A Novel Hyperspectral Unmixing Method based on Least Squares Twin Support Vector Machines

Liguo Wang¹, San Wang², Xiuping Jia³ and Tianyi Bi⁴

¹College of Information and Communications Engineering, Harbin Engineering University, Harbin, China; ²School of Engineering and Information Technology, The University of New South Wales, Canberra, Australia; ³School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China

ABSTRACT

In hyperspectral images, endmembers characterizing one class of ground object may vary due to illumination, weathering, slight variations of the materials. This phenomenon is called intra-class endmember variability which is one of the important factors affecting the performance of unmixing. However, intra-class endmember variability is often ignored in unmixing, which causes a decrease in the accuracy of unmixing. How to deal with intra-class endmember variability is the focus. To address this problem, we propose a novel hyperspectral unmixing method based on Least Squares Twin Support Vector Machines (ULSTWSVM). ULSTWSVM uses multiple training samples (endmembers) to model a pure class, which takes intra-class endmember variability into account in unmixing. At the same time, ULSTWSVM obtains abundances by calculating the distances from the mixed pixels to the classification hyperplanes, which is simple and efficient. ULSTWSVM mainly comprises three steps: (1) to obtain the two non-parallel classification hyperplanes by solving two quadratic programming problems (QP) in least squares sense, (2) to calculate distances from the mixed pixels to classification hyperplanes, and (3) to normalize the distances and convert them to abundances. Experimental results on both synthetic and real hyperspectral data show that the proposed method outperforms the methods used for comparison.

Introduction

The hyperspectral remote sensing technology has been widely used in mineral exploration (Oskouei & Babakan, 2016), soil organic carbon mapping (Bayer et al., 2016) and vegetation monitoring (Stagakis et al., 2016). The spatial resolution of the hyperspectral images, however, is often low. Then, a pixel may contain more than one class of ground objects, and it is called a mixed pixel. Hyperspectral Unmixing (HU) techniques have been developed to address this problem (Bioucas-Dias et al., 2012).

Hyperspectral unmixing consists of extracting the spectral signatures (endmembers) of the pure materials existing in the scene, and estimating their proportions (abundances) at each pixel (Uezato et al., 2018). Some endmember extraction methods based on geometry have been developed since 1990’s. These methods assume that all the pixels belong to the same geometrical simplex. Vertices of the simplex are the corresponding endmembers. The representative algorithms include the pixel purity index (PPI) algorithm (Boardman, 2013), the vertex component analysis (VCA) algorithm (Nascimento & Bioucas-Dias, 2005), the Minimum Volume Simplex Analysis (MVSA) (J. Li & Bioucas-Dias, 2008), and the simplex identification via split augmented lagrangian algorithm (SISAL) (J.M. Bioucas-Dias, 2009). The methods mentioned above fail to cope with the effect of intra-class endmember variability, for they use a single endmember to represent a pure class.

Statistical unmixing methods provide a natural framework for representing variability in endmembers (Jin et al., 2019). The core idea of these algorithms is to deduce a posteriori probability density from the prior distribution of endmembers and abundances (Zhou et al., 2018). Then, different Bayesian estimation approaches are used to get the estimation of endmembers and abundances (Ma et al., 2019). Halimi et al. (2015) introduced a generalization of the normal compositional model by considering Gaussian variability for endmembers as well as an additive Gaussian noise for modelling errors. Bhatt et al. (2014) addressed a MAP solution with the data-dependent Huber–Markov random field. Li and Yin. (2013) advanced an Independent Component Analysis (ICA) based algorithm with variational Bayesian (VB) methods. Statistical unmixing algorithms usually come with a price: higher computational complexity, and the statistical distribution defined as a priori knowledge by the user may not conform to the nature of endmembers.
In recent years, sparse unmixing algorithms have been developed. They assume that the mixed pixels can be represented as a linear combination of multiple endmembers from known and available spectral libraries (Zhang et al., 2018). Unmixing in this way seeks the optimal subset of the spectral library that can best model each mixed pixel in the scene (Iordanche et al., 2012). Ma et al. (2017) adopt the $l_{2,1}$ norm loss function to handle noises and outliers. Then, sparse unmixing can be solved by the alternative direction method of multipliers (ADMM). Chen et al. (2018) took the combination of the spatial correlation and the global row sparsity into consideration and proposed an unmixing method based on collaborative sparsity and total variation. Zheng et al. (2019) synthesized spatial smoothness and spectral collaborative sparsity in unmixing. Sparse unmixing algorithms are affected heavily on the correctness of the model assumption used. Furthermore, considering different acquisition conditions between the spectral libraries and the mixed pixels, the unmixing result may be poor without accurate atmospheric correction algorithms.

Nonnegative Matrix Factorization (NMF) (Lee & Seung, 2000) has developed rapidly in the field of HU since it was proposed. Tong et al. (2017) proposed a novel region-based NMF (R-NMF) algorithm which aims at enforcing consistent abundances within homogeneous regions while discriminating the contribution from endmembers across regions. Fang et al. (2019) added sparsity and geometric structure constraints to the multilayer NMF structure and proposed a double-constrained multilayer NMF method. Peng and Tan (2020) adopted the Hessian graph regularizer in abundance matrix to mine the intrinsic geometry of each NMF. The problem of the unmixing methods based on NMF is that they are greatly affected by the initial value of the iteration, less robust to noise, and prone to local minimum.

Considering the intra-class endmembers variability, multi-endmember unmixing methods have become a hot topic of research. Roberts et al. (1998) proposed Multiple Endmember Spectral Mixture Analysis (MESMA), which allows the types and number of endmembers to vary on a per-pixel basis. All endmember combinations are tested, and according to the minimum reconstruction error criterion the optimal endmember combinations are found while abundances are obtained. MESMA can overcome the influence of endmember variability on unmixing, but it needs large computation time. Yang et al. (2015) brought forward an unmixing method based on Relevance Vector Machine (URVM), which is a sparse model in Bayesian framework. In this algorithm, RVM learning is transformed into maximum marginal likelihood function estimation, then the probability value of each sample’s corresponding class attribution is obtained, and is transformed into abundance. Because of the sparse model, unmixing time of URVM is shorter, while training time is longer. At the same time, because there is no obvious corresponding relationship between probability value and abundance, the unmixing error is larger. Wang and Jia (2009) proposed an unmixing method based on support vector machine (USVM). In USVM, the mixed pixels are between the boundary hyperplanes. Then, the abundances are obtained by calculating the distances from the mixed pixels to boundary hyperplanes. USVM uses distances to calculate abundances, which is simple and efficient. The disadvantages of USVM are as follows: (1) USVM is applicable to the data with parallel class boundary distribution; (2) When intra-class endmember variability is high, the accuracy of USVM decreases. Based on USVM, Wang et al. (2013) proposed an unmixing method based on Least Squares Support Vector Machine (ULSSVM). In ULSSVM, equality constraints replace inequality constraints in USVM, therefore, the solutions can be obtained directly from closed-form operation. As a result, the computational efficiency of ULSSVM is higher than that of USVM. However, ULSSVM has the same disadvantages as USVM.

