

# Periodic Spline-Based Frames for Image Restoration

Amir Averbuch<sup>1</sup> Pekka Neittaanmäki<sup>2</sup> Valery Zheludev<sup>1</sup>

<sup>1</sup>School of Computer Science  
Tel Aviv University, Tel Aviv 69978, Israel

<sup>2</sup>Department of Mathematical Information Technology  
P.O. Box 35 (Agora), University of Jyväskylä, Finland

## Abstract

We present a design scheme to generate tight and semi-tight frames in the space of discrete-time periodic signals, which are originated from four-channel perfect reconstruction periodic filter banks. The filter banks are derived from interpolating and quasi-interpolating polynomial and discrete splines. Each filter bank comprises one linear phase low-pass filter (in most cases interpolating) and one high-pass filter, whose magnitude's response mirrors that of a low-pass filter. In addition, these filter banks comprise two band-pass filters. We introduce the notion of local discrete vanishing moments (LDVM). In the tight frame case, analysis framelets coincide with their synthesis counterparts. However, in the semi-tight frames, we have the option to swap LDVM between synthesis and analysis framelets. The design scheme is generic and it enables us to design framelets with any number of LDVM. The computational complexity of the framelet transforms, which consists of calculating the forward and the inverse fast Fourier transforms, practically does not depend on the number of LDVM and does depend on the size of the impulse response filters. The designed frames are used for image restoration tasks, which were degraded by blurring, random noise and missing pixels. The images were restored by the application of the Split Bregman Iterations method. The frames performances are evaluated. A potential application of this methodology is the design of a snapshot hyperspectral imager that is based on a regular digital camera. All these imaging applications are described.

## 1 Introduction

Restoration of multidimensional signals that were corrupted and/or damaged and/or noised is a major challenge that the signal/image processing community faces nowadays when rich multimedia content is the most popular data being transmitted over diverse networks types including mobile. Quality degradation in multidimensional signals can come from sampling, acquisition, transmission through noisy channels, to name some. Restoration of multidimensional signals includes denoising, deblurring, recovering missing or damaged samples or fragments (inpainting in images), resolution enhancement and super resolution. The processing goals are to improve the visual perception of still and video signals.

Hyperspectral imaging is a relatively new field of investigation. Its imager has a large number of wavebands in a given wavelength range. For example, the range can be 10, hundreds or even thousands of wavebands. Spectral imaging has a multitude of applications in many fields including scientific research, engineering and medical equipment ranging from a color camera to sophisticated astronomical equipment. Currently, known spectral imagers usually use dynamic spectral filters, arrays, scanning procedures and multiple-lens optical schemes that are quite complicated and expensive. However, a combination of optics with proper digital processing described in this paper is able to provide "snapshot" spectral imagers based on regular digital cameras [17].

Recent developments in wavelet frames (framelets) analysis provide innovative and powerful tools to face faithfully and robustly the above challenges. Framelets produce redundant expansions for multidimen-

sional signals that, in particular, provide an additional sparsity to the signals representation due to better adaptation abilities and due to redundant representations. A valuable advantage, which redundant representations hold, is their abilities to restore missing and incomplete information and to represent efficiently and compactly the data.

Frame expansions of signals demonstrate resilience to the coefficients disturbances and losses [19, 18, 24, 1]. Thus, frames can serve as a tool for error correction for signals transmitted through lossy channels.

Implicitly, this resilience is utilized in signal/image restoration, which is based on a prior assumption that a frame expansion of a given signal/image is sparse. In principle, only part of the samples/pixels is needed to (near) perfect object restoration. This approach, which is a variation of *Compressive Sensing* methodology ([14]), proved to be extremely efficient for image restoration. Practically, compressed sensing approach is implemented via minimization of a parameterized functional where the sparse representation is reflected in the  $l_1$  norm of the transform coefficients. The  $\|\cdot\|_1$  minimization does not provide an explicit solution and can be resolved only by iterative methods. The *split Bregman iteration* (SBI) scheme, which was recently introduced in [16], provides for that a fast and stable algorithm. Variations of this scheme and its application to image restoration using wavelet frames are described in [23, 22], to mention a few. A variety of impressive results on image restoration were reported in the last couple of years. A survey is given in [30] while a recent development is described in [22].

The SBI methodology proved to be useful in the design of “snapshot” spectral imagers based on regular digital cameras. It enables to reconstruct a spectral cube from a dispersed-diffused sensors’ array.

Due to applications diversity, it is important to have a library of wavelet frames in order to select a frame that fits best a specific task. Forward and inverse transforms in iterative algorithms are repeated many times, therefore, members in this library must have fast and stable transforms implementation. Waveforms symmetry with the availability of vanishing moments are also important in order to avoid distortions when thresholding is used. To satisfy these requirements, most of the framelet systems in practical use operate with the compactly supported framelets and the transforms are implemented by finite (and short) impulse response (FIR) oversampled filter banks [13, 10, 11, 8]. A common requirement is to have tight frames. Increase in the number of vanishing moments in framelets requires an increase in the filters’ length, which results in higher computational cost. It also can produce boundary artifacts in the processed images.

The oversampled perfect reconstruction (PR) filter banks generate wavelet-type frames in the space of discrete-time signals [12, 9]. Utilizing infinite impulse response (IIR) filter banks with a relaxation of the tightness requirement provides a number of additional opportunities. Properties such as symmetry, interpolation, flat spectra, combined with fine time-domain localization of framelets can be easily achieved as well as a high number of vanishing moments [6, 7]. In these papers, the key point is the low-pass filters design, which, being applied to the even subarray of a signal, well approximate the odd subarray (the prediction filters). A natural source for such filters are the discrete and polynomial splines. A number of 3-framelet systems was derived from the discrete splines in [6, 7]. The transforms are implemented in a fast way using recursive filtering. Non-compactness of the waveforms supports is compensated by their exponential decay as time goes to infinity. In principle, any number of vanishing moments can be derived but the implementation computational cost grows fast.

This drawback can be overcome by switching to a periodic setting, which is the subject of this paper. A variety of four-channel PR filter banks, where the low-pass filters are derived from interpolating and quasi-interpolating polynomial and discrete splines, are designed. These filter banks generate a library of 4-framelet tight and semi-tight frames with diverse properties.

The transforms implementation is reduced to application of the direct and the inverse fast Fourier transforms (FFT). Actually, the implementation cost does not depend on the number of vanishing moments in the framelets. Therefore, framelets utilization with any number of vanishing moments and filter banks with IIR filters become practical.

A number of framelets systems is explicitly designed and described in this paper. However, the design scheme is generic and this framelets library can be easily expanded. Preliminary results in the periodic discrete-time frame design with three framelets are reported in [32]. Recently, framelets in the space of periodic continuous functions are studied in [28].

The designed framelets libraries were tested for image restoration and demonstrated a high quality performance. The framelet libraries enable us to select a frame that best fits each specific application. For example, in the restoration experiments with relatively smooth images such as “Lena” and “Window”, the best results were achieved with compactly supported framelets derived from pseudo-splines ([8]) and from the quasi-interpolating quadratic splines. However, in the experiments with images comprising a fine texture, such as “Barbara” and “Fingerprint”, the framelets with high number of vanishing moments, which are derived from the discrete interpolating splines, significantly outperformed all others. Note that in many experiments, the semi-tight frames outperformed tight frames. For example, in the experiments with “Lena” and “Window” images, the best performance was demonstrated by the semi-tight frame derived from pseudo-splines, while in numerical simulation of the snapshot hyperspectral imaging, the semi-tight frame derived from the quasi-interpolating quadratic splines produces the best quality output.

The paper is organized as follows: The introductory Section 2 recalls the notion of periodic discrete-time signals and of periodic filter (p-filter). The PR periodic filter banks (p-filter banks) are presented and characterized via the polyphase representation of their discrete Fourier transform (DFT). Frames in the discrete-time periodic signals space, which originated from PR oversampled p-filter banks, are also introduced in Section 2. Design of four-channel p-filter banks, which generate tight and semi-tight frames, is presented in Section 3. Section 4 discusses the restoration of sampled polynomials by low-pass p-filters and introduces the notion of local discrete vanishing moments (LDVM). A collection of FIR and IIR p-filters from the interpolating and quasi-interpolating polynomial and discrete splines is derived in Section 5. A number of four-channel p-filter banks with spline p-filters that generate tight and semi-tight frames, is described in section 6. The designed tight and semi-tight frames are used in Section 7 for restoration of images, which were degraded by blurring, random noise and missing pixels. The frames performances are evaluated. The section is concluded by an extensive discussion on the restoration experiments. Section 8 describes experiments on numerical simulation of the snapshot hyperspectral imaging.

## 2 Preliminaries: Periodic discrete-time signals and filters

In this section, we present notions and some known facts.

### 2.1 Signals and transforms

**Notations:**  $\Pi[N]$  is the  $N$ -dimensional vector space consisting of  $N$ -periodic real-valued sequences  $\mathbf{x} \stackrel{\text{def}}{=} \{x[k]\}$ ,  $k \in \mathbb{Z}$  that are called signals. *Throughout the paper*,  $N = 2^j$ ,  $j \in \mathbb{N}$  and  $\omega \stackrel{\text{def}}{=} e^{2\pi i/N}$ . The inner product and the norm in  $\Pi[N]$  are defined by  $\langle \mathbf{x}, \mathbf{h} \rangle \stackrel{\text{def}}{=} \sum_{k=0}^{N-1} x[k] h[k]$ ,  $\|\mathbf{x}\| = \|\mathbf{x}\|_2 \stackrel{\text{def}}{=} \sqrt{\sum_{k=0}^{N-1} |x[k]|^2}$ , respectively. The discrete circular convolution in  $\Pi[N]$  is  $\mathbf{y} = \mathbf{h} \circledast \mathbf{x} \iff y[k] = \sum_{l=0}^{N-1} h[k-l] x[l]$ ,  $\mathbf{y} \in \Pi[N]$ .

The signal  $\delta[k]$ , such that  $\delta[0] = 1$ ,  $l \in \mathbb{Z}$  and  $\delta[k] = 0$  for  $k \neq 0$ , is called the Kronecker delta.

Note that any discrete-time signal  $\mathbf{y}$  of limited duration  $L$  can be embedded into a space  $\Pi[N]$ ,  $N \geq L$ , by the periodization  $\tilde{\mathbf{y}} \stackrel{\text{def}}{=} \{\tilde{y}[k] = \sum_{l \in \mathbb{Z}} y[k + lN]\}$ .

The discrete Fourier transform (DFT) of a signal  $\mathbf{x}$  and its inverse (IDFT) are  $\hat{x}[n] = \sum_{k=0}^{N-1} x[k] \omega^{-nk}$  and  $x[k] = 1/N \sum_{n=0}^{N-1} \hat{x}[n] \omega^{nk}$ , respectively, and they are calculated by the fast Fourier transform (FFT) algorithm. Since the signals are real-valued, the complex conjugate DFT becomes  $\hat{x}[n]^* = \hat{x}[-n]$ .

We use the notation  $\hat{x}[n]_m = \sum_{k=0}^{N/2^m-1} x[k] \omega^{-2^m nk}$ ,  $x[k] = 2^m/N \sum_{n=0}^{N/2^m-1} \hat{x}[n]_m \omega^{2^m nk}$  for the DFT of signals belonging to the space  $\Pi[N/2^m]$ . The sequence  $\{\hat{x}[n]_m\}$  is  $N/2^m$ -periodic.

Relations between signals and their DFT are:

*Parseval identities:*  $\langle \mathbf{x}, \mathbf{h} \rangle = 1/N \sum_{n=-N/2}^{N/2-1} \hat{x}[n] \hat{h}[-n]$ ,  $\|\mathbf{x}\|^2 = 1/N \sum_{n=-N/2}^{N/2-1} |\hat{x}[n]|^2$ .

*Circular convolution:* If  $\mathbf{y} = \mathbf{x} \circledast \mathbf{h}$  then  $\hat{y}[n] = \hat{h}[n] \hat{x}[n]$ .

*Circular shift:* If  $\mathbf{x}_d \stackrel{\text{def}}{=} \{x[k + d]\}$ , where  $d$  is an integer number, then  $\hat{x}_d[n] = \omega^{nd} \hat{x}[n]$ .

**Finite differences:** The forward finite differences of a signal  $\mathbf{x}$  is  $\Delta[\mathbf{x}] \stackrel{\text{def}}{=} \{x[k+1] - x[k]\}$ ,  $k \in \mathbb{Z}$ ,  $\Delta^n[\mathbf{x}] \stackrel{\text{def}}{=} \Delta[\Delta^{n-1}[\mathbf{x}]]$ . The even-order central finite differences are  $\delta^2[\mathbf{x}] \stackrel{\text{def}}{=} \{x[k+1] - 2x[k] + x[k-1]\}$ ,  $k \in \mathbb{Z}$ , and  $\delta^{2r}[\mathbf{x}] \stackrel{\text{def}}{=} \delta^2[\delta^{2n-2}[\mathbf{x}]]$ . Their DFTs are

$$\hat{\Delta}^m[\mathbf{x}][n] = (\omega^n - 1)^m \hat{x}[n], \quad \hat{\delta}^{2l}[\mathbf{x}][n] = \left(2i \sin \frac{\pi n}{N}\right)^{2l} \hat{x}[n]. \quad (1)$$

**Proposition 2.1** Assume that  $\mathbf{P}_{n-1} \stackrel{\text{def}}{=} \{P_{n-1}(k)\}$ ,  $k \in \mathbb{Z}$ , is a sampled polynomial of degree not exceeding  $n-1$ . Then,  $\Delta^n[\mathbf{P}_{n-1}] = \mathbf{0}$ . If  $n = 2r$  then  $\delta^{2r}[\mathbf{P}_{2r-1}] = \mathbf{0}$ .

**Polyphase representation of the DFT:** The signals  $\mathbf{x}_0 = \{x[2k]\}$  and  $\mathbf{x}_1 = \{x[2k+1]\}$  belonging to  $\Pi[N/2]$  are called the even and the odd polyphase components of the signal  $\mathbf{x} \in \Pi[N]$ , respectively.

Their DFTs are

$$\begin{aligned} \hat{x}[n] &= \hat{x}_{0,2}[n]_1 + \omega^{-n} \hat{x}_{1,2}[n]_1, & \hat{x}[n+N/2] &= \hat{x}_{0,2}[n]_1 - \omega^{-n} \hat{x}_{1,2}[n]_1 \\ \implies \hat{x}_{0,2}[n]_1 &= (\hat{x}[n] + \hat{x}[n+N/2])/2, & \hat{x}_{1,2}[n]_1 &= (\hat{x}[n] - \hat{x}[n+N/2])/2\omega^{-n}. \end{aligned} \quad (2)$$

## 2.2 Periodic filters

A linear operator  $\mathbf{h} : \Pi[N] \rightarrow \Pi[N]$ ,  $\mathbf{y} = \mathbf{h}\mathbf{x}$ , is time-invariant if  $\mathbf{h}\mathbf{x}[\cdot + d] = \mathbf{y}[\cdot + d]$ . Such operators are called periodic digital filters (p-filters). Application of the p-filter to a signal  $\mathbf{x} \in \Pi[N]$  is implemented via the discrete circular convolution of  $\mathbf{x}$  with the signal  $\mathbf{h} = \{h[k]\} \in \Pi[N]$ . It is called the *impulse response* (IR) of the p-filter  $\mathbf{h}$  denoted  $\mathbf{y} = \mathbf{h}\mathbf{x} = \mathbf{h} \circledast \mathbf{x} \iff y[k] = \sum_{l=0}^{N-1} h[k-l]x[l]$ .

We use the notation  $\mathbf{h}$  for both a p-filter and its IR  $\{h[k]\}$ . The DFT of the IR of the p-filter  $\mathbf{h}$  is  $\hat{h}[n] = \sum_{k=0}^{N-1} h[k]\omega^{-nk}$ ,  $n \in \mathbb{Z}$ , and it is called the frequency response (FR).

Periodic filtering of a signal reduces to multiplication in the frequency domain:  $\mathbf{h} \circledast \mathbf{x} \iff \hat{h}[n]\hat{x}[n]$ ,  $n \in \mathbb{Z}$ . The FR of a p-filter can be represented in a polar form  $\hat{h}[n] = |\hat{h}[n]| e^{i \arg(\hat{h}[n])}$ , where the positive  $N$ -periodic sequence  $|\hat{h}[n]|$  is called the magnitude response (MR) of  $\mathbf{h}$  and the real-valued  $2\pi$ -periodic sequence  $\arg(\hat{h}[n])$  is called the phase response of  $\mathbf{h}$ . A p-filter is referred to as a linear phase if its phase response is linear in  $n$ . If the IR of a filter  $\mathbf{h}$  is symmetric or antisymmetric within the interval  $k = -N/2, \dots, N/2 - 1$ , then  $\mathbf{h}$  is a linear phase filter.

## 2.3 Multirate p-filtering

Assume,  $N = 2^j$ ,  $\mathbf{x} \in \Pi[N]$ , and  $M = 2^l$ ,  $l < j$ . The operation  $(\downarrow M)\mathbf{x} = \hat{\mathbf{x}} \stackrel{\text{def}}{=} \{x[Mk]\} \in \Pi[N/M]$  is called downsampling the signal  $\mathbf{x}$  by factor  $M$ . Assume that a signal  $\mathbf{x}$  belongs to  $\Pi[N/M]$ . The operation

$$(\uparrow M)\mathbf{x} = \hat{\mathbf{x}}, \quad \hat{x}[k] = \begin{cases} x[l], & \text{if } k = lM \\ 0, & \text{otherwise} \end{cases}, \quad l \in \mathbb{Z},$$

is called upsampling the signal  $\mathbf{x}$  by factor  $M$ . In the rest of the paper, we apply down(up) sampling by factor 2. Then, the downsampled signal  $(\downarrow 2)\mathbf{x} = \mathbf{x}_0$  is the even polyphase component of the signal  $\mathbf{x}$ , while the upsampled signal  $(\uparrow 2)\mathbf{x}$  is a signal whose odd polyphase component vanishes.

If p-filtering of a signal is accompanied by down(up)sampling then it is called multirate p-filtering. Let  $\mathbf{h} = \{h[k]\}$  and  $\tilde{\mathbf{h}} = \{\tilde{h}[k]\}$  be p-filters. Application of the p-filter  $\tilde{\mathbf{h}} = \{\tilde{h}[-k]\}$ , which is the time-reversed filter  $\tilde{\mathbf{h}}$ , to the signal  $\mathbf{x}$ , which is followed by downsampling by factor 2, produces the signal  $\tilde{\mathbf{y}} = \{\tilde{y}[k]\} \in \Pi[N/2]$  where

$$\tilde{y}[k] = \sum_{l=0}^{N-1} \tilde{h}[l-2k]x[l], \quad \hat{\tilde{y}}[n] = \hat{h}_0[-n]_1 \hat{x}_0[n]_1 + \hat{h}_1[-n]_1 \hat{x}_1[n]_1, \quad n \in \mathbb{Z}. \quad (3)$$

Application of the p-filter  $\mathbf{h}$  to a signal  $\check{\mathbf{y}} = \{\check{y}[k]\} \in \Pi[N/2]$ , which is upsampled by factor 2, produces the signal  $\check{\mathbf{x}} = \{\check{x}[k]\} \in \Pi[N]$  where

$$\check{x}[k] = \sum_{l=0}^{N/2-1} h[k-2l] \check{y}[l], \quad \hat{x}[n] = \hat{h}[n] \hat{y}[n]_1 \implies \hat{x}_0[n]_1 = \hat{h}_0[n]_1 \hat{y}[n]_1, \quad \hat{x}_1[n]_1 = \hat{h}_1[n]_1 \hat{y}[n]_1. \quad (4)$$

**Interpolating p-filters:** If the DFT of the even polyphase component of a p-filter  $\mathbf{h}$  is constant, i.e.  $\hat{h}_0[n]_1 \equiv C$ , then the p-filter is called interpolating. In this case, Eq. (4) implies that the DFT of the zero polyphase component of the output signal is  $\check{\mathbf{x}} \hat{x}_0[n]_1 = C \hat{y}[n]_1$ . This means that  $\check{x}[2k] = C \check{y}[k]$ ,  $k \in \mathbb{Z}$ .

## 2.4 Filter banks

The set of p-filters  $\tilde{\mathbf{H}} \stackrel{\text{def}}{=} \{\tilde{\mathbf{h}}^s\}$ ,  $s = 0, \dots, S-1$ , which, being time-reversed and applied to an input signal  $\mathbf{x} \in \Pi[N]$ , produces the set of the output signals  $\{\tilde{\mathbf{y}}^s\}_{s=0}^{S-1}$  downsampled by factor  $M$ ,

$$\tilde{y}^s[l] = \sum_{k=0}^{N-1} \tilde{h}^s[k-Ml] x[k], \quad s = 0, \dots, S-1, \quad l \in \mathbb{Z}. \quad (5)$$

It is called the  $S$ -channel analysis p-filter bank. The set of filters  $\mathbf{H} \stackrel{\text{def}}{=} \{\mathbf{h}^s\}$ ,  $s = 0, \dots, S-1$ , which, being applied to a set of input signals  $\{\mathbf{y}^s\} \in \Pi[N/M]$ ,  $s = 0, \dots, S-1$ , which are upsampled by factor  $M$ , produces the output signal  $\check{x}[l] = \sum_{s=0}^{S-1} \sum_{k=0}^{N/M-1} h^s[l-Mk] y^s[k]$ ,  $l \in \mathbb{Z}$ , is called the  $S$ -channel synthesis p-filter bank. If the upsampled signals  $\tilde{\mathbf{y}}^s$ ,  $s = 0, \dots, S-1$ , which are defined in Eq. (5), are used as an input to the synthesis p-filter bank and the output signal is  $\check{\mathbf{x}} = \mathbf{x}$ , then the pair of analysis-synthesis p-filter banks form a perfect reconstruction (PR) p-filter bank.

If the number of channels  $S$  equals to the downsampling factor  $M$  then the p-filter bank is said to be critically sampled. If  $S > M$  then the p-filter bank is oversampled. Critically sampled PR p-filter banks are used in wavelet analysis, while oversampled PR p-filter banks serve as a source for discrete-time wavelet frames design.

In the rest of the paper, we deal with p-filter banks, whose downsampling factor is  $M = 2$  and  $\tilde{\mathbf{h}}^0$  and  $\mathbf{h}^0$  are the low-pass filters.

