SIAM J. IMAGING SCIENCES
Vol. 9, No. 3, pp. 1437–1466

Digital Affine Shear Transforms: Fast Realization and Applications in Image/Video Processing*

Xiaosheng Zhuang†

Abstract. In this paper, we discuss the digitization and applications of smooth affine shear tight frames, a recently developed new class of directional multiscale representation systems. An affine wavelet tight frame is generated by isotropic dilations and translations of directional wavelet generators, while an affine shear tight frame is generated by anisotropic dilations, shears, and translations of shearlet generators. These two tight frames are actually connected in the sense that an affine shear tight frame can be obtained from an affine wavelet tight frame through subsampling. Consequently, an affine shear tight frame has an underlying filter bank from the MRA structure of its associated affine wavelet tight frame. We discuss the digitization of digital affine shear filter banks associated with the affine shear tight frames. Moreover, we provide the detailed algorithmic steps for both the forward and backward digital affine shear transforms. Analysis of the redundancy rate and computational complexity shows that the redundancy rate of the digital affine shear transforms does not increase with respect to the number of directions and the computational complexity is proportional to the redundancy rate and the FFT time for a fixed size of input data. Numerical experiments and comparisons in image/video processing show the advantages of our digital affine shear transforms over many other state-of-the-art frame-based directional transforms.

Key words. digital affine shear transforms, smooth affine shear tight frames, smooth affine wavelet tight frames, directional filter banks, image processing, video processing, denoising, inpainting

AMS subject classifications. 41A25, 42B05, 42C40, 65T60, 68U10, 68W35, 68W40, 94A12, 94A08

DOI. 10.1137/15M1048318

1. Introduction and motivation.

Modern information technologies have become inevitable parts of people's everyday lives. For example, smartphones are now standard platforms for social activities, e-shopping, entertainment, news, etc.; wearable digital devices provide individual data acquisition and analysis for personal health and sport activities; virtual reality (VR) and augmented reality devices enable and enhance our experience of the world to another level of perspective; and the Internet of Things makes it possible to build a smart city for both public and individual services. Behind all these types of smart technologies are the generating, storage, processing, transmitting, analyzing, and applications of tremendous amounts of data. In short, we are in the era of Big Data. On the one hand, the complexity of data obtained from modern technologies grows exponentially, e.g., high-resolution images

*Received by the editors November 20, 2015; accepted for publication (in revised form) July 25, 2016; published electronically September 15, 2016. A preliminary version appeared as *smooth affine shear tight frames: Digitization and applications*, in Wavelets and Sparsity XVI, Proc. SPIE 9597, SPIE, Bellingham, WA, 2015.
http://www.siam.org/journals/siims/9-3/M104831.html
Funding: This work was supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (project CityU 11304414) and grants from City University of Hong Kong (projects 7200462 and 7004445).
†Department of Mathematics, City University of Hong Kong, Kowloon Tong, Hong Kong (xzhuang7@cityu.edu.hk, http://personal.cityu.edu.hk/~xzhuang7/).

from smart phones and cameras, video scenes in video games and VR devices, nD-signals from massive-surveillance systems, seismic data, climate data, financial data, and so on. On the other hand, the growth of the computational power predicted by Moore's law might not be able to catch up with the growth of data in the near future due to the limit of physical laws. Smart learning from high-dimensional data for information extraction and applications hence no doubt becomes one of the hotspots of today's scientific research.

The key to massive data analysis lies in the fact that the data collected are typically high-dimensional while the feature information is structurally low-dimensional (anisotropic features). Directional multiscale representation systems play a key role in the area of high-dimensional data analysis due to their ability to capture anisotropic (directional) features such as edges in cartoon-like images, surface singularities in three-dimensional (3D) objects, etc. In a nutshell, the exponential growth of data can be tackled via exploiting the low-dimensional structures (sparsity) of the data using representation systems with the ability to capture anisotropic features (directional information) while the computational efficiency must be guaranteed by the intrinsic low-redundancy representation of the directional systems. Hence, efficient representation systems with (extremely) low redundancy are crucial to the successful applications of directional multiscale representation systems in high-dimensional data analysis.

Multiscale representation systems starting from wavelets [4, 6, 24] have had a tremendous impact on many fields such as electrical engineering, image/signal processing, computer graphics, and numerical solutions of PDE. In theory, wavelet systems are investigated and developed with fewer and fewer constraints from orthogonal wavelet systems to wavelet frames (framelets). In practice, many application problems require the development of wavelet systems with more and more desirable properties, e.g., high order of vanishing moments, symmetry, and regularity. Wavelet systems are typically generated by applications of isotropic dilations and translations to wavelet generators and high-dimensional wavelets are usually obtained through the tensor product of 1D wavelet systems, e.g., tensor product Daubechies orthonormal wavelets, which only favor a few directions and hence are not able to capture adequate information of high-dimensional data. Due to the aforementioned needs of anisotropic analysis in high-dimensional data, many directional multiscale representation systems have been developed, such as dual-tree complex wavelets [26], tensor product complex tight framelets [15, 17, 18], curvelets [2, 3], contourlets and surfacelets [5, 23], and shearlets [1, 9, 10, 11, 12, 20]. By using two trees of orthonormal wavelets and a careful design of wavelet generators, one can construct a 1D complex wavelet system with generators that form a Hilbert pair whose frequency response concentrates on either the positive or the negative axis. The tensor product of such 1D complex wavelet systems is the dual-tree complex wavelets (DT-$\mathbb{C}$WT [26]) that can achieve more directionality (six directions in dimension two: $\pm 15°$, $\pm 45°$, $\pm 75°$) than the usual real-valued tensor product wavelets. Exploiting further the idea of frequency separation, the tensor product complex tight framelets TP-$\mathbb{C}$TF$_m$ can achieve even more direction selectivity (14 directions in dimension two for TP-$\mathbb{C}$TF$_6$) through the tensor product of 1D complex tight framelets $\mathbb{C}$TF$_m$ generated by framelet generators that have more frequency separability [15, 17, 18]. Though by relaxing real-valued wavelet systems to complex-valued wavelet systems, one can achieve directionality through the tensor product of 1D wavelet/framelet systems, due to isotropic dilation and the fixed number of directions

across all scales, a fundamental deficiency of tensor product wavelet systems is their inefficient representation of cartoon-like functions in high dimensions [7]. In order to achieve an optimal approximation rate of cartoon-like functions, one needs to consider non-tensor-product type representation systems. Curvelets [2], in addition to (parabolic) dilation and translation operations, employ an extra rotation operation to generate a directional representation system that provides (almost) optimal approximation rate for cartoon-like functions. However, rotation operation does not preserve integer lattice, which is undesirable in practice. By replacing rotation operation with shear operation, shearlets [8, 9, 10, 11, 16] generated from applications of parabolic dilation, translation, and shear not only provide an (almost) optimal approximation rate for cartoon-like functions but also have group structures similar to group structures of wavelets.

Using the framework of frequency-based affine systems [14], smooth affine shear tight frames, one of the directional multiscale representation systems generated through dilation, translation, and shear, have been developed and studied systematically in dimension two in [19] by the author and his collaborator. It has been shown that smooth affine shear tight frames include all known shearlet tight frames as special cases. One of the key results in [19] reveals that an affine shear tight frame can be regarded as a subsampled system of an affine wavelet tight frame that has an MRA structure, thereby associating an affine shear tight frame with an underlying directional filter bank. This key result indicates that affine shear tight frames are less "redundant" than affine wavelet tight frames. Moreover, the initial attempt of digitization in [19] shows that affine shear tight frames can be efficiently implemented using their underlying filter banks and are very similar to the standard fast wavelet transforms.

In this paper, we further investigate d-dimensional smooth affine shear tight frames for any $d \geqslant 2$. The contributions of this paper lie in the following aspects. First, we provide detailed steps for the construction, digitization, and application of d-dimensional smooth affine shear/wavelet tight frames. Second, it is well-known that one of the key features of wavelets is their unified treatment of both the digital and continuum worlds through their associated wavelet filter banks [24]. However, implementations of existing curvelet/shearlet transforms are simply through direct sampling of the curvelet/shearlet function generators on the digital lattices [3, 21, 22, 25], which lacks the unified treatment of both the continuum and digital worlds. The connection between affine shear tight frames and affine wavelet tight frames in this paper provides a faithful "bridge" between the continuum and digital realms for directional representation systems generated through the operations of dilation, translation, and shear in any dimension $d \geqslant 2$. Last but not least, unlike many other implementations of directional filter banks, e.g., [5, 22], whose redundancy rate could skyrocket as the number of directions increases due to the employment of undecimated transforms, our analysis shows a striking feature of our implementation: the redundancy rate does not increase with respect to the number of directions (see Table 1). Moreover, thanks to the use of FFT, both our forward and backward digital affine shear transforms have computational complexity proportional to $rN \log N$ with r the redundancy rate and N the size of the input data. Such features are very much desirable properties in practice, especially in image/video processing where the resolution of data becomes higher and higher and more and more directions are needed.

The structure of the paper is as follows. In section 2, we introduce d-dimensional affine shear systems for any $d \geqslant 2$ and provide the characterization of a sequence of affine shear

systems with generators having nonnegative frequency responses to be a sequence of affine shear tight frames, which greatly facilitates our construction of smooth affine shear tight frames in any dimension $d \geqslant 2$. Moreover, we introduce affine wavelet systems and show that the affine shear tight frames we constructed are closely related to affine wavelet systems in the sense that one can obtain an affine shear tight frame through subsampling of an affine wavelet tight frame with an associated directional filter bank. In section 3, we turn to the digitization of smooth affine shear tight frames: digital affine shear filter banks. We provide detailed algorithmic steps for both the forward and backward digital affine shear transforms. Furthermore, we investigate the redundancy rate and computational complexity of our d-dimensional digital affine shear transforms. In section 4, numerical experiments and comparisons on some image/video processing problems (denoising and inpainting) are given to show the efficiency and advantages of our digital affine shear transforms over many other state-of-the-art frame-based directional transforms. Technical proofs of some results are postponed to section 5.

2. Smooth affine shear tight frames in $\mathbb{R}^d$.

In this section, we extend the definition of smooth affine shear tight frames from dimension two (cf. [19]) to any dimension $d \geqslant 2$. We first introduce the notion of a sequence of d-dimensional affine shear systems and then provide the characterization for a sequence of affine shear systems to be a sequence of affine shear tight frames in $L_2(\mathbb{R}^d)$ as well as the construction of both d-dimensional affine shear tight frames and affine wavelet tight frames. The connection between these two types of systems is given in the last subsection. It should be mentioned that though the extension in this section is conceptually straightforward from dimension two to any dimension $d \geqslant 2$, it is still technical to clearly state and prove many main results in arbitrary dimension. We have made an effort to simplify our notation for dealing with high-dimensional construction in this section as well as digital realization in the next section, e.g., the summation in arbitrary dimension and the splitting function $\boldsymbol{\gamma}_\varepsilon$ in section 2.2.

2.1. Characterization of affine shear tight frames for $L_2(\mathbb{R}^d)$.

Let U be a $d \times d$ invertible matrix. Throughout the paper, we shall assume $d \geqslant 2$ and use the compact notation

$$f_{\mathsf{U};\mathsf{k},\mathsf{n}}(x) := |\det \mathsf{U}|^{1/2} f(\mathsf{U}x - \mathsf{k}) e^{-i\mathsf{n}\cdot \mathsf{U}x}, \quad \mathsf{k},\mathsf{n}, x \in \mathbb{R}^d,$$

to encode dilation U, translation k, and modulation n for a function f defined on $\mathbb{R}^d$. The shear operator $\mathsf{S}^{\vec{\tau}}$ with $\vec{\tau} = (\tau_2, \ldots, \tau_d) \in \mathbb{R}^{d-1}$, anisotropic dilation matrix A_λ, and isotropic dilation matrix M_λ with $\lambda > 1$ are of the form

$$\mathsf{S}^{\vec{\tau}} = \begin{bmatrix} 1 & \tau_2 & \cdots & \tau_d \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}, \quad \mathsf{A}_\lambda = \begin{bmatrix} \lambda^2 & 0 & \cdots & 0 \\ 0 & \lambda & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda \end{bmatrix}, \quad \text{and} \quad \mathsf{M}_\lambda = \begin{bmatrix} \lambda^2 & 0 & \cdots & 0 \\ 0 & \lambda^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda^2 \end{bmatrix}.$$

We shall use $\mathsf{N}_\lambda := \mathsf{M}_\lambda^{-\mathsf{T}}$ and $\mathsf{B}_\lambda := \mathsf{A}_\lambda^{-\mathsf{T}}$ to denote the transpose of the inverse of M_λ and A_λ, respectively. Note that $\mathsf{M}_\lambda = \mathsf{A}_\lambda \mathsf{D}_\lambda$ with $\mathsf{D}_\lambda := \operatorname{diag}(1, \lambda \mathsf{I}_{d-1})$, where I_n denotes the $n \times n$ identity matrix. Define $\mathsf{S}_{\vec{\tau}} := (\mathsf{S}^{\vec{\tau}})^{\mathsf{T}}$ and denote E_n to be the $d \times d$ elementary matrix corresponding to the coordinate exchange between the first axis and the nth one. For example, $\mathsf{E}_1 = \mathsf{I}_d$, $\mathsf{E}_2 = \operatorname{diag}(\left[\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right], \mathsf{I}_{d-2})$, $\mathsf{E}_3 = \operatorname{diag}(\left[\begin{smallmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{smallmatrix}\right], \mathsf{I}_{d-3})$, and so on.

Let Ψ_j be a set of generators for the high-frequency part at scale j given by

$$\Psi_j := \{\psi^{j,\vec{\ell}} : |\vec{\ell}| \leqslant \vec{s}_j\},\ \vec{\ell} := (\ell_2, \dots, \ell_d) \in \mathbb{Z}^{d-1}, \vec{s}_j := (s_{j,2}, \dots, s_{j,d}) \in \mathbb{N}_0^{d-1}, \tag{2.1}$$

where $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$ and $\psi^{j,\vec{\ell}}$ are functions in $L_2(\mathbb{R}^d)$. Here and after we shall use the compact notation $\sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j}$ for $\sum_{\ell_2=-s_{j,2}}^{s_{j,2}} \cdots \sum_{\ell_d=-s_{j,d}}^{s_{j,d}}$ and $|\vec{\ell}| \leqslant \vec{s}_j$ (or $\vec{\ell} = -\vec{s}_j, \dots, \vec{s}_j$) for $\{\vec{\ell} \in \mathbb{Z}^{d-1} : |\ell_2| \leqslant s_{j,2}, \dots, |\ell_d| \leqslant s_{j,d}\}$. Let $\varphi^j \in L_2(\mathbb{R}^d)$ be some scaling function for the low-frequency part at scale j. Then a *d-dimensional affine shear system* (starting at scale $J \in \mathbb{Z}$) is defined to be

$$\mathsf{AS}(\varphi^J; \{\Psi_j\}_{j=J}^\infty) := \{\varphi^J_{\mathsf{M}_\lambda^J;\mathsf{k}} : \mathsf{k} \in \mathbb{Z}^d\} \cup \{\psi^{j,\vec{\ell}}_{\mathsf{S}^{-\vec{\ell}}\mathsf{A}_\lambda^j\mathsf{E}_n;\mathsf{k}} : \mathsf{k} \in \mathbb{Z}^d,\ |\vec{\ell}| \leqslant \vec{s}_j,\ n = 1, \dots, d\}_{j=J}^\infty. \tag{2.2}$$

We say that $\mathsf{AS}(\varphi^J; \{\Psi_j\}_{j=J}^\infty)$ is an *affine shear tight frame* for $L_2(\mathbb{R}^d)$ if

$$\|f\|_2^2 = \sum_{\mathsf{k}\in\mathbb{Z}^d} |\langle f, \varphi^J_{\mathsf{M}_\lambda^J;\mathsf{k}}\rangle|^2 + \sum_{j=J}^\infty \sum_{n=1}^d \sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j} \sum_{\mathsf{k}\in\mathbb{Z}^d} |\langle f, \psi^{j,\vec{\ell}}_{\mathsf{S}^{-\vec{\ell}}\mathsf{A}_\lambda^j\mathsf{E}_n;\mathsf{k}}\rangle|^2 \quad \forall f \in L_2(\mathbb{R}^d), \tag{2.3}$$

where $\langle f, g\rangle := \int_{\mathbb{R}^d} f(x)\overline{g(x)}dx$ and $\|f\|_2 := \sqrt{\langle f, f\rangle}$ for $f, g \in L_2(\mathbb{R}^d)$.

The *Fourier transform* $\widehat{f}$ of a function $f \in L_1(\mathbb{R}^d)$ is defined to be $\widehat{f}(\xi) := \int_{\mathbb{R}^d} f(x)e^{-ix\cdot\xi}dx$ for $\xi \in \mathbb{R}^d$ and can be naturally extended to functions in $L_2(\mathbb{R}^d)$. We denote by $\mathscr{D}(\mathbb{R}^d)$ the linear space of all compactly supported C^∞ (test) functions with the usual topology. In this paper, we are only interested in affine shear tight frames with generators that are nonnegative in the frequency domain. That is, $\widehat{h} \geqslant 0$ for all $h \in \{\varphi^J\} \cup \{\Psi^j\}_{j=J}^\infty$. Following the same lines of proof as [19, Corollary 3] (see also [14, Corollary 18]), we have the following simple characterization (see section 5 for its proof) that will facilitate our construction of affine shear tight frames in the next subsection.

Theorem 2.1. *Let $J_0 \in \mathbb{Z}$ be an integer and $\mathsf{AS}(\varphi^J; \{\Psi_j\}_{j=J}^\infty)$ be defined as in (2.2) with $J \geqslant J_0$. Suppose $\widehat{h} \geqslant 0$ for all $h \in \{\{\varphi^j\} \cup \Psi_j\}_{j=J_0}^\infty$. Then, for all integer $J \geqslant J_0$, $\mathsf{AS}(\varphi^J; \{\Psi_j\}_{j=J}^\infty)$ is an affine shear tight frame for $L_2(\mathbb{R}^d)$ if and only if the following holds:*

$$\widehat{h}(\xi)\widehat{h}(\xi + 2\pi\mathsf{k}) = 0 \text{ a.e. } \xi \in \mathbb{R}^d,\ \mathsf{k} \in \mathbb{Z}^d\backslash\{0\},\ \forall h \in \{\{\varphi^j\} \cup \Psi_j\}_{j=J}^\infty, \tag{2.4}$$

$$|\widehat{\varphi^{j+1}}(\mathsf{N}_\lambda^{j+1}\xi)|^2 = |\widehat{\varphi^j}(\mathsf{N}_\lambda^j\xi)|^2 + \sum_{n=1}^d \sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j} |\widehat{\psi^{j,\vec{\ell}}}(\mathsf{S}_{\vec{\ell}}\mathsf{B}_\lambda^j\mathsf{E}_n\xi)|^2 \quad \text{a.e. } \xi \in \mathbb{R}^d,\ j \geqslant J_0, \text{ and} \tag{2.5}$$

$$\lim_{j\to\infty} \langle |\widehat{\varphi^j}(\mathsf{N}_\lambda^j\cdot)|^2, \widehat{h}\rangle = \langle 1, \widehat{h}\rangle \quad \forall \widehat{h} \in \mathscr{D}(\mathbb{R}^d). \tag{2.6}$$

We next discuss how the above characterization greatly simplifies our construction of affine shear tight frames for $L_2(\mathbb{R}^d)$. First, conditions (2.5) and (2.6) together imply that for all $J \geqslant J_0$, $\{\varphi^J\} \cup \{\Psi^j\}_{j=J}^\infty$ forms a "partition of unity" in the frequency domain:

$$1 = |\widehat{\varphi^J}(\mathsf{N}_\lambda^j\xi)|^2 + \sum_{j=J}^\infty \sum_{n=1}^d \sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j} |\widehat{\psi^{j,\vec{\ell}}}(\mathsf{S}_{\vec{\ell}}\mathsf{B}_\lambda^j\mathsf{E}_n\xi)|^2 \quad \text{a.e. } \xi \in \mathbb{R}^d.$$

Second, condition (2.4) means that in the frequency domain each of the generators does not overlap with its $2\pi\mathsf{k}$ shift for any nonzero integer k. Last but not least, (2.5) reveals that generators $\psi^{j,\vec{\ell}}$ are from the "splitting" of $|\widehat{\varphi^{j+1}}(\mathsf{N}_\lambda^{j+1}\xi)|^2 - |\widehat{\varphi^j}(\mathsf{N}_\lambda^j\xi)|^2$. Consequently, the construction of a sequence of affine shear tight frames is reduced to the construction of scaling functions φ^j and a proper "splitting" procedure. In [19], we proposed two types of construction for $\mathbb{R}^2$: one is nonstationary where φ^j is different at different scale j, and the other is quasi-stationary where $\varphi^j = \varphi$ is fixed across all scale j. In this paper, for simplicity of presentation, we consider only the quasi-stationary construction and remark that all results can be extended to the nonstationary construction similar to that in [19] without much difficulty. In such a case, (2.5) is further simplified to

$$|\widehat{\varphi}(\mathsf{N}_\lambda\xi)|^2 - |\widehat{\varphi}(\xi)|^2 = \sum_{n=1}^{d}\sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j} |\widehat{\psi^{j,\vec{\ell}}}(\mathsf{S}_{\vec{\ell}}\mathsf{D}_\lambda^j\mathsf{E}_n\xi)|^2 \quad \text{a.e. } \xi\in\mathbb{R}^d,\ j\geqslant J_0, \tag{2.7}$$

and (2.6) can be easily satisfied for bounded $\widehat{\varphi}$ satisfying $\lim_{j\to\infty}|\widehat{\varphi}(\mathsf{N}_\lambda^j\cdot)| = 1$ a.e. (see [19, Lemma 1]), which significantly reduced the complexity of the construction of affine shear tight frames.

