Appl. Comput. Harmon. Anal. 60 (2022) 446–470

Contents lists available at ScienceDirect

Applied and Computational Harmonic Analysis

www.elsevier.com/locate/acha

A tailor-made 3-dimensional directional Haar semi-tight framelet for pMRI reconstruction

Yan-Ran Li [a], Lixin Shen [b,*], Xiaosheng Zhuang [c]

[a] College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, 518060 China
[b] Department of Mathematics, Syracuse University, Syracuse, NY 13244, United States of America
[c] Department of Mathematics, City University of Hong Kong, Tat Chee Avenue, Kowloon Tong, Hong Kong

ARTICLE INFO

Article history:
Received 7 May 2021
Received in revised form 21 February 2022
Accepted 20 April 2022
Available online 27 April 2022
Communicated by Bin Han

MSC:
42C15
42C40
42B05
41A55
57N99
58C35
94A12
94C15
93C55
93C95

Keywords:
pMRI
Directional Haar tight framelets
3-dimensional framelet regularization
Semi-tight framelets
SENSE
GRAPPA
ℓ_1-SPIRiT

ABSTRACT

In this paper, we propose a model for parallel magnetic resonance imaging (pMRI) reconstruction, regularized by a carefully designed tight framelet system, that can lead to reconstructed images with much less artifacts in comparison to those from existing models. Our model is motivated from the observations that each receiver coil in a pMRI system is more sensitive to the specific object nearest to the coil, and all coil images are correlated. To exploit these observations, we first stack all coil images together as a 3-dimensional (3D) data matrix, and then design a 3D directional Haar tight framelet (3DHTF) to represent it. After analyzing sparse information of the coil images provided by the high-pass filters of the 3DHTF, we separate the high-pass filters into effective ones and ineffective ones, and we then devise a 3D directional Haar semi-tight framelet (3DHSTF) from the 3DHTF by replacing its ineffective filters with only one filter. This 3DHSTF is tailor-made for coil images, meanwhile, giving a significant saving in computation comparing to the 3DHTF. With the 3DHSTF, we propose an ℓ_1-3DHSTF model for pMRI reconstruction. Numerical experiments for MRI phantom and in-vivo data sets are provided to demonstrate the superiority of our ℓ_1-3DHSTF model in terms of the efficiency of reducing aliasing artifacts in the reconstructed images.

* Corresponding author.
E-mail addresses: lyran@szu.edu.cn (Y.-R. Li), lshen03@syr.edu (L. Shen), xzhuang7@cityu.edu.hk (X. Zhuang).

https://doi.org/10.1016/j.acha.2022.04.003
1063-5203/

1. Introduction and motivation

Since the development of magnetic resonance imaging (MRI) in the 1970s [25], it has been widely used in hospitals and clinics for medical diagnosis thanks to its non-invasive property of not requiring exposure to radiation. Most of the MRI machines in use today utilize a spin-warp imaging scheme [14], where spatial information and associated phase were encoded successively by varying the amplitude of the gradients of the radio frequency (RF) pulses. Such a scheme is a Fourier-transform based MRI method that produces data in the spatial frequency space, known as the K-space. The decoding process involves an inverse Fourier transform to obtain an image. In order to produce an accurate image, it requires enough phase-encoding steps to sufficiently cover the K-space, which can lead to long scan time [36]. To obtain accurate MRI images with less scan time, modern parallel MRI (pMRI) techniques have been developed and advancing during the past two decades. By using multiple RF coils, such as surface coils in an array, to simultaneously receive partial information of the target slice with fewer positions in the K-space data, the pMRI approach accelerates the imaging speed significantly [12], which leads to reduction of motion artifact, breath-hold time, diagnostic duration, and so on. The information loss due to reduction of samples in the K-space can be compensated by the duplicity of the data from multiple coil acquisitions using appropriate reconstruction techniques, e.g., see [22]. However, pMRI has its own drawbacks in terms of specific aliasing artifacts due to undersampling, hardware issues, field of view (FOV) selection, coil-calibration, etc. The success of the pMRI techniques depends on their ability to remove such aliasing artifacts without sacrificing too much of the diagnostic integrity. Current techniques for pMRI reconstruction can be categorized as image-based methods, K-space based methods, or their hybrids [12,42]. The sensitivity encoding (SENSE), e.g., [33,37,44], which is image-based, and the generalized autocalibrating partially parallel acquisitions (GRAPPA), e.g., [16,34,40], which is K-space based, are the two most well-known pMRI techniques for reconstruction and are commercially available for clinical purposes. We next briefly discuss these two methods.

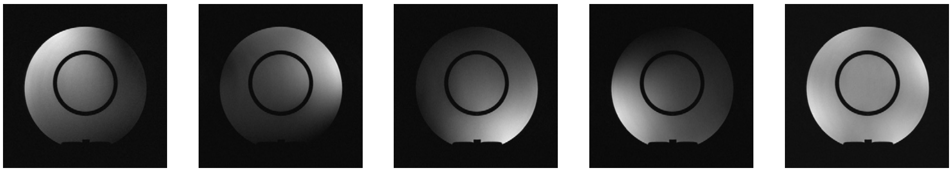
1.1. SENSE and GRAPPA

SENSE was the first pMRI method used routinely, which performs the K-space sampling in the phase-encoding direction; that is, a field of view (FOV) reduction acquisition. To recover the skipped K-space data, multiple receiver coils in an array of surface coils are used to produce multiple coil images. The coil-ι K-space data g_ι from the receiving process can be modeled as follows:

$$g_\iota = \mathcal{P}\mathcal{F}\mathcal{S}_\iota \tilde{u} + \eta_\iota, \quad \iota = 1, \ldots, p,$$

where p is the number of coils, η_ι is the white Gaussian noise, $\tilde{u}$ is the target ground-truth image, $\mathcal{S}_\iota$ is the individual coil sensitivity, $\mathcal{F}$ is the discrete Fourier transform operator, and $\mathcal{P}$ is the sampling operator with respect to the downsampling procedure. See Fig. 1(a)–(d) for an example of coil images (with full FOV selection). Due to the downsampling procedure, the obtained coil images are aliased. Moreover, the accurate estimation of coil sensitivities is needed in the SENSE-based methods, but it is often difficult to determine them due to the complex geometry of the coils. Consequently, the reconstructed images, to approximate the ground-truth image by the SENSE model (e.g., via least square methods), often suffer from aliasing and artifacts. More advanced regularization techniques using sparse representation systems with desirable properties must be employed in order to reduce the aliasing artifact. The TV (Total Variation)-based [23,43] and wavelet-based [6] regularization methods were successfully adopted into the SENSE-based reconstruction problem to suppress the noise or artifacts. Recently, a 2-dimensional (2D) directional Haar tight framelet (2DHTF) system was constructed and successfully applied for the pMRI problem in [28].

GRAPPA is currently the most commonly employed K-space based pMRI method. Unlike SENSE-based methods, GRAPPA does not need the explicit computation of the coil sensitivity $\mathcal{S}_\iota$. Instead, it uses a few



(a) Coil 1 (b) Coil 2 (c) Coil 3 (d) Coil 4 (e) SoS (full)

Fig. 1. (a)–(d) are the four coil images from the corresponding full K-space data; and (e) is the Sum-of-Squares (SoS) image.

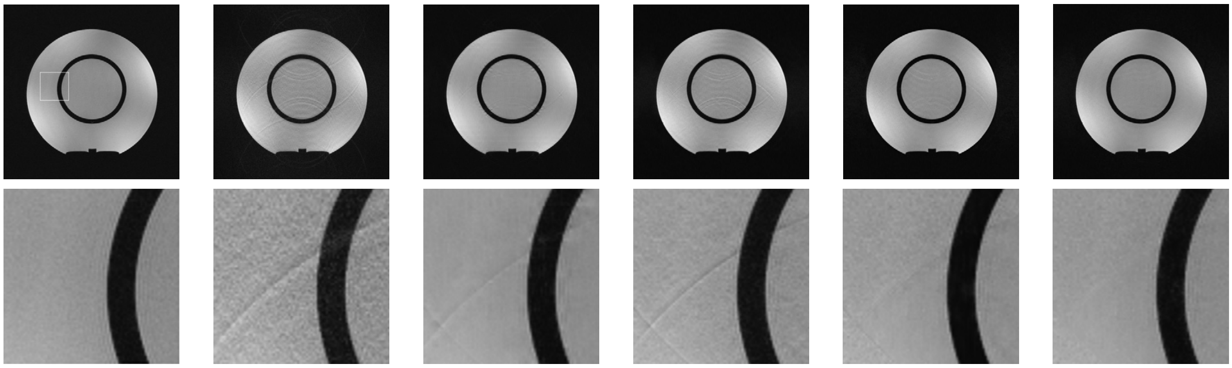
extra lines of the full K-space data, sampled at the region near the center of the K-space during the scan. Such extra lines of the K-space data are called auto-calibration signal (ACS) data. The more ACS lines are used, the more accurate K-space data are reconstructed, but it comes at the cost of increased scan time. Moreover, the number of interpolation kernels to be estimated may be extremely large, especially for the random sampling model in the K-space.

In view of the above drawbacks of GRAPPA, the ℓ_1-SPIRiT (Iterative Self-Consistent Parallel Imaging Reconstruction, [34,40]) method modifies the GRAPPA method by constructing exactly p interpolation kernels regardless of sampling patterns, only one kernel for each coil, and iteratively reconstructs the target K-space data by regularizing coil images together with the joint sparsity-promoting norm $\|\cdot\|_{1,2}$. A general study on sparsity promoting functions can be found in the recent work [38,39]. To avoid overloading symbols, we present the ℓ_1-SPIRiT model here and postpone the discussion of the GRAPPA and ℓ_1-SPIRiT with more details in Section 4:

$$\min_{u} \frac{1}{2}\|(C-I)(Qu+g)\|_2^2 + \lambda\|W_{wav}\mathcal{F}_p^{-1}(Qu+g)\|_{1,2}, \tag{1}$$

where $u = (u_1, \ldots, u_p)$ collects p coil K-space data, $Qu = (I_p \otimes (I - \mathcal{P}))u$ is the missing K-space data to be recovered, $g = (\mathcal{P}g_\iota)_{\iota=1}^{p}$ is the observed p coil K-space data, $C = (C_\iota)_{\iota=1}^{p}$ is the pre-estimated kernel matrix with C_ι being the matrix form of the kernel for coil-ι, $\mathcal{F}_p = I_p \otimes \mathcal{F}$ is the stacked Fourier transform operators, and $W_{wav} = I_p \otimes W$ is the stacked 2D wavelet transform operator W. Here I_p is the identity matrix of size $p \times p$, I is the identity whose size is consistent with that of underlying image, the symbol $\otimes$ denotes the Kronecker product of matrices. Solving the model (1) eventually results in a 3D K-space data $u_{3D} = Qu + g$, which gives a 3D image $\tilde{u}_{3D} = \mathcal{F}_p^{-1}u_{3D}$. The final reconstructed MRI image $\tilde{u}$ is obtained by the SoS (Sum-of-Square) of $\tilde{u}_{3D}$.

In model (1), only 2D transform-based systems are essentially used to decompose coil images [6,23,34,40,43]. That is, W is applied to each coil image independently. However, multiple coil images (or coil K-space data) in the pMRI system are correlated to each other since each coil image contains parts of the information of the same target slice. For example, see Fig. 1(a)–(d) for the four coil images of size 512×512 from (the inverse discrete Fourier transform of) the corresponding full K-space data. The four coil images contain essentially the same information except for varying pixel intensity due to different coil positions. Using only 2D systems may not well exploit such correlated information. In fact, Fig. 2(b) is the SoS image of the four coil images reconstructed by GRAPPA [16] while Fig. 2(c) is reconstructed by the ℓ_1-SPIRiT [34] method using 2D wavelet regularization. Compared with the GRAPPA method without regularization, one can see the effectiveness of the ℓ_1-regularization using the 2D wavelets (with sharper edges and smooth background, also cf. Fig. 2(a) for the SoS image by full K-space data). However, due to the use of 2D systems, the correlated information among coil images is considered by joint-sparsity regularization over wavelet coefficients of multiple coils and the aliasing artifacts may not be well suppressed in the reconstructed images. Fig. 2(b) by GRAPPA has obviously aliasing artifacts while Fig. 2(c) by ℓ_1-SPIRiT reduces aliasing artifacts but many of them are still observable (see their zoom-in parts, respectively).



(a) SoS(full) (b) GRAPPA (c) ℓ_1-SPIRiT (d) ℓ_1-ShearLab3D (e) ℓ_1-3DHTF (f) ℓ_1-3DHSTF

Fig. 2. Reconstruction results on 32% K-space data of the four coil images in Fig. 1(a)–(d) by the uniform sampling mode (one line taken from every four lines) with 48 ACS lines. (a) SoS image of full K-space data (Reference, upper) with zoom-in block (lower). (b) GRAPPA [16]: aliasing artifacts and noise could not be suppressed clearly. (c) ℓ_1-SPIRiT method [34]: noise removed but with aliasing artifacts. (d) ℓ_1-ShearLab3D: noise and artifacts exist. (e) ℓ_1-3DHTF: noise and artifacts are suppressed nicely. (h) ℓ_1-3DHSTF: best performance. The lower images are the zoom-in parts of upper images with respect to the same zoom-in block in (a).

In view of the above discussion, it is very natural to consider the following ℓ_1-W3D model:

$$\min_u \frac{1}{2}\|(C-I)(Qu+g)\|_2^2 + \|\Gamma W_{3D}\mathcal{F}_p^{-1}(Qu+g)\|_1, \tag{2}$$

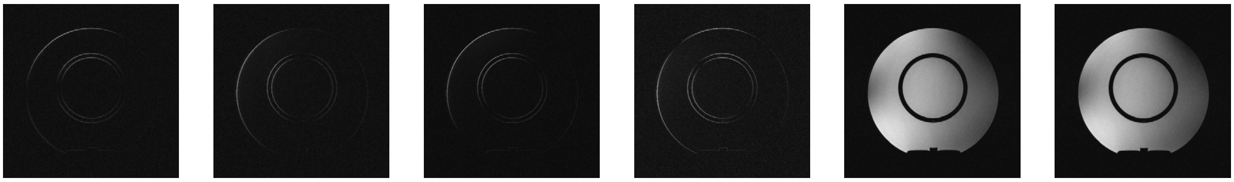
where W_{3D} is a 3D wavelet/framelet transform applied to a 3D image data directly, and Γ is a diagonal matrix with non-negative elements. Since the GRAPPA method does not need the explicit estimation of coil sensitivity functions and in view of the effectiveness of the ℓ_1-SPIRiT model (see Fig. 2(c)), we therefore focus on the development of a suitable W_{3D} system for the above GRAPPA-based model ℓ_1-W3D.

