A Stepwise Analytical Gradient Descent Search for Hyperspectral Unmixing and Its Parallel Code Vectorization

Fadi Kizel, Maxim Shoshany, and Nathan S. Netanyahu

Abstract — **Spectral mixture analysis (SMA) is a very important task for hyper-spectral image analysis, in general, and subpixel data extraction, in particular. In this paper we present a new methodology for spectral unmixing, where a vector of fractions, corresponding to a set of endmembers (EMs), is estimated for each pixel in the image. The process first provides an initial estimate of the fraction vector, followed by an iterative procedure that converges to an optimal solution. Specifically, *gradient descent* (GD) optimization is applied to the *spectral angle mapper* (SAM) objective function, so as to reduce significantly the estimation error due to amplitude (i.e., magnitude) variations in EM spectra, caused by the illumination change effect. Analytical derivations of the objective function's gradient and the optimal step size, in each iteration, are presented. To reduce the running time, we have implemented our unmixing module via code vectorization, i.e., the entire process is "folded" into a single loop, and the fractions for all of the pixels are solved for simultaneously. We call this new scheme *vectorized code gradient descent unmixing* (VCGDU). Its performance was compared to the commonly used *fully constrained least squares unmixing* (FCLSU) and the fast state-of-art method, *sparse unmixing by variable splitting and augmented Lagrangian* (SUnSAL), based on the *alternating direction method of multipliers* (ADMM). The comparison was carried out on a real Airborne Visible/Infrared Imaging Spectrometer image and several synthetic pixel sets composed of subsets of EMs (from the real image) whose fractions at each pixel were set at random. Considering all of the potential EMs at each pixel (without knowing specifically which EMs or how many of them are actually mixed in the pixel), we show that the accuracy due to VCGDU is higher than that obtained by FCLSU and SUnSAL for a relatively large**

number of actual EMs, and is considerably better under the illumination change effect. However, while our method is significantly faster than FCLSU, it is slower than SUnSAL by roughly an order of magnitude.

Fadi Kizel, and Maxim Shoshany are with the Department of Mapping and Geo-Information Engineering, Technion–Israel Institute of Technology, Haifa 32000, Israel (e-mail: fqzl@technion.ac.il; maximsh@technion.ac.il).

Nathan S. Netanyahu is with the Department of Computer Science, Bar-Ilan University, Ramat Gan 52900, Israel, and is also with the Center for Automation Research, University of Maryland, College Park, MD 20742 USA (e-mail: nathan@cs.biu.ac.il; nathan@cfar.umd.edu)

I. INTRODUCTION

Given a (hyper)spectral image, the linear mixture model assumes that the collected spectra in a given pixel is formed as a linear combination of a set of pure spectral signatures, known as endmembers (EMs). Only a few pixels in an image are essentially "pure" [1], while the rest – especially in remotely sensed images – contain more than one material. Thus, reliable analysis of acquired spectral data requires the process of *spectral unmixing*, where a vector of fractions (abundances), corresponding to the set of EMs, is estimated for each pixel in the scene. (See, e.g., [2], [3] for detailed surveys.)

The recent growing availability of airborne and satellite hyperspectral (HS) remote sensing platforms poses new challenges vis-à-vis the utilization of HS imagery in a wide range of applications. Such applications may include the important processing of urban, agricultural, and natural image regions, which requires the detection of a large number of biotic, a-biotic, and man-made materials. Unmixing image regions of this nature into merely four or five surface cover types would not meet the information level required, let alone justify the cost of acquiring HS imagery in the first place. To distinguish between a large number of EMs, one needs to handle the following two main issues that are associated inherently with HS imagery: (1)

Spectral similarities between different materials (with differences in only a few spectral features) and (2) variations in illumination angles (topographic effects) resulting in different spectral reflectance distributions for the same surface cover materials. In tackling these issues, it is important to determine first the most adequate EM set and then employ an appropriate unmixing strategy.

Methods for EM finding include, e.g., the manual EM selection tool (MEST) [4], as well as various automated algorithms, based on multidimensional geometric and statistical principles [5]. Early automated methods, e.g., the N-FINDR [6] and the improved version presented in [7], generally seek pure pixels that represent the EMs, while more recent methods [8], [9] do not assume the presence of pure pixels and try to estimate, instead, the EM spectra as the simplex vertices of the data cloud based on the principle of the *minimum volume enclosing simplex* (MVES). In addition, recently developed methods [10], [11] try to overcome the problem of non-present pure pixels by using sparse regression techniques [12]; a large number of library spectra is used to model the mixed pixels and the most suitable subset of EMs is found for each pixel during the unmixing process. See [13], for a detailed survey of EM extraction methods and [14—16], for more recent implementation approaches.

Once an adequate EM set is determined, an appropriate unmixing strategy needs to be employed to find an optimal abundance combination in fraction space. Numerous unmixing methods have been pursued over the years to meet this objective. For example, least squares-based approaches ([17], [18]) have been used in an iterative manner to provide fully constrained solutions. This was further refined by step-wise search strategies, such as *quadratic programming* [19—21], *gradient descent optimization* [22], [23], and *sequential quadratic programming* (SQP) [24]. One of the disadvantages, however, of a typical search algorithm is its low computational efficiency. Providing, if possible, semi-analytical solutions to gradient descent methods may alleviate this deficiency. A comprehensive overview of unmixing methods with an emphasis on fraction estimation can be found in [25], [26].

The choice of an objective function is also crucial for the unmixing performance and for setting a stopping criterion for the search process. The *Euclidean minimum distance* (EMD) is commonly used for this

purpose. Unfortunately, the performance of an EMD-based unmixing is highly affected by the illumination change artifact (which causes magnitude variations in the reflected spectra due to shadow and different topography [27]). Exploiting, on the other hand, the special geometric properties of the *Spectral Angle Mapper* (SAM) measure can significantly enhance the unmixing results, as was shown, for example, in [24], [28].

The study presented here aims at developing a computationally efficient approach for unmixing by applying an analytical gradient descent formulation to a variant of the SAM objective function. A detailed formulation of the analytical expression for the objective function's gradient and step size used in each iteration is provided. We combine these derived expressions with the *active set strategy* for constraint satisfaction to yield a stepwise analytical framework for fully constrained unmixing. The newly derived method can handle a relatively large number of EMs, is adaptable to any linear constraint imposed on the fractions, and is highly invariant to the illumination change effect. To overcome, however, the high processing time, typically associated with a gradient descent approach, we took advantage of the closed-form analytical expressions derived and the simplicity of the framework's components; specifically, we implemented our proposed scheme via *code vectorization*, which results in significant speedup on raster images. We call this unmixing scheme *vectorized code gradient descent unmixing* (VCGDU). The developed methodology was tested on synthetic and real data using an AISA image over a natural-urban mixed region. Also, it was compared against the commonly used FCLSU method [18] and the fast state-of-the-art method, SUnSAL [10]. (The latter provides an abundance estimate based on the *alternating direction method of multipliers* (ADMM) [29], by solving the so-called *constrained sparse regression* (CSR) problem.)

The rest of the paper is organized as follows. Section II presents the linear mixture model used in this work. Section III presents the main concepts of our framework. Section IV gives a detailed analytical derivation of the proposed gradient descent unmixing methodology, including a detailed discussion of its parallel implementation via code vectorization (VCGDU). Section V presents detailed experimental results,

including comparative performance evaluation of VCGDU versus FCLSU and SUnSAL. Finally, Section VI makes concluding remarks.

II. THE LINEAR MIXTURE MODEL

Assuming a linear mixture model (LMM), each pixel signature, $\mathbf{m} = [m_1,...,m_\lambda]^T$, in a hyperspectral image with λ bands, can be expressed as a linear combination of L EM spectra as follows:

$$\mathbf{m} = \mathbf{E}\mathbf{f} + \mathbf{n} \tag{1}$$

where $\mathbf{E}$ is a $(\lambda \times L)$ matrix whose columns are the EM spectral signatures, $\mathbf{f}$ is an $(L \times 1)$ vector containing the true fractions of the EMs, and n is a $(\lambda \times 1)$ vector, assumed to be a zero-mean Gaussian representing the system noise. During the unmixing process an estimated fraction vector, $\hat{\mathbf{f}}$, is calculated for each pixel in the image. The LMM represents the relation between the EMs, their fractions and the mixture, and it constitutes the basis for the mathematical terminology and formulations to be used during the unmixing process.

III. FRAMEWORK CONCEPTS

A. *Searching in Endmember Fraction Space for Spectral Unmixing*

Let $S \in \mathbb{R}^L$ be a scalar field spanned by the orthogonal set $\{(f_1,0,...,0),(0,f_2,...,0),...,(0,...,f_L)\}$, where each coordinate axis in *S* is represented by a single EM fraction. Each point in S represents a fraction combination which reconstructs a different mixture due to (1). A solution of the unmixing problem can be achieved by seeking the point (i.e., fraction combination) in *S* that optimally reconstructs the pixel's spectral signature. Ranking points in *S* as prospective solutions for the spectral unmixing requires a

definition of an objective function that measures the spectral distance between the reconstructed mixture and the pixel signature. Once an objective function is defined, an optimal solution can be achieved by finding the global maximum/minimum (depending on the kind of objective function). Searching for an optimal, fully constrained solution should be carried out by a examining a solution in the feasible region bounded by the non-negativity and sum-to-one constraints. Search methods for spectral unmixing should combine an objective function and an optimization method; a nonlinear objective function requires an initial estimate before applying an iterative search process. Although iterative search methods are more flexible and adaptive to different conditions and constraints, they tend to be very slow, especially when some required parameters are determined empirically. Thus the desired goal is to derive an analytical stepwise search method that would result in enhanced unmixing performance. The following three main components will be employed in our methodology:

1. *An initial estimate:*

We employ the preprocessing model presented in [30] to generate an initial estimate of the fraction vector based on the relationship between the fractions and the spectral angle mapper (SAM) values between the EMs and the mixture.

2. *Objective function:*

The objective function picked is based on the SAM measure for its advantage in reducing the illumination effect. The parameters of the search process presented in this paper will be analytically derived with respect to the SAM objective function.

3: *Stepwise analytical gradient descent with vectorized code:*

A gradient descent procedure was implemented for a search process in fraction space seeking for an optimal solution. Also, code vectorization was implemented for speeding up the entire process.

The above three components are integrated within a stepwise analytical framework, where the initial estimate is entirely empirical. The computation from this estimate is done in steps, applying an analytical search procedure between steps. At each step the result is assessed with respect to the objective function and the fraction change; if the solution is not satisfactory, then the search steps are repeated until the objective function is satisfied or the maximal fraction change is smaller than a pre-defined threshold. Fig. 1 conceptually demonstrates the algorithm's framework. In the next section we describe in detail each of these methodological components.

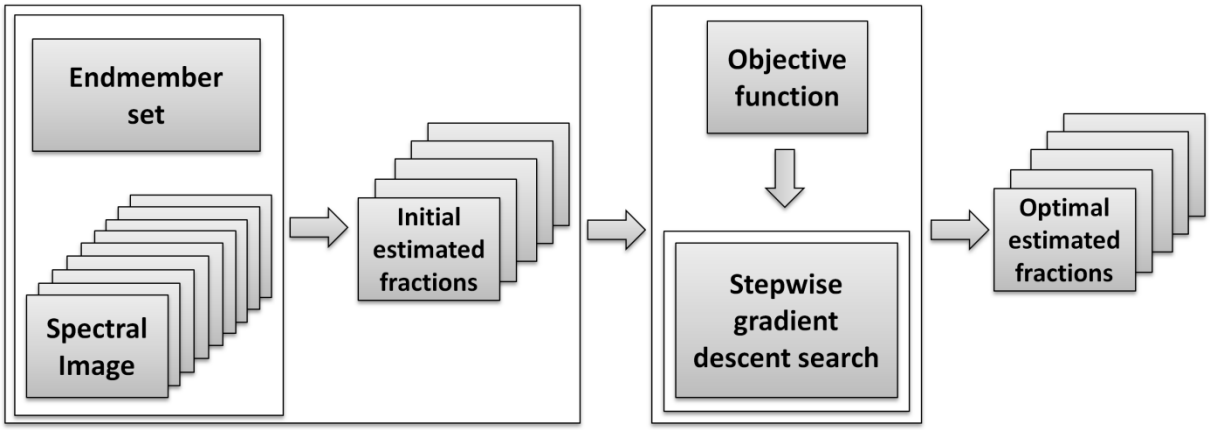

Fig. 1. Conceptual framework of unmixing methodology.

