

Inverse Surfacelet Transform for Image Reconstruction With Constrained-Conjugate Gradient Methods

Wei Huang

School of Mechanical Engineering,
Georgia Institute of Technology,
Atlanta, GA 30332
e-mail: whuang47@gatech.edu

Yan Wang¹

School of Mechanical Engineering,
Georgia Institute of Technology,
Atlanta, GA 30332
e-mail: yan.wang@me.gatech.edu

David W. Rosen

School of Mechanical Engineering,
Georgia Institute of Technology,
Atlanta, GA 30332
e-mail: david.rosen@me.gatech.edu

Image reconstruction is the transformation process from a reduced-order representation to the original image pixel form. In materials characterization, it can be utilized as a method to retrieve material composition information. In our previous work, a surfacelet transform was developed to efficiently represent boundary information in material images with surfacelet coefficients. In this paper, new constrained-conjugate-gradient based image reconstruction methods are proposed as the inverse surfacelet transform. With geometric constraints on boundaries and internal distributions of materials, the proposed methods are able to reconstruct material images from surfacelet coefficients as either lossy or lossless compressions. The results between the proposed and other optimization methods for solving the least-square error inverse problems are compared.

[DOI: 10.1115/1.4026376]

Keywords: materials modeling, feature identification, image reconstruction, implicit surface, wavelet, constrained-conjugate-gradient

1 Introduction

During image transformation, by converting the original image pixel values to a new data representation in a different domain (e.g., the reciprocal space), we can utilize the new representation for different purposes such as retrieving meta-level information and image compression. The reverse process is to reconstruct images from other representations.

The transformation can be either physical or non-physical processes. A well-known physical transformation process is tomography. In tomography, a form of ray is radiated from the source and passes through the scanned object in straight lines. A sensor collects the data in a form of physical quantity, then the data are used to reconstruct the image. This reconstruction is usually conducted through the Radon transform [1]. Image transformation can also be in the form of a non-physical process, such as image analysis for pattern recognition and material characterization. The purpose of these applications is usually to find a certain class of geometric features from images via methods such as the Hough transform [2,3].

As a generalization of the Radon transform from 2D to 3D with the additional feature identification capability similar to the Hough transform, a surfacelet transform [4] converts 3D image pixels or voxels into surfacelet coefficients, which are the integrals of voxel values over some small surfaces corresponding to the external or internal singularities or boundaries of interest, followed by a 1D wavelet transform, as shown in Fig. 1.

When surfacelet transform is applied in materials characterization, it is also desirable that the surfacelet coefficients can be converted back into voxel values. As also shown in Fig. 1, the inverse 1D wavelet transform can easily convert the surfacelet coefficients back to surface integrals. However, recovering the image voxel values from the surface integrals is not trivial. Although it is not necessary to completely retrieve the original images in a lossy compression scenario, it is required that the information of impor-

tant geometric and material features is not lost. In this paper, we propose an inverse surfacelet transform method so that the surfacelet transform formalism is complete. The inverse transform is accomplished by solving least-square error problems. Given enough surfacelet coefficients, the original images can be fully reconstructed losslessly. Furthermore, important geometric features that have been recognized are preserved in the inverse transform by applying constraints on the associated voxels in lossy compressions.

When the original image data cannot be fully recovered for lossy compressions during the reconstruction, constraints are usually added to improve the solution of this under-constrained problem. Existing approaches usually set up constraints simply as the non-negativity of pixel values for the purpose of object recognition. No attempt has been made on geometric constraints which are important in our applications in materials characterization. Here the proposed method of inverse surfacelet transform with geometric constraints from prior knowledge of materials is able to reconstruct images from much less surfacelet data than the original image pixels. The pair of forward and inverse surfacelet transforms provide a feature-based image compression method. Because this image compression method largely relies on the feature identification with existing surfacelets, images of various geometric features of interest that match the existing surfacelets are suitable for this approach. Compared to the traditional image compression methods (e.g., Refs. [5] and [6]), the surfacelet transform loses less information on the geometric features and preserves the sharpness of boundaries, similar to the directional wavelet methods that are reviewed in Sec. 2.3. Therefore, it is more suitable for processing images of structure and materials compositions than other generic image processing methods such as wavelets and Fourier-based approaches.

The overview of the complete process of surfacelet transform and inverse surfacelet transform is shown in Fig. 1. In the surfacelet transform, surface integrals are obtained from image pixels. 1D wavelet transform is then applied to obtain wavelet coefficients. In the inverse surfacelet transform, the inverse 1D wavelet transform easily retrieves surface integrals from wavelet coefficients. The scope of this paper is denoted by the dashed box in Fig. 1. The inverse problem of retrieving image pixel values

¹Corresponding author.

Contributed by the Computers and Information Division of ASME for publication in the JOURNAL OF COMPUTING AND INFORMATION SCIENCE IN ENGINEERING. Manuscript received September 11, 2013; final manuscript received December 20, 2013; published online March 7, 2014. Assoc. Editor: Charlie C.L. Wang.

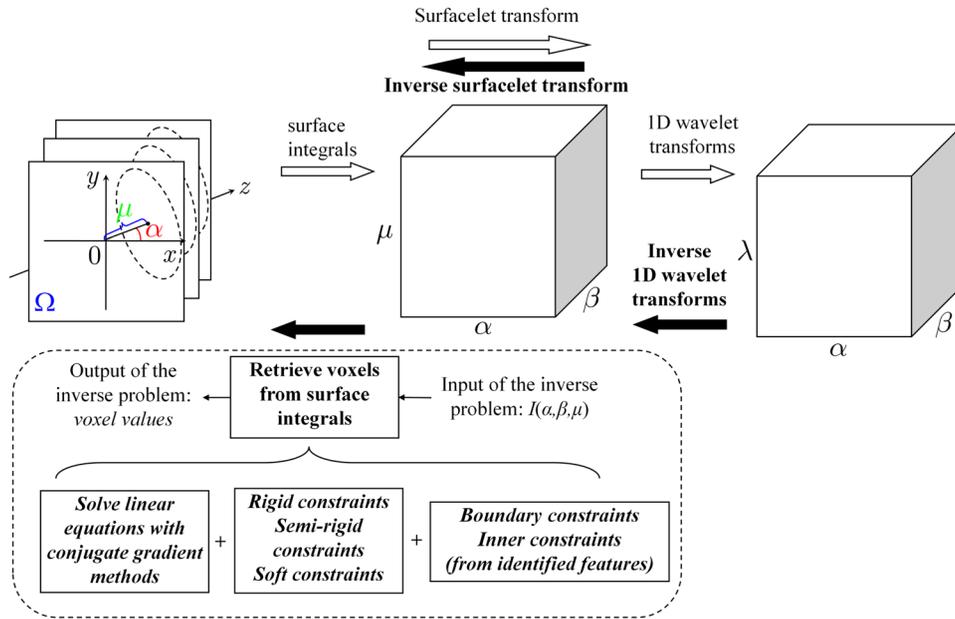


Fig. 1 The process of coupled surfacelet transform and inverse surfacelet transform

from surface integrals is solved based on three constrained conjugate-gradient-based methods with combinations of boundary constraints and inner constraints on internal distributions. The features that determine the boundary constraints are identified from the surfacelet transform process. The locations of internal pixels in the inner constraints can then be calculated from the features. Furthermore, different levels of rigidity associated with the constraints proposed in this paper provide more flexibility in controlling the constraints than a single-level approach.

In the remainder of the paper, we give a brief introduction of the surfacelet, surfacelet transform, their origins, and the closely related methods for image reconstruction in Sec. 2. In Sec. 3, three numerical methods for the inverse surfacelet transform are described. Examples are demonstrated and compared in Sec. 4.

2 Related Work

The surfacelet is the basis of the proposed methods in this paper. A surfacelet is a combination of a wavelet and an implicit surface. It is designed for multi-resolution and multi-scale materials modeling. The surfacelet transform was proposed such that images are transformed into surfacelet representations and geometric features in images can be identified. In this section, the relevant background information is provided. The surfacelet transform is a generalization of the Radon transform, which is introduced in Sec. 2.1. The proposed inverse surfacelet transform is based on the feature identification result of the surfacelet transform. Existing methods of image transformation for feature identification are summarized in Sec. 2.2. The surfacelet is also related to the so-called directional wavelet methods, which are introduced in Sec. 2.3. The surfacelet and the surfacelet transform are introduced in Sec. 2.4. In this paper, conjugate-gradient-based methods are used to solve the least-square error problem in the inverse surfacelet transform. The related conjugate-gradient-based iterative algorithms are also discussed in Sec. 2.5.

