TUNA, S., TÖREYIN, B.U., REN, J., ZHAO, H. and MARSHALL, S. 2021. Iterative enhanced multivariance products representation for effective compression of hyperspectral images. *IEEE transactions on geoscience and remote sensing* [online], 59(11), pages 9569-9584. Available from: https://doi.org/10.1109/TGRS.2020.3031016

Iterative enhanced multivariance products representation for effective compression of hyperspectral images.

TUNA, S., TÖREYIN, B.U., REN, J., ZHAO, H. and MARSHALL, S.

2021

Iterative Enhanced Multivariance Products Representation for Effective Compression of Hyperspectral Images

Süha TUNA, Behçet Uğur TÖREYİN, *Member, IEEE*, Metin DEMİRALP, Jinchang REN, *Senior Member, IEEE*, Stephen MARSHALL

***Abstract*—Effective compression of hyperspectral images is essential due to their large data volume. Since these images are high dimensional, processing them is also another challenging issue. In this work, an efficient lossy hyperspectral image compression method based on Enhanced Multivariance Products Representation (EMPR) is proposed. As an efficient data decomposition method, EMPR enables us to represent the given multidimensional data with lower dimensional entities. EMPR, as a finite expansion with relevant approximations, can be acquired by truncating this expansion at certain levels. Thus, EMPR can be utilized as a highly effective lossy compression algorithm for hyperspectral images. In addition to these, an efficient variety of EMPR is also introduced in the paper, in order to increase the compression efficiency. The results are benchmarked with several state-of-the-art lossy compression methods. It is observed that both higher peak-signal-to-noise-ratio values and improved classification accuracy are achieved from EMPR based methods.**

***Index Terms*—Hyperspectral images, lossy compression, Enhanced Multivariance Products Representation, classification accuracy, JPEG2000**

I. INTRODUCTION

HYPERSPECTRAL (HS) images are comprised of measurements of electro-magnetic energy distributed in hundreds of narrow bands. Due to the rich information in both the spectral and spatial domains, hyperspectral imagery (HSI) has a wide variety of applications, such as assessment of food quality and safety [1], [2], artwork authentication [3] and examination of drug forgeries [4]. HSI is also employed in biomedical engineering applications such as the classification of corneal epithelium injuries [5], extraction of the properties of cornea tissues [6] and gastric cancer diagnosis [7]. In addition, HSI is also widely used in many remote sensing applications [8]–[12] including image classification and pattern recognition [13], [14], and spectral unmixing [15]. Unfortunately, all of these applications come with the cost of high memory requirements due to the huge amounts of data. To this end, lossy or lossless compression of HS images has been the focus of research publications in the last decade [12], [15]–[29]. These compression algorithms adopt a variety of approaches. Traditional 2D image compression algorithms are applied to each band and achieve a compressed version of the HS cube [27], [30], [31]. These methods can provide satisfactory compression rates but fail to exploit inter-band correlation. To this end, some of these methods are extended to their 3D versions for compression of HS images [12], [32], [33], though the extended methods inevitably suffer from the high computational complexity. For this reason, sparse representations via dictionary learning methods were proposed [17], [26], [34]. Matrix and tensor decomposition as well as factorization methods were also employed in HSI compression [24], [25], [35]–[37]. Besides, wavelet based compression methods are also developed to this end [12], [38], [39]. On the other hand, with the rapid improvement in GPU technology, Convolutional Neural Networks based schemes adopted to HSI compression [40].

In this paper, a method called Enhanced Multivariance Products Representation (EMPR) [41]–[45] for hyperspectral image compression is proposed. As a finite data decomposition technique, EMPR enables multidimensional data to be represented with lower dimensional entities. By truncating this finite expansion at a certain level, an approximation for the multidimensional data under consideration can be obtained. This truncation also reveals a data reduction approach, which allows EMPR to be utilized as an algorithm for lossy compression of HS images [24], [35]. Thus, EMPR can be considered as a tensor decomposition based lossy compression method for HS images. In order to increase the compression efficiency of the present method, an EMPR variation called Iterative EMPR [45] is applied to HSI in this paper. In addition to these approaches, EMPR is combined with the JPEG2000 lossless compression method with the help of the Discrete Haar Transform (DHT) [46]. It is observed that higher PSNR values are achieved for several HS data sets acquired by various sensors in comparison with a recently published EMPR based method named as Tridiagonal Folmat

Manuscript received 16 May 2020; accepted 10 Oct. 2020. Date of publication xxx; date of current version xxx. This work is partially supported by the National Young Researchers Career Development Programme (3501 TUBİTAK CAREER); Scientific and Technical Research Council of Turkey (TUBİTAK) with grant number 114E200; the International Postdoctoral Research Scholarship Programme (2219 BİDEB); TUBİTAK funding (1059B191800733). (Corresponding author: Jinchang Ren).

S. Tuna is with Fatih Sultan Mehmet Vakıf University, Computer Engineering Department, İstanbul, Turkey (Türkiye). e-mail: stuna@fsm.edu.tr

B. U. Töreyin is with İstanbul Technical University, Informatics Institute, Applied Informatics Department, İstanbul, Turkey (Türkiye). e-mail: toreyin@itu.edu.tr

M. Demiralp is with İstanbul Technical University, Informatics Institute, Computational Science and Engineering Department, İstanbul, Turkey (Türkiye). e-mail: metin.demiralp@gmail.com

J. Ren and S. Marshall are with the Centre for Signal and Image Processing, Department of Electronic and Electrical Engineering, University of Strathclyde, Glasgow G1 1XW, U.K. e-mail: jinchang.ren@strath.ac.uk; stephen.marshall@strath.ac.uk

Enhanced Multivariance Products Representation (TFEMPR) [47]. TFEMPR is a sophisticated and recursive data reduction method based on EMPR and represents a multidimensional array as the product of two orthonormal and one tridiagonal multidimensional array by using the concepts folmats and folvecs [47].

In [24], it is shown that TFEMPR is a more efficient method for compressing HS images in comparison with some existing lossy compression methods. These methods include Compressive-Projection Principal Component Analysis (CP-PCA) [18], Generalized Orthogonal Matching Pursuit (gOMP) algorithm [19], Specialized Interior-Point (SIP) representation [20], Least Absolute Shrinkage and Selection Operator (LASSO) representation [21], Bayesian Compressive Sensing (BCS) [22], Basis Pursuit (BP) algorithm [23] and Sparsity-Based Hyperspectral Image-Compression algorithm [17].

In this paper, we propose a new Iterative EMPR approach combined with JPEG2000 by using DHT. According to the implementation and results, the proposed method is more efficient in representing HS image data than TFEMPR [24] at lower bit rates. Moreover, the proposed approach also outperforms two well-known lossy compression methods including PCA+JPEG2000 approach [27] and the 3D SPECK algorithm [48]. Besides, the proposed method is compared with two state-of-the-art low rank tensor decomposition based hyperspectral image compression techniques. These techniques are Patch-Based Low Rank Tensor Decomposition (PLTD) [36] and Non-Local Tensor Sparse Representation and Low Rank Regularization (NTSRLR) [49] methods. The results indicate that the proposed approach outperforms the corresponding lossy compression methods and preserve more detail at very low bit rates. These findings represent the first important contribution of this paper to the scientific literature. In addition, our approach is further validated by comparing the results of data classification in HSI. The resulting images after decompression are classified in comparison to the corresponding ground truth images. It is shown that the proposed approach yields higher overall accuracy and outperforms state-of-the-art techniques [12] in HS image classification. In this work, classification of HS images using EMPR based methods is put into practice for the first time in the scientific literature. This novel aspect represents the second important contribution of this work.

The remainder of this paper is organized as follows. A detailed explanation of EMPR for HSI is discussed in Section II. Iterative EMPR and its combination with JPEG2000 are described therein. The experimental setup, explanation of compression rates of the proposed method, computational complexity issues, are presented in Section III. Section IV discusses the experimental results for performance evaluation, including compression rate and classification accuracy for comparison where the section is finalized with the comments on parameter selection. Finally, some concluding remarks and comments about future studies are provided in Section V.

II. Enhanced Multivariance Products Representation (EMPR)

Enhanced Multivariance Products Representation (EMPR) is an efficient data decomposition method [41]–[45]. It enables multidimensional data to be represented in terms of lower-dimensional components. Thus, it can be considered as a series of lower dimensional structures instead of the high dimensional original data.

From the scientific or engineering experiments, one of the most important challenges in analyzing data is the "curse of dimensionality" [50]. Therefore, managing this phenomenon by reducing the number of dimensions becomes critical. To this end, EMPR can be considered as an efficient tool for addressing multidimensional problems.

Although EMPR is capable of decomposing N-dimensional structures of data, the 3-dimensional case is considered in the present paper without loss of generality. Since HS images are represented in three dimensions (two spatial, one spectral), it is more convenient to explain EMPR and its philosophy by taking multidimensionality as three throughout the paper. On the other hand, all formulations which will be given here can easily be generalized to N-dimensional case. In this section, EMPR for HSI will be introduced and discussed. An EMPR based method, Iterative EMPR, will then be presented. The combination of EMPR with JPEG2000 via Discrete Haar Transform (DHT) will be given at the end of this section.

A. Plain EMPR for HSI

Let $\mathbf{H}$ denote the 3-dimensional hyperspectral cube of size $n_1 \times n_2 \times n_3$. This means $\mathbf{H}$ has n_3 spectral bands and each band includes $n_1 \times n_2$ pixels storing intensity values at the corresponding wavelength. The EMPR expansion of this cube can be expressed as follows

$$\mathbf{H} = h^{(0)} \left[\bigotimes_{r=1}^{3} \mathbf{s}^{(r)} \right] + \sum_{i=1}^{3} \mathbf{h}^{(i)} \otimes \left[\bigotimes_{\substack{r=1 \\ r\neq i}}^{3} \mathbf{s}^{(r)} \right] + \sum_{\substack{i,j=1 \\ i<j}}^{3} \mathbf{h}^{(i,j)} \otimes \left[\bigotimes_{\substack{r=1 \\ r\neq i,j}}^{3} \mathbf{s}^{(r)} \right] + \mathbf{h}^{(1,2,3)}. \quad (1)$$

where $h^{(0)}$, $\mathbf{h}^{(i)}$ and $\mathbf{h}^{(i,j)}$ denote the zero-way, the one-way and the two-way *EMPR components*, respectively and $\otimes$ denotes the outer product operation [41]–[45]. The EMPR expansion is a finite sum hence it involves exactly 2^3 EMPR components. [41]–[45]. In (1), $h^{(0)}$ is a zero-dimensional entity which can be considered as a scalar, $\mathbf{h}^{(i)}$ denotes one-dimensional entities which are the vectors, and $\mathbf{h}^{(i,j)}$ stands for the two-dimensional entities which can be considered as the matrices. In addition to these, other entities involved in (1) and denoted by $\mathbf{s}^{(r)}$ are called the *support vectors* whence they are one dimensional entities [41]–[45]. Thus, $\mathbf{s}^{(1)}$ and $\mathbf{s}^{(2)}$ are the first and second support vectors residing on the first and second spatial axes of the 3-dimensional hyperspectral cube $\mathbf{H}$, respectively. Similarly, $\mathbf{s}^{(3)}$ denotes the third support vector laying on the third axis which defines the spectrum of $\mathbf{H}$. Thus, one can easily verify that $\mathbf{s}^{(r)}$ is a vector composed of n_r

elements where n_r is a positive integer ($r = 1, 2, 3$), assuming that the size of $\mathbf{H}$ is $n_1 \times n_2 \times n_3$. Support vectors bring flexibility to EMPR expansion and must be chosen carefully where the details of this selection will be given later in this section. This selection is critical since it has a direct impact on the representation efficiency of the relevant EMPR expansion.

As indicated above, as $\mathbf{H}$ is 3-dimensional, hence it should be represented by means of the 3-dimensional entities, as EMPR has an additive nature. Besides, three suitable support vectors should be required in order to construct the corresponding entities. By multiplying these support vectors with the relevant EMPR components following the outer product definition, 3-dimensional but less complicated structures are obtained. These new entities, acquired by the multiplication of an EMPR component with relevant support vectors, are called the *EMPR terms* [41]–[45]. Consequently, it is convenient to name the EMPR term including $h^{(0)}$ and all three corresponding support vectors as *zeroth EMPR term*. Similarly, the term including $\mathbf{h}^{(i)}$ and all the corresponding support vectors except the i-th one is called *i-th EMPR term*. Accordingly, the term composed of $\mathbf{h}^{(i,j)}$ and all the corresponding support vectors excluding the i-th and j-th ones, respectively, are called *(i, j)-th EMPR term*. All EMPR terms are of the same size as $\mathbf{H}$ which is $n_1 \times n_2 \times n_3$, but rank-one.

In order to adjust the contributions of each intensity value in $\mathbf{H}$, three weight vectors including weighting ratios can be utilized in EMPR expansion. The weight vectors are composed of non-negative real values and must satisfy the following conditions

$$\left\|\boldsymbol{\omega}^{(1)}\right\|_1 = 1, \quad \left\|\boldsymbol{\omega}^{(2)}\right\|_1 = 1, \quad \left\|\boldsymbol{\omega}^{(3)}\right\|_1 = 1. \tag{2}$$

In (2), it is clear that the sum of all elements for each weight vector should be equal to 1. This property holds due to the statistical necessities and it facilitates the relevant computations in the determining process of EMPR components.

