YAN, Y., LIU, Y., YANG, M., ZHAO, H., CHAI, Y. and REN, J. 2020. Generic wavelet-based image decomposition and reconstruction framework for multi-modal data analysis in smart camera applications. *IET computer vision* [online], 14(7): computer vision for smart cameras and camera networks, pages 471-479. Available from: https://doi.org/10.1049/iet-cvi.2019.0780

Generic wavelet-based image decomposition and reconstruction framework for multi-modal data analysis in smart camera applications.

YAN, Y., LIU, Y., YANG, M., ZHAO, H., CHAI, Y. and REN, J.

2020

A Generic Wavelet-based Image Decomposition and Reconstruction Framework for Multi-modal Data Analysis in Smart Camera Applications

Yijun Yan[1], Yiguang Liu[2], Mingqiang Yang[3], Huimin Zhao[4], Yanmei Chai[5], Jinchang Ren[4,1]*

[1]Dept. of Electronic and Electrical Engineering, University of Strathclyde, Glasgow, U.K.
[2]College of Computer Science, Sichuan University, Chengdu, China
[3]School of Information Science and Engineering, Shandong University, Jinan, China
[4]School of Computer Science, Guangdong Polytechnic Normal University, Guangzhou, China
[5]School of Information, Central University of Finance and Economics, Beijing, China

[*]Corresponding Author: Dr J Ren (jinchang.ren@strath.ac.uk)

Note: The first two authors are of equal contributions.

Abstract: Effective acquisition, analysis and reconstruction of multi-modal data such as colour and multi-/hyper- spectral imagery is crucial in smart camera applications, where wavelet-based coding and compression of images are highly demanded. Many existing discrete wavelet filtering banks have fixed coefficients hence their performance is highly dependent on the signal/image being processed. To tackle this problem, a unified framework is proposed in this paper, which can produce a series of discrete wavelet filtering banks, where many existing discrete wavelet filtering banks become special cases of the framework. For each generated filtering bank, it consists of two decomposition filters and two reconstruction filters through an optimization process. The efficacy of the filtering banks produced by the framework has been validated in two case studies, including color image decomposition and reconstruction, and hyperspectral image classification. Comprehensive experiments have demonstrated the superior performance of the proposed framework, which will benefit the efficacy of smart camera and camera network applications.

1. Introduction

Signal and image decomposition is very useful for data compression, feature extraction in signal and image processing, machine learning, big data analytics and artificial intelligence, [1-3]. Actually, more effective information can be extracted from decomposed signals, and typical application tasks can be found in the rapid approximation of a signal by trigonometric polynomials (as a function of their degree), and the decomposition of a signal into a series of polynomials for compression and feature extraction [4]. Some classical approaches are discussed in detail as follows.

For a given data matrix, principal component analysis (PCA) seeks the best (in an L2-sense) low-rank representation of it, which holds a few optimality properties when the data are only mildly corrupted by small noise [5]. Singular spectrum analysis (SSA) decomposes an original data into several independent components which can be interpreted as trend, periodic components and noise [6]. Independent component analysis (ICA) is a framework for separating a mixture of different components into its constituents and has been successfully applied in many applications [7, 8]. More recently, sparse decomposition of the signals and images has attracted increasing attention [9, 10]. Among these aforementioned approaches, most of them use Singular Value Decomposition (SVD) for matrix decomposition, where, spectral analysis via spectral decomposition of a positive-definite kernel underlies a variety of classical approaches [11], such as PCA and SSA. Essentially, the classical SVD has associated with the decomposition of one featured space into the direct sum of invariant subspaces [12].

Unfortunately, SVD can only provide linear combinations of the data samples, which are notoriously difficult to interpret in terms of the data and the process in which the data is generated. As a result, CUR matrix decompositions were developed for improved data analysis [13], where a low-rank approximation of matrices with missing entries are proposed [14-16]. In [17], Stephane G. Mallat defines an orthogonal multiresolution representation called a wavelet representation, which is computed with a pyramidal algorithm based on convolutions with quadrature mirror filters (denoted as *qmf*()). That is $qmf(l_1, l_2, \dots, l_{n-1}, l_n) = (l_n, -l_{n-1}, \dots, l_2, -l_1)$.

In fact, the wavelet representation of signals has been widely used to address different practical problems in many research fields. In [18], Daubechies wavelet family is used in discrete wavelet transform for emotion recognition in speech. In [19], an underwater image enhancement technique is developed based on a fusion-based strategy and Daubechies wavelets. In [20], Biorthogonal wavelet filters are used to compress and reconstruct the ECG signals. In [21], ultrasound images are denoised by discrete wavelet transform with Symlet filter banking. In [22], Reverse Biorthogonal wavelet transform is applied for iris recognition, where other wavelet families such as Biorthogonal, Coiflets, Symlets and Daubechies were also investigated. In [23, 24], wavelet transform is used to extract features from hyperspectral images before image classification. In [25], Coeflet wavelet is used for continues Hindi speech coding. In [26], Fejer Korovkin wavelet is integraed with a Multiples Input-Multiples Ouput Autoregressive model to forecast the monthly fish catches.

In recent year, deep learning techniques have been attracted many attentions, where wavelet transform may be integrated with deep learning models in certain applications. In [27], Daubechies is used to reconstructed the wind sequence signal, followed by the deep brief network and random forest model to predict the wind speed. In [28], wavelets and deep neural network are integrated to detect the anomalies in the Seismic Electrical Signal for predicting earthquake activity. In [29], an image compression model is proposed where a deep neural network is used to encode the spatial pixel information in the wavelet transform domain.

However, the performance of deep learning models heavily relies on the number of training annotated data. There are diverse types of data captured by different sensors and cameras, such as single-channel signals and images, RGB color images, and multispectral and hyperspectral images. As a result, large volume multi-modal data makes the manual annotation becoming a labor intersive work, which bring a big challenge to deep learning. On the other hand, the selection of the most appropriate wavelet bank also becomes one critical problem. Existing wavelet representations, such as Daubechies, Coiflets and Symlets, are composed of various filters with fixed coefficients. To select the best filters for some specific applications and data, comprehensive experimental trials are needed, which is often very time-consuming. As a result, a generic framework is essential for generating suitable filters to satisfy the needs of different applications. To the best of our knowledge, there is no framework that can straightforwardly produce general decomposition and reconstruction (D&R) filters for signals. To this end, in this paper, we propose a generic framework with only one parameter for generating D&R filters. This parameter mainly decide the length of the filter, then all coefficients in the fitler will be obtained by proposed model.In the experiment, the proposed method is validated in two case studies, i.e. D&R of color images, and hyperspectral imaging classification.

In this paper, a generic framework is proposed to unify the D&R filters, and the major contributions are highlighted as follows:

- We propose an equation set model as a generic framework for generating D&R filters;
- We demonstrate that existing discrete wavelet filters such as Daubechies, Coiflets and Symlets are special cases of the proposed generic framework, and we also show simultaneously many other generated D&R filters;
- The efficacy of the proposed model is validated in two case studies, i.e. D&R of color images, and classification of hyperspectral imagery (HSI). Inspired by the works [30, 31], the wavelet filters are used to extract low-dimensional and discrimination features for image classification. The superior performance of the proposed model over several existing approaches has fully been demonstrated in comprehensive experiments.

The rest of this paper is organized as follows. Section 2 introduces the theory of the proposed D&R model with the whole deduction process and mathematical proofs, and evaluates how the key parameter affects the convergence speed and decomposition effects of the proposed model. Section 3 describes how the proposed model is implemented to nature image reconstruction and hyperspectral image classification, where comprehensive experiments and useful analysis are given. Finally, some concluding remarks are drawn in Section 4 along with discussions of future directions.

2. The Proposed Method

The proposed framework is derived from two key theorems and the process of comprehensive mathematical proof is given in this section. The implementation of the proposed framework is detailed in three subsections, i.e. the concept, the proof of concept, and algorithm implementation.

2.1. The concept

A real data sequence $L_d = [l_1, l_2, \dots, l_{2n-1}, l_{2n}]$ can act as the scaling function or decomposed low-pass filter if it satisfies the following n conditions and 1 constraint in Eq. (1)[32].

L_d can act as the decomposed low-pass filter, and the associated high-pass filter is $H_d = -qmf(L_d) = [-l_{2n}, l_{2n-1}, \dots, -l_2, l_1]$. The reconstructed low-pass filter L_r is $rev(L_d)$ where $rev[l_1, l_2, \dots, l_{2n-1}, l_{2n}] = [l_{2n}, l_{2n-1}, \dots, l_2, l_1]$, and its associated high-pass filter is $H_r = qmf(L_r) = [l_1, -l_2, \dots, l_{2n-1}, -l_{2n}]$.