## 2.5 Characterization of p-filter banks

Assume that  $\tilde{\mathbf{H}} \stackrel{\text{def}}{=} \{\tilde{\mathbf{h}}^s\}$ ,  $s = 0, \dots, S-1$ , is an analysis p-filter bank with the downsampling factor 2. Then, its application to a signal  $\mathbf{x} \in \Pi[N]$  produces  $S$  signals from  $\Pi[N/2]$  in the following way:

$$\tilde{y}^s[l] = \sum_{k=0}^{N-1} \tilde{h}^s[k-2l] x[k], \quad \hat{y}^s[n]_1 = \hat{h}_0^s[-n]_1 \hat{x}_0[n]_1 + \hat{h}_1^s[-n]_1 \hat{x}_1[n]_1, \quad s = 0, \dots, S-1. \quad (6)$$

Assume that  $\mathbf{H} \stackrel{\text{def}}{=} \{\mathbf{h}^s\}$ ,  $s = 0, \dots, S-1$ , is a synthesis p-filter bank with the upsampling factor 2. Then, its application to the upsampled signals  $\tilde{\mathbf{y}}^s \in \Pi[N/2]$  produces a signal from  $\Pi[N]$  in the following way:

$$\check{x}[l] = \sum_{s=0}^{S-1} \sum_{k=0}^{N/2-1} h^s[l-2k] \tilde{y}^s[k], \quad \hat{x}_0[n]_1 = \sum_{s=0}^{S-1} \hat{h}_0^s[n]_1 \hat{y}^s[n]_1, \quad \hat{x}_1[n]_1 = \sum_{s=0}^{S-1} \hat{h}_1^s[n]_1 \hat{y}^s[n]_1. \quad (7)$$

Equations (6) and (7) can be rewritten in a matrix form by

$$\begin{pmatrix} \hat{y}^0[n]_1 \\ \vdots \\ \hat{y}^{S-1}[n]_1 \end{pmatrix} = \tilde{\mathbf{P}}[-n] \cdot \begin{pmatrix} \hat{x}_0[n]_1 \\ \hat{x}_1[n]_1 \end{pmatrix}, \quad \begin{pmatrix} \hat{x}_0[n]_1 \\ \hat{x}_1[n]_1 \end{pmatrix} = \mathbf{P}[n] \cdot \begin{pmatrix} \hat{y}^0[n]_1 \\ \vdots \\ \hat{y}^{S-1}[n]_1 \end{pmatrix}$$

where the  $S \times 2$  analysis and the  $2 \times S$  synthesis polyphase matrices are, respectively,

$$\tilde{\mathbf{P}}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{h}_0^0[n]_1 & \hat{h}_1^0[n]_1 \\ \vdots & \vdots \\ \hat{h}_0^{S-1}[n]_1 & \hat{h}_1^{S-1}[n]_1 \end{pmatrix} \quad \mathbf{P}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{h}_0^0[n]_1 & \dots & \hat{h}_0^{S-1}[n]_1 \\ \hat{h}_1^0[n]_1 & \dots & \hat{h}_1^{S-1}[n]_1 \end{pmatrix}, \quad n \in \mathbb{Z}.$$

If the relations

$$\mathbf{P}[n] \cdot \tilde{\mathbf{P}}[-n] = \mathbf{I}_2, \quad (8)$$

hold for all  $n \in \mathbb{Z}$  when  $\mathbf{I}_2$  is the  $2 \times 2$  identity matrix, then

$$\mathbf{P}[n] \cdot \tilde{\mathbf{P}}[-n] \cdot \begin{pmatrix} \hat{x}_0[n]_1 \\ \hat{x}_1[n]_1 \end{pmatrix} = \begin{pmatrix} \hat{x}_0[n]_1 \\ \hat{x}_1[n]_1 \end{pmatrix}.$$

Thus, Eq. (8) is the condition for the pair  $\{\tilde{\mathbf{H}}, \mathbf{H}\}$  of p-filter banks to form a PR p-filter bank.

## 2.6 Frames in the periodic signals space

**Definition 2.2** A system  $\tilde{\Phi} \stackrel{\text{def}}{=} \{\tilde{\phi}_l\}_{l=0}^{L-1}$ ,  $L \geq N$ , of signals from  $\Pi[N]$  forms a frame for the space  $\Pi[N]$  if there exist positive constants  $A$  and  $B$  such that for any signal  $\mathbf{x} = \{x[k]\} \in \Pi[N]$  we have  $A\|\mathbf{x}\|^2 \leq \sum_{l=0}^{L-1} |\langle \mathbf{x}, \tilde{\phi}_l \rangle|^2 \leq B\|\mathbf{x}\|^2$ . If the frame bounds  $A$  and  $B$  are equal to each other then the frame is said to be tight.

If the system  $\tilde{\Phi}$  is a frame then there exists another frame  $\Phi \stackrel{\text{def}}{=} \{\phi_l\}_{l=0}^{L-1}$  in the space  $\Pi[N]$  such that any signal  $\mathbf{x} \in \Pi[N]$  can be expanded into the sum  $\mathbf{x} = \sum_{l=0}^{L-1} \langle \mathbf{x}, \tilde{\phi}_l \rangle \phi_l$ . The analysis  $\tilde{\Phi}$  and the synthesis  $\Phi$  frames can be interchanged. Together they form the so-called bi-frame. If the frame is tight then  $\Phi$  can be chosen as  $\Phi = c\tilde{\Phi}$ .

If the elements  $\{\tilde{\phi}_l\}$  of the analysis frame  $\tilde{\Phi}$  are not linearly independent ( $L > N$ ) then many synthesis frames can be associated with a given analysis frame. In this case, the expansions  $\mathbf{x} = \sum_{l=0}^{L-1} \langle \mathbf{x}, \tilde{\phi}_l \rangle \phi_l$  provide redundant representations of the signal  $\mathbf{x}$ .

It was established in [12] that the PR filter banks operating in the space  $l_1$  of decaying discrete-time signals generate frames for this space. A similar fact was proved in [32] for the p-filter banks operating in  $\Pi[N]$ .

### 2.6.1 Wavelet frame transform

Assume an analysis  $\tilde{\mathbf{H}} \stackrel{\text{def}}{=} \{\tilde{\mathbf{h}}^s\}$  and a synthesis  $\mathbf{H} \stackrel{\text{def}}{=} \{\mathbf{h}^s\}$ ,  $s = 0, \dots, S-1$ , are p-filter banks with downsampling by factor 2 that form a PR p-filter bank.

Denote

$$\tilde{\psi}_{[1]}^s \stackrel{\text{def}}{=} \left\{ \tilde{\psi}_{[1]}^s[l] = \tilde{h}^s[l] \right\}_{l=0}^{N-1}, \quad \psi_{[1]}^s \stackrel{\text{def}}{=} \left\{ \psi_{[1]}^s[l] = h^s[l] \right\}_{l=0}^{N-1}, \quad s = 0, \dots, S-1, \quad (9)$$

where  $\{\tilde{h}^s[l]\}$  and  $\{h^s[l]\}$  are the impulse responses of the p-filters  $\tilde{\mathbf{h}}^s$  and  $\mathbf{h}^s$ , respectively. Then, Eqs. (6) and (7) imply that  $x[k] = \sum_{s=0}^{S-1} x_{[1]}^s[k]$ , where

$$x_{[1]}^s[k] \stackrel{\text{def}}{=} \sum_{l=0}^{N/2-1} y_{[1]}^s[l] \psi_{[1]}^s[k-2l], \quad y_{[1]}^s[l] = \sum_{\kappa=0}^{N-1} \tilde{\psi}_{[1]}^s[\kappa-2l] x[\kappa] = \langle \mathbf{x}, \tilde{\psi}_{[1]}^s[\cdot-2l] \rangle, \quad (10)$$

$$\iff \mathbf{x} = \sum_{s=0}^{S-1} \mathbf{x}_{[1]}^s = \sum_{s=0}^{S-1} \sum_{l=0}^{N/2-1} \langle \mathbf{x}, \tilde{\psi}_{[1]}^s[\cdot-2l] \rangle \psi_{[1]}^s[\cdot-2l]. \quad (11)$$

Thus, the signal  $\mathbf{x} \in \Pi[N]$  is expanded over the system  $\{\psi_{[1]}^s[\cdot - 2l]\}$ ,  $s = 0, \dots, S-1$ ,  $l = 0, \dots, \frac{N}{2} - 1$ . The expansion coefficients are the inner products  $\{\langle \mathbf{x}, \tilde{\psi}_{[1]}^s[\cdot - 2l] \rangle\}$ .

**Theorem 2.3 ([32])** *If the polyphase matrices  $\tilde{\mathbf{P}}$  and  $\mathbf{P}$  of the  $p$ -filter banks  $\tilde{\mathbf{H}}$  and  $\mathbf{H}$ , respectively, satisfy the PR conditions in Eq. (8), then the 2-sample circular shifts  $\{\tilde{\psi}_{[1]}^s[\cdot - 2l]\}$  and  $\{\psi_{[1]}^s[\cdot - 2l]\}$ ,  $s = 0, \dots, S-1$ ,  $l = 0, \dots, N/2 - 1$  of the signals  $\tilde{\psi}_{[1]}^s$  and  $\psi_{[1]}^s$ ,  $s = 0, \dots, S-1$ , respectively, defined in Eq. (9), form a bi-frame of the space  $\Pi[N]$ . If the relations*

$$\tilde{\mathbf{P}}[n]^T \cdot \tilde{\mathbf{P}}[-n] = c\mathbf{I}_2, \quad n \in \mathbb{Z}, \quad (12)$$

hold then the frame is tight.

The notation  $\cdot^T$  means matrix transposition. If the condition in Eq. (12) is satisfied, then the synthesis filter bank can be chosen to be equal to the analysis filter bank (up to a constant factor).

If  $S > 2$  then the representation in Eq. (11) of a signal from  $\Pi[N]$  is redundant. The redundancy ratio for one-level frame transform is  $\rho = S/2$ .

The signals  $\tilde{\psi}_{[1]}^s$  and  $\psi_{[1]}^s$ ,  $s = 0, \dots, S-1$ , are called the analysis and synthesis discrete-time *framelets* of the first decomposition level, respectively.

### 2.6.2 Multi-level frame transform

Assume that the analysis  $\tilde{\mathbf{H}}_{[\mu]} \stackrel{\text{def}}{=} \{\tilde{\mathbf{h}}_{[\mu]}^s\}$ , and the synthesis  $\mathbf{H}_{[\mu]} \stackrel{\text{def}}{=} \{\mathbf{h}_{[\mu]}^s\}$ ,  $s = 0, \dots, S-1$ , are  $p$ -filter banks with downsampling factor 2 that operate in the subspaces  $\Pi[N/2^\mu]$ . They form a PR  $p$ -filter bank. Their polyphase matrices are  $\tilde{\mathbf{P}}_{[\mu]}[n]$ ,  $\mu = 0, \dots, m-1$ , and  $\mathbf{P}_{[\mu]}[n]$ , respectively. Here,  $\mathbf{H}_{[0]} \stackrel{\text{def}}{=} \mathbf{H}$  and  $\tilde{\mathbf{H}}_{[0]} \stackrel{\text{def}}{=} \tilde{\mathbf{H}}$  denote the  $p$ -filter banks at the initial space  $\Pi[N]$ .  $\tilde{\mathbf{P}}_{[0]}[n]$  and  $\mathbf{P}_{[0]}[n]$  are their polyphase matrices. The simplest way to define the multilevel polyphase matrices is to have  $\tilde{\mathbf{P}}_{[\mu+1]}[n] \stackrel{\text{def}}{=} \tilde{\mathbf{P}}_{[\mu]}[2n]$  and  $\mathbf{P}_{[\mu+1]}[n] \stackrel{\text{def}}{=} \mathbf{P}_{[\mu]}[2n]$ ,  $\mu = 0, \dots, m-1$ .

To increase the redundancy of a signal representation, the frame transform is applied to the low frequency signal  $\mathbf{y}_{[1]}^0 = \{y_{[1]}^0[l]\}$ ,  $y_{[1]}^0[l] = \sum_{k=0}^{N-1} \tilde{\psi}_{[1]}^s[k-2l] x[k]$  that belongs to  $\Pi[N/2]$ . Repeating the above reasoning, we get  $x_{[1]}^0[k] = \sum_{s=0}^{S-1} \sum_{\lambda=0}^{N/4-1} \langle \mathbf{x}, \tilde{\psi}_{[2]}^s[\cdot - 4l] \rangle \psi_{[2]}^s[k-4\lambda]$ , where

$$\tilde{\psi}_{[2]}^s[\kappa] \stackrel{\text{def}}{=} \sum_{\nu=0}^{N/2-1} \tilde{h}_{[1]}^s[\nu] \tilde{\psi}_{[1]}^0[\kappa - 2\nu], \quad \psi_{[2]}^s[k] \stackrel{\text{def}}{=} \sum_{l=0}^{N/2-1} \psi_{[1]}^0[k-2l] h_{[1]}^s[l], \quad s = 0, \dots, S-1.$$

The signals  $\{\tilde{\psi}_{[2]}^s\}$  and  $\{\psi_{[2]}^s\}$  are called the analysis and the synthesis discrete time periodic framelets of the second decomposition level, respectively.

The iterated transform with the polyphase matrices  $\tilde{\mathbf{P}}_{[\mu]}[n]$  and  $\mathbf{P}_{[\mu]}[n]$  leads to the frame expansion of the signal  $\mathbf{x} = \mathbf{x}_{[m]}^0 + \sum_{\mu=1}^m \sum_{s=1}^{S-1} \mathbf{x}_{[\mu]}^s$ , where

$$\begin{aligned} x_{[m]}^s[k] &= \sum_{l \in \mathbb{Z}} \langle \mathbf{x}, \tilde{\psi}_{[m]}^s[\cdot - 2^m l] \rangle \psi_{[m]}^s[k - 2^m l], \quad s = 0, \dots, S-1, \\ x_{[\mu]}^s[k] &= \sum_{l=0}^{N/2^\mu-1} \langle \mathbf{x}, \tilde{\psi}_{[\mu]}^s[\cdot - 2^\mu l] \rangle \psi_{[\mu]}^s[k - 2^\mu l], \quad \mu < m, \quad s = 1, \dots, S-1. \end{aligned} \quad (13)$$

The synthesis and analysis framelets are derived iteratively

$$\psi_{[\mu]}^s[k] \stackrel{\text{def}}{=} \sum_{l=0}^{N/2^{\mu-1}-1} h_{[\mu-1]}^s[l] \psi_{[\mu-1]}^0[k - 2^{\mu-1}l], \quad \tilde{\psi}_{[\mu]}^s[k] \stackrel{\text{def}}{=} \sum_{l=0}^{N/2^{\mu-1}-1} \tilde{h}_{[\mu-1]}^s[l] \tilde{\psi}_{[\mu-1]}^0[k - 2^{\mu-1}l]. \quad (14)$$

### 3 Design of four-channel p-filter banks for frames generation

In this section, we discuss the design of four-channel p-filter banks that generate tight and semi-tight frames in the space  $\Pi[N]$  of periodic signals.

#### 3.1 Four-channel perfect reconstruction p-filter banks

The PR condition for a pair of the analysis  $\tilde{\mathbf{H}} = \{\tilde{\mathbf{h}}^0, \tilde{\mathbf{h}}^1, \tilde{\mathbf{h}}^2, \tilde{\mathbf{h}}^3\}$  and the synthesis  $\mathbf{H} = \{\mathbf{h}^0, \mathbf{h}^1, \mathbf{h}^2, \mathbf{h}^3\}$  p-filter banks is expressed via their polyphase matrices

$$\mathbf{P}[n] \cdot \tilde{\mathbf{P}}[-n] = \begin{pmatrix} \hat{h}_0^0[n]_1 & \hat{h}_0^1[n]_1 & \hat{h}_0^2[n]_1 & \hat{h}_0^3[n]_1 \\ \hat{h}_1^0[n]_1 & \hat{h}_1^1[n]_1 & \hat{h}_1^2[n]_1 & \hat{h}_1^3[n]_1 \end{pmatrix} \cdot \begin{pmatrix} \hat{h}_0^0[-n]_1 & \hat{h}_1^0[-n]_1 \\ \hat{h}_0^1[-n]_1 & \hat{h}_1^1[-n]_1 \\ \hat{h}_0^2[-n]_1 & \hat{h}_1^2[-n]_1 \\ \hat{h}_0^3[-n]_1 & \hat{h}_1^3[-n]_1 \end{pmatrix} = \mathbf{I}_2, \quad n \in \mathbb{Z}. \quad (15)$$

The matrix product in Eq. (15) can be split into two products:

$$\begin{aligned} \mathbf{P}[n] \cdot \tilde{\mathbf{P}}[-n] &= \mathbf{P}^{01}[n] \cdot \tilde{\mathbf{P}}^{01}[-n] + \mathbf{P}^{23}[n] \cdot \tilde{\mathbf{P}}^{23}[-n] = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \\ \mathbf{P}^{01}[n] &\stackrel{\text{def}}{=} \begin{pmatrix} \hat{h}_0^0[n]_1 & \hat{h}_0^1[n]_1 \\ \hat{h}_1^0[n]_1 & \hat{h}_1^1[n]_1 \end{pmatrix}, \quad \tilde{\mathbf{P}}^{01}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{h}_0^0[n]_1 & \hat{h}_1^0[n]_1 \\ \hat{h}_0^1[n]_1 & \hat{h}_1^1[n]_1 \end{pmatrix} \\ \mathbf{P}^{23}[n] &\stackrel{\text{def}}{=} \begin{pmatrix} \hat{h}_0^2[n]_1 & \hat{h}_0^3[n]_1 \\ \hat{h}_1^2[n]_1 & \hat{h}_1^3[n]_1 \end{pmatrix}, \quad \tilde{\mathbf{P}}^{23}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{h}_0^2[n]_1 & \hat{h}_1^2[n]_1 \\ \hat{h}_0^3[n]_1 & \hat{h}_1^3[n]_1 \end{pmatrix}. \end{aligned} \quad (16)$$

According to Theorem 2.3, a PR pair  $\{\mathbf{H}, \tilde{\mathbf{H}}\}$  of p-filter banks generate a tight frame if their polyphase matrices are linked as

$$\mathbf{P}[n] = \tilde{\mathbf{P}}[n]^T \iff \mathbf{P}^{01}[n] = \tilde{\mathbf{P}}^{01}[n]^T \quad \text{and} \quad \mathbf{P}^{23}[n] = \tilde{\mathbf{P}}^{23}[n]^T, \quad n \in \mathbb{Z}.$$

**Definition 3.1** Assume that  $\tilde{\mathbf{P}}[n]$  and  $\mathbf{P}[n]$  are polyphase matrices of the PR pair  $\{\tilde{\mathbf{H}}, \mathbf{H}\}$  (also called bi-frame) of p-filter banks that generate a bi-frame  $\{\tilde{\mathbf{F}}, \mathbf{F}\}$  in the space  $\Pi[N]$ . If the matrices  $\mathbf{P}^{01}[n] = \tilde{\mathbf{P}}^{01}[n]^T$  and  $\mathbf{P}^{23}[n] \neq \tilde{\mathbf{P}}^{23}[n]^T$ ,  $n \in \mathbb{Z}$ , then the frame  $\{\tilde{\mathbf{F}}, \mathbf{F}\}$  is called semi-tight.

Design of a four-channel filter bank begins from a linear phase low-pass filter  $\mathbf{h}^0 = \tilde{\mathbf{h}}^0$ , whose FR  $\hat{h}^0[n] = \hat{h}_0^0[n]_1 + \omega^{-n}\hat{h}_1^0[n]_1$  is assumed to be a rational function of  $\omega^n = e^{2\pi in/N}$  with real coefficients that has no poles for  $n \in \mathbb{Z}$ . In addition, assume that  $\hat{h}^0[n]$  is symmetric about the swap  $n \rightarrow -n$  that implies that  $\hat{h}_0^0[n]_1 = \hat{h}_0^0[-n]_1$  and  $\omega^{-n}\hat{h}_1^0[n]_1 = \omega^n\hat{h}_1^0[-n]_1$ . Consequently, the IR  $\{h^0[k]\}$  is symmetric about  $k = 0$ .

Assume that  $\mathbf{P}^{01}[n] = \tilde{\mathbf{P}}^{01}[n]^T$  and the product

$$\mathbf{P}^{01}[n] \cdot \mathbf{P}^{01}[-n] = \begin{pmatrix} \alpha[n]_1 & 0 \\ 0 & \beta[n]_1 \end{pmatrix} \quad (17)$$

is a diagonal matrix. The assumption in Eq. (17) is equivalent to the condition  $\hat{h}_0^0[n]_1\hat{h}_1^0[-n]_1 + \hat{h}_0^1[n]_1\hat{h}_1^1[-n]_1 = 0$ . The simplest way to satisfy this condition is to define

$$\hat{h}_0^1[n]_1 \stackrel{\text{def}}{=} -\hat{h}_1^0[-n]_1, \quad \hat{h}_1^1[n]_1 \stackrel{\text{def}}{=} \hat{h}_0^0[-n]_1 \implies \alpha[n]_1 = \beta[n]_1 = \left| \hat{h}_0^0[n]_1 \right|^2 + \left| \hat{h}_1^0[n]_1 \right|^2.$$

Due to symmetry of  $\hat{h}^0[n]$ , the FR becomes

$$\begin{aligned}\hat{h}^1[n] &= -\hat{h}_1^0[-n]_1 + \omega^{-n}\hat{h}_0^0[-n]_1 \\ &= \omega^{-n} \left( \hat{h}_0^0[n]_1 - \omega^{-n}\hat{h}_1^0[n]_1 \right) = \omega^{-n}\hat{h}^0[n + N/2].\end{aligned}\quad (18)$$

Equation (18) implies that the sequence  $\omega^{n\hat{h}^1[n]}$  is symmetric about the swap  $n \rightarrow -n$  and, consequently, the IR  $\{h^1[k]\}$  is symmetric about  $k = 1$ .

It follows from the assumption in Eq. (17) that the product

$$\mathbf{P}^{23}[n] \cdot \tilde{\mathbf{P}}^{23}[-n] = \mathbf{Q}[n] \stackrel{\text{def}}{=} \begin{pmatrix} t[n]_1 & 0 \\ 0 & t[n]_1 \end{pmatrix} \quad (19)$$

is a diagonal matrix where  $t[n]_1 \stackrel{\text{def}}{=} 1 - \left| \hat{h}_0^0[n]_1 \right|^2 + \left| \hat{h}_1^0[n]_1 \right|^2$ . Thus, design of the PR p-filter bank is reduced to factorization of the matrix  $\mathbf{Q}[n]$ .

