Appl. Comput. Harmon. Anal. 41 (2016) 603–637

Contents lists available at ScienceDirect

Applied and Computational Harmonic Analysis

www.elsevier.com/locate/acha

Directional tensor product complex tight framelets with low redundancy ☆

Bin Han [a,*], Zhenpeng Zhao [a], Xiaosheng Zhuang [b]

[a] Department of Mathematical and Statistical Sciences, University of Alberta, Edmonton, Alberta T6G 2G1, Canada
[b] Department of Mathematics, City University of Hong Kong, Tat Chee Avenue, Kowloon Tong, Hong Kong

ARTICLE INFO

Article history:
Received 13 December 2014
Received in revised form 18 June 2015
Accepted 21 July 2015
Available online 29 July 2015
Communicated by Spec. Issue Guest Editor

MSC:
42C40
42C15
94A08
68U10

Keywords:
Directional complex tight framelets
Tight framelet filter banks with mixed sampling factors
Redundancy rate
Tensor product
Image and video denoising
Image and video inpainting

ABSTRACT

Having the advantages of redundancy and flexibility, various types of tight frames have already shown impressive performance in applications such as image and video processing. For example, the undecimated wavelet transform, which is a particular case of tight frames, is known to have good performance for the denoising problem. Empirically, it is widely known that higher redundancy rate of a tight frame often leads to better performance in applications. The wavelet/framelet transform is often implemented in an undecimated fashion for the purpose of better performance in practice. Though high redundancy rate of a tight frame can improve performance in applications, as the dimension increases, it also makes the computational cost skyrocket and the storage of frame coefficients increase exponentially. This seriously restricts the usefulness of such tight frames for problems in moderately high dimensions such as video processing in dimension three. Inspired by the directional tensor product complex tight framelets $\text{TP-}\mathbb{C}\text{TF}_m$ with $m \ge 3$ in [15,20] and their impressive performance for image processing in [20,33], in this paper we introduce directional tensor product complex tight framelets $\text{TP-}\mathbb{C}\text{TF}_m^{\downarrow}$ (called reduced $\text{TP-}\mathbb{C}\text{TF}_m$) with low redundancy. Such $\text{TP-}\mathbb{C}\text{TF}_m^{\downarrow}$ are particular examples of tight framelet filter banks with mixed sampling factors. In particular, we shall develop a directional tensor product complex tight framelet $\text{TP-}\mathbb{C}\text{TF}_6^{\downarrow}$ such that it performs nearly as well as the original $\text{TP-}\mathbb{C}\text{TF}_6$ in [20] for image/video denoising/inpainting but it has significantly lower redundancy rates than $\text{TP-}\mathbb{C}\text{TF}_6$ in every dimension. The $\text{TP-}\mathbb{C}\text{TF}_6^{\downarrow}$ in d dimensions not only offers good directionality as the original $\text{TP-}\mathbb{C}\text{TF}_6$ does but also has the low redundancy rate $\frac{3^d-1}{2^d-1}$ (e.g., the redundancy rates are $2, 2\frac{2}{3}, 3\frac{5}{7}, 5\frac{1}{3}$ and $7\frac{25}{31}$ for dimension $d = 1, \dots, 5$, respectively), in comparison with the redundancy rate $2^d \times \frac{3^d-1}{2^d-1}$ of $\text{TP-}\mathbb{C}\text{TF}_6$ in dimension d. Moreover, our numerical experiments on image/video denoising and inpainting show that the performance using our proposed $\text{TP-}\mathbb{C}\text{TF}_6^{\downarrow}$ is often comparable with or

☆ Research of Bin Han and Zhenpeng Zhao was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC Canada under Grant 05865). Research of Xiaosheng Zhuang was supported by the Research Grants Council of Hong Kong (Project No. CityU 108913).
* Corresponding author. Fax: +1 780 4926826.
E-mail addresses: bhan@ualberta.ca (B. Han), zzhao7@ualberta.ca (Z. Zhao), xzhuang7@cityu.edu.hk (X. Zhuang).
URL: http://www.ualberta.ca/~bhan (B. Han).

http://dx.doi.org/10.1016/j.acha.2015.07.003

sometimes better than several state-of-the-art frame-based methods which have much higher redundancy rates than that of TP-$\mathbb{C}\mathrm{TF}_6^{\downarrow}$.

1. Introduction and motivations

Though wavelets have many useful applications, they have several shortcomings in dealing with multi-dimensional problems. For example, tensor product real-valued wavelets are known for lack of the desired properties of translation invariance and directionality [6,23,31]. There are a lot of papers in the current literature to improve the performance of classical tensor product (i.e., separable) real-valued wavelets by remedying these two shortcomings. In one direction, translation invariance of wavelets can be improved by using wavelet frames instead of orthonormal wavelets (see [6,7,12,14,16–18,30,31] and many references therein). For example, the undecimated wavelet transform [6] using Daubechies orthonormal wavelets has been known to be effective for the denoising problem. In fact, such an undecimated wavelet transform employs a particular case of tight frames with high redundancy. A countable set $\{h_k\}_{k\in\Lambda}$ of elements in a Hilbert space $\mathcal{H}$ equipped with an inner product $\langle\cdot,\cdot\rangle$ is called *a frame* if there exist positive constants C_1 and C_2 such that

$$C_1\langle h,h\rangle \leqslant \sum_{k\in\Lambda} |\langle h,h_k\rangle|^2 \leqslant C_2\langle h,h\rangle, \qquad \forall\, h\in\mathcal{H}.$$

In particular, it is called a (normalized) *tight frame* if $C_1 = C_2 = 1$. If $\mathcal{H}$ is a finite dimensional space with dimension d, then the *redundancy rate* of a frame $\{h_k\}_{k\in\Lambda}$ is naturally defined to be $\frac{\#\Lambda}{d}$, where $\#\Lambda$ is the cardinality of the index set Λ. Note that an orthonormal basis in $\mathcal{H}$ is a particular tight frame with the redundancy rate one. Comparing with an orthonormal basis, a (tight) frame is more general and has redundancy by allowing more elements into its system. The added redundancy of a tight frame not only improves the property of translation invariance but also makes the design of a tight frame more flexible (see [6,7,12,14,16–18,21,30] and references therein). In the other direction, many papers in the literature have been studying directional representation systems, to only mention a few here, curvelets in [1,2,34], contourlets in [8], shearlets in [10,11,21,23–25,27,28] and many references therein, surfacelets in [29], dual tree complex wavelet transform in [22,31,32], complex tight framelets in [14,15,17,18,20], plus many other directional representation systems. To improve directionality of tensor product real-valued wavelets, due to the requirement of the additional angular resolution for a directional representation system, it is almost unavoidable to employ either a tight frame or a frame instead of an orthonormal basis by allowing redundancy. In fact, to our best knowledge, all currently known representation systems, having either directionality and/or (near) translation invariance, employ either a frame or a tight frame with various degrees of redundancy. The directional tensor product complex tight framelets in [15,20] and their reduced versions with low redundancy in this paper are different in nature from many known directional representation systems such as curvelets and shearlets. This issue will be addressed and explained in details in Section 3.2.

In the following, let us introduce a fast framelet transform and explain by what we mean the redundancy rate of a transform or a system. To this end, let us recall the definition of a tight framelet filter bank. For $u = \{u(k)\}_{k\in\mathbb{Z}^d} \in l_1(\mathbb{Z}^d)$, we define the *Fourier series* (or *symbol*) $\widehat{u}$ of the sequence u to be $\widehat{u}(\xi) := \sum_{k\in\mathbb{Z}^d} u(k)e^{-ik\cdot\xi}$, $\xi \in \mathbb{R}^d$. Note that $\widehat{u}$ is a $2\pi\mathbb{Z}^d$-periodic function satisfying $\widehat{u}(\xi + 2\pi k) = \widehat{u}(\xi)$ for all $k \in \mathbb{Z}^d$. For $a, b_1, \dots, b_s \in l_1(\mathbb{Z}^d)$, we say that $\{a; b_1, \dots, b_s\}$ is a (d-dimensional dyadic) *tight framelet filter bank* if

$$|\widehat{a}(\xi)|^2+\sum_{\ell=1}^{s}|\widehat{b_\ell}(\xi)|^2=1 \quad \text{and} \quad \widehat{a}(\xi)\overline{\widehat{a}(\xi+\pi\omega)}+\sum_{\ell=1}^{s}\widehat{b_\ell}(\xi)\overline{\widehat{b_\ell}(\xi+\pi\omega)}=0, \qquad \forall\, \omega\in([0,1]^d\cap\mathbb{Z}^d)\backslash\{0\}$$

for almost every $\xi\in\mathbb{R}^d$. Moreover, a (d-dimensional dyadic) tight framelet filter bank $\{a;b_1,\dots,b_s\}$ with $s=2^d-1$ is called a (d-dimensional dyadic) *orthonormal wavelet filter bank*. A d-dimensional tight framelet (or orthonormal wavelet) filter bank is often obtained through tensor product. For one-dimensional filters $u_1,\dots,u_d\in l_1(\mathbb{Z})$, we define their d-dimensional tensor product filter $u_1\otimes\cdots\otimes u_d$ by $(u_1\otimes\cdots\otimes u_d)(k_1,\dots,k_d):=u_1(k_1)\cdots u_d(k_d)$ for $k_1,\dots,k_d\in\mathbb{Z}$. In particular, we define $\otimes^d u:=u\otimes\cdots\otimes u$ with d copies of u. If $\{a;b_1,\dots,b_s\}$ is a one-dimensional (dyadic) tight framelet filter bank (or an orthonormal wavelet filter bank with $s=1$), then it is straightforward to check that $\otimes^d\{a;b_1,\dots,b_s\}$ is a d-dimensional dyadic tight framelet (or orthonormal wavelet) filter bank. As discussed in [13,14], tight framelet filter banks are closely linked to tight framelets in $L_2(\mathbb{R}^d)$. See [7,12–14,16–18,30] as well as Section 2 for connections of tight framelet filter banks with tight framelets in $L_2(\mathbb{R}^d)$.

A fast wavelet/framelet transform is implemented through the operations of convolution and sampling. Let $v\in l_\infty(\mathbb{Z}^d)$ be a d-dimensional input signal and let u be a filter from a given d-dimensional tight framelet filter bank $\{a;b_1,\dots,b_s\}$. Roughly speaking, for the decomposition/forward transform, the data v is first convolved with the flip-conjugate filter $u^\star$ (that is, $u^\star(k):=\overline{u(-k)}, k\in\mathbb{Z}^d$) as $v*u^\star:=\sum_{k\in\mathbb{Z}^d}v(k)u^\star(\cdot-k)$ and then it is downsampled as $w:=(v*u^\star)\downarrow 2I_d:=(v*u^\star)(2\cdot)$, where w is called the sequence of frame coefficients. The decomposition transform can be applied recursively J times with v being replaced by $(v*a^\star)\downarrow 2I_d$ (that is, $u=a$) as the new input data, where $J\in\mathbb{N}$ is the decomposition level. For the reconstruction/backward transform, the coefficient sequence w is upsampled as $(w\uparrow 2I_d)(k):=w(k/2)$ if $k\in 2\mathbb{Z}^d$ and $(w\uparrow 2I_d)(k):=0$ for $k\in\mathbb{Z}^d\backslash[2\mathbb{Z}^d]$, and then it is convolved with u as $(w\uparrow 2I_d)*u$. Finally, all the reconstructed sequences are added together as one reconstructed data. See Fig. 2 for an illustration of a two-level fast framelet transform employing a one-dimensional dyadic tight framelet filter bank $\{a;b_1,\dots,b_s\}$ (but with $\sqrt{4},\downarrow 4,\uparrow 4$ in Fig. 2 being replaced by $\sqrt{2},\downarrow 2,\uparrow 2$, respectively). See Section 2 for more details on a fast framelet transform.

Most d-dimensional problems and data in applications have finite length. For a given real-valued data v of finite length, one first extends it into a periodic sequence v^e on $\mathbb{Z}^d$. Then one performs a wavelet/framelet transform on the extended data v^e. This induces a linear transform on the original data v and the decomposition transform can be rewritten using a matrix $\mathcal{W}$. More precisely, we can arrange the d-dimensional real-valued data v properly so that it can be regarded as an $n\times 1$ column vector in $\mathbb{R}^n$, that is, $v\in\mathbb{R}^n$. Performing a linear transform $\mathcal{W}$ on v, we obtain another column vector $w:=\mathcal{W}v\in\mathbb{R}^N$ of frame coefficients. If $\{a;b_1,\dots,b_s\}$ with $s=2^d-1$ is a real-valued orthonormal wavelet filter bank, then $N=n$ and $\mathcal{W}$ is a real-valued $n\times n$ orthogonal matrix satisfying $\mathcal{W}^\mathsf{T}\mathcal{W}=I_n$. If $\{a;b_1,\dots,b_s\}$ is a real-valued tight framelet filter bank, then we must have $N\ge n$ and $\mathcal{W}$ is a real-valued $N\times n$ matrix satisfying $\mathcal{W}^\mathsf{T}\mathcal{W}=I_n$. Therefore, the ratio N/n is the redundancy rate of the linear transform $\mathcal{W}$ or its underlying tight frame, since it is the ratio between the N number of frame coefficients over the n number of original input data. Also note that the redundancy rate N/n is independent of the length n of input data and depends only on the number s of high-pass filters and the sampling factor (which is $2I_d$ here).

We now look at the redundancy rate of an undecimated wavelet/framelet transform (denoted by UFT_s) using tensor products of a one-dimensional real-valued tight framelet filter bank $\{a;b_1,\dots,b_s\}$ (when $s=1$, it is an orthonormal wavelet filter bank and UFT_1 becomes UWT—the undecimated wavelet transform). Here the word undecimated means that the upsampling and downsampling operations in a standard wavelet/framelet transform are completely removed. Undecimated framelet transforms using spline tight framelet filter banks $\{a_2^B;\mathring{b}_1,\mathring{b}_2\}$ and $\{a_4^B;b_1,b_2,b_3,b_4\}$ with $\widehat{a_m^B}(\xi):=2^{-m}(1+e^{-i\xi})^m$ have applications to several image restoration problems as reported in [3–5,9,26] and many references therein. The tensor product d-dimensional tight framelet filter bank is $\otimes^d\{a;b_1,\dots,b_s\}$ which consists of one real-valued low-pass filter $\otimes^d a$ and $(s+1)^d-1$ real-valued high-pass filters. If the decomposition level is $J\in\mathbb{N}$, the redundancy rate of

Table 1
Comparison of redundancy rates of various tight frames for different dimensions d. UWT is the undecimated wavelet transform with decomposition level $J = 3$ and using the tensor product of a 1D real-valued orthonormal wavelet filter bank $\{a; b\}$. UFT$_s$ is the undecimated framelet transform with decomposition level $J = 3$ and using the tensor product of a 1D real-valued tight framelet filter bank $\{a; b_1, \dots, b_s\}$. (Hence, UWT is just UFT$_1$.) DT-$\mathbb{C}$WT is the dual tree complex wavelet transform. TP-$\mathbb{C}$TF$_m$ is the tensor product complex tight framelet with $m = 3, 4, 5, 6$. TP-$\mathbb{C}$TF$_6^{\downarrow}$ is our proposed tensor product complex tight framelet with low redundancy. It is interesting to point out here that TP-$\mathbb{C}$TF$_6^{\downarrow}$ has the same low redundancy rate as TP-$\mathbb{C}$TF$_3$, but TP-$\mathbb{C}$TF$_6^{\downarrow}$ enjoys the same directionality as TP-$\mathbb{C}$TF$_6$.

d	1	2	3	4	5	6	7	8	9	10
UWT	4	10	22	46	94	190	383	766	1534	3070
UFT$_2$	7	25	79	241	727	2185	6559	19 681	59 047	177 145
UFT$_4$	13	73	373	1873	9373	46 873	234 373	1 171 873	5 859 373	29 296 873
DT-$\mathbb{C}$WT	2	4	8	16	32	64	128	256	512	1024
TP-$\mathbb{C}$TF$_3$	2	$2\frac{2}{3}$	$3\frac{5}{7}$	$5\frac{1}{3}$	$7\frac{25}{31}$	$11\frac{5}{9}$	$17\frac{27}{127}$	$25\frac{37}{51}$	$38\frac{264}{511}$	$57\frac{67}{93}$
TP-$\mathbb{C}$TF$_4$	2	4	8	16	32	64	128	256	512	1024
TP-$\mathbb{C}$TF$_5$	4	8	$17\frac{5}{7}$	$41\frac{3}{5}$	$100\frac{24}{31}$	248	$615\frac{19}{127}$	$1531\frac{73}{85}$	$3822\frac{82}{511}$	$9546\frac{2}{31}$
TP-$\mathbb{C}$TF$_6$	4	$10\frac{2}{3}$	$29\frac{5}{7}$	$85\frac{1}{3}$	$249\frac{25}{31}$	$739\frac{5}{9}$	$2203\frac{27}{127}$	$6585\frac{37}{51}$	$19\,720\frac{264}{511}$	$59\,105\frac{67}{93}$
TP-$\mathbb{C}$TF$_6^{\downarrow}$	2	$2\frac{2}{3}$	$3\frac{5}{7}$	$5\frac{1}{3}$	$7\frac{25}{31}$	$11\frac{5}{9}$	$17\frac{27}{127}$	$25\frac{37}{51}$	$38\frac{264}{511}$	$57\frac{67}{93}$

the d-dimensional undecimated framelet transform using the tensor product real-valued tight framelet filter bank $\otimes^d\{a; b_1, \dots, b_s\}$ is $((s+1)^d - 1)J + 1$. To take advantages of the multiscale structure of wavelets, it is necessary that the decomposition level J should be as high as possible by taking into account the resolution of a given data or image. For example, for a standard 512×512 grayscale image, the wavelet decomposition level is often set to be at least $J = 5$ (note that $512 = 2^9$). Let us here take a moderate choice of $J = 3$ (for a typical 512×512 grayscale image) and use the smallest $s = 1$ (that is, we are using an orthonormal wavelet filter bank). For dimension $d = 3$ and $J = 3$, the redundancy rate of a tensor product undecimated wavelet transform is 22. However, as we mentioned before, tensor product real-valued orthonormal wavelets lack directionality and translation invariance. To improve directionality or translation invariance, we must use a tight framelet filter bank with $s \geq 2$. For $d = 3$ and $J = 3$, the redundancy rates of UFTs are $22, 79, 190, 373, 646$ for $s = 1, \dots, 5$, respectively. See Table 1 for a numerical illustration on redundancy rates of an undecimated wavelet/framelet transform.

By employing a pair of two correlated one-dimensional real-valued orthonormal wavelet filter banks, the dual tree complex wavelet transform (DT-$\mathbb{C}$WT) offers directionality and (near) translation invariance with the redundancy rate 2^d in d dimensions for any decomposition level $J \in \mathbb{N}$. See [22,31,32] and [20, Section 2] as well as references therein for more details on DT-$\mathbb{C}$WT. One-dimensional finitely supported complex-valued tight framelet filter banks have been extensively studied in [16–18]. A family of directional tensor product complex tight framelet filter banks (TP-$\mathbb{C}$TF) has been initially introduced in [15] and further developed in [20] for the purpose of image denoising. The family of one-dimensional complex tight framelet filter banks introduced and used in [15,20] is $\mathbb{C}$TF$_m$, where $m \geq 3$ is the total number of filters in $\mathbb{C}$TF$_m$. The low-pass filter in $\mathbb{C}$TF$_m$ is real-valued but its high-pass filters are complex-valued. If m is odd, then the d-dimensional tensor product tight framelet filter bank TP-$\mathbb{C}$TF$_m$ has one real-valued low-pass filter and $m^d - 1$ complex-valued high-pass filters. Consequently, its redundancy rate is no more than $\frac{m^d-1}{2^d-1}$ for dimension d and for any decomposition level $J \in \mathbb{N}$. If m is even, then the d-dimensional tensor product tight framelet filter bank TP-$\mathbb{C}$TF$_m$ has one real-valued low-pass filter and $m^d - 2^d$ complex-valued high-pass filters. Therefore, its redundancy rate is no more than $\frac{m^d-2^d}{2^d-1}$ for dimension d and for any decomposition level $J \in \mathbb{N}$. For both the dual tree complex wavelet transform DT-$\mathbb{C}$WT and the tensor product complex tight framelets TP-$\mathbb{C}$TF$_m$, a complex frame coefficient is counted as two real frame coefficients in the calculation of their redundancy rates. See Section 3 for more detailed explanation about the redundancy rates of TP-$\mathbb{C}$TF$_m$. The frequently used tensor product complex tight framelets for image denoising in [20] are TP-$\mathbb{C}$TF$_4$ and TP-$\mathbb{C}$TF$_6$. The TP-$\mathbb{C}$TF$_4$ has almost the same performance, directionality and redundancy rate as those of DT-$\mathbb{C}$WT. The TP-$\mathbb{C}$TF$_6$ has much better performance than TP-$\mathbb{C}$TF$_4$ and DT-$\mathbb{C}$WT for image denoising in [20] and image inpainting in [33], but it has higher redundancy rate $\frac{6^d-2^d}{2^d-1}$ for dimension d. See Table 1 for some numerical illustration on redundancy rates of TP-$\mathbb{C}$TF$_m$. See [15,20] as well as Section 3 for more detailed discussion on directional tensor product complex tight framelets and their redundancy rates.

Beyond the above tensor product (i.e., separable) transforms for multidimensional problems, to achieve directionality, there are also many nonseparable approaches. We shall use the notation dD to stand for d dimensions or d-dimensional. Some examples of such nonseparable transforms are 2D and 3D curvelets in [1,2,34], 2D contourlets in [8], 2D and 3D shearlets in [10,21,23,25,27,28] and references therein, 3D surfacelets in [29], and directional tight framelets in [12,14,21], etc. The redundancy rates of such nonseparable transforms depend on the choices of the numbers of directions at each resolution level and the decomposition level $J \in \mathbb{N}$. Generally speaking, to achieve reasonably good performance in applications, those nonseparable transforms often have much higher redundancy rates than those of the tensor product transforms using the dual tree complex wavelet transform and directional complex tight framelets. See Section 4 for the redundancy rates and performance of several nonseparable transforms using directional representation systems.

Though empirically higher redundancy rate of a tight frame often leads to better performance in applications, the computational costs increase exponentially with respect to higher redundancy rate and dimensionality. This causes serious constraints on computational expenses and storage requirement for multidimensional problems. To our best knowledge, most of the above mentioned directional representation systems and tight frames can achieve reasonably good performance with computational costs being manageable by a standard PC for two-dimensional problems. However, for applications in three or higher dimensions such as video processing, the expensive computational cost becomes a serious issue, without even mentioning the fact that one often tends to increase the redundancy rates in order to achieve reasonably good performance for applications in three or higher dimensions. This difficulty seriously restricts the usefulness of such tight frames and directional representation systems for multidimensional problems (in particular, for problems in moderately high dimensions such as video processing in dimension three). Motivated by the approach of directional tensor product complex tight framelets in [15,20], to remedy the above mentioned difficulty, in this paper we shall construct a tight wavelet frame having the following desired properties:

(i) The tight frame is obtained through the tensor product of a one-dimensional tight framelet filter bank.
(ii) The tight frame has low redundancy rate and all its high-pass elements have good directionality.
(iii) The tight frame has good performance for applications such as denoising and inpainting, comparing with more complicated directional representation systems and tight frames with much higher redundancy rates.

The tensor product structure in item (i) and low redundancy rate in item (ii) of such a tight frame make it computationally efficient and attractive, while low redundancy also significantly reduces the storage requirement for frame coefficients. Good directionality in item (ii) is needed in order to have good performance as required in item (iii). In this paper we shall achieve all the above goals by modifying the construction of directional tensor product complex tight framelet filter banks TP-$\mathbb{C}$TF$_m$ with $m \geq 3$ in [15,20]. Though our approach can be easily applied to all TP-$\mathbb{C}$TF$_m$, for simplicity of presentation, in this paper we mainly focus our attention to one particular example: the directional tensor product complex tight framelet TP-$\mathbb{C}$TF$_6$, whose underlying one-dimensional tight framelet filter bank is $\mathbb{C}$TF$_6$. As demonstrated in [20] for image denoising and in [33] for image inpainting, this TP-$\mathbb{C}$TF$_6$ has much better performance than DT-$\mathbb{C}$WT, TP-$\mathbb{C}$TF$_4$, curvelets, 2D shearlets, real-valued spline tight frames, discrete cosine transform, and many other frame-based methods. In this paper we significantly reduce the redundancy rate of TP-$\mathbb{C}$TF$_6$. As a consequence, we denote our modified directional tensor product complex tight framelet by TP-$\mathbb{C}$TF$_6^{\downarrow}$ and call it (redundancy) reduced TP-$\mathbb{C}$TF$_6$, where the superscript $\downarrow$ here means that TP-$\mathbb{C}$TF$_6^{\downarrow}$ is a reduced (or further downsampled) version of TP-$\mathbb{C}$TF$_6$ by decreasing its redundancy rate while trying to keep all the desirable properties of the original TP-$\mathbb{C}$TF$_6$. As we shall see in Section 3, the redundancy rate of TP-$\mathbb{C}$TF$_6^{\downarrow}$ is $\frac{3^d-1}{2^d-1}$ for dimension d and for any decomposition level $J \in \mathbb{N}$, while as we discussed before, the redundancy rate of TP-$\mathbb{C}$TF$_6$ is $\frac{6^d-2^d}{2^d-1} = 2^d \times \frac{3^d-1}{2^d-1}$ (that is, the redundancy rate of TP-$\mathbb{C}$TF$_6$ is 2^d times that of TP-$\mathbb{C}$TF$_6^{\downarrow}$ in dimension d). See Table 1 for an illustration and comparison of redundancy rates of various tight frames. The construction of other TP-$\mathbb{C}$TF$_m^{\downarrow}$ will also be briefly addressed in this paper. One of our main goals in this paper is to concretely construct a directional tensor product complex tight framelet TP-$\mathbb{C}$TF$_6^{\downarrow}$ such that it performs nearly as well as the original TP-$\mathbb{C}$TF$_6$ in [20] for image/video denoising/inpainting but has significantly lower redundancy rates than TP-$\mathbb{C}$TF$_6$ in every dimension.