Inspired by USVM and ULSSVM, we propose an unmixing method based on Least Squares Twin Support Vector Machines (ULSTWSVM). In ULSTWSVM, the abundances are obtained by calculating the distances from the mixed pixels to classification hyperplanes. The main difference between ULSTWSVM and USVM and ULSSVM is: ULSTWSVM uses the classification hyperplanes to calculate the distances to obtain the abundances, while USVM and ULSSVM use the boundary hyperplanes to calculate the distances to obtain the abundances. Compared with the boundary hyperplanes, the classification hyperplanes pass through the centroid of the class, which can better deal with intra-class endmember variability.

The main contributions of this paper lie in:

1) We extend LSTWSVM to hyperspectral unmixing, and propose a novel supervised unmixing method named ULSTWSVM. In ULSTWSVM, each pure class is modelled by a set of training samples (endmembers) instead of a single endmember to take the intra-class endmember variability into account.

2) In the framework of ULSTWSVM, a distance-based abundance calculation method is proposed. First, the training samples are used to obtain two classification hyperplanes, and then the distances from the mixed pixel to the classification hyperplanes are calculated and normalized to obtain the abundances. ULSTWSVM provides higher unmixing accuracy with a lower computational cost.

3) The nonlinear kernel function is introduced to ULSTWSVM and the method is named...
UNLLSTWSVM (the unmixing method based on nonlinear LSTWSVM). UNLLSTWSVM is valuable for unmixing when it is extended to cope with class data which have complicated class boundaries in the original feature space.

The remainder of this paper is organized, as follows. Section 2 introduces basic theory of LSTWSVM. Section 3 presents ULSTWSVM method. Section 4 reports the experimental results on synthetic data and real hyperspectral images. In Section 5, we further discuss the experimental results. Finally, the conclusions are drawn in Section 6.

**Basic theory of LSTWSVM**

TWSVM is a binary classifier. In contrast with conventional SVM solving a single larger-sized Quadratic Programming Problem (QPP) to obtain a single hyperplane, TWSVM generates two nonparallel hyperplanes by solving two smaller-sized QPPs (Kumar & Gopal, 2009). The geometric interpretation of TWSVM is shown in Figure 1, square data points belong to class +1, circular data points belong to class-1. \( H_1 \) represents the classification hyperplane of class+1, and \( H_2 \) represents the classification hyperplane of class-1. A new data point (red pentagon point) is classified as class +1 if it is close to \( H_1 \) or class -1 if it is close to \( H_2 \).

In the n-dimensional real space \( R^n \), let matrix \( A \in R_{n_1 \times n} \) represents the training samples set in class+1 and matrix \( B \in R_{n_2 \times n} \) represents the training samples set in class-1. \( n_1 \) and \( n_2 \) represent the number of class +1 training samples and class-1 training samples, respectively. The two non-parallel hyperplanes equations of TWSVM are shown below:

\[
x^T \omega^{(1)} + b^{(1)} = 0, \quad x^T \omega^{(2)} + b^{(2)} = 0
\]

where \( \omega^{(1)}, \omega^{(2)} \in R^n, \ b^{(1)}, b^{(2)} \in R \). Solving the following dual QPPs formulas (1) and (2), TWSVM classifier can be obtained.

\[
\text{TWSVM1: } \begin{aligned}
\min_{\omega^{(1)}, \omega^{(2)}} & \frac{1}{2} (A \omega^{(1)} + e_1 \omega^{(2)})^T (A \omega^{(1)} + e_1 \omega^{(2)}) + e_1 \omega^{(2)}^T q \\
\text{s.t.} & \quad (B \omega^{(3)} + e_2 \omega^{(2)})^T (B \omega^{(3)} + e_2 \omega^{(2)}) + e_2 \omega^{(2)}^T q \\
& \quad s.t. (A \omega^{(1)} + e_1 \omega^{(2)})^T q \geq e_1, \ e_2 \geq 0
\end{aligned}
\]

(1)

\[
\text{TWSVM2: } \begin{aligned}
\min_{\omega^{(1)}, \omega^{(2)}} & \frac{1}{2} (A \omega^{(1)} + e_1 \omega^{(2)})^T (A \omega^{(1)} + e_1 \omega^{(2)}) + e_1 \omega^{(2)}^T q \\
\text{s.t.} & \quad (B \omega^{(3)} + e_2 \omega^{(2)})^T (B \omega^{(3)} + e_2 \omega^{(2)}) + e_2 \omega^{(2)}^T q \\
& \quad s.t. (A \omega^{(1)} + e_1 \omega^{(2)})^T q \geq e_1, \ e_2 \geq 0
\end{aligned}
\]

(2)

where \( e_1 = (1, \ldots, 1)^T \in R^n \), \( e_2 = (1, \ldots, 1)^T \in R^n, \ c_1, c_2 > 0 \).

We modify formula (1) to least squares sense as formula (3) to get LSTWSVM, in which inequality constraints are replaced by equality constraints.

\[
\min_{\omega^{(1)}, \omega^{(2)}} \frac{1}{2} (A \omega^{(1)} + e_1 \omega^{(2)})^T (A \omega^{(1)} + e_1 \omega^{(2)}) + \frac{c_1}{2} q^T q.
\]

(3)

\[\text{s.t.} \quad (B \omega^{(3)} + e_2 \omega^{(2)})^T (B \omega^{(3)} + e_2 \omega^{(2)}) + e_2 \omega^{(2)}^T q = e_2
\]

Substituting equality constraints into the objective function, QPP formula (3) becomes:

\[
\min_{\omega^{(1)}, \omega^{(2)}} \frac{1}{2} \|A \omega^{(1)} + e_1 \omega^{(2)}\|^2 + \frac{c_1}{2} \|B \omega^{(3)} + e_2 \omega^{(2)}\|^2
\]

(4)

Setting the gradient of formula (4) with respect to \( \omega^{(1)} \) and \( \omega^{(2)} \) to zero, we get:

\[
A^T (A \omega^{(1)} + e_1 \omega^{(2)}) + e_1 B^T (B \omega^{(3)} + e_2 \omega^{(2)}) = 0 e_1, \\
\]

(5)

**Figure 1.** Geometric interpretation of TWSVM.
\[ e_1^T (A \omega^{(1)} + e_1 b^{(1)}) + c_1 e_2^T (B \omega^{(1)} + e_2 b^{(1)} + e_2) = 0, \]  
(6)

where \( e = (1, \ldots, 1)^T \in \mathbb{R}^n \). Arranging formulas (5) and (6) in matrix form and solving for \( \omega^{(1)} \) and \( b^{(1)} \), we get:

\[
\begin{bmatrix}
\omega^{(1)} \\
b^{(1)}
\end{bmatrix} = 
\begin{bmatrix}
(B^T B + \frac{1}{c_1} A^T A) + \frac{1}{c_1} B^T e_2 \\
-m_2 \end{bmatrix}^{-1} 
\begin{bmatrix}
B^T e_2 \\
-1
\end{bmatrix}
\]

Defining \( E = [A \ e_1] \), \( F = [B \ e_2] \), the solution of QPP formula (1) is as follows:

\[
\begin{bmatrix}
\omega^{(1)} \\
b^{(1)}
\end{bmatrix} = -(E^T F + \frac{1}{c_1} E^T E)^{-1} E^T e_2
\]