IV. STEPWISE ANALYTICAL GRADIENT DESCENT SPECTRAL UNMIXING

A. *Initial Estimate*

We provide an overview of this stage according to the presentation in [30]. Given two spectral signatures $\mathbf{s_1}$ and $\mathbf{s_2}$, their SAM similarity measure is defined as

$$SAM\left(\mathbf{s_1},\mathbf{s_2}\right)=\cos^{-1}\left(\frac{\mathbf{s_1}^T\mathbf{s_2}}{\left\|\mathbf{s_1}\right\|\cdot\left\|\mathbf{s_2}\right\|}\right) \tag{2}$$

where $\|\cdot\|$ denotes the L_2-norm. Logically, the larger the EM's fraction f_i is, the smaller the SAM measure is between its signature and the mixture signature. Actually, this relationship is a function of all of the EMs and their fractions in a given mixture and it can be mapped in order to create a basis for generating an initial estimate of the fraction vector. Given a pixel signature $\mathbf{m}$ that is a mixture of the EM set $\mathbf{E}$, the *normalized SAM* of the i^{th} EM, denoted by NS_i, is defined as:

$$NS_i = \frac{SAM(\mathbf{E}_i, \mathbf{m})}{\sum_{j=1}^{L} SAM(\mathbf{E}_j, \mathbf{m})}. \tag{3}$$

The relationship between a fraction and its NS value is intrinsic and can be described by fitting a linear function whose coefficients can be estimated according to [30]. Specifically, we carry out the following steps:

Step 1: Simulate a set of known fractions in the range $0 \le f_i \le 1$ and create for each fraction a controlled mixture $\mathbf{m}$ according to the expression

$$m^{(d)} = f_i E_i^{(d)} + \sum_{j \ne i} f_j E_i^{(d)}, \tag{4}$$

where $\mathbf{E}_i$ denotes the signature of the i^{th} EM, d is the spectral band index, and the f_j's ($j \ne i$) are picked at random subject to $\sum_{j \ne i} f_j = 1 - f_i$.

Step 2: For each fraction f_i and its corresponding controlled mixture**,** compute a corresponding NS_i according to (3)**.**

Step 3: Fit a linear function to estimate the relationship between the true fractions and their corresponding NS_i values.

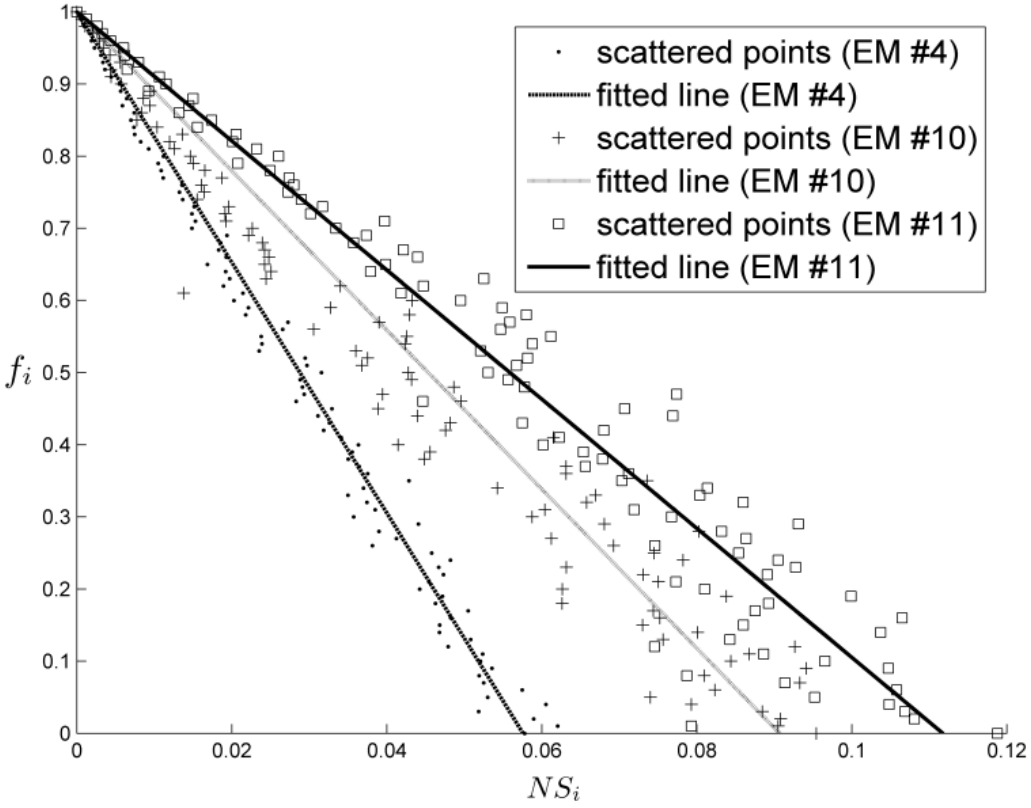

Fig. 2. Scatter plot of f_i vs. NS_i for several EMs, with fitted linear functions.

Fig. 2 presents the scatter of the fractions vs. their corresponding NS values, and the estimated linear functions for three different EMs taken from a set of EMs presented later (see Fig. 11). Each assessed coefficient is essentially a function of the EM set, regardless of the mixture to be solved; once preprocessing is applied, its results can be used for any mixture containing these EMs. Let $\boldsymbol{\alpha} = \left[\alpha_1, \alpha_2, \ldots, \alpha_L\right]$ and $\boldsymbol{\beta} = \left[\beta_1, \beta_2, \ldots, \beta_L\right]$ denote slope and intercept vectors, respectively, where α_i and β_i are the estimated coefficients of the fitted linear function for the i^{th} EM. Also, let $\mathbf{NS} = \left[NS_1, NS_2, \ldots, NS_L\right]$ denote the vector of normalized $SAMs$ computed according to (3) for a given mixture $\mathbf{m}$. Using our method, an initial estimate of the fraction vector is given by

$$\hat{\mathbf{f}}^0 = \boldsymbol{\alpha}^T \mathbf{NS} + \boldsymbol{\beta} \tag{5}$$

B. ***Gradient descent for optimal search***

Gradient descent is a standard, commonly used method for nonlinear optimization. Using it for the unmixing problem, we start with an initial estimate of the fraction vector $\hat{\mathbf{f}}^0$. Then, a stepwise computation towards the optimal solution is applied according to:

$$\hat{\mathbf{f}}^{k+1} = \hat{\mathbf{f}}^k + \gamma_k \nabla\phi\left(\hat{\mathbf{f}}^k\right) \tag{6}$$

where ϕ is the objective function, k is the iteration number, γ_k is the optimal step size in the gradient direction (giving a maximal change in ϕ), and $\nabla\phi\left(\hat{\mathbf{f}}^k\right)$ is the gradient of ϕ at $\hat{\mathbf{f}}^k$, or the derivative of the objective function with respect to the fraction vector, i.e., $\nabla\phi = \frac{\partial\phi}{\partial\hat{\mathbf{f}}}$. For a differentiable objective function ϕ, $\nabla\phi$ can be expressed analytically or can be calculated numerically. The gradient points in the direction which maximizes ϕ, but the change quantity in ϕ itself still depends on the step size γ. Finding the optimal step size, i.e., the step size that yields the highest change in ϕ (for the current iteration) is done by solving the optimization problem:

$$\gamma_k = \arg\max\{\varphi\} \tag{7}$$

where

$$\varphi = \phi\left(\hat{\mathbf{f}}^k + \gamma_k \nabla\phi\left(\hat{\mathbf{f}}^k\right)\right). \tag{8}$$

It would be desirable to find an explicit analytical solution by solving $\frac{\partial\varphi}{\partial\gamma_k} = 0$. Otherwise, a numerical solution may be applied.

C. ***Meeting the boundary constraints***

Requiring valid, feasible unmixing results, i.e., $0 \le \hat{f}_i \le 1$, constrains the optimization process. Specifically, we apply linear inequality constraints, which can be divided into the following two constraint subsets:

$$\begin{cases} \hat{f}_i \ge 0 \\ -\hat{f}_i \ge -1 \end{cases}$$

or in a unified matrix form:

$$\mathbf{A}\hat{\mathbf{f}} \ge \mathbf{b}$$

where

$$\underset{2L\times L}{\mathbf{A}} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ \dots & \dots & \dots & \dots & \dots \\ -1 & 0 & 0 & \dots & 0 \\ 0 & -1 & 0 & \dots & 0 \\ 0 & 0 & -1 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \dots & -1 \end{bmatrix} \quad \underset{L\times 1}{\hat{\mathbf{f}}} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_L \end{bmatrix} \quad \underset{2L\times 1}{\mathbf{b}} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ \dots \\ -1 \\ -1 \\ -1 \\ \vdots \\ -1 \end{bmatrix}$$

Each EM fraction f_i is bounded by two values according to rows i and $L+i$ of $\mathbf{A}$ and the corresponding elements in $\mathbf{b}$ (right-hand side of the corresponding inequalities).

Fulfilling these constraints can be done by applying the active set strategy, as presented in [31]. Let $\mathbf{a_j}$ be the j^{th} row in $\mathbf{A}$ and b_j be the corresponding value in $\mathbf{b}$, and let $\left\{ j \in I \right\}$ be the set of indices, such that, $\mathbf{a}_j\hat{\mathbf{f}} < b_j$. Define

$$\omega_j = \left\{ \frac{b_j - \mathbf{a}_j^T \hat{\mathbf{f}}^k}{\mathbf{a}_j^T \nabla\phi\left(\hat{\mathbf{f}}^k\right)} \middle| \; j \in I \right\}$$

$$\gamma_k = \begin{cases} \min\left[w_j\right] & \text{for } I \neq \varnothing \\ \arg\max\{\varphi\} & \text{for } I = \varnothing \end{cases}$$

In other words, the analytically calculated step size γ_k might yield a new point that is out of the feasible region (see Fig. 3(a), i.e., one or more boundary constraints are violated. The set I includes all the indices for which a boundary violation occurs; a maximal allowed step size ω_j is computed for each. If I is empty, we keep using the current analytically calculated step size γ_k; otherwise, the minimal value in $\boldsymbol{\omega}$ is chosen as the current step size to ensure that the next point is inside the feasible region.

Note that an additional constraint of sum-less-than-one can be easily imposed, if required, by adding a row of -1's to $\mathbf{A}$ and a value of -1 to the corresponding index in $\mathbf{b}$, i.e., $\mathbf{A}\left(2L+1,:\right) = \underset{1\times L}{-\mathbf{1}} = [-1,-1,\ldots,-1]$ and $\mathbf{b}\left(2L+1\right) = -1$. Finally, we defer the discussion about the sum-to-one constraint to the following section (see Fig. 3(b)), for clarity of presentation.

D. ***The objective function***

Among numerous available spectral unmixing methods, the objective function used mostly is the Euclidean minimum distance (EMD). One drawback of the EMD is its sensitivity to change in radiometry. Given that an EM spectral shape is fairly consistent while its amplitude varies significantly [32], it would be of interest to employ a spectral measure that minimizes the amplitude variation effect. A measure that meets this requirement is the spectral angle mapper (SAM). As indicated before, the beneficial use of the SAM as an objective function for the spectral unmixing was proved and shown in [24], [28]. In view of the previous assumptions, the objective function can be defined as

$$\psi = SAM\left(\mathbf{m}, \mathbf{E}\hat{\mathbf{f}}\right) = \cos^{-1}\left(\frac{\mathbf{m}^T\mathbf{E}\hat{\mathbf{f}}}{\left\|\mathbf{m}\right\| \cdot \left\|\mathbf{E}\hat{\mathbf{f}}\right\|}\right) \tag{9}$$

subject to

$$\sum_{i=1}^{L} \hat{f}_i \leq 1 \quad \text{and} \quad 0 \leq \hat{f}_i \leq 1.$$

An optimal estimation of $\hat{\mathbf{f}}$ can be achieved by minimizing ψ. Carrying out a gradient descent optimization requires the gradient (at any point) of the objective function. For convenience, we write:

$$\psi = \cos^{-1}\phi \tag{10}$$

The gradient of ψ can be derived by taking

$$\nabla\psi\left(\hat{\mathbf{f}}^k\right) = \frac{\partial\psi}{\partial\hat{\mathbf{f}}} = -\frac{\nabla\phi\left(\hat{\mathbf{f}}^k\right)}{\sqrt{1-\phi^2}} \tag{11}$$

We notice that the gradient of ψ is undefined at the minimum, where $\phi = 1$, and so a gradient optimization may provide unstable results when applied to ψ. Thus, we take ϕ as an alternative objective function, i.e., we want to maximize

$$\phi = \frac{\mathbf{m}^T\mathbf{E}\hat{\mathbf{f}}}{\left\|\mathbf{m}\right\| \cdot \left\|\mathbf{E}\hat{\mathbf{f}}\right\|} \tag{12}$$

subject to

$$\sum_{i=1}^{L} \hat{f}_i \leq 1 \quad \text{and} \quad 0 \leq \hat{f}_i \leq 1$$