The structure of the paper is as follows. In order to study tensor product complex tight framelets with low redundancy, in Section 2 we shall generalize the notion of dyadic tight framelet filter banks by introducing tight framelet filter banks with mixed sampling factors. Then we shall study their various properties and fast framelet transforms of such tight framelet filter banks with mixed sampling factors in Section 2. In Section 3, we shall recall the tensor product complex tight framelet filter banks TP-$\mathbb{C}$TF$_m$ and their underlying one-dimensional complex tight framelet filter banks $\mathbb{C}$TF$_m$ with $m \geq 3$ from [15,20]. Next, we shall provide some explanation for the directionality of tensor product complex tight framelets and their differences to several other known directional representation systems. We shall also discuss several features of TP-$\mathbb{C}$TF$_6$ and explain why we are particularly interested in TP-$\mathbb{C}$TF$_6$ instead of other TP-$\mathbb{C}$TF$_m$ with $m \geq 3$ for the purpose of image and video processing. Then we shall discuss the redundancy rates of TP-$\mathbb{C}$TF$_m$. Next we shall provide details on our construction of directional tensor product complex tight framelet TP-$\mathbb{C}$TF$_6^{\downarrow}$ with low redundancy. Such TP-$\mathbb{C}$TF$_6^{\downarrow}$ is a particular example of tight framelet filter banks with mixed sampling factors in Section 2. Though our approach can be easily applied to all TP-$\mathbb{C}$TF$_m$ with $m \geq 3$, for simplicity of presentation, we discuss TP-$\mathbb{C}$TF$_6$ in detail in Section 3 while we only outline the general idea for constructing other TP-$\mathbb{C}$TF$_m^{\downarrow}$. In Section 4, we shall test the performance of our proposed directional complex tight framelet TP-$\mathbb{C}$TF$_6^{\downarrow}$ with low redundancy rate and compare its performance with several state-of-the-art frame-based methods. Our numerical experiments on image/video denoising and inpainting show that the performance using our tensor product directional complex tight framelet TP-$\mathbb{C}$TF$_6^{\downarrow}$ with low redundancy is often comparable with or sometimes better than several state-of-the-art frame-based methods which often have much higher redundancy rates. Moreover, our numerical experiments show that TP-$\mathbb{C}$TF$_6^{\downarrow}$ is particularly effective for images and videos having rich textures.

2. Tight framelet filter banks with mixed sampling factors

In this section we shall introduce tight framelet filter banks with mixed sampling factors and then study their properties. As we shall see later in Section 3, our proposed directional tensor product complex tight framelet TP-$\mathbb{C}$TF$_6^{\downarrow}$ with low redundancy is a particular case of tight framelet filter banks with mixed sampling factors.

2.1. Fast framelet transform using tight framelet filter banks with mixed sampling factors

Our key idea to derive a directional tight framelet with low redundancy from the tensor product complex tight framelet filter banks TP-$\mathbb{C}$TF$_m$ in [15,20] is to use higher sampling factors such as $4I_d$ instead of $2I_d$. To this end, let us generalize the definition of a (d-dimensional dyadic) tight framelet filter bank $\{a; b_1, \dots, b_s\}$, which uses the uniform sampling matrix $2I_d$, where I_d is the $d \times d$ identity matrix.

Let M be a $d \times d$ invertible integer matrix. For a sequence $u = \{u(k)\}_{k\in\mathbb{Z}^d} : \mathbb{Z}^d \to \mathbb{C}$, the downsampling sequence $u{\downarrow}\mathsf{M}$ and the upsampling sequence $u{\uparrow}\mathsf{M}$ with the sampling matrix M are defined by

$$[u{\downarrow}\mathsf{M}](k) := u(\mathsf{M}k), \quad k \in \mathbb{Z}^d \quad \text{and} \quad [u{\uparrow}\mathsf{M}](k) := \begin{cases} u(\mathsf{M}^{-1}k), & \text{if } k \in \mathsf{M}\mathbb{Z}^d, \\ 0, & \text{if } k \in \mathbb{Z}^d \backslash [\mathsf{M}\mathbb{Z}^d]. \end{cases}$$

We call M the *sampling factor or matrix*. When $\mathsf{M}\mathbb{Z}^d = \mathbb{Z}^d$ (that is, $|\det(\mathsf{M})| = 1$), $u \downarrow \mathsf{M}$ and $u \uparrow \mathsf{M}$ are essentially the same sequence u by rearranging its indices in $\mathbb{Z}^d$. To explicitly specify the sampling matrix M associated with a filter u, we shall adopt the notation $u\,!\,\mathsf{M}$. Under the new notation, a (d-dimensional dyadic) tight framelet filter bank $\{a; b_1, \dots, b_s\}$ will be denoted more precisely as $\{a\,!\,2I_d; b_1\,!\,2I_d, \dots, b_s\,!\,2I_d\}$, since the sampling matrix is uniformly $2I_d$.

For $1 \leqslant p < \infty$, $l_p(\mathbb{Z}^d)$ consists of all the sequences $v : \mathbb{Z}^d \to \mathbb{C}$ satisfying $\|v\|^p_{l_p(\mathbb{Z}^d)} := \sum_{k\in\mathbb{Z}^d} |v(k)|^p < \infty$. Similarly, $v \in l_\infty(\mathbb{Z}^d)$ if $\|v\|_{l_\infty(\mathbb{Z}^d)} := \sup_{k\in\mathbb{Z}^d} |v(k)| < \infty$. By $l_0(\mathbb{Z}^d)$ we denote the space of all finitely supported sequences on $\mathbb{Z}^d$.

A discrete framelet transform can be described using the subdivision operator and the transition operator. For a filter $u \in l_1(\mathbb{Z}^d)$ and a $d \times d$ integer matrix M, the *subdivision operator* $\mathcal{S}_{u,\mathsf{M}} : l_\infty(\mathbb{Z}^d) \to l_\infty(\mathbb{Z}^d)$ and the *transition operator* $\mathcal{T}_{u,\mathsf{M}} : l_\infty(\mathbb{Z}^d) \to l_\infty(\mathbb{Z}^d)$ are defined to be

$$[\mathcal{S}_{u,\mathsf{M}}v](n) := |\det(\mathsf{M})| \sum_{k\in\mathbb{Z}^d} v(k)u(n - \mathsf{M}k), \qquad n \in \mathbb{Z}^d,$$

$$[\mathcal{T}_{u,\mathsf{M}}v](n) := |\det(\mathsf{M})| \sum_{k\in\mathbb{Z}^d} v(k)\overline{u(k - \mathsf{M}n)}, \qquad n \in \mathbb{Z}^d,$$

for $v \in l_\infty(\mathbb{Z}^d)$. Since $u \in l_1(\mathbb{Z}^d)$ and $v \in l_\infty(\mathbb{Z}^d)$, we see that both $\mathcal{S}_{u,\mathsf{M}}v$ and $\mathcal{T}_{u,\mathsf{M}}v$ are well-defined sequences in $l_\infty(\mathbb{Z}^d)$. Define $\Omega_\mathsf{M} := [\mathsf{M}^{-\mathsf{T}}\mathbb{Z}^d] \cap [0, 1)^d$. In terms of the Fourier series, for $u, v \in l_1(\mathbb{Z}^d)$, we have

$$\widehat{\mathcal{S}_{u,\mathsf{M}}v}(\xi) = |\det(\mathsf{M})|\widehat{v}(\mathsf{M}^\mathsf{T}\xi)\widehat{u}(\xi), \qquad \widehat{\mathcal{T}_{u,\mathsf{M}}v}(\xi) = \sum_{\omega\in\Omega_\mathsf{M}} \widehat{v}(\mathsf{M}^{-\mathsf{T}}\xi + 2\pi\omega)\overline{\widehat{u}(\mathsf{M}^{-\mathsf{T}}\xi + 2\pi\omega)}. \tag{2.1}$$

Define the flip-conjugate sequence $u^\star$ of u by $u^\star(k) := \overline{u(-k)}, k \in \mathbb{Z}^d$, that is, $\widehat{u^\star}(\xi) = \overline{\widehat{u}(\xi)}$. Then $\mathcal{S}_{u,\mathsf{M}}v = |\det(\mathsf{M})|(v\uparrow\mathsf{M}) * u$ and $\mathcal{T}_{u,\mathsf{M}}v = |\det(\mathsf{M})|(v * u^\star)\downarrow\mathsf{M}$, where $v * u := \sum_{k\in\mathbb{Z}^d} v(k)u(\cdot - k)$ is the convolution of v and u.

Let $a, b_1, \dots, b_s \in l_1(\mathbb{Z}^d)$ and let $\mathsf{M}, \mathsf{M}_1, \dots, \mathsf{M}_s$ be $d \times d$ invertible integer matrices. For $J \in \mathbb{N}$, we now describe a J-level (d-dimensional) discrete/fast framelet transform employing a filter bank $\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\}$. For a given data $v_0 \in l_\infty(\mathbb{Z}^d)$, *the J-level discrete framelet decomposition* (or forward transform) employing the filter bank $\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\}$ is

$$v_j := |\det(\mathsf{M})|^{-1/2}\mathcal{T}_{a,\mathsf{M}}v_{j-1} \quad \text{and} \quad w_{\ell,j} := |\det(\mathsf{M}_\ell)|^{-1/2}\mathcal{T}_{b_\ell,\mathsf{M}_\ell}v_{j-1}, \qquad \ell = 1, \dots, s,\ j = 1, \dots, J, \tag{2.2}$$

where v_j are called sequences of low-pass coefficients and all $w_{\ell,j}$ are called sequences of high-pass coefficients of the input signal v_0. *The J-level discrete framelet reconstruction* (or backward transform) employing the filter bank $\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\}$ can be described by

$$\mathring{v}_{j-1} := |\det(\mathsf{M})|^{-1/2}\mathcal{S}_{a,\mathsf{M}}\mathring{v}_j + \sum_{\ell=1}^{s} |\det(\mathsf{M}_\ell)|^{-1/2}\mathcal{S}_{b_\ell,\mathsf{M}_\ell}\mathring{w}_{\ell,j}, \qquad j = J, \dots, 1, \tag{2.3}$$

where $\mathring{v}_0$ is a reconstructed sequence on $\mathbb{Z}^d$. The *perfect reconstruction property* requires that the reconstructed sequence $\mathring{v}_0$ should be exactly the same as the original input data v_0 if $\mathring{v}_J = v_J$ and $\mathring{w}_{\ell,j} = w_{\ell,j}$ for $j = 1, \dots, J$ and $\ell = 1, \dots, s$. See Fig. 2 for an illustration of a two-level fast framelet transform using a one-dimensional tight framelet filter bank $\{a\,!\,2; b_1\,!\,4, \dots, b_s\,!\,4\}$.

Following [15, Theorem 2.1], we have the following result on the perfect reconstruction property of a filter bank $\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\}$.

Theorem 1. *Let $a, b_1, \dots, b_s \in l_1(\mathbb{Z}^d)$ and let $\mathsf{M}, \mathsf{M}_1, \dots, \mathsf{M}_s$ be $d \times d$ invertible integer matrices. Then the following statements are equivalent to each other:*

(i) *For every $J \in \mathbb{N}$, the J-level fast framelet transform employing the filter bank $\{a \,!\, \mathsf{M}; b_1 \,!\, \mathsf{M}_1, \dots, b_s \,!\, \mathsf{M}_s\}$ has perfect reconstruction property.*

(ii) *The one-level discrete framelet transform employing the filter bank $\{a \,!\, \mathsf{M}; b_1 \,!\, \mathsf{M}_1, \dots, b_s \,!\, \mathsf{M}_s\}$ has* perfect reconstruction property, *that is, for all $v \in l_\infty(\mathbb{Z}^d)$,*

$$v = |\det(\mathsf{M})|^{-1} \mathcal{S}_{a,\mathsf{M}} \mathcal{T}_{a,\mathsf{M}} v + |\det(\mathsf{M}_1)|^{-1} \mathcal{S}_{b_1,\mathsf{M}_1} \mathcal{T}_{b_1,\mathsf{M}_1} v + \cdots + |\det(\mathsf{M}_s)|^{-1} \mathcal{S}_{b_s,\mathsf{M}_s} \mathcal{T}_{b_s,\mathsf{M}_s} v. \tag{2.4}$$

(iii) *(2.4) holds for all $v \in l_0(\mathbb{Z}^d)$.*

(iv) *The filter bank $\{a \,!\, \mathsf{M}; b_1 \,!\, \mathsf{M}_1, \dots, b_s \,!\, \mathsf{M}_s\}$ is* a tight framelet filter bank with mixed sampling factors, *that is, the following perfect reconstruction conditions hold:*

$$|\widehat{a}(\xi)|^2 + |\widehat{b_1}(\xi)|^2 + \cdots + |\widehat{b_s}(\xi)|^2 = 1, \qquad a.e.\, \xi \in \mathbb{R}^d \tag{2.5}$$

and

$$\chi_{\Omega_\mathsf{M}}(\omega) \widehat{a}(\xi) \overline{\widehat{a}(\xi + 2\pi\omega)} + \sum_{\ell=1}^{s} \chi_{\Omega_{\mathsf{M}_\ell}}(\omega) \widehat{b_\ell}(\xi) \overline{\widehat{b_\ell}(\xi + 2\pi\omega)} = 0, \tag{2.6}$$

for almost every $\xi \in \mathbb{R}^d$ and for all $\omega \in [\Omega_\mathsf{M} \cup (\cup_{\ell=1}^s \Omega_{\mathsf{M}_\ell})] \backslash \{0\}$, where $\Omega_\mathsf{M} := (\mathsf{M}^{-\mathsf{T}} \mathbb{Z}^d) \cap [0,1)^d$, $\Omega_{\mathsf{M}_\ell} := (\mathsf{M}_\ell^{-\mathsf{T}} \mathbb{Z}^d) \cap [0,1)^d$, and χ_E denotes the characteristic function of a set $E \subseteq \mathbb{R}^d$ such that $\chi_E(\omega) = 1$ if $\omega \in E$ and $\chi_E(\omega) = 0$ if $\omega \notin E$.

Proof. The equivalence between item (i) and item (ii) is obvious, since item (ii) is just item (i) with $J = 1$ and a J-level fast framelet transform recursively employs the one-level discrete framelet transform J times.

We now prove (ii) $\Longleftrightarrow$ (iii). Since $l_0(\mathbb{Z}^d) \subset l_\infty(\mathbb{Z}^d)$, (ii)$\Longrightarrow$(iii) is trivial. We use a similar argument as in the proof of [15, Theorem 2.1] to prove (iii)$\Longrightarrow$(ii). For $v \in l_\infty(\mathbb{Z}^d)$ and $N \in \mathbb{N}$, we consider the truncated sequence

$$v_N(k) := \begin{cases} v(k), & \text{if } \|k\| \leqslant N, \\ 0, & \text{otherwise}, \end{cases} \qquad k \in \mathbb{Z}^d.$$

Clearly, $v_N \in l_0(\mathbb{Z}^d)$ and $\lim_{N\to\infty} v_N(k) = v(k)$ for every $k \in \mathbb{Z}^d$. Since $a, b_1, \dots, b_s \in l_1(\mathbb{Z}^d)$, for every $k \in \mathbb{Z}^d$, it is not difficult to verify that

$$\begin{aligned} &\lim_{N\to\infty} [\mathcal{S}_{a,\mathsf{M}} \mathcal{T}_{a,\mathsf{M}} v_N](k) = [\mathcal{S}_{a,\mathsf{M}} \mathcal{T}_{a,\mathsf{M}} v](k) \quad \text{and} \\ &\lim_{N\to\infty} [\mathcal{S}_{b_\ell,\mathsf{M}_\ell} \mathcal{T}_{b_\ell,\mathsf{M}_\ell} v_N](k) = [\mathcal{S}_{b_\ell,\mathsf{M}_\ell} \mathcal{T}_{b_\ell,\mathsf{M}_\ell} v](k), \quad \ell = 1, \dots, s. \end{aligned}$$

By item (iii), (2.4) holds with v being replaced by v_N. That is, for every $k \in \mathbb{Z}^d$, we have

$$\begin{aligned} v_N(k) = {} & |\det(\mathsf{M})|^{-1} [\mathcal{S}_{a,\mathsf{M}} \mathcal{T}_{a,\mathsf{M}} v_N](k) + |\det(\mathsf{M}_1)|^{-1} [\mathcal{S}_{b_1,\mathsf{M}_1} \mathcal{T}_{b_1,\mathsf{M}_1} v_N](k) + \cdots \\ & + |\det(\mathsf{M}_s)|^{-1} [\mathcal{S}_{b_s,\mathsf{M}_s} \mathcal{T}_{b_s,\mathsf{M}_s} v_N](k). \end{aligned}$$

Taking $N \to \infty$ in the above identity, we conclude that the above identity still holds if v_N is replaced by v. Therefore, (2.4) holds for all $v \in l_\infty(\mathbb{Z}^d)$. This proves (iii)$\Longrightarrow$(ii).

Let $v \in l_1(\mathbb{Z}^d)$. By (2.1), we see that the Fourier series of the sequence $\mathcal{S}_{b_\ell,\mathsf{M}_\ell} \mathcal{T}_{b_\ell,\mathsf{M}_\ell} v$ is

$$|\det(\mathsf{M}_\ell)| \sum_{\omega_\ell \in \Omega_{\mathsf{M}_\ell}} \widehat{v}(\xi+2\pi\omega_\ell)\widehat{b_\ell}(\xi)\overline{\widehat{b_\ell}(\xi+2\pi\omega_\ell)}.$$

Consequently, we conclude that (2.4) holds for all $v \in l_1(\mathbb{Z}^d)$ if and only if

$$\begin{aligned}\widehat{v}(\xi) &= \sum_{\omega_0 \in \Omega_{\mathsf{M}}} \widehat{v}(\xi+2\pi\omega_0)\widehat{a}(\xi)\overline{\widehat{a}(\xi+2\pi\omega_0)} + \sum_{\ell=1}^{s}\sum_{\omega_\ell \in \Omega_{\mathsf{M}_\ell}} \widehat{v}(\xi+2\pi\omega_\ell)\widehat{b_\ell}(\xi)\overline{\widehat{b_\ell}(\xi+2\pi\omega_\ell)} \\ &= \sum_{\omega\in\Omega} \widehat{v}(\xi+2\pi\omega)\left(\chi_{\Omega_{\mathsf{M}}}(\omega)\widehat{a}(\xi)\overline{\widehat{a}(\xi+2\pi\omega)} + \sum_{\ell=1}^{s}\chi_{\Omega_{\mathsf{M}_\ell}}(\omega)\widehat{b_\ell}(\xi)\overline{\widehat{b_\ell}(\xi+2\pi\omega)}\right),\end{aligned}$$

where $\Omega := \Omega_{\mathsf{M}} \cup (\cup_{\ell=1}^{s}\Omega_{\mathsf{M}_\ell})$, which can be equivalently expressed as

$$\sum_{\omega\in\Omega} \widehat{v}(\xi+2\pi\omega)\widehat{u_\omega}(\xi) = 0, \tag{2.7}$$

where $\widehat{u_\omega}, \omega \in \Omega$ are $2\pi\mathbb{Z}^d$-periodic functions defined by $\widehat{u_0}(\xi) := |\widehat{a}(\xi)|^2 + |\widehat{b_1}(\xi)|^2 + \cdots + |\widehat{b_s}(\xi)|^2 - 1$ and

$$\widehat{u_\omega}(\xi) := \chi_{\Omega_{\mathsf{M}}}(\omega)\widehat{a}(\xi)\overline{\widehat{a}(\xi+2\pi\omega)} + \sum_{\ell=1}^{s}\chi_{\Omega_{\mathsf{M}_\ell}}(\omega)\widehat{b_\ell}(\xi)\overline{\widehat{b_\ell}(\xi+2\pi\omega)}, \qquad \omega \in \Omega\backslash\{0\}.$$

If item (iv) holds, then $\widehat{u_\omega} = 0$ for all $\omega \in \Omega$ and thus, (2.7) trivially holds for all $v \in l_0(\mathbb{Z}^d)$. This proves (iv)$\Longrightarrow$(iii). We now prove (iii)$\Longrightarrow$(iv). Since we proved (iii)$\Longrightarrow$(ii) and $l_1(\mathbb{Z}^d) \subset l_\infty(\mathbb{Z}^d)$, it follows from item (iii) that (2.7) holds for all $v \in l_1(\mathbb{Z}^d)$. We observe that the shortest distance, denoted by ε, from the point 0 to any point in the set $(\Omega\backslash\{0\}) + \mathbb{Z}^d$ must be positive. Let $\xi_0 \in (-\pi,\pi)^d$ be arbitrarily fixed. Then we can take $\varepsilon > 0$ further smaller so that $(\xi_0 + [-\varepsilon,\varepsilon]^d) \subseteq (-\pi,\pi)^d$. Let $\widehat{v}$ be any $2\pi\mathbb{Z}^d$-periodic C^∞ function such that the support of $\widehat{v}$ inside the fundamental domain $(-\pi,\pi]^d$ is contained inside $E_\varepsilon := \xi_0 + (-\varepsilon/\pi, \varepsilon/\pi)^d$. Then $v \in l_1(\mathbb{Z}^d)$ and $\widehat{v}(\xi+2\pi\omega) = 0$ for all $\xi \in E_\varepsilon$ and $\omega \in \Omega\backslash\{0\}$. Consequently, (2.7) implies $\widehat{v}(\xi)\widehat{u_0}(\xi) = 0$ for all $\xi \in E_\varepsilon$. In particular, $\widehat{u_0}$ must vanish in a neighborhood of ξ_0. Since $\xi_0 \in (-\pi,\pi)^d$ is arbitrarily chosen, we conclude that $\widehat{u_0}(\xi) = 0$ a.e. $\xi \in [-\pi,\pi)^d$. Since $\widehat{u_0}$ is $2\pi\mathbb{Z}^d$-periodic, this proves (2.5). For a general $\mathring{\omega} \in \Omega$, (2.7) is equivalent to $\sum_{\omega\in\Omega}\widehat{v}(\xi+2\pi(\omega-\mathring{\omega}))\widehat{u_\omega}(\xi-2\pi\mathring{\omega}) = 0$. By the same argument, we must have $\widehat{u_{\mathring{\omega}}} = 0$. This proves (2.6). $\square$

For the special case that $\mathsf{M}_1 = \cdots = \mathsf{M}_s = \mathsf{M}$ and all $a, b_1, \ldots, b_s \in l_0(\mathbb{Z}^d)$ are finitely supported, Theorem 1 reduces to [15, Theorem 2.1]. Though here we largely followed the idea of the proof of [15, Theorem 2.1], the key step (iii) $\Longrightarrow$ (iv) is proved in [15, Theorem 2.1] in the spatial domain, while here we proved the claim in the frequency domain. This argument allows us to deal with filters $a, b_1, \ldots, b_s \in l_1(\mathbb{Z}^d)$ instead of finitely supported filters in [15, Theorem 2.1]. The condition in (2.6) for a tight framelet filter bank $\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \ldots, b_s\,!\,\mathsf{M}_s\}$ is also much more complicated than its counterpart in [15, Theorem 2.1] with $\mathsf{M}_1 = \cdots = \mathsf{M}_s = \mathsf{M}$. Note that if Ω_{M_ℓ} is chosen to be another complete set of representatives of distinct cosets in $[\mathsf{M}_\ell^{-\mathsf{T}}\mathbb{Z}^d]/\mathbb{Z}^d$ instead of the particular choice $[\mathsf{M}_\ell^{-\mathsf{T}}\mathbb{Z}^d] \cap [0,1)^d$, then for all $\ell = 1, \ldots, s$, $\chi_{\Omega_{\mathsf{M}_\ell}}$ in (2.6) have to be replaced by $\chi_{\mathsf{M}_\ell^{-\mathsf{T}}\mathbb{Z}^d}$.

2.2. *Discrete affine systems of tight framelet filter banks with mixed sampling factors*

To understand the performance and properties of the J-level fast framelet transform using a tight framelet filter bank $\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \ldots, b_s\,!\,\mathsf{M}_s\}$, as pointed out in [15], it is very important to look at the J-level discrete affine systems associated with $\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \ldots, b_s\,!\,\mathsf{M}_s\}$.

We now generalize the discrete affine systems in [15, Section 4.3] to a d-dimensional tight framelet filter bank $\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\}$ with mixed sampling factors. Let $a, b_1, \dots, b_s \in l_1(\mathbb{Z}^d)$. Note that $l_1(\mathbb{Z}^d) \subseteq l_2(\mathbb{Z}^d)$ and $l_2(\mathbb{Z}^d)$ is a Hilbert space equipped with the inner product $\langle u, v\rangle := \sum_{k\in\mathbb{Z}^d} u(k)\overline{v(k)}$ for $u, v \in l_2(\mathbb{Z}^d)$. Following [15], we define the multilevel filters a_j and $b_{\ell,j}$ with $j \in \mathbb{N}$ and $\ell = 1, \dots, s$ by

$$\widehat{a_j}(\xi) := \widehat{a}(\xi)\widehat{a}(\mathsf{M}^\mathsf{T}\xi)\cdots\widehat{a}((\mathsf{M}^\mathsf{T})^{j-2}\xi)\widehat{a}((\mathsf{M}^\mathsf{T})^{j-1}\xi) \tag{2.8}$$

and

$$\widehat{b_{\ell,j}}(\xi) := \widehat{a_{j-1}}(\xi)\widehat{b_\ell}((\mathsf{M}^\mathsf{T})^{j-1}\xi) = \widehat{a}(\xi)\widehat{a}(\mathsf{M}^\mathsf{T}\xi)\cdots\widehat{a}((\mathsf{M}^\mathsf{T})^{j-2}\xi)\widehat{b_\ell}((\mathsf{M}^\mathsf{T})^{j-1}\xi). \tag{2.9}$$

In particular, $a_1 = a$ and $b_{\ell,1} = b_\ell$. We shall also use the convention $a_0 = \boldsymbol{\delta}$, where $\boldsymbol{\delta}$ is the Dirac/Kronecker sequence on $\mathbb{Z}^d$ given by

$$\boldsymbol{\delta}(0) = 1 \quad \text{and} \quad \boldsymbol{\delta}(k) = 0, \qquad \forall\, k \in \mathbb{Z}^d\backslash\{0\}.$$

Since $a, b_1, \dots, b_s \in l_1(\mathbb{Z}^d)$, it is straightforward to see that all $a_j, b_{\ell,j}$ are well-defined filters in $l_1(\mathbb{Z}^d) \subseteq l_2(\mathbb{Z}^d)$. For $j \in \mathbb{N}$ and $k \in \mathbb{Z}^d$, we define

$$a_{j;k} := |\det(\mathsf{M})|^{j/2} a_j(\cdot - \mathsf{M}^j k), \qquad b_{\ell,j;k} := |\det(\mathsf{M})|^{(j-1)/2}|\det(\mathsf{M}_\ell)|^{1/2} b_{\ell,j}(\cdot - \mathsf{M}^{j-1}\mathsf{M}_\ell k). \tag{2.10}$$

The J*-level discrete affine system* associated with the filter bank $\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\}$ is defined by

$$\begin{aligned} &\mathrm{DAS}_J(\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\}) \\ &\quad := \{a_{J;k} \ : \ k \in \mathbb{Z}^d\} \cup \{b_{\ell,j;k} \ : \ k \in \mathbb{Z}^d, \ell = 1, \dots, s, j = 1, \dots, J\}. \end{aligned} \tag{2.11}$$

By a similar argument as in [15, Section 4.3] (also see Theorem 2 below), under the framework of the Hilbert space $l_2(\mathbb{Z}^d)$, we see that the J-level fast framelet transform using the tight framelet filter bank $\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\}$ is exactly to compute the following representation:

$$v = \sum_{u\in\mathrm{DAS}_J(\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\})} \langle v, u\rangle u = \sum_{k\in\mathbb{Z}^d} \langle v, a_{J;k}\rangle a_{J;k} + \sum_{j=1}^{J}\sum_{\ell=1}^{s}\sum_{k\in\mathbb{Z}^d} \langle v, b_{\ell,j;k}\rangle b_{\ell,j;k}, \qquad \forall\, v \in l_2(\mathbb{Z}^d), \tag{2.12}$$

where the series converges unconditionally in $l_2(\mathbb{Z}^d)$. More precisely, as we shall see later, $v_J(k) = \langle v_0, a_{J;k}\rangle$ and $w_{\ell,j}(k) = \langle v_0, b_{\ell,j;k}\rangle$ for all $j = 1, \dots, J$ and $k \in \mathbb{Z}^d$.