In the same way, the solution of QPP formula (2) is as follows:

\[
\begin{bmatrix}
\omega^{(2)} \\
b^{(2)}
\end{bmatrix} = (E^T E + \frac{1}{c_2} F^T F)^{-1} E^T e_1
\]

Once \( \omega^{(1)}, b^{(1)}, \omega^{(2)}, b^{(2)} \) are obtained, the two non-parallel hyperplanes are determined. The classification function is designed as:

\[
f(x) = \begin{cases}
+1 & \text{if } x^T \omega^{(1)} + b^{(1)} > 0 \\
-1 & \text{if } x^T \omega^{(2)} + b^{(2)} > 0
\end{cases}
\]

In the following section, we introduce the kernel function to LSTWSVM. The two non-parallel hyperplanes equations are shown below:

\[ K(x^T, C^T)u^{(1)} + b^{(1)} = 0, \quad K(x^T, C^T)u^{(2)} + b^{(2)} = 0 \]

where \( C^T = [AB]^T \), and \( K \) is any arbitrary kernel. The original QPPs of TWSVM are modified to 2 norm form of slack variables and inequality constraints replaced by equality constraints as show in formula (7), (8).

\[
\begin{align*}
\min_{u^{(1)}, b^{(1)}} & \quad \frac{1}{2} \|K(A, C^T)u^{(1)} + e_1 b^{(1)}\|^2 + \frac{c_1}{2} q^T q \\
\text{s.t.} & \quad -(K(B, C^T)u^{(1)} + e_2 b^{(1)}) + q = e_2
\end{align*}
\]

\[
\begin{align*}
\min_{u^{(2)}, b^{(2)}} & \quad \frac{1}{2} \|K(B, C^T)u^{(2)} + e_2 b^{(2)}\|^2 + \frac{c_2}{2} q^T q \\
\text{s.t.} & \quad (K(A, C^T)u^{(2)} + e_1 b^{(2)}) + q = e_1
\end{align*}
\]

By substituting equality constraints into the objective function, formulas (7) and (8) become:

\[
\begin{align*}
\min_{u^{(1)}, b^{(1)}} & \quad \frac{1}{2} \|K(A, C^T)u^{(1)} + e_1 b^{(1)}\|^2 + \frac{c_1}{2} \|K(A, C^T)u^{(1)} + e_2 b^{(1)} + e_2\|^2 \\
\min_{u^{(2)}, b^{(2)}} & \quad \frac{1}{2} \|K(B, C^T)u^{(2)} + e_2 b^{(2)}\|^2 + \frac{c_2}{2} \|K(A, C^T)u^{(2)} - e_1 b^{(2)} + e_1\|^2
\end{align*}
\]

The Solutions of QPPS formulas (9) and (10) can be acquired as:

\[
\begin{align*}
\begin{bmatrix}
u^{(1)} \\
b^{(1)}
\end{bmatrix} = -(H^T H + \frac{1}{c_1} G^T G)^{-1} H^T e_2
\end{align*}
\]

\[
\begin{align*}
\begin{bmatrix}
u^{(2)} \\
b^{(2)}
\end{bmatrix} = (G^T G + \frac{1}{c_2} H^T H)^{-1} G^T e_1
\end{align*}
\]

where \( G = [K(A, C^T) e_1] \), \( H = [K(B, C^T) e_2] \).

Once \( u^{(1)}, b^{(1)}, u^{(2)}, b^{(2)} \) are obtained, the two hyperplanes are determined. The classification function is designed as:

**Unmixing method based on LSTWSVM**

**Proposed approach**

For convenience, we rearrange the \( h \) rows, \( l \) columns, and \( n \) bands of a hyperspectral data cube into \( n \) rows, \( h \) columns (\( h = k \times l \)) matrix \( X = [x_1, x_2, \ldots, x_{h l}] \), called image matrix, where \( h \) is the number of pixels in the image, and \( x_i \) is a column vector representing a pixel.

Ignoring the random noise, the linear spectral mixture model can be described as: \( X = M \times R \), where \( M = [m_1, m_2, \ldots, m_p] \), in which \( m_i \) is the \( j \)th endmember vector and \( p \) is the number of endmembers. \( R \) is a \( p \times h \) matrix, where its element \( r_{ji} \) (\( i = 1, 2, \ldots, p \) and \( j = 1, 2, \ldots, h \)) is the abundance of endmember \( m_j \) in pixel \( x_i \). So

\[ x_i = \sum_{j=1}^{p} r_{ji} m_j \]

When \( p > 2 \), the classification problem is not binary, and becomes a multi-classification problem. The convenient "one against rest" strategy is used in this study.

As shown in Figure 2, when SVM is used for unmixing (USVM), the margins between \( H_j \) and \( H_j \) are recognized as the mixed region. \( H_j \) and \( H_j \) are parallel and formed by the pixels on the class boundaries (support vectors), called boundary hyperplanes. The two regions above \( H_j \) or below \( H_j \) are related to the pure pixels (training samples). The abundance \( r_{ji} \) of the mixed pixel \( x_i \) can be obtained by formula (13).

\[
r_{ji} = \frac{D_j(x_i)}{D_j(x_i) + D_j(x_i)}
\]

where \( D_j(x_i) \) represents the distance from \( x_i \) to \( H_j \), \( D_j(x_i) \) represents the distance from \( x_i \) to \( H_j \).}

Inspired by USVM, we propose unmixing method based on LSTWSVM, named ULSTWSVM. As shown in Figure 3, \( H_j \) and \( H_j \) are nonparallel and pass through the centroids of
classj and classj, respectively. \( H_j \) and \( H_j \) are called classification hyperplanes. The abundance \( r_{ji} \) of the mixed pixel \( x_i \) can be obtained by formula (13), where \( D_j(x_i) \) represents the distance from \( x_i \) to classification hyperplane \( H_j \). \( D_j(x_i) \) represents the distance from \( x_i \) to classification hyperplane \( H_j \). When \( r_{ji} \) is obtained by formula (13), \( x_i \) can be expressed as:

\[
x_i = \sum_{j=1}^{p} r_{ji} m_j = \sum_{j=1}^{p} \frac{D_j(x_i)}{D_j(x_i) + D_j(x_i)} m_j.
\]  

(14)

**Multiple endmembers replacing single endmember in ULSTWSVM**

Some traditional unmixing methods may get poor results as they ignore intra-class endmembers
variability. ULSTWSVM overcomes the shortcoming by using multiple pure pixels (training samples) to model a class. But formula (14) introduces rigid expression of single endmember again, so the single endmember should be replaced by multiple endmembers (training samples).