In other words, an optimal estimation of the fractions should satisfy

$$\hat{\mathbf{f}} = \arg\max\{\phi\} \tag{13}$$

To derive the gradient of ϕ, let us express it as $\phi = \frac{\mu}{\vartheta}$, where $\mu = \mathbf{m}^T \cdot \mathbf{E}\hat{\mathbf{f}}$ and $\vartheta = \left\|\mathbf{m}\right\| \cdot \left\|\mathbf{E}\hat{\mathbf{f}}\right\|$. It can be easily shown that the derivatives of μ and ϑ with respect to $\hat{\mathbf{f}}$ are $\frac{\partial \mu}{\partial \hat{\mathbf{f}}} = \mathbf{E}^T\mathbf{m}$ and $\frac{\partial \vartheta}{\partial \hat{\mathbf{f}}} = \frac{\mathbf{E}^T\mathbf{E}\hat{\mathbf{f}}}{\left\|\mathbf{E}\hat{\mathbf{f}}\right\|} \cdot \left\|\mathbf{m}\right\|$, respectively. Thus, the gradient of ϕ can be derived as follows:

$$\begin{aligned} \nabla\phi\left(\hat{\mathbf{f}}\right) &= \frac{\partial \phi}{\partial \hat{\mathbf{f}}} = \frac{\frac{\partial \mu}{\partial \hat{\mathbf{f}}}\vartheta - \frac{\partial \vartheta}{\partial \hat{\mathbf{f}}}\mu}{\vartheta^2} \\ &= \frac{\mathbf{E}^T\mathbf{m} \cdot \left\|\mathbf{m}\right\| \cdot \left\|\mathbf{E}\hat{\mathbf{f}}\right\| - \frac{\mathbf{E}^T\mathbf{E}\hat{\mathbf{f}}}{\left\|\mathbf{E}\hat{\mathbf{f}}\right\|} \cdot \mathbf{m}^T\mathbf{E}\hat{\mathbf{f}} \cdot \left\|\mathbf{m}\right\|}{\left(\left\|\mathbf{m}\right\| \cdot \left\|\mathbf{E}\hat{\mathbf{f}}\right\|\right)^2} \end{aligned} \tag{14}$$

Simplifying, somewhat, this becomes

$$\nabla\phi\left(\hat{\mathbf{f}}^k\right) = \frac{\mathbf{E}^T\mathbf{m} \cdot \left\|\mathbf{E}\hat{\mathbf{f}}\right\|^2 - \mathbf{E}^T\mathbf{E}\hat{\mathbf{f}} \cdot \mathbf{m}^T\mathbf{E}\hat{\mathbf{f}}}{\left\|\mathbf{m}\right\| \cdot \left\|\mathbf{E}\hat{\mathbf{f}}\right\|^3} \tag{15}$$

As mentioned previously, the gradient points in the direction that maximizes the objective function; the amount of change in the objective function still depends on the step size γ. An optimal step size can be achieved by differentiating $\varphi = \phi\left(\hat{\mathbf{f}}^k + \gamma_k \nabla\phi\left(\hat{\mathbf{f}}^k\right)\right)$ with respect to γ_k and requiring that $\frac{\partial \varphi}{\partial \gamma_k} = 0$.

Following (12), and omitting the index notations, such that, $\hat{\mathbf{f}} = \hat{\mathbf{f}}^k$, $\gamma = \gamma_k$, and $\nabla = \nabla\phi\left(\hat{\mathbf{f}}^k\right)$, we get:

$$\varphi = \phi\left(\hat{\mathbf{f}} + \gamma\nabla\right) = \frac{\mathbf{m}^T\mathbf{E}\left(\hat{\mathbf{f}} + \gamma\nabla\right)}{\left\|\mathbf{m}\right\| \cdot \left\|\mathbf{E}\left(\hat{\mathbf{f}} + \gamma\nabla\right)\right\|} \tag{16}$$

Using the differentiation chain rule, we obtain:

$$\frac{\partial\varphi}{\partial\gamma}=\frac{\partial\phi\left(\hat{\mathbf{f}}+\gamma\nabla\right)}{\partial\gamma}=\frac{\partial\phi\left(\hat{\mathbf{f}}+\gamma\nabla\right)}{\partial\left(\hat{\mathbf{f}}+\gamma\nabla\right)}\cdot\frac{\partial\left(\hat{\mathbf{f}}+\gamma\nabla\right)}{\partial\gamma} \tag{17}$$

Following (15), and using again the shorthand index notation, we obtain the derivative of φ with respect to $\left(\hat{\mathbf{f}}+\gamma\nabla\right)$ as

$$\frac{\partial\varphi}{\partial\left(\hat{\mathbf{f}}+\gamma\nabla\right)}=\frac{\mathbf{E}^T\mathbf{m}\cdot\left\|\mathbf{E}\left(\hat{\mathbf{f}}+\gamma\nabla\right)\right\|^2-\mathbf{E}^T\mathbf{E}\left(\hat{\mathbf{f}}+\gamma\nabla\right)\cdot\mathbf{m}^T\mathbf{E}\left(\hat{\mathbf{f}}+\gamma\nabla\right)}{\left\|\mathbf{m}\right\|\cdot\left\|\mathbf{E}\left(\hat{\mathbf{f}}+\gamma\nabla\right)\right\|^3} \tag{18}$$

Also, the derivative of $\left(\hat{\mathbf{f}}+\gamma\nabla\right)$ with respect to γ is:

$$\frac{\partial\left(\hat{\mathbf{f}}+\gamma\nabla\right)}{\partial\gamma}=\nabla^T \tag{19}$$

Requiring that $\frac{\partial\phi}{\partial\gamma}=0$ yields

$$\nabla^T\mathbf{E}^T\mathbf{m}\cdot\left\|\mathbf{E}\left(\hat{\mathbf{f}}+\gamma\nabla\right)\right\|^2=\nabla^T\mathbf{E}^T\mathbf{E}\left(\hat{\mathbf{f}}+\gamma\nabla\right)\cdot\mathbf{m}^T\mathbf{E}\left(\hat{\mathbf{f}}+\gamma\nabla\right) \tag{20}$$

This can be rewritten as

$$\nabla^T\mathbf{E}^T\mathbf{m}\left(\left(\hat{\mathbf{f}}+\gamma\nabla\right)^T\mathbf{E}^T\mathbf{E}\left(\hat{\mathbf{f}}+\gamma\nabla\right)\right)=\nabla^T\mathbf{E}^T\mathbf{E}\left(\hat{\mathbf{f}}+\gamma\nabla\right)\cdot\mathbf{m}^T\mathbf{E}\left(\hat{\mathbf{f}}+\gamma\nabla\right) \tag{21}$$

Further manipulations give the equations below:

$$\begin{gathered}\nabla^T\mathbf{E}^T\mathbf{m}\left(\hat{\mathbf{f}}^T\mathbf{E}^T\mathbf{E}\hat{\mathbf{f}}+\gamma\nabla^T\mathbf{E}^T\mathbf{E}\hat{\mathbf{f}}+\gamma\hat{\mathbf{f}}^T\mathbf{E}^T\mathbf{E}\nabla+\gamma^2\nabla^T\mathbf{E}^T\mathbf{E}\nabla\right)=\\ \nabla^T\mathbf{E}^T\mathbf{E}\left(\hat{\mathbf{f}}\mathbf{m}^T\mathbf{E}\hat{\mathbf{f}}+\gamma\nabla\mathbf{m}^T\mathbf{E}\hat{\mathbf{f}}+\gamma\hat{\mathbf{f}}\mathbf{m}^T\mathbf{E}\nabla+\gamma^2\nabla\mathbf{m}^T\mathbf{E}\nabla\right)\end{gathered} \tag{22}$$

or

$$\begin{gathered}\nabla^T\mathbf{E}^T\mathbf{m}\hat{\mathbf{f}}^T\mathbf{E}^T\mathbf{E}\hat{\mathbf{f}}+\gamma\nabla^T\mathbf{E}^T\mathbf{m}\nabla^T\mathbf{E}^T\mathbf{E}\hat{\mathbf{f}}+\gamma\nabla^T\mathbf{E}^T\mathbf{m}\hat{\mathbf{f}}^T\mathbf{E}^T\mathbf{E}\nabla+\gamma^2\nabla^T\mathbf{E}^T\mathbf{m}\nabla^T\mathbf{E}^T\mathbf{E}\nabla\\ =\nabla^T\mathbf{E}^T\mathbf{E}\hat{\mathbf{f}}\mathbf{m}^T\mathbf{E}\hat{\mathbf{f}}+\gamma\nabla^T\mathbf{E}^T\mathbf{E}\nabla\mathbf{m}^T\mathbf{E}\hat{\mathbf{f}}+\gamma\nabla^T\mathbf{E}^T\mathbf{E}\hat{\mathbf{f}}\mathbf{m}^T\mathbf{E}\nabla+\gamma^2\nabla^T\mathbf{E}^T\mathbf{E}\nabla\mathbf{m}^T\mathbf{E}\nabla\end{gathered} \tag{23}$$

Since all of the terms in (23) are scalars, the following hold true:

$$\begin{aligned}\nabla^T\mathbf{E}^T\mathbf{m}\hat{\mathbf{f}}^T\mathbf{E}^T\mathbf{E}\nabla &= \left(\nabla^T\mathbf{E}^T\mathbf{m}\hat{\mathbf{f}}^T\mathbf{E}^T\mathbf{E}\nabla\right)^T = \nabla^T\mathbf{E}^T\mathbf{E}\hat{\mathbf{f}}\mathbf{m}^T\mathbf{E}\nabla \\ \nabla^T\mathbf{E}^T\mathbf{m}\nabla^T\mathbf{E}^T\mathbf{E}\nabla &= \left(\nabla^T\mathbf{E}^T\mathbf{m}\nabla^T\mathbf{E}^T\mathbf{E}\nabla\right)^T = \nabla^T\mathbf{E}^T\mathbf{E}\nabla\mathbf{m}^T\mathbf{E}\nabla\end{aligned} \tag{24}$$

Using these two facts, the last two terms on the left-hand side cancel out the last two terms on the right-hand side in (23), respectively, so we are left with the following equation:

$$\nabla^T\mathbf{E}^T\mathbf{m}\hat{\mathbf{f}}^T\mathbf{E}^T\mathbf{E}\hat{\mathbf{f}} + \gamma\nabla^T\mathbf{E}^T\mathbf{m}\nabla^T\mathbf{E}^T\mathbf{E}\hat{\mathbf{f}} = \nabla^T\mathbf{E}^T\mathbf{E}\hat{\mathbf{f}}\mathbf{m}^T\mathbf{E}\hat{\mathbf{f}} + \gamma\nabla^T\mathbf{E}^T\mathbf{E}\nabla\mathbf{m}^T\mathbf{E}\hat{\mathbf{f}} \tag{25}$$

Finally, we can now extract the following analytical expression for γ:

$$\gamma = \frac{\nabla^T\mathbf{E}^T\mathbf{E}\hat{\mathbf{f}}\mathbf{m}^T\mathbf{E}\hat{\mathbf{f}} - \nabla^T\mathbf{E}^T\mathbf{m}\hat{\mathbf{f}}^T\mathbf{E}^T\mathbf{E}\hat{\mathbf{f}}}{\nabla^T\mathbf{E}^T\mathbf{m}\nabla^T\mathbf{E}^T\mathbf{E}\hat{\mathbf{f}} - \nabla^T\mathbf{E}^T\mathbf{E}\nabla\mathbf{m}^T\mathbf{E}\hat{\mathbf{f}}} \tag{26}$$

Fulfilling the sum-to-one constraint:

The suggested objective function ϕ is a shape similarity measure and is invariant to scalar multiplication, i.e., $\phi\left(\mathbf{m},\mathbf{E}\mathbf{f}\right) = \phi\left(\mathbf{m},a\mathbf{E}\mathbf{f}\right)$, for some scalar a. In other words, the global maximum extends along a straight line in fraction space.

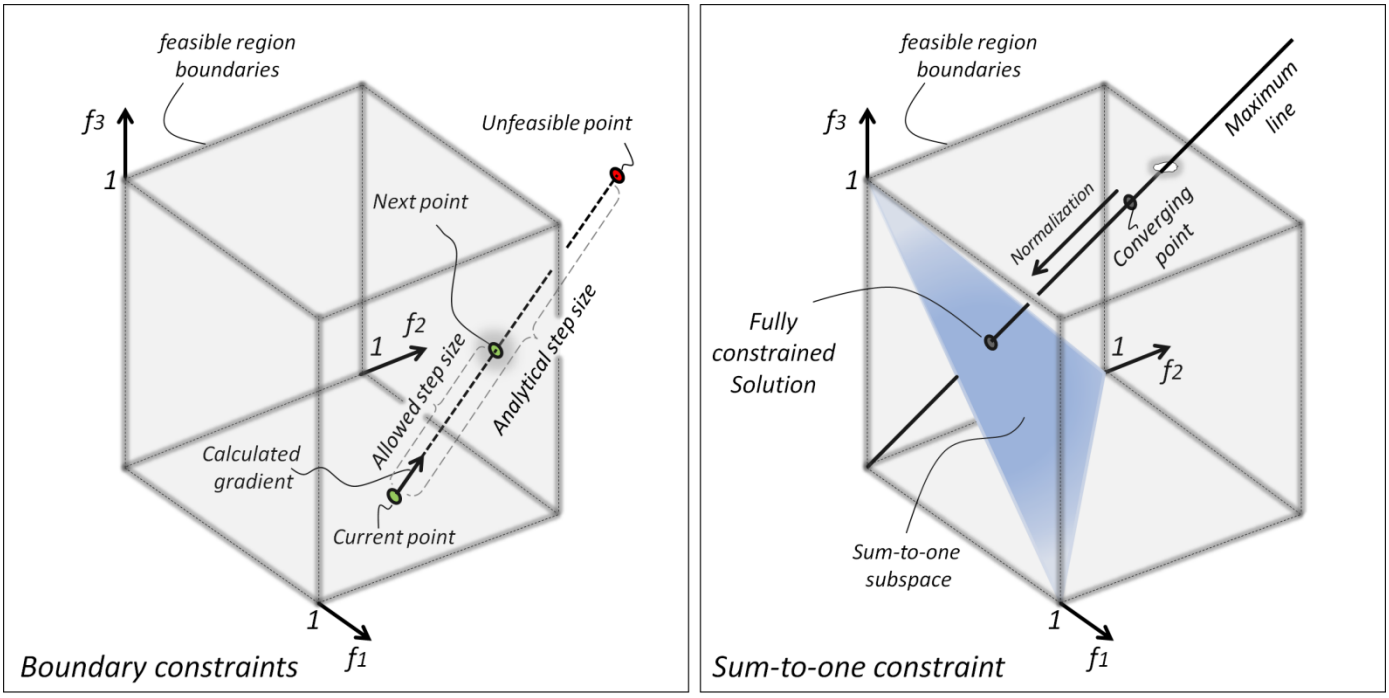

Fig. 3. Illustration of the analytical stepwise search progress for fully constrained unmixing in three-dimensional fraction space: (a) Fulfilling the non-negativity constraint by preventing step size from crossing a box facet, (b) fulfillment of sum-to-one constraint by normalizing converged solution (on the line of maximum values); extendible to unlimited number of EMs.