For instance, let $n = 3$, hence we have 3 conditions in Eq. (1). We can derive $l_2 = \frac{-l_1 l_5}{l_6}$, $l_3 = \frac{(-\frac{l_1 l_5}{l_6}+l_6-l_1-l_5)(-\frac{l_1 l_5}{l_6}+l_6)}{(-\frac{l_1 l_5}{l_6}+l_6+l_1+l_5)}$ and $l_4 = \frac{-(-\frac{l_1 l_5}{l_6}+l_6-l_1-l_5)(l_1+l_5)}{(-\frac{l_1 l_5}{l_6}+l_6+l_1+l_5)}$, where l_1, l_5 and l_6 are randomly defined parameters.

When n takes 4, if l_1, l_6, l_7 and l_8 are predefined, the remaining parameters can be derived by solving a quadratic problem. Note that some sets of l_1, l_6, l_7 and l_8 combinations may lead to an unresolvable quadratic problem, i.e. no solutions can be found for L_d, H_d, L_r and H_r.

Table 1 Existing wavelet filters in different wavelet families.

	Wavelet family	Fejer-Korovkin	Coiflets	Daubechies	Symlets	Biorthogonal
	Filter length ($2n$)	4	6	8	10	12
Coefficients	l_1	-0.0462	-0.0157	-0.0106	0.0273	-0.0138
	l_2	0.0532	-0.0727	0.0329	0.0295	0.0414
	l_3	0.7533	0.3849	0.0308	-0.0391	0.0525
	l_4	0.6539	0.8526	-0.1870	0.1994	-0.2679
	l_5	-	0.3379	-0.0280	0.7234	-0.0718
	l_6	-	-0.0727	0.6309	0.6340	0.9667
	l_7	-	-	0.7148	0.0166	0.9667
	l_8	-	-	0.2304	-0.1753	-0.0718
	l_9	-	-	-	-0.0211	-0.2679
	l_{10}	-	-	-	0.0195	0.0525
	l_{11}	-	-	-	-	0.0414
	l_{12}	-	-	-	-	-0.0138

There are formulas for solving the cubic and quartic equations. For even higher orders of equations, the Abel–Ruffini theorem asserts that there is no general formula in radicals. When n is large (e.g. $n \geq 6$), there is no explicit formula for solving L_d parameters in Eq. (1) when half of the parameters are given. However, root-finding algorithms i.e. bracketing methods and iterative methods [33] may be used to find numerical approximations of the roots for Eq. (1). An efficient iterative algorithm to solve Eq. (1) will be given later.

It is worth noting that many existing wavelet filtering banks are special solutions of Eq. (1), and actually Eq. (1) constructs a framework consisting of many wavelet transforming sets. For instance, the coefficients of L_d in Daubechies wavelets "DB3" is known as [0.0352, −0.0854,−0.1350, 0.4599, 0.8069, 0.3327]. With given l_1, l_5 and l_6 , l_2, l_3 and l_4 can be derived from Eq. (1) and the results are the same as those given in "DB3". In analogy, Coiflets coefficients are all in accordance with Eq. (1). As summarized in Table 1, that fact that many existing wavelet banks can be depicted by Eq. (1) has shown that there are many options to implement the decomposition of signals. In addition, we can also impose additional constraints to rule partial coefficients of L_d. For example, when $n = 3$, three coefficient l_1, l_5 and l_6 can be preconditioned so as to make the energies distributed in components to have larger differences. Therefore, the sparsity is outwardly stuck out and the trivial parts are removed to reduce the storage requirement.

2.2. *Proof of the concept*

When $l_1, l_2, \dots, l_{2n-1}, l_{2n}$ satisfy Eq. (1), the filters $L_d = [l_1, l_2, \dots, l_{2n-1}, l_{2n}]$ and $H_d = -qmf(L_d)$, decompose a signal into two parts. On the other hand, the two filters $L_r = rev(L_d)$ and $H_r = qmf(L_r)$ can reconstruct the primary signal from these two components.

Let $S = [s_1, s_2, \dots, s_{m-1}, s_m]$ denote a real data sequence. Before S is convoluted with $L_d = [l_1, l_2, \dots, l_{2n-1}, l_{2n}]$ (assuming $2n < m$ without loss of generality) and $H_d = -qmf(L_d)$, S is circularly extended as Eq.(2). The low-frequency component $P = [p_1, p_2, \dots, p_{n-1}, p_n]$ is derived by downsampling the convolution result, $S_e * L_d$, at all even positions. The associated high-frequency component $Q = [q_1, q_2, \dots, q_{n-1}, q_n]$ can be analogously derived from $S_e * H_d$. Each parameter in P and Q can be represented by Eq.(3), where $2(k+n) \leq m,\ \ n+k \geq 1$.

The length of P and Q is only half of S. To reconstruct S from P and Q using L_r and H_r, Eq. (4) needs to hold. By substituting Eq. (3) into Eq. (4), the first component s_1 in S can be rewritten in Eq. (5). By further substituting Eq. (4) into Eq. (5), we can have Eq.(6) and Eq.(7). To make Eq. (7) hold, the Eq. (8) must hold. In fact, 2n conditions can be obtained in Eq. (8) and each of them is produced by multiplying $[l_1, l_2, \dots, l_{2n-1}, l_{2n}]$ with each column of M (see Eq.(9)).

In Eq. (9), the 1st expression acts for normalization of L_d, which is usually required as a constraint to avoid additional energy in the signal S. The 2nd and 3rd conditions are the same, i.e. $l_1 l_3 + l_2 l_4 + l_3 l_5 + \cdots + l_{2n-2} l_{2n} = 0$. We can derive that the $2i^{th}$ and $2(i+1)^{th}$ $(i = 2, \dots, n-1)$ conditions are the same, i.e. $l_1 l_{2i+1} + l_2 l_{2i+2} + \cdots + l_{2n-2i} l_{2n} = 0$ and the final condition is an identity as shown above. On the other hand, $S * H_d$ is used to extract the high-frequency components of S, which actually performs a weighted gradient operation. Therefore, the summation of H_d equals to 0. As a result, the condition, $l_1 + l_3 + \cdots + l_{2n-1} = l_2 + l_4 + \cdots + l_{2n}$, is imposed. In summarization, we have n conditions and 1 constraint, which makes the Eq. (1) hold.

Similar like the operations on s_1, we can also derive the expression of s_{2k-1}, s_{2k} in Eq.(10). By substituting Eq. (3) and Eq. (4) into Eq. (10), we still can obtain the n conditions and 1 constraint from the above formulas. So far, the concept of our generic model has been proved.

2.3. *The algorithm*

As mentioned before, it is hard to determine the parameters in L_d when n is large. Therefore, we rewrite Eq. (1) as a cost function and optimize it until the loss is sufficiently small. There are three steps to implement this process as follows.

Step 1: Randomly initialize $l_1, l_2, \dots, l_{2n-1}, l_{2n}$, and set an error or loss threshold value ϵ and an iteration number I;

Step 2: Tune l_j for $(j = 1,2, \dots, 2n)$ in turn with the other $2n-1$ coefficient l_i $(i \neq k)$ known/given;

Step 3: Calculate the loss ϵ by summing the absolute residuals of all conditions in Eq. (1). If the total residual is less than ϵ or the iteration number reaches I, exit the iterative process. Otherwise, go back to Step 2.

After the iterations in Steps 1-3, the numerical solution of l_j in Eq. (1) can be estimated. To complete Step 2, we build a Lyapunov function (Eq.(11)) based on n conditions in Eq. (1).

Each l_j in $\mathcal{L}(\cdot)$ is kept tuning until $\mathcal{L}(\cdot)$ is minimized. When l_j is a variable, the other coefficients are regarded as constants. Therefore, Eq. (11) is equivalent to a strictly convex quadratic function and can be rewritten as Eq.(12). To solve Eq. (12), we need to address Eq. (13) to minimize $\mathcal{L}(\cdot)$.