2.1 Radon Transform and Its Inverse in Tomography. Tomography is a technique to determine the internal structure and materials composition of an object using different imaging modalities, including X-ray, computed tomography, ultrasound, magnetic resonance, microwave, and others. By converting from linear integrals of sensor data to image pixels, this non-invasive imaging technique allows for the visualization of the internal

structures of an object. The Radon transform and inverse Radon transform [1] are the mathematical bases for reconstructing tomographic images from measured projection. The Radon transform is given by

$$p(r, \phi) = \int_{-\infty}^{\infty} f(r \cos \phi - s \sin \phi, r \sin \phi + s \cos \phi) ds$$

where x and y are the horizontal and vertical coordinates. $x \cos \phi + y \sin \phi - r = 0$ defines a projection line, where r is the shortest distance from the origin to the line, and ϕ is the angle formed by the distance vector. It can be rewritten as

$$p(r, \phi) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \delta(x \cos \phi + y \sin \phi - r) dx dy$$

where δ is the Dirac delta function.

The geometric interpretation of the Radon transform is the integral along the straight line projected through the scanned target. In parallel-beam tomography, r is varied so that a detector acquires parallel projections. The detector rotates circularly around the scanned object so that ϕ also varies. Then integrals over the whole $r - \phi$ domain can be obtained.

The task of tomographic reconstruction is to find $f(x, y)$ given $p(r, \phi)$. Therefore, this process is also called inverse Radon transform or backprojection. Mathematically, the inverse Radon transform is defined as

$$f(x, y) = \int_0^{\infty} \int_0^{\pi} p(r, \phi) \delta(x \cos \phi + y \sin \phi - r) d\phi dr$$

Since Radon obtained the inverse formula of Radon transform in 1917 [1], many tomographic reconstruction techniques have been proposed. The most famous one is the Direct Fourier reconstruction. In this method, the solution to the inverse Radon transform is based on the projection-slice theorem. The projection-slice theorem [7] is given by

$$P(v, \phi) = F(v \cos \phi, v \sin \phi)$$

where $F(v_x, v_y)$, with parameters v_x and v_y , is the 2D Fourier transform of $f(x, y)$. The theorem states that the 2D Fourier transform

of $f(x, y)$ along a direction corresponding to the inclination angle ϕ and an orthogonal one to the direction, is given by the 1D Fourier transform of the Radon transform result $p(r, \phi)$ with ϕ fixed as $P(v, \phi)$. Based on the projection-slice theorem, the inverse Radon transform can be realized by three steps: (1) 1D fast Fourier transform (FFT) along the projection direction to build a polar 2D Fourier space; (2) polar to Cartesian resampling; and (3) inverse 2D-FFT to obtain the reconstructed slices.

2.2 Image Transformation Methods for Feature Identification. Image transformation can be applied in feature identification. Other than tomography, the Radon transform is also applied in feature identification for materials characterization. For instance, it has been applied to identify lines in 2D images [8–10].

The Hough transform [2,3] was initially proposed to identify lines in images. Later it was extended by Ballard [11] to recognize arbitrarily complex shapes. Recently, the Hough transform was applied in the recognition of spherical features in 3D microstructural images [12].

2.3 Directional Wavelets: Wedgelet, Curvelet, and Surflet.

Wavelet or Fourier transform performs well for objects with point singularities of zero dimension. However, they are not effective in dealing with edge discontinuities of one dimension. Several approaches have been proposed to solve this issue, including wedgelet [13] and curvelet [14], as well as their close relatives such as ridgelet [15], contourlet [16], beamlet [17,18], and platelet [19].

The wedgelet approach [13] partitions 2D space into squares as building blocks bounded by line segments. 2D images then can be

approximated by a collection of specifically chosen wedgelets. The curvelet [14] is an extension of the standard wavelet function, which includes the concepts of statistical regression and a Radon transform. It was developed to compress images containing continuous line or curve segments, where the standard wavelets are not efficient. The basic idea is to introduce an angular element in the wavelet function. If wavelets can be thought of as “fat” points with certain widths of local support, curvelets are “fat” needles.

In 3D analysis, Ying et al. [20] extended the 2D curvelet transform to 3D with similar frequency space tilings. Similarly, Lu and Do [21] extended contourlets to three dimensions in a discrete space. Chandrasekaran et al. [22] extended wedgelets to high-dimensional space and approximate functions with polynomial building blocks, called surflets, instead of linear building blocks in wedgelets.

2.4 Surfacelet and Surfacelet Transform. Recently, a surfacelet model [4] was proposed to construct the geometric boundary and internal material distribution of heterogeneous materials. As combinations of wavelets and implicit surfaces, surfacelets keep both the multi-resolution/multi-scale nature of wavelets and the advantage of implicit surfaces in constructing complex 3D geometry. Three examples of surfacelets: 3D ridgelet, cylindrical surfacelet (or cylinderlet), and ellipsoidal surfacelet (or ellipsoidlet) are, respectively, defined as follows.

A 3D ridgelet that represents plane singularities is defined as

$$\psi_{a,b,\alpha,\beta}(\mathbf{r}) = a^{-1/2}\psi(a^{-1}(\cos\beta\cos\alpha \cdot x + \cos\beta\sin\alpha \cdot y + \sin\beta \cdot z - b))$$

A cylindrical surfacelet is defined as

$$\psi_{a,\mu,\alpha,\beta,r_1,r_2}(\mathbf{r}) = a^{-1/2}\psi\left(\begin{matrix} a^{-1}[r_1(\cos\beta\cos\alpha \cdot x + \cos\beta\sin\alpha \cdot y + \sin\beta \cdot z - \mu)^2 \\ + r_2(-\sin\alpha \cdot x + \cos\alpha \cdot y)^2] \end{matrix}\right) \tag{1}$$

And an ellipsoidal surfacelet is defined as

$$\psi_{a,b,\alpha,\beta,r_1,r_2,r_3}(\mathbf{r}) = a^{-1/2}\psi\left(\begin{matrix} r_1(\cos\beta\cos\alpha \cdot x + \cos\beta\sin\alpha \cdot y + \sin\beta \cdot z - b)^2 \\ + r_2(-\sin\alpha \cdot x + \cos\alpha \cdot y)^2 \\ + r_3(-\sin\beta\cos\alpha \cdot x + \sin\beta\sin\alpha \cdot y + \cos\beta \cdot z)^2 \end{matrix}\right)$$

where $\mathbf{r} = (x, y, z)$ is the location in the Euclidean space, ψ is a wavelet function, $r_1, r_2,$ and r_3 are shape parameters that are fixed during transformation, α and β are orientation parameters, a is the scale parameter, and μ is the position parameter. The geometric interpretations of the isosurfaces for the three surfacelets for plane, cylindrical, and ellipsoidal singularities are shown in Fig. 2.

The surfacelet transform constructs the surfacelet model from 3D materials images. Using a cylindrical surfacelet as the example, the surface integrals on surfacelets are calculated and arranged in a 3D matrix with $\alpha, \beta,$ and μ as indices, as illustrated in Fig. 1. Then a 1D wavelet transform along the μ axis direction is performed for all α 's and β 's. The results are surfacelet coefficients for a particular angle. In general, the dimension of the

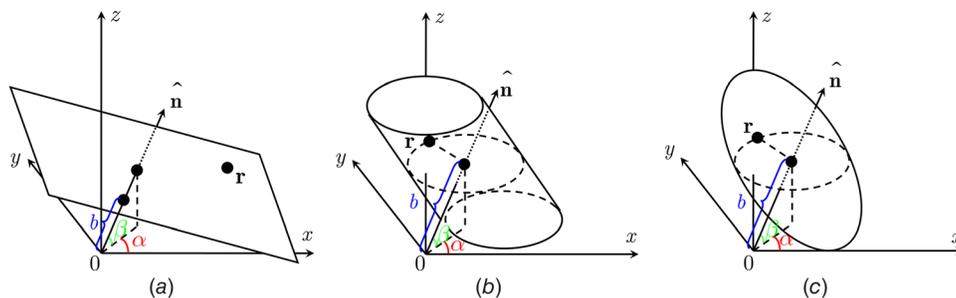


Fig. 2 Geometric interpretation of surfacelets

matrix in the transformed space corresponds to the number of orientation and position parameters used in the surfacelet.