Using this fact, the computation may converge, eventually, to any point on this line inside the boundary constraint region. Thus, meeting the sum-to-one constraint can be done simply by normalizing the result, i.e., by dividing the fraction vector by the sum of its components. Fig. 3(b) demonstrates this concept for the special case of three EMs.

E. *Code vectorization for speedup*

A main drawback of gradient descent optimization is its typical slow convergence. The running time of our proposed gradient descent unmixing (GDU) is indeed higher in comparison to that of FCLSU and SUnSAL, especially for a large number of EMs. This was observed by running the procedures on 13 synthetic datasets of 1000 mixed pixels, where each pixel in a given dataset contains (a subset of) 2, 3, …,

14 EMs. This was repeated 10 times (for each dataset) and median run-times were recorded. Fig. 4 shows the run-times per-pixel obtained running Matlab on a Microsoft 64-bit Windows 7 operating system with 6GB of RAM.

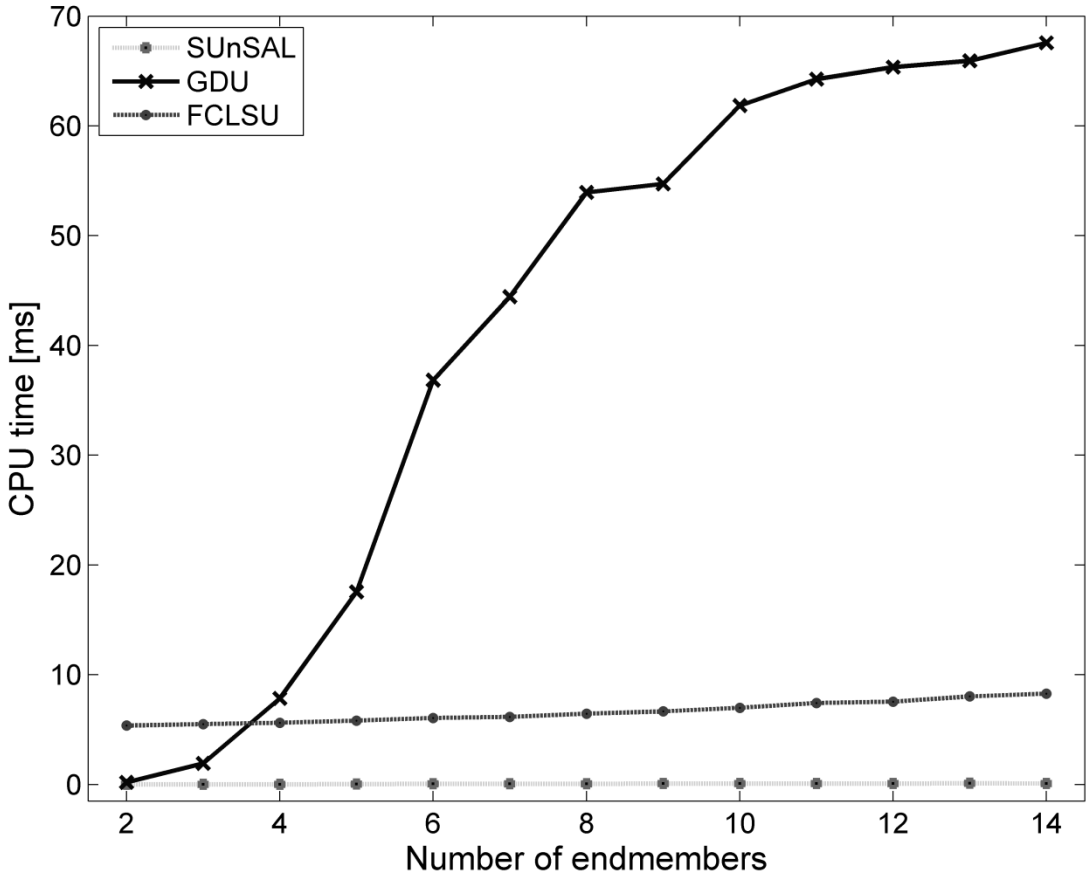

Fig. 4. Median run-times per-pixel (over 10 simulations) for GDU, FCLSU, and SUnSAL algorithms applied to 1000 mixed pixels as a function of number of EMs.

Using the basic form of the gradient descent method, the running time rapidly increases as the number of EMs exceeds five. The process to be applied (to the entire image) involves running three different loops over the image rows, the image columns, and the number of iterations until convergence. These three loops are given by the following pseudo-code:

Algorithm 1: Gradient Descent Unmixing (GDU)

Inputs: $\mathbf{E}(\lambda \times L)$: Matrix of set of EMs

$\mathbf{H}(r \times c \times \lambda)$: Cube of spectral image to be unmixed

$\hat{\mathbf{F}}^0(r \times c \times L)$: Cube of initial estimates of fraction image

1) **for** each row in image (*r* times)

2) **for** each column in image (*c* times)

3) **while** $\max\left(\left|\hat{\mathbf{f}}^k - \hat{\mathbf{f}}^{k-t}\right|\right) > threshold$.

/* The term $\left|\hat{\mathbf{f}}^k - \hat{\mathbf{f}}^{k-t}\right|$ indicates the amount of change in fraction values during the last *t* iterations, */

/* where *t* is a predefined parameter */.

4) Calculate $\nabla\phi\left(\hat{\mathbf{f}}^k\right)$ and γ_k by (15) and (26), respectively;

5) Apply active set strategy to ensure boundary constraints;

6) Set $\hat{\mathbf{f}}^{k+1} = \hat{\mathbf{f}}^k + \gamma_k \nabla\phi\left(\hat{\mathbf{f}}^k\right)$;

7) **end**

8) **end**

9) **end**

A significant advantage of the gradient descent method, given the analytical expression of the gradient and the step size, is its simple mathematical form which does not involve a complex operation such as matrix inversion. This makes the new method adaptable to code vectorization. Vectorized code compresses the entire process into a single loop of converging iterations where in each iteration the next fraction vector is calculated for all of the image pixels simultaneously by using the vectorized form of (15) and (26). We present below some matrix operations to be used for this mechanism:

Let $\underset{l\times 1}{\mathbf{1}} = [1,1,...,1]^T$ and let $\mathbf{U}, \mathbf{V} \in \mathbb{R}^{l\times s}$ be the matrices

$$\mathbf{U} = \begin{bmatrix} u_{1,1} & u_{1,2} & . & u_{1,s} \\ u_{2,1} & u_{2,2} & . & . \\ . & . & . & . \\ u_{l,1} & . & . & u_{l,s} \end{bmatrix} \quad \text{and} \quad \mathbf{V} = \begin{bmatrix} v_{1,1} & v_{1,2} & . & v_{1,s} \\ v_{2,1} & v_{2,2} & . & . \\ . & . & . & . \\ v_{l,1} & . & . & v_{l,s} \end{bmatrix}$$

The Hadamard (entry-wise) product is defined as follows:

$$\mathbf{U} \circ \mathbf{V} = \begin{bmatrix} u_{1,1}v_{1,1} & u_{1,2}v_{1,2} & . & u_{1,s}v_{1,s} \\ u_{2,1}v_{2,1} & u_{2,2}v_{2,2} & . & . \\ . & . & . & . \\ u_{l,1}v_{l,1} & . & . & u_{l,s}v_{l,s} \end{bmatrix} \tag{27}$$

We now present the following set of array operations for vectorizing the code:

Operation 1: Array multiplication $\left(.*\right)$, the same as the Hadamard product.

Operation 2: Array right division $\left(./\right)$, that is,

$$\mathbf{U}./\mathbf{V} = \begin{bmatrix} \frac{u_{1,1}}{v_{1,1}} & \frac{u_{1,2}}{v_{1,2}} & . & \frac{u_{1,s}}{v_{1,s}} \\ \frac{u_{2,1}}{v_{2,1}} & \frac{u_{2,2}}{v_{2,2}} & . & . \\ . & . & . & . \\ \frac{u_{l,1}}{v_{l,1}} & . & . & \frac{u_{l,s}}{v_{l,s}} \end{bmatrix}$$

Operation 3: Array power $\left(.\hat{}\right)$, that is,

$$\mathbf{U}.\hat{}\,q = \begin{bmatrix} {u_{1,1}}^q & {u_{1,2}}^q & . & {u_{1,s}}^q \\ {u_{2,1}}^q & {u_{2,2}}^q & . & . \\ . & . & . & . \\ {u_{l,1}}^q & . & . & {u_{l,s}}^q \end{bmatrix}$$

Creating a row vector containing the sums of matrix columns can be simply done as

$$\mathbf{1}^T\mathbf{U} = \left[\sum_{i=1}^{l} u_{i,1}, \sum_{i=1}^{l} u_{i,2}, \ldots, \sum_{i=1}^{l} u_{i,s}\right] \tag{28}$$

Converting a row vector into a matrix with the same vector duplicated along the rows: Letting $\mathbf{w} = \left[w_1, w_2, \ldots\ldots\ldots w_s\right]$, we have

$$\mathbf{1w} = \begin{bmatrix} 1 \\ 1 \\ . \\ . \\ 1 \end{bmatrix} \left[w_1, w_2, \ldots, w_s\right] = \begin{bmatrix} w_1, w_2, \ldots, w_s \\ w_1, w_2, \ldots, w_s \\ . \quad\quad\quad . \\ . \quad\quad\quad . \\ w_1, w_2, \ldots, w_s \end{bmatrix} \tag{29}$$

Using the above operations enables code vectorization of the gradient descent unmixing process. We are given a spectral image (with r rows, c columns, and λ bands), a matrix of L EMs, and a fraction image (with r rows, c columns and L bands) obtained by the initial estimation process. A matrix $\mathbf{M}$ (with λ rows and $r \cdot c$ columns) can then be created by permuting the spectral image as shown in Fig. 5. The same operation is applied to the estimated fraction image to create the matrix $\hat{\mathbf{F}}$ (with L rows and $r \cdot c$

columns); each column in $\mathbf{M}$ and $\hat{\mathbf{F}}$ contains the spectral signature and estimated fraction vector of the corresponding pixel in the spectral image and the fraction image, respectively.

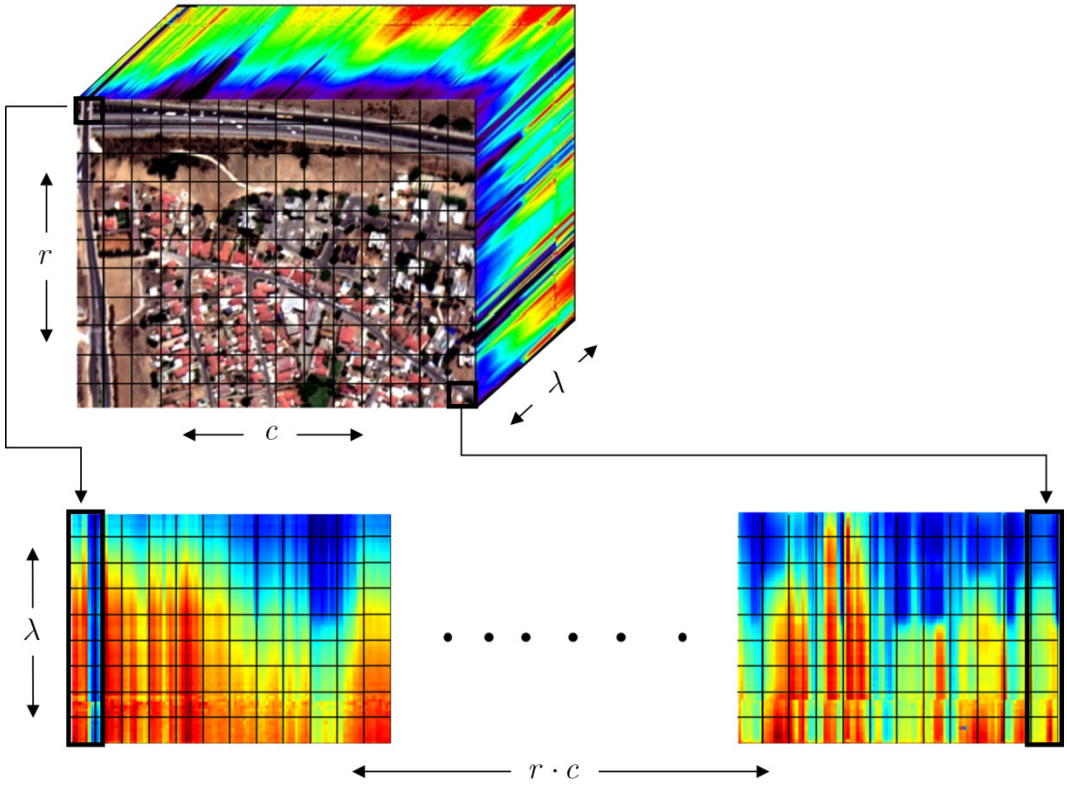

Fig. 5. Spectral cube permutation; each pixel in the 3D image is converted to a column in a 2D matrix; code vectorization is applied based on matrix operations.

