Int. J. Appl. Math. Comput. Sci., 2018, Vol. 28, No. 1, 197–207
DOI: 10.2478/amcs-2018-0015

LARGE–SCALE HYPERSPECTRAL IMAGE COMPRESSION VIA SPARSE REPRESENTATIONS BASED ON ONLINE LEARNING

İREM ÜLKÜ [a,*], ERSIN KIZGUT [b]

[a]Department of Electrical and Electronics Engineering
Çankaya University, Eskişehir Yolu 29.km, 06790 Ankara, Turkey
e-mail: iremulku@cankaya.edu.tr

[b]University Institute of Pure and Applied Mathematics (IUMPA)
Technical University of Valencia, E-46071 Valencia, Spain
e-mail: erkiz@upv.es

In this study, proximity based optimization algorithms are used for lossy compression of hyperspectral images that are inherently large scale. This is the first time that such proximity based optimization algorithms are implemented with an online dictionary learning method. Compression performances are compared with the one obtained by various sparse representation algorithms. As a result, proximity based optimization algorithms are listed among the three best ones in terms of compression performance values for all hyperspectral images. Additionally, the applicability of anomaly detection is tested on the reconstructed images.

Keywords: hyperspectral imaging, compression algorithms, dictionary learning, sparse coding.

1. Introduction

Hyperspectral images are data cubes that are composed of hundreds of narrow spectral bands generally in the visible and near-infrared spectrum. Data cubes have huge image sizes. Compression is crucial to remain within the transmission bandwidth limits during the downlink operation from satellite to ground (Penna *et al.*, 2007).

Hyperspectral image compression can be divided into two basic types: lossy and lossless. Even though lossless compression techniques maintain a full image quality, high compression ratios cannot be achieved with such methods.

One of the most popular methods that uses spectral correlation characteristics is principal component analysis (PCA) (Nowicki *et al.*, 2012; Panek *et al.*, 2015). An improved version of the PCA method is called compressive-projection principal component analysis (CPPCA) (Fowler, 2009).

Dictionary learning has recently become very popular for hyperspectral image compression (Wang *et al.*, 2014; Ülkü and Töreyin, 2015a; 2015b). Instead of using a pre-defined version, the dictionary is learned directly from the hyperspectral image. If the dictionary is fixed, then the process is called sparse coding. Data are represented by few sparse coefficients after dictionary learning and sparse coding is applied on hyperspectral images iteratively (Charles *et al.*, 2011). This is called sparse representation of data (Wright *et al.*, 2009; Zhang *et al.*, 2015).

This study analyzes sparse representation algorithms in three categories (Yang *et al.*, 2009; Zhang *et al.*, 2015). These include greedy pursuit algorithms, ℓ_p-norm regularization based algorithms and Bayesian algorithms. Greedy pursuit algorithms seek to obtain the sparsest solution by minimizing the ℓ_0-norm regularization. This category includes the matching pursuit (MP) algorithm (Mallat and Zhang, 1993). The orthogonal matching pursuit (OMP) algorithm is an improved version of the MP algorithm (Tropp and Gilbert, 2007). Besides, the OMP algorithm is also improved as the generalized OMP (gOMP) algorithm (Wang *et al.*, 2012). Other algorithms that belong to the greedy pursuit category are as follows: stagewise orthogonal matching pursuit (StOMP), regularized OMP (ROMP) and compressive

*Corresponding author

sampling matching pursuit (CoSaMP) (Donoho *et al*., 2012; Needell and Vershynin, 2009). The ℓ_p-norm regularization algorithms can be divided into two groups: those for $p \geq 1$ and those for $0 < p < 1$. Only ℓ_1-norm minimization is accepted to be sufficiently sparse (Zhang *et al*., 2015), and such algorithms can be categorized as follows: constrained based optimization algorithms, proximity based optimization algorithms and homotopy based optimization algorithms. Constrained optimization algorithms include the gradient projection sparse reconstruction (GPSR) algorithm (Nowak and Wright, 2007), the interior-point method algorithm (Boyd and Vandenberghe, 2004) and truncated Newton based interior-point method (TNIPM) algorithm (Kim *et al*., 2007). The alternating direction method of multipliers (ADMM) algorithm (Boyd *et al*., 2011) is used to solve the least absolute shrinkage and selection operator (LASSO). The last example is the active-set algorithm (Friedlander and Saunders, 2012). A dual active-set algorithm is employed to solve a basis pursuit (BP) problem (Chen *et al*., 2001).

Proximity based optimization algorithms are suitable for solutions of non-smooth, constrained and large scale problems (Parikh and Boyd, 2014). Some proximity algorithms are the iterative shrinkage thresholding algorithm (ISTA), the fast iterative shrinkage thresholding algorithm (FISTA), sparse reconstruction by separable approximation (SpaRSA), two-step IST (TwIST) algorithms (Bioucas-Dias and Figueiredo, 2007). Others are the general iterative shrinkage and thresholding (GIST) algorithm (Beck and Teboulle, 2009; Gong *et al*., 2013), the primal augmented Lagrangian method (PALM) and the dual augmented Lagrangian method (DALM) (Yang *et al*., 2013).

Non-convex ℓ_p-norm $(0 < p < 1)$ minimization problems are solved by using the generalized iterated shrinkage algorithm (GISA) (Zuo *et al*., 2013), and it is employed to compress hyperspectral data cubes with values of $p = 0.3, 0.4$ and 0.5. Homotopy based algorithms include the LASSO homotopy algorithm, which is proposed to solve LASSO problems (Donoho *et al*., 2012), and the basis pursuit denoising (BPDN) homotopy algorithm. Some algorithms that are classified as Bayesian compressive sensing algorithms (Ji *et al*., 2008) are as follows: smoothed projected Landweber (BCS-SPL), projected Landweber based on three-dimensional bivariate shrinkage (BCS PL-3DBS), and the wavelet packet transform (BCS PL-3DBS + 3DWPT 3D) (Hou and Zhang, 2014).

In this paper, the following contributions are achieved:

(a) Various sparse representation algorithms based on online dictionary learning are utilized for lossy compression of large-scale hyperspectral images. Compression performances of these algorithms are compared with those of state-of-the-art hyperspectral compression algorithms.

(b) This is the first time that proximity based optimization algorithms are implemented with the online dictionary learning method in hyperspectral image compression.

(c) Information preservation performances of different sparse representation algorithms based on online dictionary learning are tested by applying anomaly detection on the original and reconstructed hyperspectral images.

