# Digital Affine Shear Filter Banks with 2-Layer Structure and Their Applications in Image Processing

Zhihua Che and Xiaosheng Zhuang
Department of Mathematics, City University of Hong Kong, Tat Chee Avenue, Kowloon Tong, Hong Kong
Email: zhihuache2-c@my.cityu.edu.hk, xzhuang7@cityu.edu.hk

***Abstract*—Digital affine shear filter banks with 2-layer structure (DAS-2 filter banks) are constructed and are shown to be with the perfect reconstruction (PR) property. The implementation of digital affine shear transforms using the transition and subdivision operators are given. The redundancy rate analysis shows that our digital affine shear transforms have redundancy rate no more than $8$ and it *decreases* with respect to the number of directional filters. Numerical experiments on image processing demonstrate the advantage of our DAS-2 filter banks over many other state-of-the-art frame-based transforms. The connection between DAS-2 filter banks and affine shear tight frames with 2-layer structure is established. Characterizations and constructions of affine shear tight frames with 2-layer structure are provided.**

## I. Introduction and Motivations

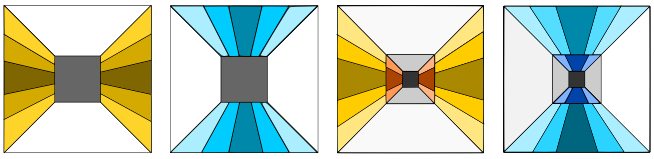
Given a 2D image, e.g., `Barbara` (see Fig. 7), it usually consists of three main parts of features: smooth areas, edges, and oscillating patterns. A key step in image processing is to capture these important features as precise as possible, which then play a crucial role in further applications such as pattern recognition, computer vision, machine learning, etc.

In the image domain, smooth areas are usually separated by edges with various orientations. Oscillating patterns can fill inside a smooth area as background noise, textures, and so on. Edges could be corrupted by noise and dissolved inside smooth areas. Though geometrically they are essentially different, all these features could mix together due to the presence of noise. In the frequency domain, smooth areas are with respect to low-frequency features. Edge information are concentrated along certain directions but could spread over the whole spectrum. Oscillating patterns are corresponding to high-frequency features. Moreover, from the point of view of time-scale analysis, smooth areas are simple structures that can be easily regarded as very coarse scale information while edges and oscillating patterns could appear from very coarse to very fine scales. In order to capture as many types of features as possible, one necessarily needs to use multiscale representation systems that have both good frequency separability and nice directional selectivity.

With respect to applications of wavelets [8] and [27], multiresolution/multiscale analysis has become an important approach and an indispensable tool in many applications. For 1D signals with singularities, wavelets have been proved to provide optimal approximation rate due to their ability of capturing singularities using wavelet coefficients at different scales. However, for 2D images, singularities are edge-like structures with various orientations which could come from the wavefront sets of certain partial differential equations [22]. Representation of such edge-like singularities requires huge amount of active wavelet coefficients due to the lack of directional selectivity if only tensor-product approach is considered, which results in poor approximate rate of cartoon-like functions in 2D [9]. The introduction of curvelets [3] solves such an issue by incorporating the rotation operations into the wavelet systems in addition to the (parabolic) dilation and translation operations. Since then, directional multiscale representation systems, e.g., contourlets [6] and shearlets [11], [12], have been shown to be superb over many other multiscale representation systems, such as the tensor-product real-value orthonormal wavelets, dual-tree complex wavelets [29] in both theory (sparse approximation for cartoon-like functions) and applications (denoising or inpainting); e.g., see [2], [4], [10], [13], [14], [20], [23], [24], [29], etc. In particularly, shearlets, which replace the rotation operations by shear operations, have gained a lot of attention due to their additional nice properties such as integer lattice preservation, rich group structure [1], coorbit space theory [7].

In practice, the applications of directional multiscale representation systems are based on their underlying (2D) directional filter banks, e.g., shearlet filter banks, that can be written as $\cup_{j=0}^{J-1}\{a; b_X^{j,\ell}, b_Y^{j,\ell}, \ell = 1, \ldots, s_j\}$, where at level $j$, the frequency domain $\mathbb{T}^2 := [-\pi, \pi)^2$ is tiled by a low-pass filter $a$ (middle squares in Fig. 1), some directional high-pass filters $b_X^{j,\ell}$ in the horizontal cone $X$, and some directional high-pass filters $b_Y^{j,\ell}$ in the vertical cone $Y$. Fig. 1(a)–(b) illustrates the frequency tiling of such a directional filter bank (e.g., DAS-1 in [32]). Each (colored) wedge-shaped area in Fig. 1(a)–(b) is the frequency response of a directional filter. It is no doubt that such kinds of filter banks are capable of capturing edge-like information and have been shown to perform very well in image processing; see [6], [16], [25], [26], etc.

However, as mentioned, other than edge features, an image could also contain oscillating patterns such as textures. Directional filter banks with frequency tiling in Fig. 1(a)–(b) might put too much weight on the edge-like structures while at the same time ignores the oscillating patterns of an image. Taking Fig. 2 for example, the image in Fig. 2(a) has a tablecloth



(a) cone $X$ (b) cone $Y$ (c) cone $X$ (d) cone $Y$

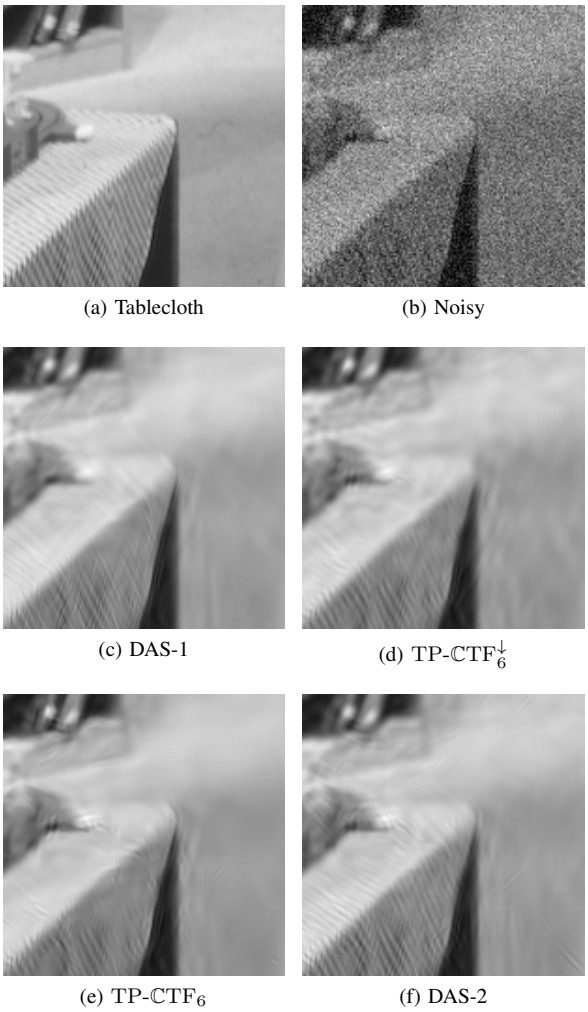
Fig. 1: (a) Frequency tiling of DAS-1 in [32] (horizontal cone $X$ in $\mathbb{T}^2$). (b) Frequency tiling of DAS-1 in [32] (vertical cone $Y$). (c) Frequency tiling of DAS-2 in this paper (horizontal cone $X$, 2-layer high-pass filters). (d) Frequency tiling of DAS-2 in this paper (vertical cone $Y$, 2-layer high-pass filters).

(cropped from the image `Barbara`) containing line texture with different orientations including patterns of the form \ \ \, / / /, and their mixture. In Fig. 2(b), the image is corrupted by Gaussian noise with noise variance $\sigma = 30$ and its line texture almost disappears. Using directional filter bank from [32] which performs standard denoising process to the noisy image, one can recover most of the information such as background and sharp edges as shown in Fig. 2(c). However, large portion of the line texture (\ \ \ patterns) on the tablecloth could not be recovered by this filter bank. One of the reasons is its lack of representing highly oscillating patterns.

In order to capture both edge-like features and highly oscillating patterns, the papers [17] and [18] introduce tensor product complex tight framelets TP-$\mathbb{C}$TF$_6$ and TP-$\mathbb{C}$TF$_6^{\downarrow}$, which have many nice properties including simple way of filter bank design, directionality, tensor product structure, etc. Here is the main idea: TP-$\mathbb{C}$TF$_6$ or TP-$\mathbb{C}$TF$_6^{\downarrow}$ is generated by the tensor product of a 1D filter bank $\{a^+, a^-; b_1^+, b_1^-, b_2^+, b_2^-\}$ having nice frequency splitting property in the sense that $a^+, a^-$ are concentrated on the positive and negative low-frequency part (near 0) of $\mathbb{T}$, respectively, while $b_i^+, b_i^-$ are concentrated on the high-frequency part (near $\pm\pi$) of positive and negative axis of $\mathbb{T}$, respectively for $i = 1, 2$. In such a case, the directional high-pass filters of TP-$\mathbb{C}$TF$_6$ or TP-$\mathbb{C}$TF$_6^{\downarrow}$ have a very nice 2-layer structure (Fig. 3). The middle white square in Fig. 3(a) is tiled by the low-pass filters from those $a^{\pm} \otimes a^{\pm}$. The inner layer squares (see Fig. 3(b)) are with respect to those high-pass filters coming from $a^{\pm} \otimes b_1^{\pm}$, $b_1^{\pm} \otimes a^{\pm}$, and $b_1^{\pm} \otimes b_1^{\pm}$. They are usually edge-like filters so that they can be used to capture edge-like structures. The outer layer squares (see Fig. 3(c)) are with respect to high-pass filters coming from $b_2^{\pm} \otimes h$ and $h \otimes b_2^{\pm}$ for all $h \in \{a^{\pm}, b_1^{\pm}, b_2^{\pm}\}$. Since they are located more far away from the low frequency spectrum than those inner layer filters, they look more 'oscillating' and hence are suitable for 'texture-like' structure.

The numerical results in [17], [18] confirm the advantages of such filter banks in image/video processing. However, due to the tensor product approach of TP-$\mathbb{C}$TF$_6$ and TP-$\mathbb{C}$TF$_6^{\downarrow}$, they also have some disadvantages. First, the number of directional high-pass filters at each level is limited and fixed (14 directions), which is undesirable in practice especially when the resolution of an image is very high that requires large number of directional filters in order to capture as many features with different orientations as possible. Second, though from their frequency tiling (Fig. 3), each high-pass filter does have certain directionality, yet in the time domain it is not well-localized along certain direction due to their square-like frequency response. Last but not least, due to the fixed number of 1D filters $\{a^{\pm}, b_1^{\pm}, b_2^{\pm}\}$, the number of free parameters is limited which prevents the search of optimal filter bank systems for image processing. Indeed, such disadvantages can be seen from Fig. 2(d)–(e), which shows the denoised results for the corrupted image in Fig. 2(b). One can see that due to the limitation of directions, large portion of line texture (\ \ \ patterns as well as some / / / patterns) on the tablecloth is still missing using either TP-$\mathbb{C}$TF$_6$ or TP-$\mathbb{C}$TF$_6^{\downarrow}$.

Motivated by the successful applications of affine shear tight frames in [16], [32], and the tensor product complex tight framelets for image/video processing in [17], [18], in this

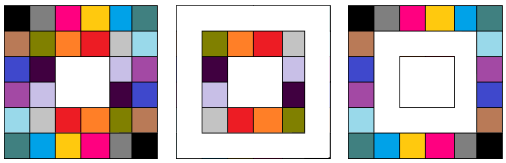


(a) Tablecloth (b) Noisy

(c) DAS-1 (d) TP-$\mathbb{C}$TF$_6^{\downarrow}$

(e) TP-$\mathbb{C}$TF$_6$ (f) DAS-2

Fig. 2: (a) Tablecloth (cropped from the image `Barbara`). (b) Corrupted by Gaussia noise ($\sigma = 30$). (c) Denoised by DAS-1 in [32]. (d) Denoised by TP-$\mathbb{C}$TF$_6^{\downarrow}$ [18]. (e) Denoised by TP-$\mathbb{C}$TF$_6$ [17]. (f) Denoised by DAS-2 in this paper.