1.2. Motivation: a tailor-made 3D directional Haar semi-tight framelet

Sparsity is always the core in the development of wavelet/framelet representation systems and their applications in image processing (e.g., see [8,7,10,18]). To capture sparsity of high dimensional signals, directionality is one of the most desired properties when designing such representation systems. In fact, directional systems have been intensively studied during the last two decades and shown to play an important role in both theory and application. For example, see curvelets and shearlets in [5,13,26] and tensor product complex tight framelets (TP-$\mathbb{C}$TFs) in [20,21], and many references therein related to directional multiscale representation systems. One would expect that the use of a 3D directional representation system W_{3D} in (2) should lead to better results compared to the use of 2D systems. Unfortunately, without carefully picking a 3D system, one would immediately run into trouble. We summarize the issues, results, and our findings, after we tested various 3D directional systems, as follows.

Unbalanced dimensions. The support of 3D input data to be decomposed by W_{3D} is not evenly distributed due to the fact that the number of coils is much smaller than the dimension of the coil images. For example, when stacking the 4 coil images of size 512×512 in Fig. 1(a)–(d), it becomes a $512 \times 512 \times 4$ cuboid data and the length 4 of the stacked dimension is significantly small compared to 512 in the other two image dimensions. Typical 2D/3D directional systems of shearlets or TP-$\mathbb{C}$TFs are bandlimited systems whose underlying filter banks are with infinitely supported filters. Even with the compactly supported TP-$\mathbb{C}$TF systems developed in [20] and the compactly supported 3D shearlets in [27], the supports of those filters are still too long.

Directionality. The more directionality of the 3D systems do not necessarily lead to the better performance of such systems in the pMRI reconstruction. For example, shearlet systems can achieve directionality



(a) b_x (b) b_y (c) b_{xy} (d) $b_{x,y}$ (e) b_z (f) b_{xz}

Fig. 3. The coil images of Fig. 2 (a)–(d) can be stacked as a 3D image data and decomposed by the 3D directional Haar tight framelet filter banks in DHTF_3^2. The framelet coefficient images (a)–(f) are slices of the 3D framelet coefficient data obtained by the filters $b_x, b_y, b_{xy}, b_{x,y}$, b_z and b_{xz}, respectively. Note that (a)–(d) are with sparse coefficients (in terms of black area), while (e) and (f) are not sparse.

with desired number of directional filters, and we used the shearlet transforms in the software package ShearLab3D[1] of [27] as the W_{3D} system for model (2) by properly setting of filter parameters (cf. [32] on MRI setting). It turns out that the performance of such a shearlet system in ShearLab3D (see Fig. 2(d)) is not as good as the ℓ_1-SPIRiT though it is better than the GRAPPA in terms of suppression of noise and aliasing artifacts. This result demonstrates that the use of general 3D systems with directionality, including the TP-$\mathbb{C}$TF systems (as demonstrated in [28]), do not necessarily perform well in the setting of pMRI reconstruction.

Coil correlated information. The 3D data from pMRI contains intra-coil essential information and inter-coil correlated information. A 3D system that does not take care of such information appropriately will not result in good pMRI reconstruction images. The 2DHF system in [28] only captures the intra-coil information. It is natural to ask whether one can extend the 2DHF system to a 3D setting. Indeed, the work in [19,41] proved that similar directional Haar tight framelet (DHTF) systems exist in any dimension. The underlying multi-dimensional high-pass filters of the DHTF system have only two nonzero filter coefficients with opposite signs. Hence, all of them naturally exhibit directionality. In particular for the 3D case, the respective 3DHTF system (an extension of the 2DHTF system) has 28 framelet functions supported on the unit cube. Since the support of each high-pass filter is extremely short (only 2 taps), it fits the setting of pMRI reconstruction well. We applied such a 3DHTF system in our model (2) and it does produce better results. See Fig. 2(e) for ℓ_1-3DHTF (i.e., W_{3D} = 3DHTF). One can see that though the noise and aliasing artifacts are still observable in Fig. 2(e), the resulted pMRI reconstruction image is clearly better than those of GRAPPA, ℓ_1-SPIRiT, and ℓ_1-ShearLab3D.

The successful application of the 3DHTF system in Fig. 2(e) as well as its drawbacks (still observable noise and artifacts) motives us to further examine the 3DHTF system carefully and eventually leads to the construction of our *tailor-made 3-dimensional directional Haar semi-tight framelet* (3DHSTF) system (see [30] for a preliminary version). Here, we briefly lay out the main ideas for the explanation of both *why* and *why not* the 3DHTF system performs well and for the construction of our 3DHSTF system. We leave the details in Section 3.

The 3DHTF system, also denoted by DHTF_3^1, consists of 28 high-pass filters, but essentially is equivalent to a filter bank, denoted by $\mathrm{DHTF}_3^2 := \{a^H; b_x, b_y, b_z, b_{xy}, b_{x,y}, b_{xz}, b_{x,z}, b_{yz},\ b_{y,z}, b_{xyz}, b_{xy,z}, b_{x,yz}, b_{xz,y}\}$, with 13 high-pass filters by eliminating same directional filters (see Section 3 for details). The low-pass filter a^H is a 3D Haar low-pass filter while the others are 3D high-pass filters with only 2 taps. In short, the low-pass filter captures essentially the inter-coil information while the high-pass filters can capture intra-coil information. This explains the better performance of the ℓ_1-3DHTF result in Fig. 2(e) than those of GRAPPA, ℓ_1-SPIRiT, and ℓ_1-ShearLab3D.

[1] The code is available at: http://www.shearlab.org.

Not all high-pass filters in DHTF_3^2, however, are effective. The subscripts x, y, z in the high-pass filters indicate the directional information that the corresponding filter can capture. The z direction is with respect to the stacked dimension (along p coils) while the x, y directions are with respect to the image dimensions. One can clearly see from Fig. 3 that the high-pass filters $b_x, b_y, b_{xy}, b_{x,y}$ produce sparse framelet coefficients while other two filters b_z, b_{xz} do not produce sparse coefficient sequences. In fact, all filters in DHTF_3^2 involving the z-axis (the z-filters) do not give sparse representations. This is because the pixel intensity varies along different coils and the z-filters taking difference between different coil images only reflect the pixel intensity difference but not the key information. As a result, the use of z-filters may bring unnecessary information that reduces the performance of the system. This answers the question why the ℓ_1-3DHTF still has observable noise and aliasing artifacts, and eventually leads to our tailor-made 3D directional Haar semi-tight framelet system 3DHSTF $:= \{a^H; b_x, b_y, b_{xy}, b_{x,y}, b_{\mathrm{aux}}\}$ from the DHTF_3^2 by replacing all filters involving the z-axis with only one filter b_{aux}. Such a system is called *semi-tight* since the system is very close to a tight framelet system up to certain modifications. One may doubt that the filters $b_x, b_y, b_{xy}, b_{x,y}$ seem to be 2D filters only. We would like to point out that together with the 3D low-pass filter a^H, they are indeed 3D filters that nicely fit to our setting of coil image data. The first level decomposition of 3DHSTF is able to capture 2D features in each coil image while the second level decomposition of 3DHSTF (dealing with data convolved with a^H already) can detect the correlated information between every two consecutive coil images. Our experimental result in Fig. 2(f) shows that model (2) with W_{3D} being our 3DHSTF performs the best among all methods in Fig. 2(b)–(f). Edge details are preserved, the noise is almost removed, and there is almost no aliasing artifacts. We remark that the framelet coefficients from both a^H and b_{aux} will not be not processed, instead will be directly used in the reconstruction of 3DHSTF.

1.3. Our contributions

The contributions of this paper mainly lie in the following four aspects. First, we propose a GRAPPA-based model using 3D wavelet/framelet regularization to reduce noise and aliasing artifacts in the pMRI reconstruction; second, we carefully design a 3DHSTF that not only captures the crucial directional features inside each coil image but also well utilizes the correlated information among different coil images. The 3DHSTF perfectly fits into the pMRI reconstruction algorithm using the GRAPPA-based model; third, fast undecimated discrete framelet transform (UDFmT) algorithms as well as the ADMM scheme [15] for efficiently solving our ℓ_1-W3D model are investigated and developed; and finally, our numerical experiments demonstrate the effectiveness and efficiency of the ℓ_1-3DHSTF model. In fact, we show that aliasing artifacts are significantly reduced using our model comparing to the GRAPPA and the ℓ_1-SPIRiT approaches.

The rest of this paper is organized as follows. In Section 2, we present the theoretical background of tight framelets and tight framelet filter banks. In Section 3 we discuss the construction of 3DHTF filter banks and our tailor-made 3DHSTF filter banks for our ℓ_1-W3D model. In Section 4, we present some details on GRAPPA method and the ℓ_1-SPIRiT method that related to our 3D wavelet/framelet regularization model for the pMRI reconstruction. Moreover, using ADMM scheme, we gives the detailed algorithm for solving our ℓ_1-W3D model step-by-step. Numerical experiments are presented in Section 5. Conclusions and further remarks are given in Section 6.

2. Preliminaries on tight framelets

In this section, we lay out the foundation for the construction of directional Haar tight framelets and introduce the fast undecimated discrete framelet transforms.

2.1. Tight framelets and tight framelet filter banks

We first discuss the connections between tight framelets and filter banks. By $L_2(\mathbb{R}^d)$, we denote the usual space of square integrable functions defined on $\mathbb{R}^d$. We say that $\{\phi;\psi_1,\dots,\psi_s\}\subset L_2(\mathbb{R}^d)$ is a (nonhomogeneous dyadic) tight framelet in $L_2(\mathbb{R}^d)$ if

$$\|f\|^2_{L_2(\mathbb{R}^d)}=\sum_{k\in\mathbb{Z}^d}|\langle f,\phi(\cdot-k)\rangle|^2+\sum_{j=0}^{\infty}\sum_{\iota=1}^{s}\sum_{k\in\mathbb{Z}^d}|\langle f,2^{jd/2}\psi_\iota(2^j\cdot-k)\rangle|^2,\qquad \forall\, f\in L_2(\mathbb{R}^d). \tag{3}$$

Denote $\ell_0(\mathbb{Z}^d)$ the set of all finitely supported sequences. A mask/filter $h=\{h(k)\}_{k\in\mathbb{Z}^d}:\mathbb{Z}^d\to\mathbb{C}$ on $\mathbb{Z}^d$ is a sequence in $\ell_0(\mathbb{Z}^d)$. For a filter h, its Fourier series is defined to be $\widehat{h}(\xi):=\sum_{k\in\mathbb{Z}^d}h(k)e^{-\mathrm{i}k\cdot\xi}$ for $\xi\in\mathbb{R}^d$. In particular, by $\boldsymbol{\delta}$ we denote the Dirac sequence such that $\boldsymbol{\delta}(0)=1$ and $\boldsymbol{\delta}(k)=0$ for all $k\in\mathbb{Z}^d\backslash\{0\}$. For $\gamma\in\mathbb{Z}^d$, we use $\boldsymbol{\delta}_\gamma$ to stand for the sequence $\boldsymbol{\delta}(\cdot-\gamma)$, i.e., $\boldsymbol{\delta}_\gamma(\gamma)=1$ and $\boldsymbol{\delta}_\gamma(k)=0$ for all $k\in\mathbb{Z}^d\backslash\{\gamma\}$. Note that $\widehat{\boldsymbol{\delta}_\gamma}(\xi)=e^{-\mathrm{i}\gamma\cdot\xi}$. We say that a filter bank $\{a;b_1,\dots,b_s\}\subset\ell_0(\mathbb{Z}^d)$ is a (d-dimensional dyadic) tight framelet filter bank if

$$\widehat{a}(\xi)\overline{\widehat{a}(\xi+\pi\omega)}+\sum_{\iota=1}^{s}\widehat{b_\iota}(\xi)\overline{\widehat{b_\iota}(\xi+\pi\omega)}=\boldsymbol{\delta}(\omega),\ \xi\in\mathbb{R}^d, \tag{4}$$

where $\omega\in\{0,1\}^d$ and for a number $x\in\mathbb{C}$, $\bar{x}$ denotes its complex conjugate. Eq. (4) is equivalent to the perfect reconstruction property of the discrete framelet transforms associated with a filter bank [18, Theorems 1.1.1 and 1.1.4].

Assume that $\widehat{a}(0)=\sum_{k\in\mathbb{Z}^d}a(k)=1$. Then one can define compactly supported tempered distributions ϕ and $\psi_1,\dots,\psi_s$ on $\mathbb{R}^d$ through

$$\widehat{\phi}(\xi):=\prod_{j=1}^{\infty}\widehat{a}(2^{-j}\xi)\quad\text{and}\quad\widehat{\psi_\iota}(\xi)=\widehat{b_\iota}(\xi/2)\widehat{\phi}(\xi/2),\xi\in\mathbb{R}^d,\iota=1,\dots,s, \tag{5}$$

where the Fourier transform $\widehat{f}$ of a Lebesgue integrable function $f\in L_1(\mathbb{R}^d)$ is defined to be $\widehat{f}(\xi):=\int_{\mathbb{R}^d}f(x)e^{-\mathrm{i}x\cdot\xi}dx$, $\xi\in\mathbb{R}^d$, and can be naturally extended for functions in $L_2(\mathbb{R}^d)$. It is known that $\{\phi;\psi_1,\dots,\psi_s\}$ is a tight framelet in $L_2(\mathbb{R}^d)$ if and only if $\{a;b_1,\dots,b_s\}$ is a tight framelet filter bank [18, Theorem 4.5.4]. Also cf. [9,11] for related results and many references therein for extensive investigation on tight framelets derived from refinable functions. Consequently, in this paper we mainly focus on the design of framelet filter banks.