However, the EMPR components should satisfy the following constraints

$$\sum_{i_p=1}^{n_p} \boldsymbol{\omega}_{i_p}^{(p)}\, \mathbf{s}_{i_p}^{(p)}\, \mathbf{h}_{i_1,\ldots,i_m}^{(1,\ldots,m)} = 0; \quad 1 \le p \le m \in \{1, 2, 3\} \tag{3}$$

where $\boldsymbol{\omega}_{i_p}^{(p)}$ and $\mathbf{s}_{i_p}^{(p)}$ are the i_p-th elements of the p-th weight vector $\boldsymbol{\omega}^{(p)}$ and p-th support vector $\mathbf{s}^{(p)}$, respectively while $\mathbf{h}_{i_1,\ldots,i_m}^{(1,\ldots,m)}$ denotes the $(i_1, \ldots, i_m)$-th entry of the corresponding EMPR component $\mathbf{h}^{(1,\ldots,m)}$. The equalities in (3) are called *vanishing conditions* and lead to two important features of EMPR components, which are the mutual orthogonality and the uniqueness under a certain set of support vectors.

By utilizing the vanishing conditions in (3) with the help of the weight vectors given in (2) and the pre-selected support vectors, the zero-way EMPR component, i.e. $h^{(0)}$, can be calculated uniquely as follows

$$h^{(0)} = \sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\sum_{k=1}^{n_3} \boldsymbol{\omega}_i^{(1)}\, \boldsymbol{\omega}_j^{(2)}\, \boldsymbol{\omega}_k^{(3)}\, \mathbf{s}_i^{(1)}\, \mathbf{s}_j^{(2)}\, \mathbf{s}_k^{(3)}\, \mathbf{H}_{ijk}. \tag{4}$$

It is worth noting that the right-hand side of the equation (4) denotes a weighted sum of $\mathbf{H}$ multiplied by the relevant support vector elements over the whole hyperspectral cube. Thus, $h^{(0)}$ associates to a special weighted average value of $\mathbf{H}$.

Besides, (4) can be re-expressed by following the tensor product definitions of multi-linear algebra given in [51] as follows

$$h^{(0)} = \mathbf{H}\, \bar{\times}_1 \left(\boldsymbol{\omega}^{(1)} \circledast \mathbf{s}^{(1)}\right) \bar{\times}_2 \left(\boldsymbol{\omega}^{(2)} \circledast \mathbf{s}^{(2)}\right) \bar{\times}_3 \left(\boldsymbol{\omega}^{(3)} \circledast \mathbf{s}^{(3)}\right)$$

where $\circledast$ denotes the Hadamard (or elementwise) product and $\bar{\times}_i$ stands for the mode$-i$ tensor-vector product [51]. These notations are widely used in tensor algebra due to their concise formulation. On the other hand, we do not intend to follow these symbols since we have exactly 3-ways in our analyses. Here, we prefer explicity rather than compactness in notation, though all EMPR components can be expressed using the above-mentioned tensor product definitions.

After giving the details about the evaluation process of the zeroth EMPR component, we can proceed with the one-way EMPR components. By combining (2) and (3) again, the corresponding elements of three one-way EMPR components for $\mathbf{H}$ are computed uniquely as follows

$$\begin{aligned}
\mathbf{h}_i^{(1)} &= \sum_{j=1}^{n_2}\sum_{k=1}^{n_3} \boldsymbol{\omega}_j^{(2)}\, \boldsymbol{\omega}_k^{(3)}\, \mathbf{s}_j^{(2)}\, \mathbf{s}_k^{(3)}\, \mathbf{H}_{ijk} - h^{(0)}\, \mathbf{s}_i^{(1)}, \\
\mathbf{h}_j^{(2)} &= \sum_{i=1}^{n_1}\sum_{k=1}^{n_3} \boldsymbol{\omega}_i^{(1)}\, \boldsymbol{\omega}_k^{(3)}\, \mathbf{s}_i^{(1)}\, \mathbf{s}_k^{(3)}\, \mathbf{H}_{ijk} - h^{(0)}\, \mathbf{s}_j^{(2)}, \\
\mathbf{h}_k^{(3)} &= \sum_{i=1}^{n_1}\sum_{j=1}^{n_2} \boldsymbol{\omega}_i^{(1)}\, \boldsymbol{\omega}_j^{(2)}\, \mathbf{s}_i^{(1)}\, \mathbf{s}_j^{(2)}\, \mathbf{H}_{ijk} - h^{(0)}\, \mathbf{s}_k^{(3)}.
\end{aligned} \tag{5}$$

while three two-way EMPR components can be obtained uniquely as

$$\begin{aligned}
\mathbf{h}_{ij}^{(1,2)} &= \sum_{k=1}^{n_3} \boldsymbol{\omega}_k^{(3)}\, \mathbf{s}_k^{(3)}\, \mathbf{H}_{ijk} - h^{(0)}\, \mathbf{s}_i^{(1)}\, \mathbf{s}_j^{(2)} \\
&\quad - \mathbf{h}_i^{(1)}\, \mathbf{s}_j^{(2)} - \mathbf{s}_i^{(1)}\, \mathbf{h}_j^{(2)}, \\
\mathbf{h}_{ik}^{(1,3)} &= \sum_{j=1}^{n_2} \boldsymbol{\omega}_j^{(2)}\, \mathbf{s}_j^{(2)}\, \mathbf{H}_{ijk} - h^{(0)}\, \mathbf{s}_i^{(1)}\, \mathbf{s}_k^{(3)} \\
&\quad - \mathbf{h}_i^{(1)}\, \mathbf{s}_k^{(3)} - \mathbf{s}_i^{(1)}\, \mathbf{h}_k^{(3)}, \\
\mathbf{h}_{jk}^{(2,3)} &= \sum_{i=1}^{n_1} \boldsymbol{\omega}_i^{(1)}\, \mathbf{s}_i^{(1)}\, \mathbf{H}_{ijk} - h^{(0)}\, \mathbf{s}_j^{(2)}\, \mathbf{s}_k^{(3)} \\
&\quad - \mathbf{h}_j^{(2)}\, \mathbf{s}_k^{(3)} - \mathbf{s}_j^{(2)}\, \mathbf{h}_k^{(3)}
\end{aligned} \tag{6}$$

in a similar manner. The three-way EMPR component which is the last element of the right-hand side in (1) can be calculated by subtracting the EMPR terms whose explicit definitions are given in (4), (5) and (6) respectively from the original data $\mathbf{H}$.

By applying truncations to the right-hand side of the expansion in (1) at certain levels, an approximation for $\mathbf{H}$ can be achieved. To this end, if only the zeroth EMPR term is taken into consideration, which means the rest of EMPR terms are neglected, the zeroth order EMPR approximant is acquired. The higher (first and second) order EMPR approximants can be obtained in a similar manner and all approximants can be stated explicitly as follows

$$\boldsymbol{\pi_0} = h^{(0)} \left[\bigotimes_{r=1}^{3} \mathbf{s}^{(r)} \right], \quad \boldsymbol{\pi_1} = \boldsymbol{\pi_0} + \sum_{i=1}^{3} \mathbf{h}^{(i)} \otimes \left[\bigotimes_{\substack{r=1 \\ r\neq i}}^{3} \mathbf{s}^{(r)} \right],$$

$$\boldsymbol{\pi_2} = \boldsymbol{\pi_1} + \sum_{\substack{i,j=1 \\ i<j}}^{3} \mathbf{h}^{(i,j)} \otimes \left[\bigotimes_{\substack{r=1 \\ r\neq i,j}}^{3} \mathbf{s}^{(r)} \right]. \quad (7)$$

Finally, one of the most important issues in EMPR, which is the selection of the support vectors will be explained. Initially, the support vectors should satisfy the following criteria

$$\sum_{i_p=1}^{n_p} \boldsymbol{\omega}_{i_p}^{(p)} \left[\mathbf{s}_{i_p}^{(p)} \right]^2 = 1; \qquad p = 1, 2, 3. \quad (8)$$

which means that all support vectors should be normalized under the corresponding weight vector. The normalization procedure is essential as the support vectors should only indicate the direction due to the fact that the contribution coefficients are stored as the elements of EMPR components.

Any convenient set of support vectors can be selected as long as they satisfy the conditions in (3) and (8). To this end, the vectors whose elements are given explicitly as

$$\begin{aligned} \mathcal{S}_i^{(1)} &= \sum_{j=1}^{n_2} \sum_{k=1}^{n_3} \boldsymbol{\omega}_j^{(2)} \boldsymbol{\omega}_k^{(3)} \mathbf{H}_{ijk}, \\ \mathcal{S}_j^{(2)} &= \sum_{i=1}^{n_1} \sum_{k=1}^{n_3} \boldsymbol{\omega}_i^{(1)} \boldsymbol{\omega}_k^{(3)} \mathbf{H}_{ijk}, \\ \mathcal{S}_k^{(3)} &= \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \boldsymbol{\omega}_i^{(1)} \boldsymbol{\omega}_j^{(2)} \mathbf{H}_{ijk}. \end{aligned} \quad (9)$$

can be assessed as the support vectors of an EMPR expansion, after normalization according to the conditions in (8). Although this selection is not the optimal case, the corresponding support vectors in (9) can be determined without any apparent difficulty and utilized in an EMPR process as long as they do not vanish [52]. Moreover, it is easy to recognize that each formula in (9) depicts a weighted average of $\mathbf{H}$ over all axes of the corresponding 3-dimensional cube excluding one direction (axis). Thus, the formulas in (9) specify averaged directions for the data under consideration. To this end, the support vectors chosen by the utilization of the formulas in (9) are called *Averaged Directional Supports (ADS)* [52]. These support vectors can be encountered in almost in all of EMPR applications existing in the scientific literature [41]–[45].

Before concluding this subsection, it becomes useful to present the similarities and differences between EMPR and the well-known CANDECOMP/PARAFAC [51]. Both EMPR and CANDECOMP/PARAFAC are the tensor decomposition methods and composed of finite number of elements. In CANDECOMP/PARAFAC, the tensor on the focus is represented in terms of rank-one tensors.

For EMPR, the target tensor is expressed as the sum of four rank-one tensors (0th, 1st, 2nd and 3rd EMPR terms), and four terms whose ranks are greater than one (4th, 5th, 6th and 7the EMPR terms) each of which is order 3 (or 3-dimensional).

In CANDECOMP/PARAFAC, the vectors used to construct the rank-one tensors by following the outer product are determined via convex optimization. Instead, in EMPR, initially a group of support vectors is selected, and the corresponding EMPR components (scalar, vectors and matrices) are determined by following the weighted mean approach as shown in Eqs. (4), (5) and (6).

B. Iterative EMPR

If we notice EMPR expansion in (1) and approximants in (7), it is possible to write down the following

$$\mathbf{h}^{(1,2,3)} = \mathbf{H} - \boldsymbol{\pi_2} \quad (10)$$

where $\mathbf{h}^{(1,2,3)}$ could be marked as the residual term of the EMPR expansion of $\mathbf{H}$. Although $\mathbf{h}^{(1,2,3)}$ can be considered as residual, it contains information about the hyperspectral cube $\mathbf{H}$. By neglecting this term in the corresponding EMPR expansion, which also means dealing with the second order EMPR approximant, some information (which may be important) belonging to $\mathbf{H}$ is ignored. Thus, the quality of the corresponding representation can be affected negatively. In order to reduce this possible undesirable effect, the EMPR procedure is applied to the residual term to take this neglected information into account. To this end, the operation in (10) can be considered as the zeroth iteration of the Iterative EMPR process. If we denote the zeroth residual term as

$$\{\mathbf{H}\}_{(1)} = \mathbf{h}^{(1,2,3)} \quad (11)$$

and apply the second order EMPR approximation to $\{\mathbf{H}\}_{(1)}$

$$\{\mathbf{H}\}_{(1)} = \{\boldsymbol{\pi_2}\}_{(1)} + \left\{\mathbf{h}^{(1,2,3)}\right\}_{(1)} \quad (12)$$

is obtained where $\{\boldsymbol{\pi_2}\}_{(1)}$ and $\left\{\mathbf{h}^{(1,2,3)}\right\}_{(1)}$ stand for the second order EMPR approximant and the corresponding residual term of the first iteration, respectively.

If we continue to apply the relevant EMPR process to the residual term of each iteration consequently

$$\left\{\mathbf{h}^{(1,2,3)}\right\}_{(m)} = \{\mathbf{H}\}_{(m)} - \{\boldsymbol{\pi_2}\}_{(m)}; \quad m = 0, 1, 2, \ldots \quad (13)$$

is achieved, where (m) denotes the iteration number of the corresponding Iterative EMPR process and $\{\mathbf{H}\}_{(0)} = \mathbf{H}$. This iteration scheme can be pursued until a satisfactory compression efficiency is achieved. On the other hand, each iteration brings more data to be stored. If the iteration number of the Iterative EMPR process is selected as m, the data-rate value to be attained becomes m times bit-per-pixel-per-band (bpppb) value of the original EMPR procedure. This issue will be discussed in Section III.