2.2. Construction of smooth affine shear tight frames for $L_2(\mathbb{R}^d)$. Let us next detail the quasi-stationary construction of smooth affine shear tight frames, in which case $\varphi^j \equiv \varphi$ for all j. Roughly speaking, in the frequency domain, using auxiliary 1D Meyer-type scaling and wavelet functions $\boldsymbol{\alpha}_{\lambda,t,\rho}$, $\boldsymbol{\beta}_{\lambda,t,\rho}$, and "bump" function $\boldsymbol{\gamma}_\varepsilon$ (see Figure 1), we build $\widehat{\varphi}$ to be the tensor product of $\boldsymbol{\alpha}_{\lambda,t,\rho}$ while the core generator shall look like $\boldsymbol{\beta}_{\lambda,t,\rho}(\xi_1)\prod_{n=2}^d\boldsymbol{\gamma}_\varepsilon(\xi_n/\xi_1)$, which is a wedge shape generator in dimension $d=2$ and a pyramid shape generator in dimension $d=3$. Application of parabolic scaling, shear, and translation operations to such a generator induces our smooth affine shear systems. Further technical treatments are then applied on such systems to achieve tightness.

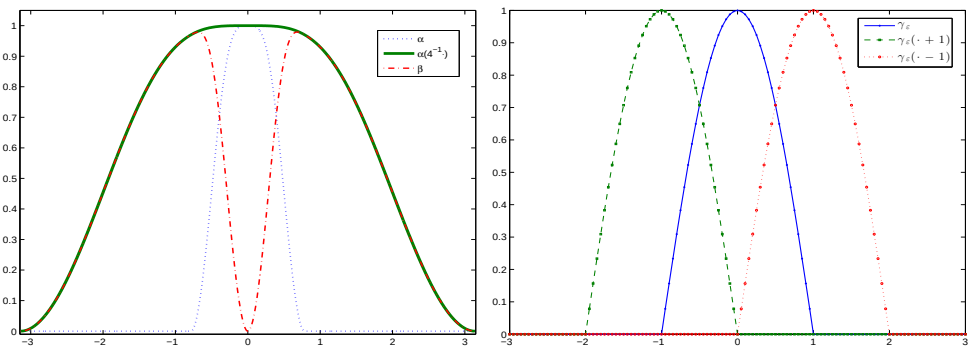

Figure 1. *Left: Graphs of $\boldsymbol{\alpha}_{\lambda,t,\rho}$ (dotted line, blue), $\boldsymbol{\alpha}_{\lambda,t,\rho}(\lambda^{-2}\cdot)$ (solid line, green), and $\boldsymbol{\beta}_{\lambda,t,\rho}$ (dash-dot line, red) for $\rho=1,\lambda=2,t=1$. Note that $\boldsymbol{\beta}_{\lambda,t,\rho}$ overlaps with $\boldsymbol{\alpha}(\lambda^{-2}\cdot)$ for $\xi\geqslant\lambda^{-2}\rho\pi$. Right: Graphs of $\boldsymbol{\gamma}_\varepsilon$ (solid line, blue), $\boldsymbol{\gamma}_\varepsilon(\cdot+1)$ (dashed line, green), and $\boldsymbol{\gamma}_\varepsilon(\cdot-1)$ (dotted line, red) with $\varepsilon=1/2$. Note that $\sum_{\ell\in\mathbb{Z}}|\boldsymbol{\gamma}_\varepsilon(\cdot+\ell)|^2=1$.*

We next briefly introduce the auxiliary functions $\boldsymbol{\alpha}_{\lambda,t,\rho}$, $\boldsymbol{\beta}_{\lambda,t,\rho}$, and $\boldsymbol{\gamma}_\varepsilon$. Let $\nu\in C_c^\infty(\mathbb{R})$ be a compactly supported C^∞ function such that $\nu(x)=0$ for $x\leqslant -1$, $\nu(x)=1$ for $x\geqslant 1$,

and $|\nu(x)|^2 + |\nu(-x)|^2 = 1$ for all $x \in \mathbb{R}$. There are many choices of such functions. For example, as in [13], we define $f(x) := e^{-1/x^2}$ for $x > 0$ and $f(x) := 0$ for $x \leqslant 0$ and let $g(x) := \int_{-1}^{x} f(1+t)f(1-t)dt$. Define $\nu(x) := \frac{g(x)}{\sqrt{|g(x)|^2+|g(-x)|^2}}$ for $x \in \mathbb{R}$. Then $\nu \in C_c^\infty(\mathbb{R})$ is a desired function. From ν, we define our building block function $\boldsymbol{\nu}_{[c,\epsilon]}$ for $c \geqslant 0$ and $0 < \epsilon \leqslant c$ as follows:

$$\boldsymbol{\nu}_{[c,\epsilon]}(x) = \begin{cases} \nu(\frac{x+c}{\epsilon}) & \text{if } x < -c+\epsilon, \\ 1 & \text{if } -c+\epsilon \leqslant x \leqslant c-\epsilon, \\ \nu(\frac{-x+c}{\epsilon}) & \text{if } x > c-\epsilon. \end{cases} \tag{2.8}$$

It is easy to see that $\boldsymbol{\nu}_{[c,\epsilon]}$ is a smooth "bump" function supported on $[-c-\epsilon, c+\epsilon]$. Let $\lambda > 1$, $0 < t \leqslant 1$, $0 < \rho \leqslant \lambda^2$, and $0 < \varepsilon \leqslant 1/2$. Define $\boldsymbol{\alpha}_{\lambda,t,\rho}, \boldsymbol{\beta}_{\lambda,t,\rho}$ of Meyer-type scaling and wavelet functions and $\boldsymbol{\gamma}_\varepsilon$ as follows (see Figure 1):

$$\begin{cases} \boldsymbol{\alpha}_{\lambda,t,\rho}(\xi) & := \boldsymbol{\nu}_{[\lambda^{-2}(1-t/2)\rho\pi, \lambda^{-2}t\rho\pi/2]}(\xi), \\ \boldsymbol{\beta}_{\lambda,t,\rho}(\xi) & := (|\boldsymbol{\alpha}_{\lambda,t,\rho}(\lambda^{-2}\xi)|^2 - |\boldsymbol{\alpha}_{\lambda,t,\rho}(\xi)|^2)^{1/2}, \\ \boldsymbol{\gamma}_\varepsilon(\xi) & := \boldsymbol{\nu}_{[1/2,\varepsilon]}(\xi). \end{cases} \tag{2.9}$$

Then $\boldsymbol{\alpha}_{\lambda,t,\rho}$, $\boldsymbol{\beta}_{\lambda,t,\rho}$, $\boldsymbol{\gamma}_\varepsilon \in C_c^\infty(\mathbb{R})$. Moreover, $\operatorname{supp} \boldsymbol{\alpha}_{\lambda,t,\rho} = [-\lambda^{-2}\rho\pi, \lambda^{-2}\rho\pi]$, $\operatorname{supp} \boldsymbol{\beta}_{\lambda,t,\rho} = [-\rho\pi, -\lambda^{-2}(1-t)\rho\pi] \cup [\lambda^{-2}(1-t)\rho\pi, \rho\pi]$, and $\operatorname{supp} \boldsymbol{\gamma}_\varepsilon = [-1/2-\varepsilon, 1/2+\varepsilon]$. Furthermore, $\boldsymbol{\alpha}_{\lambda,t,\rho}$, $\boldsymbol{\beta}_{\lambda,t,\rho}$ have their associated 2π-periodic scaling mask $\boldsymbol{\mu}_{\lambda,t,\rho}$ and wavelet mask $\boldsymbol{\upsilon}_{\lambda,t,\rho}$, respectively, defined as follows:

$$\begin{aligned} \boldsymbol{\mu}_{\lambda,t,\rho}(\xi) &:= \begin{cases} \frac{\boldsymbol{\alpha}_{\lambda,t,\rho}(\lambda^2\xi)}{\boldsymbol{\alpha}_{\lambda,t,\rho}(\xi)} & \text{if } |\xi| \leqslant \lambda^{-2}\rho\pi, \\ 0 & \text{if } \lambda^{-2}\rho\pi < |\xi| \leqslant \pi, \end{cases} \\ \boldsymbol{\upsilon}_{\lambda,t,\rho}(\xi) &:= \begin{cases} \frac{\boldsymbol{\beta}_{\lambda,t,\rho}(\lambda^2\xi)}{\boldsymbol{\alpha}_{\lambda,t,\rho}(\xi)} & \text{if } \lambda^{-4}(1-t)\rho\pi \leqslant |\xi| \leqslant \lambda^{-2}\rho\pi, \\ \mathbf{g}_{\lambda,t,\rho}(\xi) & \text{if } \xi \in [-\pi,\pi] \backslash \operatorname{supp} \boldsymbol{\beta}_{\lambda,t,\rho}(\lambda^2\cdot), \end{cases} \end{aligned} \tag{2.10}$$

where $\mathbf{g}_{\lambda,t,\rho}$ is a function in $C^\infty(\mathbb{T})$ such that $\left[\frac{d^n}{d\xi^n}\mathbf{g}_{\lambda,t,\rho}(\xi)\right]\big|_{\xi=\pm\lambda^{-2}\rho\pi} = \delta(n)$ for all $n \in \mathbb{N}_0$. The purpose of $\mathbf{g}_{\lambda,t,\rho}$ is to make the function $\boldsymbol{\upsilon}_{\lambda,t,\rho}$ smooth (see [19, section 4.1]). We have

$$\boldsymbol{\alpha}_{\lambda,t,\rho}(\lambda^2\xi) = \boldsymbol{\mu}_{\lambda,t,\rho}(\xi)\boldsymbol{\alpha}_{\lambda,t,\rho}(\xi), \text{ and } \boldsymbol{\beta}_{\lambda,t,\rho}(\lambda^2\xi) = \boldsymbol{\upsilon}_{\lambda,t,\rho}(\xi)\boldsymbol{\alpha}_{\lambda,t,\rho}(\xi), \quad \xi \in \mathbb{R}.$$

Define $\widehat{\varphi}(\xi) := [\otimes \boldsymbol{\alpha}_{\lambda,t,\rho}](\xi) = \prod_{n=1}^d \boldsymbol{\alpha}_{\lambda,t,\rho}(\xi_n)$, $\xi = (\xi_1, \ldots, \xi_d) \in \mathbb{R}^d$, and

$$\boldsymbol{\omega}_{\lambda,t,\rho}(\xi) := \sqrt{|\widehat{\varphi}(\lambda^{-2}\xi)|^2 - |\widehat{\varphi}(\xi)|^2}, \quad \xi \in \mathbb{R}^d. \tag{2.11}$$

Then $\boldsymbol{\omega}_{\lambda,t,\rho} \in C^\infty(\mathbb{R}^d)$. In view of the construction of $\boldsymbol{\alpha}_{\lambda,t,\rho}$, the refinable structure is clear. We have $\widehat{\varphi}(\lambda^2\xi) = \widehat{a}(\xi)\widehat{\varphi}(\xi)$, $\xi \in \mathbb{R}^d$, with $\widehat{a} = \otimes^d \boldsymbol{\mu}_{\lambda,t,\rho} \in C^\infty(\mathbb{T}^d)$ being the tensor product of the 1D mask $\boldsymbol{\mu}_{\lambda,t,\rho}$ given in (2.10). Moreover, we have $\boldsymbol{\omega}_{\lambda,t,\rho}(\lambda^2\xi) = \widehat{b}(\xi)\widehat{\varphi}(\xi)$ with $\widehat{b} \in C^\infty(\mathbb{T}^d)$ being given by $\widehat{b}(\xi) = (\mathbf{g}(\xi) - |\widehat{a}(\xi)|^2)^{1/2}$ for any smooth function $\mathbf{g} \in C^\infty(\mathbb{T}^d)$ such that $\mathbf{g} = 1$ on the support of $\widehat{\varphi}$. In order to obtain $\psi^{j,\vec{\ell}}$, we next split $\boldsymbol{\omega}_{\lambda,t,\rho}$ using $\boldsymbol{\gamma}_\varepsilon$. Note that for

simplicity of presentation, here and after, we omit the dependency of $\varphi, \psi^{j,\vec{\ell}}, a, b, \mathbf{\Gamma}_j$, etc., on the parameters $\lambda, t, \rho, \varepsilon$.

For $x > 1$, define $\ell_x := \lfloor x - (1/2 + \varepsilon) \rfloor + 1 = \lfloor x + (1/2 - \varepsilon) \rfloor$. It is easy to check that $\sum_{\ell \in \mathbb{Z}} |\boldsymbol{\gamma}_\varepsilon(\cdot + \ell)|^2 = 1$. Moreover,

$$\sum_{\ell=-\ell_\lambda}^{\ell_\lambda} |\boldsymbol{\gamma}_\varepsilon(\lambda x + \ell)|^2 = 1 \quad \forall |x| \leqslant \frac{\ell_\lambda + 1/2 - \varepsilon}{\lambda}. \tag{2.12}$$

Let $\vec{\boldsymbol{\gamma}}(\xi) := \prod_{n=2}^d \boldsymbol{\gamma}_\varepsilon(\xi_n/\xi_1)$, $\xi = (\xi_1, \dots, \xi_d) \in \mathbb{R}^d$. We define a d-dimensional splitting piece $\vec{\boldsymbol{\gamma}}^{j,\vec{\ell}}$ and a normalization function $\mathbf{\Gamma}_j$ at scale j as follows:

$$\vec{\boldsymbol{\gamma}}^{j,\vec{\ell}}(\xi) := \vec{\boldsymbol{\gamma}}(\mathsf{S}_{\vec{\ell}}\mathsf{B}_\lambda^j \xi) = \prod_{n=2}^d \boldsymbol{\gamma}_\varepsilon(\lambda^j \xi_n/\xi_1 + \ell_n) \text{ and } \mathbf{\Gamma}_j(\xi) := \sum_{n=1}^d \sum_{\vec{\ell}=-\vec{\ell}_{\lambda^j}}^{\vec{\ell}_{\lambda^j}} \vec{\boldsymbol{\gamma}}^{j,\vec{\ell}}(\mathsf{E}_n \xi) \tag{2.13}$$

for $\xi = (\xi_1, \dots, \xi_n) \in \mathbb{R}^d \backslash \{0\}$, $\vec{\ell} = (\ell_2, \dots, \ell_d) \in \mathbb{Z}^{d-1}$, and $\vec{\ell}_{\lambda^j} := (\ell_{\lambda^j}, \dots, \ell_{\lambda^j}) \in \mathbb{N}_0^{d-1}$. $\vec{\boldsymbol{\gamma}}^{j,\vec{\ell}}(0) = \mathbf{\Gamma}_j(0) := 1$. It is easy to show that $\mathbf{\Gamma}_j$ is smooth ($\mathbf{\Gamma}_j \in C^\infty(\mathbb{R}^d \backslash \{0\})$), positive ($0 < \mathbf{\Gamma}_j(\xi) \leqslant 2$), symmetric ($\mathbf{\Gamma}_j(\mathsf{E}_n \cdot) = \mathbf{\Gamma}_j$ for all $n = 1, \dots, d$), and scale-invariant ($\mathbf{\Gamma}_j(t\xi) = \mathbf{\Gamma}_j(\xi)$ for all $t \neq 0$ and $\xi \neq 0$). Moreover, in view of (2.12), one can show that $\mathbf{\Gamma}_j(\xi) \equiv 1$ for all $\xi = (\xi_1, \dots, \xi_d) \in \mathbb{R}^d$ satisfying $\max\{|\xi_m/\xi_n| : m \neq n, m, n = 1, \dots, d\} \leqslant \frac{\lambda^j}{\ell_{\lambda^j} + 1/2 + \varepsilon}$.

We are now ready to define $\psi^{j,\vec{\ell}}$. Recall that $\mathsf{A}_\lambda = \operatorname{diag}(\lambda^2, \lambda \mathsf{I}_{d-1})$, $\mathsf{B}_\lambda = \mathsf{A}_\lambda^{-\mathsf{T}}$, $\mathsf{D}_\lambda = \operatorname{diag}(1, \lambda \mathsf{I}_{d-1})$, $\mathsf{M}_\lambda = \lambda^2 \mathsf{I}_d = \mathsf{A}_\lambda \mathsf{D}_\lambda$, and $\mathsf{N}_\lambda = \mathsf{M}_\lambda^{-\mathsf{T}}$. Define the set Ψ_j of generators at scale j to be

$$\Psi_j := \{\psi^{j,\vec{\ell}} : |\vec{\ell}| \leqslant \vec{\ell}_{\lambda^j}\} \text{ with } \widehat{\psi^{j,\vec{\ell}}}(\xi) := \boldsymbol{\omega}_{\lambda,t,\rho}(\mathsf{D}_\lambda^{-j} \mathsf{S}_{-\vec{\ell}} \xi) \frac{\vec{\boldsymbol{\gamma}}^{j,\vec{\ell}}((\mathsf{S}_{\vec{\ell}}\mathsf{B}_\lambda^j)^{-1}\xi)}{\sqrt{\mathbf{\Gamma}_j((\mathsf{S}_{\vec{\ell}}\mathsf{B}_\lambda^j)^{-1}\xi)}}. \tag{2.14}$$

Note that $\widehat{\psi^{j,\vec{\ell}}}(\xi) = \boldsymbol{\omega}_{\lambda,t,\rho}(\xi_1, \lambda^{-j}(-\xi_1\ell_2 + \xi_2), \dots, \lambda^{-j}(-\xi_1\ell_d + \xi_d)) \frac{\prod_{n=2}^d \boldsymbol{\gamma}_\varepsilon(\xi_n/\xi_1)}{\sqrt{\mathbf{\Gamma}_j((\mathsf{S}_{\vec{\ell}}\mathsf{B}_\lambda^j)^{-1}\xi)}}$, which gives $\widehat{\psi^{j,\vec{\ell}}}(\mathsf{S}_{\vec{\ell}}\mathsf{B}_\lambda^j \xi) = \boldsymbol{\omega}_{\lambda,t,\rho}(\mathsf{N}_\lambda^j \xi) \frac{\boldsymbol{\gamma}^{j,\vec{\ell}}(\xi)}{\sqrt{\mathbf{\Gamma}_j(\xi)}}$. Since $0 < \mathbf{\Gamma}_j \leqslant 2$ and $\mathbf{\Gamma}_j$ is in $C^\infty(\mathbb{R}^d \backslash \{0\})$, we have that $\sqrt{\mathbf{\Gamma}_j}$ is infinitely differentiable for all $\xi \in \mathbb{R}^d \backslash \{0\}$. Because the support of $\boldsymbol{\omega}_{\lambda,t,\rho}$ is away from the origin, we see that $\psi^{j,\vec{\ell}}$ are functions in $C^\infty(\mathbb{R}^d)$. We have the following (quasi-stationary) *d-dimensional affine shear system*:

$$\mathsf{AS}(\varphi; \{\Psi_j\}_{j=J}^\infty) := \{\varphi_{\mathsf{M}_\lambda^J;\mathsf{k}} : \mathsf{k} \in \mathbb{Z}^d\} \cup \{\psi^{j,\vec{\ell}}_{\mathsf{S}_{-\vec{\ell}}\mathsf{A}_\lambda^j \mathsf{E}_n;\mathsf{k}} : \mathsf{k} \in \mathbb{Z}^d, n = 1, \dots, d, |\vec{\ell}| \leqslant \vec{\ell}_{\lambda^j}\}_{j=J}^\infty. \tag{2.15}$$

Similar to [19, Theorem 4], we have the following result showing that $\mathsf{AS}(\varphi; \{\Psi_j\}_{j=J}^\infty)$ is an affine shear tight frame containing an affine shear subsystem generated from one single generator. The proof follows directly from Theorem 2.1 (see section 5).