Hyperspectral image compression using sparse representation algorithms based on online dictionary learning is introduced in Section 2. Results are given in Section 3. In Section 4, conclusions are discussed.

2. Hyperspectral image compression using sparse representation algorithms based on online dictionary learning

In this section, hyperspectral image compression using online dictionary learning based sparse coding is discussed.

In the literature, an online learning approach is suggested to effectively solve large-scale optimization problems (Mairal *et al*., 2010). The online approach processes one element from the training set at a time. It performs techniques based on stochastic approximations. An iterative online learning algorithm is used in this paper that minimizes the quadratic surrogate function of the empirical cost.

2.1. Problem statement. The following parameters are used in the analyses. The number of bands in the hyperspectral data cube is defined as n_b, the number of lines in the hyperspectral data cube is described as n_l, the number of samples in the hyperspectral data cube is expressed as n_s and the number of columns in the dictionary is represented as k. $\boldsymbol{D}_0 \in \mathbb{R}^{n_b \times k}$ is the initial dictionary, $\boldsymbol{A}_0 \in \mathbb{R}^{k \times k}$ and $\boldsymbol{B}_0 \in \mathbb{R}^{n_b \times k}$ are auxiliary matrices to update the dictionary, T is the number of iterations, $\lambda \in \mathbb{R}$ is the regularization parameter and $\boldsymbol{\alpha} \in \mathbb{R}^k$ are the sparse coefficients. In the literature (Olshausen and Field, 1997), dictionary learning is regarded as optimizing the empirical cost according to a finite training set $\boldsymbol{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_T]$ in $\mathbb{R}^{n_b \times T}$. The empirical cost is defined as follows:

$$f_T(\boldsymbol{D}) \triangleq \frac{1}{T} \sum_{i=1}^{T} l(\mathbf{x}_i, \boldsymbol{D}), \tag{1}$$

where $\boldsymbol{D} \in \mathbb{R}^{n_b \times k}$ is the dictionary and l is the loss function. The latter is defined as the optimal value of an ℓ_1-sparse coding problem (Mairal *et al*., 2010),

$$l(\mathbf{x}_t, \boldsymbol{D}) \triangleq \min_{\boldsymbol{\alpha} \in \mathbb{R}^k} \frac{1}{2}\|\mathbf{x}_t - \boldsymbol{D}, \boldsymbol{\alpha}_t\|_2^2 + \lambda \|\boldsymbol{\alpha}_t\|_1, \quad (2)$$

where λ is the regularization parameter, $\mathbf{x}_t$ is the training sample at iteration t and $\boldsymbol{\alpha}_t$ is the corresponding coefficient set at iteration t. The regularization ℓ_1 yields a sparse solution in (2). In order to avoid having arbitrarily large values in $\boldsymbol{D} = [\mathbf{d}_1 \dots \mathbf{d}_k]$, which brings about having arbitrarily small $\boldsymbol{\alpha}_t$ values, a convex set of matrices C is given by

$$C \triangleq \{\boldsymbol{D} \in \mathbb{R}^{n_b \times k} : \|\mathbf{d}_j\| \leq 1, \forall j = 1, \dots k\}. \quad (3)$$

Minimizing the empirical cost $f_T(\boldsymbol{D})$ with respect to $\boldsymbol{D}$ is not convex. Therefore, the original optimization problem is reformulated as a joint optimization one. In this way, the problem can be considered convex with respect to $\boldsymbol{D}$ when the sparse coefficients $\boldsymbol{\Gamma} = [\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_T] \in \mathbb{R}^{k \times T}$ are fixed and with respect to the sparse coefficients $\boldsymbol{\Gamma}$ when $\boldsymbol{D}$ is fixed. This joint optimization problem is given by

$$\min_{\boldsymbol{D} \in C, \boldsymbol{\Gamma} \in \mathbb{R}^{k \times T}} \sum_{i=1}^{T} \left(\frac{1}{2}\|\mathbf{x}_i - \boldsymbol{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda \|\boldsymbol{\alpha}_i\|_1 \right). \quad (4)$$

In order to solve Eqn. (4), alternately one variable is fixed and the other is minimized as a convex optimization problem. As stated in the literature, the expected cost $f(\boldsymbol{D})$ can be minimized instead of minimizing the empirical cost. The expected cost is defined as

$$f(\boldsymbol{D}) \triangleq E_x[l(\mathbf{x}, \boldsymbol{D})] = \lim_{T \to \infty} f_T(\boldsymbol{D}), \quad (5)$$

where the expectation is obtained by taking into account the unknown probability distribution of the data. The equality in (5) is proved to almost certainly converge. Stochastic gradient algorithms are shown to be better for large-scale data sets in terms of the rate of convergence. The projected first order stochastic gradient descent algorithm is used in dictionary learning. This algorithm implies sequence updates of the dictionary $\boldsymbol{D}$,

$$\boldsymbol{D}_t = \prod_C \left[\boldsymbol{D}_{t-1} - \rho_t \nabla_D l(\mathbf{x}_t, \boldsymbol{D}_{t-1}) \right], \quad (6)$$

where $\boldsymbol{D}_t$ is the optimal dictionary at iteration t, ρ is the gradient step and $\prod_C$ is the orthogonal projector on C. The training set X is composed of i.i.d. samples of the unknown distribution of the data (Mairal *et al*., 2010).

2.2. Algorithm. The algorithm is composed of the consecutive parts of dictionary learning and dictionary update. First, sparse coding is carried out to acquire $\boldsymbol{\alpha}_t$ by using $\mathbf{x}_t$ and $\boldsymbol{D}_{t-1}$ from the previous iteration. Afterwards, a new dictionary $\boldsymbol{D}_t$ is obtained by minimizing the function $\widehat{f}$ over C:

$$\widehat{f}_t(\boldsymbol{D}) \triangleq \frac{1}{t} \sum_{i=1}^{t} \frac{1}{2}\|\mathbf{x}_i - \boldsymbol{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda \|\boldsymbol{\alpha}_i\|_1, \quad (7)$$

where $\boldsymbol{\alpha}_i$ values are found from the previous iterations. The quadratic function $\widehat{f}_t(\boldsymbol{D}_t)$ and $f_t(\boldsymbol{D}_t)$ converges to the same limit with probability one. Hence, function $\widehat{f}_t$ can be considered a surrogate for function f_t since function $\widehat{f}_t$ is close to function $\widehat{f}_{t-1}\boldsymbol{D}_t$ that can be acquired by using $\boldsymbol{D}_{t-1}$ as a warm restart.