C. Combining EMPR with JPEG2000 using DHT

The Discrete Haar Transform (DHT) is the most fundamental wavelet transforms in the scientific literature [46]. It helps to split a 1-D signal of even size, say $2N$, into two 1-D signals of size N. These two equally sized signals involve the low band and high band characteristics of the input signal, respectively. If we denote any 1-D signal of size $2N$ as $s[\cdot]$, the DHT of this signal can be obtained as follows

$$l[k] = \frac{s[2k] + s[2k-1]}{\sqrt{2}}, \qquad h[k] = \frac{s[2k] - s[2k-1]}{\sqrt{2}}; \quad k = 1, \ldots, N \tag{14}$$

where $l[\cdot]$ and $h[\cdot]$ stand for the low frequency and high-frequency components of signal $s[\cdot]$, respectively.

The DHT can be implemented on the hyperspectral image $\mathbf{H}$ by applying it to the spectral signal of each pixel consecutively. Thus, it becomes possible to say that $n_1 \times n_2$ signals of size n_3 are split into two equal parts according to their spectral correlations. In this way, each spectral signal of $\mathbf{H}$ is partitioned as the low frequency and high frequency portions. If we gather the low frequency portions and sort them to their pixel positions, the low band cube $\mathbf{H_{low}}$ is obtained while the high frequency components form the high band cube, $\mathbf{H_{high}}$. Both $\mathbf{H_{low}}$ and $\mathbf{H_{high}}$ have a size of $n_1 \times n_2 \times (n_3/2)$. As we mentioned before, the computation of the EMPR terms depends on weighted averages, and this property enables EMPR to represent high correlated data sets much better than that of the low correlated ones. Thus, it becomes rational to apply EMPR or Iterative EMPR to$\mathbf{H_{high}}$ portion of the corresponding HS data set $\mathbf{H}$. On the other hand, the other portion, $\mathbf{H_{low}}$, which contains the spectrally low correlated information of H and can be compressed using an efficient and easy to implement lossless compression algorithm such as JPEG2000 [53], as illustrated in Fig. 1. As

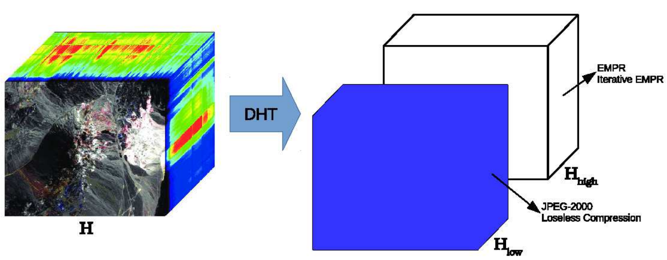


Fig. 1: Application of DHT to $\mathbf{H}$ and obtaining two equally sized cubes $\mathbf{H_{low}}$ and $\mathbf{H_{high}}$. $\mathbf{H_{low}}$ is the low frequency part while $\mathbf{H_{high}}$ stands for the high frequency part of $\mathbf{H}$.

we mentioned before, the computation of the EMPR terms depend on weighted averages. This property enables EMPR to represent high correlated data sets much better than that of the low correlated ones. Thus, it becomes rational to apply EMPR or Iterative EMPR to $\mathbf{H_{high}}$ portion of the corresponding HS data set $\mathbf{H}$. On the other hand, the other portion, $\mathbf{H_{low}}$, which contains the spectrally low correlated information of $\mathbf{H}$ can be compressed using an efficient and easy to implement lossless compression algorithm such as JPEG2000 [27], as can be seen in Fig. 1.

After the application of the EMPR based algorithm and any suitable lossless compression method, the obtained sub-cubes are combined with the help of Inverse Discrete Haar Transform (I-DHT) [46] using the following equations

$$s[2k] = \frac{l[k] + h[k]}{\sqrt{2}}, \qquad s[2k-1] = \frac{l[k] - h[k]}{\sqrt{2}}; \quad k = 1, \ldots, N \tag{15}$$

and finally, a lossy compressed version of $\mathbf{H}$ is obtained. In addition, this approach can be considered as a hybrid way of compression since the encoding part is composed of both lossy and lossless encoders. Although the application of higher level DHT is possible in our case, it is not used in this work. In the present paper, all calculations related to DHT are performed using only the first level.

III. Implementations

A. Data Sets Used

In this paper, ten different hyperspectral data sets are utilized to validate the performance of the proposed approach. These data sets are selected from various sensors to fully demonstrate how EMPR and Iterative EMPR perform with data collected from sensors with different characteristics. These data sets are handled after cropping in order to make a fair comparison with the state-of-the-art [9], [24], [27], [36], [48], [49]. All the specifications about the utilized data sets can be found in Table I and the corresponding pseudo-color images of the first four of them are shown in Fig. 2. Additionally, the pseudo-color images for the $\mathbf{H_{high}}$ and $\mathbf{H_{low}}$ parts of Low Altitude and Cuprite data sets are presented in Fig. 3. It is easy to observe from Fig. 3 that sub-figures (a) and (c) including $\mathbf{H_{high}}$ parts of the relevant images are capable of presenting edges since they include high frequency spectral signatures for the corresponding data sets. On the other hand, $\mathbf{H_{low}}$ parts included by (b) and (d) sub-figures looks similar to their original pseudo-color images shown in Fig. 2 as they include the low frequency terms of the corresponding spectral signals.

As stated earlier, EMPR depends on weighted averages, thus the selection of three weight vectors is important in representing the hyperspectral cube under consideration. Although any weight vector mentioned in Section II and satisfying the conditions in (2) could be utilized, the simplest case, which is the equally distributed weights, are used in these implementations. Thus, these weight vectors are given as follows

$$\begin{aligned} w_i^{(1)} &= 1/n_1, \qquad i = 1, \ldots, n_1 \\ w_j^{(2)} &= 1/n_2, \qquad j = 1, \ldots, n_2 \\ w_k^{(3)} &= 1/n_3, \qquad k = 1, \ldots, n_3. \end{aligned} \tag{16}$$

Additionally, selection of the support vectors is another crucial issue in EMPR's representation efficiency. Although it is possible to optimize the support vectors in EMPR, this is not carried out here as it is beyond the scope of this paper. Thus, it becomes a rational approach to deal with the Averaged Directional Supports (ADS) whose formulations are given explicitly in (9).

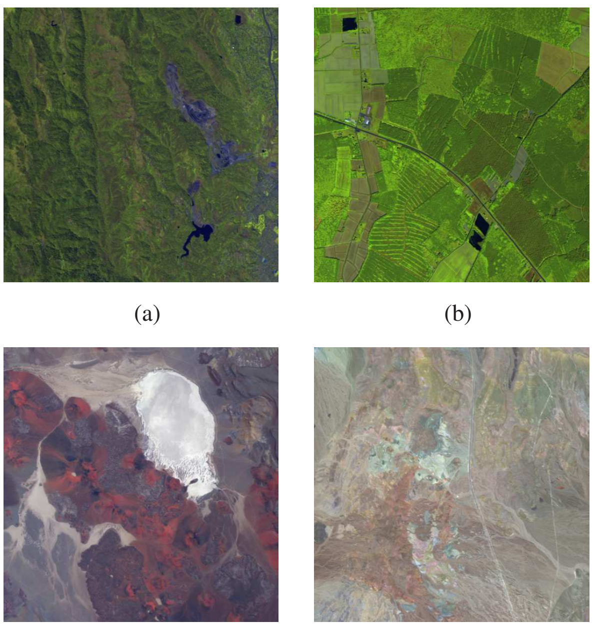
On the other hand, it is noted that three levels of EMPR approximations could be handled for representing an HS cube via certain EMPR approximants in (7). In this section, all results are obtained by performing the second order EMPR approximant, that is $\boldsymbol{\pi_2}$. This preference occurs since the zeroth and the first EMPR approximants result in too low

TABLE I: Hyperspectral data set specifications

Name (H)	Samples (n_1)	Lines (n_2)	Bands (n_3)	Bit depth	Sensor	Cropped to
Jasper Ridge (JR)	2587	614	224	16	AVIRIS	$512 \times 512 \times 224$
Low Altitude (LA)	1432	614	224	16	AVIRIS	$512 \times 512 \times 224$
Lunar Lake (LL)	3689	614	224	16	AVIRIS	$512 \times 512 \times 224$
Cuprite (CU)	512	614	224	16	AVIRIS	$512 \times 512 \times 224$
Botswana (BO)	1476	256	145	16	HYPERION	$256 \times 256 \times 144$
Pavia Uni. (PU)	610	340	103	16	ROSIS	$256 \times 256 \times 102$
92AV3 C	145	145	224	16	AVIRIS	$144 \times 144 \times 184$
Salinas B (SA)	512	217	224	16	AVIRIS	$144 \times 144 \times 184$
Washington DC MALL (WA)	1280	307	191	16	HYDICE	$1280 \times 307 \times 190$
Indian Pines (IP)	145	145	224	16	AVIRIS	$145 \times 145 \times 200$

bpppb values hence the comparisons with other methods become impossible.

All experiments are carried out using MATLAB R2020a on AMD Ryzen 7 3700X CPU @ 3.60GHz processor and 32 GB memory under Linux Ubuntu 18.04.4 LTS operating system.



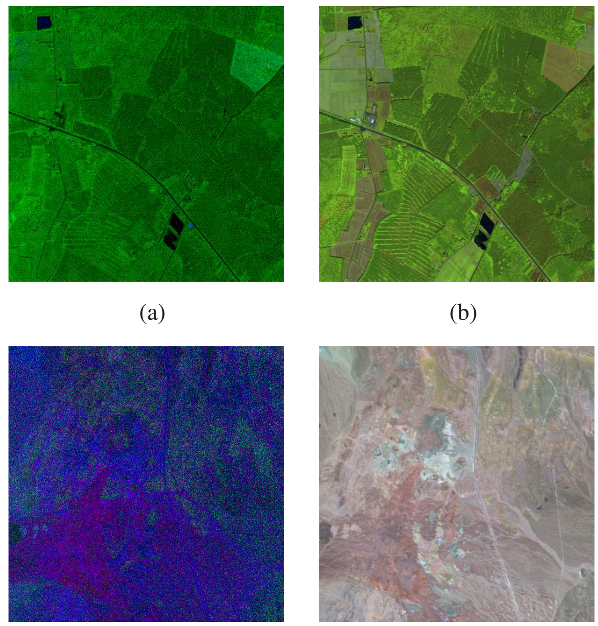
(a) (b) (c) (d)

Fig. 2: Pseudo-color images for the data sets: (a) Jasper Ridge, (b) Low Altitude, (c) Lunar Lake and (d) Cuprite

B. Compression Rates

Although we only deal with $\boldsymbol{\pi_2}$ in this section, bit-per-pixel-per-band (bpppb) values for the zeroth, first and second order EMPR approximants for an HS cube of size $n_1 \times n_2 \times n_3$ are calculated by pursuing the evaluation approach followed in [24], [25] as

$$\begin{aligned} bpppb(\boldsymbol{\pi_0}) &= \frac{32 \cdot (m+1) \cdot (1 + n_1 + n_2 + n_3)}{16 \cdot N} \\ bpppb(\boldsymbol{\pi_1}) &= bpppb(\boldsymbol{\pi_0}) + \frac{32 \cdot (m+1) \cdot (n_1 + n_2 + n_3)}{16 \cdot N} \\ bpppb(\boldsymbol{\pi_2}) &= bpppb(\boldsymbol{\pi_1}) + \frac{32 \cdot (m+1) \cdot \left(\frac{N}{n3} + \frac{N}{n_2} + \frac{N}{n_1} \right)}{16 \cdot N} \end{aligned} \quad (17)$$



(a) (b) (c) (d)

Fig. 3: Pseudo-color images for the half cubes after performing DHT: (a) $\mathbf{H_{high}}$ for Low Altitude, (b) $\mathbf{H_{low}}$ for Low Altitude, (c) $\mathbf{H_{high}}$ for Cuprite and (d) $\mathbf{H_{low}}$ for Cuprite

where m stands for the iteration number (no iteration, Plain EMPR when $m = 0$) and $N = n_1 \times n_2 \times n_3$. Since all HS cubes in Table I are composed of 16-bit values, the denominators of the above formulae are multiplied by 16. On the other hand, since the EMPR algorithm deals with floating numbers, nominators are multiplied by 32 which corresponds to single precision.

It is important to note that if n_1, n_2 and n_3 are assumed as infinitely large which means that the HS cube to be dealt with is of infinite size, the bpppb values decrease and converge to zero. This suggests that smaller bpppb values can be achieved while dealing with large HS data sets.

As stated above, only $bpppb(\boldsymbol{\pi_2})$ is on the focus in this work which means that encoder generates and transmits a scalar $h^{(0)}$; three vectors, $\mathbf{h}^{(1)}$, $\mathbf{h}^{(2)}$ and $\mathbf{h}^{(3)}$; three matrices $\mathbf{h}^{(1,2)}$, $\mathbf{h}^{(1,3)}$ and $\mathbf{h}^{(2,3)}$ and three support vectors, i.e. $\mathbf{s}^{(1)}$, $\mathbf{s}^{(2)}$ and $\mathbf{s}^{(3)}$. After the transmission process, decoder combines these components by following outer product and summation operations according to the last row in Eq. (7).