Theorem 2.2. *Let $\lambda > 1$, $0 < t \leqslant 1$, $0 < \varepsilon \leqslant 1/2$, and $0 < \rho \leqslant \frac{1}{1+2\varepsilon}$. Let $\mathsf{AS}(\varphi; \{\Psi_j\}_{j=J}^\infty)$ be defined as in (2.15) with $\widehat{\varphi} = \otimes^d \boldsymbol{\alpha}_{\lambda,t,\rho}$ and $\psi^{j,\vec{\ell}}$ be given by (2.14). Then $\mathsf{AS}(\varphi; \{\Psi_j\}_{j=J}^\infty)$*

is an affine shear tight frame for $L_2(\mathbb{R}^d)$ *for all* $J \geqslant 0$. *All elements in* $\mathsf{AS}(\varphi; \{\Psi_j\}_{j=J}^\infty)$ *are functions in* $C^\infty(\mathbb{R}^d)$. *Moreover, we have*

$$\{\psi_{\mathsf{S}^{-\vec{\ell}}\mathsf{A}_\lambda^j \mathsf{E}_n;\mathsf{k}} : \mathsf{k} \in \mathbb{Z}^d, n = 1, \dots, d, |\vec{\ell}| \leqslant \vec{r}_j\}_{j=J}^\infty \subseteq \mathsf{AS}(\varphi; \{\Psi_j\}_{j=J}^\infty), \tag{2.16}$$

where $\widehat{\psi}(\xi) := \boldsymbol{\beta}_{\lambda,t,\rho}(\xi_1) \prod_{n=2}^d \boldsymbol{\gamma}_\varepsilon(\xi_n/\xi_1)$, $\xi = (\xi_1, \dots, \xi_d) \in \mathbb{R}^d$, *and* $\vec{r}_j := (r_j, \dots, r_j) \in \mathbb{Z}^{d-1}$ *with* $r_j := \lfloor \lambda^{j-2}(1-t)\rho - (1/2+\varepsilon) \rfloor$.

Note that when $\vec{\ell} = -\vec{r}_j$, the support of $\widehat{\psi}(\mathsf{S}_{\vec{\ell}}\mathsf{B}_\lambda^j \xi) = \boldsymbol{\beta}_{\lambda,t,\rho}(\lambda^{-2j}\xi_1) \prod_{n=2}^d \gamma_\varepsilon(\lambda^j \xi_n/\xi_1 - r_j)$, $\xi = (\xi_1, \dots, \xi_d) \in \mathbb{R}^d$ satisfies $\xi_n/\xi_1 \leqslant \lambda^{-j}(r_j + 1/2 + \varepsilon) \leqslant \lambda^{-j}(\lfloor \lambda^{j-2}(1-t)\rho - 1/2 - \varepsilon \rfloor + 1/2 + \varepsilon) \leqslant \lambda^{-2}(1-t)\rho$. Hence, by the symmetry property of $\boldsymbol{\Gamma}_j$, we see that the shear subsystem in (2.16) of $\mathsf{AS}(\varphi; \{\Psi_j\}_{j=0}^\infty)$ is inside the cone area $\{\xi \in \mathbb{R}^d : \max\{|\xi_n/\xi_1| : n = 2, \dots, d\} \leqslant \lambda^{-2}(1-t)\rho\}$ in the frequency domain.

2.3. Smooth affine wavelet tight frames and their subsampling systems. As in the definition (2.2), an affine shear system is generated by three operations: anisotropic dilation, shear, and translation. On the other hand, wavelet/framelet systems are generated by two operations: dilation and translation. We next introduce a special type of directional wavelet/framelet system: affine wavelet systems. We shall see that affine shear tight frames and affine wavelet tight frames are actually connected to each other in the sense that an affine shear tight frame can be obtained from an affine wavelet tight frame from subsampling, which implies that an affine shear tight frame has an underlying directional filter bank.

Let φ^j and

$$\mathring{\Psi}_j := \{\mathring{\psi}^{j,\vec{\ell}} : |\vec{\ell}| \leqslant \vec{s}_j\}, \ \vec{\ell} = (\ell_2, \dots, \ell_d) \in \mathbb{Z}^{d-1}, \vec{s}_j = (s_{j,2}, \dots, s_{j,d}) \in \mathbb{N}_0^{d-1} \tag{2.17}$$

be generators in $L_2(\mathbb{R}^d)$. We use a fixed dilation matrix M_λ for all generators φ^j and $\mathring{\psi}^{j,\ell}$. The (quasi-stationary) *d-dimensional affine wavelet system* (starting at scale $J \in \mathbb{Z}$) is then defined to be

$$\mathsf{WS}(\varphi^J; \{\mathring{\Psi}_j\}_{j=J}^\infty) = \{\varphi^J_{\mathsf{M}_\lambda^J;\mathsf{k}} : \mathsf{k} \in \mathbb{Z}^d\} \cup \{\mathring{\psi}^{j,\vec{\ell}}_{\mathsf{M}_\lambda^j \mathsf{E}_n;\mathsf{k}} : \mathsf{k} \in \mathbb{Z}^d, n = 1, \dots, d, |\vec{\ell}| \leqslant \vec{s}_j\}_{j=J}^\infty. \tag{2.18}$$

We say that $\mathsf{WS}(\varphi^J; \{\mathring{\Psi}_j\}_{j=J}^\infty)$ is an *affine wavelet tight frame* for $L_2(\mathbb{R}^d)$ if

$$\|f\|_2^2 = \sum_{\mathsf{k} \in \mathbb{Z}^2} |\langle f, \varphi^J_{\mathsf{M}_\lambda^J;\mathsf{k}} \rangle|^2 + \sum_{j=J}^\infty \sum_{n=1}^d \sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j} \sum_{\mathsf{k} \in \mathbb{Z}^d} |\langle f, \mathring{\psi}^{j,\vec{\ell}}_{\mathsf{M}_\lambda^j;\mathsf{k}} \rangle|^2 \quad \forall f \in L_2(\mathbb{R}^d). \tag{2.19}$$

Comparing $\psi^{j,\vec{\ell}}_{\mathsf{S}^{-\vec{\ell}}\mathsf{A}_\lambda^j;\mathsf{k}}$ and $\mathring{\psi}^{j,\vec{\ell}}_{\mathsf{M}_\lambda^j;\mathsf{k}}$, and in view of the relation $\mathsf{M}_\lambda = \mathsf{A}_\lambda \mathsf{D}_\lambda$, we have the following result connecting affine shear tight frames with affine wavelet tight frames.

Theorem 2.3. *Let* $J_0 \in \mathbb{Z}$ *and* φ^j, Ψ_j, *and* $\mathring{\Psi}_j$ *be generators at scale* j *defined as in* (2.1) *and* (2.17) *for* $j \geqslant J_0$. *Suppose*

$$\mathring{\psi}^{j,\vec{\ell}} = \lambda^{-(d-1)j} \psi^{j,\vec{\ell}}(\mathsf{S}^{-\vec{\ell}} \mathsf{D}_\lambda^{-j} \cdot), \qquad |\ell| \leqslant \vec{s}_j, \tag{2.20}$$

$$\widehat{h}(\xi)\widehat{h}(\xi + 2\pi\mathsf{k}) = 0 \quad a.e.\ \xi \in \mathbb{R}^d \ \forall \mathsf{k} \in \mathbb{Z}^d \backslash \{0\}, \forall h \in \{\{\varphi^j\} \cup \Psi_j \cup \mathring{\Psi}_j\}_{j=J_0}^\infty. \tag{2.21}$$

Then $\mathsf{AS}(\varphi^J;\{\Psi_j\}_{j=J}^\infty)$ *defined as in* (2.2) *is an affine shear tight frame for* $L_2(\mathbb{R}^2)$ *for every* $J \geqslant J_0$ *if and only if* $\mathsf{WS}(\varphi^J;\{\mathring{\Psi}_j\}_{j=J}^\infty)$ *defined as in* (2.18) *is an affine wavelet tight frame for* $L_2(\mathbb{R}^2)$ *for every* $J \geqslant J_0$.

Proof. Since (2.21) is satisfied, by Theorem 2.1, $\mathsf{AS}_J(\varphi^J;\{\Psi_j\}_{j=J}^\infty)$ is an affine shear tight frame for $L_2(\mathbb{R}^d)$ for every $J \geqslant J_0$ if and only if (2.5) and (2.6) are satisfied for $J \geqslant J_0$. Observe that (2.20) implies $\widehat{\mathring{\psi}^{j,\vec{\ell}}} = \widehat{\psi^{j,\vec{\ell}}}(\mathsf{S}_{\vec{\ell}}\mathsf{D}_\lambda^j\cdot)$. Therefore, by $\mathsf{B}_\lambda^j\mathsf{M}_\lambda^j = \mathsf{D}_\lambda^j$ and $\mathsf{E}_n\mathsf{M}_\lambda^j = \mathsf{M}_\lambda^j\mathsf{E}_n$, we see that (2.5) is equivalent to

$$|\widehat{\varphi^{j+1}}(\mathsf{N}_\lambda\xi)|^2 - |\widehat{\varphi^{j+1}}(\xi)|^2 = \sum_{n=1}^d \sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j} |\widehat{\psi^{j,\vec{\ell}}}(\mathsf{S}_{\vec{\ell}}\mathsf{B}_\lambda^j\mathsf{E}_n\mathsf{M}_\lambda^j\xi)|^2 = \sum_{n=1}^d \sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j} |\widehat{\mathring{\psi}^{j,\ell}}(\mathsf{E}_n\xi)|^2 \text{ a.e. } \xi \in \mathbb{R}^d.$$

The claim now follows directly from [14, Corollary 18] and Theorem 2.1. ∎

We immediately have the following corollary.

Corollary 2.4. *Let* $\mathsf{AS}(\varphi;\{\Psi_j\}_{j=J}^\infty)$, $J \geqslant 0$, *be the sequence of affine shear tight frames given in Theorem* 2.2. *Let* $\mathring{\Psi}_j = \{\mathring{\psi}^{j,\vec{\ell}} : |\vec{\ell}| \leqslant \vec{\ell}_{\lambda^j}\}$ *be defined from* Ψ_j *as in* (2.20) *and* $\mathsf{WS}(\varphi;\{\mathring{\Psi}_j\}_{j=J}^\infty)$ *be defined in* (2.18) *with* $\varphi^j \equiv \varphi$. *Then* $\mathsf{WS}(\varphi;\{\mathring{\Psi}_j\}_{j=J}^\infty)$ *is an affine wavelet tight frame for* $L_2(\mathbb{R}^d)$ *for all* $J \geqslant 0$ *and*

$$\psi^{j,\vec{\ell}}_{\mathsf{S}^{-\vec{\ell}}\mathsf{A}_\lambda^j\mathsf{E}_n;\mathsf{k}} = \lambda^{(d-1)j/2}\mathring{\psi}^{j,\vec{\ell}}_{\mathsf{M}_\lambda^j\mathsf{E}_n;\mathsf{D}_\lambda^j\mathsf{S}^{\vec{\ell}}\mathsf{k}}, \quad |\vec{\ell}| \leqslant \vec{s}_j, n = 1,\dots,d, j \geqslant 0. \tag{2.22}$$

Proof. By (2.20), it is straightforward to check that (2.22) holds. By (2.20) and the definition of $\psi^{j,\vec{\ell}}$ in (2.14), we also have

$$\widehat{\mathring{\psi}^{j,\vec{\ell}}}(\xi) = \boldsymbol{\omega}_{\lambda,t,\rho}(\xi)\frac{\prod_{n=2}^d \boldsymbol{\gamma}_\varepsilon(\lambda^j\xi_n/\xi_1 + \ell_n)}{\boldsymbol{\Gamma}_j(\xi)}, \quad |\vec{\ell}| \leqslant \vec{\ell}_{\lambda^j}. \tag{2.23}$$

Now, it is easy to check that $\operatorname{supp}\widehat{\mathring{\psi}^{j,\vec{\ell}}} \subseteq [-\rho\pi,\rho\pi]^d$ for $0 < \rho \leqslant 1$. We immediately have $\widehat{\mathring{\psi}^{j,\vec{\ell}}}(\cdot)\widehat{\mathring{\psi}^{j,\vec{\ell}}}(\cdot + 2\pi\mathsf{k}) = 0$ for $\mathsf{k} \in \mathbb{Z}^d\backslash\{0\}$. Consequently, (2.21) holds. Now the conclusion that $\mathsf{WS}(\varphi;\{\mathring{\Psi}_j\}_{j=J}^\infty)$ is an affine wavelet tight frame for $L_2(\mathbb{R}^d)$ follows from Theorem 2.3. ∎

When λ is an integer, we have $\mathsf{D}_\lambda^j\mathsf{S}^{\vec{\ell}}\mathbb{Z}^d \subseteq \mathbb{Z}^d$. Consequently, if (2.20) holds, then $\widehat{h}(\cdot)\widehat{h}(\cdot + 2\pi\mathsf{k}) = 0$ for all $\mathsf{k} \in \mathbb{Z}^d\backslash\{0\}$ and for all $h \in \{\varphi\} \cup \Psi_j$ implies $\widehat{h}(\cdot)\widehat{h}(\cdot + 2\pi\mathsf{k}) = 0$ for all $\mathsf{k} \in \mathbb{Z}^d\backslash\{0\}$ and for all $h \in \{\varphi\} \cup \mathring{\Psi}_j$. Moreover, (2.22) shows that when λ is an integer, the affine shear tight frame $\mathsf{AS}(\varphi;\{\Psi_j\}_{j=J}^\infty)$ is indeed a subsystem of the affine wavelet tight frame $\mathsf{WS}(\varphi;\{\mathring{\Psi}_j\}_{j=J}^\infty)$ through subsampling. It is well-known that for an affine wavelet tight frame, it has an associated filter bank structure. We next present the filter bank structure for our affine wavelet tight frames.

By the definition of φ and $\boldsymbol{\omega}_{\lambda,t,\rho}$, we have smooth scaling and wavelet masks $\widehat{a}, \widehat{b} \in C^\infty(\mathbb{T}^d)$ for $\widehat{\varphi}(\lambda^2\cdot) = \widehat{a}\widehat{\varphi}$ and $\boldsymbol{\omega}_{\lambda,t,\rho}(\lambda^2\cdot) = \widehat{b}\widehat{\varphi}$ as follows:

$$\widehat{a}(\xi) = [\otimes^d \boldsymbol{\mu}_{\lambda,t,\rho}](\xi) \quad \text{and} \quad \widehat{b}(\xi) = (\mathbf{g}(\xi) - |\widehat{a}(\xi)|^2)^{1/2}, \quad \xi \in \mathbb{T}^d, \tag{2.24}$$

where $\boldsymbol{\mu}_{\lambda,t,\rho}$ is given in (2.10) and $\mathbf{g} \in C^\infty(\mathbb{T}^d)$ satisfying $\mathbf{g} \equiv 1$ on the support of $\widehat{\varphi}$. Note that $\operatorname{supp} \widehat{\mathring{\psi}^{j,\ell}}(\mathsf{M}_\lambda \cdot) \subseteq \operatorname{supp} \widehat{\varphi}$. We can define $\widehat{b^{j,\vec{\ell}}}$ for $\widehat{\mathring{\psi}^{j,\vec{\ell}}}$, $j \geqslant 0$, as follows:

$$\widehat{b^{j,\vec{\ell}}}(\xi) := \widehat{b}(\xi) \frac{\vec{\boldsymbol{\gamma}}^{j,\vec{\ell}}(\xi)}{\sqrt{\boldsymbol{\Gamma}_j(\xi)}} = \widehat{b}(\xi) \frac{\prod_{n=2}^d \boldsymbol{\gamma}_\varepsilon(\lambda^j \xi_n/\xi_1 + \ell_n)}{\sqrt{\boldsymbol{\Gamma}_j(\xi)}}, \quad |\vec{\ell}| \leqslant \vec{\ell}_{\lambda^j}. \tag{2.25}$$

Then, we have the following result.

Theorem 2.5. *Let $\lambda > 1$ be an integer. Choose $0 < \varepsilon \leqslant 1/2$, $0 < t \leqslant 1$, and $0 < \rho \leqslant \frac{1}{1+2\varepsilon}$. Let $\mathsf{WS}_J(\varphi; \{\mathring{\Psi}_j\}_{j=J}^\infty)$, $J \geqslant 0$, be defined as in Corollary 2.4 and let $a, b^{j,\vec{\ell}}$ be defined as in (2.24) and (2.25), respectively. Then $\widehat{a}, \widehat{b^{j,\vec{\ell}}} \in C^\infty(\mathbb{T}^d)$ are such that*

$$\widehat{\varphi}(\mathsf{M}_\lambda \xi) = \widehat{a}(\xi)\widehat{\varphi}(\xi) \quad \text{and} \quad \widehat{\mathring{\psi}^{j,\vec{\ell}}}(\mathsf{M}_\lambda \xi) = \widehat{b^{j,\vec{\ell}}}(\xi)\widehat{\varphi}(\xi), \quad \xi \in \mathbb{R}^d. \tag{2.26}$$

Moreover, for each $j \geqslant 0$, $\{a; b^{j,\vec{\ell}}(\mathsf{E}_n \cdot) : |\vec{\ell}| \leqslant \vec{\ell}_{\lambda^j}, n = 1, \ldots, d\}$ is an affine wavelet filter bank having the perfect reconstruction (PR) property:

$$|\widehat{a}(\xi)|^2 + \sum_{n=1}^d \sum_{\vec{\ell}=-\vec{\ell}_{\lambda^j}}^{\vec{\ell}_{\lambda^j}} |\widehat{b^{j,\vec{\ell}}}(\mathsf{E}_n \xi)|^2 = 1 \quad \text{a.e. } \xi \in \operatorname{supp} \widehat{\varphi}, \tag{2.27}$$

$$\widehat{a}(\xi)\widehat{a}(\xi + 2\pi\omega) = 0,\ \widehat{b^{j,\vec{\ell}}}(\mathsf{E}_n \xi)\widehat{b^{j,\ell}}(\mathsf{E}_n \xi + 2\pi\omega) = 0 \text{ a.e. } \xi \in \operatorname{supp} \widehat{\varphi} \cap (\operatorname{supp} \widehat{\varphi} - 2\pi\omega) \tag{2.28}$$

for all $|\vec{\ell}| \leqslant \vec{\ell}_{\lambda^j}$, $\omega \in \Omega_{\mathsf{M}_\lambda} \backslash \{0\}$, and $n = 1, \ldots, d$, where $\Omega_{\mathsf{M}_\lambda} = [\mathsf{M}_\lambda^{-\mathsf{T}} \mathbb{Z}^d] \cap [0,1)^d$.