(a) Freq. Tiling (b) Inner Layer (c) Outer Layer

Fig. 3: (a) Frequency tiling of TP-$\mathbb{C}$TF$_6$ or TP-$\mathbb{C}$TF$_6^{\downarrow}$ in [18]. (b) Inner layer filters (colored squares). (c) Outer layer filters (colored squares). Each colored square represents the frequency response of a high-pass filter.

paper, we remedy the aforementioned issues by introducing affine shear tight frames with 2-layer structure and focus on the development and applications of digital affine shear filter banks with 2-layer structure. The main idea is to incorporate the 2-layer structure in [17], [18] into the directional filter bank introduced in [16], [32]. Fig. 1(c)–(d) illustrates the the frequency tiling of our affine shear filter banks with 2-layer structure (DAS-2). At each level $j$, the frequency domain $\mathbb{T}^2$ is split into horizontal cone $X$ and vertical cone $Y$ as those of DAS-1 in [16], [32]. For each cone, in addition to the outer layer of directional high-pass filters, we further split the low-frequency domain so that it has an additional inner layer of directional high-pass filters. In such a way, we obtain a flexible directional filter bank that is capable of capturing not only edge-like features, but also oscillating patterns. The 2-layer idea seems very simple yet it yields striking improvement comparing to many state-of-the-art directional frame approaches. See Fig. 2(f) for the denoised result of our DAS-2 for the corrupted image in Fig. 2(b). One can clearly see that our DAS-2 can indeed recover a large portion of line texture on the tablecloth (including both the \\\ patterns and / / / patterns), which is also confirmed by the PSNR values of the denoised results (See more details in Section V).

The contributions of our paper lie in the following aspects. First, we introduce digital affine shear filter banks with 2-layer structure which have never been considered in the directional shearlet filter banks literature, as far as we concern. The 2-layer structure allows one to tailer-make their own DAS-2 filter banks and provides a natural bridge between edges and textures. Second, we provide analysis, characterization, and construction of affine shear tight frames with 2-layer structure. Last but not least, we demonstrate that our DAS-2 filter banks enjoy many desirable properties including arbitrary number of directions (limited by image resolution), 2-layer structure for edge- and texture-like features, low redundancy rate, fast transform algorithms, etc. The DAS-2 filter banks brings unique and indispensable features which cannot be achieved by other frame-based filter bank systems as well as significant improvements in many applications as shown but not restricted by those in the paper. The fast algorithmic transforms together with the low redundancy rate of DAS-2 filter banks make them suitable for massive data processing, especially in machine learning and pattern recognition.

The structure of the paper is as follows. In Section II, we provide the details for the construction of DAS-2 filter banks. In Section III, based on our DAS-2 filter banks, we present the digital affine shear transforms (both forward transform for decomposition and backward transform for reconstruction) based on the transition and subdivision operators. In Section IV, analysis of redundancy rate and computational complexity shows that our DAS-2 filter bank transforms have redundancy rate no more than 8 and $\mathcal{O}(N \log N)$ computational complexity with input date size $N$. Numerical experiments are conducted in Section V while the connection between our DAS-2 filter bank and affine shear tight frames with 2-layer structure is given in Section VI.

## II. Digital Affine Shear Filter Banks with 2-Layer Structure

In this section, we present the detailed construction of our digital affine shear filter banks with 2-layer structure $\mathcal{DAS}_J(\{a; \mathcal{B}^{j,in}, \mathcal{B}^{j,out}\}_{j=0,\dots,J-1})$, where at level $j$, $a$ is the low-pass filter while $\mathcal{B}^{j,\iota} = \mathcal{B}_X^{j,\iota} \cup \mathcal{B}_Y^{j,\iota}$ for $\iota \in \{in, out\}$ with

$$\mathcal{B}_X^{j,\iota} = \{b_X^{j,\ell,\iota,\pm}\}_{\ell=-2^{k_j^\iota}}^{2^{k_j^\iota}} \text{ and } \mathcal{B}_Y^{j,\iota} = \{b_Y^{j,\ell,\iota,\pm}\}_{\ell=-2^{k_j^\iota}}^{2^{k_j^\iota}}$$

is the set of high-pass filters in the horizontal cone $X$ (Fig. 1(c)) and vertical cone $Y$ (Fig. 1(d)), respectively. Here, $\iota = in$ is to indicate inner layer filters and $\iota = out$ for outer layer ones. Note that a 2-dimensional filter $u$ can be regarded in the image domain as a sequence in $l_1(\mathbb{Z}^2) := \{u : \mathbb{Z}^2 \to \mathbb{C} : \sum_k |u(k)| < \infty\}$ or in the frequency domain as a $2\pi\mathbb{Z}^2$-periodic function $\widehat{u}(\xi) := \sum_{k\in\mathbb{Z}^2} u(k) e^{-ik\cdot\xi}$, $\xi \in \mathbb{T}^2$. We build our DAS-2 filter banks from the frequency domain to have the 2-layer structure as in Fig. 1(c)–(d) (see also Fig. 4).

We first define an auxiliary 1D "bump" function $\boldsymbol{\nu}_{[c,\epsilon]}$ supported on $[-c-\epsilon, c+\epsilon]$. Let $\nu$ be a function such that $\nu(x) = 0$ for $x \le -1$, $\nu(x) = 1$ for $x \ge 1$, and $|\nu(x)|^2 + |\nu(-x)|^2 = 1$ for all $x \in \mathbb{R}$. Such a function can be constructed to be smooth in $C^\infty(\mathbb{R})$ or differentiable in $C^k(\mathbb{R})$; see [16], [32]. Define $\boldsymbol{\nu}_{[c,\epsilon]}(x) := \nu(\frac{x+c}{\epsilon})$ for $x \le -c+\epsilon$ and $\boldsymbol{\nu}_{[c,\epsilon]}(x) := \nu(\frac{-x+c}{\epsilon})$ for $x > c-\epsilon$.

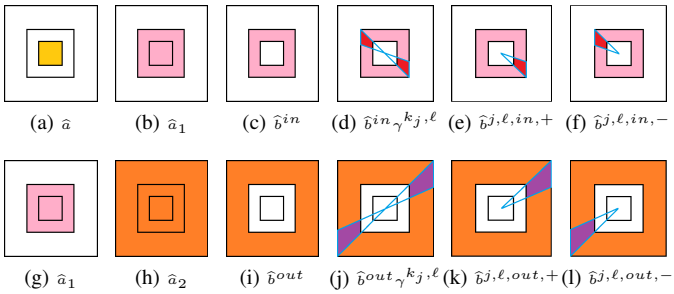
We then define the inner, middle, outer functions $\widehat{a}, \widehat{a}_1, \widehat{a}_2 \in C(\mathbb{R}^2)$ (see Fig. 4(a), (b), and (h), respectively) by

$$\begin{aligned} \widehat{a} :=& \boldsymbol{\nu}_{[c_0,\epsilon_0]} \otimes \boldsymbol{\nu}_{[c_0,\epsilon_0]}, \\ \widehat{a}_1 := \boldsymbol{\nu}_{[c_1,\epsilon_1]} \otimes& \boldsymbol{\nu}_{[c_1,\epsilon_1]},\ \widehat{a}_2 := \boldsymbol{\nu}_{[c_2,\epsilon_2]} \otimes \boldsymbol{\nu}_{[c_2,\epsilon_2]}, \end{aligned} \tag{1}$$

for some parameters $0 < c_0 < c_1 < c_2 = \pi$ and $\epsilon_0, \epsilon_1, \epsilon_2 > 0$ satisfying $c_0 + \epsilon_0 \le \pi/2$ (for downsampling by 2), $(c_1+\epsilon_1) - (c_0 - \epsilon_0) \le \pi/2$ (for downsampling by 4), and $c_1 - \epsilon_1 - \epsilon_2 \ge \pi/2$ (for downsampling by 4). We identify the function $\widehat{a}$ as a function in $C(\mathbb{T}^2)$ in frequency domain. In the image domain, it then serves as our low-pass filter $a$. The filter $\widehat{a}$ is supported inside $[-\pi/2, \pi/2]^2$. The other two functions $\widehat{a}_1, \widehat{a}_2$ are auxiliary functions for building the inner and outer layer high-pass filters. One can show that

$$\sum_{\mathsf{k}\in\mathbb{Z}^2} |\widehat{a}_2(\xi + 2\pi\mathsf{k})|^2 = 1, \qquad \xi \in \mathbb{T}^2. \tag{2}$$

Define the inner and outer functions $b^{in}, b^{out}$ by $\widehat{b}^{out} := \sqrt{|\widehat{a}_2|^2 - |\widehat{a}_1|^2}$ and $\widehat{b}^{in} := \sqrt{|\widehat{a}_1|^2 - |\widehat{a}|^2}$ (see Fig. 4(c) and (i)). The function $\widehat{b}^{out}$ is supported on the outer layer $[-c_2-\epsilon_2, c_2+\epsilon_2]^2\backslash[-c_1+\epsilon_1, c_1-\epsilon_1]^2$ while the function $\widehat{b}^{in}$ is supported on the inner layer $[-c_1-\epsilon_1, c_1+\epsilon_1]^2\backslash[-c_0+\epsilon_0, c_0-\epsilon_0]^2$. We then apply the splitting technique to $\widehat{b}^{\iota}$ for the construction of high-pass filters $b^{j,\ell,\iota}$ by using a splitting auxiliary function $\boldsymbol{\gamma}_\varepsilon := \boldsymbol{\nu}_{[1/2,\varepsilon]}$ for $0 < \varepsilon \le 1/2$. This splitting process is illustrated in Fig. 4(a)–(l).



(a) $\widehat{a}$ (b) $\widehat{a}_1$ (c) $\widehat{b}^{in}$ (d) $\widehat{b}^{in}\gamma^{k_j,\ell}$ (e) $\widehat{b}^{j,\ell,in,+}$ (f) $\widehat{b}^{j,\ell,in,-}$

(g) $\widehat{a}_1$ (h) $\widehat{a}_2$ (i) $\widehat{b}^{out}$ (j) $\widehat{b}^{out}\gamma^{k_j,\ell}$ (k) $\widehat{b}^{j,\ell,out,+}$ (l) $\widehat{b}^{j,\ell,out,-}$

Fig. 4: DAS-2 filter bank construction. (a) $\widehat{a}$. (b) $\widehat{a}_1$. (c) $\widehat{b}^{in} = \sqrt{|\widehat{a}_1|^2 - |\widehat{a}|^2}$. (d) Splitting of $\widehat{b}^{in}$ to $\widehat{b}^{in}\gamma^{k_j,\ell}$, which is further split to positive part and negative part to form (e) $\widehat{b}^{j,\ell,in,+} = \widehat{b}^{in}\gamma^{k_j,\ell,+}$ and (f) $\widehat{b}^{j,\ell,in,-} = \widehat{b}^{in}\gamma^{k_j,\ell,-}$. (g) $\widehat{a}_1$. (h) $\widehat{a}_2$. (i) $\widehat{b}^{out} = \sqrt{|\widehat{a}_2|^2 - |\widehat{a}_1|^2}$. (j) Splitting of $\widehat{b}^{out}$ to $\widehat{b}^{out}\gamma^{k_j,\ell}$, which is further split to positive part and negative part to form (k) $\widehat{b}^{j,\ell,out,+} = \widehat{b}^{out}\gamma^{k_j,\ell,+}$ and (l) $\widehat{b}^{j,\ell,out,-} = \widehat{b}^{out}\gamma^{k_j,\ell,-}$.

.

In practice, at level $j$, we use integer $2^{k_j}$ to control the number of directions. Define for $\xi = (\xi_1, \xi_2) \in \mathbb{R}^2$,

$$\begin{aligned} \boldsymbol{\gamma}^{k_j,\ell}(\xi) &:= \boldsymbol{\gamma}_\varepsilon\left(2^{k_j}\frac{\xi_2}{\xi_1} + \ell\right), \\ \boldsymbol{\Gamma}_{k_j}(\xi) &:= \sum_{\ell=-2^{k_j}}^{2^{k_j}} \left(|\boldsymbol{\gamma}^{k_j,\ell}(\mathsf{E}_X\xi)|^2 + |\boldsymbol{\gamma}^{k_j,\ell}(\mathsf{E}_Y\xi)|^2\right), \end{aligned} \tag{3}$$

where $\mathsf{E}_X = \mathsf{I}_2 := \left[\begin{smallmatrix}1 & 0\\ 0 & 1\end{smallmatrix}\right]$ is the identity matrix, $\mathsf{E}_Y := \left[\begin{smallmatrix}0 & 1\\ 1 & 0\end{smallmatrix}\right]$ is the elementary matrix. To guarantee smoothness of boundary, we further split $\boldsymbol{\gamma}^{k_j,\ell}(\xi)$ to positive part $\boldsymbol{\gamma}^{k_j,\ell,+}$ and negative part $\boldsymbol{\gamma}^{k_j,\ell,-}$ of $\xi_1$-axis by

$$\boldsymbol{\gamma}^{k_j,\ell,+}(\xi) := \boldsymbol{\gamma}^{k_j,\ell}(\xi)\chi_{\{\xi_1>0\}}, \boldsymbol{\gamma}^{k_j,\ell,-}(\xi) := \boldsymbol{\gamma}^{k_j,\ell}(\xi)\chi_{\{\xi_1<0\}}.$$

Here, $\boldsymbol{\gamma}^{k_j,\ell,\tau}$ for $\tau \in \{+,-\}$ are the splitting elements while $\boldsymbol{\Gamma}_{k_j}$ serves as the role of normalization for the tightness purpose; see Section VI.