Having defined $\mathbf{M}$, $\mathbf{E}$, and $\hat{\mathbf{F}}$, we now implement the *vectorization code of the gradient descent unmixing* (VCGDU). Specifically, the gradient $\nabla\phi\left(\hat{\mathbf{f}}^k\right)$ and the optimal step size γ_k can be simultaneously calculated for all the pixels by the following operations:

$$\nabla_1 = \mathbf{E}^T\mathbf{M} \tag{30}$$

$$\nabla_2 = \mathbf{1}\cdot\mathbf{1}^T\left(\left(\mathbf{E}\hat{\mathbf{F}}^k\right)\circ\left(\mathbf{E}\hat{\mathbf{F}}^k\right)\right) \tag{31}$$

$$\nabla_3 = \mathbf{E}^T\mathbf{E}\hat{\mathbf{F}}^k \tag{32}$$

$$\nabla_4 = \mathbf{1} \cdot \mathbf{1}^T \left(\mathbf{M} \circ \left(\mathbf{E}\hat{\mathbf{F}}^k \right) \right) \tag{33}$$

$$\nabla_5 = \mathbf{1} \cdot \mathbf{1}^T \left(\mathbf{M} \circ \mathbf{M} \right) .^{\wedge 0.5} \tag{34}$$

$$\nabla_n = \nabla_1 \circ \nabla_2 - \nabla_3 \circ \nabla_4 \tag{35}$$

$$\nabla_d = \nabla_5 \circ \left(\nabla_2 .^{\wedge 1.5} \right) \tag{36}$$

Thus, the gradient for each pixel in the image is given by:

$$\underset{entire\ \ image}{\nabla_{\hat{\mathbf{F}}^k}} = \nabla_n ./ \nabla_d \tag{37}$$

To obtain the step size, we compute the following:

$$\mathbf{\Gamma}_1 = \mathbf{1}^T \left(\mathbf{M} \circ \left(\mathbf{E}\hat{\mathbf{F}}^k \right) \right) \tag{38}$$

$$\mathbf{\Gamma}_2 = \mathbf{1}^T \left(\nabla_{\hat{\mathbf{F}}^k} \circ \left(\mathbf{E}^T\mathbf{E}\hat{\mathbf{F}}^k \right) \right) \tag{39}$$

$$\mathbf{\Gamma}_3 = \mathbf{1}^T \left(\hat{\mathbf{F}}^k \circ \left(\mathbf{E}^T\mathbf{E}\hat{\mathbf{F}}^k \right) \right) \tag{40}$$

$$\mathbf{\Gamma}_4 = \mathbf{1}^T \left(\nabla_{\hat{\mathbf{F}}^k} \circ \left(\mathbf{E}^T\mathbf{E}\nabla \right) \right) \tag{41}$$

$$\mathbf{\Gamma}_5 = \mathbf{1}^T \left(\nabla_{\hat{\mathbf{F}}^k} \circ \left(\mathbf{E}^T\mathbf{M} \right) \right) \tag{42}$$

$$\mathbf{\Gamma}_n = \mathbf{\Gamma}_2 \circ \mathbf{\Gamma}_1 - \mathbf{\Gamma}_5 \circ \mathbf{\Gamma}_3 \tag{43}$$

$$\mathbf{\Gamma}_d = \mathbf{\Gamma}_5 \circ \mathbf{\Gamma}_2 - \mathbf{\Gamma}_4 \circ \mathbf{\Gamma}_1 \tag{44}$$

And the step size for each pixel in the image is given by

$$\mathbf{\Gamma}_k = \mathbf{1}\left(\mathbf{\Gamma}_n ./ \mathbf{\Gamma}_d \right) \tag{45}$$

The iterative step at each pixel is calculated simultaneously by

$$\hat{\mathbf{F}}^{k+1} = \hat{\mathbf{F}}^{k} + \nabla_{\hat{\mathbf{F}}^{k}} \circ \mathbf{\Gamma}_{k} \tag{46}$$

Meeting the boundary constraints is also combined into the vectorized code process. The maximal allowed step size is simultaneously calculated for each boundary-exceeding fraction vector subject to

$$\mathbf{A}\hat{\mathbf{F}} \geq \mathbf{B}$$

where the columns of $\underset{2L\times(r\cdot c)}{\mathbf{B}} = \{\mathbf{b}, \mathbf{b}, \ldots, \mathbf{b}\}$ are duplication of the vector $\mathbf{b}$ of boundary values.

Testing the running time of the VCGDU relatively to FCLSU and SUnSAL reveals significant improvement. The results are shown in Fig 6.

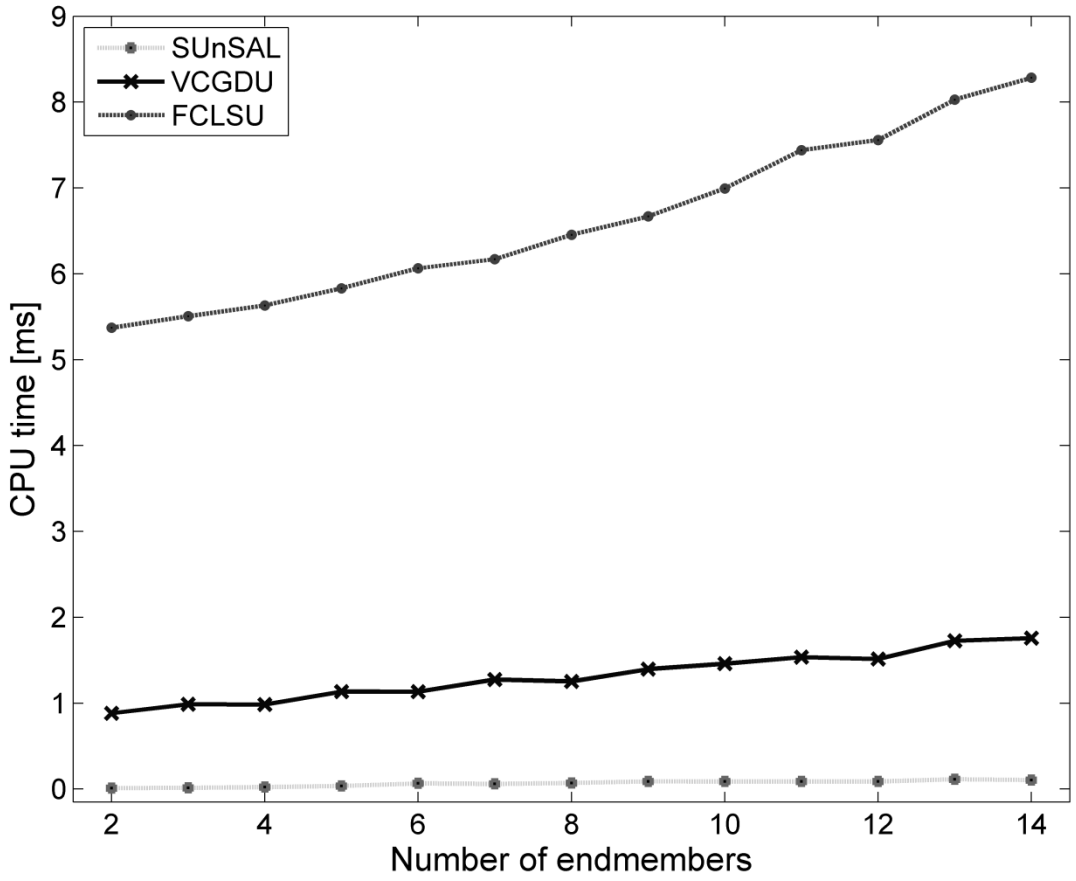

Fig. 6. Median run-times per-pixel (over 10 simulations) for VCGDU, FCLSU, and SUnSAL algorithms applied to 1000 mixed pixels as a function of number of EMs.

The new code version of the entire process is described by the pseudo-code below.

Algorithm 2: Vectorized Code Gradient Descent Unmixing (VCGDU)

Inputs: $\mathbf{E}(\lambda \times L)$: Matrix of set of EMs

$\mathbf{H}(r \times c \times \lambda)$: Cube of spectral image to be unmixed

$\hat{\mathbf{F}}^0(r \times c \times L)$: Cube of initial estimate of fraction image

1) Create $\mathbf{M}$ and $\hat{\mathbf{F}}$ by permuting $\mathbf{H}$ and $\hat{\mathbf{F}}^0$, respectively;

2) **while** $\left(\left(\max\left(\left|\hat{\mathbf{F}}^k - \hat{\mathbf{F}}^{k-t}\right|\right) > threshold\right) \vee \left(M \neq \varnothing\right)\right)$

/* The term $\left|\hat{\mathbf{F}}^k - \hat{\mathbf{F}}^{k-t}\right|$ indicates the amount of change in fraction values during the */

/* last *t* iterations, where *t* is a predefined parameter */

3) Calculate $\nabla_{\hat{\mathbf{F}}^k}$ and Γ_k by (30—37) and (38—45), respectively;

4) Apply active set strategy to ensure boundary constraints;

5) Set $\hat{\mathbf{F}}^{k+1} = \hat{\mathbf{F}}^k + \nabla_{\hat{\mathbf{F}}^k} \circ \Gamma_k$;

6) Remove from $\mathbf{M}$ all pixels for which process converged;

7) **end**

The results clearly reveal the superior efficiency of SUnSAL, whose run-time is almost invariant with respect to the number of EMs used in the unmixing. Thus SUnSAL can be used as reference for efficiency evaluation of newly proposed methods. Although VCGDU's run-time is considerably faster than (the original GDU version and) the standard, off-the-shelf FCLSU, it is about nine times slower than SUnSAL. Both VCGDU and SUnSAL use parallel processing to solve for the entire image. Whereas VCGDU solves for each pixel an independent optimization problem, running concurrently on all pixels, SUnSAL solves a single optimization problem applied to all the pixels simultaneously**.**

V. EXPERIMENTAL RESULTS AND DISCUSSION

Comparative performance evaluation of the new method was done relatively to the commonly used FCLSU and the state-of-art method SUnSAL [10]. We used an AISA image acquired over Hadera, Israel in 2006 (Fig. 10). In addition to experimenting with the real image, four different synthetic tests were created using a set of 14 image EMs derived from the image; see Fig. 7.