2.2. Discrete affine systems and fast discrete framelet transforms

A tight framelet filter bank can be used to (sparsely) represent data sequences through its associated discrete framelet transforms as well as its underlying discrete affine system [17]. More precisely, given a data sequence $v\in l(\mathbb{Z}^d)$ and a filter $h\in l_0(\mathbb{Z}^d)$, the *subdivision operator* $\mathcal{S}_h:l(\mathbb{Z}^d)\to l(\mathbb{Z}^d)$ and the *transition operator* $\mathcal{T}_h:l(\mathbb{Z}^d)\to l(\mathbb{Z}^d)$ are defined to be:

$$\begin{aligned}[]
[\mathcal{S}_hv](\gamma)&:=2^d\sum_{k\in\mathbb{Z}^d}v(k)h(\gamma-2k)=2^d[h*(v\uparrow 2)](\gamma),\qquad \gamma\in\mathbb{Z}^d,\\
[\mathcal{T}_hv](\gamma)&:=2^d\sum_{k\in\mathbb{Z}^d}v(k)\overline{h(k-2\gamma)}=2^d[(h^\star*v)\downarrow 2](\gamma),\qquad \gamma\in\mathbb{Z}^d,
\end{aligned} \tag{6}$$

where $*$ is the *convolution* operation:

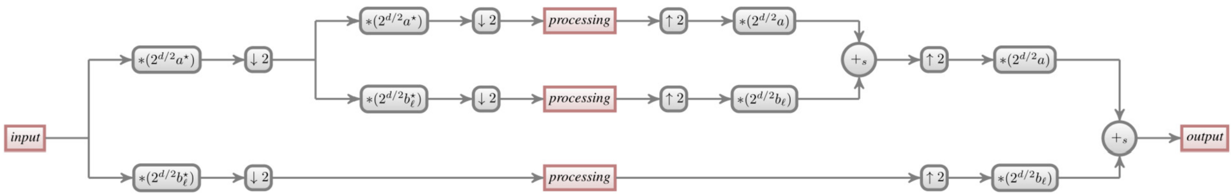


Fig. 4. Multi-level discrete framelet transforms (DFmT) associated with a filter bank $\{a; b_1, \dots, b_s\}$. Each box with b_ℓ runs through $\iota = 1, \dots, s$ and the circle with $+_s$ sums over all s outputs from boxes with b_ι.

$$[h * v](\gamma) := \sum_{k\in\mathbb{Z}^d} h(\gamma - k)v(k), \quad v \in l(\mathbb{Z}^d),\ h \in l_0(\mathbb{Z}^d),$$

$h^\star$ is a filter defined by $h^\star(k) = \overline{h(-k)}$, $k \in \mathbb{Z}^d$, and $\uparrow m$, $\downarrow m$ are the up-, down-sampling operators with $m \in \mathbb{N}$, respectively:

$$[v \uparrow m](\gamma) := \begin{cases} v(m^{-1}\gamma), & \text{if } m^{-1}\gamma \in \mathbb{Z}^d; \\ 0, & \text{otherwise,} \end{cases} \quad \text{and} \quad [v \downarrow m](\gamma) = v(m\gamma), \quad \gamma \in \mathbb{Z}^d.$$

For a given data $v \in l(\mathbb{Z}^d)$, the *one-level framelet decomposition* employing a filter bank $\{a; b_1, \dots b_s\}$ produces a set $\{v_0; w_1, \dots, w_s\}$ of framelet coefficient sequences:

$$v_0 := 2^{-d/2}\mathcal{T}_a v, \quad w_\iota = 2^{-d/2}\mathcal{T}_{b_\iota} v, \quad \iota = 1, \dots, s,$$

while the *one-level framelet reconstruction* with $\{v_0; w_1, \dots, w_s\}$ outputs a reconstruction data sequence

$$\widetilde{v} := 2^{-d/2}\left(\mathcal{S}_a v_0 + \sum_{\iota=1}^{s} \mathcal{S}_{b_\iota} w_\iota\right).$$

Iteratively employing the one-level framelet decomposition (reconstruction) with $v_J := v$ gives the *mult-level discrete framelet transforms* (DFmT):

$$\text{Decomposition:} \quad v_{j-1} = 2^{-d/2}\mathcal{T}_a v_j, \quad w_{j-1;\iota} = 2^{-d/2}\mathcal{T}_{b_\iota} v_j, \quad \iota = 1, \dots, s, \quad j = J, \dots, 1.$$

$$\text{Reconstruction:} \quad v_j = 2^{-d/2}\left(\mathcal{S}_a v_{j-1} + \sum_{\iota=1}^{s} \mathcal{S}_{b_\iota} w_{j-1;\iota}\right), \quad j = 1, \dots, J.$$

See Fig. 4 for the illustration of the multi-level discrete framelet transforms (DFmT) with $J = 2$.

Define filters a_j and $b_{\iota,j}$ for $j \geqslant 1$ by

$$\widehat{a_j}(\xi) := \widehat{a}(\xi)\widehat{a}(2\xi)\cdots\widehat{a}(2^{j-1}\xi) \quad \text{and} \quad \widehat{b_{\iota;j}}(\xi) := \widehat{a_{j-1}}(\xi)\widehat{b_\iota}(2^{j-1}\xi), \quad \iota = 1, \dots, s$$

with the convention that $a_0 := \boldsymbol{\delta}$. That is,

$$a_j = a * (a \uparrow 2) * \cdots * (a \uparrow 2^{j-1}) \quad \text{and} \quad b_{\iota,j} = a_{j-1} * (b_\iota \uparrow 2^{j-1}).$$

Define

$$a_{[j;k]} := a_j(\cdot - k) \quad \text{and} \quad b_{\iota,[j;k]} := b_{\iota,j}(\cdot - k), \quad \iota = 1, \dots, s.$$

Then, the *discrete affine system* associated with the filter bank $\{a; b_1, \dots, b_s\}$ at level J is given by

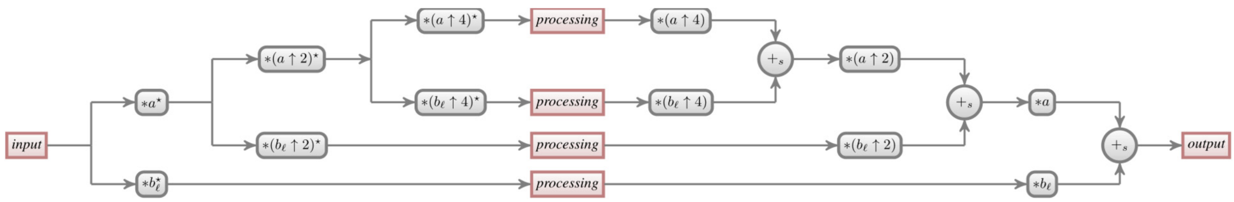


Fig. 5. Undecimated discrete framelet transforms (UDFmT) associated with a filter bank $\{a; b_1, \ldots, b_s\}$. Each box with b_ι runs through $\iota = 1, \ldots, s$ and the circle with $+_s$ sums over all s outputs from boxes with b_ι.

$$\mathrm{DAS}_J(\{a; b_1, \ldots, b_s\}) := \{2^{-J/2} a_{[J;2^J k]} : k \in \mathbb{Z}^d\} \cup \{2^{-j/2} b_{\iota,[j;2^j k]} : k \in \mathbb{Z}^d, \iota = 1, \ldots, s\}_{j=1}^J$$

One can show that ([17, Theorems 2.1 and 2.4]) a filter bank $\{a; b_1, \ldots, b_s\}$ is a tight framelet filter bank, i.e., satisfying (4) if and only if it satisfies

(a) the *perfect reconstruction property*: $\mathcal{S}_a \mathcal{T}_a v + \sum_{\iota=1}^s \mathcal{S}_{b_\iota} \mathcal{T}_{b_\iota} v = 2^d v$ for all $v \in l(\mathbb{Z}^d)$, if and only if it satisfies,

(b) the *energy preservation property*:

$$\|\mathcal{T}_a v\|_2^2 + \sum_{\iota=1}^s \|\mathcal{T}_{b_\iota} v\|_2^2 = 2^d \|v\|_2^2, \quad \forall v \in l_2(\mathbb{Z}^d), \tag{7}$$

if and only if it has,

(c) the *discrete affine tight frame representation*: $v = \sum_{u \in \mathrm{DAS}_J(\{a; b_1, \ldots, b_s\})} \langle v, u\rangle u$ for all $v \in l_2(\mathbb{Z}^d)$ and for all $J \in \mathbb{N}$.

2.3. Fast undecimated discrete framelet transforms

A tight framelet filter bank can be used to (sparsely) represent data sequences through its associated discrete framelet transforms. However, noting that due to $\mathcal{T}_h(v(\cdot - 2n)) = [\mathcal{T}_h v](\cdot - n)$, for a translated version of the input signal, the output framelet coefficient sequence may no longer be a translated version of the original framelet coefficient sequence. In signal/image/video processing, translation invariance property of a discrete framelet transform is very much desirable especially in the scenario of signal denoising/inpainting. To preserve the translation invariance property, in this paper, we consider the more redundant version of DFmT, that is, the *undecimated discrete framelet transforms* (UDFmT):

$$\begin{aligned}
&\textit{Decomposition:} \quad v_{j-1} = v_j * (a^\star \uparrow 2^{J-j}), \quad j = J, \ldots, 1,\\
&\qquad\qquad\qquad w_{j-1;\iota} = v_j * (b_\iota^\star \uparrow 2^{J-j}), \quad \iota = 1, \ldots, s.\\
&\textit{Reconstruction:} \quad v_j = v_{j-1} * (a \uparrow 2^{J-j}) + \sum_{\iota=1}^s w_{j-1;\iota} * (b_\iota \uparrow 2^{J-j}), \quad j = 1, \ldots, J.
\end{aligned}$$

Here $v_J := v$ is an input data sequence. See Fig. 5 for the illustration of UDFmT with $J = 3$.

The multi-level undecimated discrete framelet transforms correspond to a *undecimated discrete affine system*, which is given by

$$\mathrm{UDAS}_J(\{a; b_1, \ldots, b_s\}) := \{a_{[J;k]} : k \in \mathbb{Z}^d\} \cup \{b_{\iota;[j,k]} : k \in \mathbb{Z}^d, \iota = 1, \ldots, s\}_{j=1}^J$$

One can show that the undecimated discrete framelet transforms employing a filter bank $\{a; b_1, \dots, b_s\}$ have the *perfect reconstruction property*, i.e., any input data sequence and its reconstruction data sequence using UDFmT are the same, if and only if it satisfies

(a) the *partition of unity condition*:

$$|\widehat{a}(\xi)|^2 + \sum_{\iota=1}^{s} |\widehat{b_\iota}(\xi)|^2 = 1, \quad \xi \in \mathbb{R}^d, \tag{8}$$

if and only if it has,

(b) the *undecimated discrete affine tight frame representation*: $v = \sum_{u \in \mathrm{UDAS}_J(\{a; b_1,\dots,b_s\})} \langle v, u\rangle u$ for all $v \in l_2(\mathbb{Z}^d)$ and for all $J \in \mathbb{N}$.

3. A tailor-made 3D directional Haar tight and semi-tight framelet

We are ready to introduce the 3D directional Haar tight and semi-tight framelet (3DHTF and 3DHSTF) systems. The 3DHSTF is called *semi-tight* since it is very close to the tight framelet system 3DHTF with certain modifications. First, we have the following theorem from [19] that is motivated by the 2D directional Haar tight framelet (2DHTF) constructed in [28].

Theorem 1. *Let $a^H := 2^{-d} \sum_{\gamma \in \{0,1\}^d} \boldsymbol{\delta}_\gamma$ be the d-dimension Haar low-pass filter. Define the high-pass filters $b_1, \dots, b_s$ with $s := \binom{2^d}{2} = 2^{d-1}(2^d - 1)$ by $b_\iota := b_{\iota_1,\iota_2} := 2^{-d}(\boldsymbol{\delta}_{\gamma_{\iota_1}} - \boldsymbol{\delta}_{\gamma_{\iota_2}})$ and $1 \leqslant \iota_1 < \iota_2 \leqslant 2^d$, where we label the 2^d vertices in $\{0,1\}^d$ as $\{0,1\}^d = \{\gamma_{\iota_1}, \dots, \gamma_{\iota_{2^d}}\}$ and $\iota = \frac{(2^{d+1} - \iota_1)(\iota_1 - 1)}{2} + \iota_2 - \iota_1$. Then $\{a^H; b_1, \dots, b_s\}$ is a tight framelet filter bank such that all the high-pass filters $b_1, \dots, b_s$ have directionality and exhibit $\frac{1}{2}(3^d - 1)$ directions in dimension d. The functions ϕ and $\psi_1, \dots, \psi_s$ associated with $\{a^H; b_1, \dots, b_s\}$ is a compactly supported d-dimension directional Haar tight framelet in $L_2(\mathbb{R}^d)$ with*

$$\phi = \chi_{[0,1]^d}, \ \psi_\iota = \chi_{[0,\frac{1}{2}]^d}(\cdot - \frac{\gamma_{\iota_1}}{2}) - \chi_{[0,\frac{1}{2}]^d}(\cdot - \frac{\gamma_{\iota_2}}{2})$$

for $\iota = 1, \dots, s$, where $\chi_{\mathbb{A}}$ is the characteristic function of $\mathbb{A}$ such that $\chi_{\mathbb{A}}(x) = 1$ if $x \in \mathbb{A}$ and $\chi_{\mathbb{A}}(x) = 0$ if $x \notin \mathbb{A}$ for a set $\mathbb{A} \subseteq \mathbb{R}^d$.

In [19], the proof of the tightness in Theorem 1 is based on the proof of (4) from a geometric point of view. We now provide an alternative algebraic proof to show the tightness of the directional Haar tight framelets in Theorem 1 from the viewpoint of the energy preservation property of the discrete framelet transforms in (7).

Proof of Theorem 1. Because all the filters in Theorem 1 are supported inside $\{0,1\}^d$ and noting that $\mathcal{T}_h v = 2^d \sum_k v(k + 2\cdot)\overline{h(k)}$, the d-dimensional discrete framelet transform (decomposition) using the filter bank $\{a; b_1, \dots, b_s\}$ in Theorem 1 can be simply implemented by applying the discrete framelet transform acting on data supported on each disjoint $\{0,1\}^d + 2k$, $k \in \mathbb{Z}^d$, where $\{0,1\}^d$ is the set of all vertices of the unit cube $[0,1]^d$. We now exam the framelet coefficient sequences $\mathcal{T}_h v$ for $h \in \{a; b_1, \dots, b_s\}$. For simplicity, we list the vertices in $\{0,1\}^d$ as $\{\gamma_1, \dots, \gamma_{2^d}\} = \{0,1\}^d$ and assume that the data value of v at the point $\gamma_j + 2k$ is $x_j \in \mathbb{R}$. Then all the high-pass filters in Theorem 1 are given by $b_\ell := \pm 2^{-d}(\boldsymbol{\delta}_{\gamma_j} - \boldsymbol{\delta}_{\gamma_k})$ with $1 \leqslant j < k \leqslant 2^d$. The framelet coefficient sequence $\mathcal{T}_{b_\iota} v$ produced by this high-pass filter is simply $\pm(x_j - x_k)$. The coefficient sequence $\mathcal{T}_{a^H} v$ produced by the Haar low-pass filter a^H in Theorem 1 is simply $(x_1 + \cdots + x_{2^d})$. Hence, the total squared energy of all the framelet coefficient sequences is

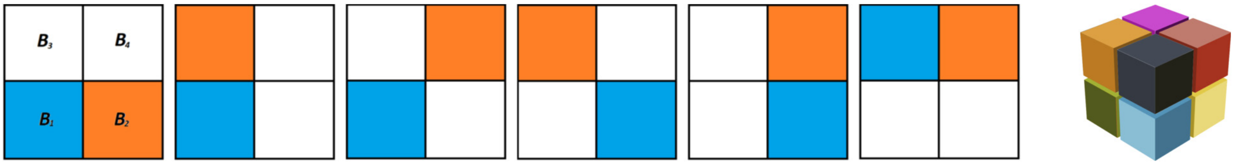


Fig. 6. The 2-Dimensional directional Haar tight framelet (2DHTF) system is generated from 6 framelet functions $\psi_1, \dots, \psi_6$ supported on the unit square $[0,1]^2$. Left to Right (First 6 squares): $\psi^1, \dots, \psi^6$. The unit square is split to 4 sub-blocks $B_1, \dots, B_4$. Each colored sub-block represents either 1 (blue) or -1 (orange) of the function value. White blocks mean 0 function value. The 6 framelet functions clearly cover the directions of $0°$, $90°$, and $\pm 45°$. The last 3D cube: The unit 3D cube $[0,1]^3$ evenly divided to 8 sub-cubes $C_1, \dots, C_8$ and it is the support of the 28 framelet generating functions $\psi_1, \dots, \psi_{28}$ for the 3-dimensional directional Haar tight framelet (3DHTF) system. Each function $\psi_i = \chi_{C_{i_1}} - \chi_{C_{i_2}}$, $1 \leqslant i_1 < i_2 \leqslant 8$ and $i = \frac{(16-i_1)(i_1-1)}{2} + i_2 - i_1$, of the 28 functions is supported on two sub-cubes C_{i_1}, C_{i_2} selected from the 8 sub-cubes. Note that $\binom{8}{2} = 28$. See Theorem 1 for more details. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

$$(x_1 + \cdots + x_{2^d})^2 + \sum_{1 \leqslant j < k \leqslant 2^d} (x_j - x_k)^2 .$$