2.2.1. Algorithm 1: Dictionary learning. Solving (2) with a fixed dictionary is called sparse coding, and it is defined as the sparse coding equation as shown in Table 1.

Algorithm 1. Dictionary learning.

1: Construct random initial dictionary $\boldsymbol{D}_0$
2: Set initial values $\boldsymbol{A}_0$ and $\boldsymbol{B}_0$ matrices to zero
3: **for** $t = 1$ to T **do**
4: Choose $\mathbf{x}_t \in \mathbb{R}^{n_b}$ randomly from the image.
5: Solve "sparse coding equation".
6: Update $\boldsymbol{A}_t = \boldsymbol{A}_{t-1} + \boldsymbol{\alpha}_t\boldsymbol{\alpha}_t^T$, $\boldsymbol{B}_t = \boldsymbol{B}_{t-1} + \mathbf{x}_t\boldsymbol{\alpha}_t^T$.
7: Find $\boldsymbol{D}_t$ using **Dictionary Update Algorithm**.
8: **end for**
9: Obtain learned dictionary $\boldsymbol{D}_t$.

2.2.2. Algorithm 2: Dictionary update. Equation (7) is defined as a dictionary update equation as shown in Table 1. $\boldsymbol{D}_{t-1}$ is used as a warm restart to update $\boldsymbol{D}_t$.

Algorithm 2. Dictionary update.

1: Calculate $\boldsymbol{D}_t$ in "dictionary update equation"
2: **repeat**
3: **for** $j = 1$ to k **do**
4: Find jth column of $\boldsymbol{D}_t$, where

$$\boldsymbol{D} = [\mathbf{d}_1 \dots \mathbf{d}_k] \in \mathbb{R}^{n_b \times k},$$
$$\boldsymbol{A} = [\mathbf{a}_1 \dots \mathbf{a}_k] \in \mathbb{R}^{k \times k},$$
$$\boldsymbol{B} = [\mathbf{b}_1 \dots \mathbf{b}_k] \in \mathbb{R}^{n_b \times k}$$

5: $\mathbf{u}_j = \frac{1}{A(j,j)}(\mathbf{b}_j - \boldsymbol{D}\mathbf{a}_j) + \mathbf{d}_j$
6: $\mathbf{d}_j = \dfrac{1}{\max(\|\mathbf{u}_j\|_2, 1)}\mathbf{u}_j$
7: $E_j = \sqrt{\sum_{n_b} |\mathbf{d}_j^t - \mathbf{d}_j^{t-1}|^2}$
8: **end for**
9: $E = \frac{1}{k}\sum_{j=1}^{k} E_j$
10: **until** $E <$ Threshold
11: Use $\boldsymbol{D}$ in **Dictionary Learning Algorithm**.

3. Results

Lossy compression of large scale hyperspectral images by using different sparse representation algorithms based on online dictionary learning is tested with the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) and Hyperion datasets. Compression performance results are compared with those of the state-of-the-art BCS PL-3DBS + 3DWPT and CPPCA compression algorithms.

In this paper, compression performance is measured by the peak signal-to-noise ratio (PSNR). Let us define r as the bit rate that is calculated in bits per sample (bps). This calculation is given as

$$r = \frac{z}{n_b}(b_d), \quad z < k, \tag{8}$$

where z is defined as the number of sparse coefficients, k represents the size of the dictionary, n_b represents the number of bands and b_d is defined as the bit depth.

3.1. Datasets. Detailed information about the datasets utilized in this paper is presented in Table 3.

3.2. Compression performance results with the AVIRIS datasets. The low Altitude, Lunar Lake and Jasper Ridge AVIRIS datasets are used (cf. Table 3). Compression performance comparison between sparse representation algorithms and state-of-the-art compression ones is presented in Table 2. Compression performance is expressed as PSNR values in dBs corresponding to different compression ratios in bps. The state-of-the-art algorithms used for the lossy compression of hyperspectral images are BCS PL-3DBS + 3DWPT and CPPCA. The three highest PSNR values are printed in boldface for each row. Among sparse representation algorithms, proximity based optimization ones are SpaRSA, FISTA, TwIST and GIST. In Table 2, they are written in boldface as well.

In Table 2, the best compression performance for the Low Altitude image at 0.1 bps bit rate belongs to the GISA with $p = 0.5$ algorithm.

The BCS PL-3DBS + 3DWPT algorithm has the highest PSNR value for the Lunar Lake image at the 0.1 bps bit rate. At the same rate, the SpaRSA algorithm shows the best performance in terms of the PSNR value for the Jasper Ridge dataset. Compression performances at 0.3 bps bit rate in Table 2 indicate that the gOMP algorithm is superior for the Low Altitude dataset. The SpaRSA algorithm has the highest PSNR value for the Lunar Lake image and the CPPCA algorithm outperforms for the Jasper Ridge image. At the 0.5 bps bit rate, the LASSO (ADMM) algorithm is superior for the Low Altitude image as seen in Table 2. The CPPCA algorithm has the highest PSNR value for both the Lunar Lake and Jasper Ridge datasets. For large scale datasets, PSNR values of the BP (Dual active set), GIST, CPPCA and LASSO (ADMM) algorithms are among the three best values for more than one dataset at 0.5 bps. The same pattern is followed by the SpaRSA, GISA and BP (Dual active set) algorithms at 0.1 bps together with the gOMP, BP (Dual active set) and SpaRSA algorithms at moderate compression ratio of 0.3 bps.

3.3. Compression performance results with the Hyperion datasets. The Erta Ale, Mt. St. Helens and Lake Monona images are used as the Hyperion datasets (cf. Table 3). Rate-distortion comparisons are illustrated in Figs. 1–3. PSNR values in dBs are plotted against three different compression ratios in bps as bar graphs. The highest three PSNR values for each bit rate are marked with small black circles below the corresponding algorithms. The proximity based optimization algorithms used in this case are SpaRSA, GIST and PALM. The TNIPM and SpaRSA algorithms are among the three best algorithms at the 0.1 bps bit rate for Erta Ale and Lake Monona images as seen in Figs. 1 and 3. At 0.3 bps bit rate, BP (Dual active set), SpaRSA and GIST algorithms are always the three best algorithms for all Hyperion images. According to 0.5 bps bit rate results, SpaRSA algorithm is among the top three algorithms for all datasets. This algorithm is followed by the GIST algorithm, which is among the best three algorithms for Mt. St. Helens and Lake Monona datasets as seen in Figs. 2 and 3.

