. 10, 2020, Art. no. 16388.
- [42] K. Aditya Mohan et al., "TIMBIR: A method for time-space reconstruction from interlaced views," *IEEE Trans. Comput. Imag.*, vol. 1, no. 2, pp. 96–111, Jun. 2015.
- [43] Z. Yu, J.-B. Thibault, J. Wang, C. A. Bouman, and K. D. Sauer, "Kinetic model for motion compensation in computed tomography," in *Proc. 11th Int. Meeting Fully Three-Dimensional Image Reconstruction Radiol. Nucl. Med.*, 2011, pp. 466–469.

- [44] V. V. Nikitin, M. Carlsson, F. Andersson, and R. Mokso, "Four-dimensional tomographic reconstruction by time domain decomposition," *IEEE Trans. Comput. Imag.*, vol. 5, no. 3, pp. 409–419, Sep. 2019.
- [45] A. Rahmim, J. Tang, and H. Zaidi, "Four-dimensional (4D) image reconstruction strategies in dynamic PET: Beyond conventional independent frame reconstruction," *Med. Phys.*, vol. 36, no. 8, pp. 3654–3670, 2009.
- [46] S. R. Meikle et al., "Parametric image reconstruction using spectral analysis of PET projection data," *Phys. Med. Biol.*, vol. 43, no. 3, pp. 651–666, 1998.
- [47] T. Nichols, J. Qi, E. Asma, and R. Leahy, "Spatiotemporal reconstruction of list-mode PET data," *IEEE Trans. Med. Imag.*, vol. 21, no. 4, pp. 396–404, Apr. 2002.
- [48] Q. Li, E. Asma, S. Ahn, and R. M. Leahy, "A fast fully 4-D incremental gradient reconstruction algorithm for list mode PET data," *IEEE Trans. Med. Imag.*, vol. 26, no. 1, pp. 58–67, Jan. 2007.
- [49] J. Verhaeghe, Y. D'Asseler, S. Vandenbergh, S. Staelens, and I. Lemahieu, "An investigation of temporal regularization techniques for dynamic PET reconstructions using temporal splines," *Med. Phys.*, vol. 34, no. 5, pp. 1766–1778, 2007.
- [50] J. Matthews, D. Bailey, P. Price, and V. Cunningham, "The direct calculation of parametric images from dynamic PET data using maximum-likelihood iterative reconstruction," *Phys. Med. Biol.*, vol. 42, no. 6, pp. 1155–1173, 1997.
- [51] A. J. Reader, F. C. Sureau, C. Comtat, R. Trébossen, and I. Buvat, "Joint estimation of dynamic PET images and temporal basis functions using fully 4D ML-EM," *Phys. Med. Biol.*, vol. 51, no. 21, pp. 5455–5474, 2006.
- [52] A. J. Reader, F. Sureau, C. Comtat, R. Trébossen, and I. Buvat, "Simultaneous estimation of temporal basis functions and fully 4D PET images," in *Proc. IEEE Nucl. Sci. Symp. Conf. Rec.*, 2006, vol. 4, pp. 2219–2223.
- [53] Y. Ko, S. Moon, J. Baek, and H. Shim, "Rigid and non-rigid motion artifact reduction in X-ray CT using attention module," *Med. Image Anal.*, vol. 67, 2021, Art. no. 101883.
- [54] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *J. Math. Imag. Vis.*, vol. 40, no. 1, pp. 120–145, 2011.
- [55] S. Y. Chun and J. A. Fessler, "Spatial resolution properties of motion-compensated tomographic image reconstruction methods," *IEEE Trans. Med. Imag.*, vol. 31, no. 7, pp. 1413–1425, Jul. 2012.
- [56] M. Heyndrickx et al., "Piecewise linear fitting in dynamic micro-CT," *Mater. Characterization*, vol. 139, pp. 259–268, 2018.
- [57] Y. Hu and M. Jacob, "Higher degree total variation (HDTV) regularization for image recovery," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2559–2571, May 2012.
- [58] B. T. Polyak, *Introduction to Optimization*. New York, NY, USA: Optim. Softw., Inc., 1987.
- [59] J. Gregor and T. Benson, "Computational analysis and improvement of SIRT," *IEEE Trans. Med. Imag.*, vol. 27, no. 7, pp. 918–924, Jul. 2008.
- [60] T. Pock and A. Chambolle, "Diagonal preconditioning for first order primal-dual algorithms in convex optimization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 1762–1769.
- [61] D. Gürsoy, F. De Carlo, X. Xiao, and C. Jacobsen, "TomoPy: A framework for the analysis of synchrotron tomographic data," *J. Synchrotron Radiat.*, vol. 21, no. 5, pp. 1188–1193, 2014.