Proof. By the definition of $\widehat{a}$ and $\widehat{b}$, we have

$$\widehat{b}(\xi) = \Big(1 - \Big|\frac{\widehat{\varphi}(\lambda^2 \xi)}{\widehat{\varphi}(\xi)}\Big|^2\Big)^{1/2} = \Big(\frac{|\widehat{\varphi}(\xi)|^2 - |\widehat{\varphi}(\lambda^2 \xi)|^2}{|\widehat{\varphi}(\xi)|^2}\Big)^{1/2} = \frac{\boldsymbol{\omega}_{\lambda,t,\rho}(\lambda^2 \xi)}{\varphi(\xi)}, \quad \xi \in \operatorname{supp} \widehat{\varphi}.$$

In view of $b^{j,\vec{\ell}}$ in (2.25) and $\mathring{\psi}^{j,\vec{\ell}}$ in (2.23), we conclude that (2.26) holds. Moreover, by the definition of $\boldsymbol{\Gamma}_j$ in (2.13), we have $\sum_{n=1}^d \sum_{\vec{\ell}=-\vec{\ell}_{\lambda^j}}^{\vec{\ell}_{\lambda^j}} |\widehat{b^{j,\vec{\ell}}}(\mathsf{E}_n \xi)|^2 = |\widehat{b}(\xi)|^2$ for $\xi \in \operatorname{supp} \widehat{\varphi}$. Hence, (2.27) holds. Equation (2.28) directly follows from (2.26) and (2.21) for $h \in \{\varphi\} \cup \mathring{\Psi}_j$. ∎

3. Digitization of smooth affine shear tight frames. In this section, we discuss the digitization of our smooth affine shear tight frames: digital affine shear filter banks and digital affine shear transforms. Our digitization is based on the affine wavelet filter banks in Theorem 2.5 associated with our (quasi-stationary) smooth affine shear tight frames. We first construct digital affine shear filter banks induced from affine wavelet filter banks and then detail the implementation of the forward and backward digital affine shear transforms based on the digital affine shear filter banks. In a nutshell, we are going to construct a sequence $\{a; b^{j,\vec{\ell}}(\mathsf{E}_n \cdot) : n = 1, \ldots, d, |\vec{\ell}| \leqslant \vec{s}_j\}_{j=0}^J$ of digital affine shear filter banks for the decomposition and reconstruction of data. With such a sequence of filter banks, we can define φ to be $\widehat{\varphi} = \lim_{J\to\infty} \prod_{j=0}^J \widehat{a}(\mathsf{N}_\lambda^{j+1} \cdot)$, which implies $\widehat{\varphi}(\mathsf{M}_\lambda \cdot) = \widehat{a}\widehat{\varphi}$, and $\mathring{\psi}^{j,\vec{\ell}}$ to be $\widehat{\mathring{\psi}^{j,\vec{\ell}}}(\mathsf{M}_\lambda^{\mathsf{T}} \cdot) = \widehat{b^{j,\vec{\ell}}}\,\widehat{\varphi}$.

Let $\mathring{\Psi}_j := \{\mathring{\psi}^{j,\vec{\ell}} : |\vec{\ell}| \leqslant \vec{s}_j\}$. Define $\mathsf{WS}_J(\varphi; \{\mathring{\Psi}_j\}_{j=J}^\infty)$ as in (2.18). Then, we have a sequence of affine wavelet tight frames $\mathsf{WS}_J(\varphi; \{\mathring{\Psi}_j\}_{j=J}^\infty)$, $J = 0, 1, 2, \ldots$, and by (2.20), we can define $\psi^{j,\vec{\ell}}$ to be $\psi^{j,\vec{\ell}} = \lambda^{(d-1)j/2}\mathring{\psi}^{j,\vec{\ell}}(\mathsf{D}_\lambda^j \mathsf{S}^{\vec{\ell}}\cdot)$. Let $\Psi_j := \{\psi^{j,\vec{\ell}} : |\vec{\ell}| \leqslant \vec{s}_j\}$. Then we also have a sequence of affine shear tight frames $\mathsf{AS}_J(\varphi, \{\Psi_j\}_{j=J}^\infty)$ as defined in (2.2).

3.1. Digital affine shear filter banks. Recall the definition of $\boldsymbol{\nu}_{[c,\epsilon]}$ in (2.8). In practice, we only need $\boldsymbol{\nu}_{[c,\epsilon]}$ to have certain regularity, for example, $\boldsymbol{\nu}_{[c,\epsilon]} \in C_c^m(\mathbb{R})$ for some $m \in \mathbb{N}_0$. Then the corresponding ν can be constructed from a polynomial. For example, let $P_m(x) := \left(\frac{1+x}{2}\right)^m \sum_{j=0}^{m-1} \binom{m-1+j}{j} \left(\frac{1-x}{2}\right)^j$. It is easy to show that $P_m(x) + P_m(-x) \equiv 1$ for all $x \in \mathbb{R}$. Define $\nu(x) = \sin(\frac{\pi}{2} P_{m+1}(x))$ or $\nu(x) = \sqrt{P_{m+1}(x)}$ for $x \in (-1, 1)$. Then $\boldsymbol{\nu}_{[c,\epsilon]}$ defined in (2.8) is the desired function in $C_c^m(\mathbb{R})$. In view of $\widehat{a}$ in (2.24), we directly define an inner function $\widehat{a} \in C(\mathbb{T}^d)$ as a low-pass filter by

$$\widehat{a}(\xi) := [\otimes^d \boldsymbol{\nu}_{[c_0,\epsilon_0]}](\xi) = \prod_{n=1}^d \boldsymbol{\nu}_{[c_0,\epsilon_0]}(\xi_n), \quad \xi = (\xi_1, \ldots, \xi_d) \in \mathbb{T}^d, \tag{3.1}$$

for some parameters $c_0 > 0$ and $\epsilon_0 > 0$ satisfying $c_0 + \epsilon_0 \leqslant \pi/2$ (for downsampling by 2).

Similarly, we define an outer function $\mathbf{g} \in C(\mathbb{R}^d)$ by $\mathbf{g} := \otimes^d \boldsymbol{\nu}_{[c_1,\epsilon_1]}$, where $c_1 = \pi$ and ϵ_1 satisfies $c_1 + \epsilon_1 - (c_0 - \epsilon_0) \leqslant \pi$ for the purpose of downsampling at least by 2 for high-pass filter coefficients. Thanks to the property of $\boldsymbol{\nu}_{[c,\epsilon]}$, one can show that

$$\sum_{\mathsf{k} \in \mathbb{Z}^d} |\mathbf{g}(\xi + 2\pi\mathsf{k})|^2 = 1 \qquad \forall \xi \in \mathbb{T}^d. \tag{3.2}$$

Now we can define an "isotropic" compactly supported function $b \in C_c(\mathbb{R}^d)$ by

$$\widehat{b}(\xi) := \begin{cases} \sqrt{|\mathbf{g}(\xi)|^2 - |\widehat{a}(\xi)|^2} & \text{if } \xi \in \operatorname{supp} \widehat{\mathbf{g}}, \\ 0 & \text{otherwise.} \end{cases} \tag{3.3}$$

Note that both $\mathbf{g}$ and b are not filters and both $\mathbf{g}$ and $\widehat{b}$ are supported on $[-\pi - \epsilon_1, \pi + \epsilon_1]^d$.

Now, we apply the splitting technique to $\widehat{b}$ for the construction of high-pass filters $b^{j,\vec{\ell}}$. In practice, at scale $j \geqslant 0$, instead of using 2^j to determine the total number of directions, we use 2^{k_j} for some nonnegative integer because the resolution of data could be limited. Define $\vec{s}_j := (2^{k_j}, \ldots, 2^{k_j}) \in \mathbb{Z}^{d-1}$. Similar to the definition of normalization function $\mathbf{\Gamma}_j$ in (2.13) with $\lambda = 2$, we define

$$\vec{\gamma}^{k_j,\vec{\ell}}(\xi) := \prod_{n=2}^d \boldsymbol{\gamma}_\varepsilon(2^{k_j}\xi_n/\xi_1 + \ell_n) \text{ and } \mathbf{\Gamma}_{k_j}(\xi) := \sum_{n=1}^d \sum_{\vec{\ell}=-\vec{s}_j}^{s_j} |\vec{\gamma}^{k_j,\vec{\ell}}(\mathsf{E}_n\xi)|^2, \quad \xi \neq 0, \tag{3.4}$$

with $\boldsymbol{\gamma}_\varepsilon$ being defined in (2.9) for some $0 \leqslant \varepsilon \leqslant \frac{\pi}{c_1+\epsilon_1} - 1/2$, and $\vec{\gamma}^{k_j,\vec{\ell}}(0) = \mathbf{\Gamma}_{k_j}(0) := 0$. To guarantee smoothness of boundary, we further split $\vec{\gamma}^{k_j,\vec{\ell}}(\xi)$ into positive part $\vec{\gamma}^{k_j,\vec{\ell},+}$ and negative part $\vec{\gamma}^{k_j,\vec{\ell},-}$ of the ξ_1-axis. Define $\vec{\gamma}^{k_j,\vec{\ell},\pm}(\xi) := \vec{\gamma}^{k_j,\vec{\ell}}(\xi)\chi_{\{\pm\xi_1>0\}}$. Note that

$\widehat{b}(\xi)\frac{\vec{\gamma}^{k_j,\vec{\ell},\pm}(\xi)}{\sqrt{\mathbf{\Gamma}_{k_j}(\xi)}}$ are not $2\pi\mathbb{Z}^d$-periodic functions. We define $b^{j,\vec{\ell},\pm}$ to be the $2\pi\mathbb{Z}^d$-periodization of $\widehat{b}(\xi)\frac{\vec{\gamma}^{k_j,\vec{\ell},\pm}(\xi)}{\sqrt{\mathbf{\Gamma}_{k_j}(\xi)}}$ as follows:

$$\widehat{b^{j,\vec{\ell},\pm}}(\xi) := \sum_{\mathsf{k}\in\mathbb{Z}^d} \widehat{b}(\xi+2\pi\mathsf{k})\frac{\vec{\gamma}^{k_j,\vec{\ell},\pm}(\xi+2\pi\mathsf{k})}{\sqrt{\mathbf{\Gamma}_{k_j}(\xi+2\pi\mathsf{k})}}, \quad \xi\in\mathbb{T}^d. \tag{3.5}$$

The total number of high-pass filters $b^{j,\vec{\ell},+}$ and $b^{j,\vec{\ell},-}$ at this scale j is $2(2^{k_j+1}+1)^{d-1}$. Each filter of $\widehat{b^{j,\vec{\ell},\pm}}$ is a $2\pi\mathbb{Z}^d$-periodic function on $\mathbb{T}^d$.

Given a sequence of nonnegative integers k_j, $j=0,\dots,J-1$, for some fixed integer $J\geqslant 0$ with respect to the finest scale, let $\mathsf{M} := 2\mathsf{I}_d$, $\mathsf{A}_j := \operatorname{diag}(2,2^{k_j}\mathsf{I}_{d-1})$, and $\mathsf{A}_{j,n} := \mathsf{E}_n\mathsf{A}_j$ for $n=1,\dots,d$. From above, we can obtain a sequence of filter banks:

$$\mathsf{DAS}(\{\mathcal{B}_j\}_{j=0}^{J-1}) \text{ with } \mathcal{B}_j := \{a\downarrow\mathsf{M},\ b^{j,\vec{\ell},\pm}(\mathsf{E}_n\cdot)\downarrow\mathsf{A}_{j,n} : |\vec{\ell}|\leqslant\vec{s}_j, n=1,\dots,d\} \tag{3.6}$$

for $j=0,\dots,J-1$. Here M in $a\downarrow\mathsf{M}$ indicates the downsampling matrix for filtered coefficients with respect to the low-pass filter a and $\mathsf{A}_{j,n}$ in $b^{j,\vec{\ell},\pm}(\mathsf{E}_n\cdot)\downarrow\mathsf{A}_{j,n}$ indicates the downsampling matrix for filtered coefficients with respect to the high-pass filter $b^{j,\vec{\ell}}(\mathsf{E}_n\cdot)$. Now, in view of (3.2) and (3.4), we have the following result.

Theorem 3.1. *Let $k_j\in\mathbb{N}_0$, $\vec{s}_j=(2^{k_j},\dots,2^{k_j})\in\mathbb{N}^{d-1}$, and $a, b^{j,\vec{\ell},\pm}$, be defined as in (3.1), (3.5), respectively, with parameters $c_0,\epsilon_0,c_1=\pi,\epsilon_1,\varepsilon$ such that $0<\epsilon_0\leqslant c_0$, $c_0+\epsilon_0\leqslant\pi/2$, $0<\epsilon_1\leqslant c_1$, $c_1+\epsilon_1-(c_0-\epsilon_0)\leqslant\pi$, and $0<\varepsilon\leqslant\frac{\pi}{c_1+\epsilon_1}-1/2$. Then $\{a, b^{j,\vec{\ell},\pm}(\mathsf{E}_n\cdot) : n=1,\dots,d, |\vec{\ell}|\leqslant\vec{s}_j\}$ forms a digital affine shear filter bank with the PR property:*

$$|\widehat{a}(\xi)|^2+\sum_{n=1}^{d}\sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j}\big(|\widehat{b^{j,\vec{\ell},+}}(\mathsf{E}_n\xi)|^2+|\widehat{b^{j,\vec{\ell},-}}(\mathsf{E}_n\xi)|^2\big)=1, \tag{3.7}$$

$$\widehat{a}(\xi)\widehat{a}(\xi+2\pi\omega)=0, \quad \widehat{b^{j,\vec{\ell},\pm}}(\mathsf{E}_n\xi)\widehat{b^{j,\vec{\ell},\pm}}(\mathsf{E}_n\xi+2\pi\omega_1)=0, \tag{3.8}$$

for all $\xi\in\mathbb{T}^d$, $|\vec{\ell}|\leqslant\vec{s}_j$, $n=1,\dots,d$, $\omega\in[\mathsf{M}^{-\mathsf{T}}\mathbb{Z}^d]\cap[0,1)^d\backslash\{0\}$, and $\omega_1\in[\mathsf{A}_j^{-\mathsf{T}}\mathbb{Z}^d]\cap[0,1)^d\backslash\{0\}$.

Proof. For $\vec{\ell}=0$, by that $0<\varepsilon\leqslant\frac{\pi}{c_1+\epsilon_1}-1/2$ and the definition of $\boldsymbol{\gamma}_\varepsilon$, we have

$$\begin{aligned}\operatorname{supp}\vec{\boldsymbol{\gamma}}^{k_j,0,+} &\subseteq \{\xi\in\mathbb{R}^d : \xi_1\in[c_0-\epsilon_0,c_1+\epsilon_1], |\xi_n|\leqslant 2^{-k_j}(1/2+\varepsilon)|\xi_1|, n=2,\dots,d\}\\ &\subseteq [c_0-\epsilon_0,c_1+\epsilon_1]\times[-2^{-k_j}\pi,2^{-k_j}\pi]^{d-1}.\end{aligned}$$

Similarly, $\operatorname{supp}\vec{\boldsymbol{\gamma}}^{k_j,0,-}\subseteq[-c_1-\varepsilon_1,-c_0+\varepsilon_0]\times[-2^{-k_j}\pi,2^{-k_j}\pi]^{d-1}$. Note that $\vec{\boldsymbol{\gamma}}^{k_j,\vec{\ell},\pm}=\vec{\boldsymbol{\gamma}}^{k_j,0,\pm}(\mathsf{S}_{\vec{\ell}}\cdot)$. By the support of $\vec{\boldsymbol{\gamma}}^{k_j,0,\pm}$, it is easy to show that $\vec{\boldsymbol{\gamma}}^{k_j,\vec{\ell},\pm}(\xi)\vec{\boldsymbol{\gamma}}^{k_j,\vec{\ell},\pm}(\xi+2\pi\mathsf{k})=0$ for $\xi\in\mathbb{R}^d$ and $\mathsf{k}\in\mathbb{Z}^d\backslash\{0\}$. Hence

$$|\widehat{b^{j,\vec{\ell},\pm}}(\xi)|^2=\sum_{\mathsf{k}\in\mathbb{Z}^d}|\widehat{b}(\xi+2\pi\mathsf{k})|^2\frac{|\vec{\boldsymbol{\gamma}}^{k_j,\vec{\ell},\pm}(\xi+2\pi\mathsf{k})|^2}{\mathbf{\Gamma}_{k_j}(\xi+2\pi\mathsf{k})}.$$

Consequently, for $\xi \in \mathbb{T}^d$, we have

$$\begin{aligned}
&\sum_{n=1}^{d}\sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j}\big(|\widehat{b^{j,\vec{\ell},+}}(\mathsf{E}_n\xi)|^2+|\widehat{b^{j,\vec{\ell},-}}(\mathsf{E}_n\xi)|^2\big)\\
&=\sum_{n=1}^{d}\sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j}\sum_{\mathsf{k}\in\mathbb{Z}^d}|\widehat{b}(\mathsf{E}_n\xi+2\pi\mathsf{k})|^2\frac{|\vec{\gamma}^{k_j,\vec{\ell},+}(\mathsf{E}_n\xi+2\pi\mathsf{k})|^2+|\vec{\gamma}^{k_j,\vec{\ell},-}(\mathsf{E}_n\xi+2\pi\mathsf{k})|^2}{\mathbf{\Gamma}_{k_j}(\mathsf{E}_n\xi+2\pi\mathsf{k})}\\
&=\sum_{\mathsf{k}\in\mathbb{Z}^d}|\widehat{b}(\xi+2\pi\mathsf{k})|^2\sum_{n=1}^{d}\sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j}\frac{|\vec{\gamma}^{k_j,\vec{\ell}}(\mathsf{E}_n(\xi+2\pi\mathsf{k}))|^2}{\mathbf{\Gamma}_{k_j}(\xi+2\pi\mathsf{k})}=\sum_{\mathsf{k}\in\mathbb{Z}^d}|\widehat{b}(\xi+2\pi\mathsf{k})|^2\\
&=\sum_{\mathsf{k}\in\mathbb{Z}^d}|\mathbf{g}(\xi+2\pi\mathsf{k})|^2-|\widehat{a}(\xi)|^2=1-|\widehat{a}(\xi)|^2.
\end{aligned}$$

Then, (3.7) holds.

Since $\operatorname{supp}\widehat{a}=\operatorname{supp}\otimes^d\boldsymbol{\nu}_{[c_0,\varepsilon_0]}=[-c_0-\varepsilon_0,c_0+\varepsilon_0]^d\subseteq[-\pi/2,\pi/2]^d$ and $[\mathsf{M}^{-\mathsf{T}}\mathbb{Z}^d]\cap[0,1)^d=\{\frac{1}{2}(m_1,\dots,m_d):m_n\in\{0,1\},n=1,\dots,d\}$, it is easy to show that for $\omega=\frac{1}{2}(m_1,\dots,m_d)\in[\mathsf{M}^{-\mathsf{T}}\mathbb{Z}^d]\cap[0,1)^d\backslash\{0\}$, we have $\widehat{a}(\xi)\widehat{a}(\xi+2\pi\omega)=\prod_{n=1}^{d}\boldsymbol{\nu}_{[c_0,\varepsilon_0]}(\xi_n)\boldsymbol{\nu}_{[c_0,\varepsilon_0]}(\xi_n+\pi m_n)=0$. Moreover, $[\mathsf{A}_j^{-\mathsf{T}}\mathbb{Z}^d]\cap[0,1)^d=\{(\frac{m_1}{2},\frac{m_2}{2^{k_j}},\dots,\frac{m_d}{2^{k_j}}):m_1\in\{0,1\},m_n\in\{0,1,\dots,2^{k_j}-1\},n=2,\dots,d\}$. Again, by the support of $\vec{\gamma}^{k_j,0,\pm}$ and that $\vec{\gamma}^{k_j,\vec{\ell},\pm}=\vec{\gamma}^{k_j,0,\pm}(\mathsf{S}_{\vec{\ell}}\,\cdot)$, for $\omega_1=(\frac{m_1}{2},\frac{m_2}{2^{k_j}},\dots,\frac{m_d}{2^{k_j}})\in[\mathsf{A}_j^{-\mathsf{T}}\mathbb{Z}^d]\cap[0,1)^d\backslash\{0\}$, we have

$$\vec{\gamma}^{k_j,\vec{\ell},\pm}(\xi)\vec{\gamma}^{k_j,\vec{\ell},\pm}(\xi+2\pi\omega_1)=\vec{\gamma}^{k_j,\vec{\ell},\pm}(\xi_1,\dots,\xi_d)\vec{\gamma}^{k_j,\vec{\ell},\pm}\Big(\xi_1+\frac{2\pi m_1}{2},\dots,\xi_d+\frac{2\pi m_d}{2^{k_d}}\Big)=0.$$

Consequently, (3.8) holds. ∎

By Theorem 3.1, for a sequence of nonnegative integers $k_0,\dots,k_{J-1}$, we can construct a sequence $\mathsf{DAS}(\{\mathcal{B}_j\}_{j=0}^{J-1})$ of *d-dimensional digital affine shear filter banks*. Note that the total number of high-pass filters in $\mathcal{B}_j$ is $2d(2^{k_j+1}+1)^{d-1}$.

3.2. Digital affine shear transforms. We next discuss the implementation of the forward transform (decomposition) and backward transform (reconstruction) of our digital affine shear filter banks. Our implementation is based on the discrete Fourier transform (DFT). We first discuss three main operations for a filter bank decomposition and reconstruction: convolution, downsampling, and upsampling.