For binary class unmixing problems, two pure pixels corresponding to two classes are mixed with known proportions to generate a mixed pixel. In this manner, the two pure pixels and the corresponding fractional abundances related to each mixed pixel are available (Wang et al., 2013). In this case, each mixed pixel \( x_i \) has relevant specific endmembers combination \( \{ m_1^i, m_2^i \} \). For multiple class unmixing problems, let the endmembers combination corresponding to \( x_i \) be \( \{ m_1^i, m_2^i, \ldots, m_p^i \} \) with abundances \( \{ r_{1i}, r_{2i}, \ldots, r_{pi} \} \), then \( x_i = \sum_{j=1}^{p} r_{ji} m_j^i \). According to “one against rest” strategy, multiple classes unmixing problems can be solved by dividing them into several binary classes unmixing ones. For the \( kth (1 \leq k \leq p) \) sub-unmixing model, \( m_k^i \) is the first corresponding endmember and its abundance is \( r_{ki} \). The other endmember is written as \( m_k^i \), with abundance \( r_{ki} = 1 - r_{ki} \), where \( m_k^i = (x_i - r_{ki} m_k^i) / (1 - r_{ki}) \). In this case, \( x_i = r_{ki} m_k^i + r_{ki} m_k^i \).

**Application of kernel function**

We call the function of computing the inner product of two vectors in the space after implicit mapping as kernel function. In linear mixing model, the mixed pixel is linear combination of endmembers weighted by the corresponding abundance fractions. Therefore, we use formula (15) as linear kernel function in ULSTWSVM. That is named UNLLSTWSVM (the unmixing algorithm based on linear LSTWSVM).

\[
K(x, y) = x^T y. \tag{15}
\]

Considering scattering, refraction, et al, the mixed pixel in real hyperspectral imaging contains many nonlinear factors. Therefore, we use formula (16) as nonlinear kernel function in ULSTWSVM. That is named UNLLSTWSVM (the unmixing algorithm based on nonlinear LSTWSVM).

\[
K(x, y) = \exp\left(-\frac{||x - y||^2}{2\sigma^2}\right). \tag{16}
\]

After introducing the kernel function, formula (13) can be rewritten as:

\[
r_{ji} = \frac{|K(x_j, C^i)u^{(j)} + b^{(j)}|}{|K(x_j, C^i)u^{(j)} + b^{(j)}| + |K(x_j, C^i)u^{(j)} + b^{(j)}|}.
\]

(17)

**Algorithm summary**

We summarize the main steps of ULSTWSVM as follows:

**Input:**

Hyperspectral image \( X = [x_1, x_2, \ldots, x_i, \ldots, x_h] \), where \( X \in R^{n \times h} \), \( n \) is the number of bands, and \( h \) is the number of mixed pixels. Each column vector \( x_i \) represents a mixed pixel (\( i = 1, 2 \cdots h \)).

**Output:** Abundance matrix \( R = [r_{ji}] \), where \( r_{ji} \geq 0 \) and \( \sum_{j=1}^{p} r_{ji} = 1, j = 1, 2 \cdots p \), \( p \) is the number of endmember classes.

**Step 1:** Evenly selecting pixels from each class to form training samples set \( S \), where \( S = \{ S_1, S_2, \cdots S_j, \cdots, S_p \} \), \( S_j \) is class \( j \) training samples set.

**Repeat1:** \( j = 1 : p \)

Let \( A = [S_j] \), then \( A \in R^{n \times m_i} \), \( m_1 \) is the number of endmembers in \( A \). According to “one against rest” strategy, let \( B = [S_1, S_2, \cdots S_{j-1}, S_{j+1}, \cdots S_p] \), which is taken as endmembers class \( j \) training samples set. Then \( B \in R^{n \times m_j} \), and \( m_2 \) is the number of endmembers in \( B \).

**Step 2:** Applying the kernel function \( K \) to calculate \( H = [K(B, C^i) e_1], \quad G = [K(A, C^T) e_1] \), where \( C^T = |AB| \), \( e_1 = (1, \cdots, 1)^T \in R^{m_1} \), \( e_2 = (1, \cdots, 1)^T \in R^{m_2} \).

**Step 3:** Applying formula (11) to calculate \( u^{(j)} \) and \( b^{(j)} \), where \( u^{(j)} \in R^{(m_1 + m_2)} \) and \( b^{(j)} \in R \).

**Step 4:** Applying formula (12) to calculate \( u^{(0)} \) and \( b^{(0)} \), where \( u^{(0)} \in R^{(m_1 + m_2)} \) and \( b^{(0)} \in R \).

**Repeat2:** \( i = 1 : h \)

**Step 5:** Applying formula (17) to calculate the abundance \( r_{ji} \).

**End Repeat2.**

**End Repeat1.**

**Step 6:** According to sum-to-one constraint, elements in each column in \( R = [r_{ji}] \) are normalized to [0, 1] to get the abundance matrix \( R \).

Next, we study the complexity of ULSTWSVM. Assuming the inner product operation of two phasors is the basic operation \( O(1) \). The computational cost of Step 2 is \( O(m_2(m_1 + m_2)) + m_1(m_1 + m_2)) \). According to formulas (11) and (12), Step 3 and Step 4 each performed one matrix inverse operation. Compared with computational cost of Step 2, the computational cost of Step 3 and Step 4 is negligible. The computational cost of Repeat2 is
\[ O((m_1 + m_2)h). \] Since \((m_1 + m_2) \ll h\), the computational cost of per-iteration in Repeat1 is 
\[ O((m_1 + m_2)(m_1 + m_2 + h)) = O((m_1 + m_2)h). \]

Reviewing the above process, we conclude that the computational complexity of ULSTWSVM is dominated by 
\[ O((m_1 + m_2)h p), \]
where \((m_1 + m_2)\) is the total number of training samples.

**Experiments and results**

**Hyperspectral remote sensing datasets**

Four hyperspectral data sets were used in the experiments. The first data set is Indian Pines Dataset, which was gathered by the AVIRIS sensor. The hyperspectral image has a spatial size of 145 \times 145 pixels and contains 16 classes of ground objects. The actual number of spectral bands used in the experiments is 200. The second data set is the U.S. Geological Survey (USGS) spectral library, which contains 498 spectra, each with 224 bands. The third data set is Salinas-A, which was collected by the 224-band AVIRIS sensor over Salinas Valley, California. Salinas-A comprises 86 \times 83 pixels and includes six classes of ground objects. The actual number of experimental bands is 204. The last data set is Pavia University, which was acquired by the ROSIS sensor. An area with a size of 144 \times 144 pixels is used for test, which mainly covers eight classes of ground objects. The actual number of spectral bands used in the experiments is 103.

**Unmixing performance evaluation method**

The performance of spectral unmixing results is usually evaluated by RMSE (Root Mean Square Error). The RMSE of the abundances corresponding to class \( j \) is defined as following:

\[
RMSE = \sqrt{\frac{1}{h} \sum_{i=1}^{h} (r_{ji} - \hat{r}_{ji})^2},
\]

where \( r_j \) is real abundance, \( \hat{r}_j \) is estimated abundance, and \( h \) is the total number of pixels. The Mean RMSE for all classes is defined as following:

\[
\text{Mean RMSE} = \sqrt{\frac{1}{p} \sum_{j=1}^{p} \frac{1}{h} \sum_{i=1}^{h} (r_{ji} - \hat{r}_{ji})^2},
\]

where \( p \) is the number of class.

In addition, we use the reconstruction error (RE) to evaluate the performance of unmixing.

\[
RE = \frac{1}{lnn} \sum_{i=1}^{h} \left\| (x_i - \hat{x}_i) \right\|_2^2,
\]

where \( x_i \) and \( \hat{x}_i \) are the actual and reconstructed spectrum of the \( i \)th pixel.