For the performances for large scale datasets, the SpaRSA and GIST algorithms have PSNR values among the top three at the 0.1 bps bit rate for at least two datasets. At the 0.3 bps rate, the BP (Dual active set), SpaRSA and GIST algorithms show better performance than the others. SpaRSA and GIST are located among the highest three PSNR valued algorithms at the 0.5 bps bit rate for at least two Hyperion images as seen Figs. 1–3.

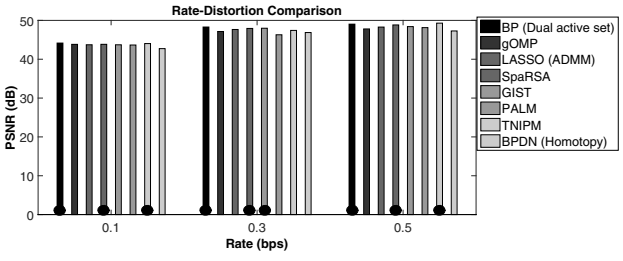

Fig. 1. Compression performance comparison between sparse representation algorithms for the Erta Ale image (cf. Table 3).

3.4. Evaluation of the results of proximity based optimization algorithms. The results of the AVIRIS

Table 1. Sparse coding and dictionary update equations of different sparse representation algorithms.

Algorithm	Sparse coding equation	Dictionary update equation
gOMP	$\boldsymbol{\alpha}_t = \arg\min\limits_{\boldsymbol{\alpha}\in\mathbb{R}^k} \frac{1}{2}\|\mathbf{x}_t - \boldsymbol{D}_{t-1}\boldsymbol{\alpha}\|_2$	$\boldsymbol{D}_t = \arg\min\limits_{\boldsymbol{D}\in C} \frac{1}{t}\sum\limits_{i=1}^{t}(\|\mathbf{x}_i - \boldsymbol{D}\boldsymbol{\alpha}_i\|_2),\ t = 1,\dots T$
LASSO (ADMM)	$\boldsymbol{\alpha}_t = \arg\min\limits_{\boldsymbol{\alpha}\in\mathbb{R}^k} \frac{1}{2}\|\boldsymbol{D}_{t-1}\boldsymbol{\alpha} - \mathbf{x}_t\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1$	$\boldsymbol{D}_t = \arg\min\limits_{\boldsymbol{D}\in C} \frac{1}{t}\sum\limits_{i=1}^{t}\left(\frac{1}{2}\|\boldsymbol{D}\boldsymbol{\alpha}_i - \mathbf{x}_i\|_2^2 + \lambda\|\boldsymbol{\alpha}_i\|_1\right),\ t = 1,\dots T$
BP (Dual active set)	$\boldsymbol{\alpha}_t = \arg\min\limits_{\boldsymbol{\alpha}\in\mathbb{R}^k} \|\boldsymbol{\alpha}_i\|$ s.t. $\boldsymbol{D}_{t-1}\boldsymbol{\alpha} = \mathbf{x}_t$	$\boldsymbol{D}_t = \operatorname{argmin}\limits_{\boldsymbol{D}\in C} \frac{1}{t}\sum\limits_{i=1}^{t}\|\boldsymbol{\alpha}_i\|_1$ s.t. $\frac{1}{t}\sum\limits_{i=1}^{t}(\boldsymbol{D}\boldsymbol{\alpha}_i) = \mathbf{x}_i,\ t = 1,\dots T$
SpaRSA	$\boldsymbol{\alpha}_t = \arg\min\limits_{\boldsymbol{\alpha}\in\mathbb{R}^k} \frac{1}{2}\|\mathbf{x}_t - \boldsymbol{D}_{t-1}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1$	$\boldsymbol{D}_t = \arg\min\limits_{\boldsymbol{D}\in C} \frac{1}{t}\sum\limits_{i=1}^{t}\left(\frac{1}{2}\|\mathbf{x}_i - \boldsymbol{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda\|\boldsymbol{\alpha}_i\|_1\right),\ t = 1,\dots T$
FISTA	$\boldsymbol{\alpha}_t = \arg\min\limits_{\boldsymbol{\alpha}\in\mathbb{R}^k} \frac{1}{2}\|\boldsymbol{D}_{t-1}\boldsymbol{\alpha} - \mathbf{x}_t\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1$	$\boldsymbol{D}_t = \arg\min\limits_{\boldsymbol{D}\in C} \frac{1}{t}\sum\limits_{i=1}^{t}\left(\frac{1}{2}\|\boldsymbol{D}\boldsymbol{\alpha}_i - \mathbf{x}_i\|_2^2 + \lambda\|\boldsymbol{\alpha}_i\|_1\right),\ t = 1,\dots T$
TwIST	$\boldsymbol{\alpha}_t = \arg\min\limits_{\boldsymbol{\alpha}\in\mathbb{R}^k} \frac{1}{2}\|\mathbf{x}_t - \boldsymbol{D}_{t-1}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1$	$\boldsymbol{D}_t = \arg\min\limits_{\boldsymbol{D}\in C} \frac{1}{t}\sum\limits_{i=1}^{t}\left(\frac{1}{2}\|\mathbf{x}_i - \boldsymbol{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda\|\boldsymbol{\alpha}_i\|_1\right),\ t = 1,\dots T$
GIST	$\boldsymbol{\alpha}_t = \arg\min\limits_{\boldsymbol{\alpha}\in\mathbb{R}^k} \frac{1}{2n_b}\|\boldsymbol{D}_{t-1}\boldsymbol{\alpha} - \mathbf{x}_t\|_2^2 + \lambda\sum\limits_{j=1}^{k}\min(\|a_j\|,\theta),\theta > 0$	$\boldsymbol{D}_t = \arg\min\limits_{\boldsymbol{D}\in C} \frac{1}{t}\sum\limits_{i=1}^{t}\left(\frac{1}{2n_b}\|\boldsymbol{D}\boldsymbol{\alpha}_i - \mathbf{x}_i\|_2^2 + \lambda\sum\limits_{j=1}^{k}\min(\|a_{ij}\|,\theta)\right),\ t = 1,\dots T$
PALM	$\boldsymbol{\alpha}_t = \arg\min\limits_{\boldsymbol{\alpha}\in\mathbb{R}^k} \|\boldsymbol{\alpha}_i\|$ s.t. $\boldsymbol{D}_{t-1}\boldsymbol{\alpha} = \mathbf{x}_t$	$\boldsymbol{D}_t = \operatorname{argmin}\limits_{\boldsymbol{D}\in C} \frac{1}{t}\sum\limits_{i=1}^{t}\|\boldsymbol{\alpha}_i\|_1$ s.t. $\frac{1}{t}\sum\limits_{i=1}^{t}(\boldsymbol{D}\boldsymbol{\alpha}_i) = \mathbf{x}_i,\ t = 1,\dots T$
TNIPM	$\boldsymbol{\alpha}_t = \arg\min\limits_{\boldsymbol{\alpha}\in\mathbb{R}^k} \frac{1}{2}\|\mathbf{x}_t - \boldsymbol{D}_{t-1}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1$	$\boldsymbol{D}_t = \arg\min\limits_{\boldsymbol{D}\in C} \frac{1}{t}\sum\limits_{i=1}^{t}(\frac{1}{2}\|\mathbf{x}_i - \boldsymbol{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda\|\boldsymbol{\alpha}_i\|_1),\ t = 1,\dots T$
BPDN (Homotopy)	$\boldsymbol{\alpha}_t = \arg\min\limits_{\boldsymbol{\alpha}\in\mathbb{R}^k} \frac{1}{2}\|\mathbf{x}_t - \boldsymbol{D}_{t-1}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1$	$\boldsymbol{D}_t = \arg\min\limits_{\boldsymbol{D}\in C} \frac{1}{t}\sum\limits_{i=1}^{t}(\frac{1}{2}\|\mathbf{x}_i - \boldsymbol{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda\|\boldsymbol{\alpha}_i\|_1),\ t = 1,\dots T$
GISA	$\boldsymbol{\alpha}_t = \arg\min\limits_{\boldsymbol{\alpha}\in\mathbb{R}^k} \frac{1}{2}\|\mathbf{x}_t - \boldsymbol{D}_{t-1}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_p^p,\ 0 \le p < 1$	$\boldsymbol{D}_t = \arg\min\limits_{\boldsymbol{D}\in C} \frac{1}{t}\sum\limits_{i=1}^{t}\left(\frac{1}{2}\|\mathbf{x}_i - \boldsymbol{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda\|\boldsymbol{\alpha}_i\|_p^p\right),\ 0 \le p < 1,\ t = 1,\dots T$