For a sequence $u=\{u(\mathsf{k})\in\mathbb{C}:\mathsf{k}\in\mathbb{Z}^d\}\in l_0(\mathbb{Z}^d)$, we define its *Fourier series* $\widehat{u}:\mathbb{T}^d\to\mathbb{C}$ to be $\widehat{u}(\xi)=\sum_{\mathsf{k}\in\mathbb{Z}^d}u(\mathsf{k})e^{-i\mathsf{k}\cdot\xi}$, $\xi\in\mathbb{T}^d$. Without loss of generality and for simplicity of presentation, we shall assume our data live on the dyadic grids $\Lambda(\mathsf{K})$ for $\mathsf{K}=(\mathsf{K}_1,\dots,\mathsf{K}_d)\in\mathbb{N}^d$ defined by

$$\Lambda(\mathsf{K})=\Lambda(\mathsf{K}_1,\dots,\mathsf{K}_d):=([0,\dots,2^{\mathsf{K}_1}-1]\times\cdots\times[0,\dots,2^{\mathsf{K}_d}-1])\cap\mathbb{N}_0^d$$

and define its corresponding lattice in the frequency domain by

$$\widehat{\Lambda}(\mathsf{K}) := \frac{2\pi}{2^{\mathsf{K}_1+\cdots+\mathsf{K}_d}}\big(([-2^{\mathsf{K}_1-1},\dots,2^{\mathsf{K}_1-1}-1]\times\cdots\times[-2^{\mathsf{K}_d-1},\dots,2^{\mathsf{K}_d-1}-1])\cap\mathbb{Z}^d\big)\subseteq\mathbb{T}^d.$$

The (centered) DFT $\mathscr{F}: v\to \widehat{v}|_{\widehat{\Lambda}(\mathsf{K})}$ maps time-domain data $v:\Lambda(\mathsf{K})\to\mathbb{C}$ to $2\pi\mathbb{Z}^d$-periodic frequency domain data $\widehat{v}|_{\widehat{\Lambda}(\mathsf{K})}$, which is defined to be

$$\widehat{v}(\mathsf{n}) = [\mathscr{F}v](\mathsf{n}) := \sum_{\mathsf{k}\in\Lambda(\mathsf{K})} v(\mathsf{k})e^{-i\mathsf{n}\cdot\mathsf{k}},\quad \mathsf{n}\in\widehat{\Lambda}(\mathsf{K}).$$

The inverse (centered) DFT of $\widehat{v}|_{\widehat{\Lambda}(\mathsf{K})}$ is given by

$$[\mathscr{F}^{-1}\widehat{v}](\mathsf{k}) = \frac{1}{2^{\mathsf{K}_1+\cdots+\mathsf{K}_d}}\sum_{\mathsf{n}\in\widehat{\Lambda}(\mathsf{K})}\widehat{v}(\mathsf{n})e^{i\mathsf{n}\cdot\mathsf{k}},\quad \mathsf{k}\in\Lambda(\mathsf{K}).$$

The centered DFT and its inverse can be implemented by `fftn`, `ifftn`, and `fftshift` in MATLAB.

Given a filter u defined by its Fourier series $\widehat{u}:\mathbb{T}^d\to\mathbb{C}$ and data $v:\Lambda(\mathsf{K})\to\mathbb{C}$, the *circular convolution* $v\circledast u:\Lambda(\mathsf{K})\to\mathbb{C}$ is determined by DFT $\mathscr{F}(v\circledast u):\widehat{\Lambda}(\mathsf{K})\to\mathbb{C}$ as follows:

$$\widehat{[v\circledast u]}(\mathsf{n}) := \widehat{v}(\mathsf{n})\cdot\widehat{u}(\mathsf{n}),\quad \mathsf{n}\in\widehat{\Lambda}(\mathsf{K}).$$

That is, $v\circledast v = \mathscr{F}^{-1}[\mathscr{F}(v)\cdot\widehat{u}|_{\widehat{\Lambda}(\mathsf{K})}]$, where $\widehat{u}|_{\widehat{\Lambda}(\mathsf{K})}$ is the sampling of $\widehat{u}$ on the lattice $\widehat{\Lambda}(\mathsf{K})$. We shall omit such dependence and simply write $v\circledast u = \mathscr{F}^{-1}[\widehat{v}\cdot\widehat{u}]$ since it can be easily told from the expression. For a downsampling matrix $\mathsf{A} := \operatorname{diag}(2^{m_1},\dots,2^{m_d})$ for $\mathsf{m} := (m_1,\dots,m_d)\in\mathbb{N}_0^d$, the *downsampling operation* $v\downarrow\mathsf{A}:\Lambda(\mathsf{K}-\mathsf{m})\to\mathbb{C}$ and *upsampling operation* $v\uparrow\mathsf{A}:\Lambda(\mathsf{K}+\mathsf{m})\to\mathbb{C}$ are defined by

$$[v\downarrow\mathsf{A}](\mathsf{k}) = v(\mathsf{A}\mathsf{k}) \text{ for } \mathsf{k}\in\Lambda(\mathsf{K}-\mathsf{m})\quad\text{and}\quad [v\uparrow\mathsf{A}](\mathsf{k}) = \begin{cases} v(\mathsf{A}^{-1}\mathsf{k}) & \text{if } \mathsf{A}^{-1}\mathsf{k}\in\Lambda(\mathsf{K}),\\ 0 & \text{otherwise.}\end{cases}$$

It is easy to show that

$$\mathscr{F}(v\downarrow\mathsf{A}) = \Big(\frac{1}{|\det\mathsf{A}|}\sum_{\omega\in\Omega_{\mathsf{A}}}\widehat{v}(\mathsf{A}^{-\mathsf{T}}\xi+2\pi\omega)\Big)\Big|_{\widehat{\Lambda}(\mathsf{K}-\mathsf{m})},\quad \Omega_{\mathsf{A}} := [\mathsf{A}^{-\mathsf{T}}\mathbb{Z}^d]\cap[0,1)^d.$$

Note that if $\widehat{v}(\cdot)\widehat{v}(\cdot+2\pi\omega)=0$ for all $\omega\in\Omega_{\mathsf{A}}$, then $\mathscr{F}(v\downarrow\mathsf{A})$ can be implemented efficiently by lattice modulation $[\mathscr{F}(v\downarrow\mathsf{A})](\mathsf{n}) := \widehat{v}(\mathsf{A}^{-\mathsf{T}}\mathsf{n}+2\pi\omega_{\mathsf{n}})$ for those n such that $\widehat{v}(\mathsf{A}^{-\mathsf{T}}\mathsf{n}+2\pi\omega_{\mathsf{n}})\neq 0$ for some $\omega_{\mathsf{n}}\in\Omega_{\mathsf{A}}$. For the upsampling operation, we have

$$\mathscr{F}(v\uparrow\mathsf{A}) = \widehat{v}(\mathsf{A}^{\mathsf{T}}\xi)|_{\widehat{\Lambda}(\mathsf{K}+\mathsf{m})},$$

which can be obtained by the periodic extension of $\mathscr{F}(v)$ in practice. The transition operator $\mathcal{T}_{u,\mathsf{A}}v$ and subdivision operator $\mathcal{S}_{u,\mathsf{A}}v$ combine the operations of circular convolution and downsampling(upsampling) together, which are defined to be

$$\mathcal{T}_{u,\mathsf{A}}v := (v\circledast u^\star)\downarrow\mathsf{A}\quad\text{and}\quad \mathcal{S}_{u,\mathsf{A}}v := |\det\mathsf{A}|[(v\uparrow\mathsf{A})\circledast u],$$

where $u^\star$ is defined by $\widehat{u^\star}(\xi) = \overline{\widehat{u}(\xi)}$. It is easy to show that

$$\mathscr{F}(\mathcal{T}_{u,\mathsf{A}}v) = \Big(\frac{1}{|\det \mathsf{A}|}\sum_{\omega\in\Omega_\mathsf{A}} [\widehat{v}\cdot\overline{\widehat{u}}](\mathsf{A}^{-\mathsf{T}}\xi + 2\pi\omega)\Big)\Big|_{\widehat{\Lambda}(\mathsf{K}-\mathsf{m})},$$
$$\mathscr{F}(\mathcal{S}_{u,\mathsf{A}}v) = \Big(|\det \mathsf{A}|\widehat{v}(\mathsf{A}^{\mathsf{T}}\xi)\widehat{u}(\xi)\Big)\Big|_{\widehat{\Lambda}(\mathsf{K}+\mathsf{m})}.$$

Consequently, $\mathscr{F}(\mathcal{S}_{u,\mathsf{A}}\mathcal{T}_{u,\mathsf{A}}v) = \big(\sum_{\omega\in\Omega_\mathsf{A}}\widehat{v}(\xi+2\pi\omega)\cdot\overline{\widehat{u}(\xi+2\pi\omega)}\widehat{u}(\xi)\big)\big|_{\widehat{\Lambda}(\mathsf{K})}$.

Given input data $v^J : \Lambda(\mathsf{K}) \to \mathbb{C}$ and a sequence $\mathsf{DAS}(\{\mathcal{B}_j\}_{j=0}^{J-1})$ of digital affine shear filter banks as in (3.6), the (multilevel) *forward digital affine shear transform* decomposes v^J to a sequence of filtered coefficients

$$\{v^0\}\cup\{w^{j,\vec{\ell},n,\pm} : n = 1,\dots,d, |\vec{\ell}| \leqslant \vec{s}_j\}_{j=0}^{J-1} \tag{3.9}$$

as follows:

$$v^j = \mathcal{T}_{a,\mathsf{M}}v^{j+1} \text{ and } w^{j,\vec{\ell},n,\pm} := \mathcal{T}_{b^{j,\vec{\ell},\pm}(\mathsf{E}_n\cdot),\mathsf{A}_{j,n}}v^{j+1} \text{ for } |\vec{\ell}| \leqslant \vec{s}_j, n = 1,\dots,d, j = J-1,\dots,0.$$

The (multilevel) *backward digital affine shear transform* reconstructs a sequence of filtered coefficients in (3.9) back to a data sequence as follows:

$$\mathring{v}^{j+1} = \mathcal{S}_{a,\mathsf{M}}\mathring{v}^j + \sum_{n=1}^{d}\sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j}\mathcal{S}_{b^{j,\vec{\ell},n,\pm}(\mathsf{E}_n\cdot),\mathsf{A}_{j,n}}w^{j,\vec{\ell},n,\pm} \text{ for } j = 0,\dots,J-1$$

with $\mathring{v}^0 = v$. It is easy to see the PR property of decomposition and reconstruction using (3.6); that is, $v^J = \mathring{v}^J$ for any input data $v^J : \Lambda(\mathsf{K}) \to \mathbb{C}$. In fact, due to (3.7) and (3.8), for each $j = 0,\dots,J-1$, we have

$$\begin{aligned}
\mathscr{F}\Big(\sum_{(b\downarrow\mathsf{A})\in\mathcal{B}_j}\mathcal{S}_{b,\mathsf{A}}\mathcal{T}_{b,\mathsf{A}}v^{j+1}\Big) &= \sum_{(b\downarrow\mathsf{A})\in\mathcal{B}_j}\mathscr{F}\left(\mathcal{S}_{b,\mathsf{A}}\mathcal{T}_{b,\mathsf{A}}v^{j+1}\right)\\
&= \sum_{(b\downarrow\mathsf{A})\in\mathcal{B}_j}\Big(\sum_{\omega\in\Omega_\mathsf{A}}\widehat{v^{j+1}}(\xi+2\pi\omega)\cdot\overline{\widehat{b}(\xi+2\pi\omega)}\widehat{b}(\xi)\Big)\Big|_{\widehat{\Lambda}(\mathsf{K})}\\
&= \sum_{(b\downarrow\mathsf{A})\in\mathcal{B}_j}\Big(\widehat{v^{j+1}}(\xi)\cdot|\widehat{b}(\xi)|^2\Big)\Big|_{\widehat{\Lambda}(\mathsf{K})} = \Big(\widehat{v^{j+1}}(\xi)\sum_{(b\downarrow\mathsf{A})\in\mathcal{B}_j}|\widehat{b}(\xi)|^2\Big)\Big|_{\widehat{\Lambda}(\mathsf{K})}\\
&= \mathscr{F}(v^{j+1}).
\end{aligned}$$

The detailed steps of the forward and backward transforms of a digital affine shear filter bank are presented in Algorithms 1 and 2. We remark that filters in $\mathcal{B}_j$ can be precomputed and stored before doing decomposition and reconstruction. Since by our design the $\operatorname{supp}(\widehat{b})$ can be precomputed for each $b \in \mathcal{B}_j$, we only need to store data of filters on its support. Moreover, we only need to compute $w^{j,\vec{\ell},n,+}$ due to the fact that $\widehat{b^{j,\vec{\ell},-}}(\xi) = \widehat{b^{j,\vec{\ell},+}}(-\xi)$ and $\widehat{v}(-\xi) = \overline{\widehat{v}(\xi)}$ for real-valued data, $\xi \in \mathbb{R}^d$. These make our transforms extremely fast. We shall give the computational complexity estimate in the next subsection.

Algorithm 1. Forward digital affine shear transform (decomposition).

(a) **Input:** d-dimensional real-valued data v on $\Lambda(\mathsf{K})$ for some $\mathsf{K} = (\mathsf{K}_1, \dots, \mathsf{K}_d) \in \mathbb{N}_0^d$ and a sequence $\mathsf{DAS}(\{\mathcal{B}_j\}_{j=0}^{J-1})$ of digital affine shear filter banks defined as in (3.6) with $J \leqslant \min\{\mathsf{K}_1 - 1, \dots, \mathsf{K}_d - 1\}$ and $k_0, \dots, k_{J-1}$ of nonnegative integers determining the number of shear directions in $\mathcal{B}_j := \{a \downarrow \mathsf{M}, b^{j,\vec{\ell},\pm}(\mathsf{E}_n\cdot) \downarrow \mathsf{A}_{j,n} : |\vec{\ell}| \leqslant \vec{s}_j, n = 1, \dots, d\}$ with $\vec{s}_j := (2^{k_j}, \dots, 2^{k_j}) \in \mathbb{N}^{d-1}$, $\mathsf{A}_{j,n} := \mathsf{E}_n \operatorname{diag}(2, 2^{k_j}\mathsf{I}_{d-1})$, and $a, b^{j,\vec{\ell},\pm}$ being defined as in (3.1) and (3.5), respectively.

(b) **Output:** Digital affine shear coefficients: $\{v^0\} \cup \{w^{j,\vec{\ell},n,+} : n = 1, \dots, d, |\vec{\ell}| \leqslant \vec{s}_j\}_{j=0}^{J-1}$.

(c) **Main steps:**

Initialization: $\widehat{v} \leftarrow \mathscr{F}(v)$ and $j \leftarrow J - 1$.
while $j \geqslant 0$ **do**
  Low-pass filtering $\mathcal{T}_{a,\mathsf{M}}v$: Initialize zero matrix $\widehat{u}$ on $\widehat{\Lambda}(\mathsf{K} - 1)$. Compute $\widehat{u}$ by $\widehat{u}(\mathsf{k}) \leftarrow \widehat{v}(\mathsf{M}^{-\mathsf{T}}\mathsf{k}) \cdot \widehat{a}(\mathsf{M}^{-\mathsf{T}}\mathsf{k})$ for $\mathsf{M}^{-\mathsf{T}}\mathsf{k} \in \operatorname{supp}(\widehat{a})$.
  for each $b^{j,\vec{\ell},+}(\mathsf{E}_n\cdot) \downarrow \mathsf{A}_{j,n} \in \mathcal{B}_j$ **do**
    $b \leftarrow b^{j,\vec{\ell},+}(\mathsf{E}_n\cdot)$ and $\mathsf{A} \leftarrow \mathsf{A}_{j,n}$. Initialize a zero matrix c on the lattice $\Lambda(\mathsf{K} - \mathsf{E}_n\vec{k}_j)$, where $\vec{k}_j = (1, k_j, \dots, k_j) \in \mathbb{N}_0^d$.
    Compute $\mathscr{F}(c)$ by $\widehat{c}(\mathsf{k}) \leftarrow \widehat{v}(\mathsf{A}^{\mathsf{T}}\mathsf{k} + 2\pi\omega_{\mathsf{k}}) \cdot \widehat{b}(\mathsf{A}^{\mathsf{T}}\mathsf{k} + 2\pi\omega_{\mathsf{k}})$ for $\mathsf{A}^{\mathsf{T}}\mathsf{k} + 2\pi\omega_{\mathsf{k}} \in \operatorname{supp}(\widehat{b})$.
    High-pass filtering $\mathcal{T}_{b^{j,\vec{\ell},+}(\mathsf{E}_n\cdot),\mathsf{A}^{j,n}}v$: $w^{j,\vec{\ell},n,+} \leftarrow \mathscr{F}^{-1}(\widehat{c})$.
  end for
  $j \leftarrow j - 1$, $\mathsf{K} \leftarrow \mathsf{K} - 1$, and $\widehat{v} \leftarrow \widehat{u}$.
end while
$v^0 \leftarrow \mathscr{F}^{-1}(\widehat{v})$.

Algorithm 2. Backward digital affine shear transform (reconstruction).

(a) **Input:** Digital affine shear coefficients: $\{v^0\} \cup \{w^{j,\vec{\ell},n,+} : n = 1, \dots, d, |\vec{\ell}| \leqslant \vec{s}_j\}_{j=0}^{J-1}$ and a sequence $\mathsf{DAS}(\{\mathcal{B}_j\}_{j=0}^{J-1})$ of digital affine shear filter banks defined as in (3.6) and $k_0, \dots, k_{J-1}$ of nonnegative integers determining the number of shear directions in $\mathcal{B}_j := \{a \downarrow \mathsf{M}, b^{j,\vec{\ell},\pm}(\mathsf{E}_n\cdot) \downarrow \mathsf{A}_{j,n} : |\vec{\ell}| \leqslant \vec{s}_j\}$ with $\vec{s}_j := (2^{k_j}, \dots, 2^{k_j}) \in \mathbb{N}^{d-1}$, $\mathsf{A}_{j,n} := \mathsf{E}_n \operatorname{diag}(2, 2^{k_j}\mathsf{I}_{d-1})$, and $a, b^{j,\vec{\ell},\pm}$ being defined as in (3.1) and (3.5), respectively. The coefficient c^0 is on the lattice $\Lambda(\mathsf{K}^0)$ for some $\mathsf{K}^0 \in \mathbb{N}^d$.

(b) **Output:** d-dimensional real-valued data v on $\Lambda(\mathsf{K})$ with $\mathsf{K} = \mathsf{K}^0 + J$.

(c) **Main steps:**

Initialization: $\widehat{v} \leftarrow \mathscr{F}(c^0)$, $j \leftarrow 0$, and $\mathsf{K} \leftarrow \mathsf{K}^0 + 1$.
while $j < J$ **do**
  Low-pass subdivision $\mathcal{S}_{a,\mathsf{M}}v$: Initialize zero matrix $\widehat{v}_{lo}$ on $\widehat{\Lambda}(\mathsf{K})$. Compute $\widehat{v}_{lo}$ by $\widehat{v}_{lo}(\mathsf{k}) \leftarrow \widehat{v}(\mathsf{M}^{\mathsf{T}}\mathsf{k}) \cdot \widehat{a}(\mathsf{k})$ for $\mathsf{k} \in \operatorname{supp}(\widehat{a})$.
  Initialize a zero matrix $\widehat{v}_{hi}$ on $\widehat{\Lambda}(\mathsf{K})$.
  for each $b^{j,\vec{\ell},+}(\mathsf{E}_n\cdot) \downarrow \mathsf{A}_{j,n} \in \mathcal{B}_j$ **do**
    $b \leftarrow b^{j,\vec{\ell},+}(\mathsf{E}_n\cdot)$, $\mathsf{A} \leftarrow \mathsf{A}_{j,n}$, and $\widehat{c} \leftarrow \mathscr{F}(w^{j,\vec{\ell},n,+})$.

7: High-pass subdivision for all $\mathcal{S}_{b^{j,\vec{\ell},+}(\mathsf{E}_n\cdot),\mathsf{A}^{j,n}}v$: $\widehat{v}_{hi}(\mathsf{A}^\mathsf{T}\mathsf{k}+2\pi\omega_\mathsf{k}) \leftarrow \widehat{v}_{hi}(\mathsf{A}^\mathsf{T}\mathsf{k}+2\pi\omega_\mathsf{k}) + \widehat{c}(\mathsf{k})\widehat{b}(\mathsf{A}^\mathsf{T}\mathsf{k}+2\pi\omega_\mathsf{k})$ for $\mathsf{A}^\mathsf{T}\mathsf{k}+2\pi\omega_\mathsf{k} \in \mathrm{supp}(\widehat{b})$.
8: **end for**
9: High-pass subdivision $\mathcal{S}_{b^{j,\vec{\ell},-}(\mathsf{E}_n\cdot),\mathsf{A}^{j,n}}v$: $\widehat{v}_{hi} \leftarrow \widehat{v}_{hi} + \overline{\widehat{v}_{hi}(-\cdot)}$.
10: $\widehat{v} \leftarrow \widehat{v}_{lo} + \widehat{v}_{hi}$, $\mathsf{K} \leftarrow \mathsf{K}+1$, $j \leftarrow j+1$.
11: **end while**
12: $v \leftarrow \mathscr{F}^{-1}(\widehat{v})$.