**Experiments on synthetic data**

**Experiment 1**

The purpose of the first experiment is to compare the traditional unmixing algorithms and ULSTWSVM (the unmixing algorithm based on linear LSTWSVM) by using synthetic data. Some traditional unmixing algorithms include PPI/FCLS, VCA/FCLS, MVSA/FCLS, SISAL/FCLS. These algorithms firstly adopt PPI, VCA, MVSA, SISAL in the endmembers extracting, then use Fully Constrained Least Squares (FCLS) method to unmix the mixed pixels. In addition, NMF algorithm (initial endmembers are extracted by VCA) and ICA algorithm are included in the comparison.

To generate synthetic data, 500 pixels are randomly selected from Soybeans-min and Woods in Indian Pines data as endmembers, respectively. Each class of endmembers are divided into 10 groups, and therefore each group contains 50 endmembers, written as \( a((k-1) \times 50+j) \) and \( b((k-1) \times 50+j), k=1, 2, \ldots, 10 \) and \( j=1, 2, \ldots, 50 \). Given these 50 endmembers increasingly corresponding abundance \( a_{kj}=k \times 0.1 - 0.05 \), 500 mixed pixels were generated by using formula (18):

\[
X((k-1) \times 50+j) = a_{kj} \times a((k-1) \times 50+j) + (1 - a_{kj}) \times b((k-1) \times 50+j), k=1, 2, \ldots, 10, j=1, 2, \ldots, 50
\]

(18)

Using this method, we can get synthetic mixed pixels and know the real abundance information. It should be emphasized that each mixed pixel is composed of two different endmembers randomly, which simulates intra-class endmembers variability. The traditional unmixing algorithms mentioned above are used to unmix the synthetic mixed pixels. Meanwhile, 50 pixels were evenly selected from Soybean-min and Woods as training samples, and the abundances of synthetic-mixed pixels were obtained by ULSTWSVM.

The abundance maps are shown in Figure 4. The abundance maps of VCA/FCLS and ICA have obvious deviation from the real one, which means the performance of VCA/FCLS and ICA is poor. The abundance maps of PPI/FCLS, MVSA/FCLS, SISAL/FCLS, NMF are similar to the real one, the unmixing results of these algorithms are ideal. The abundance map of ULSTWSVM has close similarity with the real one, especially in the light-coloured area in the centre of the abundance map. Compared with the other algorithms, ULSTWSVM produces more accurate abundance estimation.

Table 1 gives RMSEs, REs and computational time of seven algorithms. The RMSEs of PPI/FCLS, MVSA/FCLS and SISAL/FCLS are similar, about 0.1. NMF, VCA/FCLS perform worse. Their RMSEs are 0.1246 and 0.1476. The RMSE of ICA is the largest, 0.2276. ULSTWSVM produces the minimum RMSE, which
is 0.0619. At the same time, ULLSTWSVM produces the minimum RE, which is 0.0568.

The computational time is also listed in Table 1. PPI/FCLS, VCA/FCLS, MVSA/FCLS, SISAL/FCLS, take more than 2s, about 40 times the time required by ULLSTWSV at least. This is because these algorithms are more time consuming to extract the endmembers first and then obtain the abundances. NMF takes 2.4180s, a time much longer than 0.0556s of ULLSTWSVM. This is due to the complex iteration of NMF. Except ULLSTWSVM, ICA has the shortest computational time which is 1.4417s. But the RMSE of ICA is 0.1657 larger than that of ULLSTWSVM. To sum up, the proposed ULLSTWSVM outperforms the comparison algorithms.

**Experiment 2**

In this experiment, we use three classes of endmembers to synthesize mixed pixels to evaluate the unmixing effect of ULLSTWSVM. The method of abundance generation refers to Miao and Qi (2007). Three 25 x 20 abundance matrices are generated by low pass averaging filter. 500 pixels are selected randomly from each class of Corn-notill, Grass/Trees, Woods in Indian Pines data set as endmembers which are used to generate the synthetic pixels.

In the experiment, we compare Multiple Endmember Spectral Mixture Analysis (MESMA) introduced by Roberts et al. (1998) with our proposed ULLSTWSVM. MESMA is a representative multiple endmembers unmixing algorithm and has been widely used in practice. 15 pixels are evenly selected from Corn-notill, Grass/Trees and Woods data in Indian Pines (5 pixels for each class) to compose the reference spectra library, then the unmixing results are obtained by MESMA. For the fair comparison, the pixels in reference spectral library are used as training samples, and the unmixing results are obtained by

![Abundance maps of seven methods.](image)

Figure 4. Abundance maps of seven methods. (a) real abundance, (b) PPI/FCLS, (c) VCA/FCLS, (d) MVSA/FCLS, (e) SISAL/FCLS, (f) NMF, (g) ICA, and (h) ULLSTWSVM.

| algorithms     | PPI/FCLS | VCA/FCLS | MVSA/FCLS | SISAL/FCLS | NMF   | ICA   | ULLSTWSVM |
|----------------|----------|----------|-----------|------------|-------|-------|-----------|
| RMSE           | 0.1033   | 0.1476   | 0.1019    | 0.0975     | 0.1246| 0.2276| 0.0619    |
| RE             | 0.1118   | 0.0888   | 0.0694    | 0.0684     | 0.0908| 0.3221| 0.0568    |
| TIME(s)        | 2.3008   | 2.0328   | 2.4817    | 2.4456     | 2.4180| 1.4417| 0.0556    |
ULLSTWSVM. Similarly, we repeat the experiments using 30 pixels (10 pixels for each class) and 45 pixels (15 pixels for each class) from Corn-notill, Grass/Trees and Woods. Table 2 lists the quantitative results of MESMA and ULLSTWSVM unmixing synthetic data.

The results show that when the number of reference spectra is 15, the RMSEs of unmixing the three endmembers by ULLSTWSVM are obviously smaller than those by MESMA, and the Mean RMSE of ULLSTWSVM is 0.0912 less than that of MESMA. The RE of ULLSTWSVM is 0.005 less than that of MESMA. When the number of reference spectra is 30 and 45, the RMSEs of unmixing Corn-notill and Grass/Trees by ULLSTWSVM are smaller than those by MESMA. The RMSE of unmixing Woods by ULLSTWSVM is slightly larger than that by MESMA. And the Mean RMSEs of ULLSTWSVM are less than those of MESMA. When the number of reference spectra is 30, the RE of ULLSTWSVM is 0.0015 less than that of MESMA. When the number of reference spectra is 45, the RE of MESMA is 0.0049 less than that of ULLSTWSVM.

Table 2 lists the computational time of the two algorithms. MESMA is of heavy computational load, and the computational time of MESMA increases rapidly with the increase of the number of reference spectra. Compared with MESMA, ULLSTWSVM has a much shorter computational time. When the number of reference spectra number is larger, the unmixing performance is better, but the computational load would be too heavy to be practically implemented. ULLSTWSVM takes reference spectra as training samples to obtain classification hyperplanes. Then, abundances are obtained by calculating the distances from the mixed pixels to the classification hyperplanes. Compared with MESMA, ULLSTWSVM has the advantages of high accuracy and low computational burden.