We are also interested in reconstructing the original material images so that we are able to visualize the generated material distribution before physical experiments. In other words, we need the method of inverse surfacelet transform. As shown in Fig. 1, the surface integrals can be directly obtained from the surfacelet coefficients via the inverse 1D wavelet transform. However, it is not as straightforward to retrieve the individual image pixel values from the surface integrals. In this paper, we propose numerical algorithms to calculate individual image pixel values from surface integrals by solving constrained least-square error problems based on an iterative scheme. The relevant existing conjugate-gradient algorithms to solve constrained least-square error problems for image reconstruction are summarized next.

2.5 Conjugate-Gradient-Based Iterative Algorithms. The conjugate gradient method is a well-known numerical method for solving least-square error problems by iteratively minimizing the squared norm of the difference between the measured and the estimated data. However, such methods are not widely applied in image reconstruction. Limited development has been made in this class of methods [23–26]. One possible reason for the limitation is that the conjugate-gradient-based methods can only solve over-constrained problems. For image reconstruction in tomography, the number of projections should be larger than the number of pixels. However, the available data of projections are usually incomplete.

Because of the incomplete data of projections, additional constraints are needed to find reasonable solutions of the linear equation system. Tam and Perez-Mendez [5] and Kawata and Nalcioglu [6] used the so-called object-boundary constraint for geometric information. However, these two methods have three major disadvantages. First, the pixels are simply classified to be those inside and outside the object, and there are no independent constraints applied on the boundary pixels. These methods are, therefore, not suitable for image reconstruction in materials characterization, where boundaries and interfaces that are often found in materials should be clearly defined. Second, the pixels outside the object in these methods were simply constrained to be zero, leading to the result that the image information of outer portion is completely lost and not reconstructed. More importantly, the pixel positions are manually assigned in both methods, which is not applicable for complex and random object distributions, such as in the images of composites.

In this paper, we develop a new constraint-based conjugate-gradient iterative approach for image reconstruction. Our method is able to identify and reconstruct the external or internal boundaries in materials. Based on the constrained conjugate gradient approach, we propose a method of inverse surfacelet transform to retrieve the image pixel values from the surface integrals so that the surfacelet transform formalism is complete. Through matching the geometries of surfacelet primitives, this method is able to automatically identify the position and orientation of the object boundary and apply different constraints on the boundary, which provides more control on the constraints according to material properties. The forward and inverse surfacelet transforms provide a method for knowledge-based image compression that is especially suitable for images of materials with microstructures, since the components in such materials are usually known to us.

3 Inverse Surfacelet Transform

In the surfacelet transform, the surface integrals can be obtained by the summation of all the pixel values on the surfacelets. For instance, for each integral $t_q(\alpha, \beta, \mu)$ corresponding to a cylinderlet with the orientation parameters α and β , and the position parameter μ , there exists a simple linear relationship of summation to approximate the integral. Here, μ is the translation along x -axis. $\alpha \in [0, 2\pi)$ and $\beta \in [-\pi/2, \pi/2]$ are the angular parameters corresponding to

rotations around z - and y -axes in the Euclidean space, respectively. When this summation is applied to all of the surface integrals with P pixels and Q surfacelets, we can obtain

$$\mathbf{A}\mathbf{V} = \mathbf{T} \quad (2)$$

where \mathbf{A} is a $Q \times P$ matrix with coefficient components a_{qp} as either 1 or 0, $\mathbf{V} = (v_p)$ is a P -dimensional vector for pixel values v_p 's ($p = 1, \dots, P$), and $\mathbf{T} = (t_q)$ is a Q -dimensional vector for surface integrals t_q 's ($q = 1, \dots, Q$). $a_{qp} = 1$ if the corresponding pixel is on the surfacelet; and $a_{qp} = 0$ otherwise. Suppose that the numbers of discretized μ , α , and β are u , f , and g , respectively. The number of surface integrals is $Q = u \times f \times g$. At the same time, if the dimension of each image is $L \times M$ and there are N parallel images, then the total number of pixels is $P = L \times M \times N$. It should be noted that the number of pixels on a surfacelet varies from surfacelet to surfacelet, which means that the numbers of ones and zeros on each row of matrix \mathbf{A} vary. Furthermore, a zero integral will be obtained if the surfacelet is positioned or oriented such that it is out of the image domain. In this case, it does not provide useful information and should be avoided if possible. When a surfacelet only covers the pixels that are also covered by other surfacelets, dependency between rows in the coefficient matrix \mathbf{A} occurs, which is very likely to occur. As a result, the Q linear equations with non-zero integrals are not necessarily linearly independent.

The solution of Eq. (2) depends on the relationship between P and Q . It is obvious that when $P = Q$ and the Q linear equations with non-zero integrals are independent, there is only one exact real solution. When $P < Q$, the unknown v_p 's are over-constrained and there is no exact solution. When $P > Q$, the unknown v_p 's are under-constrained and there are an infinite number of solutions. Even in the ideal case of $P = Q$, it is possible that a zero integral occurs or the equations are dependent. Then the coefficient matrix \mathbf{A} is singular, and the equation will have an infinite number of solutions.

Our proposed approaches solve both cases of $P < Q$ and $P > Q$. The case of $P < Q$ applies when we need to restore image data losslessly. In contrast, $P > Q$ is the case when the surfacelet transform is used in lossy compression, which is more common in applications. In the inverse surfacelet transform, classic numerical methods are utilized to solve the case of $P < Q$. A new prior-knowledge-based image reconstruction method is proposed here for the case of $P > Q$, where the boundaries of the key geometric features known a priori are automatically identified and located by the primitives of surfacelets. Then the boundary and/or inner pixels are constrained based on the prior knowledge in order to add more conditions to those from surfacelets. A constrained conjugate gradient algorithm is used to treat these conditions as independent constraints. In addition, a new semi-rigid constrained conjugate gradient algorithm is also proposed to provide a more flexible way to constrain pixels.

We will describe the inverse surfacelet transform without constraints in Sec. 3.1, and the inverse surfacelet transform with constraints in Sec. 3.2.

3.1 Inverse Surfacelet Transform Without Constraints. Solving the linear equation $\mathbf{B}\mathbf{X} = \mathbf{b}$ is equivalent to solving

$$\min F(\mathbf{X}) = \frac{1}{2} \mathbf{X}^T \mathbf{B} \mathbf{X} - \mathbf{X}^T \mathbf{b}$$

which can be done by convex quadratic programming.

In the case of $P < Q$, approximation methods are available to numerically solve the over-constrained problems. The two most used ones are the general least-square and conjugate gradient methods. Both are based on the minimization of least-square errors. Equation (2) is then formulated as

$$\min \left(\sum_{q=1}^Q (\mathbf{A}_{q\bullet} \mathbf{V} - t_q)^2 \right)$$

where $\mathbf{A}_{q\bullet}$ is the q th row of matrix \mathbf{A} . The solution of the general least-square method is obtained by $\mathbf{V} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{T}$, where \mathbf{A} is not necessarily a square matrix.

The conjugate gradient method is an iterative approach to search for the numerical solution of linear equation systems with a symmetric and positive-definite coefficient matrix. The idea is that we iteratively search for \mathbf{V} as the solution of

$$\min_{\mathbf{V}} \frac{1}{2} \mathbf{V}^T \mathbf{B} \mathbf{V} - \mathbf{V}^T \mathbf{b} \quad (3)$$

corresponding to $\mathbf{B} \mathbf{V} = \mathbf{b}$, where \mathbf{B} is a symmetric and positive-definite matrix. Since matrix \mathbf{A} in Eq. (2) is not symmetric and positive-definite, we cannot directly apply to the method. Rather, we transform the equation to

$$\mathbf{A}^T \mathbf{A} \mathbf{V} = \mathbf{A}^T \mathbf{T} \quad (4)$$

and let $\mathbf{A}^T \mathbf{A} = \mathbf{B}$ and $\mathbf{A}^T \mathbf{T} = \mathbf{b}$, since $\mathbf{A}^T \mathbf{A}$ is symmetric. Additionally, because $t_i \geq 0$, $\mathbf{T}^T \mathbf{T} = (\mathbf{A} \mathbf{V})^T \mathbf{A} \mathbf{V} = \mathbf{V}^T \mathbf{A}^T \mathbf{A} \mathbf{V} > 0$. Therefore, $\mathbf{A}^T \mathbf{A}$ is also positive-definite. Thus, by applying the conjugate gradient method to Eq. (4), we can solve the linear equations.