3.3. Redundancy rate and computational complexity. We now analyze the redundancy rate and computational complexity of our digital affine shear transforms.

The redundancy rate measures the storage complexity of a filter bank transform, which is defined to be the ratio of the size of the output coefficients to the size of the input data. Let $N = 2^{\mathsf{K}_1+\cdots+\mathsf{K}_d}$ be the size of the input data. We next estimate the size of the output coefficients $\{v^0\} \cup \{w^{j,\vec{\ell},n,+} : |\vec{\ell}| \leqslant \vec{s}_j, n = 1,\dots,d\}_{j=0}^{J-1}$. For our digital affine shear transforms, at scale j for $j = J-1,\dots,0$, the output detailed coefficients are $w^{j,\vec{\ell},n,+}$ for $n = 1,\dots,d$, $|\vec{\ell}| \leqslant \vec{s}_j$ with $\vec{s}_j = (2^{k_j},\dots,2^{k_j}) \in \mathbb{Z}^{d-1}$. The coefficient matrix $w^{j,\vec{\ell},n,+}$ is on the lattice $\Lambda(\widetilde{\mathsf{K}}_1,\dots,\widetilde{\mathsf{K}}_d)$, where $\widetilde{\mathsf{K}}_t = \mathsf{K}_t - (J-1-j) - k_j$ for $t \neq n$ and $\widetilde{\mathsf{K}}_n = \mathsf{K}_n - (J-j)$, which is of size $\frac{N}{2^{d(J-1-j)}} \cdot \frac{1}{2^{1+(d-1)k_j}}$. The total number of high-pass outputs $w^{j,\vec{\ell},n,+}$ at scale j is $d(2^{k_j+1}+1)^{d-1}$. Consequently, the size of the total output coefficients at scale j is

$$\frac{N}{2^{d(J-1-j)}} \cdot \frac{1}{2^{1+(d-1)k_j}} \times d(2^{k_j+1}+1)^{d-1} \times 2.$$

The factor "$\times 2$" is due to the outputs $w^{j,\vec{\ell},n,+}$ being complex-valued. The low-pass coefficient v^0 is of size $N/2^{dJ}$. Therefore, the total size of output coefficients is

$$\begin{aligned} N\left(\sum_{j=0}^{J-1} \frac{2d(2^{k_j+1}+1)^{d-1}}{2^{d(J-1-j)} \cdot 2^{1+(d-1)k_j}} + \frac{1}{2^{dJ}}\right) &= N\left(\sum_{j=0}^{J-1} \frac{d(2^{-k_j}+2)^{d-1}}{2^{d(J-1-j)}} + \frac{1}{2^{dJ}}\right) \\ &\leqslant N d(2^{-k_{min}}+2)^{d-1}\left(\sum_{j=0}^{\infty} 2^{-dj}\right) \\ &= N d(2^{-k_{min}}+2)^{d-1}\frac{2^d}{2^d-1}, \end{aligned}$$

where $k_{min} := \min\{k_j : j = 0,\dots,J-1\}$. The redundancy rate r is given by

$$r = \left(\sum_{j=0}^{J-1} \frac{d(2^{-k_j}+2)^{d-1}}{2^{dj}} + \frac{1}{2^{dJ}}\right) \leqslant d(2^{-k_{min}}+2)^{d-1}\frac{2^d}{2^d-1}.$$

Table 1 gives the bounds of the redundancy rate with respect to a fixed dimension d and a fixed k_{min}.

Table 1

Theoretical redundancy bound of r for our digital affine shear transforms. The dimension $d = 2, 3, 4, 5$ and the minimal direction parameter $k_{min} = 0, \dots, 6$. The last column is the redundancy rate for TP-$\mathbb{C}$TF$_6$ *in* [18].

$d\backslash k_{min}$	0	1	2	3	4	5	6	2^d	TP-$\mathbb{C}$TF$_6$ [18]
2	8.00	6.67	6.00	5.67	5.50	5.42	5.38	4	10.67
3	30.86	21.43	17.36	15.48	14.58	14.15	13.93	8	29.71
4	115.20	66.67	48.60	40.94	37.43	35.76	34.94	16	85.33
5	418.06	201.61	132.28	105.24	93.40	87.86	85.19	32	249.80

Table 2

Redundancy rates of several directional systems in practice for 2D and 3D problems. For their performances in image/video processing, see section 4. DAS is our digital affine shear transform. DT-$\mathbb{C}$WT is the dual-tree complex wavelet transform in [26]. TP-$\mathbb{C}$TF$_6$ *is the tensor product complex tight framelet transform in* [18]. *DNST is the compactly supported shearlet transform in* [22]. *FDCT is the fast discrete curvelet transform in* [3]. *NSCT is the nonsubsampled contourlet transform in* [5]. *SURF is the surfacelet transform in* [23].

$d\backslash$systems	DAS	DT-$\mathbb{C}$WT	TP-$\mathbb{C}$TF$_6$	DNST	FDCT	NSCT	SURF
2D	6.17	4	10.67	49	2.8	53	N.A.
3D	17.88	8	29.71	154	N.A.	N.A.	6.4

Note that instead of increasing, the bound of the redundancy rate r decreases as k_{min} increases, which determines the number of shear directions $\geqslant d(2^{k_{min}+1}+1)^{d-1}$. In dimension two, $k_j = 2$ corresponds to 18 directions at scale j. In dimension three, $k_j = 2$ corresponds to 243 directions at scale j. In Table 1, we also compare with 2^d and the redundancy rate of TP-$\mathbb{C}$TF$_6$ in [18], which has redundancy rate $2^d \times \frac{3^d-1}{2^d-1}$. One can see from the table that the redundancy rate increases more slowly than TP-$\mathbb{C}$TF$_6$ with respect to dimension d for a fixed $k_{min} \geqslant 1$. Actually, the ratio between these two is bounded by $\frac{d(2^{-k_{min}}+2)^{d-1}}{3^d-1} \to 0$ as $d \to \infty$ for $k_{min} \geqslant 1$. In Table 2, we compare the redundancy rates of different directional systems in dimensions two and three. One can see that DT-$\mathbb{C}$WT, FDCT, and SURF have low redundancy rates. The redundancy rates of our DAS and TP-$\mathbb{C}$TF$_6$ are moderate and close to each other, while DNST and NSCT have extremely high redundancy rates. For their performances in terms of peak signal-to-noise ratio (PSNR) in image/video processing, see section 4.

For fast implementation, with a fixed size of data and a fixed finest level J of decomposition, the digital affine shear filter banks $\mathsf{DAS}(\{\mathcal{B}_j\}_{j=0}^{J-1})$ can be precomputed. More precisely, for each $b \in \mathcal{B}_j$, by the compact property of $\widehat{b}$, we only need to store the support indices and values $\{(\mathsf{k}, \widehat{b}(\mathsf{k})) : \widehat{b}(\mathsf{k}) \neq 0, \mathsf{k} \in \widehat{\Lambda}(\mathsf{K} - (J-1-j))\}$. Again, by the symmetry property, we only need to store half of such information. Similar to the above analysis, we can conclude that the total size is again bounded by rN for N the size of input data and r the redundancy rate of the transform. In conclusion, the storage complexity, even with precomputed filters, is proportional to rN.

We next discuss the computational complexity of the digital affine shear transforms. In Algorithm 1 (decomposition), the computational complexity of line 1 (initialization, DFT) is $N \log N$. Let us next analyze lines 3–7 in the `while` loop with respect to scale $j = J-1, \dots, 0$. At line 3, we need to perform $\frac{N}{2^{d(J-j)}}$ (complex) multiplications for computing $\widehat{v}|_{\widehat{\Lambda}(\mathsf{K}-(J-j))}$.

Since the support indices are precomputed, the main computational cost in the `for` loop is line 6 for multiplication and line 7 for inverse DFT. The total number of (complex) multiplications in the `for` loop is

$$\left(\frac{N}{2^{d(J-1-j)}}\cdot\frac{1+\log(N)-d(J-1-j)-1-(d-1)k_j}{2^{1+(d-1)k_j}}\right)\times d(2^{k_j+1}+1)^{d-1}.$$

The last step, line 11 (inverse DFT), requires $\frac{N}{2^{dJ}}(\log(N)-dJ)$ numbers of (complex) multiplications. Consequently, the total computational cost is

$$\begin{aligned}
&N\log N+\sum_{j=0}^{J-1}\left[\frac{N}{2^{d(J-1-j)}}\cdot\frac{1+\log N-d(J-1-j)-1-(d-1)k_j}{2^{1+(d-1)k_j}}\times d(2^{k_j+1}+1)^{d-1}\right.\\
&\qquad\left.+\frac{N}{2^{d(J-j)}}\right]+\frac{N}{2^{dJ}}(\log N-dJ)\\
\leqslant&N\log N+N(1+\log N)\left(\sum_{j=0}^{J-1}\left[\frac{d(2^{k_j+1}+1)^{d-1}}{2^{d(J-1-j)}\cdot 2^{1+(d-1)k_j}}+\frac{1}{2^{d(J-j)}}\right]+\frac{1}{2^{dJ}}\right)\\
\leqslant&N\log N+N(1+\log N)(r+(2^d-1)^{-1})\\
\leqslant&N[(1+\log N)(r+(2^d-1)^{-1}+1)-1]\\
\leqslant&(r+2)\cdot(N+N\log N).
\end{aligned}$$

Roughly speaking, N mainly comes from the pointwise multiplications of filters and input data, while $N\log N$ comes from the performance of DFT for output coefficients. Our computational cost is proportional to the computational complexity of DFT for input data of size N. The ratio is controlled by the redundancy rate r. Since the backward transform Algorithm 2 basically reverses the steps in Algorithm 1, the computational cost of Algorithm 2 is the same as Algorithm 1.

Table 3 gives the computational time comparison between the forward digital affine shear transform (DAS) and the standard DFT in dimension three. The level J is set to be 4 and $k_0=0$, $k_1=1$, $k_2=1$, $k_3=2$. For each $N=n^3$ with n ranging from 96 to 256 (step size 16), the computational time (in seconds) for a standard DFT (FFT time) and our forward digital affine shear transform (DAS time) are obtained (average over 10 trials in MATLAB using *timeit*). The ratio of the DAS time to the FFT time is given (see the row with respect to $t_{\mathrm{DAS}}:t_{\mathrm{FFT}}$ in Table 3). Moreover, we also run the similar experiments for some available 3D directional transform packages including TP-$\mathbb{C}$TF$_6$ [18], DT-$\mathbb{C}$WT [26], DNST [22], and SURF [23]. Note that we only run the forward transform from each package and evaluate the computational time. The results are given in Table 3. The last column shows the redundancy rate of each transform as well. We can see that the computational complexity of our digital affine shear transform is very close to $r\times t_{\mathrm{FFT}}$ with r the redundancy rate. For others, TP-$\mathbb{C}$TF$_6$ is about twice $r\times t_{\mathrm{FFT}}$, SURF is 4 times slower, while DT-$\mathbb{C}$WT and DNST are about 10 times slower than $r\times t_{\mathrm{FFT}}$. For a more detailed comparison in terms of PSNR in video processing, see section 4.

Table 3

Computational time of the forward transforms of FFT, DAS, TP-$\mathbb{C}$TF$_6$, DT-$\mathbb{C}$WT, *SURF, and DNST for* 3*D data of size* $N = n \times n \times n$ *with* n *ranging from* 96 *to* 256 *(step size* 16*). Each computational time is averaged over* 10 *runs. The ratio of each transform to the FFT time is given as well. The last column,* r*, is the redundancy rate of the corresponding transforms.*

n	96	112	128	144	160	176	192	208	224	240	256	r
t_{FFT}	0.008	0.015	0.024	0.035	0.046	0.064	0.093	0.105	0.130	0.156	0.219	1
t_{DAS}	0.303	0.433	0.559	0.898	1.153	1.542	2.050	2.546	3.118	3.827	4.459	
$t_{\mathrm{DAS}} : t_{\mathrm{FFT}}$	35.75	29.65	22.89	25.98	25.26	24.11	22.03	24.25	24.07	24.54	20.39	17.88
$t_{\text{TP-}\mathbb{C}\mathrm{TF}_6}$	0.594	0.890	1.276	1.980	2.674	3.609	4.844	6.331	8.249	10.136	13.180	
$t_{\text{TP-}\mathbb{C}\mathrm{TF}_6} : t_{\mathrm{FFT}}$	69.99	60.93	52.25	57.28	58.59	56.45	52.05	60.31	63.69	64.99	60.28	29.71
$t_{\text{DT-}\mathbb{C}\mathrm{WT}}$	1.18	1.70	2.43	3.18	4.34	5.71	6.86	8.73	10.91	13.12	16.95	
$t_{\text{DT-}\mathbb{C}\mathrm{WT}} : t_{\mathrm{FFT}}$	138.5	116.8	99.4	92.1	95.0	89.3	73.7	83.2	84.2	84.1	77.5	8
t_{SURF}	N.A.	N.A.	0.61	1.04	1.26	1.86	2.30	3.07	3.54	4.94	5.19	
$t_{\mathrm{SURF}} : t_{\mathrm{FFT}}$	N.A.	N.A.	25.0	30.1	27.6	29.1	24.7	29.2	27.3	31.7	23.7	6.4
t_{DNST}	N.A.	N.A.	N.A.	52.7	74.7	99.3	132.9	161.6	202.1	251.5	362.7	
$t_{\mathrm{DNST}} : t_{\mathrm{FFT}}$	N.A.	N.A.	N.A.	1525	1636	1553	1428	1539	1561	1613	1659	154

4. Numerical experiments on image and video processing. In this section, we shall apply our digital affine shear transforms[1] for the tasks of image/video processing including denoising and inpainting. We will mainly compare the performance of our systems to several other state-of-the-art directional multiscale representation systems.

The PSNR index is used to measure the performance of different systems and is defined to be

$$\mathrm{PSNR}(u, \widetilde{u}) = 10 \log_{10} \frac{255^2}{\mathrm{MSE}(u, \widetilde{u})}, \tag{4.1}$$

where $u : \Lambda \to \mathbb{C}$ is the original data defined on a lattice Λ, $\widetilde{u}$ is the denoised/inpainted data of u, and $\mathrm{MSE}(u, \widetilde{u})$ is the mean square error $\frac{1}{|\Lambda|} \sum_{\mathsf{k} \in \Lambda} |u(\mathsf{k}) - \widetilde{u}(\mathsf{k})|^2$ with $|\Lambda|$ the cardinality of the lattice Λ. The unit of PSNR is dB.

For the thresholding technique in the denoising/inpainting task using our digital affine shear transforms, we employ the local-soft thresholding method: For each high-pass coefficient matrix $w \in \{w^{j,\vec{\ell},n,+} : n = 1, \ldots, d, |\vec{\ell}| \leqslant \vec{s}_j\}_{j=0}^{J-1}$, let b be the filter that induces w, that is, $w = \mathcal{T}_{b,\mathsf{B}} v^J$ for the input data v^J and some downsampling matrix B, we first normalize it with respect to the norm $\|b\|_2$ of b to obtain $w_b := \frac{w}{\|b\|_2}$. The filter b can be computed by applying the backward transform to a delta data on the support of w and $\|b\|_2$ is the Frobenius norm of the reconstructed data. Let σ be the variance of a noise obeying normal distribution $N(0, \sigma^2)$ and Λ be the lattice for w. For each $\mathsf{k} \in \Lambda$, compute local coefficient variance $\sigma_w : \Lambda \to [0, \infty)$ by $\sigma_w(\mathsf{k}) := \sqrt{\left(\frac{1}{\#\Lambda_{\mathsf{k}}} \sum_{\mathsf{n} \in \Lambda_{\mathsf{k}}} |w_b(\mathsf{n})|^2 - \sigma^2\right)_+}$, where Λ_{k} is the lattice $\mathsf{k} + [-L, L]^d$ that centered at position k for some integer $L \geqslant 0$. Note that σ_w can be computed by convolution of w_b with a normalized window $\frac{[-L,L]^d \cap \mathbb{Z}^d}{(2L+1)^d}$. The threshold $t_w : \Lambda \to [0, \infty]$ is then defined by $t_w(\mathsf{k}) = \frac{\sigma^2}{\sigma_w(\mathsf{k})}$, $\mathsf{k} \in \Lambda$. The soft thresholding operator $\eta_t^{soft}(x)$ and hard thresholding operator

[1]The full MATLAB package of our digital affine shear transforms (2D and 3D) and their applications in image/vidoe processing can be downloaded from http://personal.cityu.edu.hk/~xzhuang7/softs.

$\eta_t^{hard}(x)$ are defined to be

$$\eta_t^{soft}(x) = \begin{cases} x - t\frac{x}{|x|}, & |x| > t, \\ 0 & \text{otherwise}, \end{cases} \quad \text{and} \quad \eta_t^{hard}(x) = \begin{cases} x, & |x| > t, \\ 0 & \text{otherwise}. \end{cases}$$

The local-soft thresholding $\eta_{t_w}^{ls}(w_b) : \Lambda \to \mathbb{C}$ applying to w_b with threshold t_w is then defined to be $[\eta_{t_w}^{ls}(w_b)](\mathsf{k}) := \eta_{t_w(\mathsf{k})}^{soft}(w_b(\mathsf{k}))$. The threshold coefficient $\widetilde{w}$ from w is then renormalized by $\widetilde{w} := \|b\|_2 \cdot \eta_{t_w}^{ls}(w_b)$.

Figure 2. *2D images* Lena *and* Barbara *(left two), masks Text* 1 *and Text* 2 *for inpainting (middle two), and first frame of* 3*D data* Mobile *and* Coastguard *(right two).*

4.1. Comparisons on image denoising and inpainting. In this subsection, we apply our digital affine shear transforms to the task of denoising and inpainting in image processing. The parameters $c_0, \epsilon_0, \epsilon_1$ of $a, \mathbf{g}$ in (3.1) and (3.2) are given by $c_0 = \frac{33}{32}$, $c_1 = \pi$, $\epsilon_0 = \frac{69}{128}$, $\epsilon_1 = \frac{51}{512}$, and $\varepsilon = 0.469$ for $\boldsymbol{\gamma}_\varepsilon$. We choose $J = 5$ for $\mathsf{DAS}(\{\mathcal{B}_j\}_{j=0}^{J-1})$ as in (3.6); that is, we decompose to 5 scales. The shear parameter $(k_4, k_3, k_2, k_1, k_0)$ is set to be $(2, 1, 1, 1, 0)$. That is, for the finest scale $j = 4$, we use totally $2(2^{k_4+1} + 1) = 18$ shear directions (9 on horizontal cone and 9 on vertical cone). For the next three scales $j = 3, 2, 1$, we use 10 shear directions, and for the coarsest scale, we use 6 shear directions. The redundancy rate of our system $\mathsf{DAS}(\{\mathcal{B}_j\}_{j=0}^{4})$ is 6.165. The convolution window size L to compute local coefficient variance σ_w is set to be 5, i.e., we are using 11×11 window filter.

We first apply our system to the task of denoising in image processing. We test two standard images: Lena and Barbara. See Figure 2. Both are of size 512×512. We first employ symmetric boundary extension (with 32 pixels) on the noisy image to avoid the boundary effect. We then apply our forward transform to obtain the coefficients. After performing the local-soft threshold procedure, we then apply the backward transform to the thresholded coefficients and throw away the extended boundary to obtain the final denoised image.

We compare our denoising performance to several other state-of-the-art directional multiscale representation systems: dual-tree complex wavelets [26], tensor product complex tight framelets [17, 18], curvelets [2, 3], compact support shearlets [22], and contourlets [5]. We download each of their available packages and run their denoising codes for both Lena and Barbara.

The 2D DT-$\mathbb{C}$WT in [26] has redundancy rate 4. The number of directional filters of DT-$\mathbb{C}$WT at each scale is 6 covering $\pm 15°, \pm 45°, \pm 75°$. The number of scales is 6. Bivariate shrinkage thresholding technique is employed for denoising. The TP-$\mathbb{C}\mathrm{TF}_6$ is detailed in [17, 18] and has redundancy rate 10.67 and 14 directional filters for each scale. The number of scales is 5 and it also uses the bivariate shrinkage thresholding technique.