### 3.1.1 Diagonal factorization of the matrix $\mathbf{Q}[n]$

There are many ways to factorize the matrix  $\mathbf{Q}[n]$ . One way is to define the matrices  $\mathbf{P}^{23}[n]$  and  $\tilde{\mathbf{P}}^{23}[n]$  to be diagonal:

$$\mathbf{P}^{23}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{h}_0^2[n]_1 & 0 \\ 0 & \hat{h}_1^3[n]_1 \end{pmatrix}, \quad \tilde{\mathbf{P}}^{23}[n] \stackrel{\text{def}}{=} \begin{pmatrix} \hat{h}_0^2[n]_1 & 0 \\ 0 & \hat{h}_1^3[n]_1 \end{pmatrix},$$

which means that the odd polyphase components of the p-filters  $\tilde{\mathbf{h}}^2$  and  $\mathbf{h}^2$  as well as the even polyphase components of the p-filters  $\tilde{\mathbf{h}}^3$  and  $\mathbf{h}^3$  vanish. Consequently, we have to find four sequences  $\hat{h}_0^2[n]_1, \hat{h}_0^2[-n]_1, \hat{h}_1^3[n]_1$  and  $\hat{h}_1^3[-n]_1$  such that

$$\hat{h}_0^2[n]_1 \hat{h}_0^2[-n]_1 = \hat{h}_1^3[n]_1 \hat{h}_1^3[-n]_1 = t[n]_1. \quad (20)$$

**Tight frame p-filter banks:** If the following inequality holds

$$\alpha[n]_1 \stackrel{\text{def}}{=} |h_0^0[n]_1|^2 + |h_1^0[n]_1|^2 \leq 1, \quad n \in \mathbb{Z}, \quad (21)$$

then, due to symmetry of the rational functions  $\hat{h}_0^0[n]_1$  and  $\hat{h}_1^0[n]_1$ , we have

$1 - a[n]_1 = P(\cos 2\pi n/N)/R(\cos 2\pi n/N)$ , where  $R$  is strictly positive and  $P$  is non-negative polynomials. Due to Riesz's Lemma [26], the polynomials can be factorized as  $P(\cos 2\pi n/N) = p(\omega^n)p(\omega^{-n})$ ,  $R(\cos 2\pi n/N) = q(\omega^n)q(\omega^{-n})$ , where  $p$  and  $q$  are polynomials with real coefficients and  $q(\omega^n)$  does not have roots for  $n \in \mathbb{Z}$ . Thus, we can define  $\hat{h}_0^2[n]_1 = \hat{h}_0^2[-n]_1 = \hat{h}_1^3[n]_1 = \hat{h}_1^3[-n]_1 = T[n]_1 \stackrel{\text{def}}{=} p(\omega^n)/q(\omega^n)$ . The PR p-filter bank, whose FRs are

$$\begin{aligned}\hat{h}^0[n] &= \hat{h}_0^0[n]_1 + \omega^{-n}\hat{h}_0^0[n]_1, & \hat{h}^2[n] &= T[n]_1, \\ \hat{h}^1[n] &= -\hat{h}_1^0[-n]_1 + \omega^{-n}\hat{h}_0^0[-n]_1, & \hat{h}^3[n] &= \omega^{-n}T[-n]_1,\end{aligned}$$

generates a tight wavelet frame in the space  $\Pi[N]$ . Certainly, symmetry of the FR  $\hat{h}^0[n]$  does not guarantee symmetry of the FRs  $\hat{h}^2[n]$  and  $\hat{h}^3[n]$ .

**Semi-tight frame p-filter banks:** If the condition in Eq. (21) is not satisfied then the sequence  $t[n]_1$  can be factorized to  $t[n]_1 = T[n]_1 \tilde{T}[-n]_1$ , where  $T[n]_1 \neq \tilde{T}[n]_1$ . Thus, we obtain the PR p-filter bank, whose FRs are

$$\begin{aligned}\hat{h}^0[n] &= \hat{h}_0^0[n]_1 + \omega^{-n}\hat{h}_0^0[n]_1, & \hat{\gamma}^1[n] &= -\hat{h}_1^0[-n]_1 + \omega^{-n}\hat{h}_0^0[-n]_1, \\ \hat{h}^2[n] &= T^2[n]_1, & \hat{\tilde{h}}^2[n] &= \tilde{T}^2[n]_1, \\ \hat{h}^3[n] &= \omega^{-n}T^3[n]_1, & \hat{\tilde{h}}^3[n] &= \omega^{-n}\tilde{T}^3[n]_1.\end{aligned}\quad (22)$$

where  $T^2[n]_1 \tilde{T}^2[-n]_1 = T^3[n]_1 \tilde{T}^3[-n]_1 = t[n]_1$ . The PR p-filter bank defined by Eq. (22) generates a semi-tight frame in the space  $\Pi[N]$ .

**Remark 3.1** *The rational function  $t[n]_1$  of  $\omega^n$  is symmetric about the change  $n \rightarrow -n$ . Therefore, it can be factorized into product of two symmetric rational functions  $T[n]_1$  and  $\tilde{T}[-n]_1$ . An additional advantage of the semi-tight design is the option to swap approximation properties between analysis and synthesis framelets.*

### 3.1.2 Symmetric factorization of the matrix $\mathbf{Q}[n]$

If the inequality in Eq. (21) holds then the sequence  $t[n]_1$  can be factorized as  $t[n]_1 = T[n]_1 T[-n]_1$  and the following factorization of the matrix  $\mathbf{Q}[n]$  is possible. Define,

$$\mathbf{P}^{23}[n] \stackrel{\text{def}}{=} \begin{pmatrix} A[-n]_1 & -A[-n]_1 \\ A[n]_1 & A[n]_1 \end{pmatrix} \quad \tilde{\mathbf{P}}^{23}[n] \stackrel{\text{def}}{=} \mathbf{P}^{23}[n]^T, \quad (23)$$

where  $A[n]_1 \stackrel{\text{def}}{=} T[n]_1/\sqrt{2}$ . Then, the product

$$\mathbf{P}^{23}[n] \cdot \tilde{\mathbf{P}}^{23}[-n] = \begin{pmatrix} A[-n]_1 & -A[-n]_1 \\ A[n]_1 & A[n]_1 \end{pmatrix} \cdot \begin{pmatrix} A[n]_1 & A[-n]_1 \\ -A[-n]_1 & A[-n]_1 \end{pmatrix} = \mathbf{Q}[n].$$

A consequence of this choice is the fact that the p-filters  $\mathbf{h}^2$  and  $\mathbf{h}^3$  have a linear phase:

$$\begin{aligned} \hat{h}^2[n] &= A[-n]_1 + \omega^{-n} A[n]_1, & \hat{h}^2[-n] &= A[n]_1 + \omega^n A[-n]_1 = \omega^n \hat{h}^2[n], \\ \hat{h}^3[n] &= -A[-n]_1 + \omega^{-n} A[n]_1, & \hat{h}^3[-n] &= -A[n]_1 + \omega^n A[-n]_1 = -\omega^n \hat{h}^3[n]. \end{aligned}$$

The following symmetry properties hold.

**Proposition 3.2** *The impulse response (IR) of the p-filter  $\mathbf{h}^2$  is symmetric about  $1/2$ , while the IR of  $\mathbf{h}^3$  is antisymmetric about  $1/2$ .*

**Proof:** The IR of the p-filter  $\mathbf{h}^2$  is  $h^2[k] = N^{-1} \sum_{n=0}^{N-1} \omega^{kn} \hat{h}^2[n]$ . On the other hand,

$$h^2[-k+1] = \frac{1}{N} \sum_{n=0}^{N-1} \omega^{-kn} \omega^n \hat{h}^2[n] = \frac{1}{N} \sum_{n=0}^{N-1} \omega^{kn} \hat{h}^2[-n] = \frac{1}{N} \sum_{n=0}^{N-1} \omega^{kn} \hat{h}^2[n] = h^2[k].$$

The claim about the p-filter  $\mathbf{h}^3$  is proved similarly. ■

Since  $\tilde{\mathbf{P}}^{23}[n] = \mathbf{P}^{23}[n]^T$ , the four-channel p-filter bank

$$\begin{aligned} \hat{h}^0[n] &= \hat{h}_0^0[n]_1 + \omega^{-n} \hat{h}_1^0[n]_1, & \hat{h}^1[n] &= -\hat{h}_1^0[-n]_1 + \omega^{-n} \hat{h}_0^0[-n]_1, \\ \hat{h}^2[n] &= A[-n]_1 + \omega^{-n} A[n]_1, & \hat{h}^3[n] &= -A[-n]_1 + \omega^{-n} A[n]_1. \end{aligned}$$

generates a tight frame in the space  $\Pi[N]$ .

The above symmetric factorization scheme is, in essence, similar to the scheme presented in [8]. Certainly, other schemes are possible.

### 3.1.3 Interpolating p-filter banks

Assume that the low-pass p-filter  $\mathbf{h}^0$  is interpolating and its frequency response is  $\hat{h}^0[n] = (1 + \omega^{-n} \hat{f}[n]_1) / \sqrt{2}$ , where the sequence  $\hat{f}[n]_1$  is a rational function of  $\omega^n = e^{2\pi i n/N}$  that has no poles for  $n \in \mathbb{Z}$  and

$$f[0]_1 = 1, \quad \omega^{-n} \hat{f}[n]_1 = \omega^n \hat{f}[-n]_1. \quad (24)$$

Then, the sequence  $t[n]_1$  is factorized to become

$$t[n]_1 = \frac{W[n]_1}{2}, \quad W[n]_1 \stackrel{\text{def}}{=} 1 - |\hat{f}[n]_1|^2 = 2\hat{h}^0[n]\hat{h}^0[n + N/2].$$

When the diagonal factorization of the matrix  $\mathbf{Q}[n]$  is applied, the polyphase submatrices are

$$\mathbf{P}^{01}[n] = \tilde{\mathbf{P}}^{01}[n]^T = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & \hat{f}[n]_1 \\ -\hat{f}[-n]_1 & 1 \end{pmatrix}, \quad (25)$$

$$\mathbf{P}^{23}[n] = \begin{pmatrix} T^2[n]_1 & 0 \\ 0 & T^3[n]_1 \end{pmatrix} \quad \tilde{\mathbf{P}}^{23}[n] = \begin{pmatrix} \tilde{T}^2[n]_1 & 0 \\ 0 & \tilde{T}^3[n]_1 \end{pmatrix}, \quad (26)$$

where  $T^2[n]_1 \tilde{T}^2[-n]_1 = T^3[n]_1 \tilde{T}^3[-n]_1 = W[n]_1/2$ .

In the case when the inequality Eq. (21) holds and the sequence  $t[n]_1$  is factorized as  $t[n]_1 = T[n]_1 T[-n]_1$ , the polyphase matrices  $\mathbf{P}^{01}[n] = \tilde{\mathbf{P}}^{01}[n]^T$  are given in Eq. (25), while the matrices  $\mathbf{P}^{23}[n] = \tilde{\mathbf{P}}^{23}[n]^T$  are given in Eq. (23).

The key point in the design of a four-channel PR p-filter bank that generates a (semi-)tight frame in the signal space  $\Pi[N]$  is in the definition of a relevant low-pass p-filter  $\mathbf{h}^0 = \hat{\mathbf{h}}^0$ . Once we have a low-pass filter, the rest of the p-filters is designed in a way described above. Prior to the presentation of a family of the p-filters originating from polynomial and discrete splines, we briefly discuss how the concepts of the polynomials restoration and vanishing moments can be adapted to the periodic discrete-time setting.

## 4 Restoration of sampled polynomials and discrete vanishing moments

Restoration of polynomials by low-pass filtering coupled with their elimination by respective high- and band-pass filters, which constitute filter banks (vanishing moment property), provide a sparse representation of signals/images in wavelet and frame bases. This is important, for example, for data compression and signals/images restoration.

**Restoration of sampled polynomials** Certainly, sampled polynomials do not belong to  $\Pi[N]$  and p-filters cannot be applied to them. However, the following fact holds, which, in a sense, is a discrete periodic counterpart of the classical Fix-Strang condition [31].

**Proposition 4.1** *Assume that the FR of a low-pass filter  $\mathbf{h}$  can be represented as  $\hat{h}[n] = \hat{h}[0] + (\omega^n - 1)^m \alpha[n]$ , where  $m$  is some natural number and  $\alpha[n]$  is a rational function of  $\omega^n$ , which has no poles for  $n \in \mathbb{Z}$  and  $\alpha[0] \neq 0$ . Assume  $\mathbf{p}$  is a signal from  $\Pi[N]$ , which coincides with a sampled polynomial  $\mathbf{P}_{m-1}$  of degree  $m-1$  at some interval  $p[k] = P_{m-1}[k]$ ,  $k = k_0, \dots, k_m$ , where  $m < k_m - k_0 < N$ . Then,  $\sum_{l=0}^{N-1} h[k-l]p[l] = \hat{h}[0]P_{m-1}[k]$ , as  $k = k_0, \dots, k_m - m - 1$ .*

**Proof:** Equation (1) implies that the sequence  $(\omega^n - 1)^m$  is the DFT of the finite difference and  $(\omega^n - 1)^m \hat{x}[n] = \hat{\Delta}^m[\mathbf{x}][n]$ . Under conditions of the proposition, the sequence  $\{\alpha[n]\}$  can be regarded as the FR of a low-(all)-pass p-filter  $\mathbf{a}$ . Then, application of the p-filter  $\mathbf{h}$  to a signal  $\mathbf{x} \in \Pi[N]$  can be represented as

$$\mathbf{h}\mathbf{x} = \hat{h}[0]\mathbf{x} + \mathbf{a}\Delta^m[\mathbf{x}]. \quad (27)$$

The finite difference of the signal  $\mathbf{p}$  is

$$\Delta^m[\mathbf{p}][k] = \sum_{l=0}^m (-1)^l \binom{m}{l} p[k+l] = \Delta^m[\mathbf{P}_{m-1}][k], \text{ as } k = k_0, \dots, k_m - m - 1.$$

According to Proposition 2.1,  $\Delta^m[\mathbf{P}_{m-1}][k] = 0$ . Thus, the statement follows from Eq. (27). ■

**Definition 4.2** If a low-pass  $p$ -filter  $\mathbf{h}$  satisfies the conditions of Proposition 4.1, then we say that the  $p$ -filter  $\mathbf{h}$  locally restores sampled polynomials of degree  $m - 1$ .

### Discrete vanishing moments

**Proposition 4.3** Assume that the FR of a high(band)-pass filter  $\mathbf{g}$  is represented by  $\hat{g}[n] = (\omega^n - 1)^m \alpha[n]$ , where  $m$  is a natural number and  $\alpha[n]$  is a rational function of  $\omega^n$ , which has no poles for  $n \in \mathbb{Z}$  and  $\alpha[0] \neq 0$ . Assume  $\mathbf{p}$  is a signal from  $\Pi[N]$ , which coincides with a sampled polynomial  $\mathbf{P}_{m-1}$  of degree  $m - 1$  at the interval  $p[k] = P_{m-1}[k]$ ,  $k = k_0, \dots, k_m$ , where  $m < k_m - k_0 < N$ . Then,  $\sum_{l=0}^{N-1} g[k-l] p[l] = 0$ ,  $k = k_0, \dots, k_m - m - 1$ .

**Proof:** is similar to the proof of Proposition 4.1.

**Definition 4.4** If a high(band)-pass  $p$ -filter  $\mathbf{g}$  satisfies the conditions of Proposition 4.3, we say that the  $p$ -filter  $\mathbf{g}$  locally eliminates sampled polynomials of degree  $m - 1$ . If a framelet  $\psi_{[1]}[k] \stackrel{\text{def}}{=} g[k]$ ,  $k \in \mathbb{Z}$ , we say that the framelet  $\psi_{[1]}$  has  $m$  local discrete vanishing moments (LDVM).

**Remark 4.1** The statements of Propositions 4.1 and 4.3 concerning the low-pass  $\mathbf{h}$  and high-pass  $\mathbf{g}$  filters, respectively, remain true if the frequency responses are represented as

$$\hat{h}[n] = \hat{h}[0] + \sin^m \frac{\pi n}{N} \bar{\alpha}[n], \quad \hat{g}[n] = \sin^m \frac{\pi n}{N} \bar{\alpha}[n],$$

where  $m$  is a natural number and  $\bar{\alpha}[n]$  is a rational function of  $\omega^n$  that has no poles for  $n \in \mathbb{Z}$  and  $\bar{\alpha}[0] \neq 0$ .

## 5 Spline-based low- and high-pass $p$ -filters

**The idea behind the design** Assume that the FR of a low-pass interpolating  $p$ -filter  $\mathbf{h}^0$  operating in  $\Pi[N]$  is described by  $\tilde{h}^0[n] = (1 + \omega^{-n} \hat{f}[n]_1) / \sqrt{2}$ . The FR of the corresponding high-pass  $p$ -filter  $\mathbf{h}^1$  is  $\tilde{h}^1[n] = \omega^{-n} (1 - \omega^{-n} \hat{f}[n]_1) / \sqrt{2}$ . Then, the DFT of the signal  $\mathbf{y} = \mathbf{h}^1 \mathbf{x}$ , where  $\mathbf{x} \in \Pi[N]$ , is

$$\hat{y}[n] = \frac{1}{\sqrt{2}} \left( -\hat{f}[-n]_1 \hat{x}_0[n]_1 + \omega^{-2n} \hat{x}_1[n]_1 \right) + \frac{\omega^{-n}}{\sqrt{2}} \left( \hat{x}_0[n]_1 - \hat{f}[-n]_1 \hat{x}_1[n]_1 \right).$$

Denote by  $\mathbf{f}$  the  $p$ -filter in  $\Pi[N/2]$  whose FR is  $\hat{f}[-n]_1$  and by  $\bar{\mathbf{x}}_1 \stackrel{\text{def}}{=} \{x[2k-1]\}$  the shifted odd polyphase component, while  $\mathbf{x}_1 \stackrel{\text{def}}{=} \{x[2k+1]\}$ . Then, the polyphase components of  $\mathbf{y}$  are  $\mathbf{y}_0 = (\bar{\mathbf{x}}_1 - \mathbf{f} \mathbf{x}_0) / \sqrt{2}$ ,  $\mathbf{y}_1 = (\mathbf{x}_0 - \mathbf{f} \mathbf{x}_1) / \sqrt{2}$ .

In order for the high-pass filter to eliminate smooth signals (for example, fragments of polynomials), the  $p$ -filter  $\mathbf{f}$  should be “predictive” in a sense that its application to the even polyphase component of the signal should “predict” the odd polyphase component and vice versa.

Polynomial and discrete splines are natural sources to derive prediction  $p$ -filters from. The idea is to construct either a polynomial or a discrete spline, which (quasi-)interpolates the even samples of a signal, and to predict the odd samples of the signal by spline’s values at the midpoints between the (quasi-)interpolation points. Such an approach was explored for the design of the biorthogonal wavelet transforms in [2, 5, 4]. The prediction filters derived from the discrete splines were used in [6, 7] for the design of non-periodic interpolating wavelet frames with three-channel filter banks.

### 5.1 Prediction $p$ -filters derived from polynomial splines

A polynomial spline of order  $p$  on the grid  $\{t_k\}$  is a function that has  $p - 2$  continuous derivatives. The spline of order  $p$  coincides at the intervals  $(t_k, t_{k+1})$  between the grid points with polynomials  $\mathbf{P}_k^{p-1}$  of degree  $p - 1$ .

### 5.1.1 Periodic interpolating splines

Denote by  $\mathbf{S}_K^p$  the space of splines of order  $p \in \mathbb{N}$  defined on the uniform grid  $\{k\}$ , which are periodic with the period  $K = N/2 = 2^{j-1}$ . A basis in this space is constituted from shifts of B-splines.

The first-order non-periodic B-spline  $\beta^1(t)$  on the grid  $\{k\}$  is the indicator function on the interval  $(-1/2, 1/2)$ . The B-spline of order  $p$  is derived by the iterated convolution  $\beta^1(t) \stackrel{\text{def}}{=} \beta^1(t) \star \beta^{p-1}(t)$ . It is supported on the interval  $(-p/2, p/2)$ , is strictly positive inside this interval and is symmetric about zero where it has its single maximum. There exists an explicit representation of the B-spline of order  $p$ :

$$\beta^p(t) = \frac{1}{(p-1)!} \sum_{k=0}^p (-1)^k \binom{p}{k} \left(t + \frac{p}{2} - k\right)_+^{p-1}, \quad t_+ \stackrel{\text{def}}{=} \begin{cases} t, & \text{if } t \geq 0; \\ 0, & \text{otherwise.} \end{cases} \quad (28)$$

The  $K$ -periodic B-spline  $B^p(t)$ ,  $B^p(t) \stackrel{\text{def}}{=} \sum_{l \in \mathbb{Z}} \beta^p(t + Kl)$  is the periodization of the compactly supported function  $\beta^p(t)$ .

Any spline  $S^p(t)$  from  $\mathbf{S}_K^p$  is represented by

$$S^p(t) = \sum_{k=0}^{K-1} q[k] B^p(t - k). \quad (29)$$

If the spline  $S^p(t)$  interpolates the even polyphase component  $\mathbf{x}_0 = \{x[2l]\}$  of a signal  $\mathbf{x} \in \Pi[N]$  then its coefficients  $q[k]$  can be explicitly calculated via the application of DFT

$$\begin{aligned} S^p(l) = \sum_{k=0}^{K-1} q[k] B^p(l - k) = x_0[l] &\iff \hat{q}[n]_1 u^p[n]_1 = \hat{x}_0[n]_1, \\ u^p[n]_1 \stackrel{\text{def}}{=} \sum_{k=0}^{K-1} \omega^{2nk} B^p(k) > 0 &\implies \hat{q}[n]_1 = \hat{x}_0[n]_1 / u^p[n]_1. \end{aligned} \quad (30)$$

The  $K = N/2$ -periodic sequence  $u^p[n]_1$ , which is the DFT of the sampled B-spline  $\{B^p(k)\}$ , is strictly positive. Due to symmetry and positiveness of the B-spline,  $u^p[n]_1 = u^p[-n]_1$ , thus, it is a cosine polynomial with positive coefficients. It is symmetric about  $K/2$ , where it reaches its single minimum. The maximal value of  $u^p[n]_1$  is 1, being reached when  $n = 0$ .

Approximation properties of interpolating splines are well investigated. In particular, the non-periodic interpolating spline of order  $p$ , which consists of piecewise polynomials of degree  $p - 1$ , restores these polynomials. It means that splines of order  $p$ , which interpolate a polynomial of degree  $p - 1$ , coincides with this polynomial. For periodic splines, this property holds locally. This observation justifies the choice of interpolating splines as a source for the design of prediction filters.

To be specific, if the spline  $S^p(t) \in \mathbf{S}_K^p$ , given in Eq. (29), interpolates the even polyphase component  $\mathbf{x}_0 = \{x[2l]\}$  of a signal  $\mathbf{x} \in \Pi[N]$ , then the odd polyphase component  $\mathbf{x}_1 = \{x[2l + 1]\}$  is predicted by the midpoint values of the spline  $\sigma[l] \stackrel{\text{def}}{=} S^p(l + 1/2) = \sum_{k=0}^{K-1} q[k] B^p(l + 1/2 - k)$ , whose DFT is  $\hat{\sigma}[n]_1 = \hat{q}[n]_1 v^p[n]_1 = \frac{v^p[n]_1}{u^p[n]_1} \hat{x}_0[n]_1$ , where  $v^p[n]_1 \stackrel{\text{def}}{=} \sum_{k=0}^{K-1} \omega^{2nk} B^p(l + 1/2)$  is the DFT of the B-spline  $\{B^p(k + 1/2)\}$  sampled at midpoints between grid points.

The continuous counterparts of the sequences  $u^p[n]_1$  and  $v^p[n]_1$ , which are the discrete-time Fourier transforms of the non-periodic B-splines, were introduced and studied in [29]. Some additional properties of the sequences  $u^p[n]_1$  and  $v^p[n]_1$  are established in [2, 34]. Denote

$$\hat{f}_c^p[n]_1 \stackrel{\text{def}}{=} \frac{v^p[n]_1}{u^p[n]_1}. \quad (31)$$

Obviously, the sequence  $\hat{f}_c^p[n]_1$  satisfies the conditions in Eq. (24).