Table 2. Compression performance comparison between sparse representation algorithms and state-of-the-art compression algorithms in terms of PSNR values (Hou and Zhang, 2014).

	Lunar Lake image												
	Sparse representation algorithms												
BPS	BCS PL_3DBS + 3DWPT	CPPCA	gOMP	LASSO (ADMM)	BP (Dual active set)	SpaRSA	FISTA	TwIST	GIST	BPDN (Homotopy)	GISA $p = 0.4$	GISA $p = 0.5$	GISA $p = 0.3$
0.1	**61.34**	48.43	58.37	59.54	59.55	59.51	43.23	59.17	**59.68**	59.57	**59.58**	59.49	59.43
0.3	69.38	72.19	**73.84**	73.34	**73.85**	**73.89**	47.44	65.67	73.62	68.58	69.41	69.01	69.11
0.5	72.62	**76.82**	74.92	75.2	**76.55**	75.07	53.85	67.46	**75.37**	71.81	71.57	70.45	70.88
	Jasper Ridge image												
	Sparse representation algorithms												
BPS	BCS PL_3DBS + 3DWPT	CPPCA	gOMP	LASSO (ADMM)	BP (Dual active set)	SpaRSA	FISTA	TwIST	GIST	BPDN (Homotopy)	GISA $p = 0.4$	GISA $p = 0.5$	GISA $p = 0.3$
0.1	56.78	30.2	**59.4**	59.3	**59.41**	**59.47**	47.58	59.03	58.83	59.32	59.28	59.38	59.39
0.3	64.21	**71.31**	70.01	**70.67**	69.23	**70.69**	54.54	64.77	70.15	69.56	68.45	68.66	66.94
0.5	69.95	**76.4**	71.14	**73.17**	71.71	**72.49**	55.55	67.15	72.44	72.44	70.46	69.86	69.58
	Low altitude image												
	Sparse representation algorithms												
BPS	BCS PL_3DBS + 3DWPT	CPPCA	gOMP	LASSO (ADMM)	BP (Dual active set)	SpaRSA	FISTA	TwIST	GIST	BPDN (Homotopy)	GISA $p = 0.4$	GISA $p = 0.5$	GISA $p = 0.3$
0.1	54.74	47.47	59.79	59.59	**59.96**	59.88	48.23	59.4	59.85	59.82	59.74	**59.98**	59.35
0.3	61.74	60.98	**70.28**	68.85	**70.16**	69.78	50.39	63.63	69.79	69.04	67.64	66.81	67.72
0.5	67.08	70.01	72.68	**73.52**	73.24	72.82	56.53	67.01	**73.21**	72	71.9	68.41	69.32

Table 3. Detailed information of the AVIRIS, Hyperion, and Salinas-A hyperspectral datasets.

Aviris hyperspectral data					
Name	No. samples	No. lines	No. bands	Bit depth	Year
Jasper Ridge	614	2587	224	16	1997
Lunar Lake	614	1432	224	16	1997
Low Altitude	614	3689	224	16	1996
Hyperion hyperspectral data					
Name	No. samples	No. lines	No. bands	Bit depth	Year
Lake Monona	256	3176	242	12	2009
Mt. St. Helens	256	3242	242	12	2009
Erta Ale	256	3187	242	12	2010
Salinas-A hyperspectral data					
	No. samples	No. lines	No. bands	Bit depth	Year
	83	86	204	12	1998
Pavia university hyperspectral data					
	No. samples	No. lines	No. bands	Bit depth	Year
	200	200	103	12	2002

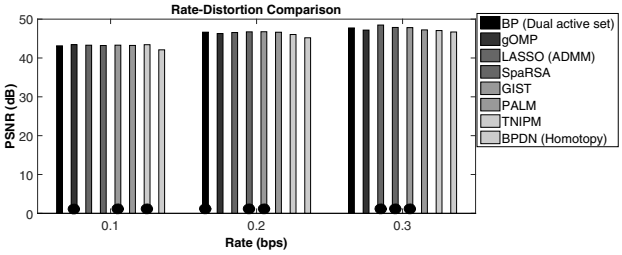

Fig. 2. Compression performance comparison between sparse representation algorithms for the Mt. St. Helens image (cf. Table 3).