If EMPR or Iterative EMPR combined with DHT whose details are given in subsection II-C, bpppb values for the corresponding representation can be determined as

$$bpppb(\text{EMPR+JPEG2000}) = bpppb(\boldsymbol{\pi_2}) + bpppb(\text{JPEG2000}) \quad (18)$$

where $bpppb$(JPEG2000) is calculated by performing JPEG2000 lossless compression to each band of $\mathbf{H_{low}}$ consecutively. As one of the most straightforward methods for compressing a multi-band image with JPEG2000, this method is called JPEG2000 band-independent fixed-rate (JPEG2000-BIFR) [31] where the fixed-rate is just 1.0 as in our case, whence we prefer to employ the lossless compression here. After the compression of each band, the size of the encoded band is calculated in bytes and accumulated across all the bands to form the total size of the corresponding $\mathbf{H_{low}}$. The total size for the $\mathbf{H_{low}}$ half cubes after band-wise JPEG2000 implementation is given in Table II for comparison.

TABLE II: Sizes of low band half of HS cubes after JPEG2000 implementation

HS Cube	Size in bytes
JR	1529974
LA	3002795
LL	175563
CU	272770
BO	16519
PU	15753
WA	317842
IP	18800

For implementation, the `imwrite` function in MATLAB is called for each spectral band in order to compress the low-frequency part $\mathbf{H}_{low}$ using JPEG2000 lossless compression with the compression mode as 'lossless'. MATLAB uses Discrete Wavelet Transform in 5 decomposition levels with 5×3 kernels following the LRCP (Layer, Resolution, Components, Position) order in its embedded encoding scheme.

C. Computational Complexity

Since EMPR is based on weighted averages, its implementation requires many additions and multiplications. The precise number of these operations depends on the size of the HS cube under consideration.

If the cube $\mathbf{H}$ is assumed to be of size $n_1 \times n_2 \times n_3$, N stands for $n_1 n_2 n_3$ and the weight vectors in (16) are utilized, the following numbers of floating point operations for calculation of EMPR components are given in Table III.

In Table III, one can easily verify that the addition amount for each EMPR component is the same. Also, the number of multiplications decreases while the order of the term increases since the number of nested sums diminishes. Also, the number of subtractions increases when the order of the term increases according to the definitions in (4), (5) and (6).

Nevertheless, the floating point operations which are required for calculating ADS vectors including normalization processes are tabulated in Table IV.

TABLE III: Operation counts for calculating EMPR components

Component	Addition	Multiplication	Subtraction
$h^{(0)}$	N	$3N+3$	$-$
$\mathbf{h}^{(1)}$	N	$2N+3n_1$	n_1
$\mathbf{h}^{(2)}$	N	$2N+3n_2$	n_2
$\mathbf{h}^{(3)}$	N	$2N+3n_3$	n_3
$\mathbf{h}^{(1,2)}$	N	$N+5n_1n_2$	$3n_1n_2$
$\mathbf{h}^{(1,3)}$	N	$N+5n_1n_3$	$3n_1n_3$
$\mathbf{h}^{(2,3)}$	N	$N+5n_2n_3$	$3n_2n_3$

TABLE IV: Operation counts for for calculating ADS (including normalization)

Support	Addition	Multiplication	Division	Square Root
$\mathbf{s}^{(1)}$	$N+n_1$	$4n_1$	n_1	1
$\mathbf{s}^{(2)}$	$N+n_2$	$4n_2$	n_2	1
$\mathbf{s}^{(3)}$	$N+n_3$	$4n_3$	n_3	1

Since the representation in (1) involves outer products and the summation of three-way arrays, each of size $n_1 \times n_2 \times n_3$, the floating point operations required to construct the EMPR approximants are given as follows

$$\begin{aligned} fl(\boldsymbol{\pi_0}) &= 9N + 6(n_1+n_2+n_3) + 6 \\ fl(\boldsymbol{\pi_1}) &= 15N + 4(n_1+n_2+n_3) + fl(\boldsymbol{\pi_0}) \\ fl(\boldsymbol{\pi_2}) &= 12N + 8(n_1n_2+n_1n_3+n_2n_3) + fl(\boldsymbol{\pi_1}) \end{aligned} \quad (19)$$

According to the analysis above, the computational complexity of the Plain EMPR with equally distributed weights and ADS is

$$\mathcal{O}\left(36\, n_1 n_2 n_3\right) \approx \mathcal{O}\left(n^3\right). \quad (20)$$

It is known that the computational complexity of single-level DHT of a 1-dimensional signal of size n is $\mathcal{O}(\log n)$ which is the same as the Inverse DHT [54]. Since we have $n_1 \times n_2$ 1-D signals of size n_3 in our case, the corresponding computational complexity of DHT in the proposed approach becomes

$$\mathcal{O}\left(n_1 n_2 \log n_3\right) \approx \mathcal{O}\left(n^2 \log n\right). \quad (21)$$

IV. Results

In this section, the performances of the proposed methods are presented using tables and figures. Moreover, the results obtained are compared with the TFEMPR [24], PCA+JPEG2000 [27], 3D SPECK [48], PLTD [36], NTSRLR [49] and 3D DCT [12] lossy compression methods in order to contrast the efficiency of the proposed approach.

A. Evaluation Metrics

For fair comparisons, some universal metrics which are utilized to measure the performance of the methods in signal and image processing are employed. These metrics are the mean squared error (MSE), maximum absolute error (MAE), signal-to-noise ratio (SNR), peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) [55]. If $\mathbf{H}$ and $\hat{\mathbf{H}}$ stand for the hyperspectral cube under consideration and its EMPR based representation (or compression), the abovementioned metrics can be defined mathematically as follows

$$MSE(\mathbf{H},\hat{\mathbf{H}}) = \frac{1}{N}\sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\sum_{k=1}^{n_3}\left(\mathbf{H}_{ijk}-\hat{\mathbf{H}}_{ijk}\right)^2,$$
$$MAE(\mathbf{H},\hat{\mathbf{H}}) = \max\{|\mathbf{H}-\hat{\mathbf{H}}|\},$$
$$SNR(\mathbf{H},\hat{\mathbf{H}}) = 10\log_{10}\left(\frac{\|\mathbf{H}\|_F^2}{MSE(\mathbf{H},\hat{\mathbf{H}})}\right),$$
$$PSNR(\mathbf{H},\hat{\mathbf{H}}) = 10\log_{10}\left(\frac{\text{peakval}^2}{MSE(\mathbf{H},\hat{\mathbf{H}})}\right),$$
$$SSIM(\mathbf{H},\hat{\mathbf{H}}) = \left[\ell(\mathbf{H},\hat{\mathbf{H}})\right]^{\alpha}\left[c(\mathbf{H},\hat{\mathbf{H}})\right]^{\beta}\left[s(\mathbf{H},\hat{\mathbf{H}})\right]^{\gamma}, \quad (22)$$

where $peakval$ in above PSNR formula is taken as $2^{16}-1$ since the data sets in Table I are stored as 16 bits.

On the other hand, $\ell(\cdot)$, $c(\cdot)$ and $s(\cdot)$ residing in the last formula of (22) stand for the similarity of luminance, contrast and structure, respectively. The exponents of these factors, i.e. α, β and γ are taken as 1 and regularization constants involved by these factors are taken as $C_1 = (0.01)^2$, $C_2 = (0.03)^2$ and $C_3 = C_2/2$ by default. SSIM is defined for 2D images to measure the similarity between two images for human perception [55]. For HS images, SSIM values are computed between two spectral bands at the same wavelength. Then, the average of these SSIM values is calculated and presented as SSIM.

B. Objective Evaluation

In Fig. 4, PSNR values which correspond to bpppb varying from 0 to 0.5 for three AVIRIS data sets are presented. Each data set in Fig. 4 is of size $512 \times 512 \times 224$ and can be considered as big HS data sets. It is obvious from Fig. 4 that, for each AVIRIS HS cube, PSNR increases while the bit rate or the number of iteration increases. Besides, EMPR+DHT+JPEG2000 combination yields consistently higher PSNR values than Plain EMPR for each HS cube. This is especially true for the low bit rates, which means less than 0.1 bpppb, where a gain of about 25 dB is achieved by performing EMPR+DHT+JPEG2000 combination rather than Plain EMPR. On the other hand, this gain decreases when using more iterations. At 0.5 bpppb value, PSNR values for Plain EMPR is about 83 dB, compared to 90 dB for EMPR+DHT+JPEG2000.

Fig. 5 depicts the comparison of distortions between the three AVIRIS data sets and their corresponding EMPR based compressed representations by using SSIM. As can be seen, SSIM values increase while the bit rates increases for all AVIRIS data sets for both Plain EMPR and EMPR+DHT+JPEG2000 when applied to the HS cube. In Fig. 5, SSIM values at initial iterations are under 0.5 which seem to be quiet low for Jasper Ridge and Low Altitude data sets. This lack of similarity can be fixed by increasing the number of iterations. Especially after the fourth iteration, SSIM values from the Plain EMPR compressions grows rapidly for each HS cube and approaches to 0.92 for Jasper Ridge and Lunar Lake, and about 0.89 for Low Altitude data set. Nonetheless, Plain EMPR cannot outperform EMPR+DHT+JPEG2000 combination for SSIM. SSIM values obtained by EMPR+DHT+JPEG2000 compression seem to be stable and vary between 0.92 and 0.99 for the bit rates changing from 0 to 0.5 for all AVIRIS data sets.

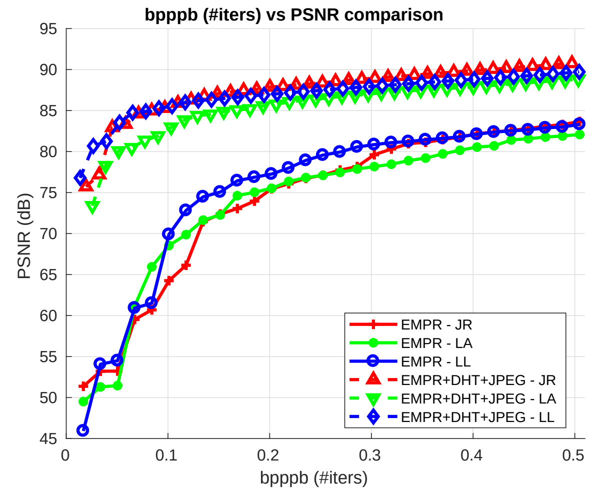


Fig. 4: bpppb vs PSNR results of EMPR and EMPR+DHT+JPEG2000 for Jasper Ridge, Low Altitude and Lunar Lake.

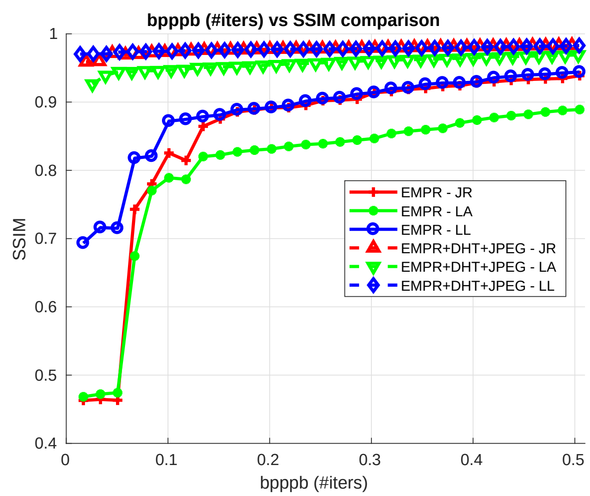


Fig. 5: bpppb vs SSIM results of EMPR and EMPR+DHT+JPEG2000 for Jasper Ridge, Low Altitude and Lunar Lake.

After analyzing the representation and similarity performance of EMPR and EMPR+DHT+JPEG2000 combination for three big AVIRIS HS cubes, the next issue is the comparison of the computation costs. In Fig. 6, it is obvious that the computation time for EMPR+DHT+JPEG2000 is less than Plain EMPR for each HS cube. The gap between the computation times of Plain EMPR and EMPR+DHT+JPEG2000 tends to increase. Since the size of each cube is the same, the measured computation times from different data sets are similar for both Plain EMPR and EMPR+DHT+JPEG2000 methods. Also, all computation times increase linearly since

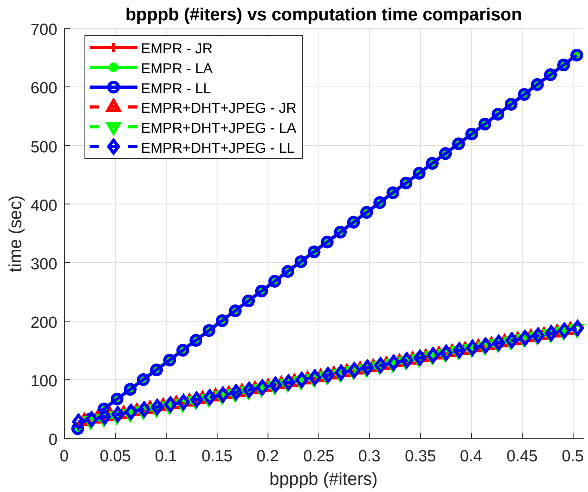


Fig. 6: bpppb vs computation time results of EMPR and EMPR+DHT+JPEG2000 for Jasper Ridge, Low Altitude and Lunar Lake.

the amount of computation is equal in each iteration. Although the computational complexity of the JPEG2000 procedure is higher than EMPR, it is implemented just once before starting the EMPR iterations. Additionally, the Iterative EMPR method is applied only to the high-frequency part of the HS cube, instead of the whole data set, which halves the computation cost. Moreover, DHT and Inverse DHT algorithms whose computational complexities are lower than EMPR and JPEG2000 in our case are also applied just a single time. All these assumptions are consistent with the results in Fig. 6 and show that the computation time of Plain EMPR is approximately 3.5 times higher than that of EMPR+DHT+JPEG2000 combination for the HS data sets of size $512 \times 512 \times 224$ under consideration.