**Proposition 5.1 ([2])** *If the spline order is either  $p = 2r$  or  $p = 2r - 1$  then*

$$\hat{h}^0[n] \stackrel{\text{def}}{=} \frac{1}{\sqrt{2}} \left(1 + \omega^{-n} \hat{f}_c^p[n]_1\right) = \hat{h}[0] + \sin^{2r} \frac{\pi n}{N} \chi^p[n] \quad (32)$$

$$\hat{h}^1 \stackrel{\text{def}}{=} \frac{\omega^{-n}}{\sqrt{2}} \left(1 - \omega^{-n} \hat{f}_c^p[-n]_1\right) = \sin^{2r} \frac{\pi n}{N} \gamma^p[n], \quad (33)$$

where  $r$  is a natural number and  $\chi^p[n]$  and  $\gamma^p[n]$  are rational functions of  $\omega^n$ , which have no poles for  $n \in \mathbb{Z}$  and no root at  $n = 0$ .

This proposition coupled with Propositions 4.1, 4.3 and Remark 4.1 imply the following Corollary.

**Corollary 5.2** *The low-pass  $\mathbf{h}^0$  and the high-pass  $\mathbf{h}^1$   $p$ -filters, whose FR are given in Eqs. (32) and (33), respectively, locally restore and eliminate sampled polynomials of degree  $2r - 1$ , respectively.*

Therefore, the  $p$ -filter  $\mathbf{f}_c^p$ , whose FR  $f_c^p[n]_1$  is defined by Eq. (31), is a proper candidate to be utilized as a prediction  $p$ -filter.

**Remark 5.1** *It is emphasized that the  $p$ -filters derived from the odd order  $2r - 1$  splines restore (eliminate) sampled polynomials of the same  $2r - 1$  degree as the  $p$ -filters derived from the even order  $2r$  splines. This is a consequence of the so-called super-convergence property of the odd order interpolating splines [34], which claims that the approximation order of such splines at the midpoints between the interpolation points is higher than those at the remaining points of the intervals between interpolation points.*

### 5.1.2 Examples of spline based $p$ -filters

The sequences  $\{v^p[n]_1\}$  and  $\{u^p[n]_1\}$ , whose ratio constitute the FR  $\hat{f}_c^p[n]_1$  of a prediction  $p$ -filter  $\mathbf{f}_c^p$ , are calculated explicitly via the application of DFT to the B-splines sampled at the points  $\{k + 1/2\}$  and  $\{k\}$ , respectively. The samples of the B-splines of any order are readily calculated from Eq. (28).

**Linear spline  $p = 2$ :**  $u^2[n]_1 = 1$ ,  $v^2[n]_1 = (1 + \omega^{2n})/2$ ,  $\hat{f}_c^2[n]_1 = (1 + \omega^{2n})/2$ ,

$$\hat{h}^0[n] \stackrel{\text{def}}{=} \frac{1}{\sqrt{2}} \left( 1 + \omega^{-n} \hat{f}_c^2[n]_1 \right) = \sqrt{2} \cos^2 \frac{\pi n}{N}, \quad \hat{h}^1[n] = \omega^{-n} \sqrt{2} \sin^2 \frac{\pi n}{N}. \quad (34)$$

**Quadratic interpolating spline  $p = 3$ :**

$$\begin{aligned} u^3[n]_1 &= \frac{\omega^{2n} + 6 + \omega^{-2n}}{8} = 1 - \frac{1}{2} \sin^2 \frac{2\pi n}{N} = \cos^4 \frac{\pi n}{N} + \sin^4 \frac{\pi n}{N}, \quad v^3[n]_1 = \frac{1 + \omega^{2n}}{2}, \\ \hat{f}_c^3[n]_1 &= 4 \frac{1 + \omega^{2n}}{\omega^{2n} + 6 + \omega^{-2n}} = \omega^n \frac{\cos^4 \pi n/N - \sin^4 \pi n/N}{\cos^4 \pi n/N + \sin^4 \pi n/N}, \\ \hat{h}^0[n] &= \sqrt{2} \frac{\cos^4 \pi n/N}{\cos^4 \pi n/N + \sin^4 \pi n/N}, \quad \hat{h}^1[n] = \omega^{-n} \sqrt{2} \frac{\sin^4 \pi n/N}{\cos^4 \pi n/N + \sin^4 \pi n/N}. \end{aligned} \quad (35)$$

**Cubic interpolating spline  $p = 4$ :**

$$\begin{aligned} \hat{f}_c^4[n]_1 &= \frac{\omega^{4n} + 23\omega^{2n} + 23 + \omega^{-2n}}{8(\omega^{2n} + 4 + \omega^{-2n})}, \\ \hat{h}^0[n] &= \sqrt{2} \frac{\cos^4 \pi n/N (2 + \cos 2\pi n/N)}{2 + \cos 4\pi n/N}, \quad \hat{h}^1[n] = \omega^{-n} \sqrt{2} \frac{\sin^4 \pi n/N (2 + \cos 2\pi n/N)}{2 + \cos 4\pi n/N}. \end{aligned} \quad (36)$$

**Comment:** We observe that the high-pass  $p$ -filters  $\mathbf{h}^1$  derived from either quadratic or cubic splines locally eliminate sampled cubic polynomials. However, the structure of the quadratic FR is much simpler than the cubic FR. Therefore, in a number of applications, the  $p$ -filters derived from quadratic spline are advantageous over the cubic spline  $p$ -filters.

**Interpolating spline of fourth degree  $p = 5$ :**

$$\hat{f}_c^5[n]_1 = 16 \frac{\omega^{4n} + 11\omega^{2n} + 11 + \omega^{-2n}}{\omega^{4n} + 76\omega^{2n} + 230 + 76\omega^{-2n} + \omega^{-4n}}. \quad (37)$$

It is readily verified that the low- and high-pass  $p$ -filters locally restore and eliminate sampled polynomials of fifth degree, respectively.

The  $p$ -filters, originated from higher order splines, are designed by the application of the DFT to the sampled B-splines.

## 5.2 Prediction p-filters derived from discrete splines

We can see from the above examples that the structure of the spline-based p-filters becomes more complicated as far as the generating spline's order increases, which is necessary for restoration(elimination) of sampled polynomials of higher degrees. However, there is a way to explicitly design a family of prediction p-filters, which provide low- and high-pass p-filters, which restore (eliminate) sampled polynomials of higher degrees. Thus, the corresponding framelets can have any number of LDVMs.

The design scheme is, in essence, similar to the scheme given in Section 5.1.1. The difference is that it is based on the discrete rather than on the polynomial interpolating splines. The discrete splines are the signals from  $\Pi[N]$ , whose properties mimic the properties of the polynomial splines. The discrete splines and the p-filters' design are described in full details in [4] while in the current paper we summarize the result.

Define the prediction p-filters  $\mathbf{f}_d^{2r}$  via the frequency responses:

$$\hat{f}_d^{2r}[n]_1 \stackrel{\text{def}}{=} \frac{\omega^n (\cos^{2r} \pi n/N - \sin^{2r} \pi n/N)}{\cos^{2r} \pi n/N + \sin^{2r} \pi n/N}. \quad (38)$$

Obviously,  $\omega^{-n} \hat{f}_d^{2r}[n]_1 = \omega^n \hat{f}_d^{2r}[-n]_1$ , the value  $\hat{f}_d^{2r}[0]_1 = 1$ , and

$$1 - \omega^{-n} \hat{f}_d^{2r}[n]_1 = \frac{\sin^{2r} \pi n/N}{\cos^{2r} \pi n/N + \sin^{2r} \pi n/N}. \quad (39)$$

If the p-filter  $\mathbf{f}_d^{2r}$ , whose frequency response is given in Eq. (38), is used as a prediction filter in the lifting scheme then the PR p-filter bank is

$$\begin{aligned} \hat{h}^0[n] &\stackrel{\text{def}}{=} \frac{1}{\sqrt{2}} \left( 1 + \omega^{-n} \hat{f}_d^{2r}[n]_1 \right) = \frac{\sqrt{2} \cos^{2r} \pi n/N}{\cos^{2r} \pi n/N + \sin^{2r} \pi n/N} \\ \hat{h}^1[n] &\stackrel{\text{def}}{=} \frac{\omega^{-n}}{\sqrt{2}} \left( 1 - \omega^n \hat{f}_d^{2r}[-n]_1 \right) = \frac{\omega^{-n} \sqrt{2} \sin^{2r} \pi n/N}{\cos^{2r} \pi n/N + \sin^{2r} \pi n/N}. \end{aligned} \quad (40)$$

All the p-filters have a linear phase. Except for the simplest case  $r = 1$ , the impulse responses of the p-filters are infinite. Nevertheless, they are well localized in time domain. Note, that the prediction p-filter  $\mathbf{f}_d^4$  coincides with the p-filter  $\mathbf{f}_c^3$  originating from the quadratic polynomial spline (see Eq. (35)).

The following proposition is the consequence of Eq. (39).

**Proposition 5.3** *The low-pass p-filters  $\mathbf{h}^0$  derived from the prediction p-filter  $\mathbf{f}_d^{2r}$  by Eqs. (40) locally restore sampled polynomials of degree  $2r - 1$ . The high-pass p-filters  $\mathbf{h}^1$  locally eliminate such polynomials. Consequently, the framelets  $\psi_{[1]}^1[k] = h^1[k]$  have  $2r$  LDVMs.*

We emphasize that Eq. (40) provides an explicit expression for p-filters with an arbitrary approximation accuracy.

**Remark 5.2** *The sequences  $\hat{h}[n]$  and  $\omega^n \hat{h}^1[n]$  are the frequency responses of the periodic half-band low- and high-pass Butterworth filters, respectively ([25]).*

**Remark 5.3** *One-pass non-periodic Butterworth filters were used in [21] for the design of orthogonal non-symmetric wavelets. The computations in [21] were conducted in time domain using recursive filtering. Biorthogonal periodic wavelets derived from discrete splines are presented in [4], while non-periodic biorthogonal wavelets are introduced in [5].*

## 5.3 Examples of non-interpolating p-FIR p-filters

The IR of all the p-filters introduced in Sections 5.1 and 5.2, except for the linear spline p-filters, are infinite, i.e. they occupy the whole  $\mathbb{Z}$ . In this section, we introduce a few p-filters, whose IR are finite up to periodization (p-FIR p-filters). One such p-filters is originated from a linear interpolating spline. Another way to design the p-FIR p-filters is to use the splines, which are quasi-interpolating quasi-interpolate rather

than interpolating the even polyphase component of a signal  $\mathbf{x}$  and to predict the odd polyphase component by the spline's values at midpoints between grid nodes. The quasi-interpolating splines are studied in [3, 33]. We describe a design of the p-FIR p-filters originated from quadratic splines. Design of p-filters using quasi-interpolating splines of higher order is similar but their IR is much longer.

### 5.3.1 Quadratic quasi-interpolating spline

The quadratic spline, which interpolates the even polyphase component  $\mathbf{x}_0$  of a signal  $\mathbf{x}$ , is represented as  $S^3(t) = \sum_{k=0}^{N/2-1} q[k] B^3(t-k)$ . Due to Eqs. (30) and (35), the DFT of the coefficients is

$$\hat{q}[n]_1 = \frac{\hat{x}_0[n]_1}{u^3[n]_1} = \frac{\hat{x}_0[n]_1}{1 - \frac{1}{2} \sin^2 2\pi n/N} = \hat{x}_0[n]_1 \sum_{l=0}^{\infty} \frac{1}{2^l} \sin^{2k} \frac{2\pi n}{N}. \quad (41)$$

Two initial terms from the series in Eq. (41) are taken and denoted as

$$U^3[n]_1 \stackrel{\text{def}}{=} 1 + \frac{1}{2} \sin^2 \frac{2\pi n}{N} = \frac{-\omega^{2n} + 10 - \omega^{-2n}}{8}. \quad (42)$$

The spline  $S^3(t) = \sum_{k=0}^{N/2-1} q[k] B^3(t-k)$ , where the DFT of the coefficients is  $\hat{q}[n]_1 = \hat{x}_0[n]_1 U^3[n]_1$ , is called the quadratic spline quasi-interpolating the signal  $\mathbf{x}_0 \in \Pi[N/2]$ . Denote the samples of the spline  $s_0[k] \stackrel{\text{def}}{=} S^3(k)$  and  $s_1[k] \stackrel{\text{def}}{=} S^3(k+1/2)$ ,  $\kappa \in \mathbb{Z}$ . Then, the DFTs  $\hat{s}_0[n]_1 = g_q^3[n]_1 \hat{x}_0[n]_1$  and  $\hat{s}_1[n]_1 = f_q^3[n]_1 \hat{x}_0[n]_1$ , where

$$\begin{aligned} g_q^3[n]_1 &= u^3[n]_1 U^3[n]_1 = (-\omega^{-4n} + 4\omega^{-2n} + 58 + 4\omega^{2n} - \omega^{4n})/64, \\ f_q^3[n]_1 &= v^3[n]_1 U^3[n]_1 = (-\omega^{-2n} + 9 + 9\omega^{2n} - \omega^{4n})/16. \end{aligned} \quad (43)$$

We define two types of low- and high-pass p-filters.

**1. Non-interpolating low-pass p-filter:** Define the p-filters by their FR:

$$\begin{aligned} \hat{h}^0[n] &\stackrel{\text{def}}{=} \frac{1}{\sqrt{2}} (g_q^3[n]_1 + \omega^{-n} f_q^3[n]_1) = \frac{1}{\sqrt{2}} \cos^4 \frac{\pi n}{N} (3 - \cos \frac{2\pi n}{N}) \\ &= \hat{h}^0[0] - \sqrt{2} \sin^4 \frac{\pi n}{N} (5 + 4 \cos \frac{2\pi n}{N} + \cos^2 \frac{2\pi n}{N}) \\ \hat{h}^1[n] &= \omega^{-n} \hat{h}^0[n + N/2] = \frac{\omega^{-n}}{\sqrt{2}} \sin^4 \frac{\pi n}{N} (3 + \cos \frac{2\pi n}{N}). \end{aligned} \quad (44)$$

The IR of the p-filters  $\mathbf{h}^0$  and  $\mathbf{h}^1$  comprise nine terms (up to periodization). The low-pass  $\mathbf{h}^0$  and the high-pass  $\mathbf{h}^1$  p-filters locally restore and eliminate sampled cubic polynomials, respectively.

**2. Interpolating low-pass p-filter:** The same approximation order can be achieved by p-filters that have shorter IR. For this, the low-pass p-filter  $\mathbf{h}^0$  is defined to be interpolating:

$$\begin{aligned} \hat{h}^0[n] &\stackrel{\text{def}}{=} \frac{1}{\sqrt{2}} (1 + \omega^{-n} f_q^3[n]_1) = \sqrt{2} \cos^4 \frac{\pi n}{N} (1 + 2 \sin^2 \frac{\pi n}{N}) = \hat{h}^0[0] - \sqrt{2} \sin^4 \frac{\pi n}{N} (2 + \cos^2 \frac{\pi n}{N}), \\ \hat{h}^1[n] &= \omega^{-n} \hat{h}^0[n + N/2] = \omega^{-n} \sqrt{2} \sin^4 \frac{\pi n}{N} (1 + 2 \cos^2 \frac{\pi n}{N}). \end{aligned} \quad (45)$$

The IR of the p-filters  $\mathbf{h}^0$  and  $\mathbf{h}^1$  comprise five terms (up to periodization). The low-pass  $\mathbf{h}^0$  and the high-pass  $\mathbf{h}^1$  p-filters locally restore and eliminate sampled cubic polynomials, respectively.

Like the interpolating spline, the quadratic quasi-interpolating spline possesses the super-convergence property at midpoints between grid nodes. This is the reason why the p-filters and  $\mathbf{h}^0$  and  $\mathbf{h}^1$  locally restore and eliminate sampled cubic polynomials, respectively.

### 5.3.2 Pseudo-spline p-filters

A family of linear phase FIR low-pass filters is introduced in [8] for non-periodic setting. In the periodic case, the design starts from the obvious identity

$$1 = \left( \cos^2 \frac{\pi n}{N} + \sin^2 \frac{\pi n}{N} \right)^{m+l}, \quad m, l \in \mathbb{N}. \quad (46)$$

The FR of the low-pass p-filter  $\mathbf{h}^0$  is derived by summing of the first  $l + 1$  terms of the binomial expansion in Eq. (46):

$$\hat{h}^0[n] = \sqrt{2} \cos^{2m} \frac{\pi n}{N} \sum_{j=0}^l \binom{m+l}{j} \sin^{2j} \frac{\pi n}{N} \cos^{2(l-j)} \frac{\pi n}{N}. \quad (47)$$

**Example:**  $m = 2, l = 1$

$$\hat{h}^0[n] = \sqrt{2} \cos^4 \frac{\pi n}{N} \left( \cos^2 \frac{\pi n}{N} + 3 \sin^2 \frac{\pi n}{N} \right). \quad (48)$$

We observe that the filter  $\mathbf{h}^0$  defined by Eq. (48) is the same as the interpolating p-filter derived from the quadratic quasi-interpolating spline, whose FR is given by Eq. (45).

**Example from [8]:**  $m = 3, l = 1$

$$\hat{h}^0[n] = \sqrt{2} \cos^6 \frac{\pi n}{N} \left( 1 + 3 \sin^2 \frac{\pi n}{N} \right), \quad \hat{h}^1[n] = \omega^{-n} \sqrt{2} \sin^6 \frac{\pi n}{N} \left( 1 + 3 \cos^2 \frac{\pi n}{N} \right). \quad (49)$$

$$\hat{h}^0[n] = \hat{h}^0[0] - \sqrt{2} \sin^4 \frac{\pi n}{N} \left( 3 \cos^4 \frac{\pi n}{N} + 2 \cos^2 \frac{\pi n}{N} + 1 \right).$$

Proposition 4.1 implies that the low-pass p-filter  $\mathbf{h}^0$  locally restores sampled cubic polynomials. However, the high-pass p-filter  $\mathbf{h}^1$  locally eliminates fifth-degree sampled polynomials. The IRs of the filters  $\mathbf{h}^0$  and  $\mathbf{h}^1$  comprise nine terms (up to periodization).

**Remark 5.4** *We emphasize that, unlike most of the previous examples, the low-pass p-filters  $\mathbf{h}^0$  defined in Eqs. (44) and (49) are not interpolating.*

## 6 Four-channel p-filter banks using spline filters

In this section, we present a few examples of PR four-channel p-filter banks, which generate tight and semi-tight frames in the space  $\Pi[N]$  of  $N$ -periodic discrete-time signals. In almost all the forthcoming examples, the low-pass p-filters are interpolating. The design scheme was described in Section 3.

*Summarizing*, if a linear phase low-pass filter  $\mathbf{h}^0$ , whose FR is  $\hat{h}^0[n] = \hat{h}_0^0[n]_1 + \omega^{-n} \hat{h}_1^0[n]_1$ , is designed then the high-pass filter  $\mathbf{h}^1$  is defined by its FR as  $\hat{h}^1[n] = \omega^{-n} \left( \hat{h}_0^0[n]_1 - \omega^{-n} \hat{h}_1^0[n]_1 \right) = \omega^{-n} \hat{h}^0[n + N/2]$ .

Design of the band-pass p-filters is based on factorization of the sequence  $t[n]_1 \stackrel{\text{def}}{=} 1 - \left| \hat{h}_0^0[n]_1 \right|^2 + \left| \hat{h}_1^0[n]_1 \right|^2 = T[n]_1 \tilde{T}[-n]_1$ . Once the factorization is accomplished with  $T[n]_1 \neq \tilde{T}[-n]_1$ , the FRs of the band-pass p-filters  $\hat{\mathbf{h}}^s$  and  $\hat{\mathbf{h}}^s$ ,  $s = 2, 3$ , are defined as

$$\begin{aligned} \hat{h}^2[n] &= T[n]_1, & \hat{h}^3[n] &= -\omega^{-n} \tilde{T}[-n]_1, \\ \hat{\tilde{h}}^2[n] &= \tilde{T}[n]_1, & \hat{\tilde{h}}^3[n] &= -\omega^{-n} T[-n]_1. \end{aligned} \quad (50)$$

The PR pair  $\left\{ \tilde{\mathbf{H}}, \mathbf{H} \right\}$  of the p-filter banks, where  $\tilde{\mathbf{H}} = \left\{ \mathbf{h}^0, \mathbf{h}^1, \tilde{\mathbf{h}}^2, \tilde{\mathbf{h}}^3 \right\}$  and  $\mathbf{H} = \left\{ \mathbf{h}^0, \mathbf{h}^1, \mathbf{h}^2, \mathbf{h}^3 \right\}$  generates a semi-tight frame in  $\Pi[N]$ .

When  $T[n]_1 = \tilde{T}[n]_1$ , we use the “Symmetric” design of the p-filter banks  $\mathbf{H}$  that generates tight frames:

$$\hat{h}^2[n] = A[-n]_1 + \omega^{-n} A[n]_1, \quad \hat{h}^3[n] = -A[-n]_1 + \omega^{-n} A[n]_1, \quad (51)$$

where  $A[n]_1 \stackrel{\text{def}}{=} T[n]_1/\sqrt{2}$ . Recall that impulse responses of the p-filters  $\tilde{\mathbf{h}}^s$  and  $\mathbf{h}^s$ ,  $s = 0, 1, 2, 3$ , are the framelets  $\psi_{[1]}^s$  and  $\psi_{[1]}^s$  of the first level, respectively.

If the low-pass p-filter is interpolating and the prediction p-filter  $\mathbf{f}$  is available then the design of the p-filter bank is reduced to factorization of the sequence  $W[n]_1 \stackrel{\text{def}}{=} 1 - |\hat{f}[n]_1|^2 = T[n]_1 \tilde{T}[-n]_1$  (Section 3.1).

The impulse and the magnitude responses of PR p-filter banks  $\mathbf{H} = \{\mathbf{h}^s\}$ ,  $s = 0, 1, 2, 3$ , which generate tight frames, are displayed in Fig. 8.

The impulse and the magnitude responses of p-filters  $\tilde{\mathbf{h}}^s$  and  $\mathbf{h}^s$ ,  $s = 2, 3$ , that together with the p-filters  $\mathbf{h}^s$ ,  $s = 0, 1$ , generate semi-tight frames, are displayed in Figs. 1–7. All the Figs. 1–7, except Fig. 5, are structured identically. Left pictures display the IR of the p-filters in the following order: Left to right  $\mathbf{h}^2 \rightarrow \tilde{\mathbf{h}}^2 \rightarrow \mathbf{h}^3 \rightarrow \tilde{\mathbf{h}}^3$ . Right pictures display the MR of  $\mathbf{h}^2$  and  $\tilde{\mathbf{h}}^3$  (which coincide with each other) (dashed lines), and the MR of  $\mathbf{h}^2$  and  $\mathbf{h}^3$  (which coincide with each other) (solid line). The top pictures pair illustrates the p-filters arising from symmetric factorization, while the bottom pair does the same when the anti symmetric factorization is used. Figure 5 comprises only one pair of pictures, which correspond to the symmetric factorization.

## 6.1 Four-channel p-filter banks with p-FIR p-filters

A few examples of p-filter banks and p-filter banks with FIR p-filters are given in this section.