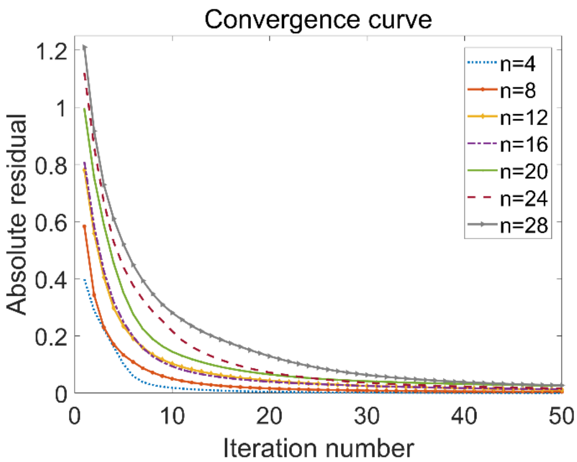

Fig. 1. *The convergence curves with changing parameter n.*

Let the i^{th} condition in Eq. (1) be C_i, Eq. (13) can be rewritten as Eq.(14). So far, Eq. (12) can be regarded as n ordinary least square problems. The deduction process is given by Eq. (15-18).

By solving the Eq. (14) and Eq. (15), we can get Eq.(16). Where $l_{j-2(n-k)}$ or $l_{j-2(n+k)}$ becomes zero if $j-2(n-k) < 0$ or $-2(n+k) > 2n$ $(k = 1,2, \dots, n-1)$.

Let $[\alpha_1, \alpha_2, \dots, \alpha_n]$ and $[b_1, b_2, \dots, b_n]$ are defined in Eq. (17), the expression of l_j can be rewritten as a standard least square representation by Eq.(18). To this end, each tuning of $l_1, l_2, \dots, l_{2n-1}, l_{2n}$ will decrease $\mathcal{L}(\cdot)$, and Eq. (18) can derive numerical solutions of Eq. (1).

2.4. *Convergence analysis*

To show the convergence performance of the method, we vary the filter length 2n from 4 to 28 with an interval of 4 and plot the curve in terms of absolute residuals vs. iteration number in Fig. 1and also compare the convergence time in Table 2. As can be seen in Fig. 1, no matter how n changes, our proposed model will converge to zero in 150 iterations, and the convergence time is less than 1 second in most cases. However, when *n* becomes large (e.g. *n*=28), the convergence time may reach or exceed 1 second because of more pending parameters that need to be tuned. Overall, our proposed model runs fast and easy to convergence.

2.5. Decomposition effect

Table 2 Convergence time with various filter lengths of *n*.

Filter length	4	8	12	16	20	24	28
Time (s)	0.0039	0.0323	0.1052	0.2538	0.4044	0.5980	1.1511

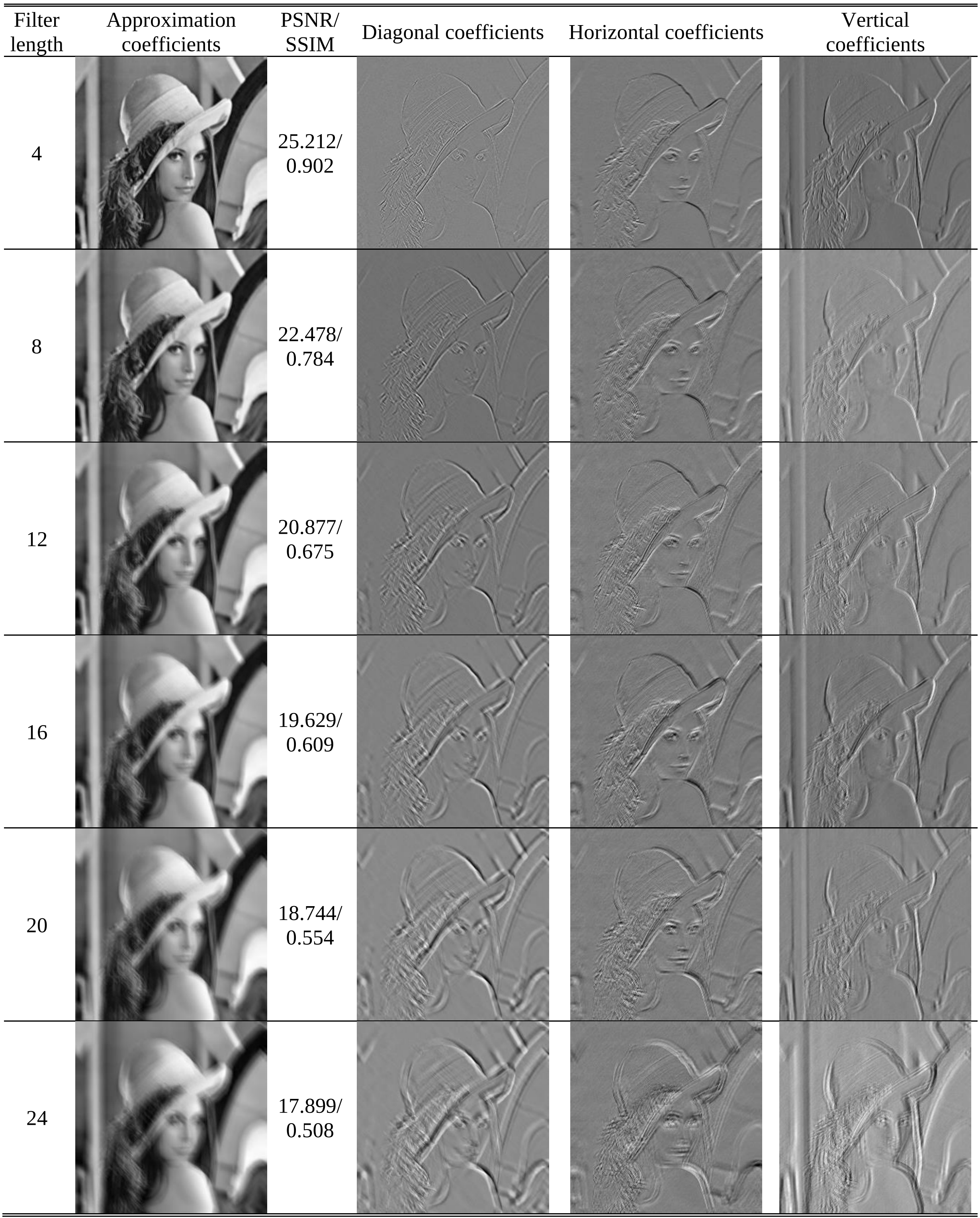

Fig. 2. *Decomposition results under different filter lengths.*

Fig. 3. *The samples in Kodak lossless true color image suite*

To show the overall decomposition effect of our proposed model, we randomly generate 100 sets of coefficients for different filters and compare the average decomposition results from these 100 runs in Fig. 2. As seen in Fig. 2, we can derive several key observations: 1) the approximation coefficient matrix contains the most cues of the image, and the other three coefficient matrices contain the details in different orientations; 2) Larger filter length tends to generate more blurred image from the approximation matrix where the corresponding PSNR and SSIM value are getting lower, though more detail will be preserved in other three coefficients; 3) The decomposition results have many potential applications, such as image compression, edge detection and feature extraction. For example, when the filter length is low, the main part contains most image information, thus the corresponding filter banks are suitable for image decomposition and compression. When the filter length is getting larger, the main part becomes over smoothed, thus these filters may be applicable to deal with the data with severe noise. In addition, the three detail coefficient matrices contain much orientation information that can be used to detect the edges and extract the shape features of the images for pattern recognition purpose [34, 35].

3. Results and discussion

In this section, we will discuss how the proposed generic D&R filtering banks work on two practical applications, i.e. colored image reconstruction and hyperspectral image classification. Six conventional and widely used wavelet families [36] are used for comparison, which include the Daubenchies (db) [18], Coiflets (coif) [25], Symlets (sym) [21], Fejer-Korovkin filters (FKfilt) [26], Biorthogonal (Bior) [20] and Reverse Biorthogonal (Rbior) [22]. Relevant experimental settings and results are presented as follows.

3.1. Color image reconstruction

Given a color image X, we apply discrete 2-D wavelet transform to each channel of X using the decomposition with a lowpass filter and a highpass filter to derive the approximation coefficients (cA), horizontal (cH), vertical (cV) and diagonal (cD) detail coefficients. We can then compute the reconstructed image X_r based on cA, cH, cV, cD and the reconstruction lowpass and highpass filters.

3.1.1. Experimental setup

For quantitative performance assessment, the maximum absolute error (MAE) and the root mean square error (RMSE) are employed as defined below.

$$MAE = \max_{1 \le i \le N}(|A_i - R_i|) \tag{19}$$

$$RMSE = \sqrt{\frac{1}{N} * \sum_{i=1}^{N}(A_i - R_i)^2} \tag{20}$$

where A_i and R_i denote the i^{th} pixel in reconstructed image and the original image, respectively. N represents the total number of pixels.