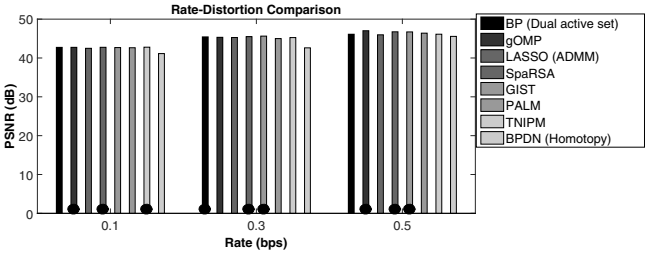

Fig. 3. Compression performance comparison between sparse representation algorithms for the Lake Monona image (cf. Table 3).

datasets indicate that algorithms corresponding to the highest three PSNR values for the Low Altitude, Lunar Lake and Jasper Ridge images at all bit rates include at least one proximity based optimization algorithm. Furthermore, test results of the Hyperion datasets point out that algorithms with the top three PSNR values at all bit rates for the Erta Ale, Mt. St. Helens and Lake Monona datasets span at least one proximity based optimization algorithm as well. The proximity based optimization algorithms used in this study are SpaRSA, FISTA, TwIST, GIST and PALM. As seen in Table 2 and Figs. 1–3, the SparRSA and GIST algorithms are included among the best algorithms at the 0.1, 0.3 and 0.5 bps bit rates for all images. On the other hand, according to the same results, the FISTA, TwIST and PALM algorithms cannot be drawn into the top three algorithms for any of the images. Consequently, among the proximity based optimization algorithms used in this study, the ones that should be preferred for large scale datasets are SpaRSA and GIST.

3.5. Application of anomaly detection on compressed and uncompressed hyperspectral images. The performances of different sparse representation algorithms based on online dictionary learning are further analyzed by doing anomaly detection on the original and the reconstructed hyperspectral images. For this purpose, the Reed–Xiaoli (RX) anomaly detection algorithm is used (Reed and Yu, 1990).

The spectral signature of the input signal is compared with the mean of each spectral band by using the Mahalanobis distance,

$$\delta_{RX}(\mathbf{x}_i) = (\mathbf{x}_i - \mathbf{M})^T \mathbf{Cov}^{-1}(\mathbf{x}_i - \mathbf{M}), \qquad (9)$$

where $\mathbf{x}_i \in \mathbb{R}^{n_b}$, $\mathbf{M}$ is the mean of each spectral band and $\mathbf{Cov}$ is the spectral covariance matrix. If $\delta_{RX}(\mathbf{x}_i) \geq \eta$, then it is assumed that an anomalous region is present, where η is a threshold value that is obtained by considering the desired false positive probability. The

Table 4. Compression performance comparison between sparse representation algorithms in terms of PSNR values.

	Salinas-A image			
	Sparse representation algorithms			
BPS	LASSO (ADMM)	BP (Dual active set)	SpaRSA	GIST
0.1	36.65	36.62	**36.78**	36.76
0.3	41.16	41.54	**42.61**	42.57
0.5	43.74	43.93	**43.96**	43.94

	Pavia University image			
	Sparse representation algorithms			
BPS	LASSO (ADMM)	BP (Dual active set)	SpaRSA	GIST
0.1	**30.96**	30.91	30.94	30.93
0.3	34.94	35.01	**35.07**	35.05
0.5	36.17	36.14	**36.22**	36.20

approximated covariance matrix, $\mathbf{Cov}$, is given by

$$\mathbf{Cov} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \mathbf{M})(\mathbf{x}_i - \mathbf{M})^T, \qquad (10)$$

where $N = n_l \times n_s$ and $i = 1, \ldots, N$. Anomaly detection is performed by using the Salinas-A hyperspectral dataset (cf. Table 3). In lossy compression, the information preservation performance measurement is significantly important. Fortunately, anomaly detection is accepted to be a valuable test of this performance (Du and Fowler, 2007). Anomaly detection results for the Salinas-A dataset are shown in Fig. 4. Figure 4(a) represents the anomaly detection result of the original hyperspectral image. According to Figs. 4(b)–(j), the anomaly which is detected in the original image can also be detected for 0.5 and 0.3 bps bit rates for SpaRSA and BP by using dual active set algorithm, and LASSO by using the ADMM algorithm. When the GIST algorithm is applied, an anomaly can only be detected for the 0.5 bps bit rate. In order to assess the robustness of the anomaly detection results given in Fig. 4, they should be based on numerical PSNR values. Therefore, the corresponding PSNR values of each sparse representation at 0.1, 0.3 and 0.5 bps levels are given in Table 4 for Salinas-A and Pavia University datasets. The highest two PSNR values are marked in boldface.

The anomaly detection performances of different sparse representation algorithms at various bit rates are evaluated by comparing their corresponding semilog ROC curves. Figure 5 presents the semilog ROC curves when SpaRSA is performed. Here, P_D represents the probability of detection and P_{FA} represents the probability of false positives. At the 0.5 bps bit rate the detection result is slightly better than those for the 0.3 bps and 0.1 bps rates. In Fig. 6, the semilog ROC curve performances belonging to 0.5 bps and 0.3 bps bit rates are similar to when BP with the active set algorithm is used.

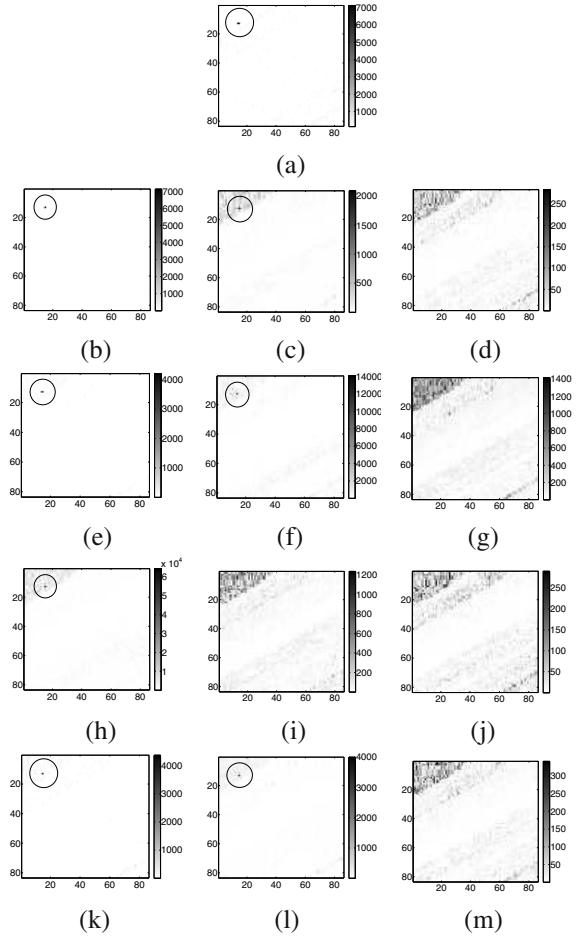

Fig. 4. RX anomaly detection results of the Salinas-A hyperspectral image: original image (a), SpaRSA with 0.5 bps (b), SpaRSA with 0.3 bps (c), SpaRSA with 0.1 bps (d), BP with 0.5 bps (e), BP with 0.3 bps (f), BP with 0.1 bps (g), GIST with 0.5 bps (h), GIST with 0.3 bps (i), GIST with 0.1 bps (j), LASSO with 0.5 bps (k), LASSO with 0.5 bps (k), LASSO with 0.3 bps (l), LASSO with 0.1 bps (m).