Noting that $(x_1 + \cdots + x_{2^d})^2 = x_1^2 + \cdots + x_{2^d}^2 + \sum_{1 \leqslant j<k \leqslant 2^d} 2x_jx_k$ and $(x_j - x_k)^2 = (x_j^2 + x_k^2) - 2x_jx_k$, we conclude that the total squared energy of all the framelet coefficient sequences is

$$\begin{aligned}\sum_{h \in \{a; b_1, \dots, s\}} \|\mathcal{T}_h v\|_2 &= (x_1 + \cdots + x_{2^d})^2 + \sum_{1 \leqslant j < k \leqslant 2^d} (x_j - x_k)^2 \\ &= (x_1^2 + \cdots + x_{2^d}^2) + \sum_{1 \leqslant j < k \leqslant 2^d} (x_j^2 + x_k^2) \\ &= 2^d (x_1^2 + \cdots + x_{2^d}^2) = 2^d \|v\|_2^2,\end{aligned}$$

which proves the energy preservation property of the discrete framelet transforms in (7). Hence, the filter bank $\{a; b_1, \dots, b_s\}$ in Theorem 1 must be a tight framelet filter bank. Their associated functions $\phi, \psi_1, \dots, \psi_s$ can be easily deduced according to (5). $\square$

(1) When $d = 1$, Theorem 1 simply gives the standard Haar orthogonal wavelet filter bank $\mathrm{DHTF}_1 := \{a^H; b\}$ with

$$a^H = \frac{1}{2}(\boldsymbol{\delta}_0 + \boldsymbol{\delta}_1) \text{ and } b = \frac{1}{2}(\boldsymbol{\delta}_0 - \boldsymbol{\delta}_1).$$

(2) When $d = 2$, Theorem 1 recovers the 2D directional Haar tight framelet filter bank $\mathrm{DHTF}_2 := \{a^H; b_1, \dots, b_6\}$ in [28, (3.5)] with $a^H = \frac{1}{4}(\boldsymbol{\delta}_{(0,0)} + \boldsymbol{\delta}_{(0,1)} + \boldsymbol{\delta}_{(1,0)} + \boldsymbol{\delta}_{(1,1)})$ and

$$\begin{aligned} b_1 &= \frac{1}{4}(\boldsymbol{\delta}_{(0,0)} - \boldsymbol{\delta}_{(0,1)}), & b_2 &= \frac{1}{4}(\boldsymbol{\delta}_{(0,0)} - \boldsymbol{\delta}_{(1,0)}), & b_3 &= \frac{1}{4}(\boldsymbol{\delta}_{(0,0)} - \boldsymbol{\delta}_{(1,1)}), \\ b_4 &= \frac{1}{4}(\boldsymbol{\delta}_{(0,1)} - \boldsymbol{\delta}_{(1,0)}), & b_5 &= \frac{1}{4}(\boldsymbol{\delta}_{(0,1)} - \boldsymbol{\delta}_{(1,1)}), & b_6 &= \frac{1}{4}(\boldsymbol{\delta}_{(1,0)} - \boldsymbol{\delta}_{(1,1)}). \end{aligned}$$

See Fig. 6 for their associated framelet functions $\psi_1, \dots, \psi_6$.

(3) When $d = 3$, Theorem 1 gives the following 3D directional Haar tight framelet filter bank $\mathrm{DHTF}_3^1 := \{a^H; b_1, \dots, b_{28}\}$ with

$$a^H = \frac{1}{8}(\boldsymbol{\delta}_{(0,0,0)} + \boldsymbol{\delta}_{(0,0,1)} + \boldsymbol{\delta}_{(0,1,0)} + \boldsymbol{\delta}_{(0,1,1)} + \boldsymbol{\delta}_{(1,0,0)} + \boldsymbol{\delta}_{(1,0,1)} + \boldsymbol{\delta}_{(1,1,0)} + \boldsymbol{\delta}_{(1,1,1)}),$$

and

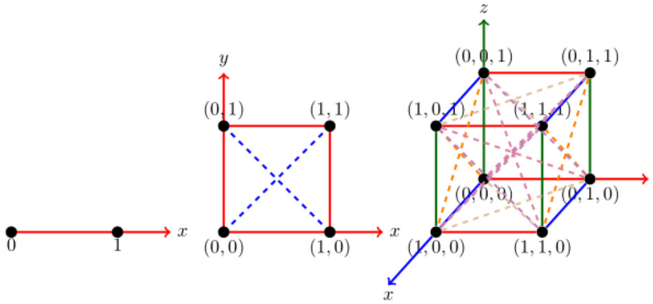


Fig. 7. Left to Right: Directional Haar tight framelet filter banks in $d = 1, 2, 3$ respectively, where each line connecting two vertices $\gamma_1, \gamma_2 \in \{0,1\}^d$ represents a high-pass filter $b := 2^{-d}(\boldsymbol{\delta}_{\gamma_1} - \boldsymbol{\delta}_{\gamma_2})$.

$$
\begin{aligned}
&b_1 = \frac{1}{8}(\boldsymbol{\delta}_{(0,0,0)} - \boldsymbol{\delta}_{(0,0,1)}), && b_2 = \frac{1}{8}(\boldsymbol{\delta}_{(0,0,0)} - \boldsymbol{\delta}_{(0,1,0)}), && b_3 = \frac{1}{8}(\boldsymbol{\delta}_{(0,0,0)} - \boldsymbol{\delta}_{(0,1,1)}), && b_4 = \frac{1}{8}(\boldsymbol{\delta}_{(0,0,0)} - \boldsymbol{\delta}_{(1,0,0)}),\\
&b_5 = \frac{1}{8}(\boldsymbol{\delta}_{(0,0,0)} - \boldsymbol{\delta}_{(1,0,1)}), && b_6 = \frac{1}{8}(\boldsymbol{\delta}_{(0,0,0)} - \boldsymbol{\delta}_{(1,1,0)}), && b_7 = \frac{1}{8}(\boldsymbol{\delta}_{(0,0,0)} - \boldsymbol{\delta}_{(1,1,1)}), && b_8 = \frac{1}{8}(\boldsymbol{\delta}_{(0,0,1)} - \boldsymbol{\delta}_{(0,1,0)}),\\
&b_9 = \frac{1}{8}(\boldsymbol{\delta}_{(0,0,1)} - \boldsymbol{\delta}_{(0,1,1)}), && b_{10} = \frac{1}{8}(\boldsymbol{\delta}_{(0,0,1)} - \boldsymbol{\delta}_{(1,0,0)}), && b_{11} = \frac{1}{8}(\boldsymbol{\delta}_{(0,0,1)} - \boldsymbol{\delta}_{(1,0,1)}), && b_{12} = \frac{1}{8}(\boldsymbol{\delta}_{(0,0,1)} - \boldsymbol{\delta}_{(1,1,0)}),\\
&b_{13} = \frac{1}{8}(\boldsymbol{\delta}_{(0,0,1)} - \boldsymbol{\delta}_{(1,1,1)}), && b_{14} = \frac{1}{8}(\boldsymbol{\delta}_{(0,1,0)} - \boldsymbol{\delta}_{(0,1,1)}), && b_{15} = \frac{1}{8}(\boldsymbol{\delta}_{(0,1,0)} - \boldsymbol{\delta}_{(1,0,0)}), && b_{16} = \frac{1}{8}(\boldsymbol{\delta}_{(0,1,0)} - \boldsymbol{\delta}_{(1,0,1)}),\\
&b_{17} = \frac{1}{8}(\boldsymbol{\delta}_{(0,1,0)} - \boldsymbol{\delta}_{(1,1,0)}), && b_{18} = \frac{1}{8}(\boldsymbol{\delta}_{(0,1,0)} - \boldsymbol{\delta}_{(1,1,1)}), && b_{19} = \frac{1}{8}(\boldsymbol{\delta}_{(0,1,1)} - \boldsymbol{\delta}_{(1,0,0)}), && b_{20} = \frac{1}{8}(\boldsymbol{\delta}_{(0,1,1)} - \boldsymbol{\delta}_{(1,0,1)}),\\
&b_{21} = \frac{1}{8}(\boldsymbol{\delta}_{(0,1,1)} - \boldsymbol{\delta}_{(1,1,0)}), && b_{22} = \frac{1}{8}(\boldsymbol{\delta}_{(0,1,1)} - \boldsymbol{\delta}_{(1,1,1)}), && b_{23} = \frac{1}{8}(\boldsymbol{\delta}_{(1,0,0)} - \boldsymbol{\delta}_{(1,0,1)}), && b_{24} = \frac{1}{8}(\boldsymbol{\delta}_{(1,0,0)} - \boldsymbol{\delta}_{(1,1,0)}),\\
&b_{25} = \frac{1}{8}(\boldsymbol{\delta}_{(1,0,0)} - \boldsymbol{\delta}_{(1,1,1)}), && b_{26} = \frac{1}{8}(\boldsymbol{\delta}_{(1,0,1)} - \boldsymbol{\delta}_{(1,1,0)}), && b_{27} = \frac{1}{8}(\boldsymbol{\delta}_{(1,0,1)} - \boldsymbol{\delta}_{(1,1,1)}), && b_{28} = \frac{1}{8}(\boldsymbol{\delta}_{(1,1,0)} - \boldsymbol{\delta}_{(1,1,1)}).
\end{aligned}
$$

See Fig. 6 the 3D unit cube for the support of their associated framelet functions $\psi_1, \ldots, \psi_{28}$.

Fig. 7 illustrates the high-pass filters of DHTF filter banks $\mathrm{DHTF}_1, \mathrm{DHTF}_2$, DHTF_3^1, respectively.

As discussed, the pMRI coil data are degenerated with noise and aliasing artifacts. For such tasks, redundant representation systems are more favor since it provides more information for data recovery. Thus, it is useful to use the UDFmT. In such a case, we only need the filter bank to satisfy the partition of unity condition in (8). However, the more the number of filters in a filter bank, the less efficiency of the UDFmT. Hence, we further simplify the filter bank DHTF_3^1. In terms of directionality, there are many filters in DHTF_3^1 characterizing the same directional property. For example, the filters in $\{b_1, b_{14}, b_{23}, b_{28}\}$ represent the same z-direction (vertical), the filters in $\{b_2, b_9, b_{25}, b_{27}\}$ represent the same y-direction, and so on so forth. Here $b_1 = \frac{1}{8}(\boldsymbol{\delta}_{(0,0,0)} - \boldsymbol{\delta}_{(0,0,1)}), b_{14} = \frac{1}{8}(\boldsymbol{\delta}_{(0,1,0)} - \boldsymbol{\delta}_{(0,1,1)})$, and others are similarly defined according to Theorem 1 (see Fig. 8 for the illustration). Consequently, the 28 high-pass filters in DHTF_3^1 can be regrouped to 13 filters in a simplified filter bank

$$
\mathrm{DHTF}_3^2 := \{a^H; b_x, b_y, b_z, b_{xy}, b_{x,y}, b_{xz}, b_{x,z}, b_{yz}, b_{y,z}, b_{xyz}, b_{xy,z}, b_{x,yz}, b_{xz,y}\}
$$

(see Fig. 8 left) with

$$
\begin{aligned}
&b_x = \frac{1}{4}(\boldsymbol{\delta}_{(1,0,0)} - \boldsymbol{\delta}_{(0,0,0)}), && b_y = \frac{1}{4}(\boldsymbol{\delta}_{(0,1,0)} - \boldsymbol{\delta}_{(0,0,0)}), b_{x,y} = \frac{\sqrt{2}}{8}(\boldsymbol{\delta}_{(1,0,0)} - \boldsymbol{\delta}_{(0,1,0)}),\\
&b_{xy} = \frac{\sqrt{2}}{8}(\boldsymbol{\delta}_{(1,1,0)} - \boldsymbol{\delta}_{(0,0,0)}), && b_z = \frac{1}{4}(\boldsymbol{\delta}_{(0,0,1)} - \boldsymbol{\delta}_{(0,0,0)}), b_{xz} = \frac{\sqrt{2}}{8}(\boldsymbol{\delta}_{(1,0,1)} - \boldsymbol{\delta}_{(0,0,0)}),\\
&b_{x,z} = \frac{\sqrt{2}}{8}(\boldsymbol{\delta}_{(1,0,0)} - \boldsymbol{\delta}_{(0,0,1)}), && b_{yz} = \frac{\sqrt{2}}{8}(\boldsymbol{\delta}_{(0,1,1)} - \boldsymbol{\delta}_{(0,0,0)}), b_{y,z} = \frac{\sqrt{2}}{8}(\boldsymbol{\delta}_{(0,1,0)} - \boldsymbol{\delta}_{(0,0,1)}),\\
&b_{xyz} = \frac{1}{8}(\boldsymbol{\delta}_{(1,1,1)} - \boldsymbol{\delta}_{(0,0,0)}), && b_{xy,z} = \frac{1}{8}(\boldsymbol{\delta}_{(1,1,0)} - \boldsymbol{\delta}_{(0,0,1)}), b_{x,yz} = \frac{1}{8}(\boldsymbol{\delta}_{(1,0,0)} - \boldsymbol{\delta}_{(0,1,1)}),
\end{aligned}
$$

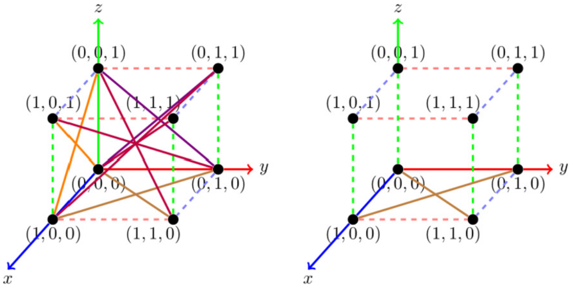


Fig. 8. The 3D directional Haar tight framelet filter banks $\mathrm{DHTF}_3^2 := \{a^H; b_x, b_y, b_z, b_{xy}, b_{x,y}, b_{xz}, b_{x,z}, b_{yz}, b_{y,z}, b_{xyz}, b_{xy,z}, b_{x,yz}, b_{xz,y}\}$ (solid edges on left), and the 3D directional Haar semi-tight framelet filter banks $3\mathrm{DHSTF} := \{a^H; b_x, b_y, b_{xy}, b_{x,y}, b_{\mathrm{aux}}\}$ (solid edges on right, b_{aux} is not shown). Each line connecting two vertices $\gamma_1, \gamma_2 \in \{0,1\}^d$ represents a high-pass filter $b := c(\boldsymbol{\delta}_{\gamma_1} - \boldsymbol{\delta}_{\gamma_2})$ for some constant c.

$$b_{xz,y} = \frac{1}{8}(\boldsymbol{\delta}_{(1,0,1)} - \boldsymbol{\delta}_{(0,1,0)}).$$

Note that the filter bank DHTF_3^2 satisfies the partition of unity condition in (8).