Now at level $j$, given integers $k_j^{in}, k_j^{out}$ for determining the number of directions, we can obtain functions $\widehat{b}^{\iota}(\xi)\frac{\boldsymbol{\gamma}^{k_j^\iota,\ell,\tau}(\xi)}{\sqrt{\boldsymbol{\Gamma}_{k_j^\iota}(\xi)}}$ concentrating along certain directions in the inner layer for $\iota = in$ and the outer layer for $\iota = out$, respectively (see Fig. 4(d) and (j)). Note that $\widehat{b}^{\iota}(\xi)\frac{\boldsymbol{\gamma}^{k_j^\iota,\ell,\tau}(\xi)}{\sqrt{\boldsymbol{\Gamma}_{k_j^\iota}(\xi)}}$ may not be $2\pi\mathbb{Z}^2$-periodic functions and hence are not filters. We define $b_X^{j,\ell,\iota,\tau}$ to be the $2\pi\mathbb{Z}^2$-periodization of $\widehat{b}^{\iota}(\xi)\frac{\boldsymbol{\gamma}^{k_j^\iota,\ell,\tau}(\xi)}{\sqrt{\boldsymbol{\Gamma}_{k_j^\iota}(\xi)}}$ as $\widehat{b_X^{j,\ell,\iota,\tau}}(\xi) := \sum_{\mathsf{k}\in\mathbb{Z}^2}\widehat{b}^{\iota}(\xi+2\pi\mathsf{k})\frac{\boldsymbol{\gamma}^{k_j^\iota,\ell,\tau}(\xi+2\pi\mathsf{k})}{\sqrt{\boldsymbol{\Gamma}_{k_j^\iota}(\xi+2\pi\mathsf{k})}}$ for $\xi \in \mathbb{T}^2$. These are the inner and outer layer high-pass filters in the horizontal cone $X$. For the vertical cone $Y$, we define $\widehat{b_Y^{j,\ell,\iota,\tau}}(\xi) := \widehat{b_X^{j,\ell,\iota,\tau}}(\mathsf{E}_Y\xi)$ as their flipped version filters. Note that each high-pass filter $b_o^{j,\ell,\iota,\tau}$ is of complex value (see Fig. 6 for examples of high-pass filters in inner and outer layers). The filter bank

$$\begin{aligned} \{a; \mathcal{B}^{j,in}, \mathcal{B}^{j,out}\} = \{a; b_o^{j,\ell,\iota,\tau} : \ell = -2^{k_j^\iota}, \dots, 2^{k_j^\iota}, \\ \tau \in \{+,-\}, \iota \in \{in,out\}, o \in \{X,Y\}\} \end{aligned} \tag{4}$$

is then said to be a *digital affine shear filter bank with 2-layer structure* (DAS-2 filter bank).

In view of (2) and (3), one can easily prove the following result, which we omit and refer to [32, Theorem 3.1].

*Theorem 1:* Retaining notations in this section and assuming $c_k, \epsilon_k, \varepsilon$ for $k = 0,1,2$ satisfying $0 < c_0 < c_1 < c_2 = \pi$, $c_0 + \epsilon_0 \le \pi/2$, $(c_1+\epsilon_1) - \frac{c_0-\epsilon_0}{3/2+\varepsilon} \le \pi/2$, $\frac{c_1-\epsilon_1}{3/2+\varepsilon} - \epsilon_2 \ge \pi/2$, and $0 < \varepsilon \le \frac{\pi}{c_2+\epsilon_2} - \frac{1}{2}$, then the filter bank $\{a; \mathcal{B}^{j,in}, \mathcal{B}^{j,out}\}$ forms a DAS-2 filter bank with the PR property:

$$|\widehat{a}(\xi)|^2 + \sum_{\tau,\iota,o}\sum_{\ell=-2^{k_j^\iota}}^{2^{k_j^\iota}} \left|\widehat{b_o^{j,\ell,\iota,\tau}}(\xi)\right|^2 = 1, \tag{5}$$

$$\widehat{a}(\xi)\widehat{a}(\xi+2\pi\omega) = 0,\ \widehat{b_o^{j,\ell,\iota,\tau}}(\xi)\widehat{b_o^{j,\ell,\iota,\tau}}(\xi+2\pi\omega_o) = 0, \tag{6}$$

for all $\xi \in \mathbb{T}^2$, $|\ell| \le 2^{k_j^\iota}$, $\omega \in [\frac{1}{2}\mathbb{Z}^2]\cap[0,1)^2\backslash\{0\}$, and $\omega_o \in [(\mathsf{A}_o^{j,\iota})^{-\mathsf{T}}\mathbb{Z}^2]\cap[0,1)^2\backslash\{0\}$, where $\iota \in \{in,out\}$, $\tau \in \{+,-\}$, $o \in \{X,Y\}$, and $\mathsf{A}_o^{j,\iota} := \mathsf{E}_o\mathrm{diag}(4, 2^{k_j^\iota})\mathsf{E}_o$.

The equation (5) shows the partition of unity property of the DAS-2 filter bank. The first equation in (6) indicates that convolution of an input data with the low-pass filter $a$ could be then downsampled by at least a factor of 2 along both dimensions. The downsampled output data is decreased by a factor of at least 4 in size. We therefore associate the low-pass filter $a$ with a sampling matrix $2\mathsf{I}_2$. The second equation in (6) indicates that convolution of data with such a filter $b_o^{j,\ell,\iota,\tau}$ could be then downsampled by at least a factor of 4 along one dimension and at least by a factor of $2^{k_j^\iota}$ along the other dimension. The downsampled output data is decreased by a factor of at least $2^{k_j^\iota+2}$ in size. We associate the high-pass filter $b_o^{j,\ell,\iota,\tau}$ with a sampling matrix $\mathsf{A}_o^{j,\iota} = \mathsf{E}_o\mathrm{diag}(4, 2^{k_j^\iota})\mathsf{E}_o$.

At level $j$, the parameter $k_j^\iota$ determines the number of directions in the inner or outer layer while the parameters $c_k, \epsilon_k, k = 0,1,2$ and $\varepsilon$ determine the shape of a directional high-pass filter. In practice, we can predefine such parameters and obtain a sequence of filter banks for multiscale data analysis. More precisely, given a sequence of nonnegative integers $k_j^\iota$, $\iota \in \{in,out\}, j = 0,\dots,J-1$ for some fixed integer $J \ge 1$ with respect to the finest scale, we can obtain a sequence of DAS-2 filter banks (see also Fig. 5)

$$\mathcal{DAS}_J\left(\{(a, 2\mathsf{I}_2); \mathcal{B}^{j,in}, \mathcal{B}^{j,out}\}_{j=0}^{J-1}\right) \tag{7}$$

with $\mathcal{B}^{j,\iota} := \{(b_o^{j,\ell,\iota,\tau}, \mathsf{A}_o^{j,\iota}) : \ell = -2^{k_j^\iota}, \dots, 2^{k_j^\iota}, \tau \in \{+,-\}, o \in \{X,Y\}\}$, which can be used in the problems of image processing. In the sequel, for simplicity, we omit

the sampling matrices and simply write $\{a; \mathcal{B}^{j,in}, \mathcal{B}^{j,out}\}$ for a DAS-2 filter bank. We also omit the ranges of $\iota \in \{in, out\}$ for inner and outer layers, $o \in \{X, Y\}$ for $X$ and $Y$ cones, and $\tau \in \{+, -\}$ for positive and negative axes.

## III. Digital Affine Shear Transforms

In this section, we detail the forward (decomposition) and backward (reconstruction) filter bank transforms based on our DAS-2 filter banks given in (7).

In a nutshell, given an image $v^J$, a DAS-2 filter bank decompose $v^J$ to an approximation data $v^{J-1}$ using the low-pass filter $a$ and a sequence of detail data $\{w_o^{J-1,\ell,\iota,\tau} : \ell, \iota, \tau\}$ using high-pass filters $\{b^{J-1,\ell,\iota,\tau} : \ell, \iota, \tau\}$. Further decomposition can be done on the approximation data $v^{J-1}$ using a DAS-2 filter bank at level $J-2$. On the other hand, given approximation and detail data at level $J-1$, a data at level $J$ can be reconstructed using corresponding DAS-2 filter bank. Such a process is illustrated in Fig. 5 for $J = 2$. More precise descriptions are given below.

For simplicity of presentation, we assume that our images live on dyadic grids $\Lambda(\mathsf{K}) := ([0, \dots, 2^{\mathsf{K}_1} - 1] \times [0, \dots, 2^{\mathsf{K}_2} - 1]) \cap \mathbb{N}_0^2$ with $\mathsf{K} := (\mathsf{K}_1, \mathsf{K}_2) \in \mathbb{N}^2$, where $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$. We remark that with simple boundary extension techniques, our numerical implementations work for images of any size.

Given an input image $v^J : \Lambda(\mathsf{K}) \to \mathbb{C}$ and a sequence $\mathcal{DAS}_J(\{a, \mathcal{B}^{j,in}, \mathcal{B}^{j,out}\}_{j=0}^{J-1})$ of DAS-2 filter banks as in (7), the (multilevel) *forward digital affine shear transform* decomposes $v^J$ to a sequence of filtered coefficients

$$\{v^0\} \cup \{w_o^{j,\ell,\iota,\tau} : |\ell| \le 2^{k_j^\iota}, \iota, \tau, o\}_{j=0}^{J-1}, \tag{8}$$

with $v^j := \mathcal{T}_{a,2\mathsf{I}_2}(v^{j+1})$, $w_o^{j,\ell,\iota,\tau} := \mathcal{T}_{b_o^{j,\ell,\iota,\tau},\mathsf{A}_o^{j,\iota}}(v^{j+1})$, whileas the (multilevel) *backward digital affine shear transform* reconstructs a sequence of filtered coefficients in (8) back to a data sequence as follows:

$$\mathring{v}^{j+1} = \mathcal{S}_{a,2\mathsf{I}_2}(\mathring{v}^j) + \sum_{\iota,\tau,o} \sum_{\ell=-2^{k_j^\iota}}^{2^{k_j^\iota}} \mathcal{S}_{b_o^{j,\ell,\iota,\tau},\mathsf{A}_o^{j,\iota}}(w_o^{j,\ell,\iota,\tau}), \tag{9}$$

for $j = 0, \dots, J-1$ with $\mathring{v}^0 := v^0$, where

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

are the transition operator and subdivision operator associated with a filter $u$ and a sampling matrix $\mathsf{A}$, respectively, which are detailed in [32, Section 3.2].

In short, let $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote the (2D) discrete Fourier transform and its inverse. Then, the circular convolution is given by $v \circledast u := \mathcal{F}^{-1}(\mathcal{F}(v) \odot \mathcal{F}(u))$, where $\odot$ is the Hadamard (entrywise) product, and hence it can be implemented fast using `fft2` and `ifft2` (e.g., in MATLAB). For a sampling matrix $\mathsf{A} := \mathrm{diag}(2^{m_1}, 2^{m_2})$ for $\mathsf{m} := (m_1, m_2) \in \mathbb{N}_0^2$, the *downsampling operation* $v \downarrow \mathsf{A}$ and the *upsampling operation* $v \uparrow \mathsf{A}$ are defined by $[v \downarrow \mathsf{A}](\mathsf{k}) := v(\mathsf{Ak})$ for $\mathsf{k} \in \Lambda(\mathsf{K} - \mathsf{m})$ and

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

In view of the PR property of each DAS-2 filter bank given in (5) and (6), we have the following result regarding the PR property in terms of the subdivision and transition operators.

*Corollary 2:* Let $\{a; \mathcal{B}^{j,in}, \mathcal{B}^{j,out}\}$ be a DAS-2 filter bank satisfying (5) and (6). Then,

$$\mathcal{S}_{a,2\mathsf{I}_2}[\mathcal{T}_{a,2\mathsf{I}_2}(v)] + \sum_{\iota,\tau,o} \sum_{\ell=-2^{k_j^\iota}}^{2^{k_j^\iota}} \mathcal{S}_{b_o^{j,\ell,\iota,\tau},\mathsf{A}_o^{j,\iota}}[\mathcal{T}_{b_o^{j,\ell,\iota,\tau},\mathsf{A}_o^{j,\iota}}(v)] = v$$

for any data sequence $v : \Lambda(\mathsf{K}) \to \mathbb{C}$.

The pseudo code for the forward and backward digital affine shear transforms are described in Algorithms 1 and 2. The implementation has been made available as a MATLAB package in http://personal.cityu.edu.hk/~xzhuang7/softs.