Although we are able to obtain a numerical solution for the case of $P < Q$, it contradicts to our initial intention of surfacelets, which is designed for image compression. As a result, we also need to cope with the case of $P > Q$, where there are many solutions of the minimization problem in Eq. (3). Extra knowledge with respect to the materials applications acting as constraints can help us to narrow the scope of solutions, as discussed in Sec. 3.2.

3.2 Inverse Surfacelet Transform With Constraints

3.2.1 General Procedure. The main goal of inverse surfacelet transform with constraints is to add more equations as constraints in an under-constrained system so as to reduce the number of possible solutions and convert it to an over-constrained system. In order to make $P < Q$ possible, even when limited surfacelets are available, additional constraints based on the prior knowledge of materials can be added. One possible type of constraint is based on geometry, or the shape of the object of interest that is already known in the images. Take fibrous porous media as an example. Suppose we already know the geometry of the fibers, and we are generally more interested in obtaining the information of orientations and positions of the fibers than the detailed microstructures of the fibers and matrix from reconstruction. As a result, the pixels on the surfaces of the fibers are more important than others. By properly restricting the pixel values of the interested object based on prior knowledge, the additional constraints provide more information of the object. Since the object is identified as a geometric feature in the inverse surfacelet transform, hereafter, we call the object *feature*.

In the forward surfacelet transform, the type and parameters of surfacelets are determined by the geometric similarity between the material compositions and some surfacelet primitives, such as 3D ridgelet, cylinderlet, and ellipsoidlet. For instance, images for fibers in composites can be reconstructed by the cylinderlet, whereas those for nano ellipsoidal fillers in nano-reinforced composites can be reconstructed by the ellipsoidlet. In grey scale images, white pixels have the value of 255, black ones have 0, and grey ones have intermediate values according to the corresponding grey scales. Usually, the geometry in the images of materials that we are interested in is brighter than other regions. The surfacelets that cover the feature boundary thus have the largest integral values. The shape and dimension of the surfacelet should be chosen to simulate the reconstructed geometry boundary with the largest integral value achieved when the surfacelet is overlapped with the feature boundary.

Boundary features are identified as follows. If there is more than one target feature to be reconstructed, such as many fibers in composite materials, it is likely that the largest integrals for each target are very close to each other and it is difficult to differentiate. In this case, the integrals can be grouped into different clusters according to their positions and orientations, and the largest integral in each cluster determines a feature boundary.

Once the corresponding pixels are identified by the rule of the largest surface integral described above, their values are set to be equal based on the fact that the grey scale of the geometry is relatively uniform. The equal values are transformed into additional linear equations by each two adjacent pixels forming an equation, such as $v_p = v_{p+1}$. The set of newly formed equations is represented in a matrix form as $\mathbf{C} \mathbf{V} = \mathbf{0}$, where \mathbf{C} is the $S \times P$ constraint coefficient matrix for S constraints with components c_{sp} as $-1, 0$, or 1 . $c_{sp} = 1$ when the corresponding voxel is on the left of the equation, $c_{sp} = -1$ when the corresponding voxel is on the right of the equation, and $c_{sp} = 0$ when the corresponding voxel is not in the equation. The S linear equations are from the $S + 1$ pixels with equal values on the boundary. Since the number $S + 1$ is usually very large, when these equations are combined with Q equations from surface integrals, the number of conditions we know can be dramatically increased. When the total number of equations is larger than the number of unknowns, the number of solutions is no longer infinite. As a result, fewer surfacelets are required so that the surface integral data needed for image reconstruction can be reduced and compression can be achieved.

Furthermore, if the inner structure or geometry of the feature is not of interest, or it has identical grey scale with the boundary such as the inside portion of fibers, the constraint can be further extended such that both the boundary pixel values and the inner ones are set to be equal. Thus, the number of constraint equations can be further increased and the data needed for image reconstruction can be further compressed.

3.2.2 Inverse Surfacelet Transform With Soft Constraints. If the constraint equations are directly added to the ones from surface integrals to form a new set of equations, all equations are treated equally. The equation set is formulated as

$$\begin{bmatrix} \mathbf{A} \\ \mathbf{C} \end{bmatrix} \mathbf{V} = \begin{bmatrix} \mathbf{T} \\ \mathbf{0} \end{bmatrix}$$

The requirement is that after the addition of constraints, the total number of equations is larger than the number of pixels. In other words, all equations have the same weight. In this case, we call the constraints soft, because these constraints will not necessarily be all satisfied in solving the over-constrained system by the least-square methods.

3.2.3 Inverse Surfacelet Transform With Rigid Constraints. If the constraint equations are treated separately from the ones from surface integrals and more rigidly restricted, the constraints weigh more. In other words, the constraints are stronger than otherwise being directly added to the equations from surface integrals. In this case, we call the constraints rigid. It is solved by the constrained conjugate gradient method with constraints separate from the equation system, similar to Ref. [27]. This is an extension of the conjugate gradient method described in Sec. 3.1. The problem can be described as

$$\begin{aligned} \min_{\mathbf{V}} \frac{1}{2} \mathbf{V}^T \mathbf{A}^T \mathbf{A} \mathbf{V} - \mathbf{V}^T \mathbf{A}^T \mathbf{T} \\ \text{s.t. } \mathbf{C} \mathbf{V} = \mathbf{0} \end{aligned} \quad (5)$$

The conjugate gradient algorithm under rigid constraints to solve Eq. (5) is listed in Table 1. The main procedure is an Arnoldi style iteration. The enforcement of rigid constraints is by a projection of the residual \mathbf{r} to \mathbf{z} through an orthogonal projection

Table 1 Rigid constrained conjugate gradient algorithm

INPUT: matrix \mathbf{A} , constraint matrix \mathbf{C} , initial guess \mathbf{V}_0 OUTPUT: \mathbf{V}_{k+1}

```

1   $\mathbf{T} := \mathbf{T}_0$ 
2   $\mathbf{H} := \mathbf{Z}(\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T$  where  $\mathbf{CZ} = \mathbf{0}$ 
3   $\mathbf{r}_0 := \mathbf{A}^T\mathbf{A}\mathbf{V}_0 - \mathbf{A}^T\mathbf{T}$ 
4   $\mathbf{z}_0 := \mathbf{H}\mathbf{r}_0$ 
5   $\mathbf{d}_0 := \mathbf{z}_0$ 
6   $k := 0$ 
7  Repeat
8     $v_k := \frac{-\mathbf{r}_k^T\mathbf{d}_k}{\mathbf{d}_k^T\mathbf{A}^T\mathbf{A}\mathbf{d}_k}$ 
9     $\mathbf{V}_{k+1} := \mathbf{V}_k + v_k\mathbf{d}_k$ 
10    $\mathbf{r}_{k+1} := \mathbf{r}_k + v_k\mathbf{A}^T\mathbf{A}\mathbf{d}_k$ 
11    $\omega_k := \frac{\mathbf{r}_{k+1}^T\mathbf{z}_{k+1}}{\mathbf{r}_k^T\mathbf{z}_k}$ 
12    $\mathbf{d}_{k+1} := -\mathbf{z}_{k+1} + \omega_k\mathbf{d}_k$ 
13   If  $\mathbf{d}_{k+1}^T\mathbf{d}_{k+1}$  is sufficiently small
14     Exit loop
15   End If
16    $k := k + 1$ 
17 End repeat
18 Return  $\mathbf{V}_{k+1}$ 

```

matrix $\mathbf{H} = \mathbf{Z}(\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T$, where \mathbf{Z} forms a basis for the null space of \mathbf{C} that is $\mathbf{CZ} = \mathbf{0}$. It should be noted that in this method, it is not required that $P < Q$ for matrix \mathbf{A} .