As pointed out in Section 1, for the output framelet coefficient sequences, information involving the z-filters, i.e., those b_z, b_{xz}, b_{xyz}, etc., are actually 'bad' features for the 3D framelet regularization. They represent local contrast discrepancy between coil images and are *not* sparse features suitable for the regularization process. More precisely, taking the coil images in Fig. 1 for example, they can be stacked as a $512 \times 512 \times 4$ data. When fed into the UDFmT decomposition with $J = 1$, we obtain one low-pass framelet coefficient sequence with respect to a^H and 13 high-pass framelet coefficient sequences of size $512 \times 512 \times 4$ with respect to those high-pass filters. Among those 13 high-pass framelet coefficient sequences, only four of them with respect to the high-pass filters $b_x, b_y, b_{xy}, b_{x,y}$ in DHTF_3^2 are sparse; see Fig. 3 (a)–(d) for image slices (512×512) from those high-pass framelet coefficient sequences. The other framelet coefficient sequences involving the z-filters are *not* sparse at all and are similar to those shown in Fig. 3 (e) and (f). Same phenomena happen for further decomposition using UDFmT with high level $J > 1$. Involving such 'non-sparse' features in our regularization process no doubt damages our purpose of sparse regularization. To regularize the framelet coefficients with true sparsity, we utilize this prior information and neglect the high-pass framelet coefficients involving the z-filters. Hence, further reduction of those filters gives us an even simplified filter bank

$$3\mathrm{DHSTF} := \{a^H; b_x, b_y, b_{xy}, b_{x,y}, b_{\mathrm{aux}}\}$$

(see Fig. 8 right) with an auxiliary filter b_{aux} defined by

$$\begin{aligned} b_{\mathrm{aux}} := \Bigg\{ & \frac{1}{2}\boldsymbol{\delta}_{(0,0,0)} - \frac{1}{16}\left(\boldsymbol{\delta}_{(0,0,1)} + \boldsymbol{\delta}_{(0,0,-1)}\right) - \frac{1}{32}\left(\boldsymbol{\delta}_{(1,0,-1)} + \boldsymbol{\delta}_{(-1,0,1)} + \boldsymbol{\delta}_{(0,1,-1)} + \boldsymbol{\delta}_{(0,-1,1)} + \boldsymbol{\delta}_{(1,0,1)}\right. \\ & \left. + \boldsymbol{\delta}_{(-1,0,-1)} + \boldsymbol{\delta}_{(0,1,1)} + \boldsymbol{\delta}_{(0,-1,-1)}\right) - \frac{1}{64}\left(\boldsymbol{\delta}_{(1,1,-1)} + \boldsymbol{\delta}_{(-1,-1,1)} + \boldsymbol{\delta}_{(1,-1,1)} + \boldsymbol{\delta}_{(1,-1,1)} + \boldsymbol{\delta}_{(-1,1,-1)}\right. \\ & \left. + \boldsymbol{\delta}_{(-1,1,1)} + \boldsymbol{\delta}_{(1,-1,1)} + \boldsymbol{\delta}_{(1,1,1)} + \boldsymbol{\delta}_{(-1,-1,-1)}\right) \Bigg\}, \end{aligned}$$

to fulfill the partition of unity condition in (8). That is, the filter b_{aux} is deduced from

$$\widehat{b_{\mathrm{aux}}} = 1 - \left(|\widehat{a^H}|^2 + |\widehat{b_x}|^2 + |\widehat{b_y}|^2 + |\widehat{b_{x,y}}|^2 + |\widehat{b_{xy}}|^2\right).$$

Since the decomposition and reconstruction filters involving b_{aux} are different in the UDFmT and the filter bank is very close to a tight framelet filter bank, we call the filter bank $\{a^H; b_x, b_y, b_{xy}, b_{x,y}, b_{\mathrm{aux}}\}$ a 3-dimensional directional Haar semi-tight framelet (3DHSTF) filter bank. Indeed, the decomposition filter

Table 1
Computation complexity in terms of multiplications and additions, and memory storage of UDFmT decomposition and reconstruction using DHTF_3^2 and 3DHSTF with $J = 1$.

UDFmT	Decomposition		Reconstruction		Memory
3DHSTF	$+$	$\times$	$+$	$\times$	size
a^H	7	1	7	1	1
b_x	1	1	2	1	1
b_y	1	1	2	1	1
$b_{x,y}$	1	1	2	1	1
b_{xy}	1	1	2	1	1
b_{aux}	18	4	1	0	1
Total	29	9	16	5	6
DHTF_3^2	$+$	$\times$	$+$	$\times$	size
a^H	7	1	7	1	1
13 b_ι	13	13	26	13	13
Total	20	14	33	14	14

bank is $\{a^H; b_x, b_y, b_{xy}, b_{x,y}, b_{\mathrm{aux}}\}$ while the reconstruction filter bank is $\{a^H; b_x, b_y, b_{xy}, b_{x,y}, \delta\}$. Using such a 3DHSTF filter bank, we have a very simple and efficient fast UDFmT. The pseudo code can be found in Algorithms 1 and 2, where $\circledast$ denotes circular convolution and the input data employ periodic extension.

Algorithm 1 *(UDFmT: Decomposition with 3DHSTF).*

1. *Input: a 3D data v_J with $J \in \mathbb{N}$ and the filter bank* $3\mathrm{DHSTF} = \{a^H; b_x, b_y, b_{xy}, b_{x,y}, b_{\mathrm{aux}}\}$.
2. *For $j = J, J-1, \ldots, 1$,*
   (a) $v_{j-1} \leftarrow v_j \circledast (a^H \uparrow 2^{J-j})^\star$;
   (b) *For $h \in \{b_x, b_y, b_{x,y}, b_{xy}, b_{\mathrm{aux}}\}$:*
      - $w_{j-1;h} \leftarrow v_j \circledast (h \uparrow 2^{J-j})^\star$;
3. *Output: framelet coefficient sequences: $\{v_0\} \cup \{w_{j,h} : h \in \{b_x, b_y, b_{x,y}, b_{xy}, b_{\mathrm{aux}}\}\}_{j=1}^J$.*

Algorithm 2 *(UDFmT: Reconstruction with 3DHSTF).*

1. *Input: framelet coefficient sequences $\{v_0\} \cup \{w_{j,h} : h \in \{b_x, b_y, b_{x,y}, b_{xy}, b_{\mathrm{aux}}\}\}_{j=1}^J$.*
2. *For $j = 1, 2, \ldots, J$,*
   (a) $v_j \leftarrow v_{j-1} \circledast (a^H \uparrow 2^{J-j})$;
   (b) *For $h \in \{b_x, b_y, b_{x,y}, b_{xy}\}$:*
      - $v_j \leftarrow v_j + w_{j-1;h} \circledast (h \uparrow 2^{J-j})$;
   (c) $v_j \leftarrow v_j + w_{j-1,b_{\mathrm{aux}}}$
3. *Output: a 3D data $v = v_J$.*

Table 1 presents the computational complexity, scaled by the size of the underlying 3D data, in terms of multiplication ($\times$), addition ($+$), and the memory storage requirement of the UDFmT with the 3DHSTF and DHTF_3^2 for $J = 1$. It clearly shows that our 3DHSTF filter bank not only is simpler but also significantly reduces the computational complexity as well as the memory storage requirement.

4. Model and algorithm for the 3D framelet regularization in pMRI reconstruction

In this section, we briefly review the GRAPPA model and the ℓ_1-SPIRiT model that lead to our ℓ_1-W3D model using 3D framelet regularization. We present an algorithm for solving the ℓ_1-W3D model using the ADMM scheme.

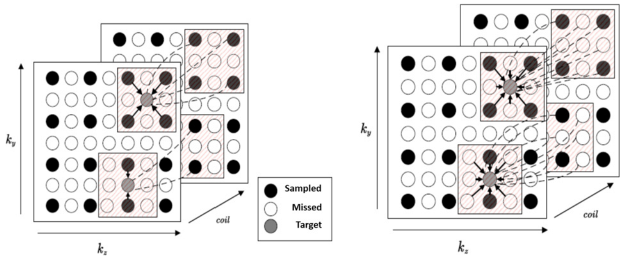


Fig. 9. Reconstruction model by GRAPPA (Left) and SPIRiT (Right) on 3×3 interpolating window. For GRAPPA, in the upper square 2D window, four K-space points (black dots) are known, p coils have $4p$ data to predict the target point (gray dot), and then interpolating kernel $\kappa_\iota^1 \in \mathbb{C}^{4p\times 1}$ is needed; in the lower square window, $2p$ points (black dots) data are collected, then interpolating kernel $\kappa_\iota^2 \in \mathbb{C}^{2p\times 1}$ is needed. For SPIRiT, the data on known and unknown points is fully utilized to predict the target, thus only one template for each coil kernel $\kappa_\iota \in \mathbb{C}^{(9p-1)\times 1}$ is needed. The shift-invariant kernel can be estimated on the ACS data according to the template by the model (9) or (10).

4.1. 3D framelet regularization for pMRI reconstruction

Suppose we have p coil K-space data $g_\iota \in \mathbb{C}^{n\times 1}$, $\iota = 1, \ldots, p$. Here, n is the size of one coil full K-space data. For example, we regard the K-space data for each coil image of 512×512 in Fig. 1 as an $n \times 1$ vector data with $n = 512^2$. It does not mean that the data is vectorized, but simply for the purpose of explaining the models in matrix form. The sampling matrix $\mathcal{P} \in \mathbb{R}^{n\times n}$ is diagonal with 0 and 1 (indicating the corresponding K-space data is skipped or not) at its diagonal elements. The collected data of each coil is denoted by $\mathcal{P}g_\iota$.

For the GRAPPA method, every K-space coefficient of a coil image can be considered as a linear combination of the data within its neighbor and the data from the same local neighbors of the other coils. The interpolation kernel may have different patterns for each coil data and its template is determined by the positions of the collected data with respect to a target point within the interpolation window. For example, in the illustration of two kernels shown in the left image in Fig. 9, one template is four K-space points (black dots) collected in the upper square 2D window of each coil, but another one is only two points known in the lower window for one coil. We denote $\kappa_\iota^i \in \mathbb{C}^{\eta_i p\times 1}$ the interpolation kernel with the i^{th} template for a missing position in the ι^{th} coil, where η_i is the number of the known data in the 2D template around the missing position of g_ι, $\iota = 1, \ldots, p$. The interpolation kernels κ_ι^i are supposed to be shift-invariant and are estimated according to the sampling model by using the ACS data, fully sampled at the region near the center of K-space. For the ι^{th} coil, we construct a matrix D_ι^i row-by-row through collecting $\eta_i p$ known data points of the i^{th} template from ACS K-space data of p coils and denote its corresponding target data as a vector d_ι^i, then the kernel κ_ι^i is estimated by

$$\min_{\kappa_\iota^i} \|D_\iota^i \kappa_\iota^i - d_\iota^i\|_2^2, \quad i = 1, \ldots, \sharp\kappa_\iota;\ \iota = 1, \ldots, p, \tag{9}$$

where $\sharp\kappa_\iota$ is the number of kernels determined by the sampling model for the ι^{th} coil. Once the interpolation kernels κ_ι^i are available, the missing coefficients of the same interpolation template can be predicted by its linear combination.

To reduce the number of interpolation kernels and reconstruct image from arbitrary sampling patterns in the K-space, iterative self-consistent parallel imaging reconstruction method, SPIRiT (see the right image in Fig. 9 for an example), is proposed to estimate exactly one interpolation kernel for each coil [31] and reconstruct the coil images through a 2D wavelet regularization [34]. The data within the cuboid, except for the target position, are all linearly combined together to predict the information in the ℓ_1-SPIRiT method. Suppose the interpolating window for each coil is of size $\eta^1 \times \eta^2$. Then the ι^{th} coil interpolation kernel, denoted by $\kappa_\iota \in \mathbb{C}^{(\eta^1\eta^2 p-1)\times 1}$, is estimated by

$$\min_{\kappa_\iota} \|D_\iota \kappa_\iota - d_\iota\|_2^2, \quad \iota = 1, \dots, p, \tag{10}$$

where D_ι and d_ι are the known data and target interpolated data from the ACS lines, respectively. For the kernel κ_ι from (10), we use $C_\iota \in \mathbb{C}^{n \times np}$ as the matrix representation of κ_ι. Once the p kernels are obtained, the optimization model by ℓ_1-SPIRiT [34] was presented by (1) in Section 1.

We make some remarks here. (i) Compared with the GRAPPA model in (9), the SPIRiT model in (10) reduces the number of interpolation kernels significantly. Though only the use of (10) may not sufficient for pMRI reconstruction, yet by using the sparsity-promoting technique $\|W_{wav}\mathcal{F}_p^{-1}(Qu+g)\|_{1,2}$ with 2D wavelet regularization, the ℓ_1-SPIRiT model in (1) improves the performance of GRAPPA. (ii) Although W_{wav} acts on the 3D data $\mathcal{F}_p^{-1}(Qu+g)$, essentially, it is just a simple stacking of the 2D wavelet transforms of coil data. The correlated information among coils is not taken into account through the ℓ_1-ℓ_2 norm of the 2D wavelet coefficients of each coil.

Each surface coil of a parallel imaging system receives some parts of the information of the target slice, and can be stacked together as 3D data with redundancy. By stacking the coil data, we treat the 3D cuboid data as a whole object so that we could make good use of correlated information and reduce the aliasing artifacts more efficiently. In view of the effectiveness of the ℓ_1-SPIRiT model and the importance of the correlated information among coils, we hence propose the ℓ_1-W3D model in (2) for pMRI reconstruction. When W_{3D} in (2) is our 3DHSTF system in Section 3, we call it ℓ_1-3DHSTF.

4.2. An algorithm for pMRI reconstruction

We elaborate on how to apply alternating direction method of multiplier (ADMM [15]) to solve the ℓ_1-W3D model (2). By introducing an auxiliary variable v, the ℓ_1-W3D can be reformulated as

$$\min_u \frac{1}{2}\|(C-I)(Qu+g)\|_2^2 + \|\Gamma v\|_1 \quad \text{subject to} \quad v = W_{3D}\mathcal{F}_p^{-1}(Qu+g). \tag{11}$$

Consequently, ADMM can be applied to solve the optimization problem (11) via solving several resulting subproblems. First, the augmented Lagrangian function of (11) can be written as

$$\begin{aligned}\mathcal{L}_\rho(u,v,\alpha) :=& \frac{1}{2}\|(C-I)(Qu+g)\|_2^2 + \|\Gamma v\|_1 + \\ & \mathrm{Re}(\alpha^\top(v - W_{3D}\mathcal{F}_p^{-1}(Qu+g))) + \frac{\rho}{2}\|v - W_{3D}\mathcal{F}_p^{-1}(Qu+g)\|_2^2,\end{aligned}$$

where Re takes the real part of a complex number, α is the Lagrange parameter vector, and $\rho > 0$ is a penalty parameter on the linear constraint. Then, the iterative scheme of ADMM can be specified below in (12).

$$\begin{cases} u^{k+1} = \arg\min_u \mathcal{L}_\rho(u, v^k, \alpha^k), \\ v^{k+1} = \arg\min_v \mathcal{L}_\rho(u^{k+1}, v, \alpha^k), \\ \alpha^{k+1} = \alpha^k + \rho(v^{k+1} - W_{3D}\mathcal{F}_p^{-1}(Qu^{k+1}+g)). \end{cases} \tag{12}$$

The convergence of the above iterative scheme is guaranteed under the condition that $\rho > 0$ ([15]). We list u-subproblem and v-subproblem at each iteration for solving (11).