**Linear interpolating spline:** The prediction p-filter is  $\hat{f}_c^2[n]_1 = (1 + \omega^{2n})/2$ . Thus,  $W[n]_1 = (-\omega^{2n} + 2 - \omega^{-2n})/4 \implies A[n]_1 = (1 - \omega^{2n})/4$ . The frequency responses of the p-filters are

$$\begin{aligned} \hat{h}^0[n] &= \sqrt{2} \cos^2 \frac{\pi n}{N}, & \hat{h}^1[n] &= \omega^{-n} \sqrt{2} \sin^2 \frac{\pi n}{N}, \\ \hat{h}^2[n] &= \frac{1 - \omega^{-2n}}{4} + \omega^{-n} \frac{1 - \omega^{2n}}{4} = -\frac{(1 + \omega^n)(1 - \omega^n)^2}{4\omega^{2n}}, \\ \hat{h}^3[n] &= -\frac{1 - \omega^{-2n}}{4} + \omega^{-n} \frac{1 - \omega^{2n}}{4} = \frac{(1 - \omega^n)(1 + \omega^n)^2}{4\omega^{2n}}. \end{aligned}$$

The p-filter  $\mathbf{h}^0$  locally restores first-degree sampled polynomials, while the p-filters  $\mathbf{h}^1$  and  $\mathbf{h}^2$  locally eliminate them. The p-filter  $\mathbf{h}^3$  locally eliminates only constants. The framelets  $\psi_{[1]}^1$  and  $\psi_{[1]}^2$ , which are the IRs of the p-filters  $\mathbf{h}^1$  and  $\mathbf{h}^2$ , respectively, have two LDVMs. Either of them is symmetric. The framelet  $\psi_{[1]}^3$  is antisymmetric and has one LDVM.

The IR of the p-filters  $\mathbf{h}^s$ ,  $s = 0, 1, 2, 3$ , and their magnitude responses are displayed in Fig. 8 (top line).

**Quadratic quasi-interpolating spline (interpolating low-pass p-filter):** The FR of the prediction p-filter  $\mathbf{f}$  low- and high-pass p-filters  $\mathbf{h}^0$  and  $\mathbf{h}^1$  are given in Eqs. (43) and (45), respectively. The p-filter  $\mathbf{h}^0$  locally restores sampled polynomials of third degree while the p-filter  $\mathbf{h}^1$  locally eliminates them. Thus, the framelet  $\psi_{[1]}^1$  has four LDVMs. The sequence  $W[n]_1$  to be factorized is

$$W[n]_1 = \left( \frac{\omega^{2n} - 2 + \omega^{-2n}}{16} \right)^2 \Gamma_c^4[n]_1, \quad \Gamma_c^4[n]_1 \stackrel{\text{def}}{=} -\omega^{2n} + 14 - \omega^{-2n}. \quad (52)$$

The following factorization modes of  $W[n]_1 = w[n]_1 \tilde{w}[-n]_1$  are possible:

1. *Non-symmetric factorization*  $W[n]_1 = w[n]_1 \tilde{w}[-n]_1$ , where  $w[n]_1 = ((\omega^{-2n} - 2 + \omega^{2n})(1 - q\omega^{2n}))/16\sqrt{q}$ ,  $q = 7 - 4\sqrt{3}$ . FRs of the p-filters  $\mathbf{h}^2$  and  $\mathbf{h}^3$  are defined in Eq. (51) with  $A[n]_1 = w[n]_1/2$ .

The p-filter bank  $\mathbf{H} = \{\mathbf{h}^s\}$ ,  $s = 0, 1, 2, 3$ , generates a tight frame. The p-filters  $\mathbf{h}^3$  and  $\mathbf{h}^2$  locally eliminate sampled polynomials of first and second degrees, respectively. The framelets  $\psi_{[1]}^3$  and  $\psi_{[1]}^2$ , which are impulse responses of the p-filters  $\mathbf{h}^3$  and  $\mathbf{h}^2$ , have three and two LDVMs, respectively. IRs of the p-filters  $\mathbf{h}^s$ ,  $s = 0, 1, 2, 3$ , and their MRs are displayed in Fig. 4 (second from top). MRs of the p-filters  $\mathbf{h}^2$  and  $\mathbf{h}^3$  mirror each other.

2. *Symmetric factorization* of  $W[n]_1 = w[n]_1 \tilde{w}[-n]_1$ , which provides equal number (two) of LDVMs to the analysis and to the synthesis framelets, are:

$$\tilde{w}[n]_1 = \frac{(\omega^{-2n} - 2 + \omega^{2n})}{4}, \quad w[n]_1 = \frac{(\omega^{-2n} - 2 + \omega^{2n}) \Gamma_c^4[n]_1}{64}.$$

FRs of the p-filters  $\tilde{\mathbf{h}}^s$  and  $\mathbf{h}^s$ ,  $s = 2, 3$ , are given in Eq. (50), where  $\tilde{T}[n]_1 = \tilde{w}[n]_1/\sqrt{2}$  and  $T[n]_1 = w[n]_1/\sqrt{2}$ .

3. *Antisymmetric factorization* of  $W[n]_1 = w[n]_1 \tilde{w}[-n]_1$ , which assigns three LDVMs to the analysis framelet  $\tilde{\psi}_{[1]}^2$  leaving only one LDVM to the synthesis framelet  $\psi_{[1]}^2$  and vice versa for the framelets  $\tilde{\psi}_{[1]}^3$  and  $\psi_{[1]}^3$ , are:

$$\tilde{w}[n]_1 = \frac{\omega^{4n} - 3\omega^{2n} + 3 - \omega^{-2n}}{8}, \quad w[n]_1 = \frac{(1 - \omega^{2n}) \Gamma_c^4[n]_1}{32}.$$

FRs of the p-filters  $\tilde{\mathbf{h}}^s$  and  $\mathbf{h}^s$ ,  $s = 2, 3$ , are given in Eq. (50). The pairs of the p-filter banks  $\mathbf{H} = \{\mathbf{h}^s\}$ ,  $s = 0, 1, 2, 3$ , and  $\tilde{\mathbf{H}} = \{\tilde{\mathbf{h}}^0, \tilde{\mathbf{h}}^1, \tilde{\mathbf{h}}^2, \tilde{\mathbf{h}}^3\}$  generate semi-tight frames.

Figure 1 displays the impulse responses of the p-filters  $\tilde{\mathbf{h}}^s$  and  $\mathbf{h}^s$ ,  $s = 2, 3$ , which are the discrete time framelets of the first level and their MR.

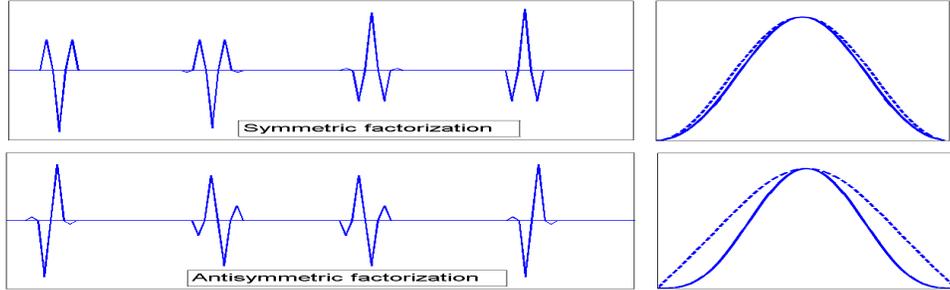


Figure 1: Left: IRs of the p-filters  $\mathbf{h}^2 \rightarrow \tilde{\mathbf{h}}^2 \rightarrow \mathbf{h}^3 \rightarrow \tilde{\mathbf{h}}^3$  derived from quadratic quasi-interpolating splines (interpolating low-pass p-filter). Right: MRs of  $\mathbf{h}^2$  and  $\tilde{\mathbf{h}}^3$  (coincide with each other - dashed lines), and MRs of  $\tilde{\mathbf{h}}^2$  and  $\mathbf{h}^3$  (coincide with each other - solid line)

**Quadratic quasi-interpolating spline (non-interpolating low-pass p-filter):** Frequency responses of the p-filter  $\mathbf{h}^0$  and  $\mathbf{h}^1$  are given in Eq. (44). The p-filter  $\mathbf{h}^0$  locally restores sampled cubic polynomials while the p-filter  $\mathbf{h}^1$  locally eliminates them. Thus, the framelet  $\psi_{[1]}^1$  has four LDVMs.

$$t[n]_1 \stackrel{\text{def}}{=} 1 - |h_0^0[n]_1|^2 - |h_1^0[n]_1|^2 = \frac{2(\omega^{2n} - 2 + \omega^{-2n})^2 \Upsilon[n]_1}{128^2},$$

$$\Upsilon[n]_1 \stackrel{\text{def}}{=} -\omega^{4n} - 12\omega^{2n} + 346 - 12\omega^{-2n} - \omega^{-4n}.$$

The following factorizations modes are possible:

1. *Non-symmetric factorization (tight frame)*  $t[n]_1 = T[n]_1 T[-n]_1$ :

$$T[n]_1 = \frac{\sqrt{2}(\omega^{2n} - 2 + \omega^{-2n})(1 - \alpha_1 \omega^{2n})(1 + \alpha_2 \omega^{2n})}{128\sqrt{\alpha_1 \alpha_2}},$$

$$\alpha_1 = 0.07395375302024, \quad \alpha_2 = 0.03912854562755.$$

FRs of the p-filters  $\mathbf{h}^2$  and  $\mathbf{h}^3$  are defined in Eq. (51) with  $A[n]_1 = T[n]_1/\sqrt{2}$ . The p-filter bank  $\mathbf{H} = \{\mathbf{h}^s\}$ ,  $s = 0, 1, 2, 3$ , generates a tight frame. The framelet  $\psi_{[1]}^3$  has three LDVMs, while  $\psi_{[1]}^2$  has two. IRs of the p-filters  $\mathbf{h}^s$ ,  $s = 0, 1, 2, 3$ , and their MRs are displayed in Fig. 4 (second from bottom).

2. *Symmetric factorization (semi-tight frame)*  $t[n]_1 = T[n]_1 \tilde{T}[-n]_1$ , where framelets  $\tilde{\psi}_{[1]}^s$  and  $\psi_{[1]}^s$ ,  $s = 2, 3$ , have two LDVMs:

$$T[n]_1 = \frac{\sqrt{2}(-\omega^{2n} + 2 - \omega^{-2n}) \Upsilon[n]_1}{2048}, \quad \tilde{T}[n]_1 = \frac{\sqrt{2}(-\omega^{2n} + 2 - \omega^{-2n})}{4}.$$

3. *Antisymmetric factorization (semi-tight frame)* of  $t[n]_1 = T[n]_1 \tilde{T}[-n]_1$ , which assigns three LDVMs to the analysis framelet  $\tilde{\psi}_{[1]}^2$  leaving only one LDVM to the synthesis framelet  $\psi_{[1]}^2$  and vice versa for the framelets  $\tilde{\psi}_{[1]}^3$  and  $\psi_{[1]}^3$ :

$$\tilde{T}[n]_1 = \sqrt{2} \frac{\omega^{4n} - 3\omega^{2n} + 3 - \omega^{-2n}}{8}, \quad T[n]_1 = \sqrt{2} \frac{(1 - \omega^{2n}) \Upsilon[n]_1}{1024}.$$

FRs of the p-filters  $\tilde{\mathbf{h}}^s$  and  $\mathbf{h}^s$ ,  $s = 2, 3$ , are defined in Eq. (50). Figure 2 displays the IRs of the p-filters  $\tilde{\mathbf{h}}^s$  and  $\mathbf{h}^s$ ,  $s = 2, 3$ , which are the discrete-time framelets of the first level with their MRs.

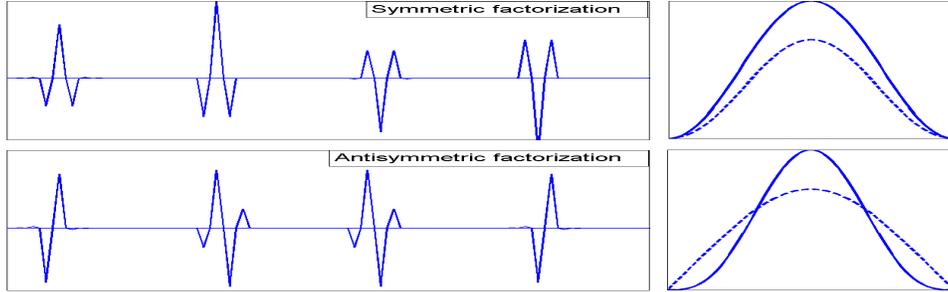


Figure 2: Left: IRs of the p-filters  $\mathbf{h}^2 \rightarrow \tilde{\mathbf{h}}^2 \rightarrow \mathbf{h}^3 \rightarrow \tilde{\mathbf{h}}^3$  derived from quadratic quasi-interpolating splines (non-interpolating low-pass p-filter). Right: MRs of  $\mathbf{h}^2$  and  $\tilde{\mathbf{h}}^3$  (coincide with each other - dashed lines), and MRs of  $\tilde{\mathbf{h}}^2$  and  $\mathbf{h}^3$  (coincide with each other - solid line)

**Pseudo-spline:** FRs of the non-interpolating low-pass p-filter  $\mathbf{h}^0$  and the corresponding high-pass p-filter  $\mathbf{h}^1$  are given by Eq. (49). The p-filter  $\mathbf{h}^0$  locally restores sampled cubic polynomials, while the p-filter  $\mathbf{h}^1$  locally eliminates sampled fifth-degree polynomials. Thus, the framelet  $\psi_{[1]}^1$  has six LDVMs. We have

$$t[n]_1 = \frac{2(\omega^{2n} - 2 + \omega^{-2n})^2 Q[n]_1}{256^2}, \quad Q[n]_1 \stackrel{\text{def}}{=} -9\omega^{4n} - 28\omega^{2n} + 1610 - 28\omega^{-2n} - 9\omega^{-4n}.$$

The following factorizations modes are possible:

1. *Non-symmetric factorization (tight frame)*  $t[n]_1 = T[n]_1 T[-n]_1$ :

$$T[n]_1 = \frac{\sqrt{2} 3(\omega^{2n} - 2 + \omega^{-2n})(1 - \alpha_1 \omega^{2n})(1 + \alpha_2 \omega^{2n})}{256 \sqrt{\alpha_1 \alpha_2}}$$

$$\alpha_1 = 0.08403672131163, \quad \alpha_2 = 0.06654171895289.$$

FRs of the p-filters  $\mathbf{h}^2$  and  $\mathbf{h}^3$  are defined in Eq. (51) with  $A[n]_1 = T[n]_1/\sqrt{2}$ . The p-filters  $\mathbf{h}^2$  and  $\mathbf{h}^3$  locally eliminate sampled polynomials of first and second degrees, respectively, thus, the framelets  $\psi_{[1]}^2$  and  $\psi_{[1]}^3$  have two and three LDVMs, respectively.

IRs of the p-filters  $\mathbf{h}^s$ ,  $s = 0, 1, 2, 3$ , and their MRs are displayed in Fig. 4 (bottom).

2. *Symmetric factorization (semi-tight frame)*  $t[n]_1 = T[n]_1 \tilde{T}[-n]_1$ :

$$T[n]_1 = \frac{\sqrt{2}(-\omega^{2n} + 2 - \omega^{-2n}) Q[n]_1}{8192}, \quad \tilde{T}[n]_1 = \frac{\sqrt{2}(-\omega^{2n} + 2 - \omega^{-2n})}{4}.$$

The p-filters  $\mathbf{h}^s$  and  $\tilde{\mathbf{h}}^s$ ,  $s = 2, 3$ , locally eliminate sampled polynomials of the first degree, thus the framelets  $\psi_{[1]}^2$  and  $\psi_{[1]}^3$  have two LDVMs.

3. *Antisymmetric factorization (semi-tight frame)*  $t[n]_1 = T[n]_1 \tilde{T}[-n]_1$ , which assigns three LDVMs to the analysis framelet  $\tilde{\psi}_{[1]}^2$  leaving only one LDVM to the synthesis framelet  $\psi_{[1]}^2$  and vice versa for the framelets  $\tilde{\psi}_{[1]}^3$  and  $\psi_{[1]}^3$ :

$$\tilde{T}[n]_1 = \sqrt{2} \frac{\omega^{4n} - 3\omega^{2n} + 3 - \omega^{-2n}}{8}, \quad T[n]_1 = \sqrt{2} \frac{(1 - \omega^{2n}) Q[n]_1}{4096}.$$

FRs of the p-filters  $\tilde{\mathbf{h}}^s$  and  $\mathbf{h}^s$ ,  $s = 2, 3$ , are defined in Eq. (50).

Figure 3 displays the IRs of the p-filters  $\mathbf{h}^s$ , which are discrete-time framelets of the first level with their MR.

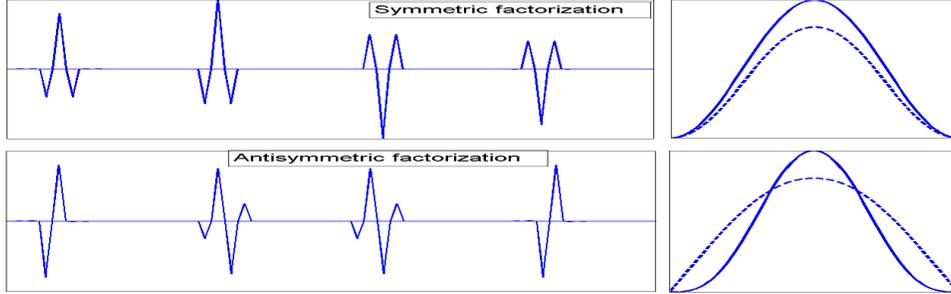


Figure 3: Left: IRs of the p-filters  $\mathbf{h}^2 \rightarrow \tilde{\mathbf{h}}^2 \rightarrow \mathbf{h}^3 \rightarrow \tilde{\mathbf{h}}^3$  derived from the pseudo-spline. Right: MRs of  $\mathbf{h}^2$  and  $\tilde{\mathbf{h}}^3$  (coincide with each other - dashed lines) and MRs of  $\tilde{\mathbf{h}}^2$  and  $\mathbf{h}^3$  (coincide with each other - solid line)

**Display of tight frame FIR p-filters:** Figure 4 displays the impulse and the magnitude responses of the p-FIR four-channel p-filter banks that generate tight frames. The impulse responses of the p-filters  $\mathbf{h}^s$  are the corresponding discrete-time framelets  $\psi_{[1]}^s[k]$ ,  $s = 0, 1, 2, 3$ , of the first decomposition level. The framelets  $\psi_{[1]}^s[k]$ ,  $s = 0, 1, 2$ , are symmetric, while the framelets  $\psi_{[1]}^3[k]$  are antisymmetric. The magnitude responses of the p-filters  $\mathbf{h}^0$  and  $\mathbf{h}^1$  mirror each other and the same is true for the band-pass pair  $\mathbf{h}^2$  and  $\mathbf{h}^3$ . In the caption of the figure, abbreviation QIS means quasi-interpolating spline.

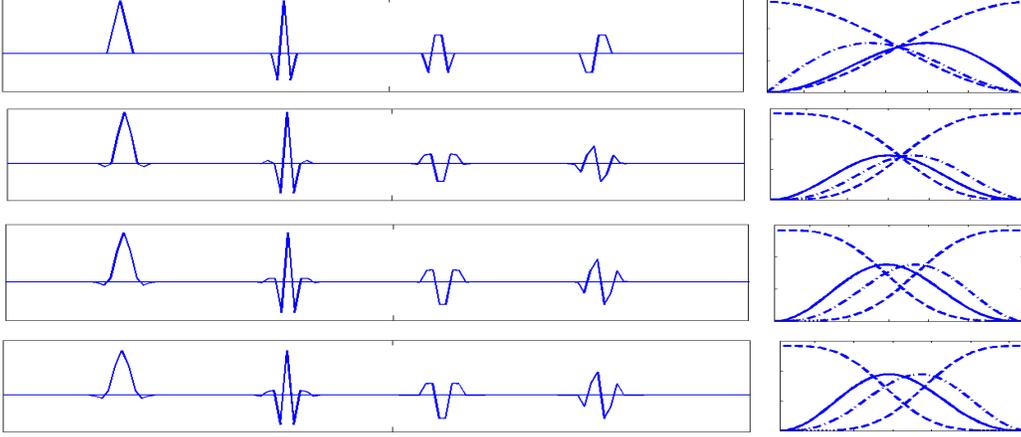


Figure 4: Impulse and magnitude responses of the p-FIR four-channel p-filter banks that generate tight frames. Left pictures display the impulse responses of p-filters where, from *left to right* we have:  $\mathbf{h}^0 \rightarrow \mathbf{h}^1 \rightarrow \mathbf{h}^2 \rightarrow \mathbf{h}^3$ . *Right*: pictures display the magnitude responses of these p-filters:  $\mathbf{h}^0$  and  $\mathbf{h}^1$  (dashed lines),  $\mathbf{h}^3$  (dash-dot line) and  $\mathbf{h}^2$  (solid line). *Top*: linear spline. *Second from top*: quadratic QIS (interpolating low-pass p-filter). *Second from bottom*: quadratic QIS (non-interpolating low-pass p-filter). *Bottom*: pseudo-spline

## 6.2 Four-channel p-filter banks with p-IIR p-filters

Unlike non-periodic setting, the implementation cost of IIR p-filters is no higher than the implementation cost of p-FIR p-filters. However, giving up the requirement of having finite impulse response provides additional flexibility in the design of p-filter banks with needed properties.

For the design of IIR filter, we use the interpolating low-pass p-filters  $\mathbf{h}^0$ . The frequency responses of the p-filters are structured as  $\hat{h}^0[n] = (1 + \omega^{-n} \hat{f}[n]_1) / \sqrt{2}$ , where  $\hat{f}[n]_1$  is the frequency response of some prediction p-filter. Prediction p-filters were derived from interpolating polynomial and discrete splines, were described in Sections 5.1 and 5.2, respectively.

### 6.2.1 Examples of p-filter banks originating from polynomial interpolating splines

**Quadratic interpolating spline  $p = 3$ :** Denote  $\Omega_c^3[n]_1 \stackrel{\text{def}}{=} \omega^{2n} + 6 + \omega^{-2n}$ . The prediction p-filter  $\mathbf{f}_c^3$  and the p-filters  $\mathbf{h}^0$  and  $\mathbf{h}^1$  are defined in Eq. (35). The sequence  $W[n]_1$  is

$$W[n]_1 = \left( \frac{\omega^{2n} - 2 + \omega^{-2n}}{\Omega_c^3[n]_1} \right)^2 = \left( \frac{\sin^2 2\pi n/N}{2\Omega_c^3[n]_1} \right)^2.$$

The p-filter  $\mathbf{h}^0$  locally restores sampled cubic polynomials while the p-filter  $\mathbf{h}^1$  locally eliminates them. Thus, the framelet  $\psi_{[1]}^1$  has four LDVMs.

Different factorizations modes of  $W[n]_1$  are possible:

1. *Symmetric factorization (tight frame)*  $W[n]_1 = w[n]_1 w[-n]_1$ , where

$$w[n]_1 = \frac{(\omega^{-2n} - 2 + \omega^{2n})}{\Omega_c^3[n]_1} = \frac{\sin^2 2\pi n/N}{2\Omega_c^3[n]_1}.$$

FRs of the p-filters  $\mathbf{h}^2$  and  $\mathbf{h}^3$  are defined in Eq. (51) where  $A[n]_1 = w[n]_1/2$ . The p-filter  $\mathbf{h}^3$  locally eliminates sampled quadratic polynomials, thus the framelet  $\psi_{[1]}^3$ , has three LDVMs, while the framelet

$\psi_{[1]}^2$ , has two LDVMs. The IRs of the p-filters  $\mathbf{h}^s$ ,  $s = 0, 1, 2, 3$ , and their MRs are displayed in Fig. 8 (top).