In order to show the efficiency of the proposed method, it is important to compare the results obtained with other lossy compression algorithms. To this end, the proposed method will be compared with TFEMPR which is superior to several lossy compression algorithms such as CPPCA, SIP, gOMP, BP and LASSO as shown in [24]. In Table V, PSNR comparisons of EMPR+DHT+JPEG2000 combination with TFEMPR for the fixed bit rates 0.1, 0.3 and 0.5 are given.

TABLE V: Image Quality Performance (PSNR) Comparison of the Proposed EMPR Based Method with TFEMPR for Jasper Ridge, Low Altitude and Lunar Lake.

bpppb	Method	JR	LA	LL
0.1	TFEMPR	70.99	70.71	73.67
	EMPR+DHT+JPEG2000	85.29	82.97	85.54
0.3	TFEMPR	80.50	78.75	81.06
	EMPR+DHT+JPEG2000	88.95	87.00	87.92
0.5	TFEMPR	83.51	82.01	83.04
	EMPR+DHT+JPEG2000	90.75	88.82	89.71

It can be observed from Table V that EMPR+DHT+JPEG2000 outperforms TFEMPR for all fixed bit rates. The proposed approach yields higher PSNR values at each fixed bit-rate for all three AVIRIS HS cubes. The difference between the PSNR values of both methods is significant which is about 14 dB at 0.1 bpppb. This gap decreases for cases of 0.3 bpppb and 0.5 bpppb while the number of iterations grows.

C. Results from Other Data Sets

It is important to address the efficiency of EMPR+DHT+JPEG2000 combination using data sets collected with other sensors. To this end, Botswana data set from HYPERION sensor Pavia University data set from ROSIS sensor and Washington DC Mall data set from HYDICE sensor are employed. The results of the proposed method and TFEMPR are tabulated in Table VI and Table VII, respectively.

TABLE VI: MSE, MAE and PSNR Comparison of the Proposed EMPR Based Method with TFEMPR for Botswana.

bpppb	Method	MSE	MAE	PSNR
0.1	TFEMPR	838	1538	67.09
	EMPR+DHT+JPEG2000	411	816	70.19
0.3	TFEMPR	340	1280	71.01
	EMPR+DHT+JPEG2000	214	704	73.03
0.5	TFEMPR	160	576	74.29
	EMPR+DHT+JPEG2000	128	584	75.24

TABLE VII: MSE, MAE and PSNR Comparison of the Proposed EMPR Based Method with TFEMPR for Pavia Uni.

bpppb	Method	MSE	MAE	PSNR
0.1	TFEMPR	1036	5704	66.17
	EMPR+DHT+JPEG2000	334	3346	71.10
0.3	TFEMPR	227	1427	72.77
	EMPR+DHT+JPEG2000	100	2813	76.33
0.5	TFEMPR	75	636	77.56
	EMPR+DHT+JPEG2000	50	554	79.31

In Table VI and Table VII, the methods are compared according to Mean Squared Error (MSE), Maximum Absolute Error (MAE) and PSNR values. In Table VI, it is clear that EMPR+DHT+JPEG2000 combination yields lower MSE and higher PSNR values at each of three bpppb rates. On the other hand, MSE value for EMPR+DHT+JPEG2000 is a bit higher compared to TFEMPR at 0.5 bpppb while EMPR+DHT+JPEG2000 approach gives lower MAE at 0.1 and 0.3 bpppb. Similarly, in Table VII, EMPR+DHT+JPEG2000 approach results in lower MSE and higher PSNR values at 0.1, 0.3 and 0.5 bpppb for Pavia Uni. data set. MAE values of EMPR+DHT+JPEG2000 combination at 0.1 and 0.5 bpppb are also lower than Plain EMPR, but Plain EMPR gives lower MAE at 0.3 bpppb. Consequently, Table V, Table VI and Table VII show that EMPR+DHT+JPEG2000 combination outperforms TFEMPR in several objective assessments (MSE, MAE, PSNR) for the HS data sets of various sizes and acquired by different sensors.

PSNR values evaluated by EMPR+DHT+JPEG2000 implementation on Botswana, Pavia Uni. and Washington DC Mall data sets compared with Plain EMPR results are given in Fig. 7. Similar to Fig. 4, EMPR+DHT+JPEG2000 outperforms

Plain EMPR. It results in gains of approximately 15 dB and 28 dB at approximately 0.03 bpppb for Botswana and Pavia Uni., respectively while the gain is 26 dB at about 0.015 bpppb for Washington DC Mall. Another observation in Fig. 7 is that DHT and JPEG2000 contribution to EMPR results in greater improvement for two complex urban images Pavia Uni. and Washington DC Mall than the Botswana data set.

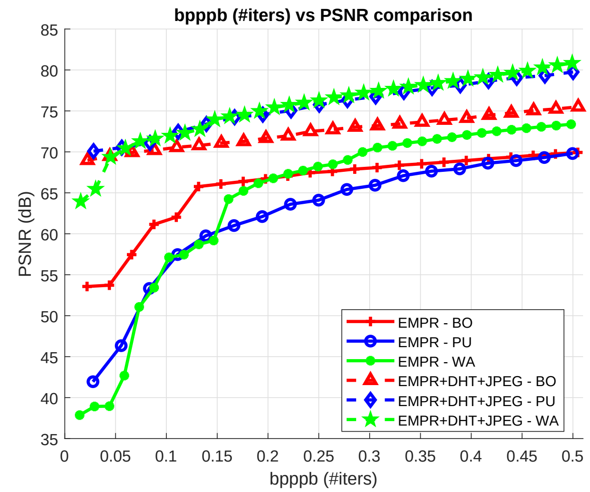


Fig. 7: bpppb vs PSNR results of EMPR and EMPR+DHT+JPEG2000 for Botswana, Pavia Uni. and Washington DC Mall

After comparing the proposed method and TFEMPR, it is time to address the similarity characteristics of EMPR and EMPR+DHT+JPEG2000 in detail via Fig. 7 and Fig. 8. The findings in these two figures are similar to those presented in Fig. 4 and Fig. 5. In Fig. 7, EMPR+DHT+JPEG2000 combination yields higher PSNR values than Plain EMPR for all HS images under consideration.

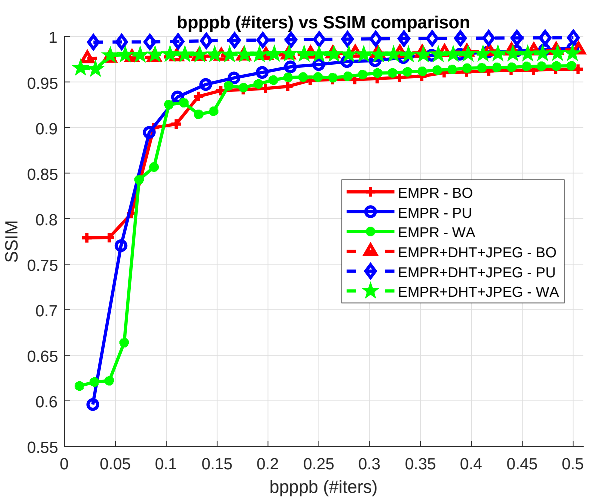


Fig. 8: bpppb vs SSIM results of EMPR and EMPR+DHT+JPEG2000 for Botswana, Pavia Uni. and Washington DC Mall

In Fig. 8, SSIM values for Botswana, Pavia Uni. and Washington DC Mall for bit rates changing from 0.015 to 0.5 bpppb are compared. The findings in Fig. 8 are consistent with those in Fig. 5. SSIM values for Plain EMPR at initial iterations or low bit rates are far smaller than the proposed approach.

TABLE VIII: SNR (dB) and time (sec) comparison of the proposed EMPR based method with PCA+JPEG2000 and 3D SPECK for Cuprite, Indian Pines and Jasper Ridge at 0.5 bpppb.

Method / Data	CU		IP		JR	
	SNR	Time	SNR	Time	SNR	Time
PCA+JPEG2000	50.5	7.2	40.2	0.4	44.1	6.9
3D SPECK	46.6	4.8	43.5	0.3	41.7	4.5
Proposed	53.0	138.3	42.6	4.3	53.3	184.6

After analyzing and comparing the compression efficiency of the proposed method with TFEMPR via several experiments, we also compared it with some other well-known and efficient lossy compression algorithms exploiting various HS data sets. The first is PCA+JPEG2000 approach presented in [27] while the second is called the 3D SPECK given in [48]. The performance in terms of SNR of our proposed EMPR+DHT+JPEG2000 approach is compared with these two techniques in Table VIII. Table VIII denotes that EMPR+DHT+JPEG2000 approach outperforms both PCA+JPEG2000 and 3D SPECK methods at the given fixed bit rate of 0.5 bpppb for the Cuprite and Jasper Ridge data sets. In Table VIII, it can be noticed that the 3D SPECK algorithm preserves the details for Indian Pines data set better than the proposed method, in which the SNR value is slightly higher which is 0.9 dB. On the other hand, EMPR+DHT+JPEG2000 approach manages to present 2.4 dB higher SNR value than PCA+JPEG2000 method for Indian Pines HS image.

Beside the rate-distortion values, related computation times are also presented in Table VIII. It is obvious from Table VIII that EMPR+DHT+JPEG2000 approach yields higher computation times due to its number of iterations. To be able to enhance the detail preserving ability of the proposed approach, the number of iterations should be increased, while the computation time grows linearly (Fig. 6). Also, the gap between computation times decreases when the size of the data set is relevantly small.

TABLE IX: PSNR (dB) comparison of the EMPR based methods with two tensor based compression methods for Indian Pines dataset at very close bit-rates (bpppb).

Method	**PSNR**	**Rate**
PLTD	36.67	0.0454
NTSRLR	41.15	0.05
EMPR	46.44	0.048
EMPR+DHT+JPEG2000	60.19	0.049

Although the computational complexity of the proposed method is quite high, the evaluation of EMPR components requires a large number of independent multiplications. This problem can be fixed by using parallel programming concepts and the overall computation time can be drastically reduced.

EMPR is a tensor decomposition based method which can be efficiently utilized in HSI compression. Then, it is important to compare it with other state-of-the-art tensor decomposition based methods. To this end, two important methods are selected and PSNR results are presented in Table IX. These methods are PLTD [36] and NTSRLR [49], respectively. In Table tab:tensorbasedcomp, the rate distortion metric, PSNR, is addressed at very low and close bit rates about 0.05. The values for PLTD and NTSRLR are acquired from [36] and [49], respectively. It is clear from Table tab:tensorbasedcomp that the proposed method outperforms the other tensor decomposition based techniques in the sense of preserving the details. Moreover, EMPR without combining JPEG2000 also yields higher PSNR values, which verifies that EMPR, alone, stands as an efficient tensor-based lossy compression technique for HSI.

As a last comment in this subsection, it is worth noting that in [56], the compression ratio produced by applying 2D JPEG2000 to each band of the AVIRIS data set consecutively is 2.15 at most which is far less than our proposed method regarding to the compression ratios at fixed bpppb values.

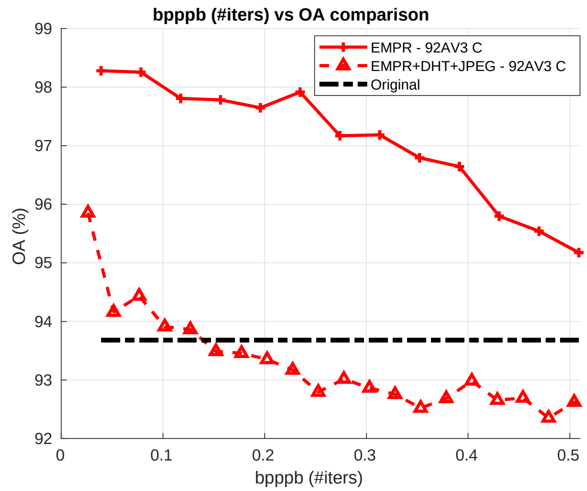


Fig. 10: Comparison of the classification accuracies of EMPR, EMPR+DHT+JPEG2000 and original image for 92AV3 C with available ground truth using 50% of all data as the training pixels

D. Classification and Overall Accuracy

In this subsection, the quality assured assessment, which is the classification performance of our proposed methods is evaluated. To this end, 92AV3 C, PaviaUni and Salinas B are employed. The first reason for selecting these data sets is that we have their ground truth images for evaluation. The second reason is to compare the classification ability of EMPR with an efficient lossy compression method based on 3D Discrete Cosine Transform (3D DCT) [12]. To this end, the HS data sets and their corresponding ground truth images are cropped. Thus, the sizes of 92AV3 C, Pavia Uni. and Salinas B are reduced to $144 \times 144 \times 184$, $144 \times 144 \times 96$ and $144 \times 144 \times 184$ respectively while the dimensions of their ground truth images are reduced accordingly. These reductions of sizes were adopted aiming for a fair comparison with the 3D DCT method [12], in which the same setting was used in coding of hyperspectral images. The corresponding ground truth images are shown in Fig. 9.