The u-subproblem in (12) can be written as

$$u^{k+1} = \arg\min_u \left\{\frac{1}{2}\|(C-I)(Qu+g)\|_2^2 + \frac{\rho}{2}\|v^k - W_{3D}\mathcal{F}_p^{-1}(Qu+g) + \frac{1}{\rho}\alpha^k\|_2^2\right\}.$$

The minimizer of the above problem is given by solving the following linear system

$$(Q(C-I)^\top(C-I)+\rho I)Qu = (W_{3D}\mathcal{F}_p^{-1}Q)^\top(\rho v^k+\alpha^k) - Q(C-I)^\top(C-I)g - \rho Qg, \tag{13}$$

where $\top$ is the complex conjugate transpose operator. The linear system (13) can be solved by the conjugate gradient method [35]. In our later numerical experiments, three iterations are performed to get an approximate solution of (13).

The v-subproblem in (12) can be written as

$$v^{k+1} = \arg\min_v \frac{1}{\rho}\|\Gamma v\|_1 + \frac{1}{2}\|v - W_{3D}\mathcal{F}_p^{-1}(Qu^{k+1}+g) + \frac{1}{\rho}\alpha^k\|_2^2,$$

whose closed-form solution will be given later.

We next present precisely what Γ is. The estimation of Γ is based on an approach in our previous work [29]. Let $\{w_{j,h} : h \in \{a^H; b_x, b_y, b_{x,y}, b_{xy}, b_{aux}\}\}_{j=1}^J$ be the set of the framelet coefficient sequences obtained from Algorithm 1. Note that each $w_{j,h}$ is a 3D data of size $n_1 \times n_2 \times p$ and can be regarded as $w_{j,h} = \{w_{j,h}^\iota \in \mathbb{C}^{n_1\times n_2} : \iota = 1,\dots,p\}$, where each $w_{j,h}^\iota$ is a 2D image slice of size $n_1 \times n_2$ from $w_{j,h}$ and p is the number of coils. That is, $w_{j,h}$ is from the stacking of $w_{j,h}^\iota$. Then $w_{j,h}^\iota(\mathsf{k}), \mathsf{k} = (k_1, k_2)$ is the framelet coefficient at position k in the ι^{th} slice at the j^{th} level decomposition with respect to the filter $h \in \{a^H; b_x, b_y, b_{x,y}, b_{xy}, b_{aux}\}$. This index $(j, h, \iota, \mathsf{k})$ corresponds to a diagonal entry of Γ, which we denote it as $\gamma_{j,h}^\iota(\mathsf{k})$ and it is defined as follows:

$$\gamma_{j,h}^\iota(\mathsf{k}) = \begin{cases} 0, & h \in \{a^H, b_{\text{aux}}\}, \\ \frac{\lambda\times 8^{J-j}}{\sigma_{j,h}^\iota(\mathsf{k})}, & h \in \{b_x, b_y, b_{x,y}, b_{xy}\}, \end{cases} \tag{14}$$

where the parameter λ is set by hand, $\sigma_{j,h}^\iota(\mathsf{k})$ is the average of the absolute value of the 3×3 neighbor coefficients around position k of $w_{j,h}^\iota$, the number 8 in $\lambda \times 8^{J-j}$ comes from that after low-pass filtering by a^H, the energy of the low-pass filtered framelet coefficient sequence is reduced to 1/8th. In our numerical experiments, UDFmTs are utilized with $J = 2$ and $\gamma_{j,h}^\iota(\mathsf{k})$ only updates 3 times in the first 10 iterations (see Algorithm 3).

For a vector v in the v-problem, define the shrinkage operator $y = \text{shrink}_{\Gamma/\rho}(v)$ by

$$y_{j,h}^\iota(\mathsf{k}) = \frac{v_{j,h}^\iota(\mathsf{k})}{|v_{j,h}^\iota(\mathsf{k})|}\max\left\{|v_{j,h}^\iota(\mathsf{k})| - \frac{\gamma_{j,h}^\iota(\mathsf{k})}{\rho}, 0\right\},$$

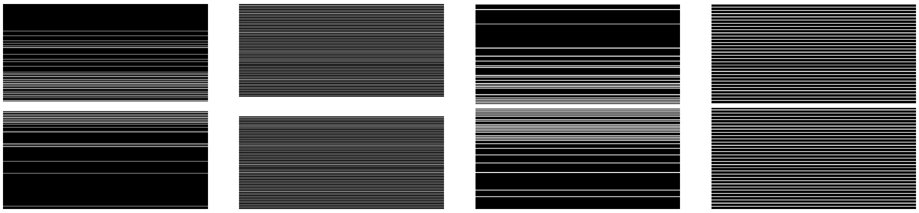
where the index $(j, h, \iota, \mathsf{k})$ is with respect to a diagonal entry of Γ indicated as above. Then, the solution of the v-subproblem can be obtained as follows:

$$v^{k+1} = \text{shrink}_{\Gamma/\rho}\left(W_{3D}\mathcal{F}_p^{-1}(Qu^{k+1}+g) - \frac{1}{\rho}\alpha^k\right). \tag{15}$$

The pMRI reconstruction algorithm for our ℓ_1-W3D model can then be described as in Algorithm 3.

Algorithm 3 *(ℓ_1-W3D pMRI Reconstruction Algorithm).*

1. *Set $\rho = 1$, $u^1 = g$, $v^1 = W_{3D}\mathcal{F}_p^{-1}(Qu^1 + g)$, $\alpha^1 = 0$;*
2. *For $k = 1, 2, \dots,$*
   (a) *u-problem: Utilize the CG algorithm to compute u^{k+1} in equation (13);*
   (b) *v-problem:*
   - *If $k = 1, 4, 7$, update Γ in formula (14);*
   - *Compute v^{k+1} by the shrinkage operator (15) for every entry of v;*



(a) 512 Random (b) 512 Uniform (c) 256 Random (d) 256 Uniform

Fig. 10. Sampling models for K-space ('white' indicating the corresponding K-space data collected, but 'black' not). (a) The 512×512 random sampling matrix of 15% K-space data with 24 ACS lines; (b) The 512×512 uniform sampling matrix (one line taken from every four lines) of 32% K-space data with 48 ACS lines; (c) The 256×256 random sampling matrix of 19% K-space data with 6 ACS lines; (d) The 256×256 uniform sampling matrix (one line taken from every four lines) of 27% K-space data with 6 ACS lines.

(c) *α-problem: α^{k+1} using* (12)*;*

(d) *Compute the 3D coil images $\tilde{u} = \mathcal{F}_p^{-1}(Qu^{k+1} + g)$ when the stopping condition is satisfied.*

Here, for the stopping condition, in our numerical experiments, we set it as k reaching the maximal number of iterations 25.

Note that $\tilde{u}$ is a 3D cuboid data and can be regarded as $\tilde{u} = \{\tilde{u}_\iota \in \mathbb{C}^{n_1\times n_2} : \iota = 1,\dots,p\}$. To get a final reconstruction image from our ℓ_1-W3D pMRI reconstruction algorithm, we use the real domain SoS image of the observed coil images $\tilde{u}_\iota$ by $\tilde{u}_{sos}(\mathsf{k}) = \left(\sum_{\iota=1}^{p} |\tilde{u}_\iota(\mathsf{k})|^2\right)^{\frac{1}{2}}$, where $\mathsf{k} = (k_1, k_2) \in \{1, 2, \dots, n_1\}\times\{1, 2, \dots, n_2\}$.

5. Numerical experiments

In this section, we illustrate the effectiveness of our proposed ℓ_1-3DHSTF model (2) for the pMRI reconstruction in comparison with the well-known model ℓ_1-SPIRiT [34].

In our experiments, we adopt four sampling models of the K-space data in the phase-encoding direction on the Cartesian coordinate that are shown in Fig. 10. Two pseudo random sampling models in Fig. 10(a) and 10(c) collect about 15% and 19% K-space data with 24 and 6 ACS lines (fully sampling), respectively. Two uniform sampling models in Fig. 10(b) and 10(d) by taking one line data from every four lines, are about 32% and 27% K-space data with 48 and 6 ACS lines, respectively. With these sampling models, both the ℓ_1-SPIRiT method and our proposed ℓ_1-3DHSTF method are using the calibration kernel of size 5×5 for each coil K-space data to reconstruct an image from the coil images. The source code of ℓ_1-SPIRiT method was downloaded from the website of one of the authors.[2] In our proposed Algorithm 3 for ℓ_1-3DHSTF model, the number of iterations for CG is set to be 3 and the number of iterations is set to be 25.

Section 5.1 presents the results of the pMRI reconstruction from phantom MR coil images acquired by an MRI machine while Section 5.2 present the results of the pMRI reconstruction from in-vivo medical MR coil images.

5.1. MRI phantoms

In this subsection, four phantom MR images of each slice from a 3T MRI system (Tim Trio, Siemens, Erlangen, Germany) are the T_2-weighted images acquired by a turbo spin-echo sequence. The detailed imaging parameters are set as follows: field of view = 256×256 mm^2, image matrix size = 512×512, slice thicknesses = 3 mm, flip angle = 180 degree, repetition time = 4000 ms, echo time = 71 ms, echo train length = 11, and number of excitation = 1. For these phantom MR images, the random sampling model in Fig. 10(a) and the uniform sampling model in Fig. 10(b) will be applied for these images to test

[2] The code is available at: http://people.eecs.berkeley.edu/~mlustig/Software.html.

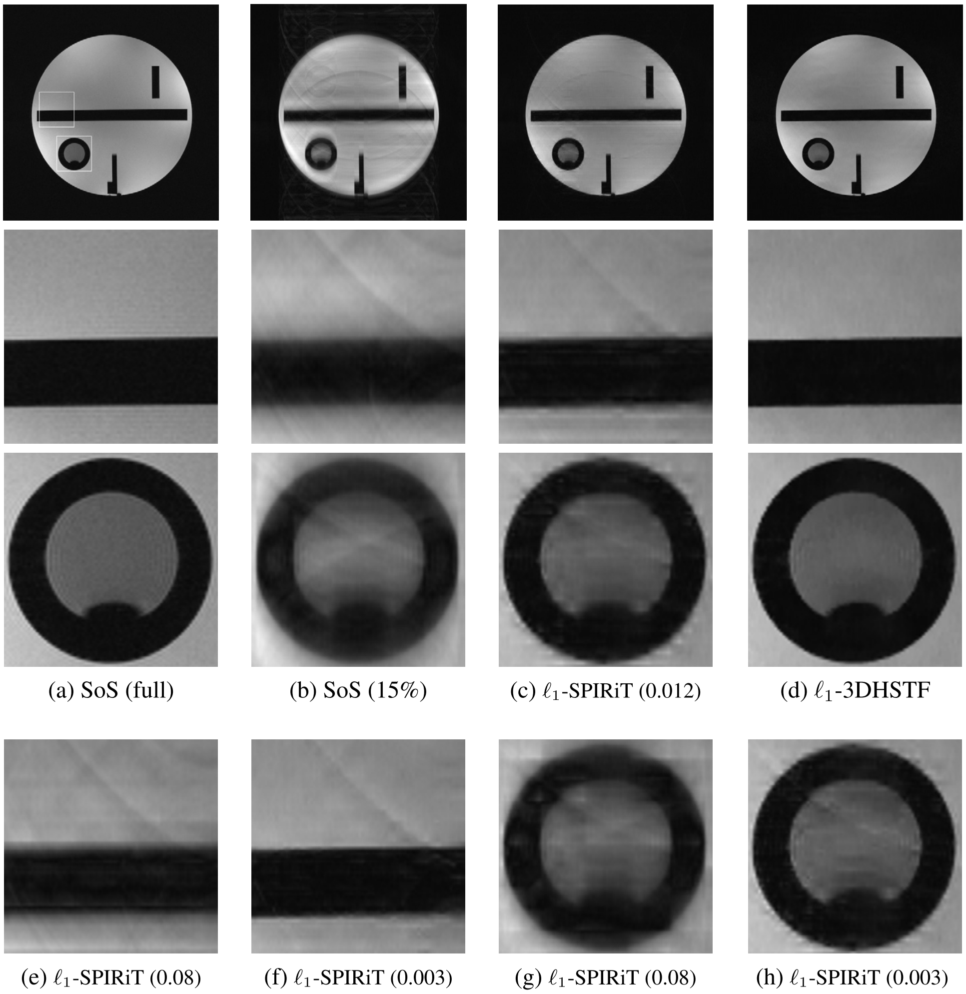
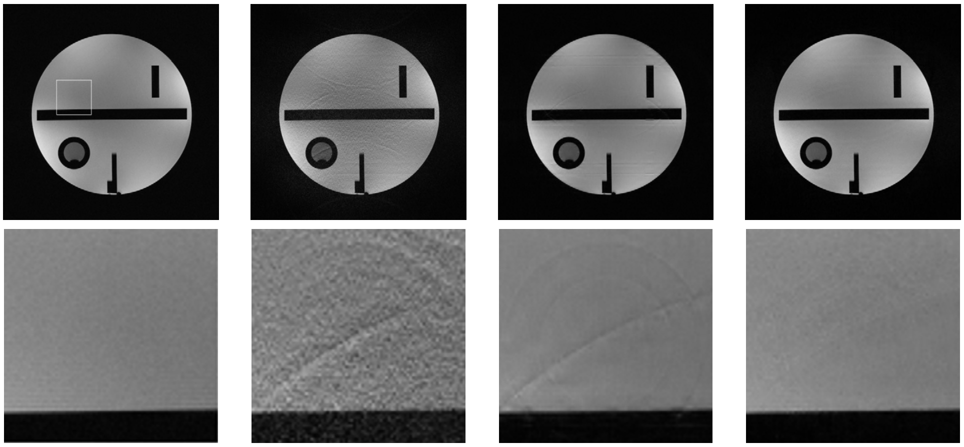


Fig. 11. Reconstruction results on 15% K-space data by the sampling matrix in Fig. 10 (a). (a) SoS image of the full K-space with zoom-in parts; (b) SoS image of the 15% K-space data; (c) ℓ_1-SPIRiT [34] with parameter 0.012; (d) the ℓ_1-3DHSTF with parameter 0.022; First row is the obtained images while the second and the third row are the corresponding zoom-in parts of the first row images. {(e),(g)} and {(f),(h)} are the zoom-in parts of same positions by ℓ_1-SPIRiT [34] with different parameter 0.08 and 0.003, respectively.

the ℓ_1-SPIRiT method and the ℓ_1-3DHSTF method in the pMRI reconstruction. The SoS image of the full K-space data from the four phantom MR images is considered as a reference image and is shown in Fig. 11(a).