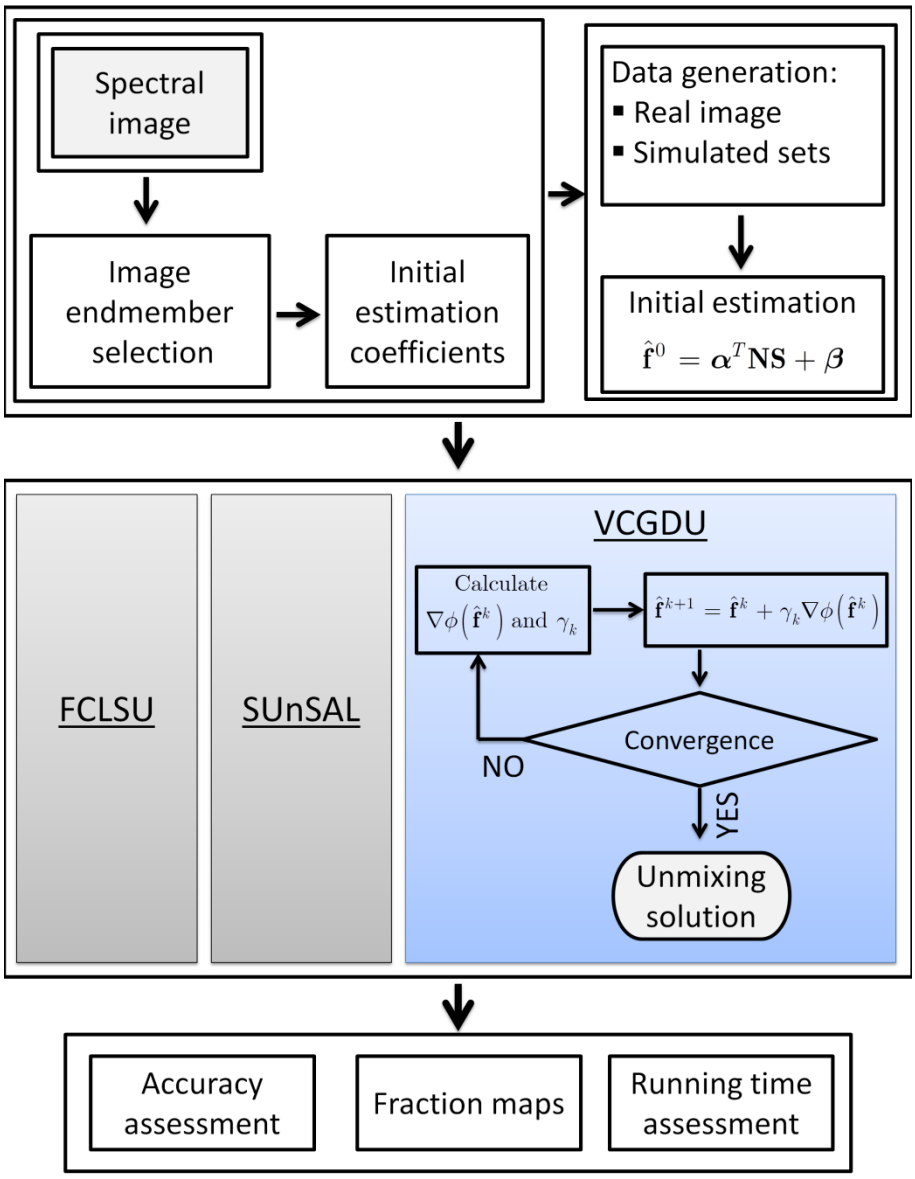

Fig. 7. Overview of comparative experimental study.

1) Experiment 1

A set of 10,000 synthetic mixed pixels was generated using the 14 EMs. The fractions in each pixel were picked at random and normalized to the sum of 1. An additive white Gaussian noise was added to the mixed spectra with SNRs of 1000 and 100. The well-known *root mean square error* (RMSE) [33] for each EM was calculated as follows:

$$RMSE_i = \sqrt{\frac{1}{N}\sum_{j=1}^{N}\left(f_{i,j} - \hat{f}_{i,j}\right)^2} \tag{47}$$

where *i* denotes the i^{th} EM, N is the number of pixels in the set (i.e., 10,000 in this case), and $f_{i,j}$ and $\hat{f}_{i,j}$ are the true and estimated fractions, respectively, of the i^{th} EM in the j^{th} pixel. The results are presented in Table I.

TA B L E I

RMSE values (per each EM used) for VCGDU, FCLSU, and SUnSAL

EM#	SNR=100			SNR=1000		
	VCGDU	**FCLSU**	**SUnSAL**	**VCGDU**	**FCLSU**	**SUnSAL**
1	0.1409	0.0882	**0.0871**	0.1031	0.0548	**0.0493**
2	**0.1038**	0.1211	0.1207	**0.0613**	0.0756	0.0748
3	**0.0898**	0.1220	0.1196	**0.0683**	0.0884	0.0785
4	**0.0889**	0.1188	0.1115	**0.0730**	0.1079	0.0876
5	**0.0932**	0.1101	0.1088	**0.0656**	0.0870	0.0791
6	**0.0895**	0.1094	0.1088	**0.0534**	0.0672	0.0631
7	**0.0896**	0.0987	0.0982	**0.0577**	0.0737	0.0695
8	**0.0927**	0.1036	0.1034	**0.0533**	0.0650	0.0635
9	**0.0954**	0.1271	0.1264	**0.0654**	0.0881	0.0851
10	**0.1018**	0.1188	0.1181	**0.0702**	0.0830	0.0801
11	**0.0726**	0.0840	0.0837	**0.0522**	0.0544	0.0524
12	**0.0728**	0.0793	0.0793	**0.0334**	0.0345	0.0344
13	**0.0759**	0.0885	0.0879	**0.0610**	0.0697	0.0647
14	0.1409	0.0882	**0.0871**	0.1031	0.0548	**0.0493**
mean	**0.0893**	0.1006	0.0994	**0.0598**	0.0689	0.0641
max	0.1409	0.1271	**0.1264**	0.1031	0.1079	**0.0876**

The RMSE values (per EM) for all the methods are correlative and increase consistently as the SNR decreases to 100. VCGDU is more accurate for 12 out of the 14 EMs while SUnSAL performs better for the other two EMs. This advantage is even more evident for SNR = 1000.

2) Experiment 2

This experiment is similar to the previous one, except that here each generated spectral mixture was multiplied by a random number between 0.75 and 1 to simulate a varying illumination effect. We evaluated again the unmixing performance, for each EM, by the RMSE expression in (47). The results are presented in Table II.

TA B L E II

RMSE values (per each EM used), under varying illumination effect, for VCGDU, FCLSU, and SUnSAL.

EM#	SNR=100			SNR=1000		
	VCGDU	**FCLSU**	**SUnSAL**	**VCGDU**	**FCLSU**	**SUnSAL**
1	**0.1417**	0.2339	0.2332	**0.1024**	0.2378	0.2359
2	**0.1026**	0.1309	0.1306	**0.0618**	0.0898	0.0899
3	**0.0899**	0.1023	0.1000	**0.0676**	0.0920	0.0920
4	**0.0879**	0.1186	0.1108	**0.0734**	0.1031	0.0984
5	**0.0935**	0.1000	0.0994	**0.0655**	0.0811	0.0777
6	**0.0894**	0.0925	0.0921	**0.0535**	0.0695	0.0671
7	0.0893	0.0824	**0.0822**	**0.0579**	0.0634	0.0624
8	**0.0921**	0.0936	0.0934	**0.0528**	0.0630	0.0623
9	**0.0945**	0.1095	0.1091	**0.0651**	0.0818	0.0805
10	**0.1021**	0.1284	0.1279	**0.0708**	0.0975	0.0965
11	**0.0734**	0.0813	0.0811	**0.0525**	0.0706	0.0702
12	0.0722	0.0677	**0.0677**	**0.0351**	0.0352	0.0352
13	**0.0763**	0.0860	0.0859	**0.0611**	0.0895	0.0896
14	**0.0421**	0.0422	0.0422	**0.0189**	0.0318	0.0318
mean	**0.0891**	0.1050	0.1040	**0.0599**	0.0861	0.0850
max	**0.1417**	0.2339	0.2332	**0.1024**	0.2378	0.2359

While the RMSE values from VCGDU remain roughly fixed, under the varying illumination effect, those from FCLSU and SUnSAL tend to generally increase. In some cases, though, they may decrease slightly,

especially for SNR = 100; however, the overall maximum and mean values increase, as can be seen from Tables I and II.

3) Experiment 3

We generated 13 synthetic image sets, each containing 10,000 mixed pixels. Each image is associated with a certain number of EMs (i.e., 2, 3, …, 14) composing the spectral mixture of each pixel. In other words, the first image consists of pixels containing only two EMs, the second image consists of pixels containing three EMs, and so on. Finally, the 13th image is composed of pixels containing all 14 EMs. The fractions in each pixel were picked at random and normalized to the sum of 1. An additive white Gaussian noise was added to the mixed spectra with SNRs of 1000 and 100. We evaluated the unmixing performance, for each synthetic image, by the following RMSE measure:

$$RMSE = \frac{1}{N}\sum_{j=1}^{N}\left(\sqrt{\frac{1}{L}\sum_{i=1}^{L}\left(f_{i,j}-\hat{f}_{i,j}\right)^2}\right) \tag{48}$$

where N = 10,000 is the number of pixels in each synthetic image, L is the number of EMs (14 in this case), and $f_{i,j}$ and $\hat{f}_{i,j}$ are the true and estimated fractions, respectively, of the i^{th} EM in the j^{th} pixel. Note that all 14 EMs are utilized during the unmixing, while taking into account the effect of non-participating EMs [34]. The experiment was then repeated by decreasing the number of spectral bands while increasing their width from 10 nm (72 bands) to 20 nm (48 bands) to 50 nm (24 bands). Fig. 8 shows the RMSE obtained for VCGDU, FCLSU, and SUnSAL, as a function of the actual number of participating EMs.

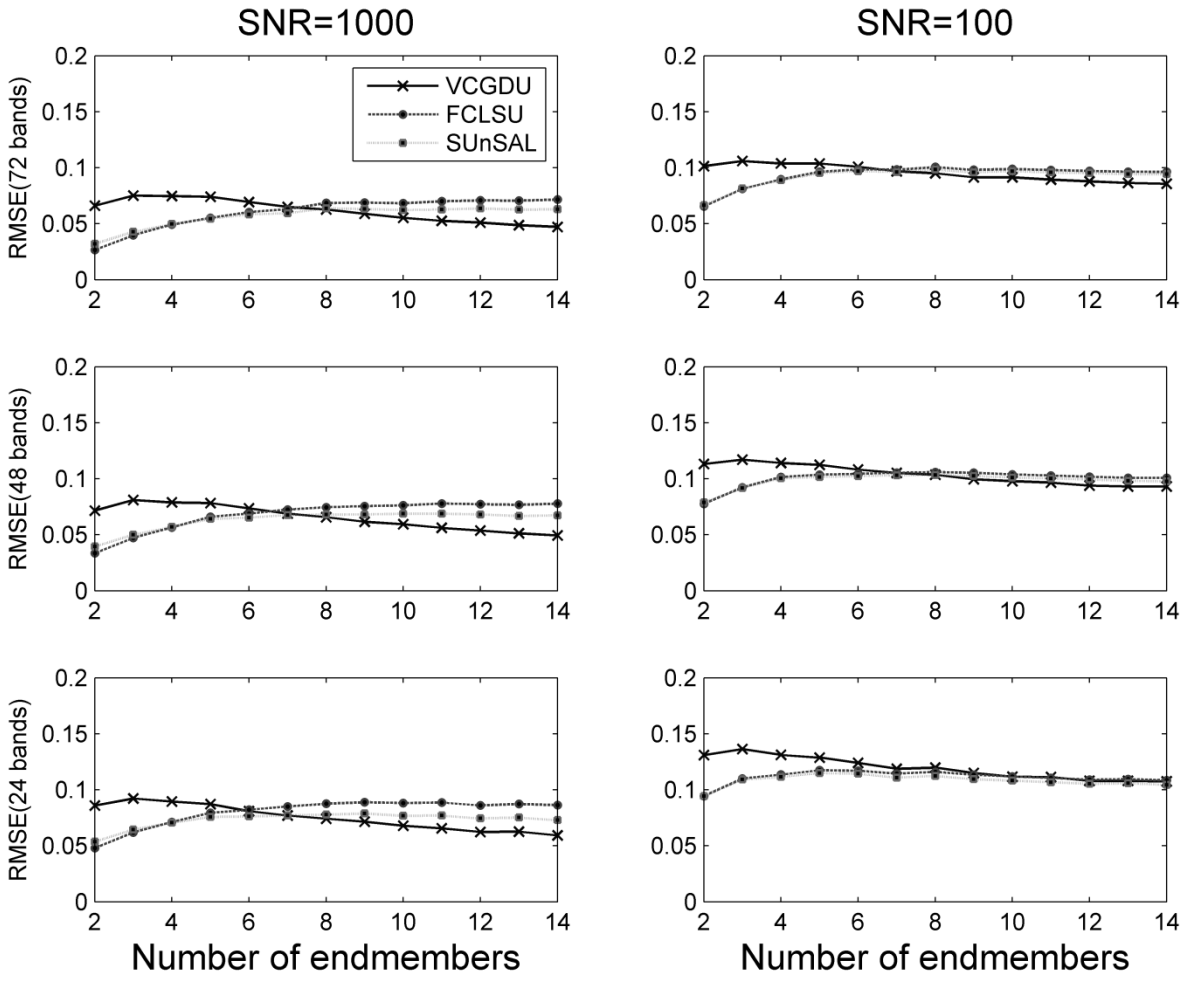

Fig. 8. RMSE values for VCGDU, FCLSU, and SUnSAL vs. actual number of EMs at two SNR levels and three spectral resolutions (bandwidth: 10, 20, and 40 nm).

The RMSE values of the three methods increase consistently as the SNR and the spectral resolution decrease. The results for FCLSU and SUnSAL are highly correlative (with some advantage to SUnSAL, especially for SNR = 1000). Both methods are advantageous if the number of actual EMs is smaller than six. For a larger number of EMs, VCGDU outperforms both FCLSU and SUnSAL, especially for SNR=1000.

4) Experiment 4

This experiment is similar to the previous one, except that here each generated spectral mixture was multiplied by a random number between 0.75 and 1 to simulate a varying illumination effect. As before, we evaluated the unmixing performance, for each synthetic image, in terms of the RMSE expression in (48).