Following the general theory developed in [15], we have the following result.

Theorem 2. *Let $a, b_1, \dots, b_s \in l_1(\mathbb{Z}^d)$ and $\mathsf{M}, \mathsf{M}_1, \dots, \mathsf{M}_s$ be $d\times d$ invertible integer matrices. For $J \in \mathbb{N}$, define $\mathrm{DAS}_J(\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\})$ as in (2.11) with a_j and $b_{\ell,j}$ being given in (2.8) and (2.9), respectively. Then the following statements are equivalent:*

(1) *$\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\}$ is a tight framelet filter bank with mixed sampling factors.*
(2) *The following identity holds:*

$$v = \sum_{k\in\mathbb{Z}^d} \langle v, a_{1;k}\rangle a_{1;k} + \sum_{\ell=1}^{s}\sum_{k\in\mathbb{Z}^d} \langle v, b_{\ell,1;k}\rangle b_{\ell,1;k}, \qquad \forall\, v \in l_2(\mathbb{Z}^d). \tag{2.13}$$

(3) $\mathrm{DAS}_1(\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\})$ *is a (normalized) tight frame for* $l_2(\mathbb{Z}^d)$*, that is,*

$$\|v\|^2_{l_2(\mathbb{Z}^d)} = \sum_{k\in\mathbb{Z}^d} |\langle v, a_{1;k}\rangle|^2 + \sum_{\ell=1}^{s}\sum_{k\in\mathbb{Z}^d} |\langle v, b_{\ell,1;k}\rangle|^2, \qquad \forall\, v \in l_2(\mathbb{Z}^d). \tag{2.14}$$

(4) *For every* $j \in \mathbb{N}$*, the following identity holds:*

$$\sum_{k\in\mathbb{Z}^d} \langle v, a_{j-1;k}\rangle a_{j-1;k} = \sum_{k\in\mathbb{Z}^d} \langle v, a_{j;k}\rangle a_{j;k} + \sum_{\ell=1}^{s}\sum_{k\in\mathbb{Z}^d} \langle v, b_{\ell,j;k}\rangle b_{\ell,j;k}, \qquad \forall\, v \in l_2(\mathbb{Z}^d), \tag{2.15}$$

where by convention $a_0 := \boldsymbol{\delta}$ *and* $a_{0;k} := \boldsymbol{\delta}(\cdot - k)$ *for* $k \in \mathbb{Z}^d$*.*

(5) *For every* $J \in \mathbb{N}$*, the identity in* (2.12) *holds.*

(6) *For every* $J \in \mathbb{N}$*,* $\mathrm{DAS}_J(\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\})$ *is a (normalized) tight frame for* $l_2(\mathbb{Z}^d)$*, that is,*

$$\|v\|^2_{l_2(\mathbb{Z}^d)} = \sum_{k\in\mathbb{Z}^d} |\langle v, a_{J;k}\rangle|^2 + \sum_{j=1}^{J}\sum_{\ell=1}^{s}\sum_{k\in\mathbb{Z}^d} |\langle v, b_{\ell,j;k}\rangle|^2, \qquad \forall\, v \in l_2(\mathbb{Z}^d). \tag{2.16}$$

Proof. Plugging $v = \boldsymbol{\delta}(\cdot - n)$ with all $n \in \mathbb{Z}^d$ into (2.13), we observe by direct calculation that the resulting equations in (2.13) with $v = \boldsymbol{\delta}(\cdot - n)$ are simply the spatial domain version of the conditions in (2.5) and (2.6) in the frequency domain. Hence, (1) $\Longleftrightarrow$ (2).

Note that (2.13) holds if and only if

$$\langle v, w\rangle = \sum_{k\in\mathbb{Z}^d} \langle v, a_{1;k}\rangle\langle a_{1;k}, w\rangle + \sum_{\ell=1}^{s}\sum_{k\in\mathbb{Z}^d} \langle v, b_{\ell,1;k}\rangle\langle b_{\ell,1;k}, w\rangle, \qquad \forall\, v, w \in l_2(\mathbb{Z}^d). \tag{2.17}$$

(2)$\Longrightarrow$(3) is trivial by plugging $w = v$ into (2.17). For any Hilbert space $\mathcal{H}$ over the complex field $\mathbb{C}$ with $\|h\|^2_{\mathcal{H}} := \langle h, h\rangle$, the following well-known polarization identity holds:

$$\langle v, w\rangle = \frac{1}{4}\left(\|v+w\|^2_{\mathcal{H}} - \|v-w\|^2_{\mathcal{H}} + i\|v+iw\|^2_{\mathcal{H}} - i\|v-iw\|^2_{\mathcal{H}}\right), \qquad \forall\, v, w \in \mathcal{H}.$$

Applying the above polarization identity with $\mathcal{H} = l_2(\mathbb{Z}^d)$ and $\mathcal{H} = \mathbb{C}$, we deduce directly from (2.14) in item (3) that (2.17) holds. Thus, (3)$\Longrightarrow$(2) and we proved (2) $\Longleftrightarrow$ (3).

(4)$\Longrightarrow$(2) is obvious since it follows from the convention $a_0 = \boldsymbol{\delta}$ that $\sum_{k\in\mathbb{Z}^d}\langle v, a_{0;k}\rangle a_{0;k} = \sum_{k\in\mathbb{Z}^d} v(k)\boldsymbol{\delta}(\cdot - k) = v$. We now prove (2)$\Longrightarrow$(4). By the definition of $b_{\ell,j}$ in (2.9) and $b_{\ell,1} = b_\ell$,

$$\begin{aligned} b_{\ell,j} &= a_{j-1} * (b_\ell \uparrow \mathsf{M}^{j-1}) = a_{j-1} * (b_{\ell,1} \uparrow \mathsf{M}^{j-1}) \\ &= \sum_{n\in\mathbb{Z}^d} a_{j-1}(\cdot - n)(b_{\ell,1}\uparrow \mathsf{M}^{j-1})(n) = \sum_{m\in\mathbb{Z}^d} a_{j-1}(\cdot - \mathsf{M}^{j-1}m) b_{\ell,1}(m). \end{aligned}$$

Therefore, by the definition of $b_{\ell,j;k}$ in (2.10),

$$\begin{aligned} b_{\ell,j;k} &= |\det(\mathsf{M})|^{(j-1)/2}|\det(\mathsf{M}_\ell)|^{1/2} b_{\ell,j}(\cdot - \mathsf{M}^{j-1}\mathsf{M}_\ell k) \\ &= |\det(\mathsf{M})|^{(j-1)/2}|\det(\mathsf{M}_\ell)|^{1/2} \sum_{m\in\mathbb{Z}^d} a_{j-1}(\cdot - \mathsf{M}^{j-1}\mathsf{M}_\ell k - \mathsf{M}^{j-1}m) b_{\ell,1}(m) \\ &= |\det(\mathsf{M})|^{(j-1)/2}|\det(\mathsf{M}_\ell)|^{1/2} \sum_{m\in\mathbb{Z}^d} a_{j-1}(\cdot - \mathsf{M}^{j-1}m) b_{\ell,1}(m - \mathsf{M}_\ell k) \\ &= \sum_{m\in\mathbb{Z}^d} a_{j-1;m} b_{\ell,1;k}(m). \end{aligned}$$

Consequently, we proved

$$\langle v, b_{\ell,j;k}\rangle = \sum_{m\in\mathbb{Z}^d} \langle v, a_{j-1;m}\rangle \overline{b_{\ell,1;k}(m)} = \langle\langle v, a_{j-1;\cdot}\rangle, b_{\ell,1;k}(\cdot)\rangle. \tag{2.18}$$

We now deduce from the above two identities that

$$\sum_{k\in\mathbb{Z}^d} \langle v, b_{\ell,j;k}\rangle b_{\ell,j;k} = \sum_{m\in\mathbb{Z}^d} a_{j-1;m} \left(\sum_{k\in\mathbb{Z}^d} \langle\langle v, a_{j-1;\cdot}\rangle, b_{\ell,1;k}\rangle b_{\ell,1;k}(m)\right).$$

The same argument can be applied to $a_{j;k}$ and the above identity still holds by replacing $b_{\ell,j;k}$ and $b_{\ell,1;k}$ with $a_{j;k}$ and $a_{1;k}$, respectively. Therefore,

$$\begin{aligned}
&\sum_{k\in\mathbb{Z}^d} \langle v, a_{j;k}\rangle a_{j;k} + \sum_{\ell=1}^{s}\sum_{k\in\mathbb{Z}^d} \langle v, b_{\ell,j;k}\rangle b_{\ell,j;k} \\
&\quad = \sum_{m\in\mathbb{Z}^d} a_{j-1;m}\left(\sum_{k\in\mathbb{Z}^d} \langle\langle v, a_{j-1;\cdot}\rangle, a_{1;k}\rangle a_{1;k}(m) + \sum_{\ell=1}^{s}\sum_{k\in\mathbb{Z}^d} \langle\langle v, a_{j-1;\cdot}\rangle, b_{\ell,1;k}\rangle b_{\ell,1;k}(m)\right) \\
&\quad = \sum_{m\in\mathbb{Z}^d} \langle v, a_{j-1;m}\rangle a_{j-1;m},
\end{aligned}$$

where we used (2.13), i.e., item (2), in the last identity. This proves (2)$\Longrightarrow$(4).

(4)$\Longrightarrow$(5) is obvious by summing all (2.15) with $j = 1, \dots, J$ together. Conversely, considering the differences between $J = j$ and $J = j-1$ in (2.12), we see that (5)$\Longrightarrow$(4). The equivalence between item (5) and item (6) is straightforward and is similar to the equivalence between item (2) and item (3). □

Theorem 2 with $\mathsf{M}_1 = \cdots = \mathsf{M}_s = \mathsf{M}$ and $a, b_1, \dots, b_s \in l_0(\mathbb{Z}^d)$ has been discussed in [15, Section 4.3] but without explicitly stating it as a theorem in [15]. Though we used the ideas from [15] to prove Theorem 2, the key step (2)$\Longrightarrow$(4) here is new.

We now show that the coefficients in the representation in (2.12) using a J-level discrete affine system can be exactly computed through the J-level fast framelet decomposition in (2.2). Since $\mathcal{T}_{u,\mathsf{M}}v = |\det(\mathsf{M})|(v * u^\star)\downarrow\mathsf{M}$ and $\widehat{a_{j-1}}(\xi) = \widehat{a}(\xi)\cdots\widehat{a}((\mathsf{M}^\mathsf{T})^{j-2}\xi)$, by [15, Lemma 4.3], we have

$$\begin{aligned}
\langle v, a_{j-1;k}\rangle &= |\det(\mathsf{M})|^{(j-1)/2}\langle v, a_{j-1}(\cdot - \mathsf{M}^{j-1}k)\rangle = |\det(\mathsf{M})|^{(1-j)/2}[\mathcal{T}_{a_{j-1},\mathsf{M}^{j-1}}v](k) \\
&= |\det(\mathsf{M})|^{(1-j)/2}[\mathcal{T}^{j-1}_{a,\mathsf{M}}v](k) = v_{j-1}(k),
\end{aligned}$$

where v_{j-1} is exactly the same sequence as obtained in the fast framelet decomposition in (2.2) with $v_0 := v$. Similarly, by (2.18) and the above identity, we have

$$\begin{aligned}
\langle v, b_{\ell,j;k}\rangle &= \langle\langle v, a_{j-1;\cdot}\rangle, b_{\ell,1;k}\rangle = |\det(\mathsf{M}_\ell)|^{1/2}\langle v_{j-1}, b_\ell(\cdot - \mathsf{M}_\ell k)\rangle \\
&= |\det(\mathsf{M}_\ell)|^{1/2}\sum_{m\in\mathbb{Z}^d} v_{j-1}(m)\overline{b_\ell(m - \mathsf{M}_\ell k)} = |\det(\mathsf{M}_\ell)|^{-1/2}[\mathcal{T}_{b_\ell,\mathsf{M}_\ell}v_{j-1}](k) = w_{\ell,j}(k).
\end{aligned}$$

This establishes the connection between the representation in (2.12) under the J-level discrete affine system and the J-level fast/discrete framelet transform in (2.2) and (2.3).

2.3. Connections to tight framelets in $L_2(\mathbb{R}^d)$

Following the general theory on frequency-based framelets in [13,14], we now discuss the natural connections of a tight framelet filter bank $\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\}$ with a tight framelet in $L_2(\mathbb{R}^d)$.

For a function $f : \mathbb{R}^d \to \mathbb{C}$ and a $d \times d$ real-valued matrix U, following [14], we shall adopt the following notation:

$$f_{U;k,n}(x) = f_{[\![U;k,n]\!]}(x) = [\![U; k, n]\!] f(x) := |\det(U)|^{1/2} e^{-in\cdot Ux} f(Ux - k), \qquad x, k, n \in \mathbb{R}^d.$$

In particular, we define $f_{U;k} := f_{U;k,0} = |\det U|^{1/2} f(U \cdot -k)$. For $f \in L_1(\mathbb{R}^d)$, its Fourier transform is defined to be $\widehat{f}(\xi) := \int_{\mathbb{R}^d} f(x)e^{-ix\cdot\xi}dx$ for $\xi \in \mathbb{R}^d$. If U is invertible, then $\widehat{f_{U;k}} = \widehat{f}_{U^{-\mathsf{T}};0,k}$.

The following result is based on the general theory developed in [13,14] on frequency-based framelets.

Theorem 3. *Let $a, b_1, \dots, b_s \in l_1(\mathbb{Z}^d)$ and $\mathsf{M}, \mathsf{M}_1, \dots, \mathsf{M}_s$ be $d \times d$ invertible integer matrices. Suppose that all the eigenvalues of M are greater than one in modulus and there exist positive numbers C and τ such that*

$$|1 - \widehat{a}(\xi)| \leqslant C\|\xi\|^\tau \qquad \textit{for all}\, \xi \in [-\pi, \pi]^d. \tag{2.19}$$

Define

$$\widehat{\phi}(\xi) := \prod_{j=1}^{\infty} \widehat{a}((\mathsf{M}^\mathsf{T})^{-j}\xi) \quad \textit{and} \quad \widehat{\psi^\ell}(\xi) := \widehat{b_\ell}(\mathsf{M}^{-\mathsf{T}}\xi)\widehat{\phi}(\mathsf{M}^{-\mathsf{T}}\xi), \qquad \xi \in \mathbb{R}^d, \ell = 1, \dots, s. \tag{2.20}$$

If $\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\}$ is a tight framelet filter bank, then $\{\phi\,!\,\mathsf{M}; \psi^1\,!\,\mathsf{M}_1, \dots, \psi^s\,!\,\mathsf{M}_s\}$ is a tight framelet in $L_2(\mathbb{R}^d)$, that is, $\phi, \psi^1, \dots, \psi^s \in L_2(\mathbb{R}^d)$ and $\mathsf{AS}_0(\{\phi\,!\,\mathsf{M}; \psi^1\,!\,\mathsf{M}_1, \dots, \psi^s\,!\,\mathsf{M}_s\})$ is a (normalized) tight frame for $L_2(\mathbb{R}^d)$:

$$\|f\|^2_{L_2(\mathbb{R}^d)} = \sum_{k\in\mathbb{Z}^d} |\langle f, \phi(\cdot - k)\rangle|^2 + \sum_{j=0}^{\infty}\sum_{\ell=1}^{s}\sum_{k\in\mathbb{Z}^d} |\langle f, |\det(\mathsf{M}^{-1}\mathsf{M}_\ell)|^{1/2}\psi^\ell_{\mathsf{M}^j;\mathsf{M}^{-1}\mathsf{M}_\ell k}\rangle|^2, \tag{2.21}$$

for all $f \in L_2(\mathbb{R}^d)$, where

$$\begin{aligned}&\mathsf{AS}_0(\{\phi\,!\,\mathsf{M}; \psi^1\,!\,\mathsf{M}_1, \dots, \psi^s\,!\,\mathsf{M}_s\})\\ &\quad := \{\phi(\cdot - k) \ : \ k \in \mathbb{Z}^d\} \cup \{|\det(\mathsf{M}^{-1}\mathsf{M}_\ell)|^{1/2}\psi^\ell_{\mathsf{M}^j;\mathsf{M}^{-1}\mathsf{M}_\ell k} \ : \ k \in \mathbb{Z}^d, \ell = 1, \dots, s, j \in \mathbb{N} \cup \{0\}\}.\end{aligned} \tag{2.22}$$

The converse direction also holds provided in addition that $\sum_{k\in\mathbb{Z}^d} |\widehat{\phi}(\xi+2\pi k)|^2 \neq 0$ for almost every $\xi \in \mathbb{R}^d$.

Proof. Though the proof here essentially follows the arguments and general theory developed in [13,14], for the convenience of the reader and due to the importance of this result, we provide a self-contained proof here.

Define $\mathsf{N} := \mathsf{M}^{-\mathsf{T}}$ and $\mathsf{N}_\ell := \mathsf{M}_\ell^{-\mathsf{T}}$. By our assumption on M and $\widehat{a}$, we see that $\widehat{\phi}$ is a well-defined bounded function in $L_\infty(\mathbb{R}^d)$ and $\lim_{j\to+\infty} \widehat{\phi}(\mathsf{N}^j\xi) = 1$ for all $\xi \in \mathbb{R}^d$. Define

$$D := \{f \in L_2(\mathbb{R}^d) \ : \ \widehat{f} \text{ has compact support and } \widehat{f} \in C^\infty(\mathbb{R}^d)\}.$$

By [14, Lemma 10], for $f \in D$, we have

$$\sum_{k\in\mathbb{Z}^d} |\langle \widehat{f}, \widehat{\phi}_{\mathsf{N}^j;0,k}\rangle|^2 = (2\pi)^d \int_{\mathbb{R}^d} \sum_{k\in\mathbb{Z}^d} \widehat{f}(\xi)\overline{\widehat{f}(\xi + 2\pi\mathsf{N}^{-j}k)}\,\overline{\widehat{\phi}(\mathsf{N}^j\xi)}\widehat{\phi}(\mathsf{N}^j\xi + 2\pi k)d\xi. \tag{2.23}$$

Since $\widehat{f}$ has compact support and all the eigenvalues of N^{-1} are greater than 1 in modulus, for sufficiently large j, $\widehat{f}(\xi)\overline{\widehat{f}(\xi+2\pi\mathsf{N}^{-j}k)}=0$ for all $\xi\in\mathbb{R}^d$ and $k\in\mathbb{Z}^d\backslash\{0\}$. Hence, for sufficiently large $j\in\mathbb{N}$, (2.23) becomes

$$\sum_{k\in\mathbb{Z}^d}|\langle\widehat{f},\widehat{\phi}_{\mathsf{N}^j;0,k}\rangle|^2=(2\pi)^d\int_{\mathbb{R}^d}|\widehat{f}(\xi)|^2|\widehat{\phi}(\mathsf{N}^j\xi)|^2d\xi.$$

Since $\widehat{\phi}$ is bounded, we have $|\widehat{f}(\xi)|^2|\widehat{\phi}(\mathsf{N}^j\xi)|^2\leqslant\|\widehat{\phi}\|^2_{L_\infty(\mathbb{R}^d)}|\widehat{f}(\xi)|^2$. By $f\in L_2(\mathbb{R}^d)$, we have $\widehat{f}\in L_2(\mathbb{R}^d)$ and thus, $|\widehat{f}|^2\in L_1(\mathbb{R}^d)$. By Lebesgue Dominated Convergence Theorem and $\lim_{j\to+\infty}\widehat{\phi}(\mathsf{N}^j\xi)=1$, we have

$$\lim_{j\to+\infty}\sum_{k\in\mathbb{Z}^d}|\langle\widehat{f},\widehat{\phi}_{\mathsf{N}^j;0,k}\rangle|^2=(2\pi)^d\int_{\mathbb{R}^d}|\widehat{f}(\xi)|^2\lim_{j\to+\infty}|\widehat{\phi}(\mathsf{N}^j\xi)|^2d\xi=(2\pi)^d\|\widehat{f}\|^2_{L_2(\mathbb{R}^d)},\qquad f\in D.\qquad(2.24)$$

Define $\eta^\ell:=\psi^\ell(\mathsf{M}^{-1}\mathsf{M}_\ell\cdot)$. Then $\psi^\ell(\cdot-\mathsf{M}^{-1}\mathsf{M}_\ell k)=\eta^\ell(\mathsf{M}_\ell^{-1}\mathsf{M}\cdot-k)$ and $\widehat{\eta^\ell}(\xi)=|\det(\mathsf{M}^{-1}\mathsf{M}_\ell)|^{-1}\widehat{\psi^\ell}(\mathsf{N}^{-1}\mathsf{N}_\ell\xi)$. By [14, Lemma 10], for $f\in D$, we have

$$\begin{aligned}
&|\det(\mathsf{M}^{-1}\mathsf{M}_\ell)|^2\sum_{k\in\mathbb{Z}^d}|\langle\widehat{f},\widehat{\eta^\ell}_{\mathsf{N}_\ell^{-1}\mathsf{N};0,k}\rangle|^2\\
&\quad=(2\pi)^d|\det(\mathsf{M}^{-1}\mathsf{M}_\ell)|^2\int_{\mathbb{R}^d}\widehat{f}(\xi)\overline{\widehat{f}(\xi+2\pi\mathsf{N}^{-1}\mathsf{N}_\ell k)}\;\overline{\widehat{\eta^\ell}(\mathsf{N}_\ell^{-1}\mathsf{N}\xi)}\widehat{\eta^\ell}(\mathsf{N}_\ell^{-1}\mathsf{N}\xi+2\pi k)d\xi\\
&\quad=(2\pi)^d\int_{\mathbb{R}^d}\sum_{k\in\mathbb{Z}^d}\widehat{f}(\xi)\overline{\widehat{f}(\xi+2\pi\mathsf{N}^{-1}\mathsf{N}_\ell k)}\overline{\widehat{\psi^\ell}(\xi)}\widehat{\psi^\ell}(\xi+2\pi\mathsf{N}^{-1}\mathsf{N}_\ell k)d\xi\\
&\quad=(2\pi)^d\int_{\mathbb{R}^d}\sum_{k\in\mathbb{Z}^d}\widehat{f}(\xi)\overline{\widehat{f}(\xi+2\pi\mathsf{N}^{-1}\mathsf{N}_\ell k)}\overline{\widehat{b}_\ell(\mathsf{N}\xi)}\widehat{b}_\ell(\mathsf{N}\xi+2\pi\mathsf{N}_\ell k)\overline{\widehat{\phi}(\mathsf{N}\xi)}\widehat{\phi}(\mathsf{N}\xi+2\pi\mathsf{N}_\ell k)d\xi\\
&\quad=(2\pi)^d\int_{\mathbb{R}^d}\widehat{f}(\xi)\overline{\widehat{\phi}(\mathsf{N}\xi)}\sum_{\omega_\ell\in\Omega_\ell}\overline{\widehat{b}_\ell(\mathsf{N}\xi)}\widehat{b}_\ell(\mathsf{N}\xi+2\pi\omega_\ell)\sum_{k\in\mathbb{Z}^d}\overline{\widehat{f}(\xi+2\pi\mathsf{N}^{-1}\omega_\ell+2\pi\mathsf{N}^{-1}k)}\widehat{\phi}(\mathsf{N}\xi+2\pi\omega_\ell+2\pi k)d\xi,
\end{aligned}$$

where we used (2.20) in the second-to-last identity and the fact that $\mathbb{Z}^d=\mathsf{M}_\ell^{\mathsf{T}}\Omega_\ell+\mathsf{M}_\ell^{\mathsf{T}}\mathbb{Z}^d$. Similarly, by (2.23) with $j=0$ and $\widehat{\phi}(\xi)=\widehat{a}(\mathsf{N}\xi)\widehat{\phi}(\mathsf{N}\xi)$ (which is a direct consequence of the first definition in (2.20)), we have

$$\begin{aligned}
&\sum_{k\in\mathbb{Z}^d}|\langle\widehat{f},\widehat{\phi}_{I_d;0,k}\rangle|^2\\
&\quad=(2\pi)^d\int_{\mathbb{R}^d}\sum_{k\in\mathbb{Z}^d}\widehat{f}(\xi)\overline{\widehat{f}(\xi+2\pi k)}\overline{\widehat{a}(\mathsf{N}\xi)}\widehat{a}(\mathsf{N}\xi+2\pi\mathsf{N}k)\overline{\widehat{\phi}(\mathsf{N}\xi)}\widehat{\phi}(\mathsf{N}\xi+2\pi\mathsf{N}k)d\xi\\
&\quad=(2\pi)^d\int_{\mathbb{R}^d}\widehat{f}(\xi)\overline{\widehat{\phi}(\mathsf{N}\xi)}\sum_{\omega_0\in\Omega_\mathsf{M}}\overline{\widehat{a}(\mathsf{N}\xi)}\widehat{a}(\mathsf{N}\xi+2\pi\omega_0)\sum_{k\in\mathbb{Z}^d}\overline{\widehat{f}(\xi+2\pi\mathsf{N}^{-1}\omega_0+2\pi\mathsf{N}^{-1}k)}\widehat{\phi}(\mathsf{N}\xi+2\pi\omega_0+2\pi k)d\xi.
\end{aligned}$$