3.2.4 Inverse Surfacelet Transform With Semi-rigid Constraints. The method with soft constraints in Sec. 3.2.2 does not enforce the constraints, whereas the one with rigid constraints in Sec. 3.2.3. requires all constraints to be satisfied. Here we also provide a third option with more flexibility about the constraints, which is called semi-rigid. The semi-rigid constraints are realized through an exterior penalty function, which includes constraints in the objective function. The rigidity of the constraints can be controlled by their weights in the objective function. With the exterior penalty function, Eq. (5) can be transformed into an optimization problem with the objective function only as

$$\frac{1}{2}\mathbf{V}^T\mathbf{A}^T\mathbf{A}\mathbf{V} - \mathbf{V}^T\mathbf{A}^T\mathbf{T} + \sum_{i=1}^S w_i(\mathbf{C}_{i\bullet}\mathbf{V})^2$$

where $\mathbf{C}_{i\bullet}$ is the i th row vector of the constraint coefficient matrix \mathbf{C} . When all weights are set to be equal, the objective function can be further simplified as

$$\frac{1}{2}\mathbf{V}^T\mathbf{A}^T\mathbf{A}\mathbf{V} - \mathbf{V}^T\mathbf{A}^T\mathbf{T} + w\mathbf{V}^T\sum_{i=1}^S \mathbf{C}_{i\bullet}^T\mathbf{C}_{i\bullet}\mathbf{V} \quad (6)$$

In order to use the conjugate gradient method to solve the new objective function in Eq. (6), it needs to be further transformed into

$$\mathbf{V}^T\mathbf{G}\mathbf{V} - \mathbf{V}^T\mathbf{A}^T\mathbf{T}$$

where $\mathbf{G} = \frac{1}{2}\mathbf{A}^T\mathbf{A} + w\sum_{i=1}^S \mathbf{C}_{i\bullet}^T\mathbf{C}_{i\bullet}$.

In our method of semi-rigid constrained conjugate gradients, some constraints can be treated in a different way other than strictly soft or rigid. For instance, in a fiber, the constraints on the internal pixels can be softer than the ones for the boundary pixels. Then the weights associated with the constraints for the internal pixels are smaller than the ones for the boundaries. Once the rigidity is controlled by assigning different weight values, softer constraints can be introduced based on the inner pixels in addition to the boundaries. With more constraints, the number of required

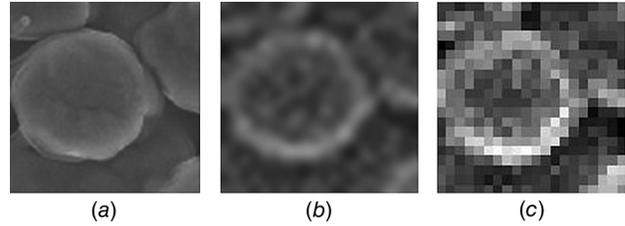


Fig. 3 The full and down-sized images of nano-fiber composites

surfacelets can be potentially further reduced without losing the information of pixels outside the object boundary.

4 Examples and Results

In this section, examples and results for the methods described in Sec. 3 are demonstrated and compared. Experiments were conducted in Matlab. For all examples, nine images of a small portion of a nano-fiber composite are used. The images are in the format of JPEG, one of them is shown in Fig. 3(a). The full size of the picture is 80×80 . Thus, the total number of pixels is $P = 80 \times 80 \times 9 = 57,600$, and the dimension of the coefficient matrix \mathbf{A} is large. The original images are down sized to 20×20 , as shown in Figs. 3(b) and 3(c), and used in the demonstration. Although the blurring image in Fig. 3(b) and the non-blurring image in Fig. 3(c) are visually different, they are exactly the same in terms of pixel values. The images are then converted into grey scale in Matlab. The total number of pixels used in the following examples is $P = 20 \times 20 \times 9 = 3600$. The nine resized images are shown in Fig. 4. They are slightly different from each other.

To quantitatively compare the results of different methods, we introduce the error measurement

$$e = \sqrt{\frac{1}{P} \sum_{i=1}^P (V_i - V_i^0)^2}$$

where V_i^0 is the i th pixel value in the original images and V_i is the one from the reconstructed images.

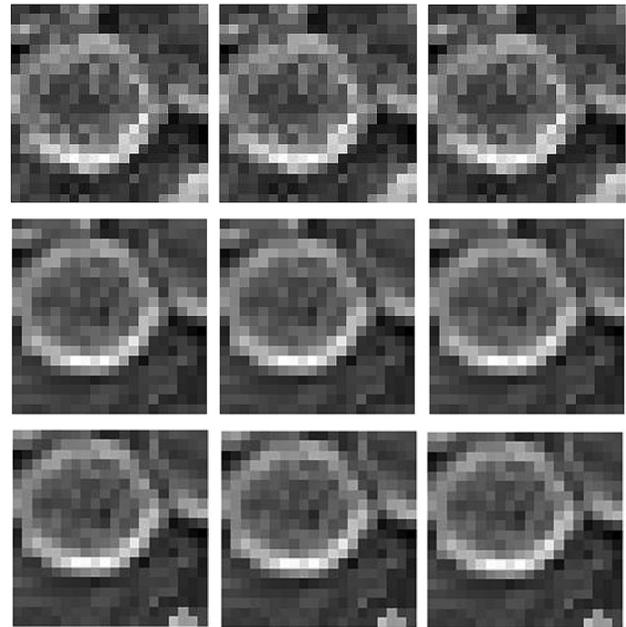


Fig. 4 The original nine parallel images of nano-fiber composites for reconstruction

The cylinderlet is used to reconstruct the material images for all examples. The shape parameters of the cylinderlet are $r_1 = 1$ and $r_2 = 2$. The ranges of the orientation parameters are set to be $\mu \in (-D/2, D/2) = (-14.84, 14.84)$, $\alpha \in [0, 2\pi)$ and $\beta \in [-\pi/2, \pi/2]$ to ensure that the surfacelets cover all of the pixels, where $D = \sqrt{20^2 + 20^2 + 9^2}$ is the diagonal length of the 3D images.

4.1 Without Constraints. Reconstruction with both the least-square and conjugate gradient methods without constraints was conducted. The result shows that the full reconstruction of the original images occurs when $u = 80$, $f = 80$, and $g = 5$. The total number of surfacelets are $Q = u \times f \times g = 32,000$ and $Q > P$. These threshold values of u , f , and g allow all of the pixels in the images to be covered by the surfacelet boundaries without missing any. With a careful comparison between pixel values, it is concluded that the least-square and conjugate gradient methods provide the same reconstruction result with these 32,000 surfacelets. The original images are retrieved losslessly.

4.2 With Constraints

4.2.1 Method of Automatic Fiber Boundary Identification.

For the example of the nano-fiber composites, the positions and orientations of the fibers are of interest. Geometric information of the fibers, such as the shape and size as prior knowledge to us, can be utilized as additional constraints. The constraints are imposed by examining the surfacelet integrals. As shown in Fig. 5, the cylinderlet denoted by the solid circle is overlapped with a fiber surface and has a larger integral value than others such as the two denoted by the circles of dashed line. Therefore surface integrals help determine the positions and orientations of the nano-fibers. If there is only one nano fiber, its position and orientation can be directly estimated by the cylinderlet with the largest integral. This can be realized by sorting the surface integrals from the results of forward surfacelet transform. If there is more than one fiber, the largest integral for one fiber can be very close to the one for another fiber, because some surfacelets are overlapped with multiple fibers. In this case, the integrals are grouped into different clusters according to their positions and orientations, and the largest integral in each cluster determines a fiber. As the constraints, the pixel values on the surfacelets can be set to be equal to each other.