The curvelet transform (FDCT at http://www.curvelab.org) has two implementations: one uses un-equispace FFT, the other uses frequency wrapping. Here we use the wrapping package; detailed information can be found in [3]. The performance of these two implementations is very close (less than 0.2 dB) and we choose the wrapping one for comparison. The total number of scales is 5. At the finest scale, the FDCT uses isotropic wavelet transform to avoid checkerboard effect. At scale 4, 32 (angular) directions are used. At scales 3 and 2, 16 (angular) directions are used. At the coarsest scale, 8 (angular) directions are used. The redundancy of the FDCT is about 2.8. The shearlet transform at http://www.shearlab.org also has many implementations and we choose the compactly supported shearlet implementation DNST as in [22], which has the best performance so far in the ShearLab package. For DNST, the total number of scales is 4. Sixteen shear directions are used for finest scale 4 and 3, while 8 shear directions are used for the other two scales. The redundancy of DNST is 49. The contourlet transform [5] (NSCT package at http://minhdo.ece.illinois.edu/software) has redundancy rate 53. It uses 4, 8, 8, 16, and 16 directions in the scales from coarser to finer. All three transforms use hard thresholding for denoising.

We compare the denoising performance over different noise levels $\sigma \in \{5, 10, 30, 50, 80, 100\}$. The comparison results are presented in Table 4. The values in the brackets are gain (+) or loss (−) of our method compared to other methods. From Table 4, we see significant improvement over FDCT and NSCT for both Lena and Barbara. Comparing our method with DNST, when the noise level is small ($\sigma \leqslant 30$), DNST performs better than our method for the image Lena, while our method performs better than DNST when the noise level is high ($\sigma > 30$) for both Lena and Barbara. The performance of DT-$\mathbb{C}$WT and our method is comparable when the noise level is small and our method outperforms DT-$\mathbb{C}$WT when the noise level is high. TP-$\mathbb{C}$TF$_6$ outperforms our method for Lena when the noise level $\sigma \leqslant 50$ and for Barbara when $\sigma \leqslant 30$. However, when the noise level is high, our method outperforms TP-$\mathbb{C}$TF$_6$. In summary, we conclude that our method is in general better than DT-$\mathbb{C}$WT, TP-$\mathbb{C}$TF$_6$, FDCT, DNST, and NSCT, especially for texture-rich image Barbara and for a high noise level ($\sigma > 30$).

We next apply our system to the task of inpainting in image processing and compare the performance with benchmark system TP-$\mathbb{C}$TF$_6$ in [18]. The directional tensor product complex framelet system TP-$\mathbb{C}$TF$_6$ has been shown in the image inpainting problem with impressive performance over many other state-of-the-art frame-based systems (see [27]). Here we employ the same inpainting framework developed in [27], which uses an iterative thresholding algorithm with gradually decreasing threshold values. The threshold technique we employ for our digital affine shear transforms is the local-soft thresholding, while the threshold technique for TP-$\mathbb{C}$TF$_6$ is bivariate shrinkage (see [27] or [18] for more details). The parameters for our digital affine shear transforms are the same as those for image denoising. We test on the two standard 512×512 images Barbara and Lena with four cases: scratch missing information with masks Text 1 and Text 2 (see Figure 2) and random missing information with 50% pixels missing and 80% pixels missing. For each case, we compare the performance with respect to different Gaussian noise for $\sigma = 0, 5, 10, 20, 30, 50$. Here $\sigma = 0$ simply means no noise for the image with missing information. The results are reported in Table 5. From Table 5, we can conclude that for texture-rich image Barbara when the noise level is low or percentage of missing pixels is small, TP-$\mathbb{C}$TF$_6$ in general performs better than digital affine

Table 4
PSNR of denoised Lena *and* Barbara *using different transforms.*

512 × 512 Lena						
σ	DAS	DT-$\mathbb{C}$WT	TP-$\mathbb{C}$TF$_6$	DNST	FDCT	NSCT
5	38.14	38.25(-0.11)	38.37(-0.23)	38.01(0.13)	35.77(2.37)	37.71(0.43)
10	35.12	35.19(-0.07)	35.48(-0.36)	35.35(-0.23)	33.37(1.75)	34.92(0.20)
30	30.61	30.50(0.11)	30.80(-0.19)	30.68(-0.07)	29.34(1.27)	30.32(0.29)
50	28.49	28.22(0.27)	28.54(-0.05)	28.21(0.28)	27.19(1.30)	28.02(0.47)
80	26.54	26.15(0.39)	26.47(0.07)	25.78(0.76)	25.16(1.38)	25.80(0.74)
100	25.63	25.20(0.43)	25.52(0.11)	24.58(1.05)	24.22(1.41)	24.71(0.92)
512 × 512 Barbara						
σ	DAS	DT-$\mathbb{C}$WT	TP-$\mathbb{C}$TF$_6$	DNST	FDCT	NSCT
5	37.32	37.37(-0.05)	37.84(-0.52)	37.17(0.15)	33.83(3.49)	36.96(0.36)
10	33.64	33.54(0.10)	34.18(-0.54)	33.62(0.02)	29.17(4.47)	33.35(0.29)
30	28.33	27.89(0.44)	28.38(-0.05)	27.97(0.36)	24.44(3.89)	27.28(1.05)
50	26.01	25.36(0.65)	25.71(0.30)	25.31(0.70)	23.38(2.63)	24.57(1.44)
80	23.99	23.27(0.72)	23.53(0.46)	22.96(1.03)	22.22(1.77)	22.65(1.34)
100	23.07	22.42(0.65)	22.64(0.43)	22.06(1.01)	21.61(1.46)	21.90(1.17)

shear transforms. However, when the noise level is high, the digital affine shear transforms significantly outperform TP-$\mathbb{C}$TF$_6$ for Barbara. The performance of digital affine shear transforms is slightly worse than that of the TP-$\mathbb{C}$TF$_6$ on Lena because edge information is not as significantly important as those for the image Barbara.

Table 5
PSNR of inpainted Lena *and* Barbara *using different transforms.*

	512 × 512 Barbara							
	Text 1		Text 2		50% missing		80% missing	
σ	DAS	[27](TP-$\mathbb{C}$TF$_6$)	DAS	[27](TP-$\mathbb{C}$TF$_6$)	DAS	[27](TP-$\mathbb{C}$TF$_6$)	DAS	[27](TP-$\mathbb{C}$TF$_6$)
0	36.16	36.59(-0.43)	33.41	32.68(0.73)	33.77	35.73(-1.96)	27.84	28.16(-0.32)
5	33.75	34.05(-0.30)	31.94	31.32(0.62)	32.14	33.42(-1.28)	27.30	27.73(-0.43)
10	31.67	31.81(-0.14)	30.39	29.85(0.54)	30.47	31.11(-0.64)	26.72	26.70(0.02)
20	29.06	28.99(0.07)	28.15	27.71(0.44)	27.99	28.00(-0.01)	25.26	24.70(0.56)
30	27.38	27.18(0.20)	26.65	26.24(0.41)	26.32	25.95(0.37)	24.16	23.34(0.82)
50	25.29	24.91(0.38)	24.73	24.30(0.43)	24.25	23.60(0.65)	22.26	21.90(0.36)
	512 × 512 Lena							
0	37.98	38.02(-0.04)	33.93	34.31(-0.38)	35.72	38.00(-2.28)	30.74	32.33(-1.59)
5	35.15	35.19(-0.04)	32.63	32.97(-0.34)	33.97	35.40(-1.43)	30.20	31.44(-1.24)
10	33.34	33.42(-0.08)	31.58	31.80(-0.22)	32.57	33.40(-0.83)	29.54	30.25(-0.71)
20	31.08	31.26(-0.18)	30.02	30.10(-0.08)	30.41	30.84(-0.43)	28.09	28.36(-0.27)
30	29.68	29.81(-0.13)	28.84	28.89(-0.05)	28.90	29.18(-0.28)	26.76	26.95(-0.19)
50	27.82	27.85(-0.03)	27.20	27.22(-0.02)	26.92	27.06(-0.14)	25.01	25.15(-0.14)

4.2. Comparisons of video denoising and inpainting. In this subsection, we apply our digital affine shear transforms to the task of denoising and inpainting in video processing. The parameters $c_0, c_1, \epsilon_0, \epsilon_1$ of $a, \mathbf{g}$ in (3.1) and (3.2) are the same as in the task of image denoising/inpainting. We choose $J = 4$ for $\mathsf{DAS}(\{\mathcal{B}_j\}_{j=0}^{J-1})$ as in (3.6); that is, we decompose to four scales. The shear parameters (k_3, k_2, k_1, k_0) are set to be $(2, 1, 1, 0)$. That is, for the

finest scale $j = 3$, we use totally $3(2^{k_3+1} + 1)^2 = 243$ shear directions (81 for each cone). For the next two scales $j = 2, 1$, we use 75 shear directions, and for the coarsest scale, we use 27 shear directions. The redundancy of our system $\mathsf{DAS}(\{\mathcal{B}_j\}_{j=0}^3)$ is 17.88. The window size L to compute local coefficient variance σ_w is set to be 2, i.e., we are using a $5 \times 5 \times 5$ window.

We next apply our system to the task of denoising in video processing. We test two videos, Mobile and Coastguard, which can be downloaded from http://www.shearlab.org (see the original source at http://www.cipr.rpi.edu/resource/sequences/). Both are of size $192 \times 192 \times 192$. We first employ symmetric boundary extension (with 16 pixels) on the noisy image to avoid the boundary effect. We then apply our forward transform to obtain the coefficients. After performing the local-soft threshold procedure, we then apply the backward transform to the thresholded coefficients and remove the extended boundary to obtain the final denoised image. See Figure 2.

We compare our denoising performance to 3D dual-tree complex wavelets [26], 3D tensor product complex tight framelets [17, 18], surfacelets [23], and 3D compact support shearlets [22]. The 3D DT-$\mathbb{C}$WT in [26] has redundancy rate 8. The number of directional filters of DT-$\mathbb{C}$WT at each scale is 56. The number of scales is five. The 3D TP-$\mathbb{C}$TF$_6$ is detailed in [18] and has redundancy rate 29.71 and 208 directional filters for each scale. The number of scales is 4. The bivariate shrinkage thresholding technique is employed for both DT-$\mathbb{C}$WT and TP-$\mathbb{C}$TF$_6$. For 3D DNST from the ShearLab package, we choose the one with redundancy rate 154 (three scales). The surfacelet transform (SURF) from SurfBox at http://minhdo.ece.illinois.edu/software has redundancy rate 6.4. The 3D DNST and sufacelet transform use hard thresholding for denoising.

We compare the denoising performance over different noise levels $\sigma \in \{10, 20, 30, 50, 80, 100\}$. The comparison results are presented in Table 6. The values in the brackets are gain $(+)$ or loss $(-)$ of our method compared to other methods. From Table 6, we see that our method outperforms both two low-redundancy 3D transforms DT-$\mathbb{C}$WT and surfacelets. For the other two high-redundant 3D transforms TP-$\mathbb{C}$TF$_6$ and DNST, our performance is worse while the noise level is small ($\sigma < 50$) but significantly better when the noise level is high ($\sigma \geqslant 50$).

We finally apply our system to the task of inpainting in video processing. Again, we only compare with the benchmark system TP-$\mathbb{C}$TF$_6$ using the same inpainting framework in [27]. The parameters for our digital affine shear transforms are the same as those for video denoising. The comparison of TP-$\mathbb{C}$TF$_6$ has been done in [18]. Interested readers are referred to [18, section 4.2] for more details. We test on two videos, Mobile and Coastguard, for two cases: 50% random missing pixels and 80% random missing pixels. For each case, the videos are also corrupted by Gaussian noise with noise variance $\sigma \in \{0, 10, 30\}$. The results are reported in Table 7. From Table 7, we can conclude that when the noise level is low and the percentage of missing pixels is small, TP-$\mathbb{C}$TF$_6$ performs in general better than digital affine shear transforms. However, when the noise level is high, the digital affine shear transforms significantly outperform TP-$\mathbb{C}$TF$_6$.

5. Proofs.

In this section, we prove Theorems 2.1 and 2.2.

Table 6
PSNR of denoised Mobile *and* Coastguard.

192 × 192 × 192 Mobile					
σ	DAS	DT-$\mathbb{C}$WT	TP-$\mathbb{C}$TF$_6$	SURF	DNST
10	34.99	34.72(0.27)	35.52(-0.53)	32.79(2.20)	35.91(-0.92)
20	31.50	30.86(0.64)	31.77(-0.27)	29.95(1.55)	32.18(-0.68)
30	29.57	28.67(0.90)	29.66(-0.09)	28.26(1.31)	29.99(-0.42)
50	27.26	26.06(1.20)	27.08(0.18)	26.11(1.15)	27.22(0.04)
80	25.20	24.00(1.20)	24.82(0.38)	24.25(0.95)	24.75(0.45)
100	24.23	23.17(1.06)	23.82(0.41)	23.40(0.83)	23.62(0.61)
192 × 192 × 192 Coastguard					
σ	DAS	DT-$\mathbb{C}$WT	TP-$\mathbb{C}$TF$_6$	SURF	DNST
10	33.70	33.21(0.49)	34.15(-0.45)	30.86(2.84)	33.81(-0.11)
20	30.27	29.61(0.66)	30.62(-0.35)	28.26(2.01)	30.28(-0.01)
30	28.47	27.71(0.76)	28.73(-0.26)	26.87(1.60)	28.40(0.07)
50	26.40	25.56(0.84)	26.48(-0.08)	25.17(1.23)	26.17(0.23)
80	24.65	23.83(0.82)	24.53(0.12)	23.61(1.04)	24.17(0.48)
100	23.86	23.08(0.78)	23.65(0.21)	22.87(0.99)	23.22(0.64)

Table 7
Performance in terms of PSNR values of two video inpainting algorithms: 3D DAS (redundancy rate 17.88*) and 3D* TP-$\mathbb{C}$TF$_6$ *(reduandancy rate* $29\frac{5}{7}$*) with respect to different Gaussian noise (*$\sigma = 0, 10, 30$*, where* $\sigma = 0$ *means no noise). The percentages* 50% *and* 80% *are experiments with* 50% *and* 80% *uniformly randomly missing pixels, respectively.*

	192 × 192 × 192 Mobile				192 × 192 × 192 Coastguard			
	50% missing		80% missing		50% missing		80% missing	
σ	DAS	TP-$\mathbb{C}$TF$_6$	DAS	TP-$\mathbb{C}$TF$_6$	DAS	TP-$\mathbb{C}$TF$_6$	DAS	TP-$\mathbb{C}$TF$_6$
0	37.91	41.74(-3.83)	30.55	28.61(1.94)	34.91	37.75(-2.84)	28.48	27.41(1.07)
10	32.19	33.09(-0.90)	28.26	27.84(0.42)	30.66	31.48(-0.82)	26.95	26.67(0.28)
30	27.56	27.87(-0.31)	25.11	23.53(1.58)	26.62	27.15(-0.53)	24.55	23.29(1.26)

5.1. Proof of Theorem 2.1. We define the following functions for the purpose of charaterization:

$$\begin{cases} \mathcal{I}^{\mathsf{k}}_{\varphi^j}(\xi) & := \quad \overline{\widehat{\varphi^j}(\xi)}\widehat{\varphi^j}(\xi + 2\pi\mathsf{k}), \quad \mathsf{k} \in \mathbb{Z}^d, \quad \xi \in \mathbb{R}^d; \\ \mathcal{I}^{\mathsf{k}}_{\Psi_j}(\xi) & := \quad \sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j} \overline{\widehat{\psi^{j,\vec{\ell}}}(\mathsf{S}_{\vec{\ell}}\xi)}\widehat{\psi^{j,\vec{\ell}}}(\mathsf{S}_{\vec{\ell}}(\xi + 2\pi\mathsf{k})), \quad \mathsf{k} \in \mathbb{Z}^d, \quad \xi \in \mathbb{R}^d; \\ \mathcal{I}^{\mathsf{k}}_{\varphi^j}(\xi) & = \quad \mathcal{I}^{\mathsf{k}}_{\Psi_j}(\xi) := 0, \quad \mathsf{k} \in \mathbb{R}^d \backslash \mathbb{Z}^d, \quad \xi \in \mathbb{R}^d. \end{cases} \tag{5.1}$$

We have the following characterization for a sequence $\mathsf{AS}(\varphi^J; \{\Psi_j\}_{j=J}^{\infty})$, $J \geqslant J_0$, of affine shear systems to be a sequence of affine shear tight frames for $L_2(\mathbb{R}^d)$ (also see [19, Theorem 2]), which include the result in Theorem 2.1 as a special case.

Theorem 5.1. *Let $J_0 \in \mathbb{Z}$ be an integer and $\mathsf{AS}(\varphi^J; \{\Psi_j\}_{j=J}^{\infty})$ be defined as in (2.2) with $J \geqslant J_0$. Then the following statements are equivalent to each other:*

(1) $\mathsf{AS}(\varphi^J; \{\Psi_j\}_{j=J}^{\infty})$ *is an affine shear tight frame for $L_2(\mathbb{R}^d)$ for every integer $J \geqslant J_0$.*

(2) *The following identities hold:*

$$\lim_{j\to\infty}\sum_{\mathsf{k}\in\mathbb{Z}^d}|\langle f,\varphi^j_{\mathsf{M}_\lambda^j;\mathsf{k}}\rangle|^2=\|f\|_2^2, \tag{5.2}$$

$$\sum_{\mathsf{k}\in\mathbb{Z}^d}|\langle f,\varphi^{j+1}_{\mathsf{M}_\lambda^{j+1};\mathsf{k}}\rangle|^2=\sum_{\mathsf{k}\in\mathbb{Z}^d}|\langle f,\varphi^j_{\mathsf{M}_\lambda^j;\mathsf{k}}\rangle|^2+\sum_{n=1}^d\sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j}\sum_{\mathsf{k}\in\mathbb{Z}^d}|\langle f,\psi^{j,\vec{\ell}}_{\mathsf{S}^{-\vec{\ell}}\mathsf{A}_\lambda^j\mathsf{E}_n;\mathsf{k}}\rangle|^2 \tag{5.3}$$

for all $j\geqslant J_0$ and for all $\widehat{f}\in\mathscr{D}(\mathbb{R}^d)$.

(3) *The following identities hold:*

$$\lim_{j\to\infty}\langle|\widehat{\varphi^j}(\mathsf{N}_\lambda^j\cdot)|^2,\widehat{h}\rangle=\langle 1,\widehat{h}\rangle\quad\forall\widehat{h}\in\mathscr{D}(\mathbb{R}^d), \tag{5.4}$$

$$\mathcal{I}^{\mathsf{N}_\lambda^j\mathsf{k}}_{\varphi^j}(\mathsf{N}_\lambda^j\xi)+\sum_{n=1}^d\mathcal{I}^{\mathsf{B}_\lambda^j\mathsf{E}_n\mathsf{k}}_{\Psi_j}(\mathsf{B}_\lambda^j\mathsf{E}_n\xi)=\mathcal{I}^{\mathsf{N}_\lambda^{j+1}\mathsf{k}}_{\varphi^{j+1}}(\mathsf{N}_\lambda^{j+1}\xi) \tag{5.5}$$

for $j\geqslant J_0$, a.e. $\xi\in\mathbb{R}^d$, and $\mathsf{k}\in([\mathsf{M}_\lambda^j\mathbb{Z}^d]\cup[\mathsf{M}_\lambda^{j+1}\mathbb{Z}^d]\cup\cup_{n=1}^d[\mathsf{E}_n\mathsf{A}_\lambda^j\mathbb{Z}^d])$, where $\mathcal{I}^{\mathsf{k}}_{\varphi^j},\mathcal{I}^{\mathsf{k}}_{\Psi_j}$ are defined as in (5.1).

In particular, if $\widehat{h}\geqslant 0$ for all $h\in\{\{\varphi^j\}\cup\{\Psi^j\}\}_{j\geqslant J_0}$, then any of items (1)–(3) is equivalent to (2.4)–(2.6).

Proof. The claim follows directly from [14, Theorem 13 and Corollary 12]. Since this result plays a central role in this paper, for the convenience of the reader, we provide a proof here by following lines developed in [14, Theorem 13].