**Experiment 3**

The purpose of this experiment is to use USGS spectral library to synthesize mixed pixels to evaluate ULLSTWSVM, URVM (unmixing algorithm based on relevance vector machine) and ULLSSVM (unmixing algorithm based on Linear Least Squares Support Vector Machine). Three classes of spectra, Olivine, Hematite and Muscovite, were chosen from the USGS spectral library to run the experiment. The numbers of within-class spectra were 17, 12 and 13, respectively. Then, the abundances selected from the abundance matrices generated in Experiment 2 and three randomly selected spectrum (one from each class) were linearly mixed to generate each mixed pixel. 500 mixed pixels were synthesized for the experiment. 6 spectra were evenly selected from each class as training samples.

Figure 5 shows the abundance of maps of the three classes. Each row represents the abundances of one class. From top-to-bottom, the figure shows the abundances of Olivine, Hematite, and Muscovite. Each column stands for different algorithms. From left-to-right, they are the results from real abundance, URVM, ULLSSVM, and ULLSTWSVM. As can be observed from abundance maps, the abundance map of URVM is quite different from real abundance map, which means the unmixing results of URVM are poor. The abundance map of ULLSSVM has certain similarity with real abundance map, but the green region (the abundance of three classes are equal, 0.333) is blurred. Compared with the abundance maps of the former two algorithms, the abundance maps of ULLSTWSVM are the closest to the real abundance maps.

The quantitative results of three algorithms are tabulated in Table 3. We can see clearly that the RMSEs of ULLSTWSVM are smaller than those of

---

**Table 2. Quantitative results of unmixing synthetic data generated by Indian pines through MESMA and ULLSTWSVM.**

| number of reference spectra | MESMA | ULLSTWSVM | MESMA | ULLSTWSVM | MESMA | ULLSTWSVM | MESMA | ULLSTWSVM | MESMA | ULLSTWSVM | MESMA | ULLSTWSVM | MESMA | ULLSTWSVM | MESMA | ULLSTWSVM | MESMA | ULLSTWSVM | MESMA | ULLSTWSVM | MESMA | ULLSTWSVM |
|-----------------------------|-------|-----------|-------|-----------|-------|-----------|-------|-----------|-------|-----------|-------|-----------|-------|-----------|-------|-----------|-------|-----------|-------|-----------|-------|-----------|-------|-----------|
| 15                          | 0.256 | 0.008     | 0.145 | 0.004     | 0.256 | 0.008     | 0.145 | 0.004     | 0.256 | 0.008     | 0.145 | 0.004     | 0.256 | 0.008     | 0.145 | 0.004     | 0.256 | 0.008     | 0.145 | 0.004     | 0.256 | 0.008     |
| 30                          | 0.091 | 0.005     | 0.102 | 0.002     | 0.091 | 0.005     | 0.102 | 0.002     | 0.091 | 0.005     | 0.102 | 0.002     | 0.091 | 0.005     | 0.102 | 0.002     | 0.091 | 0.005     | 0.102 | 0.002     | 0.091 | 0.005     |
| 45                          | 0.111 | 0.003     | 0.123 | 0.001     | 0.111 | 0.003     | 0.123 | 0.001     | 0.111 | 0.003     | 0.123 | 0.001     | 0.111 | 0.003     | 0.123 | 0.001     | 0.111 | 0.003     | 0.123 | 0.001     | 0.111 | 0.003     |

**Table 3. Quantitative results of unmixing synthetic data generated by USGS through URVM, ULLSSVM and ULLSTWSVM.**

| algorithms | Olivine | Hematite | Muscovite | Mean | RMSE | RE | Time (s) |
|------------|---------|----------|-----------|------|------|----|---------|
| ULLVM      | 0.1870  | 0.1111   | 0.1789    | 0.1626 | 0.0102 | 1.8911 |
| ULLSSVM    | 0.0597  | 0.0510   | 0.0386    | 0.0505 | 0.0712 | 1.3862 |
| ULLSTWSVM  | 0.0529  | 0.0457   | 0.0327    | 0.0449 | 0.0684 | 1.3582 |
URVM and ULLSSVM for each class. Moreover, the Mean RMSE of ULLSTWSVM is 0.1177 and 0.0056 less than that of URVM and ULLSSVM, respectively. The RE of ULLSTWSVM is 0.0337 and 0.0028 less than that of URVM and ULLSSVM, respectively. Finally, the computational time of three algorithms is discussed. ULLSTWSVM took 1.3582s, a time shorter than 1.8911s of URVM and 1.3862s of ULLSSVM. Among the three algorithms, URVM took the longest time. This is because URVM is more time consuming to estimate the prior parameters by maximizing the marginal likelihood. ULLSTWSVM can save 0.028s in comparison with ULLSSVM. This is because ULLSTWSVM solves two small size QPPs instead of a large one in ULLSSVM.

**Experiments on remote sensing data**

**Experiment 4**

In this experiment, we evaluated the unmixing performance of proposed algorithm using real hyperspectral data Salinas-A. Comparison algorithms include URVM, ULLSSVM, and ULLSTWSVM. At the same time, gaussian kernel function is introduced into LSSVM (Least Squares Support Vector Machine) and LSTWSVM, as well as the unmixing algorithms are named UNLLSSVM (unmixing algorithm based on nonlinear LSSVM) and UNLLSTWSVM (unmixing algorithm based on nonlinear LSTWSVM) respectively. Three classes of pixels, Lettuce_romaine_5 wk, Lettuce_romaine_7 wk, Corn_senesced_green_weeds were selected to be unmixed. 68, 76 and 39 pixels (5% of the total number) were evenly selected from each class as training samples. The results are shown in Figure 6 and Table 4.

Examining the maps, Figure 6(c) is found to be more blurred than the other abundance maps, especially for unmixing Lettuce_romaine_5 wk and Corn_senesced_green_weeds. In Figure 6(b), there are some white speckles in black area, some pixels belonging to Corn_senesced_green_weeds are unmixed to Lettuce_romaine_7 wk. Figure 6(e), (f) are similar and close to the true abundance map, their white areas are clearer than those of Figure 6(d).

Table 4 reports the unmixing results of five algorithms. ULLSTWSVM performs better than ULLSSVM, the RMSEs and Mean RMSE of ULLSTWSVM are reduced by 0.095, 0.0538, 0.1068 and 0.0877 than those of ULLSSVM. ULLSTWSVM also outperforms URVM. When unmixing Lettuce_romaine_5 wk and Lettuce_romaine_7 wk, ULLSTWSVM can obtain smaller RMSEs than those of URVM. However, URVM can obtain smallest RMSE when unmixing Corn_senesced_green_weeds. In addition, the Mean RMSE of ULLSTWSVM is smaller than that of URVM. Observing the REs, we
From top to bottom, rows show the abundances of Lettuce_romaine_5 wk, Lettuce_romaine_7 wk, and Corn_senesced_green_weeds, respectively. Column represents the unmixing algorithms: (a) real abundance, (b) URVM, (c) ULLSSVM, (d) ULLSTWSVM, (e) UNLLSSVM, and (f) UNLLSTWSVM.