**Algorithm 1.** Forward Digital Affine Shear Transform

**Input:** Image $v^J$ and DAS-2 filter banks $\mathcal{DAS}_J\left(\{a; \mathcal{B}^{j,in}, \mathcal{B}^{j,out}\}_{j=0}^{J-1}\right)$.
**Output:** filtered coefficients $\{v^0\} \cup \{w_o^{j,\ell,\iota,\tau} : |\ell| \le 2^{k_j^\iota}, \iota, \tau, o\}_{j=0}^{J-1}$
**Main steps:**
**for** $j = J-1$ **to** $0$ **do**
  $v^j \leftarrow \mathcal{T}_{a,2\mathsf{I}_2}(v^{j+1})$.
  **for** each $b_o^{j,\ell,\iota,\tau}$ in $\mathcal{B}^{j,in} \cup \mathcal{B}^{j,out}$ **do**
    $w_o^{j,\ell,\iota,\tau} \leftarrow \mathcal{T}_{b_o^{j,\ell,\iota,\tau},\mathsf{A}_o^{j,\iota}}(v^{j+1})$.
  **end for**
**end for**

**Algorithm 2.** Backward Digital Affine Shear Transform

**Input:** Filtered coefficients $\{v^0\} \cup \{w_o^{j,\ell,\iota,\tau} : |\ell| \le 2^{k_j^\iota}, \iota, \tau, o\}_{j=0}^{J-1}$ and DAS-2 filter banks $\mathcal{DAS}_J\left(\{a; \mathcal{B}^{j,in}, \mathcal{B}^{j,out}\}_{j=0}^{J-1}\right)$.
**Output:** Image $v^J$.
**Main steps:**
**for** $j = 0$ **to** $J-1$ **do**
  $v^{j+1} \leftarrow \mathcal{S}_{a,2\mathsf{I}_2}(v^j)$.
  **for** each $b_o^{j,\ell,\iota,\tau}$ in $\mathcal{B}^{j,in} \cup \mathcal{B}^{j,out}$ **do**
    $v^{j+1} \leftarrow v^{j+1} + \mathcal{S}_{b_o^{j,\ell,\iota,\tau},\mathsf{A}_o^{j,\iota}}(w_o^{j,\ell,\iota,\tau})$.
  **end for**
**end for**

## IV. Redundancy Rate and Computational Complexity

In this section, we estimate the redundancy rate and computational complexity of the digital affine shear transforms based on our DAS-2 filter banks.

The redundancy rate of a filter bank transform is defined to be the ratio of the size of the total output coefficients to the size of the input data. For real-value data $v$, by the symmetry of filters, we have $w_o^{j,\ell,\iota,+} = \overline{w_o^{j,\ell,\iota,-}}$ and hence we only need to compute $\{v^0\} \cup \{w_o^{j,\ell,\iota,+} : |\ell| \le 2^{k_j^\iota}, \iota, o\}_{j=0}^{J-1}$. Let $N := 2^{\mathsf{K}_1+\mathsf{K}_2}$ be the size of the input data $v^J : \Lambda(\mathsf{K}_1, \mathsf{K}_2) \to \mathbb{R}$.

At level $j$, after the downsampling processing with sampling matrix $\mathsf{A}_X^{j,\iota} = \mathrm{diag}(4, 2^{k_j^\iota})$, the coefficient matrix $w_X^{j,\ell,\iota,+} = \mathcal{T}_{b_X^{j,\ell,\iota,+},A_X^{j,\iota}} v^{j+1}$ is then living on the lattice $\Lambda(\widetilde{\mathsf{K}}_1, \widetilde{\mathsf{K}}_2)$ with

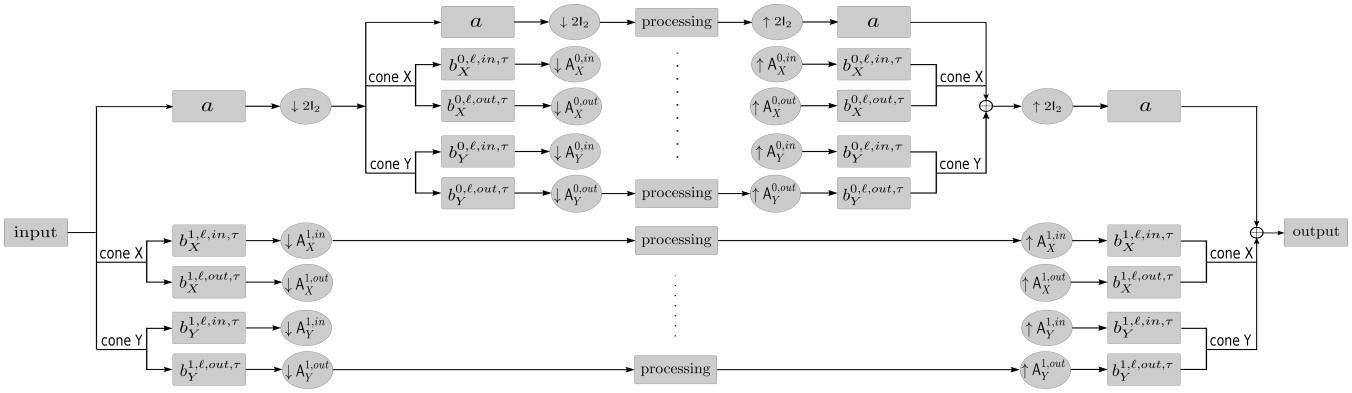



Fig. 5: Two-level DAS-2 filter bank transforms. Each box $\boxed{u}$ is a convolution operation with the filter inside, a circle with $\downarrow$ is a downsampling operation, and a circle with $\uparrow$ is a upsampling operation. A box $\boxed{u}$ together with a circle $\downarrow$ on the 'input' side (decomposition) corresponds to a transition operator $\mathcal{T}_{u,\mathsf{A}}$, while a box $\boxed{u}$ with a circle $\uparrow$ on the 'output' side (reconstruction) corresponds to a subdivision operator $\mathcal{S}_{u,\mathsf{A}}$. The index $\ell$ and $\tau$ range over their domains.

$\widetilde{\mathsf{K}}_1 = \mathsf{K}_1 - (J-1-j) - 2$ and $\widetilde{\mathsf{K}}_2 = \mathsf{K}_2 - (J-1-j) - k_j^\iota$, which is of size $\widetilde{N} = \frac{N}{2^{(J-1-j)+2}} \cdot \frac{1}{2^{(J-1-j)+k_j^\iota}} = \frac{N}{2^{2(J-j)+k_j^\iota}}$. Since the implementation of the transition operator is based on the `fft`, one can show that (see [32, Section 3.2]) the computational complexity of $w_X^{j,\ell,\iota,+}$ is $\widetilde{N}\log\widetilde{N}$. The size and computational complexity of $w_Y^{j,\ell,\iota,+}$ are the same as those of $w_X^{j,\ell,\iota,+}$.

Fixed the level $j$ and the layer $\iota$, the total number of high-pass filtered coefficient matrices $w_o^{j,\ell,\iota,+}$ for $o \in \{X, Y\}$ and $\ell \le |2^{k_j^\iota}|$ is $2(2^{k_j^\iota+1}+1)$. Consequently, the size of the total output high-pass filtered coefficients at level $j$ is $\sum_\iota \frac{N}{2^{2(J-j)+k_j^\iota}} \times 2(2^{k_j^\iota+1}+1) \times 2$, where the factor '$\times 2$' is due to the complex-value of the outputs $w_o^{j,\ell,\iota,+}$. The low-pass coefficient $v^0$ is of size $N/2^{2J}$. Therefore, after the $J$-level decomposition, the total size of output coefficients is $N\left(\sum_{j=0}^{J-1}\sum_\iota \frac{2^2\cdot(2^{k_j^\iota+1}+1)}{2^{2(J-j)+k_j^\iota}} \cdot + \frac{1}{2^{2J}}\right)$ and thus the redundancy rate $r$ is given by

$$r = \sum_{j=0}^{J-1}\sum_\iota \frac{2+2^{-k_j^\iota}}{2^{2(J-1-j)}} + \frac{1}{4^J} \le (4+2^{-k_{min}+1})\cdot\frac{4}{3} \le 8,$$

where $k_{min} := \min\{k_j^{in}, k_j^{out} : j = 0, \dots, J-1\}$.

Consequently, the redundancy rate of the digital affine shear transforms is bounded below by $16/3$ and above by $8$. More importantly, the redundancy rate does NOT increase with respect to the number of directional filters, which is a very desirable property in practice. In fact, the more directional filters we have, the lower redundancy rate of our digital affine shear transforms. For $k_{min} = 0, 1, 2, 3, 4$, the redundancy rate is bounded by $8, 6\frac{2}{3}, 6, 5\frac{2}{3}, 5\frac{1}{2}$, respectively.

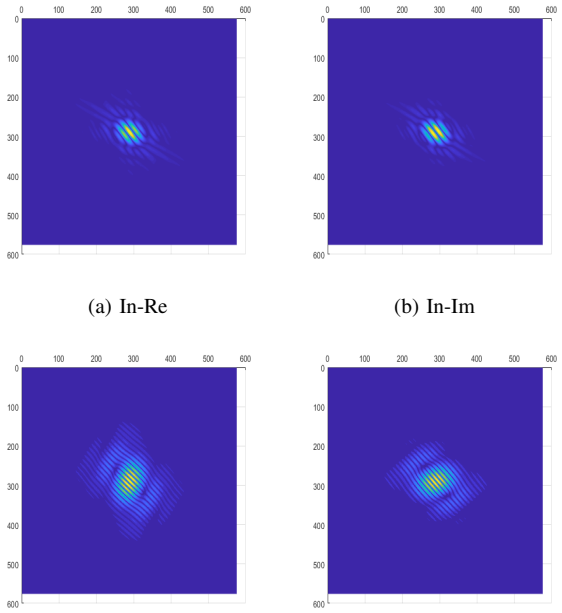
Since the transition operator $\mathcal{T}_{u,\mathsf{A}}v$ and subdivision operator $\mathcal{S}_{u,\mathsf{A}}v$ can be implemented based on the `fft2` and `ifft2`, the computational complexity of these operators are proportional to the size of the input data. Similar to [32, Section 3.2], one can show that the computational complexity of both the forward and backward transforms base on our DAS-2 filter banks is proportional to $rN\log N$ with $N$ the size of input data and $r$ the redundancy rate given above.

## V. Numerical Experiments

In this section, we apply our digital affine shear transforms based on our DAS-2 filter banks for the tasks of image processing including denoising, inpainting, and texture classification. We compare the performance of our system to several other state-of-the-art directional multiscale representation systems.

The parameters of $a, a_1$ and $a_2$ in (1) and (2) are given by $c_0 = \pi, c_1 = 1.844, c_2 = 0.844, \epsilon_0 = 0.086, \epsilon_1 = 0.157, \epsilon_2 = 0.381$, and $\varepsilon = 0.469$ for $\boldsymbol{\gamma}_\varepsilon$. We choose $J = 5$ for $\mathcal{DAS}_5(\{a; \mathcal{B}^{j,in}, \mathcal{B}^{j,out}\}_{j=0}^4)$ as in (7); that is, we decompose to 5 levels. The shear parameter $(k_4^\iota, k_3^\iota, k_2^\iota, k_1^\iota, k_0^\iota)$ is set to be $(2,2,1,1,0)$ for the outer layer $\iota = out$ and $(1,1,0,0,0)$ for the inner layer $\iota = in$. That is, for the highest level $j = 4$, we use $2(2^{k_4^{out}+1}+1) = 18$ shear directions for the outer layer (9 on horizontal cone and 9 on vertical cone) and $2(2^{k_4^{in}+1}+1) = 10$ directions for the inner layer. The redundancy rate of our system $\mathcal{DAS}_5(\{a; \mathcal{B}^{j,in}, \mathcal{B}^{j,out}\}_{j=0}^4)$ is $\frac{6545}{1024} \approx 6.39$. Fig. 6 shows some examples of our high-pass filters in the image domain. Two high-pass filters $b_X^{4,4,in,+}$ and $b_X^{4,4,out,+}$ at level $j = 4$ concentrating at $-45°$ ($\ell = 4$) are illustrated. Note that they are complex-value and hence the real and imaginary parts are given separately. One can clearly see that the inner layer filters are more edge-like while the out-layer filters are more texture-like.