Kodak lossless true color image suite [1] is a publicly available database, which is also widely used to evaluate the performance of image processing algorithms. It contains 24 standard color images released by Kodak, and each image has 768*512 pixels. These images are used to test the efficacy of the proposed filtering bank in image reconstruction.

3.1.2. Experimental results

In this section, we applied the proposed method for image decomposition and reconstruction (D&R). Fig. 3 gives the test images from the Kodak lossless true color image suite. We use two criteria (i.e. MAE and RMSE) to evaluate the performance. Although the parameters in benchmarking wavelet families are predefined and fixed, our proposed filtering bank model can produce too many solutions which may affect the assessment. For a fair comparison, we generate

[1] http://r0k.us/graphics/kodak/

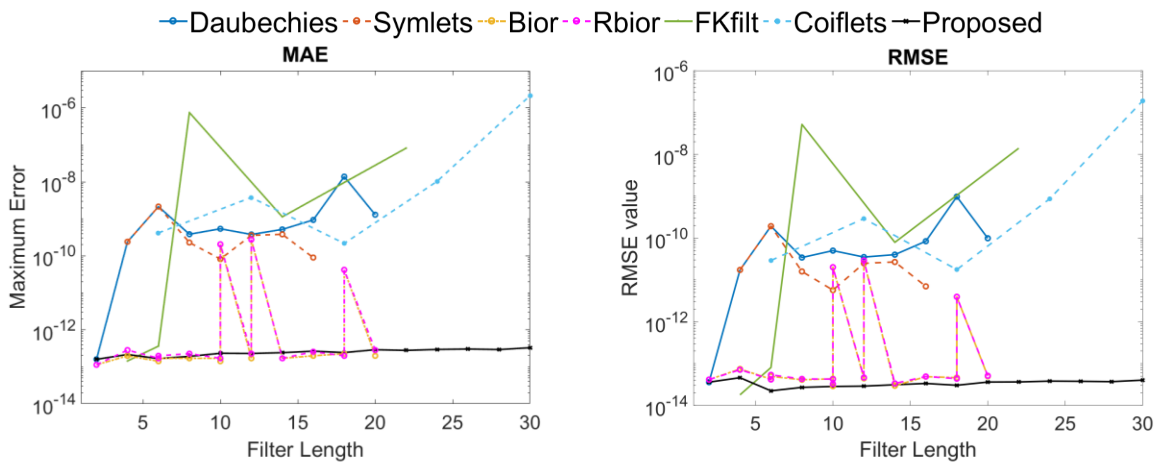

Fig. 4. *MAE and RMSE of proposed filter bank and other conventional filter banks*

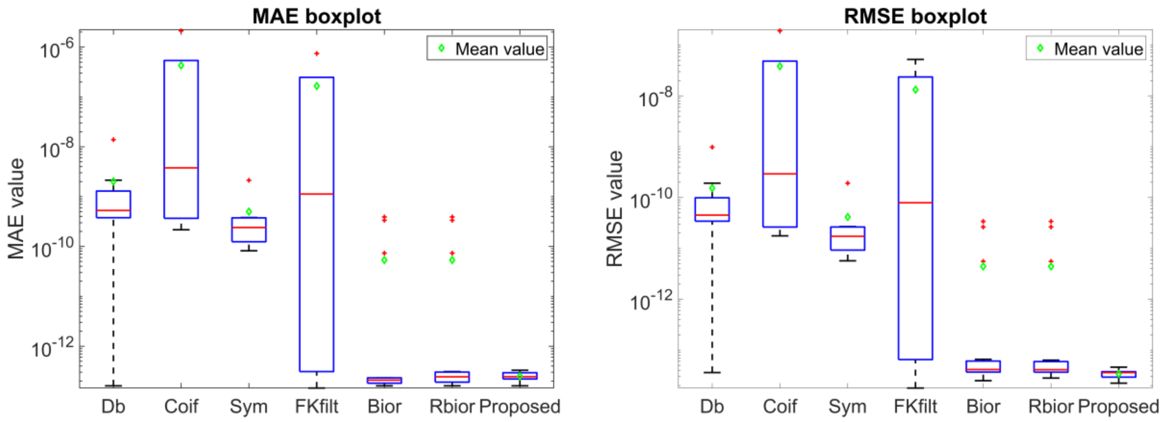

Fig. 5. *The boxplot of filtering bank families and proposed model*

solutions under different filter lengths randomly and select the best one from each filter length for comparison.

The averaged MAE and RMSE are copared in Fig. 4. As seen, the change of filter length n may affect significantly the performance of image D&R filters. However, our proposed filter bank can always yield consistently the best performance no matter how n changes. For Bior and Rbior filtering bank families, they have almost the same characteristic where each of them has some filters with similar filter length but different values. They may produce more comparable D&R results in comparison to ours when the length of their filters is no more than 10. When the predefined value of some filters is not appropriate, especially when n=10, 12, or 18, the D&R performance becomes much worse. For FKflit, its performance is slightly better than the proposed model when the filter length is 4, but it is not robust as its MAE and RMSE measures are much higher when the filter length increases. For the rest of filter banks including Daubenchies, Coiflets and Symlets, it seems that the filter length has a significant effect on their performance, where the overall performance of them is not as good as the proposed model. In addition, as the MAE and RMSE are quite close to zero, the results of visual reconstruction from different filtering banks are very similar. Therefore, the visual comparison is not used in the section.

Fig. 4 summarizes the detailed performance of different filter lengths in each wavelet family, and Fig. 5 uses two boxplots to show how our proposed model is better than other wavelet families. In Fig. 5, red cross and green rhombus indicate the outliers and the mean value, respectively. The top border and bottom border of the blue box represents the 25 and 75 percentile respectively, the red line in the blue box represents the median value, and the two black lines outside the blue box represent the maximum and minimum value. It can be seen that our proposed model is the best in terms of the least RMSE and MAE measurements. In RMSE boxplot, 1) the gap between 25 and 75 percentiles is the smallest, 2) the gap between maximum and minimum value the smallest, 3) there is no outliers, 4) the median value (50 percentile) and the mean value are very close to each other. In MAE boxplot, although the gap between 25 and 75 percentiles of the proposed model is slightly larger than that of Bior, the rest three observations as shown in the RMSE boxplot still hold true. In general, our proposed model can generate consistently satisfied filters with different lengths for D&R problems. Instead of trying different filters from other wavelet families, our model constantly gives the most reliable and robust results.

3.2. Hyperspectral image classification

Given a hyperspectral image (HSI) $X \in \mathbb{R}^{m*n*b}$, where m and n are spatial dimensions of X , and b is the number of spectral bands. Each pixel in X is a b-dimensional vector, and we apply discrete 1-D wavelet transform to each pixel to extract the approximation coefficients (cA) and detail coefficients (cD). We then employ cA as the spectral feature, i.e. the main

Table 3 Classification results for the 92AV3C dataset using DWT with the aubechies filter bank.

Daubechies	'db2'	'db3'	'db4'	'db5'	'db6'	**'db7'**	*'db8'*	'db9'	'db10'	Average
filter length	4	6	8	10	12	**14**	*16*	18	20	
AA	76.52	76.40	75.77	75.14	75.26	**75.45**	*75.87*	75.97	74.49	75.82
OA	80.98	81.18	80.93	81.01	80.98	**81.47**	*81.23*	80.73	81.01	81.13
Kappa	78.30	78.49	78.24	78.29	78.26	**78.81**	*78.55*	77.96	78.30	78.43

Table 4 Classification results for the 92AV3C dataset using DWT with the Symlets filter bank.

Symlets	'sym2'	'sym3'	'sym4'	'sym5'	**'sym6'**	'sym7'	'sym8'	Average
filter length	4	6	8	10	**12**	14	16	
AA	75.37	*75.01*	75.74	75.20	**75.56**	74.85	76.32	75.44
OA	81.06	*81.23*	81.01	81.18	**81.41**	80.83	80.72	81.06
Kappa	78.37	*78.57*	78.29	78.50	**78.76**	78.10	77.99	78.37

Table 5 Classification results for the 92AV3C dataset using DWT with the FKfilt filter bank and Coiflets filter bank.