As can be seen from Fig. 3, there are four fibers in the images. However, there is only one complete fiber. As a result, the boundary integrals for other fibers are much smaller than that of the complete one. The nano fibers are close to each other, so the small surfacelet clusters of the partial fibers could be mixed with the complete one. Therefore, it is not easy to identify those partial fibers. This problem will be addressed in future work. In this

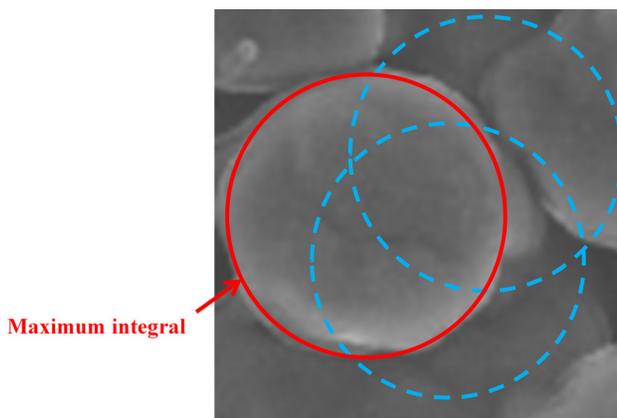


Fig. 5 The cylinderlet overlapped with a fiber surface has the maximum integral

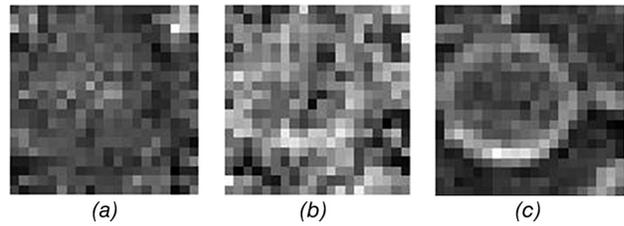


Fig. 6 Reconstruction results of soft boundary constraints with the least-square method

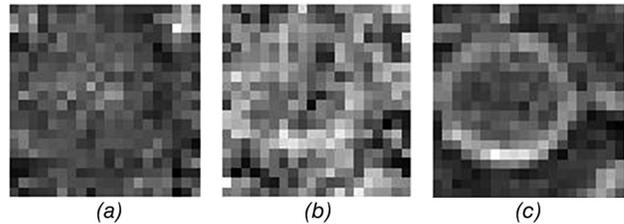


Fig. 7 Reconstruction results of soft boundary constraints with the conjugate gradient method

paper, we only identify complete fibers, and there is only one in this example.

4.2.2 Soft Constraints. In the case of soft constraints, the constraints of equal pixel values at the fiber boundaries are directly added to the original set of equations. The number of pixels on the identified fiber boundary is 340. Therefore, 339 constraint equations are added. In other words, the dimension of the constraint matrix \mathbf{C} is 339×3600 . As stated in Sec. 3, the requirement is that the total number of equations is larger than the number of pixels. The results with the least-square method are shown in Fig. 6, where the number of surfacelets used from the forward surfacelet transform is $Q = 58 \times 58 \times 4 = 13,456$, $Q = 60 \times 60 \times 4 = 14,400$, and $Q = 70 \times 70 \times 5 = 24,500$, respectively. Since the nine resulting images are similar, only one image is shown in Fig. 6 for comparison. It can be seen that when $Q = 70 \times 70 \times 5 = 24,500$, the error $e = 4.8$ is the smallest among the three. The images in this case are very close to the original ones. Similar results are obtained with the conjugate gradient method, as shown in Fig. 7. Thus, with the soft boundary constraints, the number of surfacelets required to retrieve the original images is smaller than the one in the methods without constraints in Sec. 4.1. However, the number of surfacelets is still larger than the number of pixels.

In order to further increase the number of constraints, the inner pixels of the surfacelets are also set to be equal but different from the boundary ones. Therefore, the problem can be formulated as

$$\begin{bmatrix} \mathbf{A} \\ \mathbf{C}_1 \\ \mathbf{C}_2 \end{bmatrix} \mathbf{V} = \begin{bmatrix} \mathbf{T} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$

where \mathbf{C}_1 is the constraint coefficient matrix for fiber boundary pixels and \mathbf{C}_2 is the constraint coefficient matrix for fiber inner pixels. Since the number of inner pixels is large, the increase of the number of constraint equations is significant. The number of pixels inside the identified fiber boundary is 995. Therefore, the dimension of the constraint matrix \mathbf{C}_2 is 994×3600 . The results of the method are shown in Fig. 8. When the number of surfacelets is too small as in the case of Fig. 8(a), the effective (linear independent) equations can be fewer than the unknown. Therefore, the solution is actually not unique.

From the results, it can be seen that the number of surfacelets can be further reduced from the ones with only boundary

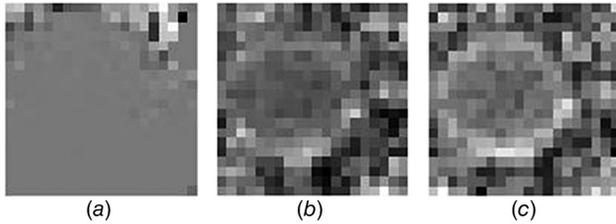


Fig. 8 Reconstruction results of soft fiber boundary and inner constraints

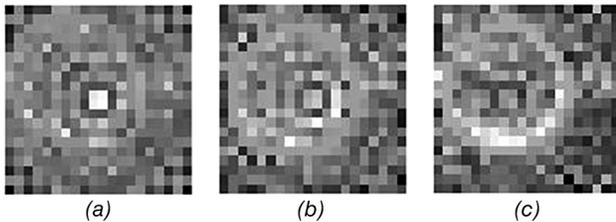


Fig. 9 Reconstruction results of rigid fiber boundary constraints

constraints as in Figs. 6 and 7 to achieve similar results. After the addition of constraining equations, the equation system becomes over-constrained. The numerical solutions by minimizing the errors provide approximations of the true values.

4.2.3 Rigid Constraints. When rigid constraints of boundaries are applied, the results are shown in Fig. 9. Compared to the method with soft fiber boundary and inner constraints, the method with rigid boundary constraints is able to reconstruct the image by much fewer surfacelets with similar errors. More importantly, this method realizes image compression, which is one of the important intentions of the surfacelet transform. It can be seen in Fig. 9(c) that as few as 1875 surfacelets can be used to reconstruct 3600 pixels with a small error. The compression rate is approximately 50% in this example. All of the resulting nine images are shown in Fig. 10.

Further compression may be realized if the inner pixels are also constrained. In other words, equality constraints are separately applied on both fiber boundary and fiber inner pixels. The problem can be formulated as the objective function of Eq. (5) with constraints $C_1 \mathbf{V} = \mathbf{0}$ and $C_2 \mathbf{V} = \mathbf{0}$. The results are shown in Fig. 11. It can be seen that the number of surfacelets is significantly reduced.

The drawback of this method is that although the key features of the complete fiber remain, the detailed information about the partial fibers is lost. The reason is that the constraints have a significant influence on the solution. Therefore, weaker constraints may provide better results.

4.2.4 Semi-rigid constraints. From the results of Sec. 4.2.3, it can be seen that some tradeoffs are needed in selecting the reconstruction methods. When the rigid constraints are applied on the boundary only, the number of surfacelets is greater than when the rigid constraints of boundary and inner pixels are both applied, where the number of constraints increases. However, the errors with the rigid boundary constraints are smaller. Here, we show that the method of semi-rigid constraints provides a third option with more flexibility.

The results for semi-rigid fiber boundary and inner constraints with different combinations of weights are shown in Figs. 12–14, respectively. It can be seen that when penalty weights for boundary pixels are equal to 1×10^{10} , the results have the smallest error. Compared to the method with rigid fiber boundary and inner constraints, the results of the semi-rigid method have clearer fiber boundaries. Compared to the method with only rigid fiber boundary

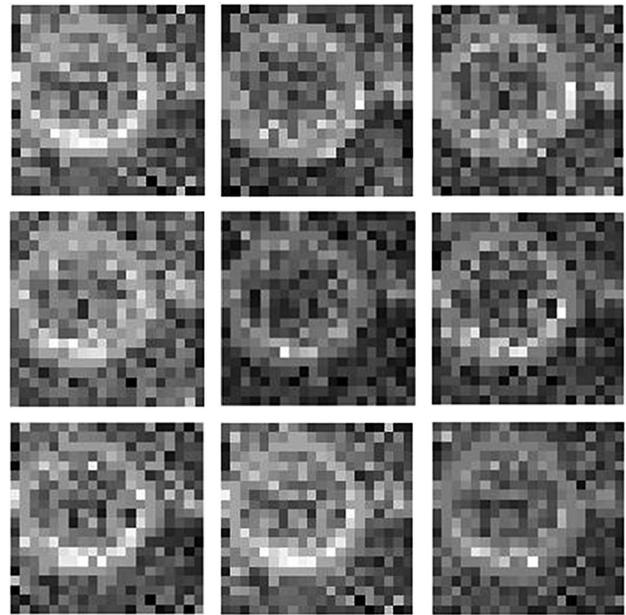


Fig. 10 Full reconstruction results of rigid fiber boundary constraints in the case of $Q = 25 \times 25 \times 3 = 1875$. The error is $e = 13.2$.