- [62] A. Biguri, M. Dosanjh, S. Hancock, and M. Soleimani, "TIGRE: A MATLAB-GPU toolbox for CBCT image reconstruction," *Biomed. Phys. Eng. Exp.*, vol. 2, 2016, Art. no. 055010.
- [63] N. Wadson and M. Basham, "Savu: A python-based, MPI framework for simultaneous processing of multiple, N-dimensional, large tomography datasets," 2016, *arXiv:1610.08015*.
- [64] J. S. Jørgensen et al., "Core Imaging Library—Part I: A versatile Python framework for tomographic imaging," *Philos. Trans. Roy. Soc. London Ser. A*, vol. 379, no. 2204, 2021, Art. no. 20200192.
- [65] J. Adler et al., "Odlgroup/odl: Odl 0.7.0," 2018. [Online]. Available: <https://doi.org/10.5281/zenodo.592765>
- [66] W. van Aarle et al., "Fast and flexible X-ray tomography using the ASTRA toolbox," *Opt. Exp.*, vol. 24, no. 22, pp. 25129–25147, 2016.
- [67] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, vol. 32, pp. 8026–8037.
- [68] V. Nikitin et al., "Distributed optimization for nonrigid nano-tomography," *IEEE Trans. Comput. Imag.*, vol. 7, pp. 272–287, 2021.
- [69] T. Sakamoto and S. Sato, "Image degradation and stroboscopic images caused by rotary motion of the object in X-ray computed tomography," *Syst. Comput. Jpn.*, vol. 24, no. 9, pp. 76–83, 1993.
- [70] C. K. Yang, S. C. Orphanoudakis, and J. W. Strohbehn, "A simulation study of motion artefacts in computed tomography," *Phys. Med. Biol.*, vol. 27, no. 1, pp. 51–61, 1982.
- [71] D. Kazantsev, V. Pickalov, S. Nagella, E. Pasca, and P. J. Withers, "Tomophantom. A software package to generate 2D–4D analytical phantoms for CT image reconstruction algorithm benchmarks," *SoftwareX*, vol. 7, pp. 150–155, 2018.
- [72] J. Kaipio and E. Somersalo, "Statistical inverse problems: Discretization, model reduction and inverse crimes," *J. Comput. Appl. Math.*, vol. 198, no. 2, pp. 493–504, 2007.
- [73] K. Bredies, K. Kunisch, and T. Pock, "Total generalized variation," *SIAM J. Imag. Sci.*, vol. 3, no. 3, pp. 492–526, 2010.
- [74] Z. Yang and M. Jacob, "Robust non-local regularization framework for motion compensated dynamic imaging without explicit motion estimation," in *Proc. IEEE 9th Int. Symp. Biomed. Imag.*, 2012, pp. 1056–1059.
- [75] J. Fessler, "Analytical approach to regularization design for isotropic spatial resolution," in *Proc. IEEE Nucl. Sci. Symp. Med. Imag. Conf.*, 2003, vol. 3, pp. 2022–2026.
- [76] C.-C. Cheng, Y.-T. Ching, P.-H. Ko, and Y. Hwu, "Correction of center of rotation and projection angle in synchrotron X-ray computed tomography," *Sci. Rep.*, vol. 8, 2018, Art. no. 9884.
- [77] F. Brun, G. Turco, S. Paoletti, and A. Accardo, "A synchrotron radiation microtomography study of wettability and swelling of nanocomposite alginate/hydroxyapatite scaffolds for bone tissue engineering," in *Proc. World Congr. Med. Phys. Biomed. Eng.*, 2015, vol. 51, pp. 288–291.
- [78] F. De Carlo et al., "TomoBank: A tomographic data repository for computational X-ray science," *Meas. Sci. Technol.*, vol. 29, 2018, Art. no. 034004.
- [79] E. Boigné et al., "X-ray computed tomography for flame-structure analysis of laminar premixed flames," *Combustion Flame*, vol. 200, pp. 142–154, 2019.
- [80] J. Dunnmon, S. Sobhani, M. Wu, R. Fahrig, and M. Ihme, "An investigation of internal flame structure in porous media combustion via X-ray computed tomography," *Proc. Combustion Inst.*, vol. 36, no. 3, pp. 4399–4408, 2017.
- [81] M. U. Ghani and W. C. Karl, "Fast enhanced CT metal artifact reduction using data domain deep learning," *IEEE Trans. Comput. Imag.*, vol. 6, pp. 181–193, 2020.