Fkfilt	**'fk4'**	*'fk6'*	'fk8'	'fk14'	'fk22'	Average	Coiflets	**'coif1'**	'coif2'	'coif3'	'coif4'	*'coif5'*	Average
length	**4**	*6*	8	14	22		**length**	**6**	12	18	24	*30*	
AA	**76.14**	*77.01*	75.58	75.80	76.34	76.17	**AA**	**76.50**	75.78	74.48	75.66	*76.58*	75.80
OA	**81.72**	*81.49*	81.25	81.11	81.07	81.33	**OA**	**81.37**	81.01	80.98	81.04	*81.02*	81.08
Kappa	**79.10**	*78.86*	78.57	78.42	78.38	78.67	**Kappa**	**78.70**	78.29	78.26	78.35	*78.32*	78.38

components, since cD is mostly considered as noise. For data classification, the support vector machines (SVM) have been widely used in HSI [3, 37]. In addition, there are many publicly available software tools supporting multiple functions of the SVM, which make it easy to implement. As a result, SVM is selected as a classifier, and LIBSVM [38] is used in our experiments for implementation.

3.2.1. Experimental setup

To assess the performance of HSI classification, three criteria including the overall accuracy (OA), the average accuracy (AA), and Kappa coefficient (KP) are employed. The OA is the percentage of correctly classified pixels, and AA is the mean value of the classification accuracies for all the classes. They are defined as follows:

$$OA = \frac{1}{n} * \sum_{i}^{T} C_i * 100 \tag{21}$$

$$AA = \frac{1}{T} * \sum_{i}^{T} \frac{C_i}{N_i} * 100 \tag{22}$$

where N_i is the all the pixels in class i, C is the classification confusion matrix, C_{ii} is the diagonal element in the confusion matrix which is also the correctly classified pixel in class i, C_{ij} is the row element and C_{ji} is the column element in the confusion matrix. T is the total number of classes and n is the total number of pixels for classification.

KP, defined below, is to measure inter-rater reliability:

$$KP = \frac{P_0 - P_e}{1 - P_e}$$
$$P_0 = OA, P_e = \frac{1}{n^2} \sum_{i}^{T} \left(\sum_{j}^{T} C_{ij} * \sum_{j}^{T} C_{ji} \right) \tag{23}$$

A popularly used HSI data sets, Indian Pines, is employed in our experiments to quantitatively evaluate the performance of the proposed model. Indian Pines, also namely 92AV3C, contains 145*145 pixels in 220 spectral bands for land mapping. It was collected by using the Airborne visible/infrared imaging spectrometer (AVIRIS) [39] in 1992 in the USA, and the whole dataset was labelled in 16 land-cover classes. After removal of bands that are severely affected by water absorption and noise [6], only 200 bands remain for classification [40-42].

3.2.2. Experimental results

In this section, the classification results on the 92AV3C dataset with different filter banks are compared in Tables 3-9, and the training rate is 10%. All the experiments are repeated 10 times, and the average results are reported for comparison. Tables 3-7 show how the change of filter lengths in different filter banks affect the classification accuracy, where the best and the second-best filters are marked in bold and italic, respectively. For most filter banks, different filter lengths seem to have wispy gaps on the classification results. In Table 8, we compare the selected filters with the best performance in each filter bank and also summarize the average performance of each filter bank in Table 9. As seen in Table 8, our proposed method with a filter length 46 gives the best result over all others. Meanwhile, larger filter length seems to perform better than shorter ones in HSI classification until the results saturate with one possible best filter length before the performance degrades. There are possibly two main reasons for it: 1) HSI data usually contains much noise, 2) our proposed model can more effectively smooth the noisy data, especially with a relatively large filter length. As seen in Table 9, our proposed filter bank almost outperforms all others, though it is quite close to those of the Rbior filter bank. Although the average OA and Kappa of our filter bank are slightly lower than that of Rbior, the average AA of ours is 2% higher than Rbior, which means our model can produce high classification accuracy to both large and small classes. This is because the proposed method is a

Table 6 Classification results for the 92AV3C dataset using wavelet transform with the Bior filter bank.

Bior	*'bior1.3'*	'bior1.5'	'bior2.2'	'bior2.4'	'bior2.6'	'bior2.8'	'bior3.1'	
filter length	*6*	10	6	10	14	18	4	
AA	*75.80*	74.68	73.82	73.57	73.96	72.26	72.25	
OA	*81.42*	80.58	79.97	80.30	80.48	79.87	79.17	
Kappa	*78.76*	77.83	77.10	77.45	77.70	76.95	76.18	
	'bior3.3'	'bior3.5'	'bior3.7'	'bior3.9'	'bior4.4'	**'bior5.5'**	'bior6.8'	**Average**
filter length	8	12	16	20	10	**12**	18	
AA	70.72	72.16	72.65	71.50	75.57	**76.16**	76.27	73.91
OA	79.68	79.37	79.67	79.55	80.84	**81.45**	81.17	80.34
Kappa	76.74	76.41	76.79	76.65	78.13	**78.80**	78.47	77.53

Table 7 Classification results for the 92AV3C dataset using wavelet transform with the Rbior filter bank.

Rbior	'rbio1.3'	'rbio1.5'	'rbio2.2'	'rbio2.4'	'rbio2.6'	'rbio2.8'	*'rbio3.1'*	
filter length	6	10	6	10	14	18	*4*	
AA	77.62	76.10	75.39	77.18	77.46	76.48	*76.48*	
OA	81.64	81.17	81.26	81.76	81.34	81.80	*82.01*	
Kappa	79.05	78.51	78.56	79.16	78.68	79.20	*79.45*	
	'rbio3.3'	'rbio3.5'	**'rbio3.7'**	'rbio3.9'	'rbio4.4'	'rbio5.5'	'rbio6.8'	**Average**
filter length	8	12	**16**	20	10	12	18	
AA	77.17	77.01	**76.07**	77.27	74.01	74.27	75.86	76.40
OA	81.85	81.77	**82.13**	81.77	81.06	80.31	80.86	81.48
Kappa	79.27	79.18	**79.56**	79.17	78.37	77.51	78.14	78.84

Table 8 The classification results of the best filters selected from different filter banks.

Methods	'db7'	'coif1'	'sym6'	'fk4'	'bior5.5'	*'rbio3.7'*	Proposed 46
AA	75.45	76.5	75.56	76.14	76.16	*76.07*	**81.78**
OA	81.47	81.37	81.41	81.72	81.45	*82.13*	**83.25**
Kappa	78.81	78.7	78.76	79.1	78.8	*79.56*	**80.84**

Table 9 The average classification results from different filter banks.

Methods	Daubechies	Coiflets	Symlets	FKfilt	Bior	*Rbior*	**Proposed** 46
AA	75.82	75.80	75.44	76.17	73.91	*76.40*	**78.31**
OA	81.13	81.08	81.06	81.33	80.34	**81.48**	*81.47*
Kappa	78.43	78.38	78.37	78.67	77.53	**78.84**	*78.69*

generic model in which the parameters are randomly initialized and optimized. As a result, it is more universal than those wavelet filtering banks with fixed parameters.

Visual comparison of these classification results from selected filters within the compared filter banks is shown in Fig. 6, where the the average accuracy (AA) achieved is also given. As seen, the proposed wavelet filtering bank has the highest AA, which indicates that the proposed method can preserve more details in classification of small classes. This has demonstrated the superior performance of the generic model than conventional ones with fixed parameters. In addition, the region of each class is found smoother and more consistent with the ground-truth, also the outliers of each class are much less than others (e.g. Soybeans-clean till, Alfalfa). In summary, our proposed model outperforms other wavelet families in HSI classification.

4. Conclusions

Wavelet transform is a classic and popular method for data coding, compression and feature extraction. Conventional wavelet filter banks have fixed parameters, and the performance depends on the data being processed. As a result, searching for a suitable wavelet filter for a specific task is quite time-consuming. With the extended types of data being captured from smart camera systems, it becomes essential to find more suitable wavelet filter banks for such applications.