For our DAS-2 as well as DAS-1, the local soft-thresholding (LS) technique is applied (see [16, Section 6]). More precisely, for each high-pass coefficient matrix $w$ in (8), we normalize $w$ to $w_b := \frac{w}{\|b\|_2}$ with $b$ being its induced filter satisfying $w = \mathcal{T}_{b,\mathsf{B}}v^J$ for the input data $v^J$ and some downsampling matrix $\mathsf{B}$. Let $\sigma$ be the variance of a noise obeying normal distribution $N(0,\sigma^2)$ and $\Lambda$ be the lattice where $w$ lives. 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



(a) In-Re (b) In-Im

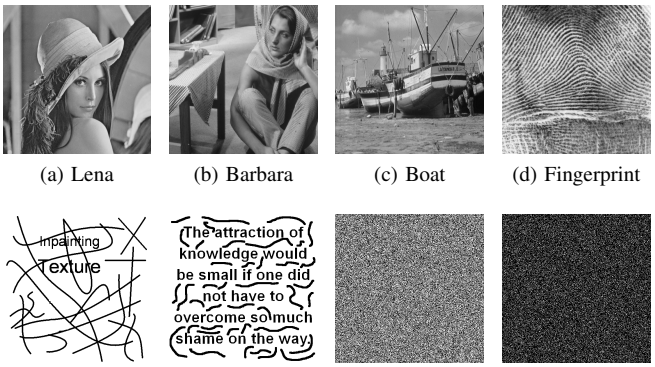
(c) Out-Re (d) Out-Im

Fig. 6: The filters $b_X^{4,4,in,+}$ and $b_X^{4,4,out,+}$. (a) Inner layer filter $b_X^{4,4,in,+}$: Real part. (b) $b_X^{4,4,in,+}$: Imaginary part. (c) Outer layer filter $b_X^{4,4,out,+}$: Real part. (d) $b_X^{4,4,out,+}$: Imaginary part.

$\Lambda_{\mathsf{k}}$ is the lattice $\mathsf{k} + [-L, L]^2$ that centered at position $\mathsf{k}$ for some integer $L \geq 0$. 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 local-soft thresholding $\widetilde{w}_b := \eta_{t_w}^{ls}(w_b)$ applying to $w_b$ is then given by $\widetilde{w}_b(\mathsf{k}) := \eta_{t_w(\mathsf{k})}^{soft}(w_b(\mathsf{k}))$, $\mathsf{k} \in \Lambda$, where $\eta_t^{soft}(x) = \max\{|x| - t, 0\}\frac{x}{|x|}$ is the soft-thresholding operator. The threshold coefficient $\widetilde{w}_b$ is then renomralized back to $\widetilde{w} := \|b\|_2 \cdot \eta_{t_w}^{ls}(w_b)$.

For image denoising, we tested 4 images: `Lena`, `Barbara`, `Boat`, and `Fingerprint` (see Fig. 7(a)–(d)). Gaussian noise $N(0, \sigma)$ with $\sigma = 5, 10, 30, 50, 80$ are added to each image. We apply the forward transform to the image and obtain a set of output filtered coefficients, which are then thresholded using the local soft-thresholding technique with the convolution window filter of size $9 \times 9$. Backward transform is then applied to the thresholded filtered coefficients.

We compare our results to several state-of-the-art frame-based denoised methods including The DAS-1 in [32], TP-$\mathbb{C}$TF$_6$ and TP-$\mathbb{C}$TF$_6^{\downarrow}$ in [18], DT-$\mathbb{C}$WT in [29], DNST in [26], and NSCT in [6]. DAS-1 decomposes the image to $J = 5$ levels with 18 directional filters in the finest level $j = 4$, 10 directional filters in levels $j = 3, 2, 1$, and 6 directional filters in level $j = 0$. Its redundancy rate is about 6.16. The DAS-1 uses local soft-thresholding technique as well with convolution window filter of size $11 \times 11$. The TP-$\mathbb{C}$TF$_6$ and TP-$\mathbb{C}$TF$_6^{\downarrow}$ decompose the image to $J = 5$ levels with each level having 14 directional filters. They use the bivariate shrinkage thresholding technique and their redundancy rates are 10.67 and 2.67, respectively. The DT-$\mathbb{C}$WT decomposes




(a) Lena (b) Barbara (c) Boat (d) Fingerprint

(e) Text1 (f) Text2 (g) M50 (h) M80

Fig. 7: (a)–(d) Images. (e)–(h) Masks.

the image to $J = 6$ levels with each level having 6 directional filters. It uses the bivariate shrinkage thresholding technique and its redundancy rate is 4. The DNST decomposes the image to $J = 4$ levels. 16 shear directions are used for finest levels $j = 3$ and $j = 2$ while 8 shear directions are used for the other two levels $j = 1, 0$. The redundancy of DNST is 49. The NSCT uses $4, 8, 8, 16, 16$ directions from the coarser to finer levels. Its redundancy rate is 53. DNST and NSCT both use hard thresholding.

The comparison results are reported in Table I in terms of PSNR (unit dB), where $\mathrm{PSNR}(v, \widetilde{v}) = 10 \log_{10} \frac{255^2}{\mathrm{MSE}(v,\widetilde{v})}$ with $\mathrm{MSE}(v, \widetilde{v}) = \frac{1}{\#\Lambda} \sum_{\mathsf{k}\in\Lambda} |v(\mathsf{k}) - \widetilde{v}(\mathsf{k})|^2$ for input image $v$ and reconstruction image $\widetilde{v}$ on a lattice $\Lambda$.

| $\sigma$ | DAS-2 (6.39) | DAS-1 (6.16) | TP-$\mathbb{C}$TF$_6^{\downarrow}$ (2.67) | TP-$\mathbb{C}$TF$_6$ (10.67) | DT-$\mathbb{C}$WT (4) | DNST (49) | NSCT (53) |
|---|---|---|---|---|---|---|---|
| $512 \times 512$ Lena | | | | | | | |
| 5 | 38.36 | 38.14(0.22) | 38.16(0.20) | **38.37**(-0.01) | 38.25(0.11) | 38.01(0.35) | 37.71(0.65) |
| 10 | 35.39 | 35.12(0.27) | 35.22(0.17) | **35.48**(-0.09) | 35.19(0.2) | 35.35(0.04) | 34.92(0.47) |
| 30 | 30.62 | 30.61(0.01) | 30.38(0.24) | **30.80**(-0.18) | 30.50(0.12) | 30.68(-0.06) | 30.32(0.3) |
| 50 | 28.30 | 28.49(-0.19) | 28.11(0.19) | **28.54**(-0.24) | 28.22(0.08) | 28.21(0.09) | 28.02(0.28) |
| 80 | 26.20 | **26.54**(-0.34) | 26.11(0.39) | 26.47(-0.27) | 26.15(0.05) | 25.78(0.42) | 25.80(0.4) |
| $512 \times 512$ Barbara | | | | | | | |
| 5 | 37.77 | 37.32(0.45) | 37.63(0.14) | **37.84**(-0.07) | 37.37(0.4) | 37.17(0.6) | 36.96(0.81) |
| 10 | 34.14 | 33.64(0.5) | 33.97(0.17) | **34.18**(-0.04) | 33.54(0.6) | 33.62(0.52) | 33.35(0.79) |
| 30 | **28.73** | 28.33(0.4) | 28.33(0.4) | 28.38(0.35) | 27.89(0.84) | 27.97(0.76) | 27.28(1.45) |
| 50 | **26.29** | 26.01(0.28) | 25.73(0.56) | 25.71(0.58) | 25.36(0.93) | 25.31(0.98) | 24.57(1.72) |
| 80 | **24.10** | 23.99(0.11) | 23.51(0.59) | 23.53(0.57) | 23.27(0.83) | 22.96(1.14) | 22.65(1.45) |
| $512 \times 512$ Boat | | | | | | | |
| 5 | **36.93** | 36.63(0.30) | 36.74(0.19) | 36.92(0.01) | 36.73(0.20) | 36.04(0.89) | 35.79(1.14) |
| 10 | 33.27 | 33.01(0.26) | 33.10(0.17) | **33.41**(-0.14) | 33.19(0.08) | 33.15(0.12) | 32.65(0.62) |
| 30 | 28.25 | 28.31(-0.06) | 27.99(0.26) | **28.44**(-0.19) | 28.23(0.03) | 28.44(-0.19) | 27.95(0.30) |
| 50 | 26.08 | 26.24(-0.16) | 25.79(0.29) | **26.25**(-0.17) | 26.06(0.02) | 26.23(-0.15) | 25.94(0.14) |
| 80 | 24.23 | **24.46**(-0.23) | 24.05(0.19) | 24.41(-0.17) | 24.22(0.01) | 24.17(0.06) | 24.11(0.12) |
| $512 \times 512$ Fingerprint | | | | | | | |
| 5 | **36.27** | 35.20(1.07) | 36.29(-0.02) | **36.27**(0.00) | 35.82(0.44) | 35.28(0.99) | 34.93(1.34) |
| 10 | 32.08 | 30.97(1.11) | **32.23**(-0.15) | 32.10(-0.02) | 31.74(0.34) | 31.76(0.31) | 31.33(0.75) |
| 30 | 26.26 | 26.24(0.02) | **26.37**(-0.11) | 26.06(0.21) | **26.37**(-0.11) | 26.20(0.07) | 26.13(0.13) |
| 50 | 24.00 | **24.11**(-0.11) | 24.01(-0.01) | 23.67(0.33) | 23.95(0.05) | 23.78(0.22) | 23.89(0.11) |
| 80 | 22.10 | **22.18**(-0.08) | 21.99(0.11) | 21.66(0.44) | 21.91(0.19) | 21.63(0.47) | 21.79(0.31) |

TABLE I: PSNR of denoised images using various transforms. Numbers in brackets are the PSNR differences between the DAS-2 column and the current column. Positive numbers indicate better performance of DAS-2 than others in dB.

From Table I, the performance of our DAS-2 is in general better than the methods of DT-$\mathbb{C}$WT, DNST, and NSCT for all images, given that the redundancy rates of DNST and NSCT are extremely high (49 and 53 due to their undecimation property). Comparing to DAS-1, the performance of DAS-2 is better when the noise variance $\sigma$ is relatively low ($< 50$).

Especially when the image is of texture-rich such as Barbara and Fingerprint, the DAS-2 could bring up to 0.5 dB improvement for Barbara and up to 1.1 dB improvement for Fingerprint. Comparing to TP-$\mathbb{C}$TF$_6^{\downarrow}$, our DAS-2 performs better for the images of Lena, Barbara, and Boat. The performance of TP-$\mathbb{C}$TF$_6^{\downarrow}$ is quit well for the image Fingerprint given that it only has redundancy rate as low as 2.67. Finally, comparing to TP-$\mathbb{C}$TF$_6$ which has redundancy rate 10.67, we see that DAS-2 and TP-$\mathbb{C}$TF$_6$ perform quite similar for all images when $\sigma$ is low ($< 30$). When $\sigma$ is high, our DAS-2 performs better than TP-$\mathbb{C}$TF$_6$ for images Barbara and Fingerprint, while for Lena and Boat, TP-$\mathbb{C}$TF$_6$ performs better than our DAS-2.

Note that most of information in the images of Lena and Boat are edge-like. The image Barbara contains both oscillating patterns (texture) and edge-like features while the image Fingerprint is mainly oscillating patterns. The PSNRs in Table I confirm our expectation of DAS-2 filter banks. It preserves the performance over general images comparing to other state-of-the-art methods while significant improves the performance for images containing both edge-like structures and texture-like structures. Fig. 2 illustrates the significant improvement in visualization of our DAS-2 comparing to other methods for the image Barbara.

Next, we test the performance of our DAS-2 filter banks in image inpainting. We use 4 types of masks: Text1 (small gaps), Text2 (larger gaps), M50 (50% of random pixel missing), and M80 (80% of random pixel missing); see Fig. 7 (e)–(h) for the 4 masks. Each images in Fig. 7 (a)–(d) is masked with each mask and added Gaussian noise for $\sigma \in \{0(\text{no noise}), 5, 10, 30, 50\}$. We then employ the same inpainting framework developed in [30], which uses an iterative thresholding algorithm with gradually decreasing threshold values. The results are reported in Table II. Since DAS-1 and TP-$\mathbb{C}$TF$_6$ outperform many other approaches (see [32, Section 4]), here we only report the comparisons among DAS-2, DAS-1, TP-$\mathbb{C}$TF$_6^{\downarrow}$, and TP-$\mathbb{C}$TF$_6$.

For masks Text1 and Text2, DAS-2 outperforms DAS-1 and TP-$\mathbb{C}$TF$_6^{\downarrow}$ for all cases except for a few cases of slightly underperformance, e.g., the cases of $\sigma = 0$ for Fingerprint for both masks and $\sigma = 0$ for Barbara with mask Text1. Comparing to TP-$\mathbb{C}$TF$_6$, DAS-2 outperforms TP-$\mathbb{C}$TF$_6$ for images Barbara and Fingerprint while slightly underperforms TP-$\mathbb{C}$TF$_6$ for images Lena and Boat. See Figs. 8 and 9 for their visual comparisons.