Table 4. Five algorithms unmixing results on salinas-A data.

| Algorithms | Lettuce_romaine_5 wk | Lettuce_romaine_7 wk | Corn_senesced_green_weeds | Mean RMSE | RE |
|------------|----------------------|----------------------|---------------------------|------------|----|
| URVM       | 0.1336               | 0.1287               | 0.0629                    | 0.1131     | 0.0950 |
| ULLSSVM    | 0.2080               | 0.1438               | 0.2255                    | 0.1956     | 0.1130 |
| ULLSTWSVM  | 0.1130               | 0.0900               | 0.1187                    | 0.1079     | 0.0830 |
| UNLLSSVM   | 0.0552               | 0.0695               | 0.0877                    | 0.0720     | 0.0798 |
| UNLLSTWSVM | 0.0484               | 0.0667               | 0.0796                    | 0.0662     | 0.0567 |

can see that the RE of ULLSTWSVM is smaller than that of ULLSSVM and URVM. The RE of UNLLSTWSVM is smaller than that of UNLLSSVM.

After introducing gaussian kernel function, UNLLSTWSVM and UNLLSSVM greatly improve the unmixing performance. Furthermore, UNLLSTWSVM outperforms the compared methods for the Mean RMSE and RE.

Experiment 5
In the last experiment, we evaluate the unmixing performance of proposed algorithm using real hyperspectral data Pavia University. The comparison algorithms include URVM, ULLSSVM, ULLSTWSVM, UNLLSSVM and UNLLSTWSVM. In the experiment, we unmix four classes of pixels over Pavia University: Meadows, Metal sheets, Soil and Bitumen. 79, 73, 403 and 79 pixels (10% of the total number) from each class are evenly selected as training samples. The results are shown in Figure 7 and Table 5.

Observing Figure 7, we can find that the abundance maps of URVM are quite different from the reference abundance maps. The white areas in the abundance maps of ULLSSVM are clearer than those in the abundance maps of URVM. The abundance maps of ULLSTWSVM are similar to those of UNLLSSVM. Among all the abundance maps, the abundance maps of UNLLSTWSVM are the closest to the reference ones. Quantitative evaluation is shown in Table 5 which lists RMSEs, Mean RMSEs and REs of unmixing algorithms. Obviously, URVM results in a lower performance than the other four algorithms. The performance of ULLSTWSVM and UNLLSSVM is similar, and their Mean RMSEs are obviously less than that of ULLSSVM. For endmembers Meadows, Soil and Bitumen, UNLLSTWSVM obtains the minimum RMSEs. In addition, UNLLSTWSVM produces the minimum Mean RMSE and RE, which are 0.1013 and 0.0689, respectively.

Discussion
In Experiment 1, we evaluated ULLSTWSVM and some representative single endmember unmixing methods (Figure 4 and Table 1). ULLSTWSVM outperforms the other compared methods. This is because the single endmember unmixing methods use a fixed endmember to represent a pure class, the intra-class endmember variability is ignored. ULLSTWSVM uses multiple endmembers (training samples) to model a pure class to take intra-class endmember variability into account, and therefore
the number of class $j$ increases, the within-class endmember variability of class $j$ becomes larger, which in turn leads to an increase in RMSE.

In Experiment 2, we evaluated ULLSTWSVM and MESMA (Table 2). In MESMA, all endmember combinations are attempted, and the optimal endmember combinations are found according to the “minimum reconstruction error criterion”, while abundances are obtained. But “the minimum reconstruction error” is purely in the mathematical sense, it is completely possible that the endmembers and abundances are quite different from the true ones. ULLSTWSVM obtains the abundances by calculating the distances from the mixed pixel to the classification hyperplanes, which reflects the contribution of all endmembers (training samples) to the unmixing. ULLSTWSVM is a more reasonable unmixing model. In addition, it is easy to find that the unmixing effects of MESMA are greatly affected by the number of reference spectra. When the number of reference spectra increases from 15 to 30, the Mean RMSEs of MESMA decreased significantly. However, the
computation time increases from 323.59 seconds to 5011.68 seconds. On the contrary, the Mean RMSEs of ULLSTWSVM doesn’t change significantly. **Figure 8** shows the change of Mean RMSEs of ULLSTWSVM with the number of training samples. When the number of training samples is greater than 45, the Mean RMSE curve is stable. This shows that ULLSTWSVM is a small sample learning method. In the case of training with small number of samples, ULLSTWSVM has strong generalization ability.

The experimental results of Experiment 4 and Experiment 5 reveal that introducing a gaussian kernel function can improve the unmixing performance when unmixing real hyperspectral images (**Figure 6**, **Figure 7**, **Table 4** and **Table 5**). For example, UNLLSSVM outperforms ULLSSVM, and UNLLTSTWSVM outperforms ULLSTWSVM. The introduction of nonlinear kernel function mainly solves two problems: (1) In the original feature space, the endmember classes are linearly inseparable. (2) Nonlinear factors, such as scattering and reflection, affect the results of unmixing.

**Conclusion**

In this paper, to incorporate the endmember variability, a novel unmixing method based on Least Squares Twin Support Vector Machines, namely ULLSTWSVM, is proposed for hyperspectral remote sensing imagery. According to the use of different kernel functions, ULLSTWSVM includes two forms: ULLSTWSVM (the unmixing algorithm based on linear Least Squares Twin Support Vector Machines) and UNLLSTWSVM (the unmixing algorithm based on nonlinear Least Squares Twin Support Vector Machines). Five experiments on four hyperspectral image datasets are carried out to verify the superior performance of ULLSTWSVM. The first three experiments on synthetic data evaluated the performance of ULLSTWSVM and the last two experiments on real remote sensing data evaluated the performance of UNLLSTWSVM as well as ULLSTWSVM. The experimental results show that ULLSTWSVM delivers better performance compared with the other unmixing algorithms tested.

In the future work, unmixing uncertainty which exists in all kinds of unmixing models will be investigated. ULLSTWSVM can be a good method to use for unmixing uncertainty analysis.

**Disclosure statement**

No potential conflict of interest was reported by the authors.

**Funding**

This work was supported by National Natural Science Foundation of China (61675051).

**References**

Bayer, A. D., Bachmann, M., Rogge, D., Muller, A., & Kaufmann, H. (2016). Combining field and imaging spectroscopy to map soil organic carbon in a semiarid environment. *IEEE Journal of Selected Topics in Applied Earth Observations & Remote Sensing*, 9(9), 3997–4010. https://doi.org/10.1109/JSTARS.2016.2585674.

Bhatt, J. S., Joshi, M. V., & Raval, M. S. (2014). A data-driven stochastic approach for unmixing hyperspectral imagery. *IEEE Journal of Selected Topics in Applied Earth Observations & Remote Sensing*, 7(6), 1936–1946. https://doi.org/10.1109/jstars.2014.2328597.

---

**Figure 8**. Effect of number of training samples on ULLSTWSVM.
Bioucas-Dias, J. M. (2009, August 26–28). A variable splitting augmented lagrangian approach to linear spectral unmixing [paper presentation]. 2009 first workshop on hyperspectral image and signal processing: evolution in remote sensing data conference, Grenoble, France.