In this paper, a generic framework is proposed to build a series of decomposition and reconstruction filter banks for discrete data sequence. Only one parameter n needs to determine, which is the half length of the filter. Then $2n$ coefficients of certain filter can be calculated by n+1

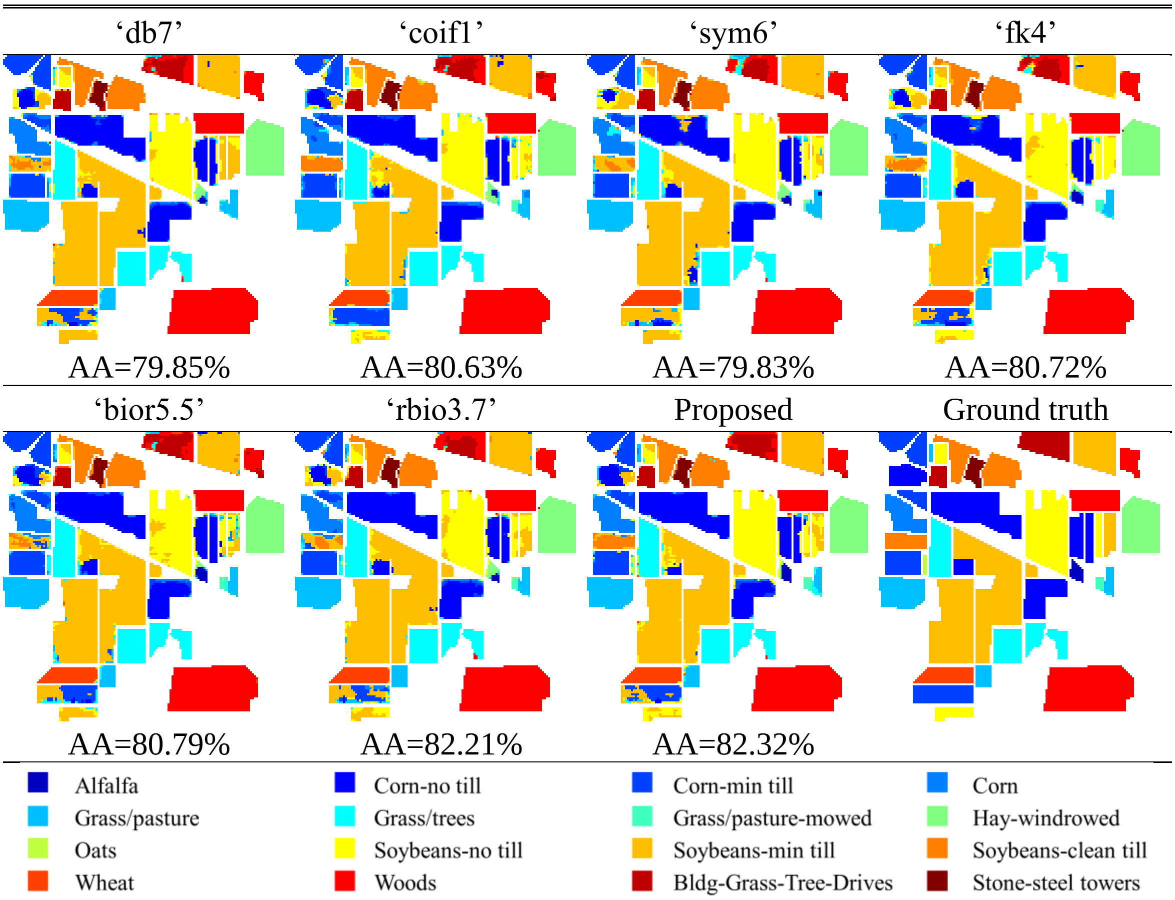

Fig. 6. *Visual comparison of selected filters from each filtering bank.*

equations in the proposed model. All essential mathematical formulae are carried out to prove their validity. To evaluate the efficacy of the proposed model, we first measure its convergence performance and then validate its practical values by applying it in two applications including image D&R and classification of hyperspectral images. The proposed generic D&R framework actually provides a unified fundamental for signal processing. The most important finding in this paper is that most conventional filter banks can be the special solutions of our proposed generic model, where only one parameter, the filter length, needs to be determined in our model. The proposed model has many potentials of extendable capabilities. Comprehensive experiments have demonstrated the fast convergence speed of the proposed model. Also its superior performance on color image D&R and hyperspectral image classification over conventional wavelets have shown its promising potentials in real image processing applications.

For future work, we will focus on applying this framework for many other applications such as image restoration, denoising and data compression, et al. For those specific applications, some special constraints can be imposed on the model to make the decomposition more effective and purpose targeted. Some other optimization methods such as gravitational search algorithm [43] and stochastic gradient descent can also be considered to further improve the convergence speed.

Appendix: Detailed deduction of Eq.(1)-(18)

$$\begin{cases} l_1 l_{2n-1} + l_2 l_{2n} = 0 & 1^{st}\text{ condition} \\ l_1 l_{2n-3} + l_2 l_{2n-2} + l_3 l_{2n-1} + l_4 l_{2n} = 0 & 2^{nd}\text{ condition} \\ \vdots & \vdots \\ l_1 l_{2n-(2n-5)} + l_2 l_{2n-(2n-6)} + \cdots + l_{2n-5} l_{2n-1} + l_{2n-4} l_{2n} = 0 & (n-2)^{th}\text{ condition} \\ l_1 l_{2n-(2n-3)} + l_2 l_{2n-(2n-4)} + \cdots + l_{2n-3} l_{2n-1} + l_{2n-2} l_{2n} = 0 & (n-1)^{th}\text{ condition} \\ \sum_{i=1,3,\ldots,2n-1} l_i = \sum_{i=2,3,\ldots,2n} l_i & n^{th}\text{ condition} \\ s.t.\ l_1^2 + l_2^2 + \cdots + l_{2n-1}^2 + l_{2n}^2 = 1 & 1^{st}\text{ contraint} \end{cases} \quad (1)$$

$$S_e = [s_{2n-1}, s_{2n-2}, \ldots, s_2, s_1, s_1, s_2, \ldots, s_{m-1}, s_m] \quad (2)$$

$$\begin{aligned} p_{n+k} &= s_{2k+1} l_{2n} + s_{2k+2} l_{2n-1} + \cdots + s_{2(k+n)-1} l_2 + s_{2(k+n)} l_1 \\ q_{n+k} &= s_{2k+1} l_1 - s_{2k+2} l_2 + \cdots + s_{2(k+n)-1} l_{2n-1} - s_{2(k+n)} l_{2n} \end{aligned} \quad (3)$$

$$S=[0,p_1,0,p_2,\dots,0,p_n]*L_r+[0,q_1,0,q_2,\dots,0,q_n]*H_r \tag{4}$$

$$\begin{aligned} s_1 &= [0,p_1,0,p_2,\dots,0,p_n][l_1,l_2,\dots,l_{2n-1},l_{2n}]^T \\ &+[0,q_1,0,q_2,\dots,0,q_n][-l_{2n},l_{2n-1},\dots,-l_2,l_1]^T \\ &= [q_n,p_1,q_{n-1},p_2,\dots,q_2,p_{n-1},q_1,p_n][l_1,l_2,\dots,l_{2n-1},l_{2n}]^T \\ &= [l_1,l_2,\dots,l_{2n-1},l_{2n}][q_n,p_1,q_{n-1},p_2,\dots,q_2,p_{n-1},q_1,p_n]^T \end{aligned} \tag{5}$$

$$s_1=[l_1,l_2,\dots,l_{2n-1},l_{2n}]M[s_1,s_2,\dots,s_{2n-1},s_{2n}]^T \tag{6}$$

$$M\triangleq\begin{bmatrix} l_1 & -l_2 & l_3 & -l_4 & \cdots & -l_{2n-2} & l_{2n-1} & -l_{2n} \\ l_3+l_2 & l_1+l_4 & l_5 & \cdots & l_{2n-1} & l_{2n} & 0 & 0 \\ l_3-l_2 & l_1-l_4 & l_5 & \cdots & l_{2n-1} & -l_{2n} & 0 & 0 \\ l_4+l_5 & l_3+l_6 & \cdots & l_{2n} & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ l_{2n-1}-l_{2n-2} & l_{2n-3}-l_{2n} & l_{2n-4} & \cdots & -l_2 & l_1 & 0 & 0 \\ l_{2n} & l_{2n-1} & l_{2n-2} & \cdots & l_4 & l_3 & l_2 & l_1 \end{bmatrix} \tag{7}$$

$$[1,0,\dots,0,0]=[l_1,l_2,\dots,l_{2n-1},l_{2n}]M \tag{8}$$

$$\begin{aligned} 1 &= [l_1,l_2,\dots,l_{2n-1},l_{2n}][l_1,l_3+l_2,l_3-l_2,\dots,l_{2n-1}-l_{2n-2},l_{2n}]^T \\ &= l_1^2+l_2^2+l_3^2+\cdots+l_{2n-1}^2+l_{2n}^2=\sum_{j=1}^{2n} l_j^2 \\ 0 &= [l_1,l_2,\dots,l_{2n-1},l_{2n}][-l_2,l_1+l_4,l_1-l_4,\dots,l_{2n-3}-l_{2n},l_{2n-1}]^T \\ &= -l_1l_2+l_2(l_1+l_4)+l_3(l_1-l_4)+\cdots+l_{2n-1}(l_{2n-3}-l_{2n})+l_{2n}l_{2n-1} \\ &= l_1l_3+l_2l_4+l_3l_5+\cdots+l_{2n-3}l_{2n-1}+l_{2n-2}l_{2n} \\ &\qquad\qquad\cdots\cdots \\ 0 &= [l_1,l_2,\dots,l_{2n-1},l_{2n}][l_{2n-1},0,0,\dots,0,l_2]^T=l_1l_{2n-1}+l_{2n}l_2 \\ 0 &= [l_1,l_2,\dots,l_{2n-1},l_{2n}][-l_{2n},0,0,\dots,0,l_1]^T=-l_1l_{2n}+l_{2n}l_1 \end{aligned} \tag{9}$$