Note that by our assumption on M_λ and A_λ, it is easy to show that

$$\{j\in\mathbb{Z}:j\geqslant J_0,[\mathsf{N}_\lambda^jB_c(0)]\cap\mathbb{Z}^d\neq\{0\}\}\quad\text{is a finite set for every } c\in[1,\infty), \tag{5.6}$$

where $B_c(x)$ denotes the d-dimensional ball of radius c centred at $x\in\mathbb{R}^d$.

(1)⇒(2) Consider (2.3) with two consecutive J and $J+1$ with $J\geqslant J_0$. Then the difference gives (5.3). Now by (5.3), it is easy to deduce that

$$\sum_{\mathsf{k}\in\mathbb{Z}^d}|\langle f,\varphi^{J'}_{\mathsf{M}_\lambda^{J'};\mathsf{k}}\rangle|^2=\sum_{\mathsf{k}\in\mathbb{Z}^d}|\langle f,\varphi^{J}_{\mathsf{M}_\lambda^{J};\mathsf{k}}\rangle|^2+\sum_{j=J}^{J'-1}\sum_{n=1}^d\sum_{\vec{\ell}=-\vec{s}_j}^{\vec{s}_j}\sum_{\mathsf{k}\in\mathbb{Z}^d}|\langle f,\psi^{j,\vec{\ell}}_{\mathsf{S}^{-\vec{\ell}}\mathsf{A}_\lambda^j\mathsf{E}_n;\mathsf{k}}\rangle|^2\ \forall J'\geqslant J. \tag{5.7}$$

Therefore, by (2.3) and letting $J'\to\infty$, we see that (5.2) holds.

(2)⇒(1) By (5.3), we deduce that (5.7) holds. By letting $J'\to\infty$ and in view of (5.2), we conclude that (2.3) holds.

(2)⇔(3) By [14, Lemma 10], we can show that (5.3) is equivalent to

$$\int_{\mathbb{R}^d}\sum_{\mathsf{k}\in\Lambda_j}\widehat{f}(\xi)\overline{\widehat{f}(\xi+2\pi\mathsf{k})}\mathcal{I}_j(\xi)d\xi=0, \tag{5.8}$$

where $\Lambda_j=[\mathsf{M}_\lambda^j\mathbb{Z}^d]\cup[\mathsf{M}_\lambda^{j+1}\mathbb{Z}^d]\cup\cup_{n=1}^d[\mathsf{E}_n\mathsf{A}_\lambda^j\mathbb{Z}^d]$ and

$$\mathcal{I}_j(\xi):=\Big[\mathcal{I}^{\mathsf{N}_\lambda^j\mathsf{k}}_{\varphi^j}(\mathsf{N}_\lambda^j\xi)+\sum_{n=1}^d\mathcal{I}^{\mathsf{B}_\lambda^j\mathsf{E}_n\mathsf{k}}_{\Psi_j}(\mathsf{B}_\lambda^j\mathsf{E}_n\xi)\Big]-\mathcal{I}^{\mathsf{N}_\lambda^{j+1}\mathsf{k}}_{\varphi^{j+1}}(\mathsf{N}_\lambda^{j+1}\xi).$$

Since $\mathsf{M}_\lambda = \lambda^2 \mathsf{I}_n$ and $\mathsf{A}_\lambda = \mathrm{diag}(\lambda^2, \lambda \mathsf{I}_{d-1})$ with $\lambda > 1$, we see that the lattice Λ_j is discrete. By the same argument as in the proof of [14, Theorem 13], we see that (5.8) is equivalent to (5.5).

By [14, Lemma 10], we see that (5.2) is equivalent to

$$\lim_{j\to\infty} \int_{\mathbb{R}^d} \sum_{\mathsf{k}\in[\mathsf{M}_\lambda^j\mathbb{Z}^d]} \widehat{f}(\xi)\overline{\widehat{f}(\xi+2\pi\mathsf{k})}\mathcal{I}_{\varphi^j}^{\mathsf{N}_\lambda^j\mathsf{k}}(\mathsf{N}_\lambda^j\xi) = \|f\|_2^2 \quad \forall \widehat{f} \in \mathscr{D}(\mathbb{R}^d). \tag{5.9}$$

Since $\widehat{f} \in \mathscr{D}(\mathbb{R}^d)$ is compactly supported, there exists $c > 0$ such that $\widehat{f}(\xi)\overline{\widehat{f}(\xi+2\pi\mathsf{k})} = 0$ for all $\xi \in \mathbb{R}^d$ and $|\mathsf{k}| > c$. By (5.6), there exists $J'' \geqslant J_0$ such that $\widehat{f}(\xi)\overline{\widehat{f}(\xi+2\pi\mathsf{k})}\mathcal{I}_{\varphi^j}^{\mathsf{N}_\lambda^j\mathsf{k}}(\mathsf{N}_\lambda^j\xi) = 0$ for all $\xi \in \mathbb{R}^d$, $\mathsf{k} \in [\mathsf{M}_\lambda^j\mathbb{Z}^d]\backslash\{0\}$, and $j \geqslant J''$. Consequently, for $j \geqslant J''$, (5.9) becomes

$$\lim_{j\to\infty} \int_{\mathbb{R}^d} |\widehat{f}(\xi)|^2 \mathcal{I}_{\varphi^j}^0(\mathsf{N}_\lambda^j\xi) = \|f\|_2^2 \quad \forall \widehat{f} \in \mathscr{D}(\mathbb{R}^d),$$

which is equivalent to (5.4).

When all generators are nonnegative in the frequency domain, by item (3) of Theorem 5.1, for $\mathsf{k} \in \mathbb{Z}^d\backslash\{0\}$, (5.5) is equivalent to (2.4). For $\mathsf{k} = 0$, (5.5) is equivalent to (2.5). Together with condition (2.6) and by item (3) of Theorem 5.1, the claim of Theorem 2.1 follows from the equivalence between item (1) and item (3) of Theorem 5.1. ∎

5.2. Proof of Theorem 2.2. In this subsection, we prove Theorem 2.2.

Proof. By our construction, we have for $\xi \in \mathbb{R}^d$,

$$\begin{aligned}|\widehat{\varphi}(\mathsf{N}_\lambda^j\xi)|^2 + \sum_{n=1}^d \sum_{\vec{\ell}=-\vec{\ell}_{\lambda j}}^{\vec{\ell}_{\lambda j}} |\widehat{\psi^{j,\vec{\ell}}}(\mathsf{S}_{\vec{\ell}}\mathsf{B}_\lambda^j\mathsf{E}_n\xi)|^2 &= |\widehat{\varphi}(\mathsf{N}_\lambda^j\xi)|^2 + \frac{|\boldsymbol{\omega}_{\lambda,t,\rho}(\mathsf{N}^j\xi)|^2}{\boldsymbol{\Gamma}_j(\xi)} \sum_{n=1}^d \sum_{\vec{\ell}=-\vec{\ell}_{\lambda j}}^{\vec{\ell}_{\lambda j}} |\boldsymbol{\gamma}^{j,\vec{\ell}}(\mathsf{E}_n\xi)|^2 \\ &= |\widehat{\varphi}(\mathsf{N}_\lambda^j\xi)|^2 + |\boldsymbol{\omega}_{\lambda,t,\rho}(\mathsf{N}^j\xi)|^2 = |\widehat{\varphi}(\mathsf{N}^{j+1}\xi)|^2.\end{aligned}$$

Hence, (2.5) holds. By the definition of φ, (2.6) also holds. Note that all generators satisfy $\widehat{\psi^{j,\vec{\ell}}} \geqslant 0$ and $\widehat{\varphi} \geqslant 0$. For φ, we have $\widehat{\varphi}(\xi)\widehat{\varphi}(\xi+2\pi\mathsf{k}) = 0$ for $\mathsf{k} \neq 0$ since $\mathrm{supp}\,\widehat{\varphi} \subseteq [-\rho\pi, \rho\pi]^2$ with $\rho \leqslant 1$. By (2.14), we have $\widehat{\psi^{j,\vec{\ell}}}(\xi) = \boldsymbol{\omega}_{\lambda,t,\rho}(\xi_1, \lambda^{-j}(\xi_2 - \ell_2\xi_1), \ldots, \lambda^{-j}(\xi_d - \ell_d\xi_1)) \frac{\prod_{n=2}^d \boldsymbol{\gamma}_\varepsilon(\xi_n/\xi_1)}{\sqrt{\boldsymbol{\Gamma}_j((\mathsf{S}_{\vec{\ell}}\mathsf{B}_\lambda^j)^{-1}\xi)}}$. Hence the support of $\widehat{\psi^{j,\vec{\ell}}}$ on ξ_1 is controlled by $[-\rho\pi, \rho\pi]$, while the support of $\widehat{\psi^{j,\vec{\ell}}}$ on ξ_n is controlled by that of $\boldsymbol{\gamma}_\varepsilon(\xi_n/\xi_1)$ for $n = 2, \ldots, d$. Now it is easy to show that $\mathrm{supp}\,\widehat{\psi^{j,\vec{\ell}}} \subseteq [-\rho\pi, \rho\pi] \times [-(1+2\varepsilon)\rho\pi, (1+2\varepsilon)\rho\pi]^{d-1} \subseteq [-\pi, \pi]^d$ by that $\rho \leqslant \frac{1}{1+2\varepsilon}$. Hence, $\widehat{\psi^{j,\vec{\ell}}}(\xi)\widehat{\psi^{j,\vec{\ell}}}(\xi + 2\pi\mathsf{k}) = 0$ for $\mathsf{k} \neq 0$. Consequently, (2.4) holds. Now, by Theorem 2.1, we conclude that $\mathsf{AS}(\varphi; \{\Psi_j\}_{j=J}^\infty)$ is an affine shear tight frame for $L_2(\mathbb{R}^d)$ for all $J \geqslant 0$. Since all generators $\varphi, \psi^{j,\vec{\ell}}$ are functions in $C^\infty(\mathbb{R}^d)$, all elements in $\mathsf{AS}(\varphi; \{\Psi_j\}_{j=J}^\infty)$ are functions in $C^\infty(\mathbb{R}^d)$.

By the definition of $\boldsymbol{\omega}_{\lambda,t,\rho}$, it is easy to see that

$$|\boldsymbol{\omega}_{\lambda,t,\rho}(\xi)|^2 = \Big|\boldsymbol{\beta}_{\lambda,t,\rho}(\xi_1)\prod_{n=2}^{d-1}\boldsymbol{\alpha}_{\lambda,t,\rho}(\xi_n)\Big|^2 + \Big|\boldsymbol{\beta}_{\lambda,t,\rho}(\xi_2)\boldsymbol{\alpha}_{\lambda,t,\rho}(\xi_1)\prod_{n=3}^{d-1}\boldsymbol{\alpha}_{\lambda,t,\rho}(\xi_n)\Big|^2 + \cdots + \Big|\prod_{n=1}^{d-1}\boldsymbol{\beta}_{\lambda,t,\rho}(\xi_n)\boldsymbol{\alpha}_{\lambda,t,\rho}(\xi_d)\Big|^2 + \Big|\prod_{n=1}^{d}\boldsymbol{\beta}_{\lambda,t,\rho}(\xi_n)\Big|^2.$$

When $\xi = (\xi_1,\dots,\xi_d) \in \mathbb{R}^d$ satisfies $|\xi_n| \leqslant \lambda^{-2}(1-t)\rho\pi$ for $n = 2,\dots,d$, by the definition of $\boldsymbol{\alpha}_{\lambda,t,\rho}$ and $\boldsymbol{\beta}_{\lambda,t,\rho}$, we have $\boldsymbol{\omega}_{\lambda,t,\rho}(\xi_1,\dots,\xi_n) = \boldsymbol{\beta}_{\lambda,t,\rho}(\xi_1)\prod_{n=2}^{d}\boldsymbol{\alpha}_{\lambda,t,\rho}(\xi_n) = \boldsymbol{\beta}_{\lambda,t,\rho}(\xi_1)$. Consequently, if the support $\{\xi = (\xi_1,\dots,\xi_d) \in \mathbb{R}^d : \xi \in \operatorname{supp}\widehat{(\psi^{j,\vec{\ell}})}_{\mathsf{S}_\ell\mathsf{B}_\lambda^j;0,\mathsf{k}}\}$ satisfies $|\xi_n| \leqslant \lambda^{2j-2}(1-t)\rho\pi$ for $n = 2,\dots,d$, then we have

$$\begin{aligned}\widehat{(\psi^{j,\vec{\ell}})}_{\mathsf{S}_{\vec{\ell}}\mathsf{B}_\lambda^j;0,\mathsf{k}}(\xi) &= \lambda^{-(d+1)j/2}\boldsymbol{\omega}_{\lambda,t,\rho}(\lambda^{-2j}\xi)\prod_{n=2}^{d}\boldsymbol{\gamma}_\varepsilon(\lambda^j\xi_n/\xi_1+\ell_n)\cdot e^{-i\mathsf{k}\cdot\mathsf{S}_{\vec{\ell}}\mathsf{B}_\lambda^j\xi}\\ &= \lambda^{-(d+1)j/2}\boldsymbol{\beta}_{\lambda,t,\rho}(\lambda^{-2j}\xi_1)\prod_{n=2}^{d}\boldsymbol{\gamma}_\varepsilon(\lambda^j\xi_n/\xi_1+\ell_n)\cdot e^{-i\mathsf{k}\cdot\mathsf{S}_{\vec{\ell}}\mathsf{B}_\lambda^j\xi}\\ &= \widehat{\psi}_{\mathsf{S}_{\vec{\ell}}\mathsf{B}_\lambda^j;0,\mathsf{k}}(\xi).\end{aligned}$$

Now let us find the range of $\vec{\ell}$ such that the support constraint holds. At scale j, we have

$$\operatorname{supp}\boldsymbol{\omega}_{\lambda,t,\rho}(\lambda^{-2j}\cdot) \subseteq [-\lambda^{2j}\rho\pi,\lambda^{2j}\rho\pi]^d\backslash[-\lambda^{2j-2}(1-t)\rho\pi,\lambda^{2j-2}(1-t)\rho\pi]^d.$$

Then, the support constraint means that at scale j, one needs $|\xi_n/\xi_1| \leqslant \lambda^{-2}(1-t)\rho$ for all $n = 2,\dots,d$. Hence, the support of $\boldsymbol{\gamma}_\varepsilon(\lambda^j\xi_n/\xi_1+\ell_n)$ must satisfy

$$-\lambda^{-2}(1-t)\rho \leqslant -\lambda^{-j}(1/2+\varepsilon+\ell_n) \leqslant \xi_n/\xi_1 \leqslant \lambda^{-j}(1/2+\varepsilon-\ell_n) \leqslant \lambda^{-2}(1-t)\rho.$$

Consequently, we obtain

$$-\lambda^{j-2}(1-t)\rho + (1/2+\varepsilon) \leqslant \ell_n \leqslant \lambda^{j-2}(1-t)\rho - (1/2+\varepsilon).$$

That is, $|\ell_n| \leqslant \lambda^{j-2}(1-t)\rho - (1/2+\varepsilon)$. In summary, letting $r_j := \lfloor\lambda^{j-2}(1-t)\rho - (1/2+\varepsilon)\rfloor$ and defining $\vec{\ell}$ and $\vec{r}_j$ as in the theorem, we have

$$\{\psi(\mathsf{S}^{-\vec{\ell}}\cdot) : |\vec{\ell}| \leqslant \vec{r}_j\} \subseteq \Psi_j,\ j \geqslant J,$$

and

$$\{\psi_{\mathsf{S}^{-\vec{\ell}}\mathsf{A}_\lambda^j\mathsf{E}_n;\mathsf{k}} : \mathsf{k} \in \mathbb{Z}^d, |\vec{\ell}| \leqslant \vec{r}_j\}_{j=J}^\infty \subseteq \mathsf{AS}(\varphi;\{\Psi_j\}_{j=J}^\infty).$$

We are done. ∎

Acknowledgments. We would like to thank anonymous referees for their valuable comments and suggestion that helped improve the paper.

REFERENCES

[1] B. G. Bodmann, G. Kutyniok, and X. Zhuang, *Gabor shearlets*, Appl. Comput. Harmon. Anal., 38 (2015), pp. 87–114.

[2] E. J. Candès and D. L. Donoho, *New tight frames of curvelets and optimal representations of objects with piecewise C^2 singularities*, Comm. Pure Appl. Math., 57 (2004), pp. 219–266.

[3] E. J. Candès, L. Demannet, D. Donoho, and L. Ying, *Fast discrete curvelet transforms*, Multiscale Model. Simul., 5 (2006), pp. 861–899.

[4] C. K. Chui, *An Introduction to Wavelets*, Academic Press, New York, 1992.

[5] A. L. Cunha, J. Zhou, and M. N. Do, *The nonsubsampled contourlet transform: Theory, design, and applications*, IEEE Trans. Image Process., 15 (2006), pp. 3089–3101.

[6] I. Daubechies, *Ten Lectures on Wavelets*, CBMS-NSF Regional Conf. Ser. in Appl. Math. 61, SIAM, Philadelphia, PA, 1992.

[7] D. L. Donoho, *Sparse components of images and optimal atomic decomposition*, Constr. Approx., 17 (2001), pp. 353–382.

[8] G. Easley, D. Labate, and W. Lim, *Sparse directional image representations using the discrete shearlet transform*, Appl. Comput. Harmon. Anal., 25 (2008), pp. 25–46.

[9] K. Guo, D. Labate, W. Lim, G. Weiss, and E. Wilson, *Wavelets with composite dilations*, Electron. Res. Announ. AMS, 10 (2004), pp. 78–87.

[10] K. Guo, G. Kutyniok, and D. Labate, *Sparse multidimensional representations using anisotropic dilation and shear operators*, in Wavelets and Splines (Athens, GA, 2005), Nashboro Press, Nashville, TN, 2006, pp. 189–201.

[11] K. Guo and D. Labate, *Optimal sparse multidimensional representation using shearlets*, SIAM J. Math. Anal., 9 (2007), pp. 298–318.

[12] K. Guo and D. Labate, *The construction of smooth Parseval frames of shearlets*, Math. Model. Nat. Phenom., 8 (2013), pp. 82–105.

[13] B. Han, *On dual wavelet tight frames*, Appl. Comput. Harmon. Anal., 4 (1997), pp. 380–413.

[14] B. Han, *Nonhomogeneous wavelet systems in high dimensions*, Appl. Comput. Harmon. Anal., 32 (2012), pp. 169–196.

[15] B. Han, *Properties of discrete framelet transforms*, Math. Model. Nat. Phenom., 8 (2013), pp. 18–47.

[16] S. Häuser and G. Steidl, *Fast Finite Shearlet Transform: A Tutorial*, arXiv:1202.1773, 2014.

[17] B. Han and Z. Zhao, *Tensor product complex tight framelets with increasing directionality*, SIAM J. Imaging Sci., 7 (2014), pp. 997–1034.

[18] B. Han, Z. Zhao, and X. Zhuang, *Directional tensor product complex tight framelets with low redundancy*, Appl. Comput. Harmon. Anal., 41 (2016), pp. 603–637.

[19] B. Han and X. Zhuang, *Smooth affine shear tight frames with MRA structures*, Appl. Comput. Harmon. Anal., 39 (2015), pp. 300–338.

[20] E. J. King, G. Kutyniok, and X. Zhuang, *Analysis of data separation and recovery problems using clustered sparsity*, J. Math. Imaging Vision, 48 (2014), pp. 205–234.

[21] G. Kutyniok, M. Sharam, and X. Zhuang, *ShearLab: A rational design of a digital parabolic scaling algorithm*, SIAM J. Imaging Sci., 5 (2012), pp. 1291–1332.

[22] W.-Q. Lim, *Nonseparable shearlet transform*, IEEE Trans. Image Process., 22 (2013), pp. 2056–2065.

[23] Y. M. Lu and M. N. Do, *Multidimensional directional filter banks and surfacelets*, IEEE Trans. Image Process., 16 (2007), pp. 918–931.

[24] S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, New York, 2008.

[25] P. S. Negi and D. Labate, *3-D discrete shearlet transform and video processing*, IEEE Trans. Image Process., 21 (2012), pp. 2944–2954.

[26] I. W. Selesnick, R. G. Baraniuk, and N. G. Kingsbury, *The dual-tree complex wavelet transform*, IEEE Signal Process. Mag., 22 (2005), pp. 123–151.

[27] Y. Shen, B. Han, and E. Braverman, *Image Inpainting Using Directional Tensor Product Complex Tight Framelets*, arXiv:1407.3234, 2014.

[28] X. Zhuang, *Smooth affine shear tight frames: Digitization and applications*, in Wavelets and Sparsity XVI, Proc. SPIE 9597, SPIE, Bellingham, WA, 2015.