We first present the results using the random sampling model. Fig. 11(b) is the SoS image of the coil images obtained by applying the inverse discrete Fourier transform for the collected K-space data with zero-padding for missing data. We can clearly see aliasing artifacts and blurred edges in this image. The image in Fig. 11(c) is the result from the ℓ_1-SPIRiT method using the default settings in the source code of ℓ_1-SPIRiT algorithm except that the calibration kernel is size of 5×5, and the regularization parameter λ is set to be 0.012 after an extensive trial-and-error searching the best one. The reconstruction image by the ℓ_1-3DHSTF method with regularization parameter $\lambda = 0.022$ is shown in Fig. 11(d). Clearly, aliasing artifacts appeared in Fig. 11(b) are significantly suppressed by both ℓ_1-SPIRiT and ℓ_1-3DHSTF. However, the aliasing artifacts in the reconstruction image by the ℓ_1-SPIRiT method is more obvious than those in the image by the ℓ_1-3DHSTF method.

To further evaluate the quality of the reconstructions, two regions shown in Fig. 11(a) are zoomed in the second and third rows of Fig. 11. The structural similarity index measure (SSIM) [45] is used for measuring



(a) SoS(full) (b) GRAPPA (c) ℓ_1-SPIRiT (d) ℓ_1-3DHSTF

Fig. 12. Reconstruction results on 32% K-space data by the uniform sampling mode (one line taken from every four lines) with 48 ACS lines in Fig. 10(b). (a) SoS image of the full K-space with zoom-in part; (b) GRAPPA [16]; (c) ℓ_1-SPIRiT method [34] with parameter 0.005; (d) the ℓ_1-3DHSTF method with parameter 0.003. The second row is the zoom-in parts of first row, respectively.

the similarity between two zoom-in images. The higher index value means that the input image is closer to the reference one. For the 'rectangular' region, the zoom-in image in column (d) by the ℓ_1-3DHSTF method preserves the rectangular edges and reduces aliasing artifacts in the smoothed area, which are close to the reference one in (a) and more shaper than that in the zoom-in images in (c) by the ℓ_1-SPIRiT method. The SSIM indexes for the 'rectangular' region are 0.686 and 0.875 by ℓ_1-SPIRiT, and ℓ_1-3DHSTF, respectively. For the 'circle' region, the edges of two circles in the zoom-in image (c) reconstructed by the ℓ_1-SPIRiT method are blurring with ringing artifacts, but the ℓ_1-3DHSTF method in the zoom-in image (d) can remove the artifacts and retrieve the shape of the circle more close to the reference one. The SSIM index by ℓ_1-3DHSTF is 0.884, but it is 0.645 by ℓ_1-SPIRiT. The last row in Fig. 11 is to show the ability of the ℓ_1-SPIRiT method to remove aliasing artifacts and preserve edges by its regularization parameter λ. The values of λ used in ℓ_1-SPIRiT is 0.012 in Fig. 11(c), 0.08 in Fig. 11(e) and 11(g), and 0.003 in Fig. 11(f) and 11(h). We see that the aliasing artifacts caused by the downsampling operation appeared in all images and can not be removed by using larger regularization parameters.

Next, we present the results for the uniform sampling model. The SoS image in Fig. 12(a) is identical to the one in Fig. 11(a). Fig. 12(b) is reconstructed by the GRAPPA method, and Fig. 12(c) and 12(d) are reconstructed by the ℓ_1-SPIRiT method and the ℓ_1-3DHSTF method with regularization parameters 0.005 and 0.003, respectively. Both ℓ_1-SPIRiT and ℓ_1-3DHSTF reconstruct most of the target information, and are better than the GRAPPA method. For the zoom-in images, aliasing artifacts occur in Fig. 12(c) by the ℓ_1-SPIRiT method, but are efficiently removed by the ℓ_1-3DHSTF method. The SSIM indexes for zoom-in images of Fig. 12(c) and (d) by ℓ_1-SPIRiT and ℓ_1-3DHSTF, respectively, are 0.873 and 0.878.

In summary, for the random and uniform sampling cases on MRI phantoms, the ℓ_1-3DHSTF method performs much better than the ℓ_1-SPIRiT method in terms of keeping edges and remove aliasing artifacts. Moreover, unlike the sensitivity of the ℓ_1-SPIRiT model to the regularization parameter λ, our specific designed Γ in (14) makes the ℓ_1-3DHSTF model robust to the regularization parameter λ. For the MRI phantom cases in Fig. 2 and Fig. 12 with the same uniform sampling model in Fig. 10(b), though the target slices are different, the ℓ_1-3DHSTF method is efficient to reconstruct high quality images by the same parameter $\lambda = 0.003$. It shows that our model is not sensitive to λ for the K-space data acquired on the same MRI System with the same sampling model.

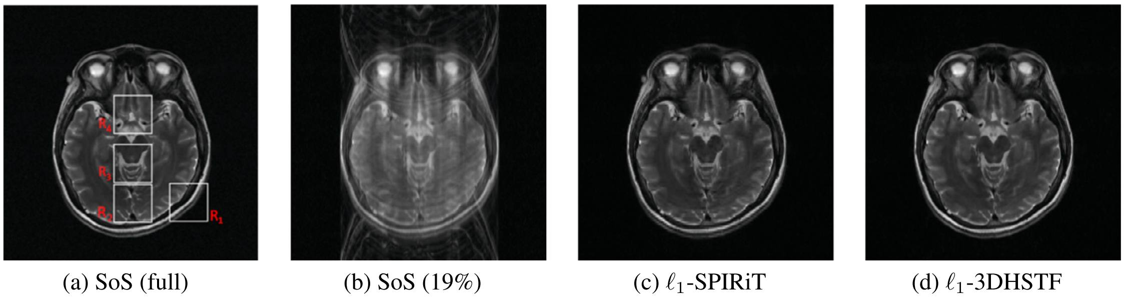


Fig. 13. Reconstruction results on 19% K-space data of the matrix in Fig. 10 (c). (a) SoS image of the full K-space with zoom-in parts; (b) SoS image of the 19% K-space data; (c) ℓ_1-SPIRiT [34] with parameter 0.018; (d) the ℓ_1-3DHSTF with parameter 0.0003.

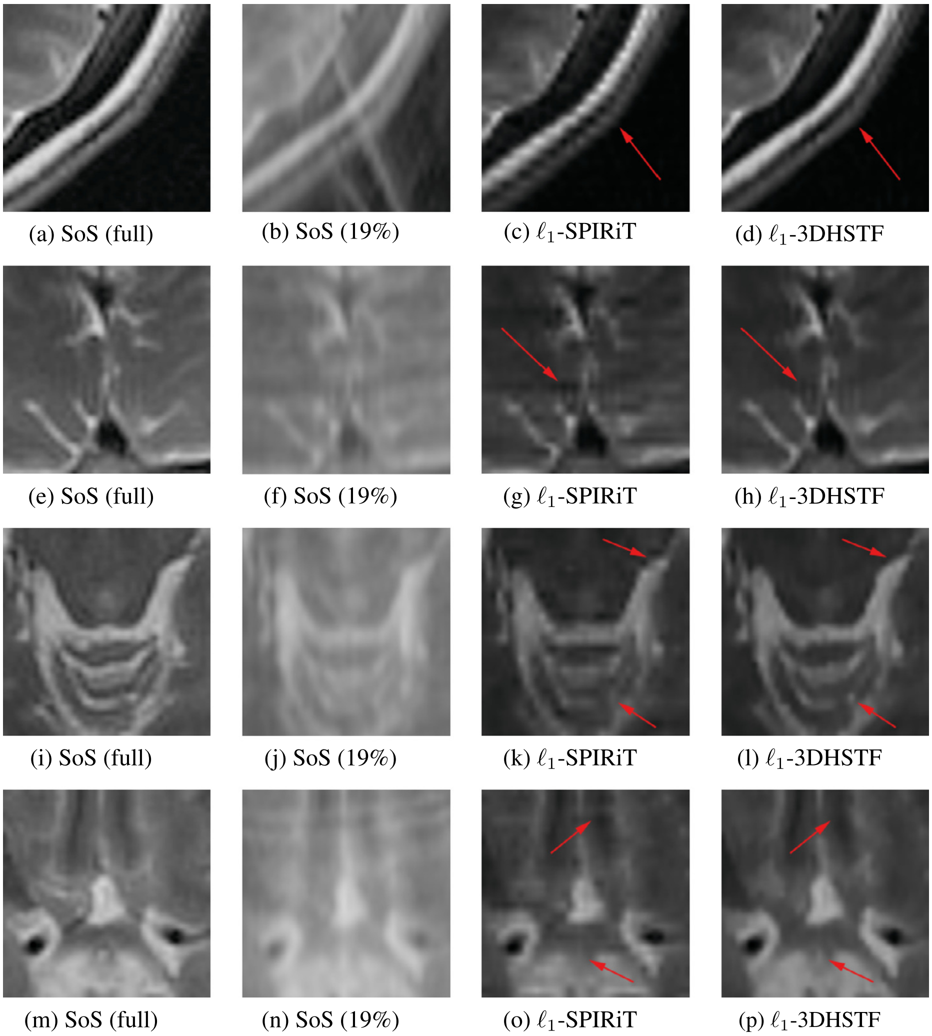
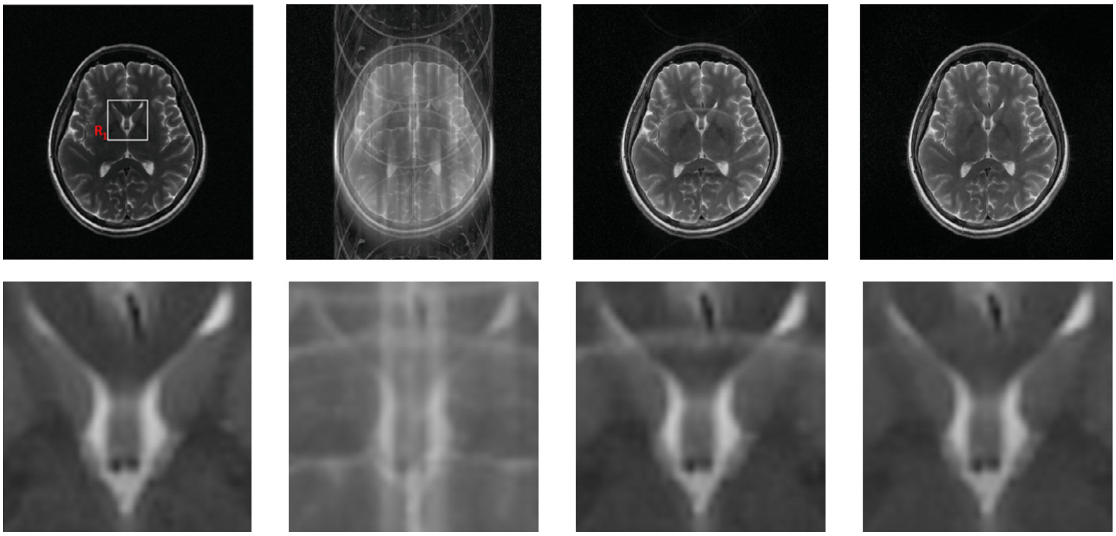


Fig. 14. Zoom-in parts of the reconstruction results in Fig. 13. First column: (a), (e), (i) and (m) SoS image of the full K-space. Second column: (b), (f), (j) and (n) SoS image of the 19% K-space data. Third column: (c), (g), (k) and (o) ℓ_1-SPIRiT [34] with parameter 0.018. Fourth column: (d), (h), (l) and (p) the ℓ_1-3DHSTF with parameter 0.0003.

5.2. In-vivo data

In this subsection we test the ℓ_1-3DHSTF method on MRI data that is obtained by head examination from a healthy volunteer. The imaging was done on a 3T MRI system. Transverse T_2-weighted images were



(a) SoS (full) (b) SoS (27%) (c) ℓ_1-SPIRiT (d) ℓ_1-3DHSTF

Fig. 15. Reconstruction results (First row) and corresponding zoom-in parts (Second row) on 27% K-space data by the uniform sampling mode (one line taken from every four lines) with 6 ACS lines in Fig. 10 (d). (a) SoS image of the full K-space; (b) SoS image of the 27% K-space data; (c) ℓ_1-SPIRiT method [34] with parameter 0.008; (d) the ℓ_1-3DHSTF method with parameter 0.0003. Second row is the zoom-in parts of first row, respectively.

Table 2
SSIM index for the zoom-in parts from reconstructed images on In-vivo data.

Fig. 13 (a)	SoS (19%)	ℓ_1-SPIRiT	ℓ_1-3DHSTF	Fig. 14
R_1	0.600	0.879	0.912	First row
R_2	0.683	0.862	0.933	Second row
R_3	0.662	0.857	0.920	Third row
R_4	0.718	0.872	0.924	Fourth row
Fig. 15 (a)	SoS (27%)	ℓ_1-SPIRiT	ℓ_1-3DHSTF	Fig. 15
R_1	0.572	0.919	0.958	Second row

acquired with a turbo spin-echo sequence. The detail imaging parameters are as follows: field of view = 256×256 mm^2, image matrix size = 256×256, slice thicknesses = 3 mm, flip angle = 150 degree, repetition time = 5920 ms, echo time = 101 ms, echo train length = 11 and number of excitation = 1. Two slices of 32-coil images were collected to compare the performance of the ℓ_1-SPIRiT method and the ℓ_1-3DHSTF method.

For the first slice, the full K-space data of 32-coil images are collected and their SoS image is considered as a reference image shown in Fig. 13(a). About 19% full K-space data with only 6 ACS lines are collected using the sampling model in Fig. 10(c). The resulting SoS image of the 19% full K-space data in Fig. 13(b) is noisy and the brain structures in this image are blurry. Furthermore, faint semicircle-like aliasing artifacts can be seen in the upper and lower portions of the image due to accelerating K-space sampling model. The regularization parameters of the ℓ_1-SPIRiT method and the ℓ_1-3DHSTF method are respectively set to be 0.018 and 0.0003 to reconstruct high quality images. From Fig. 13(c) and 13(d), we see that the ℓ_1-SPIRiT and the ℓ_1-3DHSTF reconstruct edge information of structure and suppress aliasing artifacts which are observable in the downsampling SoS image in Fig. 13(b). For conveniently comparing the difference, four parts labeled by R_1, R_2, R_3 and R_4 in Fig. 13(a) are zoomed-in in Fig. 14, and the zoom-in images in the first, second, third and fourth columns are corresponding to Fig. 13(a)–(d), respectively.