$$\begin{aligned} s_{2k-1} &= [l_1,l_2,\dots,l_{2n-1},l_{2n}][q_{n+k-1},p_k,q_{n+k-2},p_{k+1},\dots,q_{k+1},p_{n+k},q_k,p_{n+k-1}]^T \\ s_{2k} &= [l_1,l_2,\dots,l_{2n-1},l_{2n}][p_k,-q_{n+k-1},p_{k+1},-q_{n+k-2},\dots,p_{n+k},-q_{k+1},p_{n+k-1},-q_k]^T \end{aligned} \tag{10}$$

$$\begin{aligned} \mathcal{L}(l_1,\dots,l_{2n}) &= (l_1l_{2n-1}+l_2l_{2n})^2+(l_1l_{2n-3}+l_2l_{2n-2}+l_3l_{2n-1}+l_4l_{2n})^2+\cdots \\ &+\left(l_1l_{2n-(2n-3)}+l_2l_{2n-(2n-4)}+\cdots+l_{2n-3}l_{2n-1}+l_{2n-2}l_{2n}\right)^2 \\ &+\left(\sum\nolimits_{i=1,3,\dots,2n-1} l_i-\sum\nolimits_{i=2,3,\dots,2n} l_i\right)^2 \end{aligned} \tag{11}$$

$$l_j=\underset{l_j}{\operatorname{argmin}}\,\mathcal{L}(l_1,l_2,\dots,l_{2n-1},l_{2n})\,, where\ j\in[1,2n], s.t. \sum_{j=1}^{2n} l_j^2=1 \tag{12}$$

$$\frac{\partial\mathcal{L}(l_1,l_2,\dots,l_{2n-1},l_{2n})}{\partial l_j}=0 \tag{13}$$

$$\sum_{i=1}^{n}\frac{\partial C_i^2}{\partial l_j}=0 \tag{14}$$

$$\begin{gathered} \frac{\partial(l_1l_{2n-1}+l_2l_{2n})^2}{\partial l_j}=0, \\ \frac{\partial(l_1l_{2n-3}+l_2l_{2n-2}+l_3l_{2n-1}+l_4l_{2n})^2}{\partial l_j}=0, \\ \cdots\cdots \\ \frac{\partial\left(l_1l_{2n-(2n-3)}+l_2l_{2n-(2n-4)}+\cdots+l_{2n-3}l_{2n-1}+l_{2n-2}l_{2n}\right)^2}{\partial l_j}=0, \\ \frac{\partial\left(\sum_{i=1,3,\dots,2n-1} l_i-\sum_{i=2,3,\dots,2n} l_i\right)^2}{\partial l_j}=0, \end{gathered} \tag{15}$$

$$l_j*\left(\left(l_{j-2(n-1)}+l_{j+2(n-1)}\right)^2+\left(l_{j-2(n-2)}+l_{j+2(n-2)}\right)^2+\cdots+\left(l_{j-2(n-k)}+l_{j+2(n-k)}\right)^2+1^2\right)\equiv \tag{16}$$

$$-\left(l_1 l_{2n-1} + l_2 l_{2n} - l_{j-2(n-1)} - l_{j+2(n-1)}\right) * \left(l_{j-2(n-1)} + l_{j+2(n-1)}\right) - \left(l_1 l_{2n-3} + l_2 l_{2n-2} + l_3 l_{2n-1} + l_4 l_{2n} - l_{j-2(n-2)} - l_{j+2(n-2)}\right) * \left(l_{j-2(n-2)} + l_{j+2(n-2)}\right) - \cdots - \left(l_1 l_{2n-(2n-3)} + l_2 l_{2n-(2n-4)} + \cdots + l_{2n-3} l_{2n-1} + l_{2n-2} l_{2n} - l_{j-2(n-k)} - l_{j+2(n-k)}\right) * \left(l_{j-2(n-k)} + l_{j+2(n-k)}\right) + \left(\sum_{i=1,3,\ldots,2n-1} l_i - \sum_{i=2,3,\ldots,2n} l_i - l_j\right) * 1,$$

$$\begin{aligned}
[\alpha_1, \alpha_2, \ldots, \alpha_n] &= [-\left(l_1 l_{2n-1} + l_2 l_{2n} - l_{j-2(n-1)} - l_{j+2(n-1)}\right), \\
&\quad -\left(l_1 l_{2n-3} + l_2 l_{2n-2} + l_3 l_{2n-1} + l_4 l_{2n} - l_{j-2(n-2)} - l_{j+2(n-2)}\right), \ldots, \\
&\quad -\left(l_1 l_{2n-(2n-3)} + l_2 l_{2n-(2n-4)} + \cdots + l_{2n-3} l_{2n-1} + l_{2n-2} l_{2n} - l_{j-2(n-k)} - l_{j+2(n-k)}\right), \\
&\quad \left(\sum_{i=1,3,\ldots,2n-1} l_i - \sum_{i=2,3,\ldots,2n} l_i - l_j\right)] \\
[\beta_1, \beta_2, \ldots, \beta_n] &= \left[\left(l_{j-2(n-1)} + l_{j+2(n-1)}\right), \left(l_{j-2(n-2)} + l_{j+2(n-2)}\right), \ldots, \left(l_{j-2(n-k)} + l_{j+2(n-k)}\right), 1\right]
\end{aligned} \tag{17}$$

$$\begin{gathered}
[\beta_1, \beta_2, \ldots, \beta_n][\beta_1, \beta_2, \ldots, \beta_n]^T l_j = [\alpha_1, \alpha_2, \ldots, \alpha_n][\beta_1, \beta_2, \ldots, \beta_n]^T \\
l_j = \underset{l_j}{\operatorname{argmin}} \mathcal{L}(l_1, l_2, \ldots, l_{2n-1}, l_{2n}) = \underset{l_j}{\operatorname{argmin}} \left\|\beta l_j - \alpha\right\|, \qquad \text{s.t.} \sum_{j=1}^{2n} l_j^2 = 1
\end{gathered} \tag{18}$$

References

[1]Szeliski, R., *Computer Vision: Algorithms and Applications*, (Springer Science & Business Media, 2010)

[2]Candes, E. and Romberg, J., 'Sparsity and Incoherence in Compressive Sampling', *Inverse problems*, 2007, 23, (3), p. 969.

[3]Qiao, T., Ren, J., Wang, Z., et al., 'Effective Denoising and Classification of Hyperspectral Images Using Curvelet Transform and Singular Spectrum Analysis', *IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING*, 2017, 55, (1), pp. 119-133.

[4]Katznelson, Y., *An Introduction to Harmonic Analysis*, (Cambridge University Press, 2004)

[5]Wright, J., Ganesh, A., Rao, S., Peng, Y., and Ma, Y., 'Robust Principal Component Analysis: Exact Recovery of Corrupted Low-Rank Matrices Via Convex Optimization', in, *Advances in neural information processing systems*, (2009)

[6]Zabalza, J., Ren, J., Zheng, J., et al., 'Novel Two-Dimensional Singular Spectrum Analysis for Effective Feature Extraction and Data Classification in Hyperspectral Imaging', *IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING*, 2015, 53, (8), pp. 4418-4433.

[7]Kao, L.-J., Chiu, C.-C., Lu, C.-J., and Yang, J.-L., 'Integration of Nonlinear Independent Component Analysis and Support Vector Regression for Stock Price Forecasting', *Neurocomputing*, 2013, 99, pp. 534-542.

[8]Mercuri, M., Liu, Y.-H., Lorato, I., et al., 'A Direct Phase-Tracking Doppler Radar Using Wavelet Independent Component Analysis for Non-Contact Respiratory and Heart Rate Monitoring', *IEEE transactions on biomedical circuits and systems*, 2018, 12, (3), pp. 632-643.