Consequently,

$$\begin{aligned}
&\sum_{k\in\mathbb{Z}^d}|\langle\widehat{f},\widehat{\phi}_{I_d;0,k}\rangle|^2+\sum_{\ell=1}^{s}|\det(\mathsf{M}^{-1}\mathsf{M}_\ell)|^2\sum_{k\in\mathbb{Z}^d}|\langle\widehat{f},\widehat{\eta^\ell}_{\mathsf{N}_\ell^{-1}\mathsf{N};0,k}\rangle|^2\\
&\quad=(2\pi)^d\int_{\mathbb{R}^d}\widehat{f}(\xi)\overline{\widehat{\phi}(\mathsf{N}\xi)}\sum_{\omega\in\Omega}\sum_{k\in\mathbb{Z}^d}\overline{\widehat{f}(\xi+2\pi\mathsf{N}^{-1}\omega+2\pi\mathsf{N}^{-1}k)}\widehat{\phi}(\mathsf{N}\xi+2\pi\omega+2\pi k)
\end{aligned}$$

$$\times \left(\chi_{\Omega_{\mathsf{M}}}(\omega)\overline{\widehat{a}(\mathsf{N}\xi)}\widehat{a}(\mathsf{N}\xi+2\pi\omega) + \sum_{\ell=1}^{s} \chi_{\Omega_{\mathsf{M}_\ell}}(\omega)\overline{\widehat{b_\ell}(\mathsf{N}\xi)}\widehat{b_\ell}(\mathsf{N}\xi+2\pi\omega) \right) d\xi, \tag{2.25}$$

where $\Omega := \Omega_{\mathsf{M}} \cup (\cup_{\ell=1}^{s} \Omega_{\mathsf{M}_\ell})$. Suppose that $\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1, \dots, b_s\,!\,\mathsf{M}_s\}$ is a tight framelet filter bank. Then (2.5) and (2.6) are satisfied and consequently, we deduce from (2.25) that

$$\begin{aligned}
&\sum_{k\in\mathbb{Z}^d} |\langle \widehat{f}, \widehat{\phi}_{I_d;0,k}\rangle|^2 + \sum_{\ell=1}^{s} |\det(\mathsf{M}^{-1}\mathsf{M}_\ell)|^2 \sum_{k\in\mathbb{Z}^d} |\langle \widehat{f}, \widehat{\eta^\ell}_{\mathsf{N}_\ell^{-1}\mathsf{N};0,k}\rangle|^2 \\
&= (2\pi)^d \int_{\mathbb{R}^d} \widehat{f}(\xi)\overline{\widehat{\phi}(\mathsf{N}\xi)} \sum_{k\in\mathbb{Z}^d} \overline{\widehat{f}(\xi+2\pi\mathsf{N}^{-1}k)}\widehat{\phi}(\mathsf{N}\xi+2\pi k)d\xi.
\end{aligned}$$

Applying (2.23) with $j = 1$ to the right-hand side of the above identity, we conclude that

$$\sum_{k\in\mathbb{Z}^d} |\langle \widehat{f}, \widehat{\phi}_{I_d;0,k}\rangle|^2 + \sum_{\ell=1}^{s} |\det(\mathsf{M}^{-1}\mathsf{M}_\ell)|^2 \sum_{k\in\mathbb{Z}^d} |\langle \widehat{f}, \widehat{\eta^\ell}_{\mathsf{N}_\ell^{-1}\mathsf{N};0,k}\rangle|^2 = \sum_{k\in\mathbb{Z}^d} |\langle \widehat{f}, \widehat{\phi}_{\mathsf{N};0,k}\rangle|^2, \qquad f \in D. \tag{2.26}$$

By a simple scaling technique $\langle \widehat{f}_{\mathsf{N}^{-j};0}, \widehat{g}\rangle = \langle \widehat{f}, \widehat{g}_{\mathsf{N}^j;0}\rangle$, replacing $\widehat{f}$ in (2.26) by $\widehat{f}_{\mathsf{N}^{-j};0}$, we conclude that

$$\sum_{k\in\mathbb{Z}^d} |\langle \widehat{f}, \widehat{\phi}_{\mathsf{N}^j;0,k}\rangle|^2 + \sum_{\ell=1}^{s} |\det(\mathsf{M}^{-1}\mathsf{M}_\ell)|^2 \sum_{k\in\mathbb{Z}^d} |\langle \widehat{f}, \widehat{\eta^\ell}_{\mathsf{N}_\ell^{-1}\mathsf{N}^{j+1};0,k}\rangle|^2 = \sum_{k\in\mathbb{Z}^d} |\langle \widehat{f}, \widehat{\phi}_{\mathsf{N}^{j+1};0,k}\rangle|^2$$

for all $j \in \mathbb{Z}$ and $f \in D$. For any $m, n \in \mathbb{Z}$ with $m < n$, adding the above identities with $j = m, \dots, n-1$, we have

$$\sum_{k\in\mathbb{Z}^d} |\langle \widehat{f}, \widehat{\phi}_{\mathsf{N}^m;0,k}\rangle|^2 + \sum_{j=m}^{n-1}\sum_{\ell=1}^{s} |\det(\mathsf{M}^{-1}\mathsf{M}_\ell)|^2 \sum_{k\in\mathbb{Z}^d} |\langle \widehat{f}, \widehat{\eta^\ell}_{\mathsf{N}_\ell^{-1}\mathsf{N}^{j+1};0,k}\rangle|^2 = \sum_{k\in\mathbb{Z}^d} |\langle \widehat{f}, \widehat{\phi}_{\mathsf{N}^n;0,k}\rangle|^2. \tag{2.27}$$

Taking $n \to \infty$ in (2.27), we conclude from (2.24) and the above identity (2.27) that for every $m \in \mathbb{Z}$,

$$\sum_{k\in\mathbb{Z}^d} |\langle \widehat{f}, \widehat{\phi}_{\mathsf{N}^m;0,k}\rangle|^2 + \sum_{j=m}^{\infty}\sum_{\ell=1}^{s} |\det(\mathsf{M}^{-1}\mathsf{M}_\ell)|^2 \sum_{k\in\mathbb{Z}^d} |\langle \widehat{f}, \widehat{\eta^\ell}_{\mathsf{N}_\ell^{-1}\mathsf{N}^{j+1};0,k}\rangle|^2 = (2\pi)^d \|\widehat{f}\|^2_{L_2(\mathbb{R}^d)} \tag{2.28}$$

for all $f \in D$. In particular, the above identity (2.28) implies $|\langle \widehat{f}, \widehat{\phi}_{\mathsf{N}^m;0,0}\rangle|^2 \leqslant (2\pi)^d \|\widehat{f}\|^2_{L_2(\mathbb{R}^d)}$ for all $f \in D$. Since D is dense in $L_2(\mathbb{R}^d)$, this inequality implies $\widehat{\phi} \in L_2(\mathbb{R}^d)$. By the same argument, we have $\widehat{\eta^1}, \dots, \widehat{\eta^s} \in L_2(\mathbb{R}^d)$. This proves that $\phi, \eta^1, \dots, \eta^s$ are well-defined functions in $L_2(\mathbb{R}^d)$. Consequently, we proved $\phi, \psi^1, \dots, \psi^s \in L_2(\mathbb{R}^d)$. Since D is dense in $L_2(\mathbb{R}^d)$, we also see that (2.28) holds for all $f \in L_2(\mathbb{R}^d)$ and $m \in \mathbb{Z}$. By $\eta^\ell = \psi^\ell(\mathsf{M}^{-1}\mathsf{M}_\ell \cdot)$, the Fourier transform of $\psi^\ell_{\mathsf{M}^j;\mathsf{M}^{-1}\mathsf{M}_\ell k}$ is $\widehat{\psi^\ell}_{\mathsf{N}^j;0,\mathsf{M}^{-1}\mathsf{M}_\ell k}$ which is further equal to

$$\begin{aligned}
\widehat{\psi^\ell}_{\mathsf{N}^j;0,\mathsf{M}^{-1}\mathsf{M}_\ell k} &= |\det(\mathsf{N})|^{j/2} e^{-i(\mathsf{M}^{-1}\mathsf{M}_\ell k)\cdot \mathsf{N}^j\xi}\widehat{\psi^\ell}(\mathsf{N}^j\xi) \\
&= |\det(\mathsf{N})|^{j/2} |\det(\mathsf{M}^{-1}\mathsf{M}_\ell)| e^{-ik\cdot\mathsf{N}_\ell^{-1}\mathsf{N}^{j+1}\xi}\widehat{\eta^\ell}(\mathsf{N}_\ell^{-1}\mathsf{N}^{j+1}\xi) = |\det(\mathsf{M}^{-1}\mathsf{M}_\ell)|^{1/2}\widehat{\eta^\ell}_{\mathsf{N}_\ell^{-1}\mathsf{N}^{j+1};0,k}.
\end{aligned}$$

By Plancherel's Theorem, we deduce from (2.28) with $m = 0$ that (2.21) must hold for all $f \in L_2(\mathbb{R}^d)$.

Conversely, if (2.21) holds, then (2.28) must hold with $m = 0$. By a simple scaling technique, (2.28) must hold for all $m \in \mathbb{Z}$. Considering the difference between $m = 0$ and $m = 1$ in (2.28), we see that (2.26) must hold for all $f \in L_2(\mathbb{R}^d)$. By (2.25) and (2.23) with $j = 1$, we see that (2.26) is equivalent to

$$\int_{\mathbb{R}^d} \widehat{f}(\xi)\overline{\widehat{\phi}(\mathsf{N}\xi)} \sum_{\omega\in\Omega}\sum_{k\in\mathbb{Z}^d} \overline{\widehat{f}(\xi+2\pi\mathsf{N}^{-1}\omega+2\pi\mathsf{N}^{-1}k)}\widehat{\phi}(\mathsf{N}\xi+2\pi\omega+2\pi k)$$
$$\times\left(\chi_{\Omega_{\mathsf{M}}}(\omega)\overline{\widehat{a}(\mathsf{N}\xi)}\widehat{a}(\mathsf{N}\xi+2\pi\omega)+\sum_{\ell=1}^{s}\chi_{\Omega_{\mathsf{M}_\ell}}(\omega)\overline{\widehat{b_\ell}(\mathsf{N}\xi)}\widehat{b_\ell}(\mathsf{N}\xi+2\pi\omega)-\boldsymbol{\delta}(\omega)\right)d\xi=0. \tag{2.29}$$

By a similar argument as in [13, Lemma 5] (also see the argument of (2.7)$\Longrightarrow$(2.5) in the proof of Theorem 1), we can conclude that (2.29) holds if and only if

$$\overline{\widehat{\phi}(\xi)}\widehat{\phi}(\xi+2\pi\omega+2\pi k)\left(\chi_{\Omega_{\mathsf{M}}}(\omega)\overline{\widehat{a}(\xi)}\widehat{a}(\xi+2\pi\omega)+\sum_{\ell=1}^{s}\chi_{\Omega_{\mathsf{M}_\ell}}(\omega)\overline{\widehat{b_\ell}(\xi)}\widehat{b_\ell}(\xi+2\pi\omega)-\boldsymbol{\delta}(\omega)\right)=0, \tag{2.30}$$

for almost every $\xi\in\mathbb{R}^d$ and for all $\omega\in\Omega_{\mathsf{M}}\cup(\cup_{\ell=1}^{s}\Omega_{\mathsf{M}_\ell})$ and $k\in\mathbb{Z}^d$. If $\sum_{k\in\mathbb{Z}^d}|\widehat{\phi}(\xi+2\pi k)|^2\neq 0$ for almost every $\xi\in\mathbb{R}^d$, then it is easy to deduce that (2.30) is equivalent to (2.5) and (2.6). This proves the converse direction. $\square$

A filter bank $\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1,\dots,b_s\,!\,\mathsf{M}_s\}$ satisfying (2.30) is called *a generalized tight framelet filter bank* in [14] (for the case $\mathsf{M}_1=\cdots=\mathsf{M}_s=\mathsf{M}$). In fact, under the condition (2.19), the above proof shows that $\{\phi\,!\,\mathsf{M};\psi^1\,!\,\mathsf{M}_1,\dots,\psi^s\,!\,\mathsf{M}_s\}$ is a tight framelet in $L_2(\mathbb{R}^d)$ if and only if $\{a\,!\,\mathsf{M}; b_1\,!\,\mathsf{M}_1,\dots,b_s\,!\,\mathsf{M}_s\}$ is a generalized tight framelet filter bank. Since $\mathsf{M}^{-1}\mathsf{M}_\ell\mathbb{Z}^d=\mathbb{Z}^d$ may not hold any more for all $\ell=1,\dots,s$, the system $\mathsf{AS}_0(\{\phi\,!\,\mathsf{M};\psi^1\,!\,\mathsf{M}_1,\dots,\psi^s\,!\,\mathsf{M}_s\})$ in (2.22) is not covered by the traditional theory of wavelet analysis.

3. Directional tensor product complex tight framelets with low redundancy

In this section we first briefly recall the directional tensor product complex tight framelets from [15,20]. Then we shall briefly explain the directionality of tensor product complex tight framelets TP-$\mathbb{C}$TF$_m$ and our particular choice of TP-$\mathbb{C}$TF$_6$. Built on the results on tight framelet filter banks with mixed sampling factors in Section 2, we shall provide the details on our proposed directional tensor product complex tight framelet filter bank TP-$\mathbb{C}$TF$_6^{\downarrow}$ with low redundancy rate as well as the more general TP-$\mathbb{C}$TF$_m^{\downarrow}$ with $m\geq 3$.

3.1. Tensor product complex tight framelets and their redundancy rates

For $c_L<c_R$ and positive numbers $\varepsilon_L,\varepsilon_R$ satisfying $\varepsilon_L+\varepsilon_R\leqslant c_R-c_L$, we define a bump function $\chi_{[c_L,c_R];\varepsilon_L,\varepsilon_R}$ on $\mathbb{R}$ [12,15,20] by

$$\chi_{[c_L,c_R];\varepsilon_L,\varepsilon_R}(\xi):=\begin{cases}0, & \xi\leqslant c_L-\varepsilon_L \text{ or } \xi\geq c_R+\varepsilon_R,\\ \cos\left(\frac{\pi(c_L+\varepsilon_L-\xi)}{4\varepsilon_L}\right), & c_L-\varepsilon_L<\xi<c_L+\varepsilon_L,\\ 1, & c_L+\varepsilon_L\leqslant\xi\leqslant c_R-\varepsilon_R,\\ \cos\left(\frac{\pi(\xi-c_R+\varepsilon_R)}{4\varepsilon_R}\right), & c_R-\varepsilon_R<\xi<c_R+\varepsilon_R.\end{cases} \tag{3.1}$$

Note that $\chi_{[c_L,c_R];\varepsilon_L,\varepsilon_R}$ is a continuous function supported on $[c_L-\varepsilon_L,c_R+\varepsilon_R]$.

Let $s\in\mathbb{N}$ and $0<c_1<c_2<\cdots<c_{s+1}:=\pi$ and $\varepsilon_0,\varepsilon_1,\dots,\varepsilon_{s+1}$ be positive real numbers satisfying

$$\varepsilon_0+\varepsilon_1\leqslant c_1\leqslant\tfrac{\pi}{2}-\varepsilon_1 \quad\text{and}\quad \varepsilon_\ell+\varepsilon_{\ell+1}\leqslant c_{\ell+1}-c_\ell\leqslant\pi-\varepsilon_\ell-\varepsilon_{\ell+1}, \qquad \forall\,\ell=1,\dots,s.$$

A real-valued low-pass filter a and $2s$ complex-valued high-pass filters $b_1^+,\dots,b_s^+,b_1^-,\dots,b_s^-$ are defined through their 2π-periodic Fourier series on the basic interval $[-\pi,\pi)$ as follows:

$$\widehat{a} := \chi_{[-c_1,c_1];\varepsilon_1,\varepsilon_1}, \quad \widehat{b_\ell^+} := \chi_{[c_\ell,c_{\ell+1}];\varepsilon_\ell,\varepsilon_{\ell+1}}, \quad \widehat{b_\ell^-} := \overline{\widehat{b_\ell^+}(-\cdot)}, \qquad \ell = 1,\dots,s. \tag{3.2}$$

Note that different notations b_ℓ^p and b_ℓ^n instead of the above notations b_ℓ^+ and b_ℓ^- are used in [15,20]. Then $\mathbb{C}\mathrm{TF}_{2s+1} := \{a; b_1^+,\dots,b_s^+, b_1^-,\dots,b_s^-\}$ is a (one-dimensional dyadic) tight framelet filter bank. The tensor product complex tight framelet filter bank TP-$\mathbb{C}\mathrm{TF}_{2s+1}$ for dimension d is simply

$$\text{TP-}\mathbb{C}\mathrm{TF}_{2s+1} := \otimes^d \mathbb{C}\mathrm{TF}_{2s+1} = \otimes^d\{a; b_1^+,\dots,b_s^+, b_1^-,\dots,b_s^-\}.$$

We can write TP-$\mathbb{C}\mathrm{TF}_{2s+1} = \{\otimes^d a; \text{TP-}\mathbb{C}\mathrm{TF}\text{-HP}_{2s+1}\}$ with TP-$\mathbb{C}\mathrm{TF}$-HP$_{2s+1}$:= TP-$\mathbb{C}\mathrm{TF}_{2s+1}\backslash\{\otimes^d a\}$. This tensor product tight framelet filter bank TP-$\mathbb{C}\mathrm{TF}_{2s+1}$ has one real-valued low-pass filter $\otimes^d a$ and $(2s+1)^d - 1$ complex-valued high-pass filters in TP-$\mathbb{C}\mathrm{TF}$-HP$_{2s+1}$. This family of tensor product complex tight framelets has been introduced in [15].

To further improve the directionality of TP-$\mathbb{C}\mathrm{TF}_{2s+1}$, another closely related family of tensor product complex tight framelet filter banks TP-$\mathbb{C}\mathrm{TF}_{2s+2}$ has been introduced in [20]. Define filters $a, b_1^+,\dots,b_s^+, b_1^-,\dots,b_s^-$ as in (3.2). Define two auxiliary complex-valued filters a^+, a^- through their 2π-periodic Fourier series by

$$\widehat{a^+} := \chi_{[0,c_1];\varepsilon_0,\varepsilon_1}, \qquad \widehat{a^-} := \overline{\widehat{a^+}(-\cdot)}. \tag{3.3}$$

Then $\mathbb{C}\mathrm{TF}_{2s+2} := \{a^+, a^-; b_1^+,\dots,b_s^+, b_1^-,\dots,b_s^-\}$ is also a (one-dimensional dyadic) tight framelet filter bank. Now the tensor product complex tight framelet filter bank TP-$\mathbb{C}\mathrm{TF}_{2s+2}$ for dimension d is defined to be

$$\text{TP-}\mathbb{C}\mathrm{TF}_{2s+2} := \{\otimes^d a; \text{TP-}\mathbb{C}\mathrm{TF}\text{-HP}_{2s+2}\},$$

where TP-$\mathbb{C}\mathrm{TF}$-HP$_{2s+2}$ consists of total $(2s+2)^d - 2^d$ complex-valued high-pass filters given by

$$\Big(\otimes^d \{a^+, a^-; b_1^+,\dots,b_s^+, b_1^-,\dots,b_s^-\}\Big)\backslash\Big(\otimes^d \{a^+, a^-\}\Big).$$

The sampling matrices/factors for all tensor product complex tight framelet filter banks TP-$\mathbb{C}\mathrm{TF}_m$ with $m \ge 3$ are $2I_d$. See [15,19,20,33] for detailed discussions on tensor product complex tight framelets and their applications to image processing.

We now discuss the redundancy rates of TP-$\mathbb{C}\mathrm{TF}_m$ with $m \ge 3$. Note that $\widehat{b_\ell^-} = \overline{\widehat{b_\ell^+}(-\cdot)}$ is equivalent to $b_\ell^- = \overline{b_\ell^+}$, that is, $b_\ell^-(k) = \overline{b_\ell^+(k)}$ for all $k \in \mathbb{Z}$. Therefore, by the definitions in (3.2) and (3.3), we can always rewrite the tight framelet filter bank TP-$\mathbb{C}\mathrm{TF}_m$ as

$$\text{TP-}\mathbb{C}\mathrm{TF}_m = \{\otimes^d a; u, \bar{u} \text{ with } u \in \text{TP-}\mathbb{C}\mathrm{TF}\text{-CHP}_m\}, \tag{3.4}$$

where TP-$\mathbb{C}\mathrm{TF}$-CHP$_m$ is a subset of TP-$\mathbb{C}\mathrm{TF}$-HP$_m$ satisfying TP-$\mathbb{C}\mathrm{TF}$-HP$_m = \{u, \bar{u} \;:\; u \in$ TP-$\mathbb{C}\mathrm{TF}$-CHP$_m\}$. Note that TP-$\mathbb{C}\mathrm{TF}$-CHP$_m$ has exactly n_m filters, where $n_m := \frac{m^d-1}{2}$ for odd integers m and $n_m := \frac{m^d-2^d}{2}$ for even integers m. For a complex-valued filter $u : \mathbb{Z}^d \to \mathbb{C}$, we can uniquely write $u = \mathrm{Re}(u) + i\,\mathrm{Im}(u)$, where $\mathrm{Re}(u)$ and $\mathrm{Im}(u)$ are two real-valued filters defined by $\mathrm{Re}(u)(k) := \mathrm{Re}(u(k))$ and $\mathrm{Im}(u)(k) := \mathrm{Im}(u(k))$ for all $k \in \mathbb{Z}^d$. Due to the identity in (3.4), we observe that the complex-valued tight framelet filter bank TP-$\mathbb{C}\mathrm{TF}_m$ leads to the following real-valued tight framelet filter bank:

$$\{\otimes^d a\} \cup \{\sqrt{2}\,\mathrm{Re}(u), \sqrt{2}\,\mathrm{Im}(u) \;:\; u \in \text{TP-}\mathbb{C}\mathrm{TF}\text{-CHP}_m\}, \tag{3.5}$$

which has one real-valued low-pass filter and $2n_m$ real-valued high-pass filters. Therefore, since the sampling matrices are $2I_d$ with determinant 2^d, the redundancy rate of TP-$\mathbb{C}\mathrm{TF}_m$ in dimension d is no more than

$$\frac{2n_m}{2^d}\sum_{j=0}^{\infty}\frac{1}{2^{dj}}=\frac{2n_m}{2^d-1}=\begin{cases}\frac{m^d-1}{2^d-1}, & \text{if } m \text{ is an odd integer,}\\ \frac{m^d-2^d}{2^d-1}, & \text{if } m \text{ is an even integer.}\end{cases}$$

3.2. Directionality of tensor product complex tight framelets

In this subsection we first address directionality of tensor product complex tight framelets TP-$\mathbb{C}$TF$_m$ with $m \geq 3$. Then we shall discuss the differences of TP-$\mathbb{C}$TF$_m$, in comparison with other known directional representation systems such as discrete cosine transform (DCT), curvelets, shearlets, and the dual tree complex wavelet transform (DT-$\mathbb{C}$WT). We shall also explain why the tensor product complex tight framelet TP-$\mathbb{C}$TF$_6$ is of particular interest to us for the purpose of image and video processing.

To effectively capture edge (and other) singularities, directionality of a representation system is claimed to be very important for many multidimensional applications such as image and video processing [1,8,10,11,14,22–24,31,34] and many references therein. A widely known directional representation system is the discrete cosine transform (DCT) which employs the directional cosine wave-like elements with different frequencies. The DCT is known to be effective for handling textures in an image but it is often incapable of capturing the edge singularities in an image, mainly because all the directional cosine elements in DCT are globally supported (i.e., lack spatial localization).