For masks M50 and M80, DAS-2 outperforms DAS-1 for all images when noise variance $\sigma < 50$. DAS-2 outperforms TP-$\mathbb{C}$TF$_6^{\downarrow}$ for images Lena, Barbara, and Boat while it underperforms TP-$\mathbb{C}$TF$_6^{\downarrow}$ for the image Fingerprint for $\sigma < 50$. Comparing to TP-$\mathbb{C}$TF$_6$, DAS-2 performs better for images Barbara and Fingerprint while TP-$\mathbb{C}$TF$_6$ performs better for the other two images.

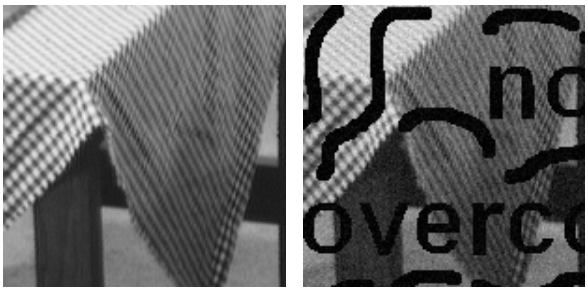
Finally, we introduce an experiment on texture classification, to further justify that our DAS-2 is more desirable for texture representation than other transforms. We use a feature descriptor HSC introduced in [28] for our texture

| | Text1 | | | | Text2 | | | |
|---|---|---|---|---|---|---|---|---|
| $\sigma$ | DAS-2 | DAS-1 | TP-$\mathbb{C}$TF$_6^{\downarrow}$ | TP-$\mathbb{C}$TF$_6$ | DAS-2 | DAS-1 | TP-$\mathbb{C}$TF$_6^{\downarrow}$ | TP-$\mathbb{C}$TF$_6$ |
| 512 × 512 Lena | | | | | | | | |
| 0 | 37.91 | 37.98(-0.07) | 37.71 (0.2) | **38.02**(-0.11) | 34.04 | 33.93(0.11) | 33.92(0.12) | **34.31**(-0.27) |
| 5 | 35.16 | 35.15(0.01) | 34.87(0.29) | **35.19**(-0.03) | 32.81 | 32.63(0.18) | 32.52(0.29) | **32.97**(-0.16) |
| 10 | 33.41 | 33.34(0.07) | 33.08(0.33) | **33.42**(-0.01) | 31.69 | 31.58(0.11) | 31.32(0.37) | **31.80**(-0.11) |
| 30 | 29.66 | 29.68(-0.02) | 29.32(0.34) | **29.81**(-0.15) | 28.72 | 28.84(-0.12) | 28.34(0.38) | **28.89**(-0.17) |
| 50 | 27.64 | 27.82(-0.18) | 27.33(0.31) | **27.85**(-0.21) | 27.02 | 27.20(-0.18) | 26.60(0.42) | **27.22**(-0.2) |
| 512 × 512 Barbara | | | | | | | | |
| 0 | 36.60 | 36.16(0.44) | **36.68**(-0.08) | 36.59(0.01) | **33.69** | 33.41(0.28) | 32.99 (0.7) | 32.68(1.01) |
| 5 | **34.14** | 33.75(0.39) | 33.97(0.17) | 34.05(0.09) | **32.40** | 31.94(0.46) | 31.54(0.86) | 31.32(1.08) |
| 10 | **32.11** | 31.67(0.44) | 31.76(0.35) | 31.81 (0.3) | **30.81** | 30.39(0.42) | 29.99(0.82) | 29.85(0.96) |
| 30 | **27.80** | 27.38(0.42) | 27.21(0.59) | 27.18(0.62) | **26.95** | 26.65 (0.3) | 26.24(0.71) | 26.24(0.71) |
| 50 | **25.57** | 25.29(0.28) | 24.91(0.66) | 24.91(0.66) | **24.95** | 24.73(0.22) | 24.18(0.77) | 24.30(0.65) |
| 512 × 512 Boat | | | | | | | | |
| 0 | 34.8 | 34.69(0.11) | 34.57(0.23) | **34.96**(-0.16) | 30.73 | 30.74(-0.01) | 30.39(0.34) | **30.80**(-0.07) |
| 5 | 32.76 | 32.63(0.13) | 32.46(0.30) | **32.81**(-0.05) | **29.84** | 29.82(0.02) | 29.42(0.42) | 29.83(0.01) |
| 10 | 30.95 | 30.82(0.13) | 30.65(0.30) | **31.04**(-0.09) | 28.82 | **28.85**(-0.03) | 28.40(0.42) | 28.80(0.02) |
| 30 | 27.31 | 27.38(-0.07) | 26.95(0.36) | **27.41**(-0.10) | 26.18 | **26.34**(-0.16) | 25.79(0.39) | 26.24(-0.06) |
| 50 | 25.47 | **25.60**(-0.13) | 25.11(0.36) | 25.57(-0.10) | 24.72 | **24.84**(-0.12) | 24.32(0.40) | 24.80(-0.08) |
| 512 × 512 Fingerprint | | | | | | | | |
| 0 | 31.78 | 31.22(0.56) | **31.87**(-0.09) | 31.35(0.43) | 28.30 | 27.70(0.60) | **28.36**(-0.06) | 27.78(0.52) |
| 5 | **30.40** | 29.85(0.55) | 30.39(0.01) | 30.03(0.37) | **27.69** | 27.13(0.56) | 27.65(0.04) | 27.17(0.52) |
| 10 | **28.79** | 28.19(0.60) | 28.77(0.02) | 28.46(0.33) | **26.76** | 26.32(0.44) | 26.67(0.09) | 26.24(0.52) |
| 30 | **25.02** | 24.99(0.03) | 24.98(0.04) | 24.70(0.32) | **24.12** | **24.12**(0.00) | 23.99(0.13) | 23.59(0.53) |
| 50 | 23.22 | **23.25**(-0.03) | 23.05(0.17) | 22.76(0.46) | 22.60 | **22.64**(-0.04) | 22.39(0.21) | 22.00(0.60) |

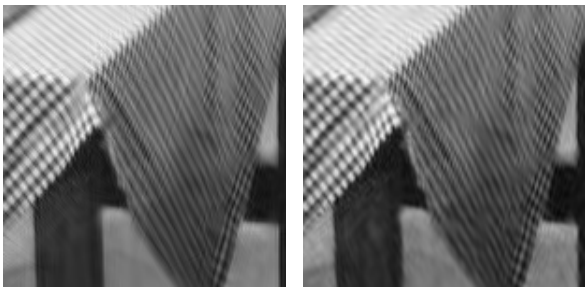
| | 50% missing | | | | 80% missing | | | |
|---|---|---|---|---|---|---|---|---|
| $\sigma$ | DAS-2 | DAS-1 | TP-$\mathbb{C}$TF$_6^{\downarrow}$ | TP-$\mathbb{C}$TF$_6$ | DAS-2 | DAS-1 | TP-$\mathbb{C}$TF$_6^{\downarrow}$ | TP-$\mathbb{C}$TF$_6$ |
| 512 × 512 Lena | | | | | | | | |
| 0 | 37.69 | 35.72(1.97) | 37.68(0.01) | **38.00**(-0.31) | 31.86 | 30.74(1.12) | 31.99(-0.13) | **32.33**(-0.47) |
| 5 | 35.09 | 33.97(1.12) | 34.99 (0.1) | **35.40**(-0.31) | 31.03 | 30.20(0.83) | 30.76(0.27) | **31.44**(-0.41) |
| 10 | 33.08 | 32.57(0.51) | 32.86(0.22) | **33.40**(-0.32) | 29.92 | 29.54(0.38) | 29.34(0.58) | **30.25**(-0.33) |
| 30 | 28.89 | 28.90(-0.01) | 28.52(0.37) | **29.18**(-0.29) | 26.61 | 26.76(-0.15) | 25.94(0.67) | **26.95**(-0.34) |
| 50 | 26.75 | 26.92(-0.17) | 26.39(0.36) | **27.06**(-0.31) | 24.78 | 25.01(-0.23) | 24.11(0.67) | **25.15**(-0.37) |
| 512 × 512 Barbara | | | | | | | | |
| 0 | **35.86** | 33.77(2.09) | 35.75(0.11) | 35.73(0.13) | **29.32** | 27.84(1.48) | 28.55(0.77) | 28.16(1.16) |
| 5 | **33.44** | 32.14 (1.3) | 33.23(0.22) | 33.42(0.02) | **28.57** | 27.30(1.27) | 27.73(0.84) | 27.73(0.85) |
| 10 | **31.29** | 30.47(0.82) | 30.94(0.35) | 31.11(0.18) | **27.43** | 26.72(0.71) | 26.56(0.87) | 26.70(0.73) |
| 30 | **26.73** | 26.32(0.41) | 25.95(0.78) | 25.95(0.78) | **24.23** | 24.16(0.07) | 23.18(1.05) | 23.34(0.89) |
| 50 | **24.47** | 24.25(0.22) | 23.59(0.88) | 23.60(0.87) | **22.30** | 22.26(0.04) | 21.42(0.88) | 21.90 (0.4) |
| 512 × 512 Boat | | | | | | | | |
| 0 | 34.27 | 32.65(1.62) | 34.00(0.27) | **34.42**(-0.15) | 28.16 | 27.24(0.92) | 28.03(0.13) | **28.58**(-0.42) |
| 5 | 32.17 | 31.25(0.92) | 32.01(0.16) | **32.50**(-0.33) | 27.48 | 26.83(0.65) | 27.28(0.20) | **27.98**(-0.50) |
| 10 | 30.33 | 29.98(0.35) | 30.11(0.22) | **30.65**(-0.32) | 26.61 | 26.45(0.16) | 26.23(0.38) | **27.08**(-0.47) |
| 30 | 26.39 | 26.48(-0.09) | 26.07(0.32) | **26.66**(-0.27) | 24.22 | 24.34(-0.12) | 23.75(0.47) | **24.46**(-0.24) |
| 50 | 24.56 | 24.63(-0.07) | 24.23(0.33) | **24.75**(-0.19) | 22.76 | 22.89(-0.13) | 22.41(0.35) | **22.96**(-0.20) |
| 512 × 512 Fingerprint | | | | | | | | |
| 0 | 33.64 | 29.35(4.29) | **34.19**(-0.55) | 34.12(-0.48) | 26.25 | 23.87(2.38) | **26.77**(-0.52) | 26.00(0.25) |
| 5 | 31.12 | 28.40(2.72) | **31.54**(-0.42) | 31.50(-0.38) | 25.58 | 23.64(1.94) | **25.87**(-0.29) | 25.33(0.25) |
| 10 | 28.84 | 27.57(1.27) | **29.09**(-0.25) | 28.88(-0.04) | 24.54 | 24.10(0.44) | **24.60**(-0.06) | 24.12(0.42) |
| 30 | 24.33 | 24.35(-0.02) | **24.43**(-0.10) | 24.07(0.26) | **22.12** | 22.10(0.02) | 21.81(0.31) | 21.51(0.61) |
| 50 | **22.41** | 22.40(0.01) | 22.32(0.09) | 22.01(0.40) | **20.60** | 20.57(0.03) | 20.26(0.34) | 19.96(0.64) |

TABLE II: PSNR of inpainted Lena, Barbara, Boat, and Fingerprint using different transforms for masks Text1, Text2, M50 and M80.

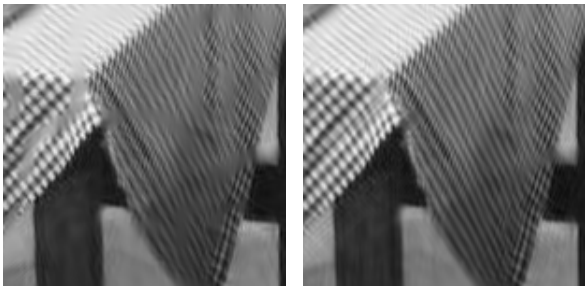
classification: Fixed a transform in DAS-2, DAS-1, TP-$\mathbb{C}$TF$_6^{\downarrow}$, TP-$\mathbb{C}$TF$_6$, DT-$\mathbb{C}$WT, DNST, and NSCT, application of such a transform to an image results in a sequence of filtered coefficient matrices $w$ with respect to various orientations. A histogram (vector) is computed from the absolute values of the entries in $w$. Such histograms from all the output filtered coefficient matrices are then concatenated as a feature vector for the given image with respect to such a transform, and normalized with respect to the $L_2$-norm. We use the UMD texture dataset [31] to extract feature representations for each transform and then use the approach of SVMs [5] to perform texture classification. The UMD texture database consists of 25 classes and 40 samples of $1280 \times 960$ pixels in each class (1000 images in total). In the classification process, we take 10 samples randomly from each class for training and the remaining 30 samples for testing. 30 repeated experiments are carried out to obtain the average classification rate. The results are reported in Table III and it can be seen that DAS-2 outperforms the other transforms consistently, of which only TP-$\mathbb{C}$TF$_6$ gives a comparable performance. But DAS-2 has a significantly lower redundancy rate.