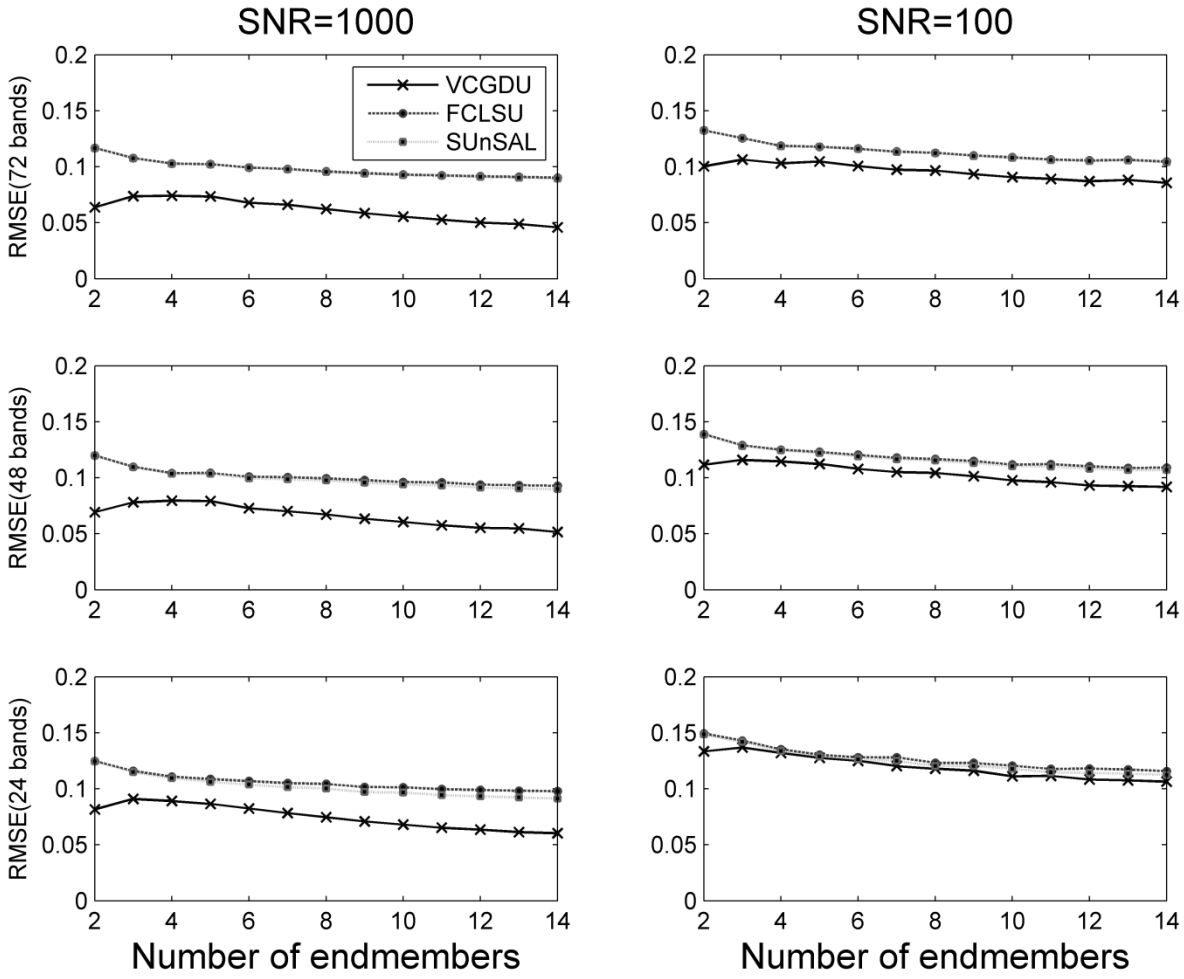

Fig. 9. RMSE values, under varying illumination effect, for VCGDU, FCLSU, and SUnSAL vs. actual number of EMs at two SNR levels and three spectral resolutions (bandwidth: 10, 20, and 40 nm).

As can be seen from Fig. 9, the robustness of VCGDU to the varying illumination effect yields RMSE values that are fairly fixed. Note, on the other hand, the increased RMSE values obtained for FCLSU and

SUnSAL. In other words, VCGDU significantly outperforms FCLSU and SUnSAL, under the varying illumination effect, especially for SNR = 1000.

5) Experiment 5 (using real data)

VCGDU, FCLSU, and SUnSAL were applied to the AISA real image (Fig. 10), considering 12 out of the 14 EMs selected manually from the image.

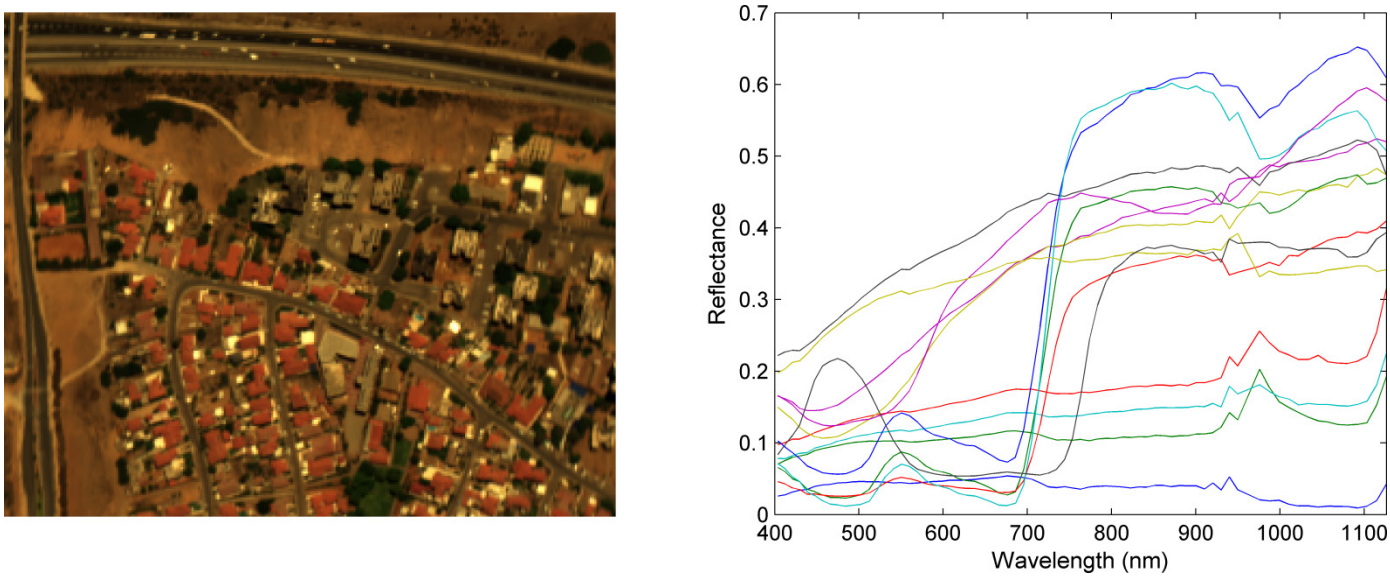

Fig. 10. RGB composite (left) and reflectance spectra (right) of the 14 EMs selected from the 2006 AISA image, containing 4 vegetation types, 4 soil/rock types, 3 kinds of pavement, and 3 roof types.

Figures 11 and 12 below show the estimated fraction maps for all EMs obtained by the methods.

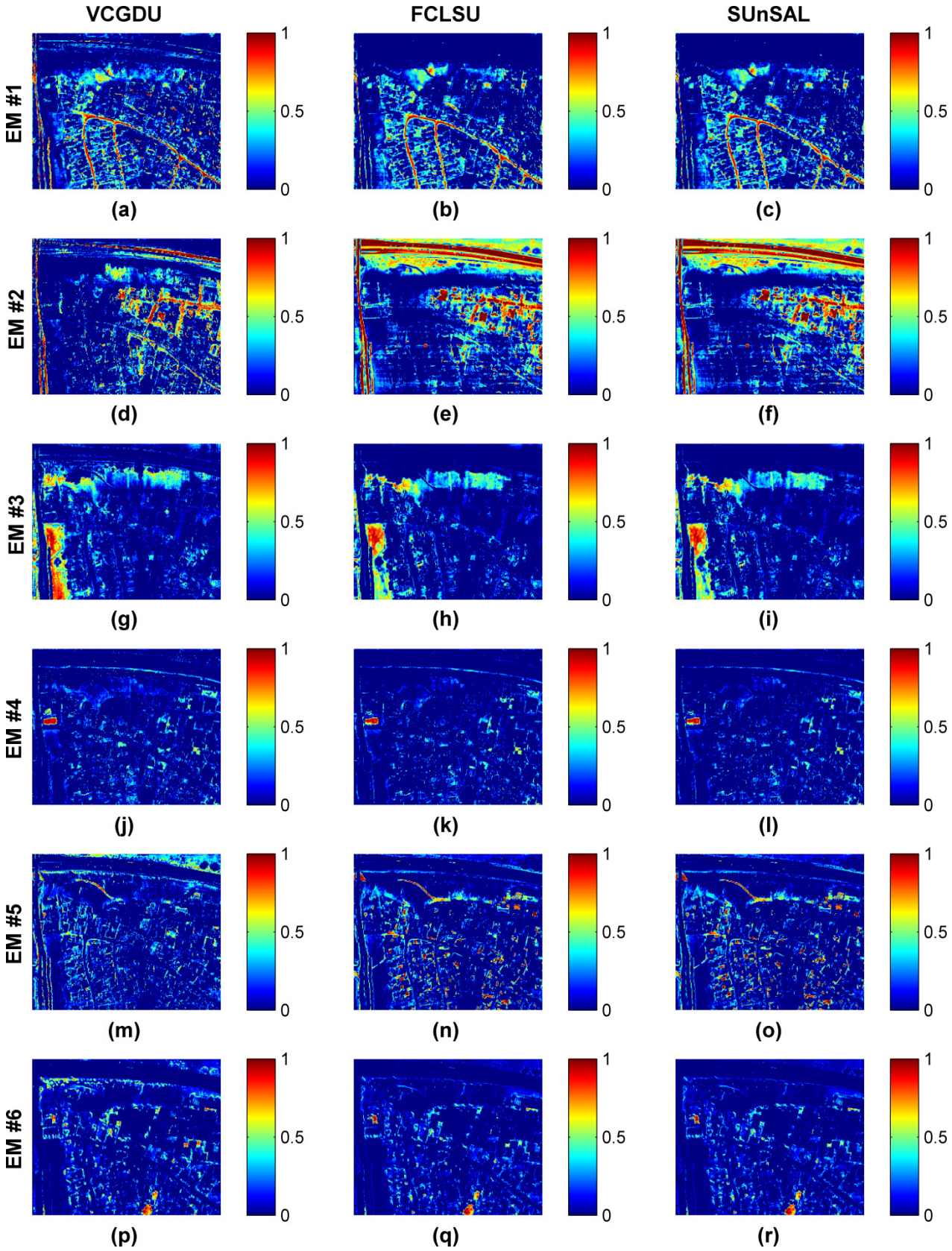

Fig. 11. Abundance fraction images for EMs 1, 2, 3, 4, 5, and 6 as estimated by VCGDU, FCLSU, and SUnSAL.