All the DCT, TP-$\mathbb{C}$TF$_m$, and DT-$\mathbb{C}$WT employ tensor product to build d-dimensional representation systems. Thus, their associated transforms are separable and have simple efficient algorithms which can be easily implemented through filter banks. As discussed in [20, Section 6.2], the underlying mechanism for TP-$\mathbb{C}$TF$_m$ and DT-$\mathbb{C}$WT to have directionality is quite different in nature to nonseparable representation systems such as curvelets and shearlets. The directionality of the complex-valued TP-$\mathbb{C}$TF$_m$ and DT-$\mathbb{C}$WT is in fact achieved by combining some ideas from both wavelets and the DCT. For the convenience of the reader, in the following we briefly recall the argument from [20, Section 6.2] with some added information to explain the directionality of TP-$\mathbb{C}$TF$_m$ and DT-$\mathbb{C}$WT in dimension two. For simplicity, we only discuss complex-valued functions $\psi : \mathbb{R}^2 \to \mathbb{C}$, which are the generators in TP-$\mathbb{C}$TF$_m$ and DT-$\mathbb{C}$WT (see Section 2); the same argument can be applied to high-pass filters $b : \mathbb{Z}^2 \to \mathbb{C}$ as well. Since ψ is obtained by tensor product as $\psi = \psi_1 \otimes \psi_2$, we have $\widehat{\psi}(\xi_1,\xi_2) = \widehat{\psi_1}(\xi_1)\widehat{\psi_2}(\xi_2)$. Due to the frequency separation property of TP-$\mathbb{C}$TF$_m$ and DT-$\mathbb{C}$WT (see Fig. 3 and Section 3.3), we see that

$$\widehat{\psi_1}(\xi) = \chi_{[\zeta_1-c_1,\zeta_1+c_1],\varepsilon_1,\varepsilon_1} \quad \text{and} \quad \widehat{\psi_2}(\xi) = \chi_{[\zeta_2-c_2,\zeta_2+c_2],\varepsilon_2,\varepsilon_2} \tag{3.6}$$

for some $\zeta_1, \zeta_2 \in \mathbb{R}$ and $c_1, c_2, \varepsilon_1, \varepsilon_2 > 0$. In other words, most energy of $\widehat{\psi}$ lies inside the rectangle $[\zeta_1 - c_1, \zeta_1 + c_1] \times [\zeta_2 - c_2, \zeta_2 + c_2]$ whose center is the point $(\zeta_1, \zeta_2)^\mathsf{T}$. Quite often $c_1 \approx c_2$ by design. The relations in (3.6) only hold approximately for the DT-$\mathbb{C}$WT. Define

$$g(\xi_1,\xi_2) = \chi_{[-c_1,c_1],\varepsilon_1,\varepsilon_1}(\xi_1)\chi_{[-c_2,c_2],\varepsilon_2,\varepsilon_2}(\xi_2), \qquad \xi_1,\xi_2 \in \mathbb{R}$$

and let f denote its inverse Fourier transform, that is, $\widehat{f} = g$. Noting that $\overline{g(-\xi)} = g(\xi)$, we see that f is a *real-valued* tensor product function, which is almost isotropic (due to $c_1 \approx c_2$) and concentrates around the origin. Define a vector $\zeta := (\zeta_1, \zeta_2)^\mathsf{T}$ which is the mass center of the function $\widehat{\psi}$. Then $\widehat{\psi}(\xi) = g(\xi - \zeta) = \widehat{f}(\xi - \zeta)$ for all $\xi \in \mathbb{R}^2$, from which we conclude $\psi(x) = f(x)e^{i\zeta\cdot x}$. As discussed in [20, (6.1)], this directly leads to

$$\psi^{[r]}(x) = f(x)\cos(\zeta \cdot x), \qquad \psi^{[i]}(x) = f(x)\sin(\zeta \cdot x), \quad x \in \mathbb{R}^2, \tag{3.7}$$

where the real-valued functions $\psi^{[r]}$ and $\psi^{[i]}$ are the real and imaginary parts of the complex-valued function ψ satisfying $\psi(x) = \psi^{[r]}(x) + i\psi^{[i]}(x)$. Now it is easy to see that $\psi^{[r]}$ and $\psi^{[i]}$ indeed have directionality,

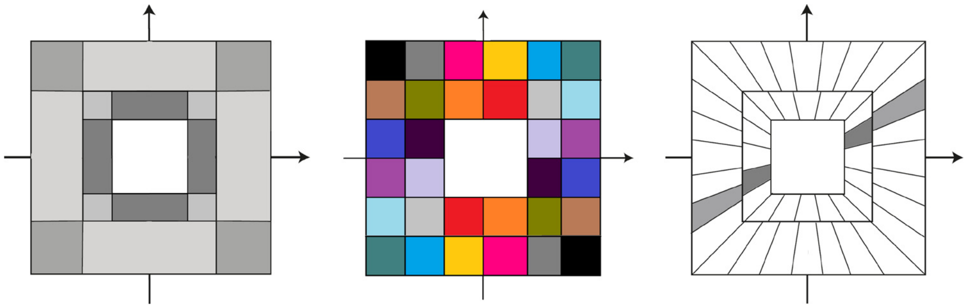


Fig. 1. The frequency tilings of two-dimensional representation systems. Left: Tensor product real-valued orthonormal wavelets with two-level decomposition. It only offers horizontal and vertical directions. Middle: TP-$\mathbb{C}$TF$_6$ with one-level decomposition. Each color represents one directional element in the frequency domain. There are total 16 directional elements with $\pm 45°$ directions repeated once. Right: Shearlets with two-level decomposition. The frequency tiling of curvelets is quite similar to that of shearlets by using rotation instead of shear operation and therefore, the rectangles are replaced by circles in the frequency tiling of curvelets. Directionality of shearlets and curvelets increases with scales.

mainly due to the directional cosine waves $\cos(\zeta \cdot x)$ and sine waves $\sin(\zeta \cdot x)$ (provided $\zeta \neq 0$). Since g is a well localized smooth function, the real-valued window function f is well localized around the origin with rapid decay. When $\|\zeta\| \neq 0$ is small, the cosine wave $\cos(\zeta \cdot x)$ and the sine wave $\sin(\zeta \cdot x)$ have low frequency (i.e., slowly oscillating waves). As a consequence, the elements $\psi^{[r]}$ and $\psi^{[i]}$ exhibit edge-like shapes (we call them edge-like directional elements). Such edge-like directional elements can be used to capture edge singularities. On the other hand, if $\|\zeta\| \neq 0$ is relatively large, the cosine and sine waves have high frequency (i.e., rapidly oscillating waves). As a consequence, the elements $\psi^{[r]}$ and $\psi^{[i]}$ exhibit DCT-like (or texture-like) shapes (we call them texture-like directional elements). See Fig. 4 for the edge-like and texture-like directional elements.

The directionality of elements ψ in curvelets [1,2,34] and shearlets [10,11,21,23–25,27,28] is achieved by designing $\widehat{\psi}$ so that its support obeys the parabolic law in [1]. Roughly speaking, $\widehat{\psi}$ is a needle-like element and is symmetric about the origin. Consequently, ψ itself is an edge-like (or needle-like) element in the spatial domain. As the resolution/scale increases, the angular resolution also increases so that there are more and more directional elements at different angles in finer resolutions. Therefore, curvelets and shearlets cannot have the Cartesian tensor product structure as the separable representation systems such as DCT, TP-$\mathbb{C}$TF$_m$ and DT-$\mathbb{C}$WT. The main difference between curvelets and shearlets lies in that curvelets use rotation operation [1,2,34] while shearlets use shear matrices to get different directions of angles [10]. It has been proved in [1] for curvelets and in [11,24] for shearlets that they provide optimal sparse approximation property for piecewise C^2 cartoon-like functions/images. Note that to achieve such optimal sparse approximation properties, curvelets and shearlets must have low controllable redundancy rates. To reduce the redundancy rates caused by the increased angular resolution, curvelets and shearlets are more sparsely translated in the spatial domain than the traditional wavelets. For more detailed discussions, see [1,2,34] and references therein for curvelets and see [10,11,21,23–25,27,28] for shearlets. The nonseparable directional tight framelets in [14] (also [12]) also achieve directionality in a similar fashion, but with the advantage of having underlying filter banks. As argued in [21], certain types of shear tight frames can be regarded as a subsystem of directional tight framelets in [14]. However, the nonseparable directional tight framelets in [14] have much higher redundancy rates and consequently, their approximation properties are similar to those of traditional wavelets. See Fig. 1 for comparison of frequency tilings for three two-dimensional directional representation systems: tensor product real-valued wavelets, TP-$\mathbb{C}$TF$_6$, and shearlets, while the frequency tiling of curvelets is similar to that of shearlets by using rotation operation for curvelets, instead of shear operation for shearlets.

The number of directions from edge-like directional elements in any tensor product representation systems is intrinsically limited and the total number of angles does not increase as the resolution level increases. As a consequence, the approximation properties of TP-$\mathbb{C}$TF$_m$ are similar to those of traditional wavelets.

However, many natural images have both cartoon-like parts and texture-like parts. The tensor product complex tight framelets offer the flexibility of having both edge-like directional elements and texture-like elements. This makes them particularly appealing for applications in image and video processing.

In the following, let us explain why we are particularly interested in TP-$\mathbb{C}$TF$_6$ and what are the differences among TP-$\mathbb{C}$TF$_m$ with $m \ge 3$. For $m = 3$, TP-$\mathbb{C}$TF$_3$ only offers four directions in dimension two: edge-like directional elements along $0°, \pm 45°, 90°$. Though TP-$\mathbb{C}$TF$_3$ performs better than traditional wavelets, having only four directions in dimension two makes it inadequately effective for the purpose of image processing. The more general TP-$\mathbb{C}$TF$_{2s+1}$ with $s \in \mathbb{N}$ contains s groups of directional elements: The group with the lowest frequency consists of edge-like directional elements along the four directions $0°, \pm 45°, 90°$ (just as in TP-$\mathbb{C}$TF$_3$), while the other $s - 1$ groups consist of texture-like directional elements with different frequency bands, ranging from moderate frequencies to high frequencies.

The family TP-$\mathbb{C}$TF$_{2s+2}$ with $s \in \mathbb{N}$ is built on TP-$\mathbb{C}$TF$_{2s+1}$ with better edge-like directionality by splitting the real-valued low-pass filter a into two auxiliary complex-valued filters a^+ and a^-. As discussed in [20], the TP-$\mathbb{C}$TF$_4$ behaves quite similar to DT-$\mathbb{C}$WT (though they are quite different in nature) and has six directions in dimension two: edge-like directional elements approximately along $\pm 15°, \pm 45°, \pm 75°$. Similarly, the more general TP-$\mathbb{C}$TF$_{2s+2}$ with $s \in \mathbb{N}$ contains s groups of directional elements: The group with the lowest frequency consists of edge-like directional elements along the six directions $\pm 15°, \pm 45°, \pm 75°$ (just as in TP-$\mathbb{C}$TF$_4$), while the other $s - 1$ groups consist of texture-like directional elements with different frequency bands, ranging from moderate frequencies to high frequencies.

For applications in image and video processing, TP-$\mathbb{C}$TF$_{2s+2}$ with $s \in \mathbb{N}$ are often used since they offer better edge-like directionality than TP-$\mathbb{C}$TF$_{2s+1}$. To reduce computational complexity, one often takes $s = 2$ (which leads to TP-$\mathbb{C}$TF$_6$) so that we have one group of edge-like directional elements with six directions to capture cartoon parts and the other group of texture-like directional elements with ten directions to handle texture parts of an image. Many numerical experiments in [20,33] on image processing confirm that TP-$\mathbb{C}$TF$_6$ has the best practical performance among the family of TP-$\mathbb{C}$TF$_m$ with $m \ge 3$. Hence, in this paper we mainly focus on TP-$\mathbb{C}$TF$_6$ by modifying it appropriately so that it has the desired low redundancy rate while keeps all the desirable properties of the original TP-$\mathbb{C}$TF$_6$.

3.3. Directional tensor product complex tight framelets with low redundancy

Now we are ready to construct directional tensor product complex tight framelets with low redundancy by using large sampling factors for TP-$\mathbb{C}$TF$_m$. Though all our arguments in this subsection can be applied to every TP-$\mathbb{C}$TF$_m$ with $m \ge 3$, since the directional tensor product complex tight framelet TP-$\mathbb{C}$TF$_6$ has been known to have superior performance for image denoising in [20] and for image inpainting in [33], we shall only concentrate here on the modification of TP-$\mathbb{C}$TF$_6$. But we shall outline the key ideas for reducing the redundancy rates of a general TP-$\mathbb{C}$TF$_m$ at the end of this subsection.

As discussed in detail in [15,19,20], the directionality of the tensor product complex tight framelets is closely related to the frequency separation property of the high-pass filters in its underlying one-dimensional tight framelet filter bank. More precisely, for a filter u, we say that u has *good frequency separation property* if either $\widehat{u}(\xi) \approx 0$ for all $\xi \in [-\pi, 0]$ or $\widehat{u}(\xi) \approx 0$ for all $\xi \in [0, \pi]$. Moreover, we say that a filter u has the *ideal frequency separation property* if either $\widehat{u}(\xi) = 0$ for all $\xi \in [-\pi, 0]$ or $\widehat{u}(\xi) = 0$ for all $\xi \in [0, \pi]$.

In this subsection, we are interested in building a one-dimensional tight framelet filter bank $\mathbb{C}\mathrm{TF}_6^{\downarrow}$ (called reduced $\mathbb{C}$TF$_6$ or $\mathbb{C}$TF$_6$ down 4), which consists of one real-valued low-pass filter a, two auxiliary complex-valued filters a^+, a^-, and four complex-valued high-pass filters $b_1^+, b_2^+, b_1^-, b_2^-$ such that

(1) $a^- = \overline{a^+}$, $b_1^- = \overline{b_1^+}$, and $b_2^- = \overline{b_2^+}$.
(2) Both $\{a \,!\, 2; b_1^+ \,!\, 4, b_2^+ \,!\, 4, b_1^- \,!\, 4, b_2^- \,!\, 4\}$ and $\mathbb{C}\mathrm{TF}_6^{\downarrow} := \{a^+ \,!\, 4, a^- \,!\, 4; b_1^+ \,!\, 4, b_2^+ \,!\, 4, b_1^- \,!\, 4, b_2^- \,!\, 4\}$ are tight framelet filter banks.

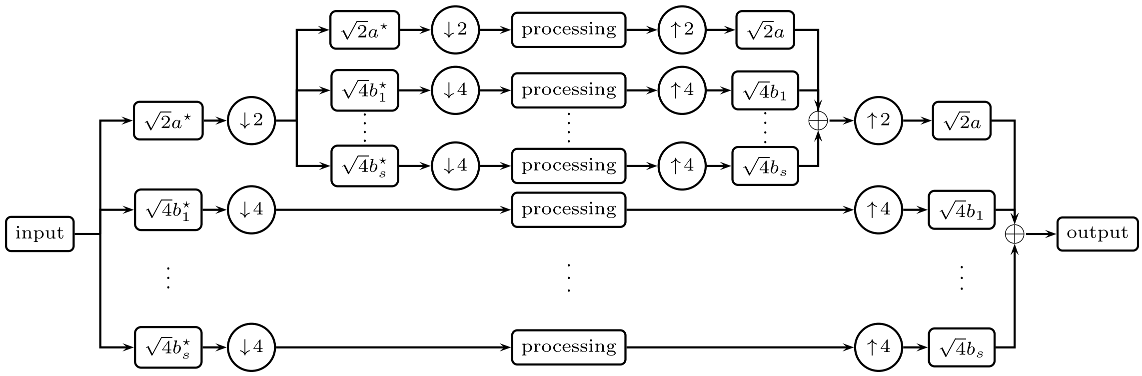


Fig. 2. Diagram of the one-dimensional two-level discrete framelet transform using a one-dimensional tight framelet filter bank $\{a\,!\,2; b_1\,!\,4, \dots, b_s\,!\,4\}$. Here each box with a filter inside means convolution with the filter inside the box. Note that $\mathcal{T}_{a,2}v = 2(v * a^\star)\downarrow 2$ and $\mathcal{S}_{a,2}v = 2(v\uparrow 2) * a$, while $\mathcal{T}_{b_\ell,4}v = 4(v * b_\ell^\star)\downarrow 4$ and $\mathcal{S}_{b_\ell,4}v = 4(v\uparrow 4) * b_\ell$ for $\ell = 1, \dots, s$. Note that $a^\star$ is the flip-conjugate sequence of a given by $a^\star(k) := \overline{a(-k)}$ for all $k \in \mathbb{Z}$, or equivalent, $\widehat{a^\star}(\xi) = \overline{\widehat{a}(\xi)}$.

(3) The auxiliary filters a^+, a^- and all the high-pass filters $b_1^+, b_2^+, b_1^-, b_2^-$ have good frequency separation property.

See Fig. 2 for an illustration of a one-dimensional multilevel fast framelet transform employing a filter bank $\{a\,!\,2; b_1\,!\,4, \dots, b_s\,!\,4\}$.

The directionality of the tensor product complex tight framelet TP-$\mathbb{C}\mathrm{TF}_6^\downarrow$, which we shall introduce later, largely depends on the frequency separation property of all the high-pass filters in the J-level discrete affine system $\mathrm{DAS}_J(\{a\,!\,2; b_1^+\,!\,4, b_2^+\,!\,4, b_1^-\,!\,4, b_2^-\,!\,4\})$ as well as the frequency separation property of the two auxiliary filters a^+ and a^-. For $j \in \mathbb{N}$ and $\ell = 1, 2$, we define

$$\widehat{a_j}(\xi) := \widehat{a}(\xi)\widehat{a}(2\xi)\cdots\widehat{a}(2^{j-2}\xi)\widehat{a}(2^{j-1}\xi), \tag{3.8}$$

$$\widehat{b_{\ell,j}^+} := \widehat{a_{j-1}}(\xi)\widehat{b_\ell^+}(2^{j-1}\xi) = \widehat{a}(\xi)\widehat{a}(2\xi)\cdots\widehat{a}(2^{j-2}\xi)\widehat{b_\ell^+}(2^{j-1}\xi), \tag{3.9}$$

$$\widehat{b_{\ell,j}^-} := \widehat{a_{j-1}}(\xi)\widehat{b_\ell^-}(2^{j-1}\xi) = \widehat{a}(\xi)\widehat{a}(2\xi)\cdots\widehat{a}(2^{j-2}\xi)\widehat{b_\ell^-}(2^{j-1}\xi). \tag{3.10}$$

Note that $a_1 = a, b_{\ell,1}^+ = b_\ell^+$ and $b_{\ell,1}^- = b_\ell^-$. We also define

$$a_{j;k} := 2^{j/2}a_j(\cdot - 2^jk), \quad b_{\ell,j;k}^+ := 2^{(j+1)/2}b_{\ell,j}^+(\cdot - 2^{j+1}k), \quad b_{\ell,j;k}^- := 2^{(j+1)/2}b_{\ell,j}^-(\cdot - 2^{j+1}k)$$

for $\ell = 1, 2$, $j \in \mathbb{N}$, and $k \in \mathbb{Z}$. Then its associated one-dimensional J-level discrete affine system is given by

$$\mathrm{DAS}_J(\{a\,!\,2; b_1^+\,!\,4, b_2^+\,!\,4, b_1^-\,!\,4, b_2^-\,!\,4\}) = \{a_{J;k} \ : \ k \in \mathbb{Z}\} \cup \{b_{\ell,j;k}^+, b_{\ell,j;k}^- \ : \ k \in \mathbb{Z}, \ell = 1, 2, j = 1, \dots, J\}.$$

A detailed construction of $\mathbb{C}\mathrm{TF}_6^\downarrow$ is given in the following result by defining the filters a and $b_1^+, b_2^+, b_1^-, b_2^-$ as in (3.2) with $s = 2$ and a^+, a^- as in (3.3).

Theorem 4. *Let $0 < c_0 < c_1 < c_2 < \pi$ and $\varepsilon_0, \varepsilon_1, \varepsilon_2, \varepsilon_3$ be positive real numbers. The filters a, a^+, b_1^+, b_2^+ are constructed by defining their 2π-periodic Fourier series on the basic interval $[-\pi, \pi)$ as follows:*

$$\widehat{a} := \chi_{[-c_1,c_1];\varepsilon_1,\varepsilon_1}, \quad \widehat{a^+} := \chi_{[0,c_1];\varepsilon_0,\varepsilon_1} \quad \textit{and} \quad \widehat{b_1^+} := \chi_{[c_1,c_2];\varepsilon_1,\varepsilon_2}, \quad \widehat{b_2^+} := \chi_{[c_2,\pi];\varepsilon_2,\varepsilon_3}. \tag{3.11}$$

Define

$$a^- := \overline{a^+}, \quad b_1^- := \overline{b_1^+}, \quad b_2^- := \overline{b_2^+}. \tag{3.12}$$

If

$$\varepsilon_0+\varepsilon_1\leqslant c_1\leqslant \tfrac{\pi}{2}-\varepsilon_0-\varepsilon_1,\quad \tfrac{\pi}{2}+\varepsilon_2+\varepsilon_3\leqslant c_2\leqslant \pi-\varepsilon_2-\varepsilon_3,\quad \varepsilon_1+\varepsilon_2\leqslant c_2-c_1\leqslant \tfrac{\pi}{2}-\varepsilon_1-\varepsilon_2, \tag{3.13}$$

then both $\{a\,!\,2; b_1^+\,!\,4, b_2^+\,!\,4, b_1^-\,!\,4, b_2^-\,!\,4\}$ *and* $\{a^+\,!\,4, a^-\,!\,4; b_1^+\,!\,4, b_2^+\,!\,4, b_1^-\,!\,4, b_2^-\,!\,4\}$ *are tight framelet filter banks. If both* (3.13) *and the following additional conditions are satisfied:*

$$\tfrac{1}{2}c_2+\tfrac{1}{2}\varepsilon_2+c_1+\varepsilon_1\leqslant \pi \quad \textit{and} \quad c_1+\varepsilon_1+\tfrac{1}{2}\varepsilon_3\leqslant \tfrac{\pi}{2}, \tag{3.14}$$

then all the high-pass filters $b^+_{1,j;k}, b^+_{2,j;k}, b^-_{1,j;k} b^-_{2,j;k}, k\in\mathbb{Z}$ *at all scale levels* $j\ge 2$ *in the* J*-level discrete affine system* $\mathrm{DAS}_J(\{a\,!\,2; b_1^+\,!\,4, b_2^+\,!\,4, b_1^-\,!\,4, b_2^-\,!\,4\})$ *have the ideal frequency separation property for every* $J\ge 2$*, more precisely,*

$$\widehat{b^+_{\ell,j}}(\xi)=0,\quad \forall\,\xi\in[-\pi,0]\quad \textit{and}\quad \widehat{b^-_{\ell,j}}(\xi)=0,\quad \forall\,\xi\in[0,\pi]\qquad \textit{for all}\, j\ge 2\quad \textit{and}\quad \ell=1,2, \tag{3.15}$$

where $\widehat{b^+_{\ell,j}}$ *and* $\widehat{b^-_{\ell,j}}$ *are defined in* (3.9) *and* (3.10)*, respectively.*

Proof. By Theorem 1, $\{a\,!\,2; b_1^+\,!\,4, b_2^+\,!\,4, b_1^-\,!\,4, b_2^-\,!\,4\}$ is a tight framelet filter bank if and only if

$$|\widehat{a}(\xi)|^2+|\widehat{b_1^+}(\xi)|^2+|\widehat{b_2^+}(\xi)|^2+|\widehat{b_1^-}(\xi)|^2+|\widehat{b_2^-}(\xi)|^2=1, \tag{3.16}$$

$$\widehat{a}(\xi)\overline{\widehat{a}(\xi+\pi)}+\sum_{\ell=1}^{2}\Big(\widehat{b_\ell^+}(\xi)\overline{\widehat{b_\ell^+}(\xi+\pi)}+\widehat{b_\ell^-}(\xi)\overline{\widehat{b_\ell^-}(\xi+\pi)}\Big)=0, \tag{3.17}$$

$$\sum_{\ell=1}^{2}\Big(\widehat{b_\ell^+}(\xi)\overline{\widehat{b_\ell^+}(\xi+\tfrac{\pi}{2})}+\widehat{b_\ell^-}(\xi)\overline{\widehat{b_\ell^-}(\xi+\tfrac{\pi}{2})}\Big)=0, \tag{3.18}$$

$$\sum_{\ell=1}^{2}\Big(\widehat{b_\ell^+}(\xi)\overline{\widehat{b_\ell^+}(\xi+\tfrac{3\pi}{2})}+\widehat{b_\ell^-}(\xi)\overline{\widehat{b_\ell^-}(\xi+\tfrac{3\pi}{2})}\Big)=0. \tag{3.19}$$

By the definition of the bump function, it is easy to check that the identity in (3.16) holds. By our assumption in (3.13), we see that for all $\xi\in\mathbb{R}$,

$$\widehat{a}(\xi)\widehat{a}(\xi+\pi)=0,\quad \widehat{a^+}(\xi)\widehat{a^+}(\xi+\tfrac{\gamma\pi}{2})=0,\quad \widehat{a^-}(\xi)\widehat{a^-}(\xi+\tfrac{\gamma\pi}{2})=0,\qquad \forall\,\gamma=1,2,3 \tag{3.20}$$

and

$$\widehat{u}(\xi)\widehat{u}(\xi+\tfrac{\gamma\pi}{2})=0,\qquad \forall\,\gamma=1,2,3,\ u\in\{b_1^+,b_2^+,b_1^-,b_2^-\}. \tag{3.21}$$

Therefore, all the three identities in (3.17)–(3.19) trivially hold. Thus, $\{a\,!\,2; b_1^+\,!\,4, b_2^+\,!\,4, b_1^-\,!\,4, b_2^-\,!\,4\}$ is a tight framelet filter bank.

By Theorem 1, $\{a^+\,!\,4, a^-\,!\,4; b_1^+\,!\,4, b_2^+\,!\,4, b_1^-\,!\,4, b_2^-\,!\,4\}$ is a tight framelet filter bank if and only if

$$|\widehat{a^+}(\xi)|^2+|\widehat{a^-}(\xi)|^2+|\widehat{b_1^+}(\xi)|^2+|\widehat{b_2^+}(\xi)|^2+|\widehat{b_1^-}(\xi)|^2+|\widehat{b_2^-}(\xi)|^2=1 \tag{3.22}$$

and for all $\gamma=1,2,3$,

$$\widehat{a^+}(\xi)\overline{\widehat{a^+}(\xi+\tfrac{\gamma\pi}{2})}+\widehat{a^-}(\xi)\overline{\widehat{a^-}(\xi+\tfrac{\gamma\pi}{2})}+\sum_{\ell=1}^{2}\Big(\widehat{b_\ell^+}(\xi)\overline{\widehat{b_\ell^+}(\xi+\tfrac{\gamma\pi}{2})}+\widehat{b_\ell^-}(\xi)\overline{\widehat{b_\ell^-}(\xi+\tfrac{\gamma\pi}{2})}\Big)=0. \tag{3.23}$$

By the definition of the bump function, it is easy to check that the identity in (3.22) holds. It also follows directly from (3.20) and (3.21) that (3.23) trivially holds. Hence, $\{a^+ ! 4, a^- ! 4; b_1^+ ! 4, b_2^+ ! 4, b_1^- ! 4, b_2^- ! 4\}$ is a tight framelet filter bank.