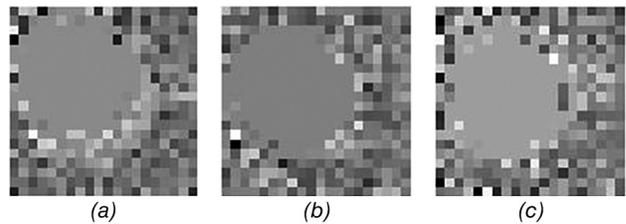


Fig. 11 Reconstruction results of rigid fiber boundary and inner constraints

constraints, although the error is similar, the pixels inside the fiber have smoother transition, which increases the contrast between the pixels on and off the fiber. Although the contrast is also increased if rigid constraints are added on both boundaries and inner pixels as in Fig. 12, the boundary is not as clear as in Fig. 13.

The new constrained conjugate gradient methods developed in this paper can be generalized to the image reconstruction of any composite material with emphasis on the locations and orientations of the boundaries of the fillers.

5 Evaluation and Comparison

The quasi-Newton method with line search is used as a comparison with the proposed constrained conjugate gradient methods. When the constraints are only applied on the boundary, the results for different numbers of equations are as shown in Fig. 15.

The comparison of the results from the different methods is shown in Table 2. The experiments are conducted on a personal computer (PC) with 2.00 GHz central processing unit (CPU) and 4.00 GB random-access memory (RAM). The number of equations used in the comparison is $25 \times 25 \times 3 = 1875$, and the constraints are for the boundary pixels only.

The robustness of the methods is evaluated by using four different initial guesses (0, 1, 100, and 255) for all pixel values to start the optimization algorithms. The initial guesses of 0 and 255 are the lowest and highest possible pixel values. Therefore, they are the worst cases as initial guesses. The initial guesses of 1 and 100 are two samples of intermediate values. If the optimization

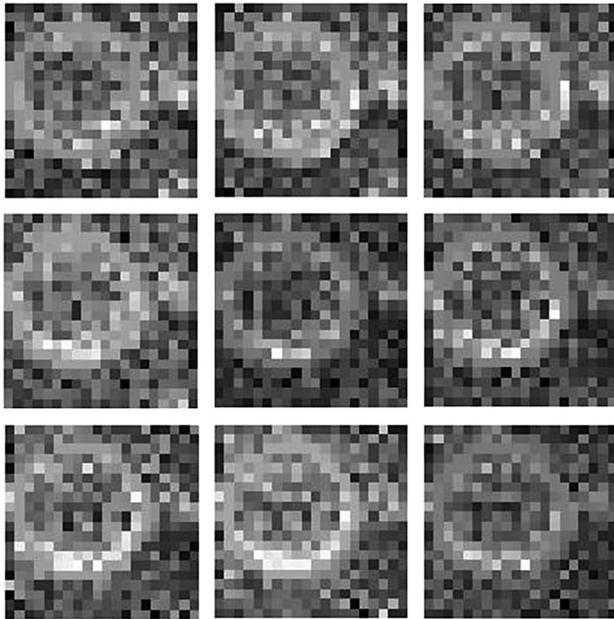


Fig. 12 Reconstruction results of semi-rigid constraints with penalty weights for boundary pixels equal to 1×10^{10} and for inner pixels equal to 10. The error is $e = 13.2$.

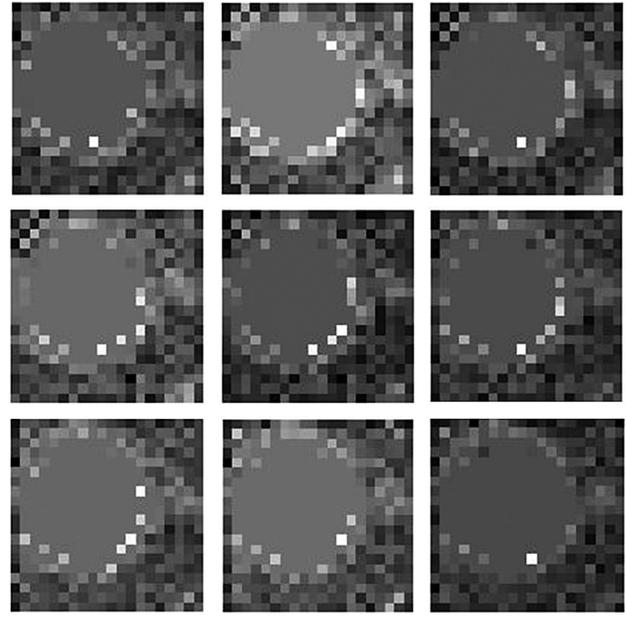


Fig. 14 Reconstruction results of semi-rigid constraints with penalty weights for both boundary and inner pixels equal to 1×10^{10} . The error is $e = 16.5$.

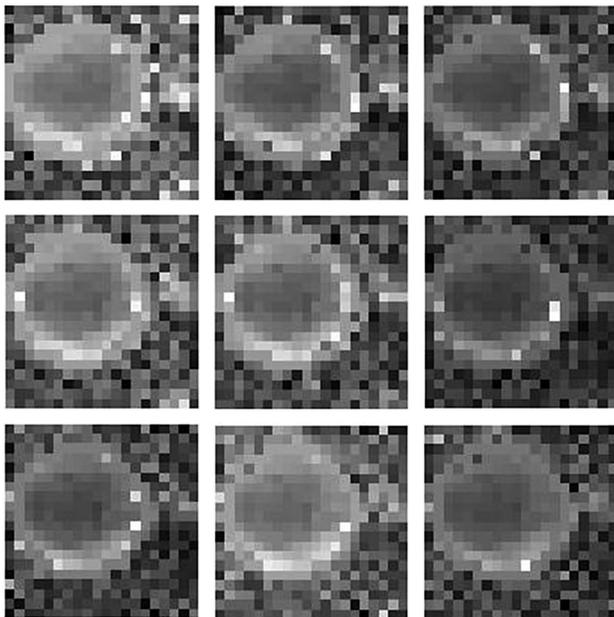


Fig. 13 Reconstruction results of semi-rigid constraints with penalty weights for boundary pixels equal to 1×10^{10} and for inner pixels equal to 1×10^3 . The error is $e = 12.9$.

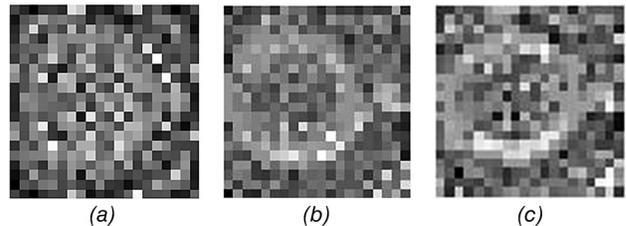


Fig. 15 Reconstruction results of Quasi-Newton method with line search with fiber boundary constraints

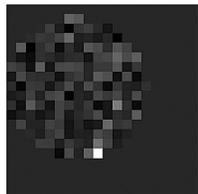
results of an algorithm are nearly the same for those four guesses, the algorithm is regarded as being robust. Our results show that the differences between the resulting errors from the different initial guesses in our methods are all less than 5. Therefore the methods are robust. As can be seen in Table 2, the conjugate gradient algorithm with rigid constraints and the quasi-Newton method with line search show the best results in terms of error and time efficiency. Between the quasi-Newton method and the constrained conjugate gradient algorithm, the latter shows slightly worse results with greater error but much better computational efficiency. The constrained conjugate gradient method is 20 times more efficient than the quasi-Newton method. The constrained

conjugate gradient method realizes image compression with good reconstruction results. It can be seen that 1875 surfacelets can reconstruct 3600 pixels reasonably well with small errors. The rigid and semi-rigid constrained conjugate gradient methods both show similar and good precision, but they also have different advantages. The rigid method has significantly higher computational efficiency, whereas the semi-rigid method produces more distinguishable boundaries of objects. It should be noted that, with similar boundary pixels, when the contrast between the pixels of inner object and the ambient pixels is larger, the object is more distinguishable. In this paper, this contrast is mathematically defined as the standard deviation difference. In the above example, the rigid method has the absolute standard deviation difference value of 0.34. In contrast, the semi-rigid method has the absolute standard deviation difference value of 4.38. The geometric information of inner boundaries is of particular interest to material engineers for the purpose of material modeling or property estimation. Although all methods discussed above are able to automatically identify the boundaries, more distinguishable boundaries in reconstruction are very important to engineers for easy identification with human eyes from the image reconstruction point of view. It should be noted that, for the soft constraints with the least-square method, because the number of equations is smaller than the number of unknown pixels, no specific reconstruction result is available.