(a) (b) (c)

Fig. 9: Ground truth images for the data sets: (a) 92AV3 C, (b) Pavia Uni. and (c) Salinas B.

The abovementioned HS data sets are classified using a support vector machine (SVM) algorithm with the help of an important open source SVM library called LIBSVM [57]. First, the encoded bands of $\mathbf{H}_{low}$ are decoded consecutively and the corresponding half cube of size $n_1 \times n_2 \times n_3/2$ is obtained. Then, the second half cube of the same size, which is encoded by EMPR (or iterative EMPR) is constructed using the EMPR components and the support vectors by following Eq. (7). Two reconstructed half cubes are reunited by applying the inverse DHT to each spectral signal of these two half cubes whose pixel indexes match. After attaining the corresponding lossy compressed version of the original data, the spectral signals are extracted and used as feature vectors for training or testing on the SVM for data classification.

The results obtained are reported as Overall Accuracy (OA) for each HS cube. The presented OA values are the average of 10 independent SVM experiments for each data set. In each experiment, 50% of the data is selected randomly for training and the other 50% for testing. For each data set, the optimal classification parameters are determined by grid-search and cross-validation procedures after data normalization. Then the model is trained using an SVM algorithm with the radial basis functions (RBF) as the classification kernel and a testing phase is applied. The same classification procedure is also employed for the original HS cubes in order to compare the classification results.

In Fig. 10, the classification results of Plain EMPR, EMPR+DHT+JPEG2000 combination and the original HS cube for 92AV3 C data set are reported. The OA value for original 92AV3 C is calculated as 93.68% after 10 random experiments. On the other hand, Plain EMPR (no iteration) result is above 98%. After the first iteration, the corresponding classification accuracy starts to decrease. Nonetheless, OA values for Iterative EMPR are always higher than the classification accuracy of the original data up to the 12th iteration which corresponds to approximately 0.5 bpppb. Following this, the EMPR+DHT+JPEG2000 combination produces about 96% accuracy which is again higher than the classification accuracy of original 92AV3 C. The corresponding accuracy tends to decrease while the number of iterations grows, similar to Plain EMPR. Nevertheless, after the 4th iteration, which corresponds to 0.16 bpppb, the accuracy drops below that of the original HS cube. On the other hand, in paper [12], it is reported that the 3D DCT method yields about 45% OA at 0.02 bpppb and the classification accuracy increases strictly whilst the bit rate (bpppb) grows, though it always stays below the accuracy of the original HS cube up to 1.0 bpppb. These results show that Plain EMPR outperforms the 3D DCT, especially at very low bpppb values.

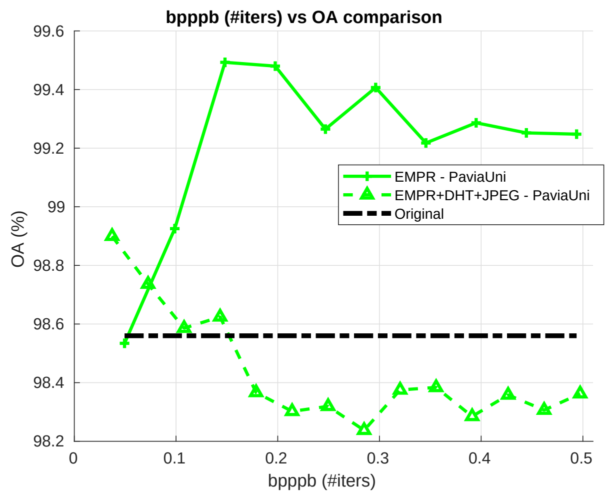


Fig. 11: Comparison of the classification accuracies of EMPR, EMPR+DHT+JPEG2000 and original image for Pavia Uni. with available ground truth using 50% of all data as the training pixels

In Fig. 11, the classification results for Pavia Uni. data set are reported. It is seen that Plain EMPR with no iteration yields a slightly lower classification accuracy than the original data. But on applying the first iteration, it rises above the OA of the original data and stays above for the bpppb values from 0.08 to 0.5. On the other hand, EMPR+DHT+JPEG2000 approach gives higher OA values than the original classification accuracy up to the third iteration, but then it tends to decrease ans stays below the OA of the original data. In [12], it is reported that the 3D DCT based method achieves at most 98.56% for the corresponding cropped version of Pavia Uni. data set which is lower than that of the original OA for this HS image. Therefore, it is convenient to say that Plain EMPR outperforms the 3D DCT method regarding to the classification accuracy for Pavia Uni. data set at each bit rate up to 0.5 bpppb, while EMPR+DHT+JPEG2000 approach yields better results than the 3D DCT at low bit rates for the same HS data set.

In Fig. 12, the classification accuracy results for Salinas B are presented.It is clear from the figure that both Plain EMPR and the EMPR+DHT+JPEG2000 combination yield higher classification accuracy than the corresponding original data set. According to the results in Fig. 12, EMPR outperforms EMPR+DHT+JPEG2000 combination and the 3D DCT method which is reported as 99.02% at most in [12] regarding classification for bpppb values from 0.03 bpppb to 0.5 bpppb.

After comparing the classification performances of EMPR based methods and the 3D DCT based lossy compression algorithm, the next step is to compare their representation

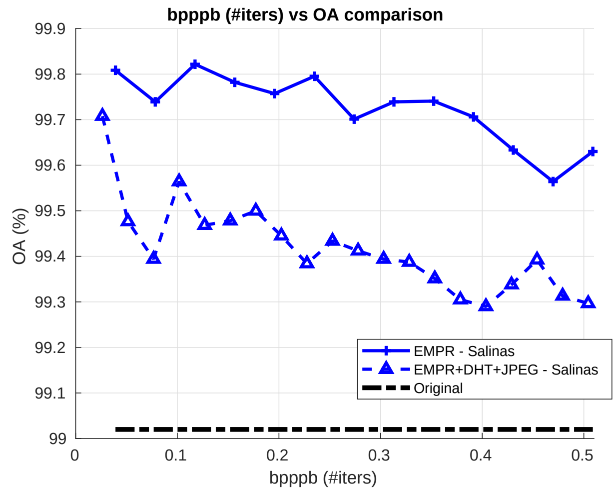


Fig. 12: Comparison of the classification accuracies of EMPR, EMPR+DHT+JPEG2000 and original image for Salinas with available ground truth using 50% of all data as the training pixels

efficiencies. To this end, it is important to discuss the objective assessments as well as classification. In Fig. 13, one can see the SNR results for three data sets obtained by employing EMPR and EMPR+DHT+JPEG2000 combination. It is obvious that EMPR+DHT+JPEG2000 combination results in higher SNR values than Plain EMPR for all data sets. All the

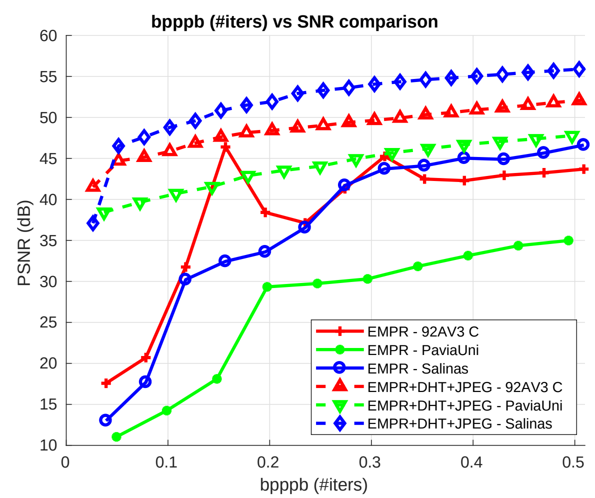


Fig. 13: bpppb vs SNR results of EMPR and EMPR+DHT+JPEG2000 for 92AV3 C, Pavia Uni. and Salinas B.

curves in Fig. 13 tend to increase except the one for Plain EMPR of 92AV3 C. SNR values for 92AV3 C increase up to the 3rd iteration and then decrease. After the 5th iteration, they increase again then slightly reduce after the 7th iteration. On the other hand, though its performance regarding SNR is lower than EMPR+DHT+JPEG2000 combination, Plain EMPR can outperform the 3D DCT based method at bpppb values less than 0.5. In [12], it is reported that SNR values for all three data sets tend to increase as the bit rate grows where SNR values at 0.5 bpppb for 92AV3 C, Pavia Uni. and Salinas B in that work are about 30 dB, 29 dB and 31 dB respectively. It is apparent from Fig. 13 that Plain EMPR outperforms the 3D DCT method for each data set under consideration in terms of SNR.

Fig. 14 shows SSIM results at bpppb values changing from 0.03 to 0.5 obtained by employing Plain EMPR and EMPR+DHT+JPEG2000 approaches to three data sets. The results are similar to Fig. 5 and Fig. 8. At initial iterations, Plain EMPR has low values of SSIM. After applying the second iteration, SSIM value increases rapidly for each HS data set while the number of iterations grows. For each data set, EMPR is capable of achieving similarity ratios greater than 0.9 after 0.3 bpppb. Additionally, SSIM values obtained by the EMPR+DHT+JPEG2000 approach are always greater than EMPR. If we compare the performance of the proposed methods with the 3D DCT lossy compression algorithm in [12] regarding structural similarity, it is seen that both EMPR and EMPR+DHT+JPEG2000 approaches are more capable of compressing HS data by preserving the similarity than the 3D DCT method at lower bit rates. [12] reports that the SSIM

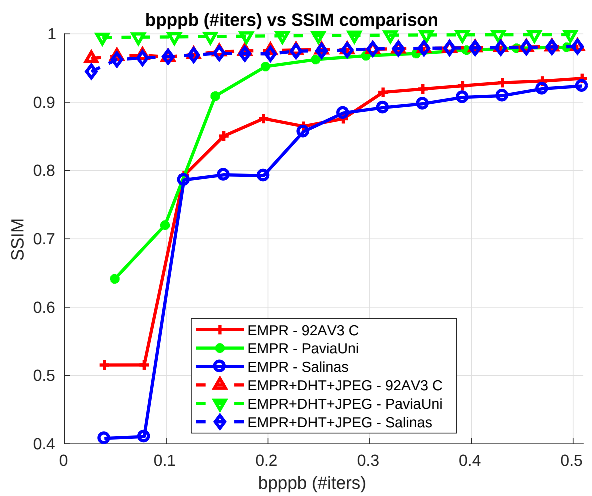


Fig. 14: bpppb vs SSIM results of EMPR and EMPR+DHT+JPEG2000 for 92AV3 C, Pavia Uni. and Salinas B.

values at lower bit rates are less than 0.2 for each of the corresponding HS data set. On the other hand, the 3D DCT based method manages to increase the similarity while the compression rate grows. At 0.5 bpppb, the 3D DCT based method generates approximately 0.9 SSIM value. Thus, it is convenient to say, EMPR based methods are much more efficient than the 3D DCT at lower bit rates in terms of similarity, though both EMPR+DHT+JPEG2000 and the 3D DCT approaches result in similar SSIM values at higher bit rates, which are 0.5 bpppb and further.

Before concluding this subsection, it is rational to analyze the classification efficiency of the EMPR based methods using a lower training ratio, say 10%, which is a more reasonable training ratio for many applications in HS imagery. To this end, the same HS data sets which are 92AV3 C, Pavia Uni. and Salinas are taken into consideration again. The SVM classification method whose tuning details are given above are also utilized in order to calculate the corresponding classification accuracy of each data set.

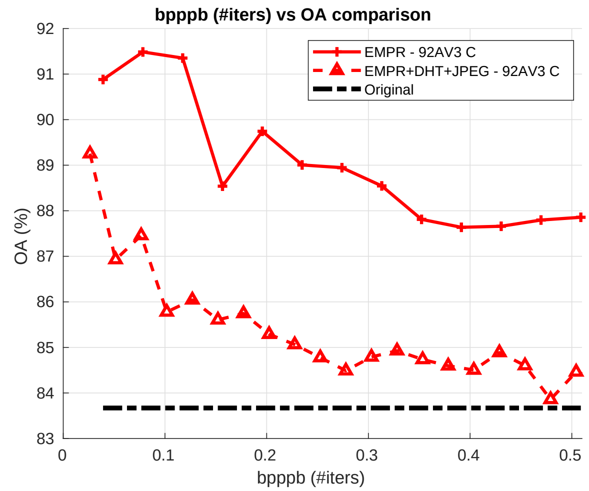


Fig. 15: Comparison of the classification accuracies of EMPR, EMPR+DHT+JPEG2000 and original image for 92AV3 C with available ground truth using 10% of all data as the training pixels

In Fig. 15, the OA values of Plain EMPR, EMPR+DHT+JPEG2000 approach and the OA value for the original data for 92AV3 C HS cube using 10% as the training pixels are presented. One can verify that Plain EMPR yields higher accuracy ratios than the original data at each bit rate. Moreover, Plain EMPR outperforms EMPR+DHT+JPEG2000 method regarding OA. Plain EMPR achieves its maximum OA at the first iteration which corresponds to 0.08 bpppb.