The ROC performance scheme in Fig. 8 is very similar to the one observed in Fig. 6, where curves at the 0.5 bps and 0.3 bps bit rates show similar behavior. In Fig. 8, LASSO by using ADMM algorithm is used. When the GIST algorithm is applied, the resulting ROC performances are depicted in Fig. 7. The ROC performances are compatible with the anomaly detection results in Figs. 4(h), (i) and (j). Anomaly detection results for the Salinas-A hyperspectral dataset suggest that SpaRSA performs better than other algorithms at 0.5 bps bit rate in terms of information preservation as seen in Fig. 5. In additionally, according to Table 4, SpaRSA has

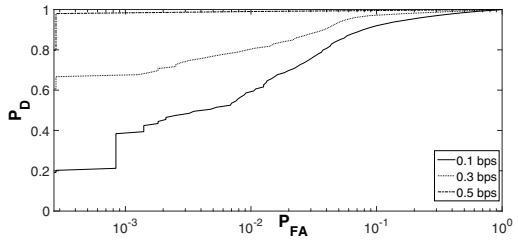

Fig. 5. Semilog ROC curves for the Salinas-A dataset at 0.1, 0.3 and 0.5 bps by using SpaRSA.

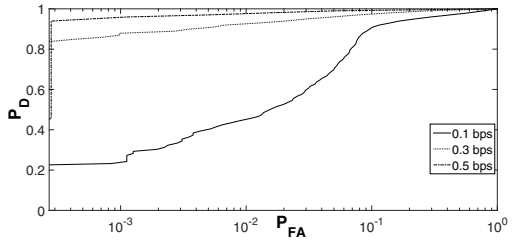

Fig. 6. Semilog ROC curves for the Salinas-A dataset at 0.1, 0.3 and 0.5 bps by using BP with the dual active set algorithm.

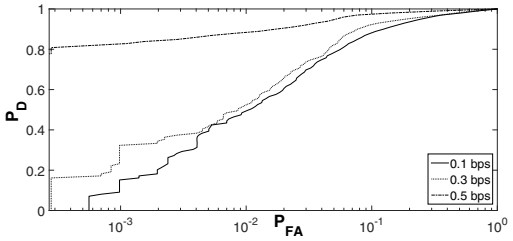

Fig. 7. Semilog ROC curves for the Salinas-A dataset at 0.1, 0.3 and 0.5 bps by using the GIST algorithm.

the highest PSNR value, which is 68.1968 at the 0.5 bps rate. SpaRSA is also among the best two algorithms in terms of the PSNR values at each bit rate for the Salinas-A dataset, as presented in Table 4. Due to the learning ability

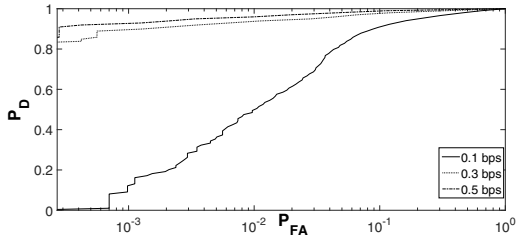

Fig. 8. Semilog ROC curves for the Salinas-A dataset at 0.1, 0.3 and 0.5 bps by using the LASSO algorithm.

of the online dictionary learning method, the anomaly part in the original dataset can be detected even at the 0.3 bps bit rate for SpaRSA and BP by using dual active set algorithm and for LASSO by using the ADMM algorithm, as seen in Fig. 4. Figure 4 also points out that at 0.1 bps bit rate none of these algorithms can detect the desired anomaly.

4. Conclusion

An analysis of the most effective sparse representation algorithms that are to be used for large scale hyperspectral image compression was carried out. For this purpose, sparse representation algorithms regarding various categories were tested by an online dictionary learning based method. The results were compared with the state-of-the-art lossy compression methods.

By giving weight to proximity based optimization algorithms, many algorithms that belong to this category were tested. This is the first time that proximity based optimization algorithms are used in conjunction with online dictionary learning method for compressing hyperspectral datasets. These algorithms are among the three best algorithms according to the PSNR values for all hyperspectral datasets at all bit rates.

According to the tests applied to large scale hyperspectral datasets, the SpaRSA and GIST algorithms from the proximity based optimization algorithms category and the BP (Dual active set) algorithm yield the best PSNR values at all compression ratio values. Other algorithms give the best compression performances only for particular compression ratio values.

Obviously, the anomaly detection results indicate that the compressed image roughly at the 0.5 bps bit rate can be a good approximation of the original hyperspectral image. Indeed, real-world applications such as anomaly detection can directly be applied to the reconstructed image of a smaller size.

By adding up-to-date sparse representation algorithms from each category, the analysis and comparison can be improved. In the future, by considering recent developments, more effective compression algorithms can possibly be obtained for large scale datasets.

Acknowledgment

The authors would like to thank the anonymous reviewers for their helpful and constructive comments that greatly contributed to improving the final version of the paper. They are also thankful to Prof. Halil T. Eyyuboğlu for his useful suggestions and comments. This research was partially supported by the Turkish Scientific and Technical Research Council.

References

Beck, A. and Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems, *SIAM Journal on Imaging Sciences* **2**(1): 183–202.

Bioucas-Dias, J.M. and Figueiredo, M.A. (2007). A new twist: Two-step iterative shrinkage/thresholding algorithms for image restoration, *IEEE Transactions on Image Processing* **16**(12): 2992–3004.

Boyd, S., Parikh, N., Chu, E., Peleato, B. and Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers, *Foundations and Trends in Machine Learning* **3**(1): 1–122.

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*, Cambridge University Press, Cambridge.

Charles, A.S., Olshausen, B.A. and Rozell, C.J. (2011). Learning sparse codes for hyperspectral imagery, *IEEE Journal of Selected Topics in Signal Processing* **5**(5): 963–978.