For the first row of Fig. 14, zoom-in images in Fig. 14(c) and 14(d) have better structures of skull and scalp than those in Fig. 14(b), and their corresponding SSIM values are 0.879 and 0.912 according to SSIM index in Table 2. Comparing with reference SoS image of the full K-space in Fig. 14(a), the image in Fig. 14(c) by the ℓ_1-SPIRiT method has faint ripple artifacts (arrow pointing to), but the image in Fig. 14(d) by the ℓ_1-3DHSTF method does not suffer from these artifacts and is more close to the reference one. For the second region of lobus occipitalis, 'black concave artifacts' (arrow pointing to) obviously occurs in the Fig. 14(g) with SSIM value 0.862 by the ℓ_1-SPIRiT method, but the ℓ_1-3DHSTF method can inhibit these artifacts by 3D semi-tight framelet regularization and provides close structures in Fig. 14(h) with SSIM value 0.933 with respect to the reference one in Fig. 14(e).

For the zoom-in images in the third row of Fig. 14, the cerebellum lobulus in Fig. 14(j) is discernible and blurred. However, images in Fig. 14(k) and 14(l) have better structures of cerebellum than images in Fig. 14(j). Comparing with reference SoS image of the full K-space in Fig. 14(i), the image in Fig. 14(l) with SSIM value 0.920 by the ℓ_1-3DHSTF method obviously preserves tiny detail (lower arrow pointing to) and edges (upper arrow pointing to) more noticeable than those in the Fig. 14(k) by the ℓ_1-SPIRiT method. The lobulus structures by the ℓ_1-3DHSTF method are high contrast and more obvious to be observed in Fig. 14(l), but the geometrical structures in Fig. 14(k) with SSIM value 0.857 are blurred by the ℓ_1-SPIRiT method. The final region of suprasellar cistern is provided in the last row of Fig. 14. 'White aliasing artifacts' (upper arrow pointing to) occurs in Fig. 14(o) with SSIM value 0.872 by the ℓ_1-SPIRiT method. However, the ℓ_1-3DHSTF method can remove these aliasing artifacts and provide distinguishable structures (lower arrow pointing to) at upper-middle position of Fig. 14(p) with SSIM value 0.924.

For the second set of 32 coil images, the reference SoS image of full K-pace data is shown in Fig. 15(a) and the SoS image of 27% K-space data by the uniform sampling model with 6 ACS lines in Fig. 10(d) is presented in Fig. 15(b). Regularization parameters of the ℓ_1-SPIRiT method and the ℓ_1-3DHSTF method are set to be 0.008 and 0.0003, respectively. The reconstruction images in Fig. 15(c) and 15(d) respectively by the ℓ_1-SPIRiT method and our ℓ_1-3DHSTF method mostly reduce the up and down half aliasing circles which are seen in Fig. 15(b). But one aliasing circle still obviously exists at the middle and the lower position of Fig. 15(c) by the ℓ_1-SPIRiT method, which is removed in Fig. 15(d) by our ℓ_1-3DHSTF method. We zoom in the region of genu corpus callosum in the second row of Fig. 15 to compare the difference between the ℓ_1-SPIRiT and ℓ_1-3DHSTF methods. Edge geometrical structures in Fig. 15(b) of the SoS image of 27% K-space data are blurred and discernible. From the zoom-in images in Fig. 15 (c) and 15(d) with their corresponding SSIM value 0.919 and 0.958, we see that the ℓ_1-3DHSTF method preserves edges much shaper and removes aliasing artifacts better than the ℓ_1-SPIRiT method, and provides almost as same as the reference zoom-in one by SoS image of the full K-space data.

These experiments show that the ℓ_1-3DHSTF method efficiently removes aliasing artifacts through considering correlation information of coil images. It has a grater capacity of preserving edges, tiny details, and structures in constructed images to facilitate doctor's diagnosis.

6. Conclusions and further remarks

In this paper, we propose a ℓ_1-W3D model for the pMRI reconstruction with 3DHSTF system that is tailor-made for the sparse representation of 3D cuboid data from different coil images. The 3DHSTF system has many desirable properties that nicely fits into the setting pMRI reconstruction. We use ADMM scheme to solve our ℓ_1-W3D model and our numerical experiments demonstrate the effectiveness and efficiency of the ℓ_1-3DHSTF model in removing aliasing artifacts and preserving edges.

We remark that the ℓ_1-3DHSTF model reconstructs images with significantly less aliasing artifacts and at the same time requires only a few ACS lines. Moreover, the ℓ_1-3DHSTF model is robust to the regularization parameter when the sampling model and number of coils are fixed. Further improvement of our ℓ_1-W3D could be considered. For example, we could consider 3D directional tight framelet systems with higher order of vanishing moments and short support; or incorporated with machine learning techniques, which have recently been proposed to improve the pMRI reconstruction quality; see e.g., [4,24]. These techniques include both image domain approaches for better image regularization and K-space approaches for better K-space completion.

We also remark that the neural network has recently been used for the pMRI reconstruction [2]. The challenges of the neural network based pMRI problem are (i) lack of public databases with a large number of multi-coil K-space data [24]; (ii) varying imaging parameters' setting of each MRI machine (for example, field of view, slice thicknesses, and so on), which are essential for a successful reconstruction [1]; and (iii) the patients' heartbeat, slight body moving and other factors in the process of scanning that can form gradient information similar to adversarial attack, which affects the accuracy of prediction, resulting in blurred anatomical structure details and artifacts in reconstructed MRI images [3]. Hence, in this paper, we do not consider neural network approach for pMRI reconstitution but focus on the pMRI reconstruction via optimization model (2) regularized by the proposed framelet systems.

Acknowledgment

The work of L. Shen was supported in part by the National Science Foundation under grant DMS-1913039 and Syracuse CUSE grant. The work of X. Zhuang was supported in part by the Research Grants Council of Hong Kong (Project no. CityU 11302218) and City University of Hong Kong (Project nos. 7005497 and 7005603). X. Zhuang thanks Prof. Bin Han from the University of Alberta for his helpful discussion and idea on the alternative algebraic proof of Theorem 1.

References

[1] M. Doneva, Mathematical models for magnetic resonance imaging reconstruction: an overview of the approaches, problems, and future research areas, IEEE Signal Process. Mag. 37 (1) (2020) 24–32.
[2] K.H. Jin, M.T. McCann, E. Froustey, M. Unser, Deep convolutional neural network for inverse problems in imaging, IEEE Trans. Image Process. 26 (9) (2017) 4509–4522.
[3] S.G. Finlayson, J.D. Bowers, J. Ito, J.L. Zittrain, A.L. Beam, I.S. Kohane, Adversarial attacks on medical machine learning, Science 363 (6433) (2019) 1287–1289.
[4] M. Akçakaya, S. Moeller, S. Weingärtner, K. Uğurbil, Scan-specific robust artificial-neural-networks for k-space interpolation (RAKI) reconstruction: database-free deep learning for fast imaging, Magn. Reson. Med. 81 (1) (2019) 439–453.
[5] E.J. Candès, D.L. Donoho, New tight frames of curvelets and optimal representations of objects with piecewise c^2 singularities, Commun. Pure Appl. Math. 57 (2) (2004) 219–266.
[6] L. Chaâri, J.C. Pesquet, A. Benazza-Benyahia, P. Ciuciu, A wavelet-based regularized reconstruction algorithm for SENSE parallel MRI with applications to neuroimaging, Med. Image Anal. 12 (2) (2011) 185–201.
[7] R. Chan, T. Chan, L. Shen, Z. Shen, Wavelet algorithms for high-resolution image reconstruction, SIAM J. Imaging Sci. 24 (4) (2003) 1408–1432.
[8] R. Chan, S. Riemenschneider, L. Shen, Z. Shen, Tight frame: an efficient way for high-resolution image reconstruction, Appl. Comput. Harmon. Anal. 17 (1) (2004) 91–115.
[9] C.K. Chui, W. He, J. Stöckler, Compactly supported tight and sibling frames with maximum vanishing moments, Appl. Comput. Harmon. Anal. 13 (3) (2002) 224–262.
[10] I. Daubechies, Ten Lectures on Wavelets, SIAM, Philadelphia, PA, 1992.
[11] I. Daubechies, B. Han, A. Ron, Z. Shen, Framelets: MRA-based constructions of wavelet frames, Appl. Comput. Harmon. Anal. 14 (1) (2003) 1–46.
[12] A. Deshmane, V. Gulani, M.A. Griswold, N. Seiberlich, Parallel MR imaging, J. Magn. Reson. Imaging 36 (1) (2012) 55–72.
[13] D.L. Donoho, G. Kutyniok, M. Shahram, X. Zhuang, A rational design of a digital shearlet transform, in: SampTA'11, Singapore, 2011.
[14] W.A. Edelstein, J.M.S. Hutchison, G. Johnson, T. Redpath, Spin warp imaging and applications to human whole body imaging, Phys. Med. Biol. 25 (1980) 751–756.
[15] D. Gabay, B. Mercier, A dual algorithm for the solution of nonlinear variational problems via finite-element approximations, Comput. Math. Appl. 2 (1) (1976) 17–40.
[16] M.A. Griswold, P.M. Jakob, R.M. Heidemann, M. Nittka, V. Jellus, J. Wang, B. Kiefer, A. Haase, Generalized autocalibrating partially parallel acquisitions (GRAPPA), Magn. Reson. Med. 47 (6) (2002) 1202–1210.
[17] B. Han, Properties of discrete framelet transforms, Math. Model. Nat. Phenom. 8 (1) (2013) 18–47.
[18] B. Han, Framelets and Wavelets: Algorithms, Analysis, and Applications, Birkhäuser, Springer International Publishing, 2018.
[19] B. Han, T. Li, X. Zhuang, Directional compactly supported box spline tight framelets with simple geometric structure, Appl. Math. Lett. 91 (2019).

[20] B. Han, Q. Mo, Z. Zhao, X. Zhuang, Directional compactly supported tensor product complex tight framelets with applications to image denoising and inpainting, SIAM J. Imaging Sci. 12 (4) (2019) 1739–1771.
[21] B. Han, Z. Zhao, X. Zhuang, Directional tensor product complex tight framelets with low redundancy, Appl. Comput. Harmon. Anal. 41 (2) (2016) 603–637.
[22] K.H. Jinn, D. Lee, J.C. Ye, A general framework for compressed sensing and parallel MRI using annihilating filter based low-rank Hankel matrix, IEEE Trans. Comput. Imaging 2 (4) (2016) 480–495.
[23] S.L. Keeling, C. Clason, M. Hintermüller, F. Knoll, A. Laurain, G. von Winckel, An image space approach to Cartesian based parallel MR imaging with total variation regularization, Med. Image Anal. 16 (1) (2012) 189–200.
[24] F. Knoll, K. Hammernik, C. Zhang, S. Moeller, T. Pock, D.K. Sodickson, M. Akcakaya, Deep-learning methods for parallel magnetic resonance imaging reconstruction: a survey of the current approaches, trends, and issues, IEEE Signal Process. Mag. 37 (1) (2020) 128–140.
[25] A. Kumar, D. Welti, R.R. Ernst, NMR Fourier Zeugmatography, J. Magn. Reson. 18 (1975) 69–83.
[26] G. Kutyniok, D. Labate, Shearlets: Multiscale Analysis for Multivariate Data, Birkhauser, 2012.
[27] G. Kutyniok, W.-Q. Lim, R. Reisenhofer, ShearLab 3D: faithful digital shearlet transforms based on compactly supported shearlets, ACM Trans. Math. Softw. 42 (2016) 5.
[28] Y.-R. Li, R.H. Chan, L. Shen, Y.-C. Hsu, W.-Y.I. Tseng, An adaptive directional haar framelet-based reconstruction algorithm for parallel magnetic resonance imaging, SIAM J. Imaging Sci. 9 (2) (2016) 794–821.
[29] Y.R. Li, L. Shen, B.W. Suter, Adaptive inpainting algorithm based on DCT induced wavelet regularization, IEEE Trans. Image Process. 22 (2) (2013) 752–763.
[30] Y.-R. Li, X. Zhuang, Parallel magnetic resonance imaging reconstruction algorithm by 3-dimension directional haar tight framelet regularization, Proc. SPIE, San Diego (2019), pp. 111381C-1-8.
[31] M. Lustig, J.M. Pauly, SPIRiT: iterative self-consistent parallel imaging reconstruction from arbitrary K-space, Magn. Reson. Med. 64 (2) (2010) 457–471.
[32] J. Ma, M. März, S. Funk, J. Schulz-Menger, G. Kutyniok, T. Schaeffter, C. Kolbitsch, Shearlet-based compressed sensing for fast 3D cardiac MR imaging using iterative reweighting, Phys. Med. Biol. 63 (23) (2018) 235004.
[33] R.L. Morrison, M. Jacob, M.N. Do, Multichannel estimation of coil sensitivities in parallel MRI, in: 2007 4th IEEE International Symposium on Biomedical Imaging: from Nano to Macro, IEEE, 2007, pp. 117–120.
[34] M. Murphy, M. Alley, J. Demmel, K. Keutzer, S. Vasanawala, M. Lustig, Fast ℓ_1-SPIRiT compressed sensing parallel imaging MRI: scalable parallel implementation and clinically feasible runtime, IEEE Trans. Med. Imaging 31 (6) (2012) 1250–1262.
[35] J. Nocedal, S. Wright, Numerical Optimization, Springer-Verlag, 1999.
[36] R.A. Novelline, L.F. Squire, Squire's Fundamentals of Radiology, 5th edition, Harvard University Press, Cambridge, 1997.
[37] K.P. Pruessmann, M. Weiger, M.B. Scheidegger, P. Boesiger, SENSE: sensitivity encoding for fast MRI, Magn. Reson. Med. 42 (5) (1999) 952–962.
[38] L. Shen, B.W. Suter, E.E. Tripp, Structured sparsity promoting functions, J. Optim. Theory Appl. 183 (2019) 386–421.
[39] L. Shen, B.W. Suter, E.E. Tripp, Algorithmic versatility of SPF-regularization methods, Anal. Appl. 19 (01) (2021) 43–69.
[40] D.S. Weller, J.R. Polimeni, L. Grady, L.L. Wald, E. Adalsteinsson, V.K. Goyal, Sparsity-promoting calibration for GRAPPA accelerated parallel MRI reconstruction, IEEE Trans. Med. Imaging 32 (7) (2013) 1325–1335.
[41] Y. Xiao, X. Zhuang, Adaptive directional Haar tight framelets on bounded domains for digraph signal representations, J. Fourier Anal. Appl. 27 (2) (2021).
[42] N. Yanasak, G. Clarke, R.J. Stafford, F. Goerner, M. Steckner, I. Bercha, J. Och, M. Amurao, Parallel Imaging in MRI: Technology, Applications, and Quality Control, AAPM Report No. 118, 2015.
[43] X. Ye, Y. Chen, F. Huang, Computational acceleration for MR image reconstruction in partially parallel imaging, IEEE Trans. Med. Imaging 30 (5) (2011) 1055–1063.
[44] M. Uecker, T. Hohage, K.T. Block, et al., Image reconstruction by regularized nonlinear inversion joint estimation of coil sensitivities and image content, Magn. Reson. Med. 60 (3) (2008) 674–682.
[45] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Trans. Image Process. 13 (4) (2004) 600–612.