[9]Liu, X., Zhao, G., Yao, J., and Qi, C., 'Background Subtraction Based on Low-Rank and Structured Sparse Decomposition', *IEEE TRANSACTIONS ON IMAGE PROCESSING*, 2015, 24, (8), pp. 2502-2514.

[10]Fadlallah, Y., Aïssa-El-Bey, A., Amis, K., Pastor, D., and Pyndiah, R., 'New Iterative Detector of Mimo Transmission Using Sparse Decomposition', *IEEE Trans. Vehicular Technology*, 2015, 64, (8), pp. 3458-3464.

[11]Belabbas, M.-A. and Wolfe, P.J., 'Spectral Methods in Machine Learning and New Strategies for Very Large Datasets', *Proceedings of the National Academy of Sciences*, 2009, p. pnas. 0810600105.

[12]Kannan, R. and Vempala, S., 'Spectral Algorithms', *Foundations and Trends® in Theoretical Computer Science*, 2009, 4, (3–4), pp. 157-288.

[13]Mahoney, M.W. and Drineas, P., 'Cur Matrix Decompositions for Improved Data Analysis', *Proceedings of the National Academy of Sciences*, 2009, p. pnas. 0803205106.

[14]Liu, Y., Lei, Y., Li, C., Xu, W., and Pu, Y., 'A Random Algorithm for Low-Rank Decomposition of Large-Scale Matrices with Missing Entries', *IEEE Trans. Image Processing*, 2015, 24, (11), pp. 4502-4511.

[15]Jain, P., Netrapalli, P., and Sanghavi, S., 'Low-Rank Matrix Completion Using Alternating Minimization', in, *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, (ACM, 2013)

[16]Liu, Y., Yang, S., Wu, P., Li, C., and Yang, M., '$ L_ {1} $-Norm Low-Rank Matrix Decomposition by Neural Networks and Mollifiers', *IEEE transactions on neural networks and learning systems*, 2016, 27, (2), pp. 273-283.

[17]Mallat, S.G., 'A Theory for Multiresolution Signal Decomposition: The Wavelet Representation', *IEEE Trans Pattern Anal Mach Intell*, 1989, 11, (7), pp. 674-693.

[18]Campo, D., Quintero, O., and Bastidas, M., 'Multiresolution Analysis (Discrete Wavelet Transform) through Daubechies Family for Emotion Recognition in Speech', in, *Journal of Physics: Conference Series*, (IOP Publishing, 2016)

[19]Wang, Y., Ding, X., Wang, R., Zhang, J., and Fu, X., 'Fusion-Based Underwater Image Enhancement by Wavelet Decomposition', in, *2017 IEEE International Conference on Industrial Technology (ICIT)*, (IEEE, 2017)

[20]Abhishek, S., Veni, S., and Narayanankutty, K., 'Biorthogonal Wavelet Filters for Compressed Sensing Ecg Reconstruction', *Biomedical Signal Processing and Control*, 2019, 47, pp. 183-195.

[21]Yadav, A.K., Roy, R., Kumar, A.P., Kumar, C.S., and Dhakad, S.K., 'De-Noising of Ultrasound Image Using Discrete Wavelet Transform by Symlet Wavelet and Filters', in, *2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, (IEEE, 2015)

[22]Szewczyk, R., Grabowski, K., Napieralska, M., et al., 'A Reliable Iris Recognition Algorithm Based on Reverse Biorthogonal Wavelet Transform', *Pattern Recognition Letters*, 2012, 33, (8), pp. 1019-1026.
[23]Tang, Y.Y., Lu, Y., and Yuan, H., 'Hyperspectral Image Classification Based on Three-Dimensional Scattering Wavelet Transform', *IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING*, 2015, 53, (5), pp. 2467-2480.
[24]Cao, X., Xu, L., Meng, D., Zhao, Q., and Xu, Z., 'Integration of 3-Dimensional Discrete Wavelet Transform and Markov Random Field for Hyperspectral Image Classification', *Neurocomputing*, 2017, 226, pp. 90-100.
[25]Narkhedkar, S.G. and Patel, P.K., 'Recipe of Speech Compression Using Coiflet Wavelet', in, *Proceedings of 2014 International Conference on Contemporary Computing and Informatics, IC3I 2014*, (2014)
[26]Rodriguez, N. and Barba, L., 'Fejer-Korovkin Wavelet Based Mimo Model for Multi-Step-Ahead Forecasting of Monthly Fishes Catches', *Polibits*, 2017, 56, pp. 71-76.
[27]Jiajun, H., Chuanjin, Y., Yongle, L., and Huoyue, X., 'Ultra-Short Term Wind Prediction with Wavelet Transform, Deep Belief Network and Ensemble Learning', *Energy Conversion and Management*, 2020, 205, p. 112418.
[28]Kanarachos, S., Christopoulos, S.R.G., Chroneos, A., and Fitzpatrick, M.E., 'Detecting Anomalies in Time Series Data Via a Deep Learning Algorithm Combining Wavelets, Neural Networks and Hilbert Transform', *Expert Systems with Applications*, 2017, 85, pp. 292-304.
[29]Yang, C., Zhao, Y., and Wang, S., 'Deep Image Compression in the Wavelet Transform Domain Based on High Frequency Sub-Band Prediction', *IEEE Access*, 2019, 7, pp. 52484-52497.
[30]Luo, F., Zhang, L., Zhou, X., et al., 'Sparse-Adaptive Hypergraph Discriminant Analysis for Hyperspectral Image Classification', *IEEE GEOSCIENCE AND REMOTE SENSING LETTERS*, 2019.
[31]Luo, F., Huang, H., Duan, Y., Liu, J., and Liao, Y., 'Local Geometric Structure Feature for Dimensionality Reduction of Hyperspectral Imagery', *Remote Sensing*, 2017, 9, (8), p. 790.
[32]Liu, Y., 'A Unifying Decomposition and Reconstruction Model for Discrete Signals', *arXiv preprint arXiv:1804.01227*, 2018.
[33]Press, W.H., Teukolsky, S.A., Vetterling, W.T., and Flannery, B.P., *Numerical Recipes 3rd Edition: The Art of Scientific Computing*, (Cambridge university press, 2007)
[34]Qayyum, H., Majid, M., Anwar, S.M., and Khan, B., 'Facial Expression Recognition Using Stationary Wavelet Transform Features', *Mathematical Problems in Engineering*, 2017, 2017.
[35]Maheshwari, S., Pachori, R.B., and Acharya, U.R., 'Automated Diagnosis of Glaucoma Using Empirical Wavelet Transform and Correntropy Features Extracted from Fundus Images', *IEEE journal of biomedical and health informatics*, 2016, 21, (3), pp. 803-813.
[36]Daubechies, I., *Ten Lectures on Wavelets*, (Siam, 1992)
[37]Zabalza, J., Qing, C., Yuen, P., et al., 'Fast Implementation of Two-Dimensional Singular Spectrum Analysis for Effective Data Classification in Hyperspectral Imaging', *Journal of the Franklin Institute*, 2018, 355, (4), pp. 1733-1751.
[38]Chang, C.-C. and Lin, C.-J., 'Libsvm: A Library for Support Vector Machines', *ACM transactions on intelligent systems and technology (TIST)*, 2011, 2, (3), p. 27.
[39]Green, R.O., Eastwood, M.L., Sarture, C.M., et al., 'Imaging Spectroscopy and the Airborne Visible/Infrared Imaging Spectrometer (Aviris)', *Remote sensing of environment*, 1998, 65, (3), pp. 227-248.
[40]Archibald, R. and Fann, G., 'Feature Selection and Classification of Hyperspectral Images with Support Vector Machines', *IEEE GEOSCIENCE AND REMOTE SENSING LETTERS*, 2007, 4, (4), pp. 674-677.
[41]Luo, F., Du, B., Zhang, L., Zhang, L., and Tao, D., 'Feature Learning Using Spatial-Spectral Hypergraph Discriminant Analysis for Hyperspectral Image', *IEEE transactions on cybernetics*, 2018, 49, (7), pp. 2406-2419.
[42]Luo, F., Zhang, L., Du, B., and Zhang, L., 'Dimensionality Reduction with Enhanced Hybrid-Graph Discriminant Learning for Hyperspectral Image Classification', *IEEE Transactions on Geoscience and Remote Sensing*, 2020.
[43]Sun, G., Ma, P., Ren, J., Zhang, A., and Jia, X., 'A Stability Constrained Adaptive Alpha for Gravitational Search Algorithm', *Knowledge-Based Systems*, 2018, 139, pp. 200-213.