Chen, S.S., Donoho, D.L. and Saunders, M.A. (2001). Atomic decomposition by basis pursuit, *SIAM Review* **43**(1): 129–159.

Donoho, D.L., Tsaig, Y., Drori, I. and Starck, J.L. (2012). Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit, *IEEE Transactions on Information Theory* **58**(2): 1094–1121.

Du, Q. and Fowler, J.E. (2007). Hyperspectral image compression using JPEG2000 and principal component analysis, *IEEE Geoscience and Remote Sensing Letters* **4**(2): 201–205.

Fowler, J.E. (2009). Compressive-projection principal component analysis, *IEEE Transactions on Image Processing* **18**(10): 2230–2242.

Friedlander, M. and Saunders, M. (2012). A dual active-set quadratic programming method for finding sparse least-squares solutions, *Online*, University of British Columbia, Vancouver, BC, `http://web.stanford.edu/group/SOL/software/asp/bpdual.pdf`.

Gong, P., Zhang, C., Lu, Z., Huang, J. and Ye, J. (2013). A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems, *30th International Conference on Machine Learning (ICML), Atlanta, GA, USA*, pp. 37–45.

Hou, Y. and Zhang, Y. (2014). Effective hyperspectral image block compressed sensing using three-dimensional wavelet transform, *IEEE Geoscience and Remote Sensing Symposium (IGARSS), Quebec City, QC, Canada*, pp. 2973–2976.

Ji, S., Xue, Y. and Carin, L. (2008). Bayesian compressive sensing, *IEEE Transactions on Signal Processing* **56**(6): 2346–2356.

Kim, S.J., Koh, K., Lustig, M., Boyd, S. and Gorinevsky, D. (2007). An interior-point method for large-scale-regularized least squares, *IEEE Journal of Selected Topics in Signal Processing* **1**(4): 606–617.

Mairal, J., Bach, F., Ponce, J. and Sapiro, G. (2010). Online learning for matrix factorization and sparse coding, *Journal of Machine Learning Research* **11**: 19–60.

Mallat, S.G. and Zhang, Z. (1993). Matching pursuits with time-frequency dictionaries, *IEEE Transactions on Signal Processing* **41**(12): 3397–3415.

Needell, D. and Vershynin, R. (2009). Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit, *Foundations of Computational Mathematics* **9**(3): 317–334.

Nowak, R.D. and Wright, S.J. (2007). Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems, *IEEE Journal of Selected Topics in Signal Processing* **1**(4): 586–597.

Nowicki, A., Grochowski, M. and Duzinkiewicz, K. (2012). Data-driven models for fault detection using kernel PCA: A water distribution system case study, *International Journal of Applied Mathematics and Computer Science* **22**(4): 939–949, DOI: 10.2478/v10006-012-0070-1.

Olshausen, B.A. and Field, D.J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by v1?, *Vision Research* **37**(23): 3311–3325.

Panek, D., Skalski, A., Gajda, J. and Tadeusiewicz, R. (2015). Acoustic analysis assessment in speech pathology detection, *International Journal of Applied Mathematics and Computer Science* **25**(3): 631–643, DOI: 10.1515/amcs-2015-0046.

Parikh, N. and Boyd, S.P. (2014). Proximal algorithms, *Foundations and Trends in Optimization* **1**(3): 127–139.

Penna, B., Tillo, T. and Olmo, G. (2007). Transform coding techniques for lossy hyperspectral data compression, *IEEE Transactions on Geoscience and Remote Sensing* **45**(5): 1408–1421.

Reed, S.I. and Yu, X. (1990). Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution, *IEEE Transactions on Acoustics, Speech, and Signal Processing* **38**(10): 1760–1770.

Tropp, J.A. and Gilbert, A.C. (2007). Signal recovery from random measurements via orthogonal matching pursuit, *IEEE Transactions on Information Theory* **53**(12): 4655–4666.

Ülkü, İ. and Töreyin, B.U. (2015a). Sparse coding of hyperspectral imagery using online learning, *Signal, Video and Image Processing* **9**(4): 959–966.

Ülkü, İ. and Töreyin, B.U. (2015b). Sparse representations for online-learning-based hyperspectral image compression, *Applied Optics* **54**(29): 8625–8631.

Wang, J., Kwon, S. and Shim, B. (2012). Generalized orthogonal matching pursuit, *IEEE Transactions on Signal Processing* **60**(12): 6202–6216.

Wang, Z., Nasrabadi, N.M. and Huang, T.S. (2014). Spatial-spectral classification of hyperspectral images using discriminative dictionary designed by learning vector quantization, *IEEE Transactions on Geoscience and Remote Sensing* **52**(8): 4808–4822.

Wright, J., Yang, A.Y., Ganesh, A. and Sastry, S.S. (2009). Robust face recognition via sparse representation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **31**(2): 210–227.

Yang, A.Y., Zhou, Z., Balasubramanian, A.G., Sastry, S.S. and Ma, Y. (2013). Fast-minimization algorithms for robust face recognition, *IEEE Transactions on Image Processing* **22**(8): 3234–3246.

Yang, J., Peng, Y., Xu, W. and Dai, Q. (2009). Ways to sparse representation: An overview, *Science in China F: Information Sciences* **52**(4): 675–703.

Zhang, Z., Xu, Y., Yang, J., Li, X. and Zhang, D. (2015). A survey of sparse representation: Algorithms and applications, *IEEE Access* **3**: 490–530.

Zuo, W., Meng, D., Zhang, L., X.F. and Zhang, D. (2013). A generalized iterated shrinkage algorithm for non- convex sparse coding, *Proceedings of the IEEE International Conference on Computer Vision (ICCV), Sydney, Australia*, pp. 217–224.

İrem Ülkü received her BSc degrees (valedictorian) in industrial engineering and in electronics and communication engineering (highest honor) from Çankaya University in 2009 and 2010, respectively. She obtained the MSc degree in electrical and electronics engineering from Middle East Technical University in 2013. She received the PhD degree in electronics and communication engineering from Çankaya University in 2017. Her research interests include hyperspectral image processing, dictionary learning, compressive sensing and sparse coding.

Ersin Kızgut received a BSc degree (high honor) in mathematics and computer science from Çankaya University and a PhD degree in mathematics from Middle East Technical University in 2009 and 2016, respectively. His research interests include applied functional analysis, operator theory, complex analysis, and topological vector spaces.

Received: 10 February 2017
Revised: 19 June 2017
Re-revised: 7 September 2017
Accepted: 16 October 2017