(a) Tablecloth (b) Missing data



(c) DAS-1 (d) TP-$\mathbb{C}$TF$_6^{\downarrow}$



(e) TP-$\mathbb{C}$TF$_6$ (f) DAS-2

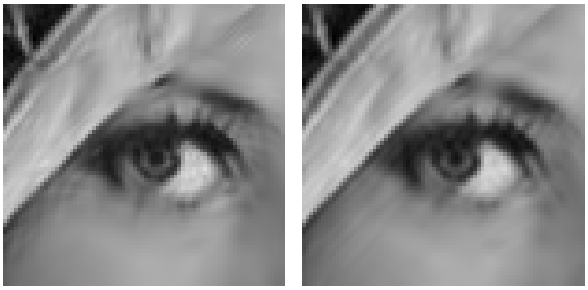
Fig. 8: Comparison results of inpainting. (a) Original (tablecloth cropped from `Barbara`). (b) Missing data (with mask `Text2`) plus corrupted by Gaussian noise with $\sigma = 10$. (c) DAS-1: look similar to DAS-2. (d) TP-$\mathbb{C}$TF$_6^{\downarrow}$: mask can be seen. (e) TP-$\mathbb{C}$TF$_6$: clearly see the mask. (f) DAS-2: noise removed and missing area is inpainted.

| Texture Classification | | | | | | |
|---|---|---|---|---|---|---|
| DAS-2 (6.39) | DAS-1 (6.16) | TP-$\mathbb{C}$TF$_6^{\downarrow}$ (2.67) | TP-$\mathbb{C}$TF$_6$ (10.67) | DT-$\mathbb{C}$WT (4) | DNST (49) | NSCT (53) |
| **85.8%** | 80.71% | 83.4% | 85.31% | 81.85% | 77.79% | 79.9% |

TABLE III: Average classification rate of various transforms on the UMD texture database.

(a) Eye (b) Missing Data

(c) DAS-1 (d) TP-$\mathbb{C}$TF$_6^{\downarrow}$



(e) TP-$\mathbb{C}$TF$_6$ (f) DAS-2

Fig. 9: Comparison results of inpainting. We compare three parts of the eye: the eyebrow, the bottom eyelash, and the iris. (a) Original (eye part cropped from `Lena`). (b) Missing data (with mask `Text2`) plus corrupted by Gaussian noise with $\sigma = 10$. (c) DAS-1: recover the eyebrow better than the other methods but not so good at the bottom eyelash and the iris. (d) TP-$\mathbb{C}$TF$_6^{\downarrow}$: not as good as other methods for all three parts. (e) TP-$\mathbb{C}$TF$_6$: better than DAS-1 and TP-$\mathbb{C}$TF$_6^{\downarrow}$ for the bottom eyelash and the iris. (f) DAS-2: recover more details at the iris and the bottom eyelash than other methods.

## VI. Affine Shear Tight Frames with 2-Layer Structure

In this section, we connect our DAS-2 filter banks to the affine shear tight frames with 2-layer structure for theoretical justifications of the sparse representation property.

Given a sequence $\{a; b_o^{j,\ell,\iota,\tau}\}, j \in \mathbb{N}_0$ of DAS-2 filter banks, it naturally connects to a sequence of directional framelet systems (with 2-layer structure). In fact, one can defined a function (or distribution) $\varphi$ through $\widehat{\varphi}(\xi) := \lim_{J\to\infty} \prod_{j=0}^{J} \widehat{a}(2^{-j-1}\xi), \xi \in \mathbb{R}^2$ which automatically satisfies the refinement relation $\widehat{\varphi}(2\xi) = \widehat{a}(\xi)\varphi(\xi), \xi \in \mathbb{R}^2$. From $\varphi$ and the high-pass filters $b_o^{j,\ell,\iota,\tau}$, one can define the functions (or distributions) $\widetilde{\psi}_o^{j,\ell,\iota,\tau}$ by $\widehat{\widetilde{\psi}_o^{j,\ell,\iota,\tau}}(\mathsf{A}^{j,\iota}\xi) = \widehat{b}_o^{j,\ell,\iota,\tau}(\xi)\widehat{\varphi}(\xi)$, $\xi \in \mathbb{R}^2$. One can then form a sequence of affine systems $\mathcal{AS}_J(\varphi; \{\widetilde{\psi}_o^{j,\ell,\iota,\tau} : \ell, \iota, \tau, o\}_{j=J}^{\infty})$ similar to those defined in [16]. Under certain subsampling relations [16, Theorem 5], one can obtain a sequence of affine shear systems (with 2-

layer structure), which we detail below.

Recall that $\mathsf{E}_X = \mathsf{I}_2$ and $\mathsf{E}_Y = \left[\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right]$, $\iota \in \{in, out\}$, and $o \in \{X, Y\}$. We define $\mathsf{S}^\ell := \left[\begin{smallmatrix} 1 & \ell \\ 0 & 1 \end{smallmatrix}\right]$ and $\mathsf{S}_\ell := \left[\begin{smallmatrix} 1 & 0 \\ \ell & 1 \end{smallmatrix}\right]$ as the shear matrices. At scale $j$, we consider the anisotropic dilation matrices $\mathsf{A}^{j,in} := \mathrm{diag}(2^{j-1/2}, \sqrt{2}^{j-1/2})$ and $\mathsf{A}^{j,out} := \mathrm{diag}(2^j, \sqrt{2}^j)$ for the inner and outer layers, respectively. We also define $\mathsf{B}^{j,\iota} := (\mathsf{A}^{j,\iota})^{-1}$ for $\iota \in \{in, out\}$.

An affine shear system is obtained by applying shear, dilation, and translation to generators at different scales. To balance the shear operation, we consider cone-adapted systems [12], [14], [15], [21], which usually consist of three subsystems: one subsystem covers the low frequency region, one subsystem covers the horizontal cone $X := \{\xi = (\xi_1, \xi_2) \in \mathbb{R}^2 : |\xi_2/\xi_1| \le 1\}$, and one subsystem covers the vertical cone $Y := \{\xi = (\xi_1, \xi_2) \in \mathbb{R}^2 : |\xi_1/\xi_2| \le 1\}$ in the frequency plane. Unlike the affine shear systems introduced in [16] that have only a single layer structure, in this paper we introduce affine shear systems with 2-layer structure. That is, at each scale $j$, the horizontal and vertical cones are further divided into an inner layer and an outer layer. More precisely, Let $\{\varphi\} \cup \Psi^{j,in} \cup \Psi^{j,out}$ to be the set of generators in $L_2(\mathbb{R}^2)$ with $\Psi^{j,\iota} := \{\psi_X^{j,\ell,\iota}, \psi_Y^{j,\ell,\iota} : |\ell| \le s_j^\iota\}$, where $s_j^{in}, s_j^{out}$ are nonnegative integers. An *affine shear system* (with 2-layer structure and with the initial scale $J$) is then defined to be

$$\begin{aligned} &\mathcal{AS}_J(\varphi; \{\Psi^{j,in}, \Psi^{j,out}\}_{j=J}^\infty) := \{2^j \varphi(2^j \cdot - \mathsf{k}) : \mathsf{k} \in \mathbb{Z}^2\} \\ &\cup \{|\det \mathsf{A}^{j,\iota}|^{\frac{1}{2}} \psi_o^{j,\ell,\iota}(\mathsf{S}^{-\ell}\mathsf{A}^{j,\iota} \cdot - \mathsf{k}) : \mathsf{k} \in \mathbb{Z}^2, |\ell| \le s_j^\iota, \iota, o\}. \end{aligned} \tag{10}$$

In a nutshell, at scale $j$, the set $\{\varphi(2^j \cdot - \mathsf{k}) : \mathsf{k} \in \mathbb{Z}^2\}$ of functions covers the low-frequency spectrum. The set $\{\psi_o^{j,\ell,\iota}(\mathsf{S}^{-\ell}\mathsf{A}^{j,\iota} \cdot - \mathsf{k}) : \ell = -s_j^\iota, \dots, s_j^\iota, \mathsf{k} \in \mathbb{Z}^2\}$ covers the horizontal and vertical cone for $o = X, Y$ and inner and outer cone for $\iota = in, out$, respectively.

We say that $\mathcal{AS}_J(\varphi; \{\Psi^{j,in}, \Psi^{j,out}\}_{j=J}^\infty)$ is an *affine shear tight frame* (with 2-layer structure) for $L_2(\mathbb{R}^2)$ if all generators $\{\varphi\} \cup \{\Psi^{j,in}, \Psi^{j,out}\}_{j=J}^\infty \subseteq L_2(\mathbb{R}^2)$ and for all $f \in L_2(\mathbb{R}^2)$, $\|f\|_2^2 = \sum_{\mathsf{k}\in\mathbb{Z}^2} \left|\left\langle f, 2^J \varphi(2^J \cdot - \mathsf{k})\right\rangle\right|^2 + \sum_{j=J}^\infty \sum_{\iota,o} \sum_\ell \sum_{\mathsf{k}\in\mathbb{Z}^2} \left|\left\langle f, |\det \mathsf{A}^{j,\iota}|^{\frac{1}{2}} \psi_o^{j,\ell,\iota}(\mathsf{S}^{-\ell}\mathsf{A}^{j,\iota} \cdot - \mathsf{k})\right\rangle\right|^2$.

MRA structure is a very desirable property because a filter bank and a fast transform algorithm can be easily introduced based on it. As argued in [19], a sequence of affine systems intrinsically gives an MRA structure and hence a filter bank and its associated fast transform algorithm. We next follow the notion of a sequence of affine shear systems [16], [32] to construct our affine shear tight frames with 2-layer structure. Similar to [16, Theorem 2], one can give a complete characterization of a sequence of affine shear systems to be a sequence of affine shear tight frames. In this paper, we are interested in the case that all generators are nonnegative in the frequency domain (that is $\widehat{\varphi} \ge 0$ and $\widehat{\psi_o^{j,\ell,\iota}} \ge 0$ for all $j, \ell, \iota, o$), since it leads to simple characterization conditions and easy construction of digital affine shear filter banks. In such a case, we have the following simple characterization result for a sequence of affine shear systems (with 2-layer structure) to be a sequence of affine shear tight frames (with 2-layer structure)(see [32, Theorem 2.1]).

*Theorem 3:* Fixed $J_0 \in \mathbb{Z}$, let $\mathcal{AS}_J(\varphi; \{\Psi^{j,in}, \Psi^{j,out}\}_{j=J}^\infty)$ be defined as in (10) for integer $J \ge J_0$. Suppose $\widehat{h}(\xi) \ge 0$ for all $h \in \{\varphi\} \cup \{\Psi^{j,in}, \Psi^{j,out}\}_{j=J_0}^\infty$. Then for all $J \ge J_0$, $\mathcal{AS}_J(\varphi; \{\Psi^{j,in}, \Psi^{j,out}\}_{j=J}^\infty)$ is an affine shear tight frame for $L_2(\mathbb{R}^2)$ if and only if the following identities hold:

$$\lim_{j\to\infty} \langle |\widehat{\varphi}(2^{-j}\cdot)|^2, g\rangle = \langle 1, g\rangle \quad \forall g \in C_c^\infty(\mathbb{R}^2), \tag{11}$$

for all $j \ge J_0$,

$$|\widehat{\varphi}(2^{-j-1}\cdot)|^2 = |\widehat{\varphi}(2^{-j}\cdot)|^2 + \sum_{\iota,o} \sum_{\ell=-s_j^\iota}^{s_j^\iota} \left|\widehat{\psi_o^{j,\ell,\iota}}(\mathsf{S}_\ell \mathsf{B}^{j,\iota}\cdot)\right|^2, \tag{12}$$

and for every $h$ in $\{\varphi\} \cup \{\Psi^{j,in}, \Psi^{j,out}\}_{j=J_0}^\infty$,

$$\widehat{h}(\xi)\widehat{h}(\xi + 2\pi\mathsf{k}) = 0, \quad a.e.\ \xi \in \mathbb{R}^2, \mathsf{k} \in \mathbb{Z}^2 \backslash \{0\}. \tag{13}$$

The conditions in Theorem 3 greatly simplify our construction of affine shear tight frames with 2 layer structure. In fact, condition (12) implies that we can obtain the generators $\psi_o^{j,\ell,\iota}$ from the splitting of $|\widehat{\varphi}(2^{-j-1}\cdot)|^2 - |\widehat{\varphi}(2^{-j}\cdot)|^2$, and condition (13) shows that the generators need to have nonoverlap $2\pi\mathbb{Z}^2$ shifts. We next briefly present the construction of affine shear tight frames (with 2-layer structure).