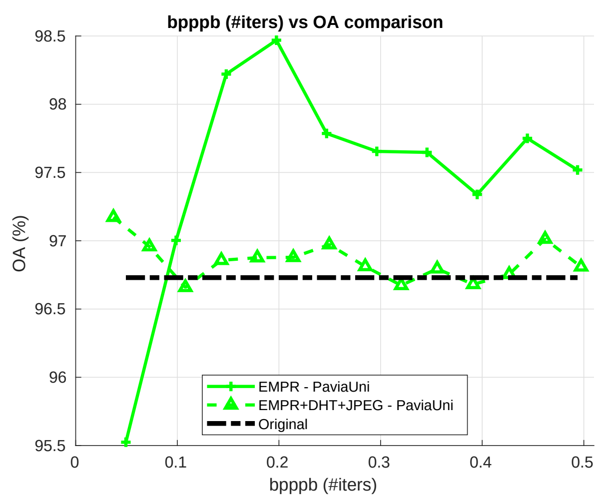


Fig. 16: Comparison of the classification accuracies of EMPR, EMPR+DHT+JPEG2000 and original image for Pavia Uni. with available ground truth using 10% of all data as the training pixels

In Fig. 16, the classification results for Pavia Uni. are reported for 10% trainig ratio. The results in Fig. 16 are similar to those in Fig. 11. Again, Plain EMPR gives higher classification accuracy than the original data set for the bpppb values changing from 0.1 to 0.5 and outperforms EMPR+DHT+JPEG2000 method for all iterations except the initial one, that is the zeroth iteration or no iteration.

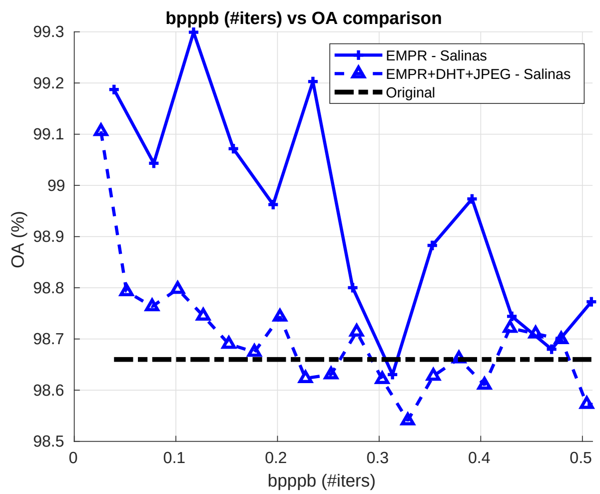


Fig. 17: Comparison of the classification accuracies of EMPR, EMPR+DHT+JPEG2000 and original image for Salinas with available ground truth using 10% of all data as the training pixels

In Fig. 17, the related OA results are presented for Salinas data set by amploying 10% of the all pixels as the training pixels. It is shown that, Plain EMPR gives higher OA values than the original data at each iteration except the seventh one. On the other hand, the overall accuracies yielded by both Plain EMPR and EMPR+DHT+JPEG2000 approaches tend to decrease as the number of iteration grows with an oscillatory behaviour. Also, it is clear from Fig. 17 that, Plain EMPR outperforms EMPR+DHT+JPEG2000 regarding OA for Salinas data set at 10% training ratio.

E. Spectral Analysis

In classification of HS images, investigating the spectral similarity right along side the structural similarity is also another important issue. To this end, the spectral angle (SA) and the spectral correlation coefficient (SCC) metrics whose formulations are given in (23) will be exploited.

$$
\begin{aligned}
SA(\mathbf{x}, \hat{\mathbf{x}}) &= \arccos\left(\frac{\langle \mathbf{x}, \hat{\mathbf{x}}\rangle}{\|\mathbf{x}\|_2 \, \|\hat{\mathbf{x}}\|_2}\right) \\
SCC(\mathbf{x}, \hat{\mathbf{x}}) &= \frac{\langle \mathbf{x}-\overline{\mathbf{x}}, \hat{\mathbf{x}}-\overline{\hat{\mathbf{x}}}\rangle}{\|\mathbf{x}-\overline{\mathbf{x}}\|_2 \, \|\hat{\mathbf{x}}-\overline{\hat{\mathbf{x}}}\|_2}
\end{aligned} \tag{23}
$$

In HS imagery, SA is utilized for measuring the spectral similarity of the processed and the reference image, while SCC is benefited to distinguish between positive and negative correlations amongst the spectral signals.

In Fig. 18 and Fig. 19, average SA and SCC values for increasing bit rates (number of iterations) for 92AV3 C, Pavia Uni. and Salinas B data sets are presented, respectively. In both figures, each value is calculated as the mean value of the corresponding metric determined for the all pixels of the corresponding image.

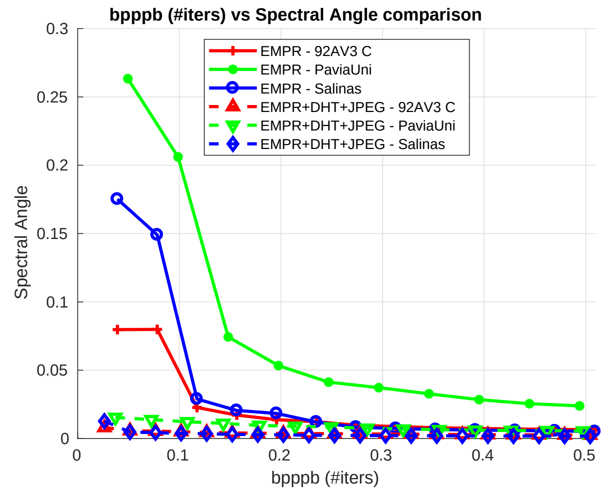


Fig. 18: Spectral angle comparison of the EMPR based methods

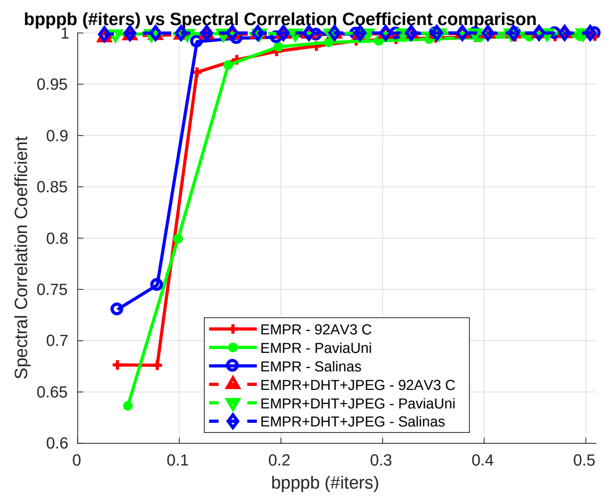


Fig. 19: Spectral correlation coefficient comparison of the EMPR based methods

It is shown in Fig. 18 that the SA values obtained by performing EMPR+DHT+JPEG2000 are higher than the ones obtained by the EMPR application to the corresponding data sets, especially for the bit rates up to 0.25 bpppb. The observations indicate that EMPR+DHT+JPEG2000 approach preserves the spectral similarity better than the Plain EMPR for the corresponding data sets. On the other hand, SA values for each method and the data set tend to decrease while the number of iterations grows.

In Fig. 19, it is clear that the spectral correlations for each data set achieved by performing EMPR is low at initial iterations. But, after the first iterations, all coefficients increase rapidly and catches the EMPR+DHT+JPEG2000 after 0.3 bpppb. Beside, EMPR+DHT+JPEG lossy compression approach preserves the spectral correlation since the corresponding values in Fig. 19 is always close to 1.0 for all bit rates.

F. On Parameter Selection

As we discussed in Section II, the proposed method is a hybrid approach combining (Iterative) EMPR and JPEG2000. EMPR, alone itself, is a promising method for the HSI compression but has limited ability for preserving the image details. This issue can be overcome by taking the residual term into account, which means performing the EMPR on each residual iteratively. Thus, an efficient method called Iterative EMPR emerges. It is clear to observe from Fig. 4 to Fig. 8 that increasing the iteration count increases the corresponding PSNR and SSIM values consistently for all HS data sets. Moreover, by adding new residuals to the expansion, meaning that performing more iterations also improves the spectral similarity as one can observe from Fig. 18 and Fig. 19. Besides, each iteration comes with an additional computation cost and this cost can be considerably high for the HS data sets whose sizes are quite large.

On the other hand, in contrast with the preserving the details and spectral properties ability of Iterative EMPR, the increment in iteration count reduces the classification accuracy. EMPR depends on the weighted averages of the HS cube and somehow denoises the raw data using these averages which helps to improve the classification accuracy. Performing additional iterations may cause the unenviable details which can be considered as the *noise*. This phenomenon results in reduced classification ability which is addressed through Fig. 14, Fig. 15, Fig. 16 and Fig. 17.

As a result, while determining the value of the iteration number parameter, one should consider a trade-off between the compression quality and computation time. Also, it is important to remark that a few iterations may not increase the computation cost drastically, while they can represent a satisfactory amount of details.

V. Conclusion

A new and easy to implement high dimensional data modelling method namely Enhanced Multivariance Products Representation (EMPR) is proposed and utilized as an highly efficient tool for the lossy compression of HS images. An Iterative EMPR scheme is also proposed and its efficiency on lossy compression is presented via several implementations, which is compared with some state-of-the-art lossy compression algorithms.

The proposed EMPR based approaches are further supported with the decorrelation ability of the Haar transform (DHT) generating two subband data sets. The high-subband part which has a higher correlation is decomposed using the EMPR approach while the low-subband part is compressed using the JPEG2000 scheme in a lossless manner. Consequently, the overall approach yields superior PSNR values at the fixed bit-rates, when compared with another state-of-the-art high dimensional modelling based recursive lossy compression algorithm, TFEMPR and existing methods, PCA+JPEG2000 approach and 3D SPECK algorithm. The proposed approach also outperforms two state-of-the-art tensor decomposition based HSI compression methods which are PLTD and NTSRLR, respectively.

Combining EMPR with DHT and JPEG2000 also decreases the computational complexity for sufficiently large HS data sets. Results indicate that the proposed approach is especially suitable for the lossy compression of HS data at low bit rates.

Another noteworthy issue is that increasing the iteration number in Iterative EMPR approach improves the constructed image quality while the data to be stored increases.

In order to assess the classification capability of the proposed method, we employed an SVM procedure for several HS data sets using their available ground truth maps. According to observed classification accuracy results, EMPR has a greater capability of classifying HS images at low bit-rates. Also, it outperforms one of the important lossy compression algorithms based on the 3D DCT. This power of EMPR is presented to the scientific literature for the first time. These results show that EMPR is an efficient method for feature extraction in HSI. On the other hand, it is observed that, combining EMPR with DHT and JPEG2000 reduces classification accuracy, though it increases the representation quality. However, another observation shows that increasing the iteration number in Iterative EMPR reduces the overall accuracy.

In summary, the main contributions of this paper can be highlighted as follows:

- An iterative EMPR scheme is proposed and applied to HSI compression for the first time, which can improve the bit-rate and preserve more details for effective compression of HSI;
- Combining the iterative EMPR with JPEG2000 and DHT has enhanced the representation quality of the corresponding compression scheme;
- The proposed approach has outperformed several state-of-the-art lossy compression methods;
- The feature extraction capability of EMPR has been addressed and validated via the accuracy of data classification of SVM in HSI for the first in the scientific literature.

References

[1] H. Huang, L. Liu, and M. O. Ngadi, "Recent developments in hyperspectral imaging for assessment of food quality and safety," *Sensors*, vol. 14, no. 4, pp. 7248–7276, 2014.

[2] A. Gowen, C. O'Donnell, P. Cullen, G. Downey, and J. Frias, "Hyperspectral imaging–an emerging process analytical tool for food quality and safety control," *Trends in Food Science & Technology*, vol. 18, no. 12, pp. 590–598, 2007.

[3] A. Polak, T. Kelman, P. Murray, S. Marshall, D. J. Stothard, N. Eastaugh, and F. Eastaugh, "Hyperspectral imaging combined with data classification techniques as an aid for artwork authentication," *Journal of Cultural Heritage*, vol. 26, pp. 1–11, 2017.

[4] O. Y. Rodionova, L. P. Houmøller, A. L. Pomerantsev, P. Geladi, J. Burger, V. L. Dorofeyev, and A. P. Arzamastsev, "NIR spectrometry for counterfeit drug detection: a feasibility study," *Analytica Chimica Acta*, vol. 549, no. 1-2, pp. 151–158, 2005.

[5] S. S. M. Noor, K. Michael, S. Marshall, and J. Ren, "Hyperspectral image enhancement and mixture deep-learning classification of corneal epithelium injuries," *Sensors*, vol. 17, no. 11, p. 2644, 2017.

[6] K. Michael, S. Siti Salwa Binti Md Noor, J. Tschannerl, J. Ren, and S. Marshall, "The properties of the cornea based on hyperspectral imaging," *Investigative Ophthalmology and Visual Science*, vol. 58, no. 8, 2017.

[7] H. Ogihara, Y. Hamamoto, Y. Fujita, A. Goto, J. Nishikawa, and I. Sakaida, "Development of a gastric cancer diagnostic support system with a pattern recognition method using a hyperspectral camera," *Journal of Sensors*, vol. 2016, 2016.