Using (3.13) and (3.14), by calculation we can directly check that the ideal frequency separation property in (3.15) holds. □

We now discuss the tensor product tight framelet filter bank TP-$\mathbb{C}$TF$_6^{\downarrow}$ derived from the one-dimensional tight framelet filter banks in Theorem 4. Define TP-$\mathbb{C}$TF-HP$_6^{\downarrow}$ to be the set consisting of total $6^d - 2^d$ complex-valued high-pass filters as follows:

$$\text{TP-}\mathbb{C}\text{TF-HP}_6^{\downarrow} := \left(\otimes^d\{a^+, a^-; b_1^+, b_2^+, b_1^-, b_2^-\}\right) \setminus \left(\otimes^d\{a^+, a^-\}\right).$$

Then the directional tensor product complex tight framelet filter bank TP-$\mathbb{C}$TF$_6^{\downarrow}$ (called reduced TP-$\mathbb{C}$TF$_6$ or TP-$\mathbb{C}$TF$_6$ down 4) for dimension d is defined to be

$$\text{TP-}\mathbb{C}\text{TF}_6^{\downarrow} := \{\otimes^d a ! 2I_d; u ! 4I_d \text{ with } u \in \text{TP-}\mathbb{C}\text{TF-HP}_6^{\downarrow}\}. \tag{3.24}$$

Note that the low-pass filter $\otimes^d a$ is real-valued and due to the relations in (3.12), we see that $\bar{u} \in$ TP-$\mathbb{C}$TF-HP$_6^{\downarrow}$ for any $u \in$ TP-$\mathbb{C}$TF-HP$_6^{\downarrow}$. Therefore, we can always rewrite the tight framelet filter bank TP-$\mathbb{C}$TF$_6^{\downarrow}$ as

$$\text{TP-}\mathbb{C}\text{TF}_6^{\downarrow} = \{\otimes^d a ! 2I_d; u ! 4I_d, \bar{u} ! 4I_d \text{ with } u \in \text{TP-}\mathbb{C}\text{TF-CHP}_6^{\downarrow}\},$$

where TP-$\mathbb{C}$TF-CHP$_6^{\downarrow}$ is a subset of TP-$\mathbb{C}$TF-HP$_6^{\downarrow}$ and has exactly $\frac{6^d-2^d}{2}$ filters. Consequently, the complex-valued tight framelet filter bank TP-$\mathbb{C}$TF$_6^{\downarrow}$ leads to the following real-valued tight framelet filter bank:

$$\{\otimes^d a ! 2I_d; \sqrt{2}\,\mathrm{Re}(u) ! 4I_d, \sqrt{2}\,\mathrm{Im}(u) ! 4I_d \text{ with } u \in \text{TP-}\mathbb{C}\text{TF-CHP}_6^{\downarrow}\}. \tag{3.25}$$

Therefore, we essentially have only total $(6^d - 2^d)/2$ number of complex-valued high-pass filters in TP-$\mathbb{C}$TF-HP$_6^{\downarrow}$. Thus, the number of real coefficients (by counting a complex number as two real numbers) produced by all the complex-valued filters in TP-$\mathbb{C}$TF$_6^{\downarrow}$ is the same as those produced by the real-valued tight framelet filter bank in (3.25). That is, up to a multiplicative constant $\sqrt{2}$, TP-$\mathbb{C}$TF-HP$_6^{\downarrow}$ produces exactly the same set of real coefficients (by identifying a complex number with two real numbers: its real and imaginary parts) as the $6^d - 2^d$ real-valued filters in (3.25) do. Note that the sampling matrix is $4I_d$ for all high-pass filters from $\otimes^d\{a^+, a^-, b_1^+, b_2^+, b_1^-, b_2^-\}$, while we only perform sampling by $2I_d$ for the low-pass filter $\otimes^d a$. Consequently, regardless of the decomposition level, the redundancy rate of the fast framelet transform employing TP-$\mathbb{C}$TF$_6^{\downarrow}$ for dimension d is no more than

$$\frac{6^d - 2^d}{4^d} \sum_{j=0}^{\infty} \frac{1}{2^{dj}} = \frac{3^d - 1}{2^d - 1}.$$

For example, the redundancy rates of TP-$\mathbb{C}$TF$_6^{\downarrow}$ are $2, 2\frac{2}{3}, 3\frac{5}{7}, 5\frac{1}{3}$ and $7\frac{25}{31}$ for $d = 1, \dots, 5$, respectively. See Table 1 for more details on the redundancy rates of TP-$\mathbb{C}$TF$_6^{\downarrow}$. Note that the redundancy rate of the original TP-$\mathbb{C}$TF$_6$ is 2^d times that of TP-$\mathbb{C}$TF$_6^{\downarrow}$ for dimension d.

We now briefly explain how to reduce the redundancy rate of a general TP-$\mathbb{C}$TF$_m$ with $m \geq 3$. For $s \in \mathbb{N}$, to construct TP-$\mathbb{C}$TF$_{2s+1} = \{a; b_1^+, \dots, b_s^+, b_1^-, \dots, b_s^-\}$ with $b_\ell^- = \overline{b_\ell^+}$ for $\ell = 1, \dots, s$, we often require $\widehat{a}(\xi)\widehat{a}(\xi + \pi) = 0$ and

$$\widehat{b_\ell^+}(\xi)\widehat{b_\ell^+}(\xi + \pi) = 0 \qquad \forall\, \ell = 1, \dots, s. \tag{3.26}$$

Let m_ℓ be the largest positive integer such that

$$\widehat{b_\ell^+}(\xi)\widehat{b_\ell^+}(\xi + \tfrac{2\pi k}{m_\ell}) = 0 \qquad \forall\, k = 1, \dots, m_\ell - 1, \tag{3.27}$$

then we can use the sampling factor m_ℓ for the high-pass filters b_ℓ^+ and b_ℓ^-. This leads to a tight framelet filter bank $\text{TP-}\mathbb{C}\text{TF}_{2s+1}^{\downarrow} = \{a\,!\,2; b_1^+\,!\,m_1, \dots, b_s^+\,!\,m_s, b_1^-\,!\,m_1, \dots, b_s^-\,!\,m_s\}$ with reduced redundancy rate. For $s = 1$, it is not difficult to observe that m_1 cannot be larger than 2 and therefore, the above method does not apply to $\text{TP-}\mathbb{C}\text{TF}_3$ to reduce its redundancy rate. For $s > 1$, by a similar construction as $\text{TP-}\mathbb{C}\text{TF}_6^{\downarrow}$, the above method indeed leads to $\text{TP-}\mathbb{C}\text{TF}_{2s+1}^{\downarrow}$ with lower reduced redundancy rate than that of the original $\text{TP-}\mathbb{C}\text{TF}_{2s+1}$.

For $s \in \mathbb{N}$, to construct $\text{TP-}\mathbb{C}\text{TF}_{2s+2} = \{a^+, a^-; b_1^+, \dots, b_s^+, b_1^-, \dots, b_s^-\}$ with $a^- = \overline{a^+}$ and $b_\ell^- = \overline{b_\ell^+}$ for $\ell = 1, \dots, s$, we often require (3.26) to be satisfied. Let m_ℓ be the largest positive integer such that (3.27) holds. Let m_0 be the largest positive integer such that

$$\widehat{a^+}(\xi)\widehat{a^+}(\xi + \tfrac{2\pi k}{m_0}) = 0 \qquad \forall\, k = 1, \dots, m_0 - 1.$$

Then we have a tight framelet filter bank $\text{TP-}\mathbb{C}\text{TF}_{2s+2}^{\downarrow} = \{a^+\,!\,m_0, a^-\,!\,m_0; b_1^+\,!\,m_1, \dots, b_s^+\,!\,m_s, b_1^-\,!\,m_1, \dots, b_s^-\,!\,m_s\}$ with reduced redundancy rate. For $s = 1$, it is not difficult to observe that m_0 and m_1 cannot be larger than 2 and therefore, the above method does not apply to $\text{TP-}\mathbb{C}\text{TF}_4$ to reduce its redundancy rate. For $s > 1$, by a similar construction as $\text{TP-}\mathbb{C}\text{TF}_6^{\downarrow}$, the above method indeed leads to $\text{TP-}\mathbb{C}\text{TF}_{2s+2}^{\downarrow}$ with lower reduced redundancy rate than that of the original $\text{TP-}\mathbb{C}\text{TF}_{2s+2}$.

Considering the J-level discrete affine systems induced by $\text{TP-}\mathbb{C}\text{TF}_m$ with $m \geq 3$, we also point out that it is possible to slightly further reduce the redundancy rates of $\text{TP-}\mathbb{C}\text{TF}_m^{\downarrow}$ with $m \geq 3$ by a more refined method. However, this leads to a slightly more complicated algorithm and we shall not discuss the detail here. Instead we shall provide numerical experiments for both $\text{TP-}\mathbb{C}\text{TF}_6^{\downarrow}$ and $\text{TP-}\mathbb{C}\text{TF}_5^{\downarrow} = \otimes^d\{a\,!\,2; b_1^+\,!\,4, b_2^+\,!\,4, b_1^-\,!\,4, b_2^-\,!\,4\}$. Note that the redundancy rate of $\text{TP-}\mathbb{C}\text{TF}_5^{\downarrow}$ is $\frac{3^d-1}{2^d-1}$, which is the same as that of $\text{TP-}\mathbb{C}\text{TF}_6^{\downarrow}$ and $\text{TP-}\mathbb{C}\text{TF}_3$.

4. Numerical experiments on image and video processing

In this section, we shall test the performance of our constructed directional tensor product complex tight framelet $\text{TP-}\mathbb{C}\text{TF}_6^{\downarrow}$ with low redundancy in Section 3. Then we shall compare it with many other frame-based methods for image and video processing such as the denoising and inpainting problems.

For the directional tensor product complex tight framelet $\text{TP-}\mathbb{C}\text{TF}_6^{\downarrow}$ with low redundancy that will be used in this section for image and video processing, the parameters in Theorem 4 are set to be

$$c_1 = \tfrac{\pi}{2} - 0.425, \quad c_2 = 2.0, \quad \varepsilon_0 = 0.125, \quad \varepsilon_1 = 0.3, \quad \varepsilon_2 = 0.35, \quad \varepsilon_3 = 0.0778. \tag{4.1}$$

$\text{TP-}\mathbb{C}\text{TF}_5^{\downarrow}$ has the same parameters as those of $\text{TP-}\mathbb{C}\text{TF}_6^{\downarrow}$ for $c_1, c_2, \varepsilon_1, \varepsilon_2, \varepsilon_3$. Note that the above parameters satisfy the conditions in both (3.13) and (3.14). The parameters for other $\text{TP-}\mathbb{C}\text{TF}_m$ are set to be the same as those in the paper [20]. For the convenience of the reader, we explicitly list these parameters here: For $\text{TP-}\mathbb{C}\text{TF}_3$, we set

$$c_1 = \tfrac{33}{32}, \quad c_2 = \pi, \quad \varepsilon_1 = \tfrac{69}{128}, \quad \varepsilon_2 = \tfrac{51}{512}.$$

We compare $\text{TP-}\mathbb{C}\text{TF}_3$ and $\text{TP-}\mathbb{C}\text{TF}_5^{\downarrow}$ with $\text{TP-}\mathbb{C}\text{TF}_6^{\downarrow}$ mainly because they have the same redundancy rate as $\text{TP-}\mathbb{C}\text{TF}_6^{\downarrow}$. For $\text{TP-}\mathbb{C}\text{TF}_6$, we set

$$c_1 = \tfrac{119}{128}, \quad c_2 = \tfrac{\pi}{2} + \tfrac{119}{256}, \quad c_3 = \pi, \quad \varepsilon_0 = \tfrac{35}{128}, \quad \varepsilon_1 = \tfrac{81}{128}, \quad \varepsilon_2 = \tfrac{115}{256}, \quad \varepsilon_3 = \tfrac{115}{256}.$$

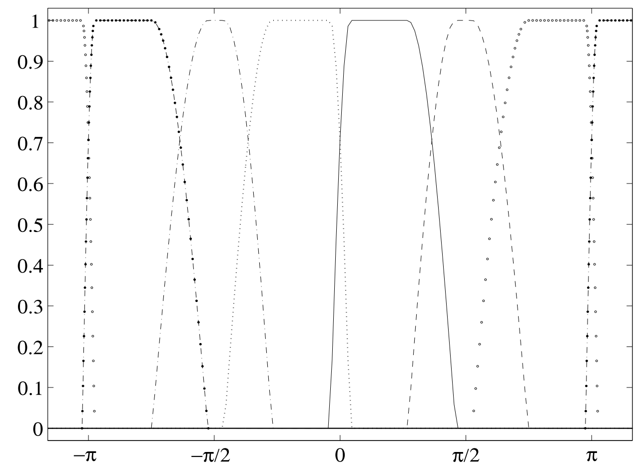


Fig. 3. The one-dimensional tight framelet filter bank $\mathbb{C}\mathrm{TF}_6^{\downarrow} = \{a^+ \,!\, 4, a^- \,!\, 4; b_1^+ \,!\, 4, b_2^+ \,!\, 4, b_1^- \,!\, 4, b_2^- \,!\, 4\}$ in Theorem 4 with parameters in (4.1). Solid line for $\widehat{a^+}$, dotted line for $\widehat{a^-}$, dashed line for $\widehat{b_1^+}$, dash–dotted line for $\widehat{b_1^-}$, circled line for $\widehat{b_2^+}$, and circle–dotted line for $\widehat{b_2^-}$.

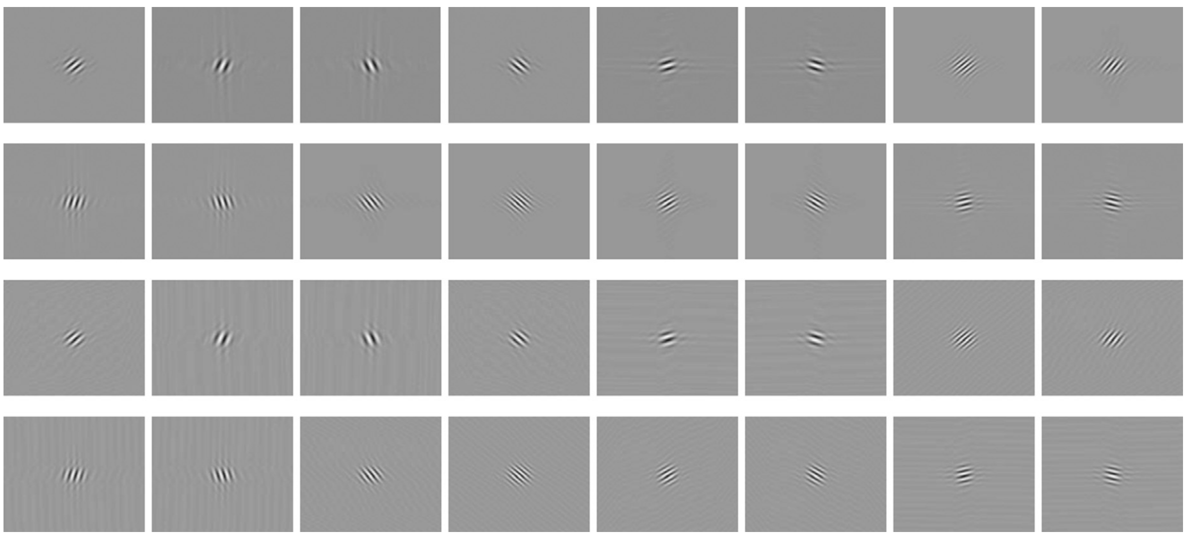


Fig. 4. The first two rows show the real part and the last two rows show the imaginary part of the 2D high-pass filters at the level 4 in $\mathrm{DAS}_5(\mathrm{TP}\text{-}\mathbb{C}\mathrm{TF}_6^{\downarrow})$ for dimension two. Among these 16 graphs for the first two rows or the last two rows, the directions along $\pm 45°$ are repeated once. Hence, there are a total of 14 directions in the 2D discrete affine system $\mathrm{DAS}_5(\mathrm{TP}\text{-}\mathbb{C}\mathrm{TF}_6^{\downarrow})$.

To have some ideas about the filters in $\mathbb{C}\mathrm{TF}_6^{\downarrow}$, see Fig. 3 for the frequency response of the filters in $\mathbb{C}\mathrm{TF}_6^{\downarrow}$. For the directionality of $\mathrm{TP}\text{-}\mathbb{C}\mathrm{TF}_6^{\downarrow}$ in dimension two, see Fig. 4 for some elements of $\mathrm{DAS}_J(\mathrm{TP}\text{-}\mathbb{C}\mathrm{TF}_6^{\downarrow})$ in dimension two with $J = 5$.

As usual, the performance is measured by the peak signal-to-noise ratio (PSNR) which is defined to be

$$\mathrm{PSNR}(u, \mathring{u}) = 10 \log_{10} \frac{255^2}{\mathrm{MSE}(u - \mathring{u})} \quad \text{with} \quad \mathrm{MSE}(u - \mathring{u}) := \frac{1}{N^2} \sum_{j=1}^{N} \sum_{k=1}^{N} |u(j,k) - \mathring{u}(j,k)|^2. \tag{4.2}$$

where u is an original/true image supported on $[1, N]^2$ and $\mathring{u}$ is a reconstructed data.

4.1. Image denoising and image inpainting

We first compare the performance of TP-$\mathbb{C}$TF$_6^{\downarrow}$ for image denoising. We compare the performance of TP-$\mathbb{C}$TF$_6^{\downarrow}$ with two groups of different approaches. The first group uses tensor-product approach including TP-$\mathbb{C}$TF$_3$, TP-$\mathbb{C}$TF$_5^{\downarrow}$ (both have the same redundancy rate $2\frac{2}{3}$ as TP-$\mathbb{C}$TF$_6^{\downarrow}$), and TP-$\mathbb{C}$TF$_6$ (which has the same directionality as TP-$\mathbb{C}$TF$_6^{\downarrow}$ but has a higher redundancy rate $10\frac{2}{3}$), as well as the dual tree complex wavelet transform (DT-$\mathbb{C}$WT) in [31] (which has the redundancy rate 4). The second group employs non-tensor-product approach including curvelets, shearlets, and smooth affine shear tight frames. Curvelets in [2] and compactly supported shearlets in [27,28] can be downloaded from the corresponding authors' websites. We download each of their packages and run their denoising codes on the four 512×512 grayscale test images in (a)–(d) of Fig. 5. Smooth affine shear tight frames (ASTF) are developed by two of the authors of this paper in [21].

The CurveLab package at http://www.curvelab.org has two subpackages: one uses un-equispace FFT and the other uses frequency wrapping. Here we use the frequency wrapping package; detailed information on CurveLab package can be found in [2]. The performance of these two subpackages are very close to each other (less than 0.2 dB differences) and here we choose the one with the frequency wrapping for comparison. For CurveLab, the total number of scales is 5. At the finest scale level, the CurveLab uses an isotropic wavelet transform to avoid checkerboard effect. At the scale level 4, 32 (angular) directions are used. At the scale levels 3 and 2, 16 (angular) directions are used. At the coarsest scale level, 8 (angular) directions are used. The redundancy rate of the CurveLab wrapping package is about 2.8. Hard thresholding $\eta_\lambda^{hard}(c)$ is applied to curvelet coefficients, where

$$\eta_\lambda^{hard}(c) = \begin{cases} c, & |c| > \lambda, \\ 0, & \text{otherwise.} \end{cases}$$

The threshold value λ depends on the scale level and curvelet filters. At the finest scale, $\lambda = 4\sigma_b$ while $\lambda = 3\sigma_b$ for all other scales, where $\sigma_b := \sigma\|b\|_2$, σ is the noise standard deviation and b is the curvelet high-pass filter inducing the coefficient c. That is, σ_b is the standard deviation of the Gaussian noise at the frame band using the high-pass filter b.

The ShearLab package at http://www.shearlab.org also has many subpackages for different implementations. Here we choose two subpackages using compactly supported shearlet frames. One is DST as described in [27] and the other is DNST as described in [28], where DST is a tight frame while DNST is a frame. The DNST in [28] has the best performance so far in the ShearLab package. For DST, the total number of scales is 5. Ten shear directions are used across all scale levels. The redundancy rate of the DST is 40. Hard thresholding $\eta_\lambda^{hard}(c)$ is applied to DST coefficients, where λ depends on the scale. For the finest scale $\lambda = 3.6\sigma_b$ while $\lambda = 2.7\sigma_b$ for all other scales. For DNST, the total number of scales is 4. Sixteen shear directions are used for the finest scale levels 4 and 3; while 8 shear directions are used for the other two scale levels. All filters are implemented in an undecimated fashion. The redundancy rate of DNST is 49. Hard thresholding $\eta_\lambda^{hard}(c)$ is applied to DNST coefficients, where λ depends on the scale. For the finest scale $\lambda = 3.8\sigma_b$ while $\lambda = 2.5\sigma_b$ for all other scales.

For the smooth affine shear tight frames (ASTF) in [21], the total number of scales is 5. Total 16 shear directions are used for the finest scale level. For the next three scales, 8 shear directions are used, and for the coarsest scale level, 4 shear directions are used. The redundancy rate of this system is about 5.4. The soft thresholding $\eta_\lambda^{soft}(c)$ is defined to be

$$\eta_\lambda^{soft}(c) = \begin{cases} c - \lambda\frac{c}{|c|}, & |c| > \lambda, \\ 0, & \text{otherwise.} \end{cases}$$

Local-soft thresholding $\eta^{ls}_{\lambda_c}(c)$ is applied to the normalized ASTF coefficients, which is a slight modification of the soft thresholding and is give by

$$\eta^{ls}_{\lambda_c}(c) := \eta^{soft}_{\lambda_c}(c) \qquad \text{with} \quad \lambda_c = \sigma_b^2/\sigma_c,$$

where $\sigma_b := \sigma\|b\|_2$ with b being the high-pass filter inducing the frame coefficient c and

$$\sigma_c := \begin{cases} \sqrt{\breve{\sigma}_c^2 - \sigma_b^2}, & \breve{\sigma}_c > \sigma_b, \\ 0, & \text{otherwise} \end{cases} \qquad \text{with} \qquad \breve{\sigma}_c^2 := \frac{1}{\#W_c}\sum_{j\in W_c}|c_j|^2, \tag{4.3}$$

where $\#W_c$ is the cardinality of the window set W_c which is taken here to be the $[-4,4]^2$ window centering around the frame coefficient c at the band induced by the filter b. See [21] for more details.

The decomposition levels for all directional tensor product complex tight framelets TP-$\mathbb{C}$TF$_m$ are set to be $J = 5$, while the decomposition level for the dual tree complex wavelet transform is set to be $J = 6$ (see [31,32]). We use symmetric boundary extension for all test images to avoid the boundary effect with the boundary extension size for all test images being 16 pixels. The strategy for processing frame coefficients for all tensor product transforms is the bivariate shrinkage proposed in [32] with window size 7×7 and constant $\sqrt{3}$. Let σ denote the standard deviation of the i.i.d. Gaussian noise. More precisely, a frame coefficient c is processed by the bivariate shrinkage function η^{bs}_λ which is another slight modification of the soft thresholding by taking into account the parent coefficient:

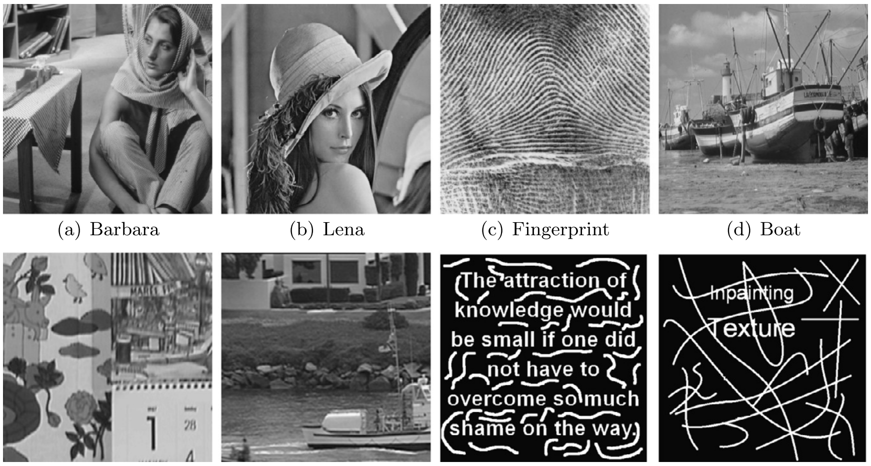
$$\eta^{bs}_{\lambda}(c) := \eta^{soft}_{\lambda_c}(c) \qquad \text{with} \quad \lambda_c = \frac{\sqrt{3}\sigma_b^2}{\sigma_c\sqrt{1+|c_p/c|^2}}, \tag{4.4}$$

where $\sigma_b := \sigma\|b\|_2$ with b being the high-pass filter inducing the frame coefficient c, the frame coefficient c_p is the parent coefficient of c in the immediate higher scale, and σ_c is calculated as in (4.3) with a $[-3,3]^2$ window set W_c centering around the frame coefficient c at the band induced by the filter b.

For transforms producing real-valued coefficients (e.g., transforms using real-valued orthonormal wavelets, curvelets or shearlets), the hard thresholding is often used with the popular heuristic choice of the threshold value $3\sigma_b$. Since TP-$\mathbb{C}$TF$_m$ and TP-$\mathbb{C}$TF$^{\downarrow}_m$ are transforms producing complex coefficients as DT-$\mathbb{C}$WT does, we simply adopt the above bivariate shrinkage in (4.4) which has been used for DT-$\mathbb{C}$WT in [31,32]. For transforms producing complex coefficients such as DT-$\mathbb{C}$WT and TP-$\mathbb{C}$TF$_m$ (as well as TP-$\mathbb{C}$TF$^{\downarrow}_m$), the real part and the imaginary part of a complex coefficient are highly correlated to each other, mainly due to the relation in (3.7). Hence, the popular hard thresholding in the literature may no longer be a suitable choice for transforms producing complex (correlated) coefficients. Since different transforms often have different features and characteristics, as long as the computational complexity of different strategies for processing coefficients is comparable to each other, suitable strategies to be adopted for processing coefficients are at the hands of the researchers for their own developed transforms in order to achieve the best possible performance in applications. All the above discussed shrinkage strategies used for comparison in this paper have similar computational complexity and we do not discuss in this paper what are the possible impacts and consequences of different shrinkage strategies applied to different transforms. For transforms producing complex coefficients, it is our opinion and experience that simple effective better shrinkage strategies other than bivariate shrinkage remain to be developed in the future.

See Fig. 5 for the four 512×512 grayscale test images: *Barbara*, *Lena*, *Fingerprint*, and *Boat*. The comparison results of performance are reported in Table 2 for image denoising under independent identically distributed Gaussian noise with noise standard deviation $\sigma = 5, 10, 25, 40, 50, 80, 100$.