The proposed methods for inverse surfacelet transform are based on solving a quadratic optimization problem from linear equations. The computational time is related to the number of pixels in the images. The CPU time of 219 s for the rigid constrained

Table 2 The comparison among different methods with boundary constraints

Optimization method	Average error	CPU time	Iterations (generations)	Robustness
Direct linear equations solution	31 (But with 7 times more equations required)	186 s	N/A	N/A
Rigid constrained conjugate gradient algorithm	13.2	219 s	522	Robust
Quasi-Newton method with line search	8	4230 s	139	Robust
Semi-rigid constrained conjugate gradient algorithm	12.9	2025 s	2512	Robust
Soft constraints with the least square method	N/A	198 s	N/A	N/A

**Fig. 16 Reconstruction result (the first image only) based on the object-boundary constraint in Kawata and Nalcioğlu [6]**

conjugate gradient algorithm is not short for the small example images. Yet, if implemented in other generic machine-oriented languages such as C/C++ in real applications, the computational time can be reduced.

With the same number of surfacelets, we also compare our constraint imposition approach with the one used by Kawata and Nalcioğlu [6]. The first of the nine reconstructed images based on the object-boundary constraint in Kawata and Nalcioğlu [6] is shown in Fig. 16. As can be seen, there is no clear differentiation between the boundary and inner pixels. In addition, all ambient pixels are blacked out. Therefore, a large amount of microstructural information is lost during the reconstruction process.

6 Conclusion and Future Work

In this paper, we proposed a constrained conjugate-gradient based strategy for the inverse surfacelet transform to complete the surfacelet transform formalism. By identifying and applying constraints on important pixels of interest, features can be preserved and retrieved during reconstruction. The proposed methods of inverse surfacelet transform with constraints are able to reconstruct images with fewer surfacelets than image pixels for the purpose of compression, by utilizing the prior knowledge of geometric features. Compared to the generic image compression methods, our method allows us to preserve boundary information of features more efficiently in material images, in addition to the integrated capability of feature identification.

In the future, the proposed algorithm will be further optimized for the improvement of computational efficiency, which will enable larger images of more complex material microstructures to be processed. Parallel computing and a machine-oriented language can be utilized. Additionally, in this paper, we only demonstrated features similar to the cylinderlet primitives. In future, techniques for reconstructing more complex features with combination of different types of surfacelet primitives will be developed. Furthermore, the prior knowledge of microstructure characteristics other than feature geometry, such as mechanics of materials, can also be considered.

References

- [1] Radon, J., 1986, "On the Determination of Functions from their Integral Values Along Certain Manifolds," *IEEE Trans. Med. Imaging*, 5(2), pp. 170–176.

- [2] Hough, P. V. C., 1959, "Machine Analysis of Bubble Chamber Pictures," *Proceeding of International Conference High Energy Accelerators and Instrumentation*.
- [3] Duda, R. O., and Hart, P. E., 1972, "Use of the Hough Transformation to Detect Lines and Curves in Pictures," *Comm. ACM*, 15, pp. 11–15.
- [4] Wang, Y., and Rosen, D. W., 2010, "Multiscale Heterogeneous Modeling With Surfacelets," *Comput.-Aided Des. Appl.*, 7, pp. 759–776.
- [5] Tam, K. C., and Perez-Mendez, V., 1981, "Tomographical Imaging With Limited-Angle Input," *J. Opt. Soc. Amer.*, 71(5), pp. 582–592.
- [6] Kawata, S., and Nalcioğlu, O., 1985, "Constrained Iterative Reconstruction by the Conjugate Gradient Method," *IEEE Trans. Med. Imaging*, 4, pp. 65–71.
- [7] Bracewell, R. N., 1990, "Numerical Transforms," *Science*, 248(4956), pp. 697–704.
- [8] Leavers, V. F., and Boyce, J. F., 1987, "The Radon Transform and Its Application to Shape Parametrization in Machine Vision," *Image Vision Comput.*, 5(2), pp. 161–166.
- [9] Jeong, N., Rosen, D. W., and Wang, Y., 2013, "A Comparison of Surfacelet-Based Method for Recognizing Linear Geometric Features in Material Microstructure," *Proceeding of IDETC2013*, Portland, Oregon, August 4–7, DETC2013-13370.
- [10] Rosen, D. W., Jeong, N., and Wang, Y., 2013, "A Method for Reverse Engineering of Material Microstructure for Heterogeneous CAD," *Comput.-Aided Des.*, 45(7), pp. 1068–1078.
- [11] Ballard, D. H., 1981, "Generalizing the Hough Transform to Detect Arbitrary Shapes," *Pattern Recognit.*, 13(2), pp. 111–122.
- [12] Niezgodá, S. R., and Kalidindi, S. R., 2010, "Applications of the Phase-Coded Generalized Hough Transform to Feature Detection, Analysis, and Segmentation of Digital Microstructures," *Comput. Mater., Contin.*, 14(2), pp. 79–98.
- [13] Donoho, D., 1999, "Wedgelets: Nearly Minimax Estimation of Edges," *Ann. Stat.*, 27(3), pp. 895–897.
- [14] Candèsand, E. J., and Donoho, D. L., 2002, "Recovering Edges in Ill-Posed Inverse Problems: Optimality of Curvelet Frames," *Ann. Stat.*, 30(3), pp. 784–842.
- [15] Candès, E. J., 1998, "Ridgelets: Theory and Applications," Ph.D. dissertation, Stanford University, Stanford, CA.
- [16] Do, M. N., and Vetterli, M., 2005, "The Contourlet Transform: An Efficient Directional Multiresolution Image Representation," *IEEE Trans. Image Process.*, 14(12), pp. 2091–2106.
- [17] Donoho, D. L., and Huo, X., 2002, "Beamlets and Multiscale Image Analysis," *Multiscale and Multiresolution Methods: Theory and Applications*, T. J. Barth, T. Chan, and R. Haimes, Eds., Springer, Berlin Heidelberg, pp. 149–196.
- [18] Huo, X., 2005, "Beamlets and Multiscale Modeling," *Entry for the 2nd Edition of Encyclopedia of Statistical Sciences*, C. B. Read, N. Balakrishnan, and B. Vidakovic, Eds., Wiley & Sons, NJ.
- [19] Willett, R. M., and Nowak, R. D., 2003, "Platelets: A Multiscale Approach for Recovering Edges and Surfaces in Photon-Limited Medical Imaging," *IEEE Trans. Med. Imaging*, 22(3), pp. 332–350.
- [20] Ying, L., Demanet, L., and Candès, E., 2005, "3D Discrete Curvelet Transform," *Proceeding of SPIE Conference Wavelet Applications in Signal & Image Processing*, San Diego, CA, p. 5914.
- [21] Lu, Y. M., and Do, M. N., 2007, "Multidimensional Directional Filter Banks and Surfacelets," *IEEE Trans. Image Process.*, 16(4), pp. 918–931.
- [22] Chandrasekaran, V., Wakin, M. B., Baron, D., and Baraniuk, R. G., 2009, "Representation and Compression of Multidimensional Piecewise Functions Using Surflets," *IEEE Trans. Inf. Theory*, 55(1), pp. 374–400.
- [23] Scales, J. A., 1987, "Tomographic Inversion Via the Conjugate Gradient Method," *Geophysics*, 52(2), pp. 179–185.
- [24] Piccolomini, E. L., and Zama, F., 1999, "Conjugate Gradient Regularization Method in Computed Tomography Problems," *Appl. Math. Comput.*, 102(1), pp. 87–99.
- [25] Qiu, W., Tittle-Peloquin, D., and Soleimani, M., 2012, "Blockwise Conjugate Gradient Methods for Image Reconstruction in Volumetric CT," *Comput. Methods Program Biomed.*, 108(2), pp. 669–678.
- [26] Arunprasath, T., Rajasekaran, M. P., Kannan, S., and Kalasalingam, V. A., 2012, "Reconstruction of PET Brain Image Using Conjugate Gradient Algorithm," *Proceedings of 2012 World Congress on Information and Communication Technologies*, Oct. 30 2012–Nov. 2 2012, Trivandrum, India, pp. 73–78.
- [27] Shariff, M. H. B. M., 1995, "A Constrained Conjugate Gradient Method and the Solution of Linear Equations," *Comput. Math. Appl.*, 30(11), pp. 25–37.