Let $\boldsymbol{\nu}_{[c,\epsilon]}$ be the 1D 'bump' function as given in Section II. Let $\boldsymbol{\gamma}_\varepsilon := \boldsymbol{\nu}_{[1/2,\varepsilon]}$ for $0 < \varepsilon \le 1/2$ and $\boldsymbol{\alpha} := \boldsymbol{\nu}_{[c,\epsilon]}$ for $c + \epsilon < \frac{1}{1+2\varepsilon} \cdot \frac{\pi}{2}$. Define 2D functions $\boldsymbol{\gamma}(\xi_1, \xi_2) := \boldsymbol{\gamma}_\varepsilon(\xi_2/\xi_1)$ and $\widehat{\varphi} := \boldsymbol{\alpha} \otimes \boldsymbol{\alpha}$. Define $\boldsymbol{\omega}^{in}, \boldsymbol{\omega}^{out}$ as follow:

$$\begin{aligned} \boldsymbol{\omega}^{out} &:= \sqrt{|\widehat{\varphi}(2^{-1}\cdot)|^2 - |\widehat{\varphi}(2^{-1/2}\cdot)|^2}, \\ \boldsymbol{\omega}^{in} &:= \sqrt{|\widehat{\varphi}(2^{-1/2}\cdot)|^2 - |\widehat{\varphi}(\cdot)|^2}. \end{aligned} \tag{14}$$

For $\lambda > 1$, define $\ell_\lambda := \lfloor \lambda - (1/2 + \varepsilon) \rfloor + 1 = \lfloor \lambda + (1/2 - \varepsilon) \rfloor$, $\lambda_{j,in} := \sqrt{2}^{j-1/2}$, and $\lambda_{j,out} := \sqrt{2}^j$. Define $\boldsymbol{\Gamma}_j^\iota(\xi) := \sum_o \sum_{\ell=-s_j^\iota}^{s_j^\iota} |\boldsymbol{\gamma}(\mathsf{S}_\ell \mathsf{B}^{j,\iota} \mathsf{E}_o \xi)|^2 = \sum_{\ell=-s_j^\iota}^{s_j^\iota} \left(|\boldsymbol{\gamma}_\varepsilon(\lambda_{j,\iota} \frac{\xi_2}{\xi_1} + \ell)|^2 + |\boldsymbol{\gamma}_\varepsilon(\lambda_{j,\iota} \frac{\xi_1}{\xi_2} + \ell)|^2\right)$ for $\xi \ne 0$ and $\iota \in \{in, out\}$, where $s_j^\iota = \ell_{\lambda_{j,\iota}}$. It is easy to check that $\boldsymbol{\Gamma}_j^\iota$ has the following properties: $0 < \boldsymbol{\Gamma}_j^\iota \le 2$, $\boldsymbol{\Gamma}_j^\iota(\mathsf{E}_o \cdot) = \boldsymbol{\Gamma}_j^\iota(\cdot)$, and $\boldsymbol{\Gamma}_j^\iota(t\xi) = \boldsymbol{\Gamma}_j^\iota(\xi)$ for $\xi \ne 0$.

Now define $\psi_o^{j,\ell,\iota}$ by

$$\begin{aligned} \widehat{\psi_X^{j,\ell,\iota}} &:= \boldsymbol{\omega}^\iota(2^{-j}(\mathsf{S}_\ell \mathsf{B}_{\lambda_j^\iota})^{-1}\xi) \frac{\boldsymbol{\gamma}_\varepsilon(\xi_2/\xi_1)}{\sqrt{\boldsymbol{\Gamma}_j^\iota((\mathsf{S}_\ell \mathsf{B}_{\lambda_j^\iota})^{-1}\xi)}}, \\ \widehat{\psi_Y^{j,\ell,\iota}} &:= \boldsymbol{\omega}^\iota(2^{-j}(\mathsf{S}_\ell \mathsf{B}_{\lambda_j^\iota})^{-1}\xi) \frac{\boldsymbol{\gamma}_\varepsilon(\xi_1/\xi_2)}{\sqrt{\boldsymbol{\Gamma}_j^\iota((\mathsf{S}_\ell \mathsf{B}_{\lambda_j^\iota})^{-1}\xi)}}, \end{aligned} \tag{15}$$

for $\iota \in \{in, out\}$. It is easy to show that (11), (12), and (13) hold for the above construction of $\widehat{\varphi}, \widehat{\psi_o^{j,\ell,\iota}}$. Consequently, we have the following result.

*Corollary 4:* Let $\mathcal{AS}_J(\varphi, \{\Psi^{j,in}, \Psi^{j,out}\}_{j=J}^\infty)$, $J \ge J_0$ be a sequence of affine shear systems with 2-layer structure and with $\varphi$, $\psi_o^{j,\ell,\iota}$ being defined as in (14), (15), respectively. Then

$\mathcal{AS}_J(\varphi, \{\Psi^{j,in}, \Psi^{j,out}\}_{j=J}^{\infty})$ is an affine shear tight frame for every $J \geq J_0$.

As proved in [16], an affine shear tight frame has a underlying (generalized) filter bank [19]. Similarly, for an affine shear tight frame with 2-layer structure given above, it also has an underlying (generalized) DAS-2 filter bank. Indeed, define $\widehat{a}(\xi) := \frac{\widehat{\varphi}(2\xi)}{\widehat{\varphi}(\xi)}$, $\widehat{b^\iota}(\xi) := \frac{\boldsymbol{\omega}^\iota(2\xi)}{\varphi(\xi)}$, and $\widehat{b_o^{j,\ell,\iota}}(\xi) := \frac{\widehat{\psi_o^{j,\ell,\iota}}(\mathsf{S}_\ell \mathsf{B}^{j,\iota} 2^{j+1}\xi)}{\widehat{\varphi}(\xi)}$. Then it is easy to show $\widehat{b_X^{j,\ell,\iota}}(\xi) = \widehat{b^\iota}(\xi)\frac{\boldsymbol{\gamma}_\varepsilon(\lambda_{j,\iota}\frac{\xi_2}{\xi_1}+\ell)}{\boldsymbol{\Gamma}_j^\iota(\xi)}$ and $\widehat{b_Y^{j,\ell,\iota}}(\xi) = \widehat{b^\iota}(\xi)\frac{\boldsymbol{\gamma}_\varepsilon(\lambda_{j,\iota}\frac{\xi_1}{\xi_2}+\ell)}{\boldsymbol{\Gamma}_j^\iota(\xi)}$. Further splitting of $\widehat{b_o^{j,\ell,\iota}}$ to positive axis and negative axis, we obtain a sequence of DAS-2 filter banks similar to those in Section II. Moreover, fix $J_0 = 0$ and rearrange $\Psi^{j,in} = \Psi^{2j-1}, \Psi^{j,out} = \Psi^{2j}$; i.e., $\psi_o^{2j-1,\ell} := \psi_o^{j,\ell,in}$ and $\psi_o^{2j,\ell} := \psi_o^{j,\ell,out}$. Then, the affine shear tight frame with 2-layer structure $\mathcal{AS}_J(\varphi, \{\Psi^{j,in}, \Psi^{j,out}\}_{j=J}^{\infty})$ can be regarded as a special case of an affine shear tight frame $\mathcal{AS}_J(\varphi, \{\Psi^j\}_{j=0}^{\infty})$ in [16] with $\lambda = \sqrt{2}$. Since the affine shear tight frame provides (nearly) optimal sparse approximation for cartoon-like functions [14], we conclude that our affine shear tight frame with 2-layer structure also provides (nearly) optimal sparse approximation for cartoon-like functions.

We remark that though in the function setting, our affine shear tight frames with the 2-layer structure can be regraded as a special case of the affine shear tight frames using non-dyadic scale parameter, yet in practice, our DAS-2 filter banks cannot be replaced by applying the DAS-1 filter banks iteratively. One of the reasons is that the downsampling process causes the information loss of the images. More precisely, due to the downsampling process, the low-pass filtered image (obtained from the DAS-1 filter bank at previous level) to be processed already lost certain information which originally might be captured if the 2-layer filter bank is used in the previous level. Thus, the two-layer structure is necessarily and important when one would like to extract more information from a given image at certain level.

## ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions that help to improve the quality and acceptability of the paper. The research of this paper was supported in part by the Research Grants Council of Hong Kong (Project No. CityU 11304414) and City University of Hong Kong (Project No.: 7200462 and 7004445).

## REFERENCES

[1] J.-P. Antoine, R. Murenzi, and P. Vanderheynst, Directional wavelets revisited: Cauchy wavelets and symmetry detection in pattern, *Appl. Comput. Harmon. Anal.* **6** (3) (1999) 314 – 345.

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

[3] 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** (2) (2004) 219–266.

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

[5] C.-C. Chang and C.-J. Lin. A library for support vector machines. *ACM Trans. Intell. Syst. Technol.* **27** (2011) 271–27.

[6] A. L. Cunha, J. Zhou, and M. N. Do, The nonsubsampled contourlet transform: Theory, design, and applications, *IEEE Trans. Image Proc.* **15** (10) (2006) 3089–3101.

[7] S. Dahlke, G. Kutyniok, G. Steidl, and G. Teschke, Shearlet coorbit spaces and associated Banach frames, *Appl. Comput. Harmon. Anal.* **27** (2) (2009) 195–214.

[8] I. Daubechies, Ten Lectures on Wavelets, CBMS-NSF Regional Conference Series in Applied Mathematics, **61**, SIAM, Philadelphia, PA, 1992.

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

[10] D. L. Donoho, G. Kutyniok, M. Shahram, and X. Zhuang, A rational design of a digital shearlet transform. The 9th International Conference on Sampling Theory and Applications (SampTA 2011), Singapore.

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

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

[13] K. Guo, D. Labate, W. Lim, G. Weiss, and E. Wilson, Wavelets with composite dilations and their MRA properties, *Appl. Comput. Harmon. Anal.* **20** (2006) 231–249.

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

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

[16] B. Han and X. Zhuang, Smooth affine shear tight frames with MRA structures. *Appl. Comput. Harmon. Anal.* **39** (2) (2015) 300–338.

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

[18] B. Han, Z. Zhao, and X. Zhuang, Directional tensor product complex tight framelets with low redundancy. *Appl. Comput. Harmon. Anal.* **41** (2) (2016) 603 – 637.

[19] B. Han. Nonhomogeneous wavelet systems in high dimensions. *Appl. Comput. Harmon. Anal.* **32** (2) (2012) 169 – 196.

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

[21] P. Kittipoom, G. Kutyniok, and W.-Q. Lim, Construction of compactly supported shearlet frames, *Constr. Appr.* **35** (1) (2012) 21–72.

[22] G. Kutyniok and D. Labate, Resolution of the wavefront set using continuous shearlets. *Trans. Amer. Math. Soc.* **361** (2009) 2719–2754.

[23] G. Kutyniok, M. Sharam, and X. Zhuang, ShearLab: A rational design of a digital parabolic scaling algorithm, *SIAM J. Imag. Sci.* **5** (4) (2012) 1291–1332.

[24] Y.-R. Li, R. H. Chan, L. Shen, Y.-C. Hsu, and W.-Y. I. Tseng, An adaptive directional Haar framelet-based reconstruction algorithm for parallel magnetic resonance imaging, *SIAM J. Imag. Sci.* **9** (2) (2016) 794–821.

[25] W.-Q. Lim, The discrete shearlet transform: A new directional transform and compactly supported shearlet frames, *IEEE Trans. Image Proc.* **19** (5) (2010) 1166–1180.

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

[27] S. Mallat, A Wavelet Tour of Signal Processing, Academic Press, 2008.

[28] W. R. Schwartz, R. D. Da Silva, L. S. Davis, and H. Pedrini. A novel feature descriptor based on the shearlet transform. *IEEE Intl. Conf. Imag. Proc.* (2011) 1033 – 1086.

[29] I. W. Selesnick, R. G. Baraniuk, and N. G. Kingsbury, The dual-tree complex wavelet transform, *IEEE Signal Process. Mag.* **22** (6) (2005) 123–151.

[30] Y. Shen, B. Han, and E. Braverman, Image inpainting from partial noisy data by directional complex tight framelets, Anziam Journal, 2017, 58(3-4):1-9.

[31] Y. Xu, S.-B. Huang, H. J., and C. Fermuller. Combining Powerful Local and Global Statistics for Texture Description. *CVPR* (2009) 573–580.

[32] X. Zhuang. Digital affine shear transforms: fast realization and applications in image/video processing. *SIAM J. Imag. Sci.* **9** (3) (2016) 1437 – 1466.