2. *Antisymmetric factorization (semi-tight frame)* of  $W[n]_1 = w[n]_1 \tilde{w}[-n]_1$ , which assigns three LDVMs to the analysis framelet  $\tilde{\psi}_{[1]}^2$  leaving only one LDVM to the synthesis framelet  $\psi_{[1]}^2$  and vice versa for the framelets  $\tilde{\psi}_{[1]}^3$  and  $\psi_{[1]}^3$ :

$$\tilde{w}[n]_1 = \frac{\omega^{4n} - 3\omega^{2n} + 3 - \omega^{-2n}}{2\Omega_c^3[n]_1}, \quad w[n]_1 = \frac{2(1 - \omega^{2n})}{\Omega_c^3[n]_1}.$$

FRs of the p-filters  $\tilde{\mathbf{h}}^s$  and  $\mathbf{h}^s$ ,  $s = 2, 3$ , are defined in Eq. (50). Figure 5 displays IRs of the p-filters  $\tilde{\mathbf{h}}^s$  and  $\mathbf{h}^s$ ,  $s = 2, 3$ , which are discrete-time framelets of the first level and their MRs.

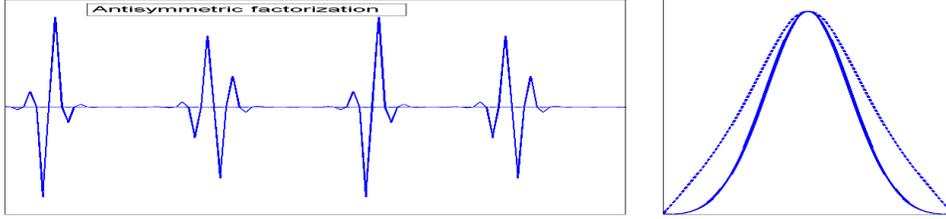


Figure 5: Left: IRs of the p-filters  $\mathbf{h}^2 \rightarrow \tilde{\mathbf{h}}^2 \rightarrow \tilde{\mathbf{h}}^3 \rightarrow \mathbf{h}^3$  derived from quadratic interpolating splines. Right: MRs of  $\mathbf{h}^2$  and  $\tilde{\mathbf{h}}^3$  (coincide with each other - dashed lines), and MRs of  $\tilde{\mathbf{h}}^2$  and  $\mathbf{h}^3$  (coincide with each other - solid line)

**Cubic interpolating spline,  $p = 4$ :**  $\Omega_c^4[n]_1 \stackrel{\text{def}}{=} \omega^{2n} + 4 + \omega^{-2n}$ ,  $\Gamma_c^4[n]_1 \stackrel{\text{def}}{=} -\omega^{2n} + 14 - \omega^{-2n}$ . The prediction p-filter  $\mathbf{f}_c^4$  and the p-filters  $\mathbf{h}^0$  and  $\mathbf{h}^1$  are defined in Eq. (36). The p-filter  $\mathbf{h}^0$  locally restores the sampled cubic polynomials while the p-filter  $\mathbf{h}^1$  locally eliminates them. Thus, the framelet  $\psi_{[1]}^1$  has four LDVM. We have

$$W[n]_1 = \left( \frac{\omega^{2n} - 2 + \omega^{-2n}}{8\Omega_c^4[n]_1} \right)^2 \Gamma_c^4[n]_1. \quad (53)$$

Comparing Eq. (53) with Eq. (52), we observe that the numerator of the sequences  $W[n]_1$  in both equations is the same. Therefore, factorization of  $W[n]_1$  of the cubic interpolating spline is similar to the factorization of the quadratic quasi-interpolating spline. Here are some factorization examples:

1. *Non-symmetric factorization (tight frame)*  $W[n]_1 = w[n]_1 w[-n]_1$ , where

$$w[n]_1 = \frac{\omega^{2n} - 2 + \omega^{-2n}}{8\sqrt{q}\Omega_c^4[n]_1} (1 - q\omega^{2n}), \quad q = 7 - 4\sqrt{3}.$$

FRs of the p-filters  $\mathbf{h}^2$  and  $\mathbf{h}^3$  are defined in Eq. (51) where  $A[n]_1 = w[n]_1/2$ . The framelet  $\psi_{[1]}^3$  has three LDVMs, while the framelet  $\psi_{[1]}^2$  has two LDVMs. IRs of the p-filters  $\mathbf{h}^s$ ,  $s = 0, 1, 2, 3$ , and their MRs are displayed in Fig. 8 (second from top).

2. *Symmetric factorization (semi-tight frame)*  $W[n]_1 = w[n]_1 \tilde{w}[-n]_1$ , where

$$\tilde{w}[n]_1 = \frac{\omega^{2n} - 2 + \omega^{-2n}}{8\Omega_c^4[n]_1}, \quad w[n]_1 = \tilde{w}[n]_1 \Gamma_c^4[n]_1.$$

The framelets  $\psi_{[1]}^2$  and  $\psi_{[1]}^s$  have two LDVMs each.

3. *Antisymmetric factorization (semi-tight frame)*  $W[n]_1 = w[n]_1 \tilde{w}[-n]_1$ , which assigns three LDVMs to the analysis  $\tilde{\psi}^2_{[1]}$  and one LDVM to the synthesis framelet  $\psi^2_{[1]}$  and vice versa for  $\tilde{\psi}^3_{[1]}$  and  $\psi^3_{[1]}$ :

$$\tilde{w}[n]_1 = \frac{\omega^{4n} - 3\omega^{2n} + 3 - \omega^{-2n}}{4\Omega_c^4[n]_1}, \quad w[n]_1 = \frac{(1 - \omega^{2n}) \Gamma_c^4[n]_1}{16\Omega_c^4[n]_1}.$$

Figure 6 displays IRs of the p-filters  $\tilde{\mathbf{h}}^s$  and  $\mathbf{h}^s$ ,  $s = 2, 3$ , which are discrete-time framelets of the first level, and their MRs.

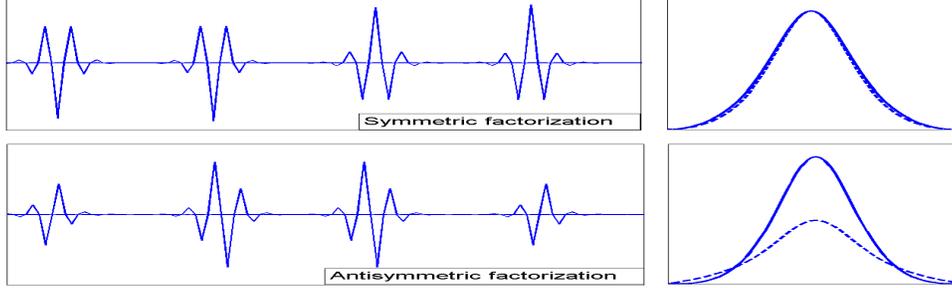


Figure 6: Left: IRs of the p-filters  $\mathbf{h}^2 \rightarrow \tilde{\mathbf{h}}^2 \rightarrow \mathbf{h}^3 \rightarrow \tilde{\mathbf{h}}^3$  derived from cubic interpolating splines. Right: MRs of  $\mathbf{h}^2$  and  $\tilde{\mathbf{h}}^3$  (coincide with each other) (dashed lines), and MRs of  $\tilde{\mathbf{h}}^2$  and  $\mathbf{h}^3$  (coincide with each other) (solid line)

**Interpolating spline of fourth degree,  $p = 5$ :** Denote  $\Omega_c^5[n]_1 \stackrel{\text{def}}{=} \omega^{4n} + 76\omega^{2n} + 230 + 76\omega^{-2n} + \omega^{-4n}$ ,  $\Gamma_c^5[n]_1 \stackrel{\text{def}}{=} -\omega^{2n} + 98 - \omega^{-2n}$ . Then,

$$\begin{aligned} \hat{f}_c^5[n]_1 &= 16 \frac{\omega^{4n} + 11\omega^{2n} + 11 + \omega^{-2n}}{\Omega_c^5[n]_1}, & W[n]_1 &= (-\omega^{2n} + 2 - \omega^{-2n})^3 \frac{\Gamma_c^5[n]_1}{(\Omega_c^5[n]_1)^2} \\ \hat{h}^0[n] &= \sqrt{2} 128 \frac{\cos^6 \pi n/N (2 + \cos^2 \pi n/N)}{\Omega_c^5[n]_1}, & \hat{h}^1[n] &= \sqrt{2} 128 \frac{\sin^6 \pi n/N (2 + \sin^2 \pi n/N)}{\Omega_c^5[n]_1}. \end{aligned} \quad (54)$$

The p-filter  $\mathbf{h}^0$  locally restores the sampled polynomials of the fifth degree while the p-filter  $\mathbf{h}^1$  locally eliminates them. Thus, the framelet  $\psi^1_{[1]}$  has six LDVM. Here are some examples for  $W[n]_1$  factorization.

1. *Non-symmetric factorization (tight frame)*  $W[n]_1 = w[n]_1 w[-n]_1$ , where

$$w[n]_1 = \frac{\omega^{2n} - 3 + 3\omega^{-2n} - \omega^{-4n}}{\sqrt{q} \Omega_c^5[n]_1} (1 - q\omega^{2n}),$$

$q = 49 - 20\sqrt{6}$ . FRs of the p-filters  $\mathbf{h}^2$  and  $\mathbf{h}^3$  are defined in Eq. (51) where  $A[n]_1 = w[n]_1/2$ . The framelets  $\psi^2_{[1]}$  has four LDVMs and  $\psi^3_{[1]}$  has three LDVMs. IRs of the p-filters  $\mathbf{h}^s$ ,  $s = 0, 1, 2, 3$ , and their MRs are displayed in Fig. 8 (third from top).

2. *Symmetric factorization (semi-tight frame)*  $W[n]_1 = w[n]_1 \tilde{w}[-n]_1$ , which assigns four LDVMs to the analysis  $\tilde{\psi}^2_{[1]}$  and two LDVMs to the synthesis framelet  $\psi^2_{[1]}$  and vice versa for  $\tilde{\psi}^3_{[1]}$  and  $\psi^3_{[1]}$ :

$$w[n]_1 = -\frac{\omega^{2n} - 2 + \omega^{-2n}}{8\sqrt{2} \Omega_c^5[n]_1} \Gamma_c^5[n]_1, \quad \tilde{w}[n]_1 = 8\sqrt{2} \frac{(\omega^{2n} - 2 + \omega^{-2n})^2}{\Omega_c^5[n]_1}.$$

3. *Antisymmetric factorization (semi-tight frame)*  $W[n]_1 = w[n]_1 \tilde{w}[-n]_1$ , where all the framelets  $\tilde{\psi}^s_{[1]}$  and  $\psi^s_{[1]}$ ,  $s = 2, 3$ , have three LDVMs:

$$\tilde{w}[n]_1 = \frac{\omega^{2n} - 3 + 3\omega^{-2n} - \omega^{-4n}}{8\sqrt{2} \Omega_c^5[n]_1}, \quad w[n]_1 = \tilde{w}[n]_1 \frac{\Gamma_c^5[n]_1}{128}. \quad (55)$$

In both the symmetric and the antisymmetric cases, FRs of the p-filters  $\tilde{\mathbf{h}}^s$  and  $\mathbf{h}^s$ ,  $s = 2, 3$ , are defined in Eq. (50). Figure 7 displays IRs of the p-filters  $\tilde{\mathbf{h}}^s$  and  $\mathbf{h}^s$ ,  $s = 2, 3$ , which are the discrete time framelets of the first level and their MR.



Figure 7: Left: IRs of the p-filters  $\mathbf{h}^2 \rightarrow \tilde{\mathbf{h}}^2 \rightarrow \mathbf{h}^3 \rightarrow \tilde{\mathbf{h}}^3$  derived from interpolating splines of the fifth order (fourth degree). Right: MRs of  $\mathbf{h}^2$  and  $\tilde{\mathbf{h}}^3$  (coincide with each other - dashed lines), and MRs of  $\tilde{\mathbf{h}}^2$  and  $\mathbf{h}^3$  (coincide with each other - solid line)

**Four-channel p-filter banks derived from discrete splines:** Denote  $\Omega_d^{2r}[n]_1 \stackrel{\text{def}}{=} \cos^{2r} \pi n/N + \sin^{2r} \pi n/N$ . The prediction p-filters  $\mathbf{f} = \mathbf{f}_d^{2r}$  are given in Eq. (39) and the FRs of the low- and high-pass p-filters are defined in Eq. (40). The sequences  $W[n]_1$  are

$$W[n]_1 = \frac{4 \sin^{2r} 2\pi n/N}{2^{2r} (\Omega_d^{2r}[n]_1)^2} = \frac{4(-1)^r \omega^{-2nr} (1 - \omega^{2n})^{2r}}{4^{2r} (\Omega_d^{2r}[n]_1)^2}.$$

The sequence  $W[n]_1$  is factorized to be  $W[n]_1 = w[n]_1 w[-n]_1$ , where

$$\begin{aligned} w[n]_1 &= \begin{cases} 2\omega^{-2np} (1 - \omega^{2n})^r / (4^r \Omega_d^{2r}[n]_1) & \text{if } r \text{ is odd, } p \in \mathbb{Z}; \\ 2 \sin^r 2\pi n/N / (2^r \Omega_d^{2r}[n]_1), & \text{if } r \text{ is even,} \end{cases} \\ h^2[n] &= A[-n]_1 + \omega^{-n} A[n]_1, \quad h^3[n] = -A[-n]_1 + \omega^{-n} A[n]_1, \end{aligned}$$

where  $A[n]_1 = w[n]_1/2$ . The interpolating low-pass p-filter  $\mathbf{h}^0$  locally restores sampled polynomials of degree  $2r - 1$ , while the high-pass p-filter  $\mathbf{h}^1$  eliminates them (the framelet  $\psi_{[1]}^1$  has  $2r$  LDVMs). The p-filter bank  $\mathbf{H} = \{\mathbf{h}^s\}$ ,  $s = 0, 1, 2, 3$ , generates a tight frame. If  $r$  is even then the framelet  $\psi_{[1]}^2$  has  $r + 1$  LDVMs, while the framelet  $\psi_{[1]}^3$  has  $r$  LDVMs, and vice versa for  $r$  odd.

**Display of tight frame IIR p-filters:** Figure 8 displays the impulse and the magnitude responses of IIR four-channel p-filter banks that generate tight frames. The p-filters are derived from the polynomial interpolating splines of orders 3,4,5 and from the discrete splines of orders 6, 8, 10 and 12. Impulse responses of the p-filters  $\mathbf{h}^s$  are the corresponding discrete-time framelets  $\psi_{[1]}^s[k]$ ,  $s = 0, 1, 2, 3$ , of the first decomposition level. The framelets  $\psi_{[1]}^s[k]$ ,  $s = 0, 1, 2$ , are symmetric, while the framelet  $\psi_{[1]}^3[k]$  is antisymmetric. MRs of the p-filters  $\mathbf{h}^0$  and  $\mathbf{h}^1$  mirror each other and the same is true for the band-pass pair  $\mathbf{h}^2$  and  $\mathbf{h}^3$ . Note that, as the spline order is growing, the MRs shapes of the p-filters  $\mathbf{h}^0$  and  $\mathbf{h}^1$  are approaching rectangles while the MRs of the p-filters  $\mathbf{h}^2$  and  $\mathbf{h}^3$  are shrinking. In the caption of Fig. 8, IS means interpolating spline and DS means discrete spline.

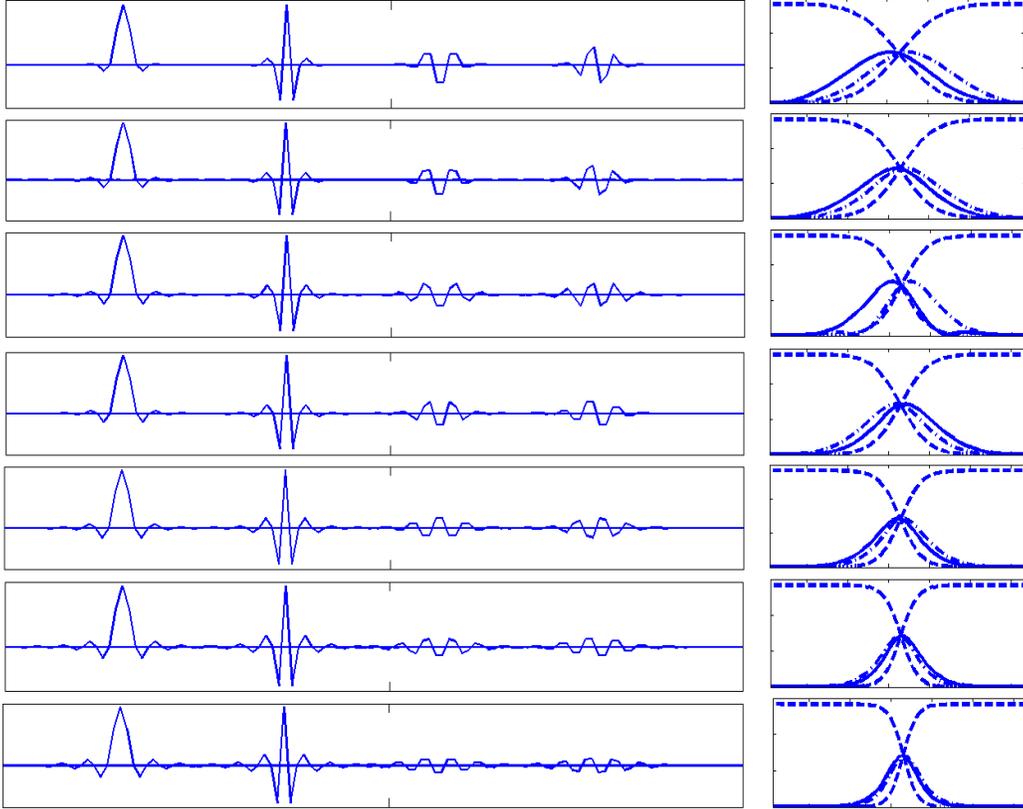


Figure 8: Impulse and magnitude responses of the IIR p-filter banks that generate tight frames. IS means interpolating spline and DS means discrete spline. Left pictures display the impulse responses of the p-filters. Left to right in the left picture:  $\mathbf{h}^0 \rightarrow \mathbf{h}^1 \rightarrow \mathbf{h}^2 \rightarrow \mathbf{h}^3$ . Right pictures display the magnitude responses of these p-filters:  $\mathbf{h}^0$  and  $\mathbf{h}^1$  (dashed lines),  $\mathbf{h}^3$  (dash-dot line) and  $\mathbf{h}^2$  (solid line). Top: quadratic IS. Second from top: cubic IS. Third from top: four degree IS. Center: DS of order 6. Third from bottom: DS of order 8. Second from bottom: DS of order 10. Bottom: discrete spline of order 12

## 7 Experimental results

In this section, we present experimental results where the developed framelets are applied to solve two image processing problems.

1. Image restoration from an input, which was degraded by blurring aggravated by random noise and by random loss of significant number of pixels. The images are transformed by periodic frames designed in Section 6, which are extended to the 2D setting in a standard tensor product way. The goal of our experiments is to compare between the performances of different tight and semi-tight frames in identical conditions.
2. Hyperspectral snapshot imaging.

### 7.1 Experiments on image restoration

In this section, we compare between the performance of different wavelet frames applied to a restoration task of degraded images.

### 7.1.1 Outline of the restoration scheme

The images are restored by the application of the *split Bregman iteration* (SBI) scheme [16] that uses the analysis based approach (see for example [20]).

Denote by  $\mathbf{u} = \{u[\kappa, \nu]\}$  the original image array to be restored from the degraded array  $\mathbf{f} = \mathbf{K} \mathbf{u} + \varepsilon$ , where  $\mathbf{K}$  denotes the 2D discrete convolution operator of the array  $\mathbf{u}$  with a kernel  $\mathbf{k} = \{k[\kappa, \nu]\}$  and  $\varepsilon = \{e_{k,n}\}$  is a random error array.  $\bar{\mathbf{K}}$  denotes the conjugate operator of  $\mathbf{K}$  that implements the discrete convolution with the transposed kernel  $\mathbf{k}^T$ . If some number of pixels are missing then the image  $\mathbf{u}$  should be restored from the available data

$$\mathbf{P}_\Lambda \mathbf{f} = \mathbf{P}_\Lambda (\mathbf{K} \mathbf{u} + \varepsilon), \quad (56)$$

where  $\mathbf{P}_\Lambda$  denotes the projection on the remaining pixels set.

The solution scheme is based on the assumption that the original image  $\mathbf{u}$  has a sparse representation in a frame domain. Denote by  $\tilde{\mathbf{F}}$  the frame expansion operator of the image  $\mathbf{u}$  where  $\mathbf{C} \stackrel{\text{def}}{=} \tilde{\mathbf{F}} \mathbf{u}$ ,  $\mathbf{C} = \{c[\kappa, \nu]\}$ , is the set of frame coefficients. Denote by  $\mathbf{F}$  the operator of reconstruction of the image  $\mathbf{u}$  from the set  $\mathbf{C}$ . We get  $\mathbf{F} \mathbf{C} = \mathbf{u}$ , and  $\mathbf{F} \tilde{\mathbf{F}} = \mathbf{I}$ , where  $\mathbf{I}$  is the identity operator.

An approximated solution to Eq. (56) is derived via the unconstrained minimization of the functional

$$\min_u \frac{1}{2} \|\mathbf{P}_\Lambda (\mathbf{K} \mathbf{u} - \mathbf{f})\|_2^2 + \lambda \|\tilde{\mathbf{F}} \mathbf{u}\|_1, \quad (57)$$

where  $\|\cdot\|_1$  and  $\|\cdot\|_2$  are the  $l_1$  and  $l_2$  norms of the sequences, respectively. If  $\mathbf{x} = \{x[\kappa, \nu]\}$ ,  $\kappa = 0, \dots, k$ ,  $\nu = 0, \dots, n$ , then

$$\|\mathbf{x}\|_1 \stackrel{\text{def}}{=} \sum_{\kappa=0}^{k-1} \sum_{\nu=0}^{n-1} |x[\kappa, \nu]|, \quad \|\mathbf{x}\|_2 \stackrel{\text{def}}{=} \sqrt{\sum_{\kappa=0}^{k-1} \sum_{\nu=0}^{n-1} |x[\kappa, \nu]|^2}.$$

Denote by  $\mathbf{T}_\vartheta$  the soft thresholding operator:

$$\mathbf{T}_\vartheta \mathbf{x} = \{x_\vartheta[\kappa, \nu]\}, \quad x_\vartheta[\kappa, \nu] \stackrel{\text{def}}{=} \text{sgn}(x[\kappa, \nu]) \max\{0, |x[\kappa, \nu]| - \vartheta\}.$$

Following [20], we solve the minimization problem in Eq. (57) by an iterative algorithm. We begin with the initialization  $\mathbf{u}^0 = 0$ ,  $\mathbf{d}^0 = \mathbf{b}^0 = 0$ . Then,

$$\begin{aligned} \mathbf{u}^{k+1} &:= (\bar{\mathbf{K}} \mathbf{P}_\Lambda \mathbf{K} + \mu \mathbf{I}) \mathbf{u} = \bar{\mathbf{K}} \mathbf{P}_\Lambda \mathbf{f} + \mu \mathbf{F} (\mathbf{d}^k - \mathbf{b}^k), \\ \mathbf{d}^{k+1} &= \mathbf{T}_{\lambda/\mu} (\tilde{\mathbf{F}} \mathbf{u}^{k+1} + \mathbf{b}^k), \\ \mathbf{b}^{k+1} &= \mathbf{b}^k + (\tilde{\mathbf{F}} \mathbf{u}^{k+1} - \mathbf{d}^{k+1}). \end{aligned} \quad (58)$$

The linear system in the first line of Eq. (58) is solved by the application of the *conjugate gradient* algorithm. The operations in the second and third lines are straightforward. The choice of the parameters  $\lambda$  and  $\mu$  depends on experimental conditions.