For texture-rich test images such as *Barbara* and *Fingerprint*, we can see from Table 2 that TP-$\mathbb{C}$TF$^{\downarrow}_6$ outperforms TP-$\mathbb{C}$TF$^{\downarrow}_5$, TP-$\mathbb{C}$TF$_3$, DT-$\mathbb{C}$WT, CurveLab, DST and DNST. It can have up to 1.32 dB PSNR



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

(e) Mobile (f) Coastguard (g) Text 1 (h) Text 2

Fig. 5. (a)–(d) are the four 512×512 grayscale test images: *Barbara*, *Lena*, *Fingerprint*, and *Boat*. (e)–(f) are the first frame of the $192 \times 192 \times 192$ videos: *Mobile* and *Coastguard*. (e) and (f) are inpainting masks of size 512×512.

value improvement over TP-$\mathbb{C}$TF$_3$ for *Barbara* at $\sigma = 40$ and about 0.5 dB improvement over DT-$\mathbb{C}$WT for *Fingerprint* at $\sigma = 10$. In comparison with TP-$\mathbb{C}$TF$_6$, TP-$\mathbb{C}$TF$_6^\downarrow$ outperforms TP-$\mathbb{C}$TF$_6$ for the test image *Fingerprint* for all σ noise levels but has slightly worse performance than TP-$\mathbb{C}$TF$_6$ for the test image of *Barbara*. CurveLab (Wrap) also has low redundancy rate, yet its performance is not as good as others for all the test images. DST and DNST have high redundancy rates almost 20 times of that of TP-$\mathbb{C}$TF$_6^\downarrow$. However, for such images of *Barbara* and *Fingerprint*, the performance of DST and DNST is not as good as our TP-$\mathbb{C}$TF$_6^\downarrow$. With redundancy rate about 2 times of TP-$\mathbb{C}$TF$_6^\downarrow$, the performance of ASTF is better than TP-$\mathbb{C}$TF$_6^\downarrow$ only when the noise level is high $\sigma > 40$.

It can be seen from the test images of *Lena* and *Boat* in Fig. 5 that most of their edges are concentrating along the horizontal, the vertical, or the two diagonal directions. For such test images, when σ is small ($\sigma < 40$), the performance of TP-$\mathbb{C}$TF$_6^\downarrow$ is almost the same as TP-$\mathbb{C}$TF$_3$, TP-$\mathbb{C}$TF$_5^\downarrow$ and DT-$\mathbb{C}$WT. Only when σ is high ($\sigma \geq 40$), TP-$\mathbb{C}$TF$_6^\downarrow$ performs not as well as TP-$\mathbb{C}$TF$_3$ and DT-$\mathbb{C}$WT, but generally within less than 0.3 dB loss of performance. For comparison among TP-$\mathbb{C}$TF$_6^\downarrow$ and TP-$\mathbb{C}$TF$_6$, DNST, ASTF, we see at most 0.48 dB loss of performance of TP-$\mathbb{C}$TF$_6^\downarrow$ for both *Lena* and *Boat*. TP-$\mathbb{C}$TF$_6^\downarrow$ outperforms TP-$\mathbb{C}$TF$_5^\downarrow$, DST, and CurveLab for the test images of *Lena* and *Boat*.

In addition to the experimental results in Table 2 for image denoising, we also perform additional experiments by applying the same hard thresholding strategy to all transforms. All the settings in Table 3 are the same as in Table 2, except that all the strategies for processing coefficients are now replaced by the hard thresholding $\eta_\lambda^{hard}(c)$ with $\lambda = 3.6\sigma_b$ for the finest scale and $\lambda = 3\sigma_b$ for all other scales. Numerical results are reported in Table 3. We can see from Table 3 that in general, the performance of TP-$\mathbb{C}$TF$_6^\downarrow$ is worse than TP-$\mathbb{C}$TF$_6$, DST, and DNST but better than other transforms including TP-$\mathbb{C}$TF$_3$, DT-$\mathbb{C}$WT, CurveLab, and ASTF. Note that TP-$\mathbb{C}$TF$_6$, DST, and DNST are transforms with high redundancy rates. Our experiments in Table 3 seem to indicate that the popular hard thresholding method favors transforms with high redundancy rates more than it does for transforms with low redundancy rates. Moreover, comparing results between Tables 2 and 3, we see that TP-$\mathbb{C}$TF$_6^\downarrow$ doesn't outperform TP-$\mathbb{C}$TF$_6$ any more for texture-rich images when the hard thresholding replaces the bivariate shrinkage. This indicates that for different transforms, different thresholding methods should be used so that all the potentials of a particular transform can be exploited.

TP-$\mathbb{C}$TF$_6$ has recently been used in [33] for the image inpainting problem with impressive performance over many other inpainting algorithms. Here we simply use the same inpainting algorithm as developed in [33] but with TP-$\mathbb{C}$TF$_6$ being replaced by TP-$\mathbb{C}$TF$_6^\downarrow$. As similar to most frame-based inpainting algorithms in the

Table 2
Comparison results on image denoising, in terms of PSNR values, of several image denoising methods using our proposed directional tensor product complex tight framelet TP-$\mathbb{C}$TF$_6^\downarrow$ with the redundancy rate $2\frac{2}{3}$, tensor product complex tight framelet TP-$\mathbb{C}$TF$_6$ with the redundancy rate $10\frac{2}{3}$ (having the same directionality as TP-$\mathbb{C}$TF$_6^\downarrow$), TP-$\mathbb{C}$TF$_3$ and TP-$\mathbb{C}$TF$_5^\downarrow$ with redundancy rate $2\frac{2}{3}$ (having the same redundancy rate as TP-$\mathbb{C}$TF$_6^\downarrow$), dual tree complex wavelet transform DT-$\mathbb{C}$WT with the redundancy rate 4 in [31], CurveLab (Wrap) with redundancy rate 2.8 in [2], DST with redundancy rate 40 in [27], DNST with redundancy rate 49 in [28], and ASTF with redundancy rate 5.8 in [21]. The TP-$\mathbb{C}$TF$_6^\downarrow$, TP-$\mathbb{C}$TF$_6$, TP-$\mathbb{C}$TF$_5^\downarrow$, TP-$\mathbb{C}$TF$_3$, DT-$\mathbb{C}$WT are separable transforms using tensor product tight frames while the CurveLab, DST, ASTF are nonseparable transforms using 2D nonseparable tight frames while DNST is only a nonseparable (undecimated) frame. The values in parentheses are the PSNR gain/loss of TP-$\mathbb{C}$TF$_6^\downarrow$ over the compared transform: positive numbers in parentheses mean that TP-$\mathbb{C}$TF$_6^\downarrow$ performs better than the corresponding transform, while negative numbers in parentheses mean that TP-$\mathbb{C}$TF$_6^\downarrow$ performs worse than the corresponding transform.

σ	TP-$\mathbb{C}$TF$_6^\downarrow$	TP-$\mathbb{C}$TF$_6$	TP-$\mathbb{C}$TF$_5^\downarrow$	TP-$\mathbb{C}$TF$_3$	DT-$\mathbb{C}$WT	CurveLab	DST	DNST	ASTF
	512 × 512 Barbara								
5	37.63	37.84(−0.21)	37.59(0.04)	37.16(0.47)	37.37(0.26)	33.83(3.80)	37.76(−0.13)	37.17(0.46)	37.40(0.23)
10	33.97	34.18(−0.21)	33.81(0.16)	33.19(0.78)	33.54(0.43)	29.17(4.80)	33.94(0.03)	33.62(0.35)	33.74(0.23)
25	29.28	29.35(−0.07)	28.86(0.42)	28.04(1.24)	28.81(0.47)	24.83(4.45)	28.90(0.38)	28.93(0.35)	29.29(−0.01)
40	26.85	26.86(−0.01)	26.38(0.47)	25.53(1.32)	26.45(0.40)	23.87(2.98)	26.36(0.49)	26.48(0.37)	27.08(−0.23)
50	25.73	25.71(0.02)	25.24(0.49)	24.48(1.25)	25.36(0.37)	23.38(2.35)	25.22(0.51)	25.31(0.42)	26.05(−0.32)
80	23.51	23.53(−0.02)	23.09(0.42)	22.82(0.69)	23.27(0.24)	22.22(1.29)	23.11(0.40)	22.96(0.55)	23.97(−0.46)
100	22.58	22.64(−0.06)	22.29(0.29)	22.25(0.33)	22.42(0.16)	21.61(0.97)	22.23(0.35)	22.06(0.52)	23.02(−0.44)
	512 × 512 Lena								
5	38.16	38.37(−0.21)	38.14(0.02)	37.98(0.18)	38.25(−0.09)	35.77(2.39)	38.22(−0.06)	38.01(0.15)	38.19(−0.03)
10	35.22	35.48(−0.26)	35.16(0.06)	34.93(0.29)	35.19(0.03)	33.37(1.85)	35.19(0.03)	35.35(−0.13)	35.18(0.04)
25	31.20	31.60(−0.40)	31.07(0.13)	31.17(0.03)	31.29(−0.09)	30.07(1.13)	31.09(0.11)	31.51(−0.31)	31.40(−0.20)
40	29.10	29.52(−0.42)	28.98(0.12)	29.24(−0.14)	29.22(−0.12)	28.15(0.95)	28.92(0.18)	29.32(−0.22)	29.40(−0.30)
50	28.11	28.54(−0.43)	28.00(0.11)	28.34(−0.23)	28.22(−0.11)	27.19(0.92)	27.89(0.22)	28.21(−0.10)	28.46(−0.35)
80	26.11	26.47(−0.36)	25.98(0.13)	26.42(−0.31)	26.15(−0.04)	25.16(0.95)	25.71(0.40)	25.78(0.33)	26.44(−0.34)
100	25.21	25.52(−0.31)	25.09(0.12)	25.52(−0.31)	25.20(0.01)	24.22(0.99)	24.67(0.54)	24.58(0.63)	25.48(−0.27)
	512 × 512 Fingerprint								
5	36.29	36.27(0.02)	36.28(0.01)	35.29(1.00)	35.82(0.47)	33.35(2.94)	36.02(0.27)	35.28(1.01)	35.20(1.09)
10	32.23	32.10(0.13)	32.20(0.03)	30.97(1.26)	31.74(0.49)	30.61(1.62)	31.95(0.28)	31.76(0.47)	30.97(1.26)
25	27.27	26.98(0.29)	27.22(0.05)	26.56(0.71)	27.26(0.01)	26.03(1.24)	27.04(0.23)	27.10(0.17)	26.95(0.32)
40	25.02	24.68(0.34)	24.96(0.06)	24.75(0.27)	24.98(0.04)	23.92(1.10)	24.79(0.23)	24.82(0.20)	25.01(0.01)
50	24.01	23.67(0.34)	23.94(0.07)	23.84(0.17)	23.95(0.06)	23.00(1.01)	23.77(0.24)	23.78(0.23)	24.07(−0.06)
80	21.99	21.66(0.33)	21.87(0.12)	21.73(0.26)	21.91(0.08)	21.18(0.81)	21.65(0.34)	21.63(0.36)	22.11(−0.12)
100	21.09	20.75(0.34)	20.93(0.16)	20.69(0.40)	21.01(0.08)	20.37(0.72)	20.63(0.46)	20.56(0.53)	21.22(−0.13)
	512 × 512 Boat								
5	36.74	36.92(−0.18)	36.73(0.01)	36.45(0.29)	36.73(0.01)	33.59(3.15)	36.51(0.23)	36.04(0.70)	36.66(0.08)
10	33.10	33.41(−0.31)	33.08(0.02)	32.97(0.13)	33.19(−0.09)	30.60(2.50)	33.07(0.03)	33.15(−0.05)	33.07(0.03)
25	28.81	29.26(−0.45)	28.75(0.06)	28.98(−0.17)	29.03(−0.22)	27.51(1.30)	28.75(0.06)	29.23(−0.42)	29.10(−0.29)
40	26.72	27.19(−0.47)	26.64(0.08)	26.98(−0.26)	26.99(−0.27)	25.96(0.76)	26.71(0.01)	27.20(−0.48)	27.14(−0.42)
50	25.79	26.25(−0.46)	25.71(0.08)	26.07(−0.28)	26.06(−0.27)	25.18(0.61)	25.78(0.01)	26.23(−0.44)	26.23(−0.44)
80	24.05	24.41(−0.36)	23.92(0.13)	24.29(−0.24)	24.22(−0.17)	23.55(0.50)	23.90(0.15)	24.17(−0.12)	24.41(−0.36)
100	23.27	23.58(−0.31)	23.13(0.14)	23.50(−0.23)	23.39(−0.12)	22.79(0.48)	23.05(0.22)	23.17(0.10)	23.57(−0.30)

literature, the inpainting algorithm in [33] uses iterative thresholding algorithm with gradually decreasing threshold values. For a detailed description of the inpainting algorithm using TP-$\mathbb{C}$TF$_6$, see [33]. For image inpainting without noise, here we only compare the performance of our TP-$\mathbb{C}$TF$_6^\downarrow$ with three state-of-the-art inpainting algorithms: (1) [33] using TP-$\mathbb{C}$TF$_6$ with redundancy rate $10\frac{2}{3}$. (2) [26] using a tight frame built from the undecimated DCT-Haar wavelet filter which is derived from the discrete cosine transform (DCT) with a block size 7×7 and has the redundancy rate 49. (3) DNST in [28] using undecimated compactly supported nonseparable shearlet frames which has the redundancy rate 49 with $16, 16, 8, 8$ high-pass filters and one low-pass filter. See [3,26,28,33] for image inpainting and comparison results with other frame-based image inpainting algorithms. The inpainting algorithms in [26,28,33] have been generously provided to us by their own authors. The numerical results on image inpainting without noise are presented in Table 4.

We now look at the image inpainting problem with i.i.d. Gaussian noise. The image inpainting algorithm proposed in [33] using TP-$\mathbb{C}$TF$_6$ not only performs well for image inpainting without noise but is also stable and works well for the image inpainting problem with noise. On the other hand, most available image inpainting algorithms (e.g., [3,26,28] and references therein) are not stable and barely work well for image

Table 3
Comparison results on image denoising, in terms of PSNR values, of several image denoising methods using the same hard thresholding $\eta_\lambda^{hard}(c)$ for all the transforms with $\lambda = 3.6\sigma_b$ for the finest scale and $\lambda = 3\sigma_b$ for all other scales.

σ	TP-$\mathbb{C}$TF$_6^\downarrow$	TP-$\mathbb{C}$TF$_6$	TP-$\mathbb{C}$TF$_3$	DT-$\mathbb{C}$WT	CurveLab	DST	DNST	ASTF
	512 × 512 Barbara							
5	36.97	37.60(−0.63)	36.36(0.61)	36.47(0.50)	34.23(2.74)	37.62(−0.65)	37.12(−0.15)	36.15(0.82)
10	33.41	33.94(−0.53)	32.18(1.23)	32.46(0.95)	29.61(3.80)	33.75(−0.34)	33.56(−0.15)	32.56(0.85)
25	28.50	28.71(−0.21)	26.46(2.04)	27.02(1.48)	24.92(3.58)	28.61(−0.11)	28.84(−0.34)	27.83(0.67)
40	25.91	26.05(−0.14)	24.15(1.76)	24.56(1.35)	23.81(2.10)	26.05(−0.14)	26.40(−0.49)	25.50(0.41)
50	24.72	24.85(−0.13)	23.39(1.33)	23.69(1.03)	23.30(1.42)	24.94(−0.22)	25.24(−0.52)	24.37(0.35)
80	22.35	22.89(−0.54)	22.07(0.28)	22.25(0.10)	22.08(0.27)	22.90(−0.55)	22.97(−0.62)	22.50(−0.15)
100	21.44	22.18(−0.74)	21.42(0.02)	21.64(−0.20)	21.42(0.02)	22.12(−0.68)	22.14(−0.70)	21.86(−0.42)
	512 × 512 Lena							
5	37.48	38.07(−0.59)	37.21(0.27)	37.40(0.08)	35.94(1.54)	38.02(−0.54)	37.91(−0.43)	37.02(0.46)
10	34.64	35.33(−0.69)	34.20(0.44)	34.55(0.09)	33.40(1.24)	34.98(−0.34)	35.25(−0.61)	34.21(0.43)
25	30.55	31.37(−0.82)	30.43(0.12)	30.63(−0.08)	29.98(0.57)	30.85(−0.30)	31.41(−0.86)	30.66(−0.11)
40	28.37	29.30(−0.93)	28.49(−0.12)	28.52(−0.15)	28.00(0.37)	28.71(−0.34)	29.27(−0.90)	28.71(−0.34)
50	27.31	28.33(−1.02)	27.48(−0.17)	27.51(−0.20)	27.01(0.30)	27.73(−0.42)	28.20(−0.89)	27.78(−0.47)
80	24.99	26.24(−1.25)	25.23(−0.24)	25.46(−0.47)	24.83(0.16)	25.68(−0.69)	25.97(−0.98)	25.86(−0.87)
100	23.85	25.25(−1.40)	24.11(−0.26)	24.52(−0.67)	23.82(0.03)	24.76(−0.91)	24.90(−1.05)	24.96(−1.11)
	512 × 512 Fingerprint							
5	35.19	36.00(−0.81)	34.29(0.90)	34.86(0.33)	33.83(1.36)	35.82(−0.63)	35.27(−0.08)	33.05(2.14)
10	31.02	31.76(−0.74)	30.02(1.00)	30.96(0.06)	30.61(0.41)	31.64(−0.62)	31.44(−0.42)	29.35(1.67)
25	26.42	26.67(−0.25)	26.39(0.03)	26.10(0.32)	25.89(0.53)	26.69(−0.27)	26.62(−0.20)	25.65(0.77)
40	24.24	24.54(−0.30)	24.40(−0.16)	23.84(0.40)	23.61(0.63)	24.48(−0.24)	24.39(−0.15)	23.57(0.67)
50	23.21	23.60(−0.39)	23.27(−0.06)	22.83(0.38)	22.57(0.64)	23.47(−0.26)	23.38(−0.17)	22.60(0.61)
80	21.16	21.57(−0.41)	20.82(0.34)	20.77(0.39)	20.43(0.73)	21.36(−0.20)	21.27(−0.11)	20.61(0.55)
100	20.23	20.57(−0.34)	19.72(0.51)	19.78(0.45)	19.40(0.83)	20.34(−0.11)	20.20(0.03)	19.62(0.61)
	512 × 512 Boat							
5	35.68	36.42(−0.74)	35.42(0.26)	35.53(0.15)	33.84(1.84)	36.34(−0.66)	36.02(−0.34)	35.02(0.66)
10	32.36	33.16(−0.80)	32.22(0.14)	32.31(0.05)	30.75(1.61)	32.83(−0.47)	32.98(−0.62)	31.96(0.40)
25	28.05	28.90(−0.85)	28.22(−0.17)	28.08(−0.03)	27.47(0.58)	28.44(−0.39)	29.00(−0.95)	28.01(0.04)
40	26.00	26.84(−0.84)	26.23(−0.23)	26.08(−0.08)	25.87(0.13)	26.41(−0.41)	26.97(−0.97)	26.08(−0.08)
50	25.08	25.95(−0.87)	25.33(−0.25)	25.22(−0.14)	25.05(0.03)	25.51(−0.43)	26.03(−0.95)	25.18(−0.10)
80	23.20	24.12(−0.92)	23.44(−0.24)	23.42(−0.22)	23.35(−0.15)	23.74(−0.54)	24.10(−0.90)	23.52(−0.32)
100	22.26	23.27(−1.01)	22.50(−0.24)	22.58(−0.32)	22.50(−0.24)	22.98(−0.72)	23.23(−0.97)	22.78(−0.52)

Table 4
Performance in terms of PSNR values of several image inpainting algorithms without noise. The first two rows for Text 1 and Text 2 are for the inpainting masks Text 1 and Text 2 in Fig. 5. The last two rows are for 50% or 80% randomly missing pixels. [33] uses TP-$\mathbb{C}$TF$_6$ with the redundancy rate $10\frac{2}{3}$. TP-$\mathbb{C}$TF$_6^\downarrow$ uses the same inpainting algorithm as in [33] but with TP-$\mathbb{C}$TF$_6$ being replaced by TP-$\mathbb{C}$TF$_6^\downarrow$ which has the redundancy rate $2\frac{2}{3}$. [26] uses a tight frame built from the undecimated DCT-Haar wavelet filter with redundancy rate 49. [28] uses undecimated compactly supported nonseparable shearlet frames with redundancy rate 49.

	TP-$\mathbb{C}$TF$_6^\downarrow$	[33] (TP-$\mathbb{C}$TF$_6$)	[26]	[28] (DNST)	TP-$\mathbb{C}$TF$_6^\downarrow$	[33] (TP-$\mathbb{C}$TF$_6$)	[26]	[28] (DNST)
	512 × 512 Barbara				512 × 512 Lena			
Text 1	36.68	36.59(0.09)	35.03(1.65)	35.17(1.51)	37.71	38.02(−0.31)	36.73(0.98)	38.17(−0.46)
Text 2	32.99	32.68(0.31)	31.51(1.48)	32.45(0.54)	33.92	34.31(−0.39)	32.10(1.82)	34.10(−0.18)
50%	35.75	35.73(0.02)	33.85(1.90)	34.13(1.62)	37.68	38.00(−0.32)	37.65(0.03)	36.49(1.19)
80%	28.55	28.16(0.39)	26.39(2.16)	28.22(0.33)	31.99	32.33(−0.34)	30.55(1.44)	31.64(0.35)
	512 × 512 Fingerprint				512 × 512 Boat			
Text 1	31.87	31.35(0.52)	30.44(1.43)	31.05(0.82)	34.57	34.96(−0.18)	34.62(0.29)	34.66(0.01)
Text 2	28.36	27.78(0.58)	26.11(2.25)	27.17(1.19)	30.39	30.80(−0.31)	30.35(0.13)	30.65(−0.09)
50%	34.19	34.12(0.07)	33.26(0.93)	31.18(3.01)	34.00	34.42(−0.45)	34.08(−0.17)	33.07(−0.22)
80%	26.77	26.00(0.77)	25.72(1.05)	25.38(1.39)	28.03	28.58(−0.55)	27.89(0.14)	28.01(0.02)

inpainting with noise. Similar to image denoising, for image inpainting with noise, though the noise level σ can be effectively estimated, we assume for simplicity that the noise level σ is known in advance. Such an assumption is commonly adopted in the literature. As pointed out in [33], there are no parameters to be tuned in the image inpainting algorithm proposed in [33] and the image inpainting without noise is simply a special case by taking $\sigma = 0$.

Table 5
Performance in terms of PSNR values for image inpainting with the noise level $\sigma = 10, \dots, 50$. TP-$\mathbb{C}$TF$_6^{\downarrow}$ uses the same inpainting algorithm as in [33] by replacing TP-$\mathbb{C}$TF$_6$ with TP-$\mathbb{C}$TF$_6^{\downarrow}$.

	Text 1		Text 2		50% missing		80% missing	
σ	TP-$\mathbb{C}$TF$_6^{\downarrow}$	[33] (TP-$\mathbb{C}$TF$_6$)	TP-$\mathbb{C}$TF$_6^{\downarrow}$	[33] (TP-$\mathbb{C}$TF$_6$)	TP-$\mathbb{C}$TF$_6^{\downarrow}$	[33] (TP-$\mathbb{C}$TF$_6$)	TP-$\mathbb{C}$TF$_6^{\downarrow}$	[33] (TP-$\mathbb{C}$TF$_6$)
	512 × 512 Barbara							
10	31.76	31.81(−0.05)	29.99	29.85(0.14)	30.94	31.11(−0.17)	26.56	26.70(−0.14)
20	29.00	28.99(0.01)	27.76	27.71(0.05)	27.94	28.00(−0.06)	24.48	24.70(−0.22)
30	27.21	27.18(0.03)	26.24	26.24(0.00)	25.95	25.95(0.00)	23.18	23.34(−0.16)
40	25.91	25.88(0.03)	25.10	25.14(−0.04)	24.58	24.56(0.02)	22.14	22.45(−0.31)
50	24.91	24.91(0.00)	24.18	24.30(−0.12)	23.59	23.60(−0.01)	21.42	21.90(−0.48)
	512 × 512 Lena							
10	33.08	33.42(−0.34)	31.32	31.80(−0.48)	32.86	33.40(−0.54)	29.34	30.25(−0.91)
20	30.83	31.26(−0.43)	29.58	30.10(−0.42)	30.19	30.84(−0.65)	27.33	28.36(−1.03)
30	29.32	29.81(−0.49)	28.34	28.89(−0.55)	28.52	29.18(−0.66)	25.94	26.95(−1.01)
40	28.21	28.72(−0.51)	27.37	27.97(−0.60)	27.35	27.98(−0.63)	24.92	25.93(−1.01)
50	27.33	27.85(−0.52)	26.60	27.22(−0.62)	26.39	27.06(−0.67)	24.11	25.15(−1.04)
	512 × 512 Fingerprint							
10	28.77	28.46(0.31)	26.67	26.24(0.43)	29.09	28.88(0.21)	24.60	24.12(0.48)
20	26.46	26.20(0.26)	25.11	24.72(0.39)	26.10	25.76(0.34)	22.93	22.49(0.54)
30	24.98	24.70(0.28)	23.99	23.59(0.40)	24.43	24.07(0.36)	21.81	21.51(0.30)
40	23.90	23.61(0.29)	23.10	22.72(0.38)	23.10	22.91(0.19)	20.96	20.68(0.28)
50	23.05	22.76(0.29)	22.39	22.00(0.39)	22.39	22.01(0.38)	20.26	19.96(0.30)
	512 × 512 Boat							
10	30.65	31.04(−0.39)	28.40	28.80(−0.40)	30.11	30.65(−0.54)	26.23	27.08(−0.85)
20	28.40	28.84(−0.44)	26.88	27.32(−0.44)	27.61	28.20(−0.59)	24.76	25.56(−0.80)
30	26.95	27.41(−0.46)	25.79	26.24(−0.45)	26.07	26.66(−0.59)	23.75	24.46(−0.71)
40	25.90	26.38(−0.48)	24.98	25.43(−0.45)	25.01	25.56(−0.55)	23.05	23.60(−0.55)
50	25.11	25.57(−0.46)	24.32	24.80(−0.48)	24.23	24.75(−0.52)	22.41	22.96(−0.55)

As we did for image inpainting without noise, for image inpainting with noise, we not only test the performance of the inpainting algorithm in [33] using TP-$\mathbb{C}$TF$_6$ and our modified algorithm using TP-$\mathbb{C}$TF$_6^{\downarrow}$ but also run the inpainting algorithms in [26] and [28]. However, for the noise levels $\sigma = 10, \dots, 50$, the inpainting algorithms in [26] and [28] often have significantly lower performance than [33]. For example, for the test image of Barbara with inpainting mask Text 1 and with the noise level $\sigma = 50$, the performance of PNSR values are 24.91, 24.91, 14.66, 14.48 for TP-$\mathbb{C}$TF$_6^{\downarrow}$, [33] using TP-$\mathbb{C}$TF$_6$, [26,28], respectively. This indicates that the inpainting algorithms in [26] and [28] are mainly designed for the image inpainting problem without noise. As a consequence, for image inpainting with noise, we do not report the comparison results using inpainting algorithms in [26] and [28]. Instead, we only report experimental comparison results using TP-$\mathbb{C}$TF$_6^{\downarrow}$ with the inpainting algorithm in [33] using TP-$\mathbb{C}$TF$_6$ in Table 5.