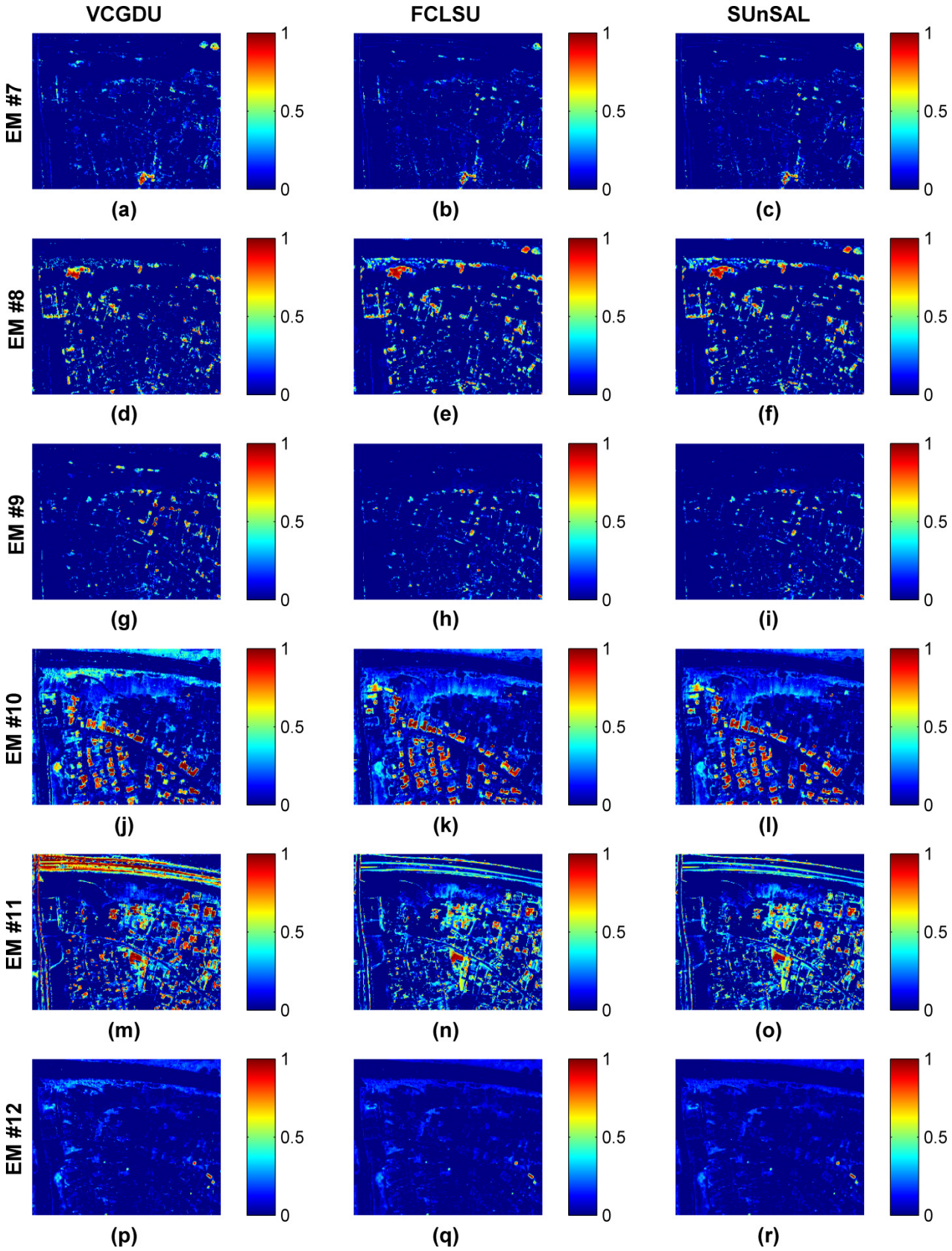

Fig. 12. Abundance fraction images for EMs 7, 8, 9, 10, 11, and 12 as estimated by VCGDU, FCLSU, and SUnSAL.

The results obtained by FCLSU and SUnSAL are nearly identical; also they largely agree with those due to VCGDU, modulo some differences in a few EM fractions. For example, FCLSU and SUnSAL overestimate the fractions of the second kind of asphalt in areas of dark soil as can be noticed in the EM #2 fraction maps (Fig. 11 (d), (e), and (f)). On the other hand, VCGDU overestimates the concrete fractions, especially along the main road; see EM #11 fraction maps (Fig. 14 (m), (n), and (o)).

VI. CONCLUSION AND FUTURE WORK

In this study, we presented a novel methodology for a fully constrained spectral unmixing, the *vectorized code gradient descent unmixing* (VCGDU). The newly proposed scheme performs iterative search in EM fraction space, based on analytical gradient descent optimization with respect to a variant of the SAM similarity measure as an objective function. The detailed derivation determines also analytically the optimal step size in each iteration and employs the active set strategy to fulfill the required constraints imposed on the fractions. The entire scheme was implemented via code vectorization, which is basically a special form of parallel computing. In particular, we showed how to "fold" the computational process (applied to an entire image) into a single loop of matrix operations. This results in a considerably more efficient performance of the fully constrained spectral unmixing proposed.

The methodology presented is capable of applying the unmixing process to a (relatively) large number of EMs, thereby taking advantage of the numerous amounts of available hyperspectral imagery. An assessment of the proposed scheme was done relatively to FCLSU and the fast state-of-the-art SUnSAL using simulated data and a real AISA spectral image. The experimental results suggest that the new methodology is advantageous when the number of actual EMs is larger than six, which is pretty common for urban, agricultural, and natural regions. Moreover, the new scheme improves significantly the unmixing

performance under the varying illumination effect due to the inherent advantage of the SAM-based objective function.

The suggested framework can be easily adapted to other objective functions, especially if they are differentiable with respect to the fraction vector, so that the gradient and the step size can be analytically formulated. Otherwise, the gradient and step size should be calculated numerically, but the running time is likely to increase, of course. Also, the VCGDU framework could be easily modified to handle any linear constrains that might be imposed on the fractions. For example, we show in [35] how to determine first a subset of EMs that are actually present in each pixel using novel spatial-spectral preprocessing; VCGDU is then applied with predefined fraction upper bounds of 0 to non-participating EMs in a given pixel. In summary, the algorithmic framework is rather modular and its components are easy to understand and implement.

For the sake of completeness, it would be of interest to assess the unmixing performance of a fully automated, end-to-end scheme, i.e., test the unmixing performance of our VCGDU-based module on an automatically extracted set of EMs. Also, as part of future research – it would be of interest to reduce further the run-time (beyond the run-time obtained by code vectorization). This could be done, for example, by applying the VCGDU simultaneously to a number of image sub-regions via parallel multi-core computing. In an attempt to improve the rate of convergence, it would also be of interest to apply methods, such as the conjugate gradient optimization, to the objective function presented and compare the accuracy and run-time obtained to those of VCGDU. Finally, one could investigate the influence of initialization on the unmixing performance.

REFERENCES

[1] J. J. Settle and N. A. Drake, “Linear mixing and the estimation of ground cover proportions,” International Journal of Remote Sensing, vol. 14, no. 6, pp. 1159–1177, 1993.

[2] N. Keshava and J. F. Mustard, “Spectral unmixing,” IEEE Signal Processing Magazine, vol. 19, no. 1, pp. 44–57, 2002.

[3] A. Plaza, Q. Du, J. Bioucas-Dias, X. Jia, and F. Kruse, “Foreword to the special issue on spectral unmixing of remotely sensed data,” IEEE Transactions on Geoscience and Remote Sensing, vol. 49, no. 1, pp. 4103–4110, 2011.

[4] C. A. Bateson and B. Curtiss, “A method for manual endmember selection and spectral unmixing,” Remote Sensing of Environment, vol. 55, pp. 229–243, 1996.

[5] M. Parente and A. Plaza, "Survey of geometric and statistical unmixing algorithms for hyperspectral images," in Proceedings of the Second IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, Reykjavik, Iceland, June 14—16, 2010, pp. 1—4, DOI: 10.1109/WHISPERS.2010.5594929

[6] M. E. Winter, “N-FINDR: An algorithm for fast autonomous spectral end-member determination in hyperspectral data,” in Proceedings of the SPIE Conference on Imaging Spectrometry V, Denver, Colorado, July 1999, vol. 3753, pp. 266–275.

[7] C.-I Chang, C.-C. Wu, W. Liu, and Y.-C. Ouyang, “A new growing method for simplex-based endmember extraction algorithm,” IEEE Transactions on Geoscience and Remote Sensing, vol. 44, no. 10, pp. 2804–2819, 2006.

[8] E. M. T. Hendrix, I. García, J. Plaza, G. Martín, and A. Plaza, "A new minimum-volume enclosing algorithm for endmember identification and abundance estimation in hyperspectral data," IEEE Transactions on Geoscience and Remote Sensing, vol. 50, no. 7, pp. 2744—2757, 2012.

[9] J. M. P. Nascimento and J. M. Bioucas-Dias, "Hyperspectral unmixing based on mixtures of Dirichlet components," IEEE Transactions on Geoscience and Remote Sensing, vol. 50, no.3, pp. 863—878, 2012.

[10] J. Bioucas-Dias and M. Figueiredo, "Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing," in Proceedings of the IEEE GRSS Workshop on Hyperspectral Image Signal Processing: Evolution in Remote Sensing, 2010, vol. 1, pp. 1–4.

[11] M. D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Collaborative sparse regression for hyperspectral unmixing," IEEE Transactions on Geoscience and Remote Sensing, vol. 52, no. 1, pp. 341–354, 2014.

[12] B. Natarajan, "Sparse approximate solutions to linear systems," SIAM Journal on Computing, vol. 24, no. 2, pp. 227–234, 1995.

[13] A. Plaza, P. Matrinez, R. Perez, and J. Plaza, "A quantitative and comparative analysis of endmember extraction algorithms from hyperspectral data," IEEE Transactions on Geoscience and Remote Sensing, vol. 42, no. 3, pp. 650–663, 2004.

[14] S. Sánchez, G. Martín, and A. Plaza, "Parallel implementation of the N-FINDR endmember extraction algorithm on commodity graphics processing units," in Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, Caceres, Spain, July 25–30, 2010, pp. 955–958.

[15] C. Gonzalez, D. Mozos, J. Resano, and A. Plaza, "FPGA implementation of the N-FINDR algorithm for remotely sensed hyperspectral image analysis," IEEE Transactions on Geoscience and Remote Sensing, vol. 50, no. 2, pp. 374–388, 2012.

[16] C.-I Chang, C.-C.Wu, C. S. Lo, and M. L. Chang, "Real-time simplex growing algorithms for hyperspectral endmember extraction," IEEE Transactions on Geoscience and Remote Sensing, vol. 48, no. 4, pp. 1834–1850, 2010.

[17] Y. E. Shimabukuro and J. A. Smith, "The least-squares mixing models to generate fraction images derived from remote sensing multispectral data," IEEE Transactions on Geoscience and Remote Sensing, vol. 29, pp. 16–20, 1991.

[18] D. Heinz and C.-I. Chang, "Fully constrained least squares linear mixture analysis for material quantification in hyperspectral imagery," IEEE Transactions on Geoscience and Remote Sensing , vol. 39, pp. 529–545, 2000.

[19] J. Boardman, "Inversion of high spectral resolution data," in Proceedings of the SPIE conference on Imaging Spectroscopy of the Terrestrial Environment, Orlando, Florida, April 16, 1990 vol. 1298, pp. 222–233.

[20] Y. E. Shimabukuro, “Shade images derived from linear mixing models of multispectral measurements of forested areas,” Ph.D. thesis, Department of Forest Wood Sciences, Colorado State University, Fort Collins, Colorado, 1987.

[21] E. A. Ashton and A. Schaum, “Algorithms for the detection of sub-pixel targets in multispectral imagery,” Photogrammetric Engineering & Remote Sensing, pp. 723–731, 1998.

[22] A. Halimi, Y. Altmann, N. Dobigeon, and J.-Y. Tourneret, "Unmixing hyperspectral images using the generalized bilinear model," in Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, Vancouver, Canada, July 24-29, 2011, pp. 1886—1889.

[23] J. Chen, C. Richard, H. Lantéri, C. Theys, and P. Honeine, “A gradient based method for fully constrained least-squares unmixing of hyperspectral images,” in Proceedings of the IEEE Statistical Signal Processing Workshop, Nice, France, June 2011, pp. 301–304.

[24] J. Chen, X. Jia, W. Yang, and B. Matsushita, "Generalization of subpixel analysis for hyperspectral data with flexibility in spectral similarity measures," IEEE Transactions on Geoscience and Remote Sensing, vol. 47, no. 7, pp. 2165-2171, 2009.

[25] N. Keshava, “A survey of spectral unmixing algorithms,” Lincoln Laboratory Journal, vol. 14, no. 1, pp. 55–78, 2003.

[26] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, “Hyperspectral unmixing overview: Geometrical, statistical and sparse regression-based approaches,” IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 5, no. 2, pp. 354–379, 2012.

[27] F. Tsai and W. D. Philpot, “Derivative analysis of hyperspectral data,” Remote Sensing of Environment, vol. 66, no. 1, pp. 41–51, 1998.

[28] Z. Rabah, I. Farah, G. Mercier, and B. Solaiman, “A new method to change illumination effect reduction based on spectral angle constraint for hyperspectral image unmixing,” IEEE Geoscience and Remote Sensing Letters, vol. 8, no. 6, pp. 1110–1114, 2011.

[29] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical leaning via the alternating direction method of multipliers," Foundations and Trends in Machine Learning, vol. 3, no. 1, pp. 1—122, Jan. 2011.

[30] M. Shoshany, F. Kizel, N. S. Netanyahu, N. Goldshlager, T. Jarmer, and G. Even-Tzur, "An iterative search in end-member fraction space for spectral unmixing," IEEE Geoscience and Remote Sensing Letters, vol. 8, no. 4, pp. 706-709, 2011.

[31] P. E. Gill, W. Murray, and M. H. Wright, "Practical Optimization," Academic Press, 1981.

[32] G. Shaw and H. Burke, “Spectral imaging for remote sensing,” Lincoln Laboratory Journal , vol. 14, no. 1, pp. 3 –28, 2003.

[33] F. J. García-Haro, S. Sommer, and T. Kemper, “A new tool for variable multiple end-member spectral mixture analysis (VMESMA),” International Journal of Remote Sensing, vol. 26, no. 10, pp. 2135–2162, 2005.

[34] D. M. Rogge, B. Rivard, J. Zhang, and J. Feng, “Iterative spectral un-mixing for optimizing per-pixel endmember sets,” IEEE Transactions on Geoscience and Remote Sensing, vol. 44, no. 12, pp. 3725–3736, 2006.

[35] F. Kizel, "Novel methods for stepwise analytical and spatially adaptive hyperspectral unmixing," Ph.D. thesis, Department of Civil and Environmental Engineering, Technion Israel Institute of Technology, Sept. 2014.