### 7.1.2 Results

The restoration algorithms were applied to the ‘‘Window’’, ‘‘Barbara’’, ‘‘Lena’’ and ‘‘Fingerprint’’ images. These images were blurred by the application of convolution that uses either the *motion kernel from Matlab* or a Gaussian kernel. Then, the images are further degraded by random removal of a large number of pixels. In some experiments, the degradation was aggravated by adding zero mean random noise.

In the experiments, we compare between the performances of a number of tight frames (TF) and semi-tight frames (STF) using four-channel filter banks, which are designed in Section 6. For comparison, we also included the frame, which uses the three-channel filter banks originated from interpolating second-order spline ([27]) - see Fig. 9.



Figure 9: Impulse and magnitude responses of the three-channel second-order spline filter banks that generate a tight frame

The framelets in Table 1 are denoted as follows:

$T_{1,0}^4$	linear 4-channel IS TF,	$S_{2,1}^4$	QU IS STF (ASY),
$T_{3,0}^4$	CU IS TF	$T_{4,0}^4$	4-th degree IS TF,
$S_{4,2}^4$	4-th degree IS STF (ASY),	$T_{6,0}^4$	QU QIS TF (non-interp. LPF),
$S_{6,2}^4$	QU QIS STF (ASY, non-interp. LPF),	$T_{8,0}^4$	pseudo-spline TF,
$S_{8,2}^4$	pseudo-spline STF (ASY),	$T_{100,0}^4$	DS of 10-th order TF,
$T_{120,0}^4$	DS of 12-th order TF,	$T_{1,0}^3$	linear 3-channel IS TF,

Table 1: List of spline-based frames using 4-channel p-filter banks. IS – interpolating spline, QIS – quasi-interpolating spline, DS – discrete spline, TF – tight frame, STF – semi-tight frame, LPF – low-pass p-filter, SY – symmetric, ASY – antisymmetric, QU – quadratic, CU – cubic.

The proximity between an image  $\tilde{\mathbf{u}}$  and the original image  $\mathbf{u}$  is evaluated visually and by the Peak-Signal-to-Noise ratio (PSNR)

$$PSNR \stackrel{\text{def}}{=} 10 \log_{10} \left( \frac{M 255^2}{\sum_{k=1}^M (x_k - \tilde{x}_k)^2} \right) dB \quad (59)$$

where  $M$  is the number of pixels in the image (in our experiments,  $M = 512^2$ ),  $\{x_k\}_{k=1}^M$  are the original pixels of the image  $\mathbf{u}$  and  $\{\tilde{x}_k\}_{k=1}^M$  are the pixels of the image  $\tilde{\mathbf{u}}$ .

**Restoration experiments for the “Window” image:** This image was taken from [20]. The colored image is of size of size  $512 \times 512 \times 3$  given in **Red**, **Green** and **Blue** wavebands.

**Experiment 1.1, image was blurred and pixels were removed:** Each component of the RGB image was blurred by the application of convolution that uses the motion kernel (MATLAB function `fspecial('motion',15,45)`). The PSNR of the colored blurred image became 23.31 dB. Then, 30% of the pixels from each color component were randomly removed (different pixels in different components). This reduced the PSNR to 9.43 dB. No random noise was added. The image was restored by 40 SBI using the parameters  $\lambda = 0.001$ ,  $\mu = 0.0007$  in Eq. (58), which were applied separately to each of the RGB components. The conjugate gradient solver used 100 iterations. The tight and semi-tight frames listed in Table 1 were tested. The decomposition was implemented down to the fifth level. The restored PSNR results are given in Table 2.

Frame	$T_{1,0}^4$	$S_{2,1}^4$	$T_{3,0}^4$	$T_{4,0}^4$	$S_{4,2}^4$	$T_{6,0}^4$	$S_{6,2}^4$	$T_{8,0}^4$	$S_{8,2}^4$	$T_{100,0}^4$	$T_{120,0}^4$	$T_{1,0}^3$
PSNR	40.83	41.52	41.44	41.02	41.01	41.94	41.96	42.13	<b>42.16</b>	40.48	40.25	41.08

Table 2: PSNR results after the restoration of the “Window” image from a blurred input where of 30% of its pixels were randomly removed

The best PSNR results (42.16 dB) was achieved by using the semi-tight frames  $S_{8,2}^4$  that were derived from the pseudo-spline using the anti-symmetric factorization. The pseudo-spline tight frame  $T_{8,0}^4$  produced almost the same result and results produced by the frames  $S_{6,2}^4$  and  $S_{6,0}^4$ , which were derived from the quadratic QIS, are very close. The performance of the tight frames  $T_{1,0}^4$  and  $T_{1,0}^3$ , which were derived from the piece-wise linear splines, were significantly worse as well as the performance of the tight frames  $T_{100,0}^4$  and  $T_{120,0}^4$  that have many LDVMs.

Figure 10 displays the restoration result. Visually, the restored image hardly can be distinguished from the original one.



Figure 10: Top left: Source input - “Window” image. Top right: Blurred. PSNR=23.31 dB. Bottom right: After random removal of 30% of its pixels from each color component. PSNR=9.43 dB. Bottom left: The best restored image achieved by the semi-tight frame  $S_{8,2}^4$ . PSNR=42.16 dB

**Experiment 1.2, Image blurred noisy and pixels were randomly removed:** The image was blurred by convolution with the motion kernel (MATLAB function `fspecial('motion',15,45)`) and a weak random zero-mean Gaussian noise, whose STD is  $\sigma = 5$ , was added to each of the RGB matrices separately. The PSNR was 22.96 dB. Then, 30% of the pixels from each matrix were randomly removed from each color component. This reduces the PSNR to 9.42 dB. The image was restored by 40 SBI, using the parameters  $\lambda = 0.3$ ,  $\mu = 0.07$  in Eq. (58). The conjugate gradient solver used 10 iterations. Tight and semi-tight frames listed in Table 1 were tested. The restored PSNR results are given in Table 3. Decomposition is implemented down to the second level.

Frame	$T_{1,0}^4$	$S_{2,1}^4$	$T_{3,0}^4$	$T_{4,0}^4$	$S_{4,2}^4$	$T_{6,0}^4$	$S_{6,2}^4$	$T_{8,0}^4$	$S_{8,2}^4$	$T_{100,0}^4$	$T_{120,0}^4$	$T_{1,0}^3$
PSNR	27.47	27.49	27.44	27.18	27.20	27.66	27.66	27.673	<b>27.674</b>	26.70	26.43	26.92

Table 3: PSNR results after the restoration of the “Window” image from a blurred noised input where of 30% of its pixels in each color component were randomly removed.

As in the previous experiment, the best PSNR 27.674 dB result was achieved by the application of the four-channel semi-tight frame  $\mathbf{S}_{8,2}^4$  derived from the pseudo-spline using anti-symmetric factorization. Again, the pseudo-spline tight frame  $T_{8,0}^4$  produced almost the same result and results produced by the frames  $S_{6,2}^4$  and  $S_{6,0}^4$  derived from the quadratic QIS are very close. However, in this experiment, performance of the majority of the remaining frames does not differ much from the performance of the above mentioned ones. It is not true for the tight frames  $T_{1,0}^3$  derived from piece-wise linear spline and the tight frames  $T_{100,0}^4$  and  $T_{120,0}^4$ , which have many LDVMs.

The restoration result, which uses the frame  $\mathbf{S}_{8,2}^4$ , is displayed in Fig. 11. We observe that the noise is completely removed and the image is deblurred. The colors in the image are restored satisfactorily.



Figure 11: Top left: Source input - the “Window” image. Top right: Blurred and noised. PSNR=22.96 dB. Bottom right: After random removal of 30% of its pixels from each color component. PSNR=9.42 dB. Bottom left: The image restored by the semi-tight frame  $\mathbf{S}_{8,2}^4$ . PSNR=27.674 dB

### Results from the restoration of the “Barbara” image

**Experiment 2.1, Image blurred and pixels removal:** In these experiment, the grayscale “Barbara” image was restored after it was blurred by a convolution with the Gaussian kernel (MATLAB function `fspecial('gaussian',[5 5],5)`) and its PSNR became 23.32 dB. Then, 50% of its pixels were randomly removed.

This reduces the PSNR to 7.55 dB. Random noise was not added. The image was restored by 50 SBI using the parameters  $\lambda = 0.001$ ,  $\mu = 0.005$  in Eq. (58). The conjugate gradient solver used 100 iterations. Tight and semi-tight frames listed in Table 1 were tested. The decomposition is implemented down to the second level. The restored PSNR results are given in Table 4.

Frame	$T_{1,0}^4$	$S_{2,1}^4$	$T_{3,0}^4$	$T_{4,0}^4$	$S_{4,2}^4$	$T_{6,0}^4$	$S_{6,2}^4$	$T_{8,0}^4$	$S_{8,2}^4$	$T_{100,0}^4$	$T_{120,0}^4$	$T_{1,0}^3$
PSNR	29.29	29.83	30.11	30.43	30.28	29.84	29.71	29.91	29.76	<b>30.50</b>	30.48	29.21

Table 4: PSNR results after the restoration of the “Barbara” image from a blurred input where of 50% of its pixels were randomly removed.

Unlike the experiments with the “Window” image, the best performance was achieved by using the tight frames  $T_{100,0}^4$  and  $T_{120,0}^4$ , which are derived from the interpolating discrete splines of tenth and twelfth orders, respectively. Recall that, for the frame  $T_{100,0}^4$ , the framelet  $\psi_{[1]}^1$  has 10 LDVMs, while  $\psi_{[1]}^2$  and  $\psi_{[1]}^3$  have 4 and 6 LDVMs, respectively. A close result was achieved by using the tight frame  $T_{4,0}^4$  derived from the fifth-order interpolating polynomial spline. The restoration result from the application of the frame  $T_{100,0}^4$  is displayed in Fig. 12. We observe that fine texture of the image, which is undistinguishable in the blurred image, is restored accurately.



Figure 12: Top left: Source input - “Barbara” image. Top right: Blurred and noised. PSNR=23.32 dB. Bottom right: After random removal of 50% of its pixels. PSNR=7.55 dB. Bottom left: The restored image by application of the tight frame  $T_{100,0}^4$ . PSNR=30.5 dB

### Results from the restoration of the “Lena” image

**Experiment 3.1, Image strongly blurred and distorted by curves:** The source image was strongly blurred by convolution with a Gaussian kernel (MATLAB function `fspecial('gaussian',[12,12],12)`) to get PSNR=23.64 dB. The blurred image was distorted by randomly drawn curves to get PSNR=16.44 dB. Noise was not added. The image was restored by 70 SBI with the parameters  $\lambda = 0.03$  and  $\mu = 0.17$  by the application of the four-channel frames listed in Table 1. The decomposition was implemented down to the fifth level. The conjugate gradient solver used 15 iterations. The PSNR results are given in Table 5.

Frame	$T_{1,0}^4$	$S_{2,1}^4$	$T_{3,0}^4$	$T_{4,0}^4$	$S_{4,2}^4$	$T_{6,0}^4$	$S_{6,2}^4$	$T_{8,0}^4$	$S_{8,2}^4$	$T_{100,0}^4$	$T_{120,0}^4$	$T_{1,0}^3$
PSNR	29.12	29.13	29.07	29.01	28.99	29.16	29.222	29.19	<b>29.227</b>	28.90	28.88	29.10

Table 5: PSNR results after the restoration of the “Lena” image from a blurred input distorted by randomly drawn curves

The best results PSNR=29.227 dB and PSNR=29.222 dB were achieved by the application of the four-channel semi-tight frames  $S_{8,2}^4$  and  $S_{6,2}^4$  derived from the the pseudo-spline and from the quadratic quasi-interpolating spline, respectively. The respective tight frames  $T_{8,0}^4$  and  $T_{6,0}^4$  produced a little bit of a lower PSNR. However, the rest of the tested frames, except of  $T_{100,0}^4$  and  $T_{120,0}^4$ , performed similarly. Figure 13 displays the best restoration result. We observe that the corrupting curves are completely removed and the image becomes deblurred.



Figure 13: Top left: Source input - “Lena” image. Top right: Blurred. PSNR=23.64 dB. Bottom right: Distorted by randomly drawn curves. PSNR=16.44 dB. Bottom left: The restored image by the application of the semi-tight frame  $S_{8,2}^4$ . PSNR=29.227 dB

### Results from the restoration of the “Fingerprint” image

**Experiment 4.1, Image blurred, strongly noisy and pixels missing:** The “Fingerprint” image was affected by a strong zero-mean white noise with STD  $\sigma = 20$  after being blurred by convolution with the Gaussian kernel (MATLAB function `fspecial('gaussian',[5 5],5)`) to get PSNR=19.65 dB. Then, 50% of its pixels were randomly removed and this produced PSNR=9.05 dB. The image was restored by 50 SBI with the parameters  $\lambda = 2.3$  and  $\mu = 0.45$  using the tight and semi-tight frames listed in Table 1. The decomposition is implemented down to the sixth level. The conjugate gradient solver used 11 iterations. The PSNR results are given in Table 6.

Frame	$T_{1,0}^4$	$S_{2,1}^4$	$T_{3,0}^4$	$T_{4,0}^4$	$S_{4,2}^4$	$T_{6,0}^4$	$S_{6,2}^4$	$T_{8,0}^4$	$S_{8,2}^4$	$T_{100,0}^4$	$T_{120,0}^4$	$T_{1,0}^3$
PSNR	22.64	23.50	23.64	23.78	23.75	23.29	23.25	23.30	23.24	<b>23.80</b>	23.73	22.20

Table 6: PSNR results after the restoration of the “Fingerprint” image from a blurred noisy input where 50% of its pixels were randomly removed

Here, the distribution of the PSNR results is very similar to the distribution from the “Barbara” Experiment 2.1. The best PSNR=23.80 dB and PSNR=23.78 dB results were achieved by the application of the tight frames  $T_{100,0}^4$  and  $T_{4,0}^4$  derived from the tenth-order interpolating discrete spline and interpolating polynomial spline of fifth order. Close results were produced by the semi-tight frame  $S_{4,2}^4$  and the tight frame  $T_{120,0}^4$ . Figure 14 displays the restoration result produced by application of the  $T_{100,0}^4$  frame. Despite the strong noise and loss of a significant number of pixels, the structure of the image is effectively restored.

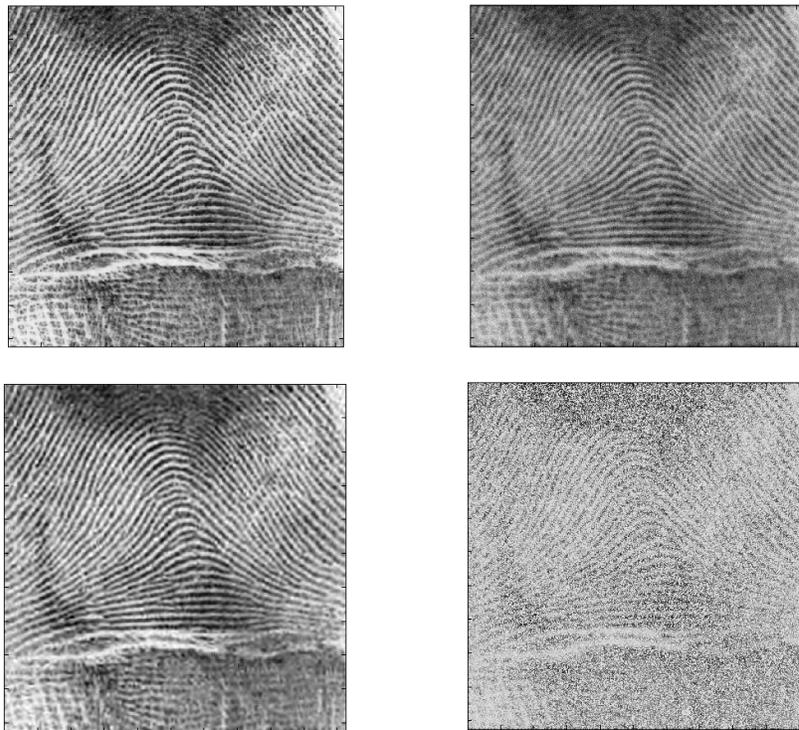


Figure 14: Top left: Source input - “Fingerprint” image. Top right: Blurred. PSNR=19.65 dB. Bottom right: After random removal of 50% of its pixels. PSNR=9.05 dB. Bottom left: The image was restored by the application of the tight frame  $T_{100,0}^4$ . PSNR=23.80 dB

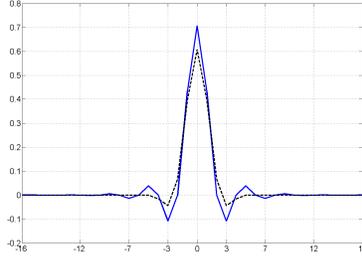


Figure 15: Solid line: IR of the IIR filter  $\mathbf{h}_{4,0}^0$ . Dashed line: IR of the FIR filter  $\mathbf{h}_{8,0}^0$

### 7.1.3 Discussion

In the above experiments, the designed four-channel frames proved to be highly useful for image restoration. The presented examples can be separated into two groups:

**“Window” and “Lena” images:** For these images (Experiments 1.1, 1.2 and 3.1), which are relatively smooth, the best results were produced by using the semi-tight frames that were derived from the quasi-interpolating quadratic spline and the pseudo-spline. The low-pass p-filters in generating p-filter banks are non-interpolating and have a long IR (9 taps). Note that in all the three experiments, the semi-tight frames outperformed their tight counterparts. In these experiments, the tight frames  $T_{100,0}^4$  and  $T_{120,0}^4$  with high number of LDVMs, which are derived from interpolating discrete splines, produced significantly inferior results.

**“Barbara” and “Fingerprint” images:** Both images have fine texture, which are satisfactorily restored after strong blurring, corruption by noise (“Fingerprint”) and after random removal of 50% of their pixels. For this pair of images, the distribution of the restoration results between frames are different. The best results were produced by using the tight frames  $T_{100,0}^4$ ,  $T_{120,0}^4$  and  $T_{4,0}^4$ . All these frames outperformed the frames that were derived from the pseudo-spline and from the quadratic QIS, which dominated the achievement of the best results in the Experiments 1.1,1.2 and 3.1. The frames  $T_{1,0}^4$  and  $T_{1,0}^3$ , which were derived from the piece-wise linear splines, demonstrated the worst performance.

All the three tight frames  $T_{100,0}^4$ ,  $T_{120,0}^4$  and  $T_{4,0}^4$ , originated from interpolating splines and the low-pass p-filters in generating filter banks, are interpolating. All the filter banks use p-filters whose IR are infinite. The framelets in all these frames have many LDVMs. In non-periodic setting, filtering with such filters is implemented in a recursive mode (see for example [5]). The transfer functions of non-periodic counterparts of the corresponding filter banks are rational functions, where numerators and denominators are the Laurent polynomials of high degrees. Therefore, for the transform’s implementation, many recursive passes are needed that results in a high computational cost.

However, the computational cost of the recursive implementation of IIR filtering is comparable and, sometimes even lower than the computational cost of FIR filtering, where filters have similar properties. Some results of such a comparison are given in [35].

There is a well known belief that infinite IR cause the corresponding wavelets to have large support in comparison to the wavelets generated by FIR filters. However, the impulse responses of IIR filters with rational transfer functions and the corresponding wavelets decay exponentially as time grows ([35]). Therefore, the effective support of these wavelets is compact. Figure 15 displays the impulse responses of the low-pass filters  $\mathbf{h}_{4,0}^0$  (IIR) and  $\mathbf{h}_{8,0}^0$  (FIR), which generate the frames  $T_{4,0}^4$  and  $T_{8,0}^4$ , respectively. Recall that the IIR filter  $\mathbf{h}_{4,0}^0$  locally restores fifth-degree polynomials, while the FIR filter  $\mathbf{h}_{8,0}^0$  locally restores only cubic polynomials.

A great advantage of the periodic setting is that the computational cost of the transform’s implementation does not depend on the spline order and on the number of LDVMs. There is no difference between the application of FIR and IIR p-filters. For example, the CPU time for implementing 50 Bregman iterations on the ‘Fingerprint’ image with simple FIR filter bank  $T_{1,0}^4$  takes 15.71 seconds, while the same procedure with the complicated IIR filter bank  $T_{120,0}^4$  takes 15.70 seconds. This fact proves the flexibility in choosing the proper frame.

There are some reservations in the image processing community about using periodic setting. There are claims that, while time-domain implementation of wavelet and frame transforms require some extension of signal/image beyond the boundaries, the periodic extension may produce undesirable artifacts near the boundaries. But this is not the case in our scheme. Actually, we do not extend the processed images. Filtering is implemented via the discrete Fourier transform that is perfectly invertible. *No boundary artifacts are produced.* This statement is illustrated in Fig. 16. This figure consists of three sections separated by vertical lines. In the top left section, column # 15 from the original “Barbara” image is displayed versus the column taken from the image restored by 50 SBI in Experiment 2.1. The two images in the bottom display fragments adjacent to the upper and lower edges of the column. In the central section, the same exposition was done for column # 370 from “Barbara”. The right section displays column # 470 from the “Fingerprint” image, which was restored from blurred and severely noised input in Experiment 4.1.