The experimental results in Tables 4 and 5 show that TP-$\mathbb{C}$TF$_6^{\downarrow}$ performs as well as TP-$\mathbb{C}$TF$_6$ in [33] for image inpainting with or without noise. Both TP-$\mathbb{C}$TF$_6^{\downarrow}$ and TP-$\mathbb{C}$TF$_6$ often outperform the state-of-the-art inpainting algorithms in [26,28].

4.2. Video denoising and video inpainting

For video denoising in three dimensions, the directional tensor product complex tight framelet TP-$\mathbb{C}$TF$_6^{\downarrow}$ has the redundancy rate $3\frac{5}{7}$. We compare the performance of TP-$\mathbb{C}$TF$_6^{\downarrow}$ with the directional tensor product complex tight framelet TP-$\mathbb{C}$TF$_6$ (which has the same directionality as TP-$\mathbb{C}$TF$_6^{\downarrow}$ but has the redundancy rate $29\frac{5}{7}$), TP-$\mathbb{C}$TF$_3$ (which has the same redundancy rate $3\frac{5}{7}$ as TP-$\mathbb{C}$TF$_6^{\downarrow}$), the 3D dual tree complex wavelet transform (DT-$\mathbb{C}$WT, which has the redundancy rate 8), the 3D nonseparable surfacelets in [29] (which has the redundancy rate 6.4), the low-redundancy fast curvelet transform (LR-FCT) in [34] (which has the redundancy rate 10.29), and the 3D nonseparable compactly supported shearlet frames DNST^{3D}-42

Table 6
Comparison results, in terms of PSNR values, of several video denoising methods using our proposed 3D directional tensor product complex tight framelet TP-$\mathbb{C}$TF$_6^{\downarrow}$ with the redundancy rate $3\frac{5}{7}$, 3D tensor product complex tight framelet TP-$\mathbb{C}$TF$_6$ with the redundancy rate $29\frac{5}{7}$ (having the same directionality as TP-$\mathbb{C}$TF$_6^{\downarrow}$), TP-$\mathbb{C}$TF$_3$ with the redundancy rate $3\frac{5}{7}$ (having the same redundancy rate as TP-$\mathbb{C}$TF$_6^{\downarrow}$), the 3D dual tree complex wavelet transform (DT-$\mathbb{C}$WT) with the redundancy rate 8, the 3D non-separable surfacelets in [29] with the redundancy rate 6.4, the low-redundancy fast curvelet transform in [34] with the redundancy rate 10.29, and the 3D nonseparable compactly supported shearlet frames DNST^{3D}-42 and DNST$_2^{3D}$-154 with the redundancy rates 42 and 154, respectively.

σ	TP-$\mathbb{C}$TF$_6^{\downarrow}$	TP-$\mathbb{C}$TF$_6$	TP-$\mathbb{C}$TF$_3$	DT-$\mathbb{C}$WT	Surfacelets	LR-FCT	DNST^{3D}-42	DNST^{3D}-154
	192 × 192 × 192 Mobile							
10	35.26	35.52(−0.26)	33.40(1.86)	34.11(1.15)	32.79(2.47)	34.13(1.13)	35.27(−0.01)	35.91(−0.65)
20	31.58	31.77(−0.19)	29.90(1.68)	30.53(1.05)	29.95(1.63)	30.58(1.00)	31.32(0.26)	32.18(−0.60)
30	29.52	29.66(−0.14)	28.03(1.51)	28.55(0.97)	28.26(1.26)	28.62(0.91)	29.00(0.52)	29.99(−0.47)
40	28.10	28.20(−0.10)	26.76(1.34)	27.17(0.93)	27.05(1.05)	27.24(0.86)	27.37(0.73)	28.42(−0.32)
50	27.01	27.08(−0.07)	25.79(1.22)	26.15(0.86)	26.11(0.90)	26.16(0.86)	26.13(0.88)	27.22(−0.21)
80	24.82	24.82(0.00)	23.87(0.95)	24.03(0.79)	24.25(0.57)	23.83(1.00)	23.69(1.13)	24.75(0.07)
100	23.87	23.82(0.05)	23.06(0.81)	23.06(0.81)	23.40(0.47)	22.70(1.17)	22.63(1.24)	23.62(0.25)
	192 × 192 × 192 Coastguard							
10	33.86	34.15(−0.29)	32.59(1.27)	33.16(0.70)	30.86(3.00)	32.56(1.31)	33.13(0.73)	33.81(0.05)
20	30.26	30.62(−0.36)	29.21(1.05)	29.66(0.60)	28.26(2.00)	29.02(1.24)	29.45(0.81)	30.28(−0.02)
30	28.38	28.73(−0.35)	27.46(0.92)	27.82(0.56)	26.87(1.51)	27.18(1.20)	27.50(0.88)	28.40(−0.02)
40	27.13	27.45(−0.32)	26.28(0.85)	26.58(0.53)	25.91(1.21)	25.94(1.18)	26.17(0.96)	27.13(−0.00)
50	26.18	26.48(−0.30)	25.40(0.78)	25.66(0.52)	25.17(1.01)	25.02(1.17)	25.17(1.01)	26.17(0.01)
80	24.30	24.53(−0.23)	23.67(0.63)	23.84(0.46)	23.61(0.69)	23.05(1.25)	23.17(1.13)	24.17(0.13)
100	23.47	23.65(−0.18)	22.91(0.56)	22.98(0.49)	22.87(0.60)	22.08(1.38)	22.24(1.23)	23.22(0.25)

and DNST$_2^{3D}$-154 in [28] in ShearLab with DNST^{3D}-42 and DNST$_2^{3D}$-154 having the redundancy rates 42 and 154, respectively.

We perform two groups of comparison tests for video denoising with two video sequences: *Mobile* and *Coastguard*, which are the same test videos as used in the paper [28] and can be downloaded from the ShearLab 3D package at http://www.shearlab.org. See Fig. 5 for the first frame of these two videos *Mobile* and *Coastguard*. The first group of tests uses the default settings of each software packages for the comparison among TP-$\mathbb{C}$TF$_6^{\downarrow}$, TP-$\mathbb{C}$TF$_6$, TP-$\mathbb{C}$TF$_3$, DT-$\mathbb{C}$WT, Surfacelets, LR-FCT, DNST^{3D}-42, and DNST$_2^{3D}$-154. The second group of tests uses the same hard thresholding for comparison among TP-$\mathbb{C}$TF$_6^{\downarrow}$, TP-$\mathbb{C}$TF$_6$, TP-$\mathbb{C}$TF$_3$, DT-$\mathbb{C}$WT, and LR-FCT. The experimental results for the first group are reported in Table 6 and the results for the second group are reported in Table 7. The comparison results of performance are all under independent identically distributed Gaussian noise with noise standard deviation $\sigma = 10, 20, 30, 40, 50, 80, 100$.

For the first group of tests, the decomposition level for all tensor product complex tight framelets TP-$\mathbb{C}$TF$_m$ is set to be $J = 4$ and the boundary extension size for all TP-$\mathbb{C}$TF$_m$ is set to be 16 pixels. The strategy for processing frame coefficients for all TP-$\mathbb{C}$TF$_m$ and DT-$\mathbb{C}$WT is the same bivariate shrinkage as outlined in (4.4) but with window size 3 instead of 7. The constant $\sqrt{3}$ in the bivariate shrinkage function in (4.4) for DT-$\mathbb{C}$WT is still set to be $\sqrt{3}$, but this constant is replaced by $\sqrt{4}$ for TP-$\mathbb{C}$TF$_m$ (though there are no significant performance differences if the constant $\sqrt{3}$ is used for TP-$\mathbb{C}$TF$_m$). All parameters for 3D surfacelets, LR-FCT, and the two 3D shearlets DNST^{3D}-42 and DNST$_2^{3D}$-154 are the same as those described in [28,29,34]. The MATLAB routines for Surfacelets, DNST^{3D}-42, and DNST$_2^{3D}$-154 are included in the ShearLab 3D package at http://www.shearlab.org. The executable MATLAB program (without source MATLAB codes) for LR-FCT can be downloaded from http://jstarck.free.fr/cur3d.html. The Surfacelets (4 scales), DNST^{3D}-42 (3 scales), and DNST$_2^{3D}$-154 (3 scales) are all using hard thresholding $\eta_\lambda^{hard}(c)$ with $\lambda = 4\sigma_b$ for the finest scale while $\lambda = 3\sigma_b$ for all other scales. For LR-FCT (3 scales), it also uses hard thresholding $\eta_\lambda^{hard}(c)$ but with $\lambda = 3\sigma_b$ for all scales. Here $\sigma_b = \sigma\|b\|_2$ with σ being the noise standard deviation and b being the high-pass filter inducing the coefficient c.

The second group of tests consists of transforms with low redundancy rates and adopts the same hard thresholding for all transforms. Here we choose TP-$\mathbb{C}$TF$_6^{\downarrow}$, TP-$\mathbb{C}$TF$_6$, TP-$\mathbb{C}$TF$_3$, DT-$\mathbb{C}$WT, and LR-FCT

Table 7
Comparison results using the same hard thresholding for all the transforms, in terms of PSNR values, of several video denoising methods: our proposed 3D directional tensor product complex tight framelet TP-$\mathbb{C}$TF$_6^{\downarrow}$ with the redundancy rate $3\frac{5}{7}$, 3D tensor product complex tight framelet TP-$\mathbb{C}$TF$_6$ with the redundancy rate $29\frac{5}{7}$ (having the same directionality as TP-$\mathbb{C}$TF$_6^{\downarrow}$), TP-$\mathbb{C}$TF$_3$ with the redundancy rate $3\frac{5}{7}$ (having the same redundancy rate as TP-$\mathbb{C}$TF$_6^{\downarrow}$), the 3D dual tree complex wavelet transform (DT-$\mathbb{C}$WT) with the redundancy rate 8, and the low-redundancy fast curvelet transform LR-FCT in [34] with the redundancy rate 10.29, respectively.

σ	TP-$\mathbb{C}$TF$_6^{\downarrow}$	TP-$\mathbb{C}$TF$_6$	TP-$\mathbb{C}$TF$_3$	DT-$\mathbb{C}$WT	LR-FCT
	192 × 192 × 192 Mobile				
10	34.84	35.79(−0.95)	32.81(2.03)	34.08(0.76)	34.13(0.71)
20	30.79	31.64(−0.85)	28.90(1.89)	30.01(0.78)	30.58(0.21)
30	28.53	29.25(−0.72)	26.81(1.72)	27.73(0.80)	28.62(−0.09)
40	26.99	27.64(−0.65)	25.44(1.55)	26.23(0.76)	27.24(−0.25)
50	25.87	26.46(−0.59)	24.49(1.38)	25.16(0.71)	26.16(−0.29)
80	23.69	24.19(−0.50)	22.81(0.88)	23.10(0.59)	23.83(−0.14)
100	22.75	23.28(−0.53)	22.08(0.67)	22.18(0.57)	22.70(0.05)
	192 × 192 × 192 Coastguard				
10	33.07	33.90(−0.83)	31.52(1.55)	32.44(0.63)	32.56(0.51)
20	29.40	30.21(−0.81)	28.08(1.32)	28.75(0.65)	29.02(0.38)
30	27.46	28.25(−0.79)	26.25(1.21)	26.86(0.60)	27.18(0.28)
40	26.16	26.95(−0.79)	25.04(1.12)	25.63(0.53)	25.94(0.22)
50	25.18	25.98(−0.80)	24.14(1.04)	24.73(0.45)	25.02(0.16)
80	23.24	24.05(−0.81)	22.48(0.76)	22.93(0.31)	23.05(0.19)
100	22.38	23.19(−0.81)	21.80(0.58)	22.08(0.30)	22.08(0.30)

for comparison, since all of them have relatively small redundancy rates in dimension three. All the settings in Table 7 are exactly the same as those in Table 6 for the first group of tests except that all the thresholding methods are now replaced by the hard thresholding $\eta_\lambda^{hard}(c)$: $\lambda = 3.6\sigma_b$ for the finest scale while $\lambda = 3\sigma_b$ for all other scales. The only exception is LR-FCT, where $\lambda = 3\sigma_b$ is used across all scales by default, due to the fact that the source MATLAB code for LR-FCT is not available for us to change $\lambda = 3\sigma_b$ into $\lambda = 3.6\sigma_b$ for the finest scale for LR-FCT.

From Table 6, we see that the loss of performance of TP-$\mathbb{C}$TF$_6^{\downarrow}$ is not significant in comparison with TP-$\mathbb{C}$TF$_6$ for both *Mobile* and *Coastguard*. TP-$\mathbb{C}$TF$_6^{\downarrow}$ can outperform DNST$_2^{3D}$-154 when the noise level σ is high ($\sigma > 50$) despite the fact that DNST$_2^{3D}$-154 has the highest redundancy rate 154 which is 41.5 times the redundancy rate of TP-$\mathbb{C}$TF$_6^{\downarrow}$. Generally, TP-$\mathbb{C}$TF$_6^{\downarrow}$ outperforms all other methods (excluding TP-$\mathbb{C}$TF$_6$) for any noise level σ (except a slightly worse performance at $\sigma = 10$ comparing with DNST^{3D}-42 for *Mobile*). Significant improvement can be seen in comparison with the nonseparable 3D surfacelets in [29] (up to 2.47 dB for *Mobile* and 3 dB for *Coastguard*), the low-redundancy fast curvelet transform (LR-FCT) in [34] (up to 1.17 dB for *Mobile* and 1.38 dB for *Coastguard*), and DNST^{3D}-42 in [28] (up to 1.24 dB for *Mobile* and 1.23 dB for *Coastguard*).

From Table 7 using the same hard thresholding strategy, we see that the loss of performance of TP-$\mathbb{C}$TF$_6^{\downarrow}$ is again not that significant in comparison with TP-$\mathbb{C}$TF$_6$ for both *Mobile* and *Coastguard*. TP-$\mathbb{C}$TF$_6^{\downarrow}$ outperforms TP-$\mathbb{C}$TF$_3$ and DT-$\mathbb{C}$WT for both videos while for the video *coastguard*, TP-$\mathbb{C}$TF$_6^{\downarrow}$ outperforms TP-$\mathbb{C}$TF$_3$, DT-$\mathbb{C}$WT, and LR-FCT. For the comparison between TP-$\mathbb{C}$TF$_6^{\downarrow}$ and LR-FCT for the video *Mobile*, we see that the performance of these two methods are very close: When the noise level is low $\sigma < 30$, TP-$\mathbb{C}$TF$_6^{\downarrow}$ performs better than LR-FCT while LR-FCT has better performance when the noise level is high. However, the redundancy rate of LR-FCT is 2.77 times that of TP-$\mathbb{C}$TF$_6^{\downarrow}$ in dimension three.

For video inpainting, we use the same inpainting algorithm as developed in [33] but with 2D tensor product complex tight framelet TP-$\mathbb{C}$TF$_6$ and TP-$\mathbb{C}$TF$_6^{\downarrow}$ being replaced by 3D tensor product complex tight framelet TP-$\mathbb{C}$TF$_6$ and TP-$\mathbb{C}$TF$_6^{\downarrow}$, respectively. We compare the performance of TP-$\mathbb{C}$TF$_6^{\downarrow}$ with surfacelets in [29] and 3D nonseparable compactly supported shearlet frames DNST^{3D}-42 and DNST^{3D}-154 in ShearLab 3D package. The numerical results on video inpainting are presented in Table 8.

From Table 8, we see that the loss of performance of TP-$\mathbb{C}$TF$_6^{\downarrow}$ is acceptable in comparison with TP-$\mathbb{C}$TF$_6$ for both *Mobile* and *Coastguard* in view of the redundancy rate of TP-$\mathbb{C}$TF$_6^{\downarrow}$. Surfacelets do not perform well

Table 8
Performance in terms of PSNR values of several video inpainting algorithms. Gaussian noise with noise levels are taken to be $\sigma = 0, 10, 30$, where $\sigma = 0$ means no noise. 50% and 80% are experiments with 50% and 80% randomly missing pixels, respectively. Comparisons are among 3D tensor product complex tight framelet TP-$\mathbb{C}$TF$_6^{\downarrow}$ with the redundancy rate $3\frac{5}{7}$, 3D tensor product complex tight framelet TP-$\mathbb{C}$TF$_6$ with the redundancy rate $29\frac{5}{7}$ (having the same directionality as TP-$\mathbb{C}$TF$_6^{\downarrow}$), the 3D nonseparable surfacelets in [29] with the redundancy rate 6.4, the 3D nonseparable compactly supported shearlet frame DNST^{3D}-42 with the redundancy rates 42. the 3D nonseparable compactly supported shearlet frame DNST^{3D}-154 with the redundancy rates 154. The masks for inpainting are 50% or 80% uniformly randomly missing pixels.

σ	TP-$\mathbb{C}$TF$_6^{\downarrow}$	TP-$\mathbb{C}$TF$_6$	Surfacelets	DNST^{3D}-42	DNST^{3D}-154
	192 × 192 × 192 Mobile (50% missing)				
0	41.15	41.74(−0.59)	32.09(9.06)	39.54(1.61)	40.71(0.44)
10	32.65	33.09(−0.44)	24.70(7.95)	28.94(3.71)	29.20(3.45)
30	27.56	27.87(−0.31)	16.35(11.21)	20.08(7.48)	20.35(7.21)
	192 × 192 × 192 Mobile (80% missing)				
0	28.22	28.61(−0.39)	22.27(5.95)	31.09(−2.87)	33.21(−4.99)
10	27.32	27.84(−0.52)	20.47(6.85)	27.60(−0.28)	28.45(−1.13)
30	22.89	23.53(−0.64)	15.81(7.08)	21.27(1.62)	21.60(1.29)
	192 × 192 × 192 Coastguard (50% missing)				
0	37.19	37.75(−0.56)	28.67(8.52)	35.74(1.45)	36.69(0.50)
10	30.88	31.48(−0.60)	23.61(7.27)	28.17(2.71)	28.51(2.37)
30	26.59	27.15(−0.56)	16.13(10.46)	19.92(6.67)	20.17(6.42)
	192 × 192 × 192 Coastguard (80% missing)				
0	26.63	27.41(−0.78)	20.96(5.67)	28.56(−1.93)	30.02(−3.39)
10	26.07	26.67(−0.60)	19.73(6.34)	26.18(−0.11)	26.92(−0.85)
30	22.68	23.29(−0.61)	15.81(6.87)	20.87(1.81)	21.10(1.58)

in the inpainting tests even though its redundancy rate is about twice of that of TP-$\mathbb{C}$TF$_6^{\downarrow}$. When the missing pixels are 50%, TP-$\mathbb{C}$TF$_6^{\downarrow}$ outperforms DNST^{3D}-42 and DNST^{3D}-154, especially when the noise level is high ($\sigma = 30$). When the missing pixels are 80%, DNST^{3D}-42 and DNST^{3D}-154 have better performance with low noise level ($\sigma = 0, 10$) comparing to TP-$\mathbb{C}$TF$_6^{\downarrow}$. However, when the noise level is high ($\sigma = 30$), they no longer produce good results as TP-$\mathbb{C}$TF$_6^{\downarrow}$ probably due to the reason that DNST^{3D} employs undecimated transforms.

In summary, in this paper we proposed a family of directional tensor product complex tight framelets TP-$\mathbb{C}$TF$_m^{\downarrow}$ with low redundancy rates. In particular, we constructed a directional tensor product complex tight framelet TP-$\mathbb{C}$TF$_6^{\downarrow}$ (as well as TP-$\mathbb{C}$TF$_5^{\downarrow}$) with low redundancy such that it performs nearly as well as the original TP-$\mathbb{C}$TF$_6$ for image/video denoising/inpainting but it has significantly lower redundancy rates than TP-$\mathbb{C}$TF$_6$ in every dimension. The proposed directional tensor product complex tight framelet TP-$\mathbb{C}$TF$_6^{\downarrow}$ with low redundancy often performs better than other directional representation systems when an image or video is texture-rich, while it performs comparably with other directional representation systems for most other types of images and videos with significantly low redundancy rate of TP-$\mathbb{C}$TF$_6^{\downarrow}$ in comparison with many other separable or nonseparable systems.

References

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

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

[3] J.-F. Cai, R.H. Chan, Z. Shen, A framelet-based image inpainting algorithm, Appl. Comput. Harmon. Anal. 24 (2008) 131–149.

[4] J.-F. Cai, B. Dong, S. Osher, Z. Shen, Image restoration: total variation, wavelet frames, and beyond, J. Amer. Math. Soc. 25 (2012) 1033–1089.

[5] R.H. Chan, S.D. Riemenschneider, L. Shen, Z. Shen, Tight frame: an efficient way for high-resolution image reconstruction, Appl. Comput. Harmon. Anal. 17 (2004) 91–115.

[6] R.R. Coifman, D.L. Donoho, Translation-invariant de-noising, in: Wavelets and Statistics, in: Lecture Notes in Statistics, vol. 103, Springer-Verlag, New York, 1995, pp. 125–150.

[7] I. Daubechies, B. Han, A. Ron, Z. Shen, Framelets: MRA-based constructions of wavelet frames, Appl. Comput. Harmon. Anal. 14 (2003) 1–46.
[8] M.N. Do, M. Vetterli, Contourlets, in: G.V. Welland (Ed.), Beyond Wavelets, Academic Press, 2008, pp. 83–105.
[9] B. Dong, Z. Shen, MRA-Based Wavelet Frames and Applications, IAS/Park City Mathematics Series, vol. 19, 2010.
[10] K. Guo, G. Kutyniok, 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, D. Labate, Optimal sparse multidimensional representation using shearlets, SIAM J. Math. Anal. 9 (2007) 298–318.
[12] B. Han, On dual wavelet tight frames, Appl. Comput. Harmon. Anal. 4 (1997) 380–413.
[13] B. Han, Pairs of frequency-based nonhomogeneous dual wavelet frames in the distribution space, Appl. Comput. Harmon. Anal. 29 (2010) 330–353.
[14] B. Han, Nonhomogeneous wavelet systems in high dimensions, Appl. Comput. Harmon. Anal. 32 (2012) 169–196.
[15] B. Han, Properties of discrete framelet transforms, Math. Model. Nat. Phenom. 8 (2013) 18–47.
[16] B. Han, Matrix splitting with symmetry and symmetric tight framelet filter banks with two high-pass filters, Appl. Comput. Harmon. Anal. 35 (2013) 200–227.
[17] B. Han, Symmetric tight framelet filter banks with three high-pass filters, Appl. Comput. Harmon. Anal. 37 (2014) 140–161.
[18] B. Han, Algorithm for constructing symmetric dual framelet filter banks, Math. Comp. 84 (2015) 767–801.
[19] B. Han, Q. Mo, Z. Zhao, Compactly supported tensor product complex tight framelets with directionality, SIAM J. Math. Anal. 47 (3) (2015) 2464–2494.
[20] B. Han, Z. Zhao, Tensor product complex tight framelets with increasing directionality, SIAM J. Imaging Sci. 7 (2014) 997–1034.
[21] B. Han, X. Zhuang, Smooth affine shear tight frames with MRA structure, Appl. Comput. Harmon. Anal. 39 (2015) 300–338.
[22] N.G. Kingsbury, Complex wavelets for shift invariant analysis and filtering of signals, Appl. Comput. Harmon. Anal. 10 (2001) 234–253.
[23] G. Kutyniok, D. Labate, Shearlets: Multiscale Analysis for Multivariate Data, Birkhäuser, 2012.
[24] G. Kutyniok, W.-Q. Lim, Compactly supported shearlets are optimally sparse, J. Approx. Theory 163 (2011) 1564–1589.
[25] G. Kutyniok, M. Shahram, X. Zhuang, ShearLab: a rational design of a digital parabolic scaling algorithm, SIAM J. Imaging Sci. 5 (2012) 1291–1332.
[26] Y.-R. Li, L. Shen, B.W. Suter, Adaptive inpainting algorithm based on DCT induced wavelet regularization, IEEE Trans. Image Process. 22 (2013) 752–763.
[27] W.-Q. Lim, The discrete shearlet transform: a new directional transform and compactly supported shearlet frames, IEEE Trans. Image Process. 19 (2010) 1166–1180.
[28] W.-Q. Lim, Nonseparable shearlet transform, IEEE Trans. Image Process. 22 (2013) 2056–2065.
[29] Y.M. Lu, M.N. Do, Multidimensional directional filter banks and surfacelets, IEEE Trans. Image Process. 16 (2007) 918–931.
[30] A. Ron, Z. Shen, Affine systems in $L_2(\mathbb{R}^d)$: the analysis of the analysis operator, J. Funct. Anal. 148 (1997) 408–447.
[31] I.W. Selesnick, R.G. Baraniuk, N.G. Kingsbury, The dual-tree complex wavelet transform, IEEE Signal Process. Mag. 22 (6) (2005) 123–151.
[32] L. Sendur, I.W. Selesnick, Bivariate shrinkage with local variance estimation, IEEE Signal Process. Lett. 9 (2002) 438–441.
[33] Y. Shen, B. Han, E. Braverman, Image inpainting using directional tensor product complex tight framelets, arXiv: 1407.3234, 2014.
[34] A. Woiselle, J.-L. Starck, J. Fadili, 3-D data denoising and inpainting with the low-redundancy fast curvelet transform, J. Math. Imaging Vision 39 (2011) 121–139.