[8] D. Manolakis, R. Lockwood, and T. Cooley, *Hyperspectral Imaging Remote Sensing: Physics, Sensors, and Algorithms*. Cambridge University Press, 2016.

[9] T. Qiao, J. Ren, Z. Wang, J. Zabalza, M. Sun, H. Zhao, S. Li, J. A. Benediktsson, Q. Dai, and S. Marshall, "Effective denoising and classification of hyperspectral images using curvelet transform and singular spectrum analysis," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 1, pp. 119–133, 2017.

[10] L. Mou, P. Ghamisi, and X. X. Zhu, "Deep recurrent neural networks for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 7, pp. 3639–3655, 2017.

[11] L. Fang, N. He, S. Li, A. J. Plaza, and J. Plaza, "A new spatial-spectral feature extraction method for hyperspectral images using local covariance matrix representation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. PP, no. 99, pp. 1–13, 2018.

[12] T. Qiao, J. Ren, M. Sun, J. Zheng, and S. Marshall, "Effective compression of hyperspectral imagery using an improved 3D DCT approach for land-cover analysis in remote-sensing applications," *International journal of remote sensing*, vol. 35, no. 20, pp. 7316–7337, 2014.

[13] K. Makantasis, K. Karantzalos, A. Doulamis, and N. Doulamis, "Deep supervised learning for hyperspectral data classification through convolutional neural networks," in *Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International*. IEEE, 2015, pp. 4959–4962.

[14] S. Li, Z. Zheng, Y. Wang, C. Chang, and Y. Yu, "A new hyperspectral band selection and classification framework based on combining multiple classifiers," *Pattern Recognition Letters*, vol. 83, pp. 152–159, 2016.

[15] A. Karami, R. Heylen, and P. Scheunders, "Hyperspectral image compression optimized for spectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 10, pp. 5884–5894, 2016.

[16] K. S. Babu, V. Ramachandran, K. Thyagharajan, and G. Santhosh, "Hyperspectral image compression algorithms–a review," in *Artificial Intelligence and Evolutionary Algorithms in Engineering Systems*. Springer, 2015, pp. 127–138.

[17] I. Ülkü and B. U. Töreyin, "Sparse representations for online-learning-based hyperspectral image compression," *Appl. Opt.*, vol. 54, no. 29, pp. 8625–8631, Oct 2015. [Online]. Available: http://ao.osa.org/abstract.cfm?URI=ao-54-29-8625

[18] J. E. Fowler, "Compressive-projection principal component analysis," *IEEE Transactions on Image Processing*, vol. 18, no. 10, pp. 2230–2242, 2009.

[19] J. Wang, S. Kwon, and B. Shim, "Generalized orthogonal matching pursuit," *IEEE Transactions on signal processing*, vol. 60, no. 12, pp. 6202–6216, 2012.

[20] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale ell_1-regularized least squares," *IEEE journal of selected topics in signal processing*, vol. 1, no. 4, pp. 606–617, 2007.

[21] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[22] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2346–2356, 2008.

[23] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM review*, vol. 43, no. 1, pp. 129–159, 2001.

[24] Z. Gündoğar, B. U. Töreyin, and M. Demiralp, "Tridiagonal folmat enhanced multivariance products representation based hyperspectral data compression," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 9, pp. 3272–3278, 2018.

[25] A. Karami, M. Yazdi, and G. Mercier, "Compression of hyperspectral images using discrete wavelet transform and tucker decomposition," *IEEE journal of selected topics in applied earth observations and remote sensing*, vol. 5, no. 2, pp. 444–450, 2012.

[26] W. Fu, S. Li, L. Fang, and J. A. Benediktsson, "Adaptive spectral–spatial compression of hyperspectral image with sparse representation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 2, pp. 671–682, 2016.

[27] Q. Du and J. E. Fowler, "Hyperspectral image compression using JPEG2000 and principal component analysis," *IEEE Geoscience and Remote sensing letters*, vol. 4, no. 2, pp. 201–205, 2007.

[28] K.-j. Cheng and J. Dill, "Lossless to lossy dual-tree bezw compression for hyperspectral images," *IEEE transactions on geoscience and remote sensing*, vol. 52, no. 9, pp. 5765–5770, 2014.

[29] C. C. for Space Data Systems (CCSDS), "Low-complexity lossless and near-lossless multispectral and hyperspectral image compression," 2019.

[30] S. Mei, B. M. Khan, Y. Zhang, and Q. Du, "Low-complexity hyperspectral image compression using folded PCA and JPEG2000," in *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2018, pp. 4756–4759.

[31] J. T. Rucker, J. E. Fowler, and N. H. Younan, "JPEG2000 coding strategies for hyperspectral data," in *Proceedings. 2005 IEEE International Geoscience and Remote Sensing Symposium, 2005. IGARSS'05.*, vol. 1. IEEE, 2005, pp. 4–pp.

[32] B. Penna, T. Tillo, E. Magli, and G. Olmo, "Progressive 3-d coding of hyperspectral images based on JPEG2000," *IEEE Geoscience and remote sensing letters*, vol. 3, no. 1, pp. 125–129, 2006.

[33] E. Christophe, C. Mailhes, and P. Duhamel, "Hyperspectral image compression: adapting spiht and ezw to anisotropic 3-d wavelet coding," *IEEE Transactions on Image Processing*, vol. 17, no. 12, pp. 2334–2346, 2008.

[34] W. Jifara, F. Jiang, B. Zhang, H. Wang, J. Li, A. Grigorev, and S. Liu, "Hyperspectral image compression based on online learning spectral features dictionary," *Multimedia Tools and Applications*, vol. 76, no. 23, pp. 25 003–25 014, 2017.

[35] A. Sukhanov, S. Tuna, and B. U. Töreyin, "Lossy compression of hyperspectral images by using enhanced multivariance products representation (empr) method," in *Signal Processing and Communication Application Conference (SIU), 2016 24th*. IEEE, 2016, pp. 1925–1928.

[36] B. Du, M. Zhang, L. Zhang, R. Hu, and D. Tao, "PLTD: Patch-based low-rank tensor decomposition for hyperspectral images," *IEEE Transactions on Multimedia*, vol. 19, no. 1, pp. 67–79, 2016.

[37] S. Yang, M. Wang, P. Li, L. Jin, B. Wu, and L. Jiao, "Compressive hyperspectral imaging via sparse tensor and nonlinear compressed sensing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 11, pp. 5943–5957, 2015.

[38] D. Valsesia and E. Magli, "Fast and lightweight rate control for onboard predictive coding of hyperspectral images," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 3, pp. 394–398, 2017.

[39] S. Álvarez-Cortés, N. Amrani, M. Hernández-Cabronero, and J. Serra-Sagristà, "Progressive lossy-to-lossless coding of hyperspectral images through regression wavelet analysis," *International Journal of Remote Sensing*, vol. 39, no. 7, pp. 2001–2021, 2018.

[40] D. Valsesia and E. Magli, "High-throughput onboard hyperspectral image compression with ground-based cnn reconstruction," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 12, pp. 9544–9553, 2019.

[41] B. Tunga and M. Demiralp, "The influence of the support functions on the quality of enhanced multivariance product representation," *Journal of Mathematical Chemistry*, vol. 48, no. 3, pp. 827–840, Oct 2010. [Online]. Available: https://doi.org/10.1007/s10910-010-9714-2

[42] M. A. Tunga and M. Demiralp, "A novel method for multivariate data modelling: Piecewise generalized EMPR," *Journal of Mathematical Chemistry*, vol. 51, no. 10, pp. 2654–2667, Nov 2013. [Online]. Available: https://doi.org/10.1007/s10910-013-0228-6

[43] E. K. Özay and M. Demiralp, "Weighted tridiagonal matrix enhanced multivariance products representation (WTMEMPR) for decomposition of multiway arrays: applications on certain chemical system data sets," *Journal of Mathematical Chemistry*, vol. 55, no. 2, pp. 455–476, Feb 2017. [Online]. Available: https://doi.org/10.1007/s10910-016-0687-7

[44] S. Tuna and B. Tunga, "A novel piecewise multivariate function approximation method via universal matrix representation," *Journal of Mathematical Chemistry*, vol. 51, no. 7, pp. 1784–1801, Aug 2013. [Online]. Available: https://doi.org/10.1007/s10910-013-0179-y

[45] B. Tunga and M. Demiralp, "An iterative scheme for enhanced multivariance product representation method," in *Proceedings for the 1st IEEEAM Conference on Applied Computer Science (ACS)*, 2010, pp. 247–255.

[46] A. Haar, "Zur theorie der orthogonalen funktionensysteme," *Mathematische Annalen*, vol. 69, no. 3, pp. 331–371, Sep 1910. [Online]. Available: https://doi.org/10.1007/BF01456326

[47] Z. Gündoğar and M. Demiralp, "Formulation of tridiagonal folmat enhanced multivariance products representation (tfempr)," in *AIP Conference Proceedings*, vol. 1702, no. 1. AIP Publishing, 2015, p. 170005.

[48] G. Motta, F. Rizzo, and J. A. Storer, *Hyperspectral data compression*. Springer Science & Business Media, 2006.

[49] J. Xue, Y. Zhao, W. Liao, and J. C.-W. Chan, "Nonlocal tensor sparse representation and low-rank regularization for hyperspectral image compressive sensing reconstruction," *Remote Sensing*, vol. 11, no. 2, p. 193, 2019.

[50] J. H. Friedman, “On bias, variance, 0/1–loss, and the curse-of-dimensionality,” *Data mining and knowledge discovery*, vol. 1, no. 1, pp. 55–77, 1997.
[51] A. Cichocki, R. Zdunek, A. H. Phan, and S.-i. Amari, *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*. John Wiley & Sons, 2009.
[52] S. Tuna and M. Demiralp, “Bivariate enhanced multivariance products representation (empr) at zero volume limit via geometric separation,” in *AIP Conference Proceedings*, vol. 1702, no. 1. AIP Publishing, 2015, p. 170009.
[53] P. T. Chiou, Y. Sun, and G. Young, “A complexity analysis of the JPEG image compression algorithm,” in *2017 9th Computer Science and Electronic Engineering (CEEC)*. IEEE, 2017, pp. 65–70.
[54] J. A. R. Macias and A. G. Exposito, “Efficient computation of the running discrete haar transform,” *IEEE transactions on power delivery*, vol. 21, no. 1, pp. 504–505, 2006.
[55] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli *et al.*, “Image quality assessment: from error visibility to structural similarity,” *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.
[56] B. U. Töreyin, O. Yilmaz, and Y. M. Mert, “Evaluation of on-board integer wavelet transform based spectral decorrelation schemes for lossless compression of hyperspectral images,” in *2014 6th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*. IEEE, 2014, pp. 1–4.
[57] C.-C. Chang and C.-J. Lin, “LIBSVM: A library for support vector machines,” *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Süha Tuna received his Ph.D. degree from Computational Science and Engineering in İstanbul Technical University, İstanbul, Turkey. He currently is an Assistant Professor at Computer Engineering Department in Fatih Sultan Mehmet Vakıf University. His research interests are high dimensional modelling, high performance computing, scientific computing and hyperspectral imaging.

Behçet Uğur Töreyin received the B.S. degree from the Middle East Technical University, Ankara, Turkey in 2001 and the M.S. and Ph.D. degrees from Bilkent University, Ankara, in 2003 and 2009, respectively, all in electrical and electronics engineering. He is now an Associate Professor with the Informatics Institute at Istanbul Technical University. His research interests broadly lie in signal processing and pattern recognition with applications to computational intelligence. His research is focused on developing novel algorithms to analyze and compress signals from multitude of sensors such as visible/infra-red/hyperspectral cameras, microphones, passive infra-red sensors, vibration sensors and spectrum sensors for wireless communications.

Metin Demiralp is the founder and the first director of Informatics Institute in İstanbul Technical University, İstanbul, Türkiye (Turkey). He currently keeps a faculty member position as an emeritus professor in the same institute. He has various collaborations at international scale. One can find quite detailed information about his research interests, academic life, past and current scientific activities, publications etc. through the following URL. https://web.itu.edu.tr/demiralp/homepage/

Jinchang Ren received the B.E. degree in computer software, the M.Eng. degree in image processing, and the D.Eng. degree in computer vision from Northwestern Polytechnical University, Xi’an, China, and the Ph.D. degree in electronic imaging and media communication from Bradford University, U.K. He is currently a Reader with the Department of Electronic and Electrical Engineering, University of Strathclyde, Glasgow, U.K. His research interests include image processing, computer vision, machine learning, hyperspectral imaging, remote sensing, and big data analytics. He has published over 300 scientific papers, and sits in the editorial board of five international journals. His students have been awarded a few Best Paper and Best Poster Prizes as well as the Best PhD Thesis from the IET Image & Vision Section in 2016.

Stephen Marshall received the B.Sc. degree in electrical and electronic engineering from the University of Nottingham and the Ph.D. degree in image processing from the University of Strathclyde, U.K. He is currently a Professor with the Department of Electronic and Electrical Engineering, Strathclyde. With more than 150 papers published, his research activities focus in nonlinear image processing and hyperspectral imaging. Dr. Marshall is a Fellow of the IET.