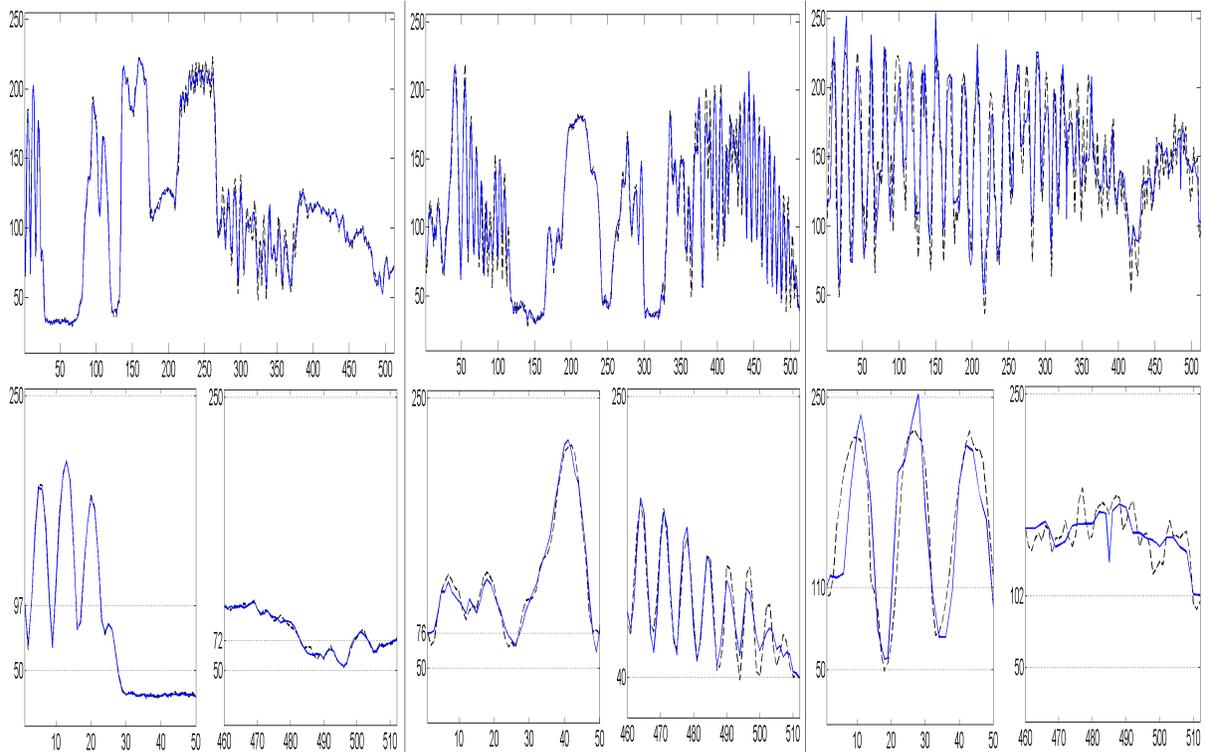


Figure 16: Left section: Top – original column # 15 from the “Barbara” image (dashed line) versus the column from the restored image. Bottom: left – left fragments of the lines; right – right fragments of the lines. Central section: The same for column # 370 from “Barbara”. Right section: The same for column # 470 from the “Fingerprint” image

We see that in the “Barbara” columns, the restored lines coincide with the original lines at the edges. In the “Fingerprint” column, the discrepancy between the restored and the original curves at the edges is small and does not exceed the discrepancy at the internal points. No boundary effects persist. Both images were restored using the tight frame  $T_{100,0}^4$  that was originated from the tenth-order discrete interpolating spline.

Increase above 50 of the number of SBIs does not contribute to the restoration quality. In some cases, as in the “Window” and “Barbara” experiments, a big number of iterations of the conjugate gradient solver increased the restoration quality. However, in the experiments with the strongly degraded “Lena” and “Fingerprint” images, increase in the number of iterations above 15 depleted the quality.

The choice of the regularization parameters  $\lambda$  and  $\mu$  is of critical importance. The restoration quality is sensitive to them. Even a small change in these parameters significantly affects the restoration quality. The number of decomposition levels is also important.

## 8 Snapshot spectral imaging

In this section, we briefly outline a potential application for spline-based wavelet frames: developing of a snapshot hyperspectral imaging (HSI) camera. HSI is commonly known as imaging with a large number of wavelengths in a given wavelength range. For example, the resolution range can be between 10 and hundreds or even thousands of wavelengths. “Snapshot” HSI imagers perform simultaneous (instantaneous) acquisition of spatial and spectral data in a single snapshot. The acquired data is processed that results in a spectral cube of a source object. The spectral cube includes light intensity data in two spatial dimensions and one spectral dimension and it is expressed as a three-dimensional (3D) matrix.

An algorithm, which transforms a snapshot image taken by a regular digital camera into a spectral cube, is presented in [17] and it will be described in full details in one of our next papers. Here, we present the experimental results from a mathematical modeling of the imaging procedure. The snapshot spectral imager comprises of an imaging lens, a dispersed image sensor and a diffuser inserted in the optical path between a source image and a pixelated (as. e.g. in a digital camera) image sensor. The diffuser provides a dispersed-diffused image at the dispersed image sensor. A plurality of spectral images of the source object in different spectral bands is obtained in a single shot by reconstruction involving the dispersed-diffused image. The diffuser may be included internally in a digital camera.

Each pixel in a dispersed-diffused image includes a linear mixture of spectral and spatial information from surrounding pixels. A reconstruction of the spectral cube is performed using an optimization process aimed to compensate for the underdetermined nature of the problem.

### 8.1 Mathematical model of the HSI snapshot

As source images we used the hyperspectral (HS) digital images from [15] that we downloaded from the cite [http://personalpages.manchester.ac.uk/staff/david.foster/Hyperspectral\\_images\\_of\\_natural\\_scenes\\_04.html](http://personalpages.manchester.ac.uk/staff/david.foster/Hyperspectral_images_of_natural_scenes_04.html). The results are outlined from experiments on a fragment of the image called `ref_ribeira1bbb_reg1`. This image is a spectral cube of size  $1017 \times 1340 \times 33$ , It contains 33 monochromatic images of size  $1017 \times 1340$ . Each image from the sequence of 33 images represents the intensity from the reflectance spectrum of 10 nanometers width. This covers the diapason 400 to 720 nanometers. The whole HS image and the fragment  $\mathbf{F}_0$  of size  $256 \times 256 \times 33$ , which the experiment were conducted on, are displayed in Fig. 17. For better visualization, the HS images were converted to RGB colored images using a conversion routine taken from the above website.

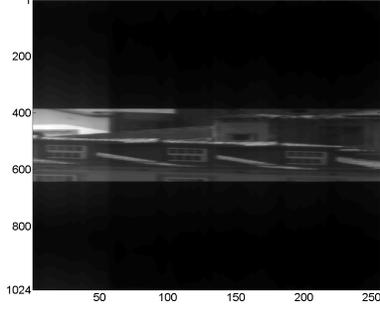


Figure 18: Dispersed-diffused image (DDI)  $\mathbf{Y}$



Figure 17: Left: The whole HS image converted to RGB. Right: A fragment of size  $256 \times 256$  pixels

The spectral cube  $\mathbf{F}_0$  was transformed into a 2D block-wise source matrix  $\mathbf{X}_0$  of size  $8448 \times 256$  ( $8448 = 256 \times 33$ ) by concatenating the  $256 \times 256$  matrices  $\mathbf{X}_0^l$ ,  $l = 1, \dots, 33$ , related to different wavebands:  $\mathbf{Z}_0 = (\mathbf{Z}_0^1, \mathbf{Z}_0^2, \dots, \mathbf{Z}_0^{33})^T$ .

The physical diffuser is a diffraction grating with a specific design of grooves. Its action on the optical flow is modeled by multiplication of the source matrix  $\mathbf{Z}_0$  with a block-wise Toeplitz matrix  $\mathbf{A} = (\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_{33})$ , where each sub-matrix  $\mathbf{A}_l$  is of size  $1024 \times 256$ . The produced matrix  $\mathbf{Y} \stackrel{\text{def}}{=} \mathbf{A} \cdot \mathbf{Z}_0 = \sum_{l=1}^{33} \mathbf{A}_l \cdot \mathbf{Z}_0^l$ , of size  $1024 \times 256$  is regarded as a dispersed-diffused image (DDI), which serves as an input to the restoration processing. The DDI is displayed in Fig. 18.

## 8.2 Outline of the restoration scheme

Unlike the unconstrained minimization, which was utilized in Section 7 for restoration of corrupted images, we reconstruct the HS image from the DDI  $\mathbf{Y}$  by solving the following constrained minimization problem:

$$\min_{\mathbf{X}} \|\tilde{\mathbf{F}} \mathbf{X}\|_1, \text{ subject to } \|\mathbf{A} \cdot \mathbf{X} - \mathbf{Y}\|_2^2 \leq \sigma^2, \quad (60)$$

where  $\tilde{\mathbf{F}} \mathbf{X}$  denotes a 2D direct frame transform applied to blocks of a block-wise matrix  $\mathbf{X} = (\mathbf{X}^1, \mathbf{X}^2, \dots, \mathbf{X}^{33})^T$ :

$$\tilde{\mathbf{F}} \mathbf{X} = \mathbf{D} = (\mathbf{D}^1, \mathbf{D}^2, \dots, \mathbf{D}^{33})^T, \quad \mathbf{D}^l \stackrel{\text{def}}{=} \tilde{\mathbf{F}} \mathbf{X}^l, \quad l = 1, \dots, 33. \quad (61)$$

The inverse frame transform applied to blocks of a block-wise matrix  $\mathbf{D}$  is

$$\mathbf{F} \mathbf{D} = \bar{\mathbf{X}} = (\bar{\mathbf{X}}^1, \bar{\mathbf{X}}^2, \dots, \bar{\mathbf{X}}^{33})^T, \quad \bar{\mathbf{X}}^l \stackrel{\text{def}}{=} \mathbf{F} \mathbf{D}^l, \quad l = 1, \dots, 33. \quad (62)$$

Following [23], we solve the minimization problem in Eq. (60) by the following SBI algorithm. We begin with the initialization  $\mathbf{X}^0 = 0$ ,  $\mathbf{d}^0 = \mathbf{b}^0 = \mathbf{c}^0 = 0$ . Then,

$$\begin{aligned}\mathbf{X}^{k+1} &= (\mu \mathbf{A}^T \cdot \mathbf{A} + \lambda \mathbf{I})^{-1} (\mu \mathbf{A}^T \cdot (\mathbf{Y} - \mathbf{c}^k) + \lambda \mathbf{F}(\mathbf{d}^k - \mathbf{b}^k)), \\ \mathbf{d}^{k+1} &= \mathbf{T}_{1/\lambda} \left( \tilde{\mathbf{F}} \mathbf{X}^{k+1} + \mathbf{b}^k \right), \\ \mathbf{b}^{k+1} &= \mathbf{b}^k + \left( \tilde{\mathbf{F}} \mathbf{X}^{k+1} - \mathbf{d}^{k+1} \right), \\ \mathbf{c}^{k+1} &= \mathbf{c}^k + (\mathbf{A} \cdot \mathbf{X}^{k+1} - \mathbf{Y}).\end{aligned}\tag{63}$$

The linear system in the first line of Eq. (63) is solved by the direct matrix multiplication. The operations in the second to fourth lines are straightforward. The parameters choice  $\lambda$  and  $\mu$  depends on experimental conditions. The iterations are stopped once the inequality in Eq. (60) for the norms  $\|\mathbf{A} \cdot \mathbf{X}^{k+1} - \mathbf{Y}\|_2^2$  is satisfied.

### 8.3 Results

In this experiment, the spectral cube of size  $256 \times 256 \times 33$  was developed from the DDI  $\mathbf{Y}$  of size  $1024 \times 256$  displayed in Fig. 18. The cube was formed as a result of 35 split Bregman iterations presented in Eq. (63). Different tight and semi-tight frames were tested. The best restoration results were achieved by the semi-tight  $S_{6,2}^4$  originated from the quadratic quasi-interpolating spline. The frame transform was decomposed down into the fourth level. Figure 19 displays the restored spectral cube converted to RGB color image versus the similarly converted original spectral cube. We observe that both images are very similar to each other and the colors are restored. The PSNR computed for all the pixels of the spectral cubes is 28.18 dB.



Figure 19: Left: Fragment of the original HS image converted to RGB of size  $256 \times 256$  pixels; four control points are marked. Right: Restored fragment, PSNR=28.18 dB

Restoration of the spectra at four control points, which are marked on Fig. 19, is displayed in Fig. 20.

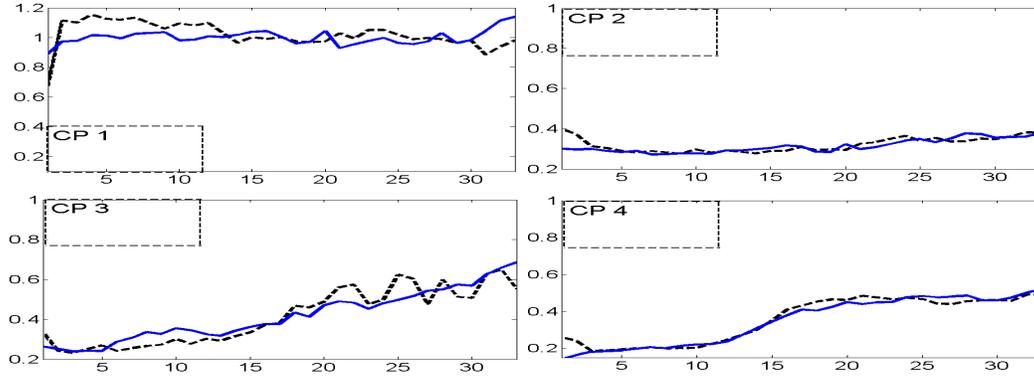


Figure 20: Spectra restoration at four control points, which are marked on Fig. 19. Dashed lines – original spectra; solid lines – restored spectra

We observe that the spectra at control points are satisfactorily restored. The diagram in Fig. 21 illustrates the restoration quality of different wavebands from the spectral cube. It shows the distribution of the PSNR values between the 33 wavebands. All the bands except band #1 (400 ÷ 410 nm), which is strongly noised, are satisfactorily restored. The highest PSNR=32.59 dB is observed at the restored band #19 (580 ÷ 590 nm).

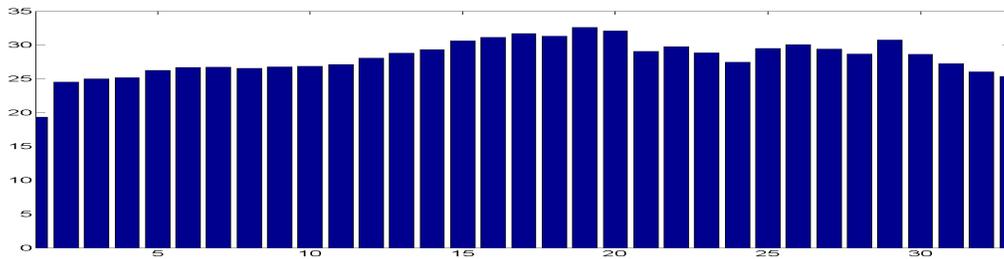


Figure 21: Restoration of different wavebands from the spectral cube. The bars represent PSNR values

Figure 22 displays the restoration of the wavebands #14 (530 – 540 nm), #19 (580 – 590 nm) and #29 (680 – 690 nm), with PSNR values 29.28 dB, 32.59 db and 30.95 dB, respectively. We observe that bands #19 and #29 are restored almost perfectly.

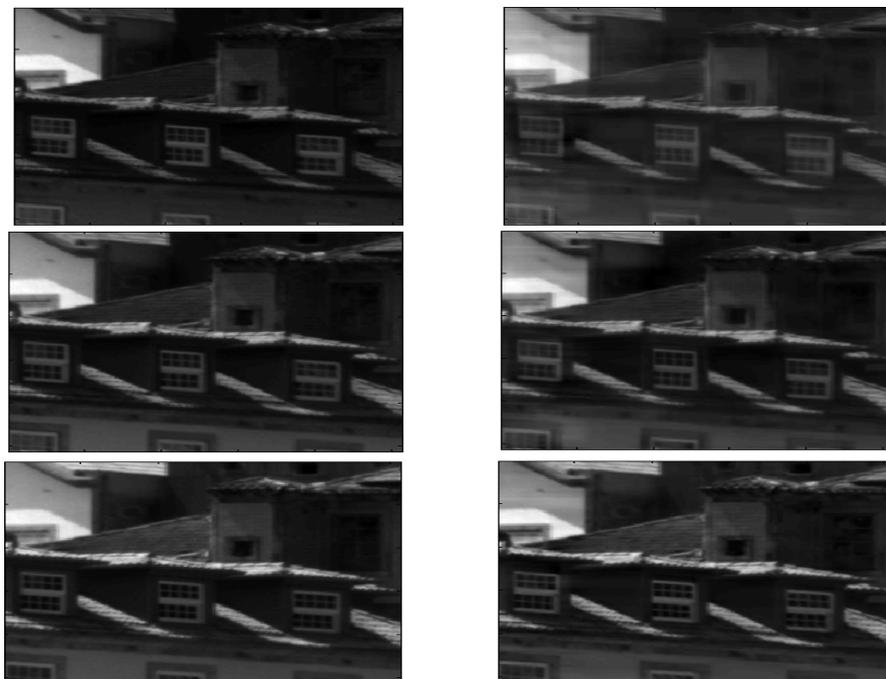


Figure 22: Restoration of wavebands #14 (top), #19 (center) and #29 (bottom). Left – original monochromatic images. Right – restored images

This and a number of other experiments led us to the conclusion that the Split Bregman Iteration scheme coupled with wavelet frames designed in the paper are a proper mathematical tool for converting the dispersed-diffused snapshot image into a spectral cube. Potentially, it paves the way for regular digital cameras to serve as hyperspectral imagers.

## References

- [1] O. Amrani, A. Averbuch, T. Cohen, and V. A. Zheludev. Symmetric interpolatory framelets and their error correction properties. *International Journal of Wavelets, Multiresolution and Information Processing*, 5(4):541–566, 2007.
- [2] A. Averbuch and V. Zheludev. Construction of biorthogonal discrete wavelet transforms using interpolatory splines. *Applied and Comp. Harmonic Analysis*, 12(12):25–56, 2002.
- [3] A. Averbuch and V. Zheludev. Wavelet transforms generated by splines. *International Journal of Wavelets, Multiresolution and Information Processing*, 5(2):257–292, 2007.
- [4] A. Z. Averbuch, A. B. Pevnyi, and V. A. Zheludev. Biorthogonal Butterworth wavelets derived from discrete interpolatory splines. *IEEE Trans. on Sign. Proc.*, 49:2682–2692, 2001.
- [5] A. Z. Averbuch, A. B. Pevnyi, and V. A. Zheludev. Butterworth wavelet transforms derived from discrete interpolatory splines: Recursive implementation. *Signal Processing*, 81:2363–2382, 2001.
- [6] A. Z. Averbuch, V. A. Zheludev, and T. Cohen. Interpolatory frames in signal space. *IEEE Trans. Sign. Proc.*, 54(6):2126–2139, June 2006.

- [7] A. Z. Averbuch, V. A. Zheludev, and T. Cohen. Tight and sibling frames originated from discrete splines. *Sign. Proc. J.*, 86(7):1632–1647, 2006.
- [8] B. Dong, Z. Shen. Pseudo-splines, wavelets and framelets. *Applied and Computational Harmonic Analysis*, 22(1):78–104, January 2007.
- [9] H. Bölcskei, F. Hlawatsch, and H. G. Feichtinger. Frame-theoretic analysis of oversampled filter banks. *IEEE Transactions on Sign. Proc.*, 46(12):3256–3268, Dec 1998.
- [10] C. K. Chui and W. He. Compactly supported tight frames associated with refinable functions. *Applied and Comp. Harmonic Analysis*, 8:293–319, 2000.
- [11] C. K. Chui, W. He, and J. Stöckler. Compactly supported tight and sibling frames with maximum vanishing moments. *Applied and Comp. Harmonic Analysis*, 13:224–262, 2002.
- [12] Z. Cvetković and M. Vetterli. Oversampled filter banks. *IEEE Transactions on Signal Processing*, 46(5):1245–1255, May 1998.
- [13] I. Daubechies, Bin Han, A. Ron, and Z. Shen. Framelets: Mra-based constructions of wavelet frames. *Applied and Computational Harmonic Analysis*, 14(1–46), 2003.
- [14] J. Romberg E. Candes and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59(8):1207–1223, 2006.
- [15] D. H. Foster, K. Amano, S. M. C. Nascimento, and M.J. Foster. Frequency of metamerism in natural scenes. *Journal of the Optical Society of America A*, 23:2359–2372, 2006.
- [16] T. Goldstein and S. Osher. The split Bregman method for l1-regularized problems. *SIAM J. Imaging Sciences*, 2(2):323–343, 2009.
- [17] M. Golub, M. Nathan, A. Averbuch, A. Kagan, V. A. Zheludev, and R. Malinsky. Snapshot spectral imaging based on digital cameras. *US Patent Application Publication Golub et al, No.13/762,560*, US 2013/0194481 A1, 2013.
- [18] V. K Goyal, J. Kovacevic, and J.A. Kelner. Quantized frame expansions with erasures. *Appl. and Comput. Harmonic Analysis*, 10(3):203–233, 2001.
- [19] V. K Goyal, M. Vetterli, and N. T. Thao. Quantized overcomplete expansions in  $\mathbb{R}^n$ : Analysis, synthesis and algorithms. *IEEE Trans. on Information Theory*, 44(1):16–31, 1998.
- [20] Z. Shen H. Ji and Y. Xu. Wavelet frame based image restoration with missing/damaged pixels. *East Asia Journal on Applied Mathematics*, 1:108–131, 2011.
- [21] C. Herley and M. Vetterli. Wavelets and recursive filter banks. *IEEE Trans. Signal Proc.*, 41(12):2536–2556, 1993.
- [22] S. Osher J. Cai, B. Dong and Z. Shen. Image restoration: total variation, wavelet frames, and beyond. *Journal of the American Mathematical Society*, 25:1033–1089, 2012.
- [23] Z. Shen J. Cai, S. Osher. Split Bregman methods and frame based image restoration. *Multiscale Modeling and Simulation: A SIAM Interdisciplinary Journal*, 8(2):337–369, 2009.
- [24] J. Kovacevic, P.L. Dragotti, and V. K Goyal. Filter bank frame expansions with erasures. *IEEE Trans. Inform*, 48(6):1439–1450, 2002.
- [25] A. V. Oppenheim and R. W. Schaffer. *Discrete-time signal processing*. New York, Prentice Hall, 2010.
- [26] G. Polya and G. Szegő. *Aufgaben und Lehrsätze aus der Analysis, Vol. II*. Springer, Berlin, 1971.

- [27] A. Ron and Z. Shen. Compactly supported tight affine spline frames in  $l^2(\mathbb{R})$ . *Mathematics of Computation*, 67:191–207, 1998.
- [28] B. Han S. Song Goh and Z. Shen. Tight periodic wavelet frames and approximation orders. *Applied and Computational Harmonic Analysis*, 31:228–248, 2011.
- [29] I. J. Schoenberg. Contribution to the problem of approximation of equidistant data by analytic functions. *Quart. Appl. Math.*, 4:45–99, 112–141, 1946.
- [30] Z. Shen. Wavelet frames and image restorations. In *Proc. Int. Congress of Mathematicians, vol. IV, Rajendra Bhatia eds.*, pages 2834–2863, Hyderabad, India, September 2010.
- [31] G. Strang and G. Fix. A Fourier analysis of the finite element variational method. *Construct. Asp. Funct. Anal.*, pages 796–830, 1971.
- [32] V. A. Zheludev, V. N. Malozemov and A. B. Pevnyi. Filter banks and frames in the discrete periodic case. *Proceedings of the St. Petersburg Mathematical Society*, **14**, (2008), *AMS Translations, Ser. 2*, 228:1–12, 2009.
- [33] V. A. Zheludev. Local spline approximation on a uniform grid. *U.S.S.R. Comput. Math. & Math. Phys.*, 27(5):8–19, 1987.
- [34] V. A. Zheludev. Periodic splines and the fast Fourier transform. *Comput. Math. & Math Phys.*, 32:149–165, 1991.
- [35] V. A. Zheludev. Interpolatory subdivision schemes with infinite masks originated from splines. *Advances in Comp. Math.*, 25:475–506, 2006.