Bioucas-Dias, J. M., Plaza, A., Dobigeon, N., Parente, M. D., Du, Q., Gader, P., & Chanussot, J. et al. (2012). Hyperspectral unmixing overview: geometrical, statistical, and sparse regression-based approaches. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 5(2), 354–379. https://doi.org/10.1109/JSTARS.2012.2194696.

Boardman, J. W. (2013, October). Automating spectral unmixing of aviris data using convex geometry concepts [Paper presentation]. The Fourth Annual Airborne Geoscience Workshop.

Chen, Y. J., Ge, W. D., & Sun, L. (2018). A novel linear hyperspectral unmixing method based on collaborative sparsity and total variation. Acta Automatica Sinica, 44(1), 116–128. https://doi.org/10.16383/j.aas.2018.c160414.

Fang, H., Li, A. H., Wang, T., & Xu, H. X. (2019). Hyperspectral unmixing using double-constrained multilayer NMF. Remote Sensing Letters, 10(3), 224–233. https://doi.org/10.1080/2150704X.2018.1541107.

Halimi, A., Dobigeon, N., & Tourneret, J. (2015). Unsupervised unmixing of hyperspectral images accounting for endmember variability. IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society, 24(12), 4904–4917. https://doi.org/10.1109/TIP.2015.2471182.

Iordache, M. D., Bioucas-Dias, J. M., & Plaza, A. (2012, July). Collaborative sparse unmixing of hyperspectral data [paper presentation]. 2012 IEEE Geoscience and remote sensing symposium, Munich, Germany.

Jin, Q. W., Ma, Y., Pan, E. T., Fan, F., Huang, J., Li, H., Jin, S. H., & Mei, X. G. (2019). Hyperspectral unmixing with gaussian mixture model and spatial group sparsity. Remote Sensing, 11(20), 2434. https://doi.org/10.3390/rs11202434.

Kumar, M. A., & Gopal, M. (2009). Least squares twin support vector machines for pattern classification. Expert Systems with Applications, 36(4), 7535–7543. https://doi.org/10.1016/j.eswa.2008.09.066.

Lee, D. D., & Seung, H. S. (2000, January). Algorithms for non-negative matrix factorization. 13th international conference on neural information processing systems [paper presentation]. Breckenridge, America.

Li, C., & Yin, J. (2013). A multispectral remote sensing data spectral unmixing algorithm based on variational bayesian ICA. Journal of the Indian Society of Remote Sensing, 41(2), 259–268. https://doi.org/10.1007/s12524-012-0245-0.

Li, J., & Bioucas-Dias, J. M. (2008, July 8–11). Minimum volume simplex analysis: a fast algorithm to unmix hyperspectral data [paper presentation]. Geoscience and remote sensing symposium, Boston, America.

Ma, Y., Jin, Q., Mei, X., Dai, X., Fan, F., Li, H., & Huang, J. (2019). Hyperspectral unmixing with gaussian mixture model and low-rank representation. Remote Sensing, 11(8), 911. https://doi.org/10.3390/rs11080911.

Ma, Y., Li, C., Mei, X., & Liu, C. (2017). Robust sparse hyperspectral unmixing with norm. IEEE Transactions on Geoscience and Remote Sensing, 55(3), 1227–1239. https://doi.org/10.1109/TGRS.2016.2616161.

Miao, L., & Qi, H. (2007). Endmember extraction from highly mixed data using minimum volume constrained nonnegative matrix factorization. IEEE Transactions on Geoscience and Remote Sensing, 45(3), 765–777. https://doi.org/10.1109/TGRS.2006.888466.

Nascimento, J. M. P., & Bioucas-Dias, J. M. (2005). Vertex component analysis: A fast algorithm to unmix hyperspectral data. IEEE Transactions on Geoscience and Remote Sensing, 43(4), 898–910. https://doi.org/10.1109/TGRS.2005.844293.

Oskouei, M. M., & Babakan, S. (2016). Detection of alteration minerals using hyperion data analysis in lahroud. Journal of the Indian Society of Remote Sensing, 44(5), 1–9. https://doi.org/10.1007/s12524-016-0549-6.

Peng, T. L., & Tan, W. Q. (2020). A manifold hessian-regularized NMF for hyperspectral data unmixing. Remote Sensing Letters, 11(1), 86–95. https://doi.org/10.1080/2150704X.2019.1678792.

Roberts, D. A., Gardner, M., Church, R., Ustin, S., Scheer, G., & Green, R. O. (1998). Mapping chaparral in the santa monica mountains using multiple endmember spectral mixture models. Remote Sensing of Environment, 65(3), 267–279. https://doi.org/10.1016/S0034-4257(98)00037-6.

Stagakis, S., Vanikiotis, T., & Sykioti, O. (2016). Estimating forest species abundance through linear unmixing of CHRIS/PROBA imagery. ISPRS Journal of Photogrammetry & Remote Sensing, 119, 79–89. https://doi.org/10.1016/j.isprsjprs.2016.05.013.

Tong, L., Zhou, J., Li, X., & Qian, Y. (2017). Region-based structure preserving non-negative matrix factorization for hyperspectral unmixing. IEEE Journal of Selected Topics in Applied Earth Observations & Remote Sensing, 10(4), 1575–1588. https://doi.org/10.1109/jstares.2016.2621003.

Ueato, T., Fauvel, M., & Dobigeon, N. (2018). Hyperspectral image unmixing with LiDAR data-aided spatial regularization. IEEE Transactions on Geoscience and Remote Sensing, 56(7), 4098–4108. https://doi.org/10.1109/ieeeexplore.org/document/8347066.

Wang, L., & Jia, X. (2009). Integration of soft and hard classifications using extended support vector machines. IEEE Geoscience & Remote Sensing Letters, 6(3), 543–547. https://doi.org/10.1109/LGRS.2009.2020924.

Wang, L., Liu, D., & Wang, Q. (2013). Spectral unmixing model based on least squares support vector machine with unmixing residue constraints. Ieee Geoscience and Remote Sensing Letters, 10(6), 1592–1596. https://doi.org/10.1109/LGRS.2013.2262371.

Yang, J., Wang, L., & Qian, J. (2015). An unmixing algorithm based on the relevance vector machine for hyperspectral imagery. Journal of Harbin Engineering University, 36(2), 267–270+286. https://doi.org/10.1063/1.10067804.

Zhang, S. Q., Li, J., Li, H. C., Deng, C. Z., & Plaza, A. (2018). Spectral-spatial weighted sparse regression for hyperspectral image unmixing. IEEE Transactions on Geoscience and Remote Sensing, 56(6), 3265–3276. https://doi.org/10.1109/TGRS.2017.2797700.

Zheng, Y. H., Wu, F. Y., Shim, H. J., & Sun, L. (2019). Sparse unmixing for hyperspectral image with nonlocal low-rank prior. Remote Sensing, 11(24), 24. https://doi.org/10.1007/978-3-030-36189-1.

Zhou, Y., Rangarajan, A., & Gader, P. D. (2018). A gaussian mixture model representation of endmember variability in hyperspectral unmixing. IEEE Transactions on Image Processing, 27(5), 2242–2256. https://doi.org/10.1109/TIP.2018.2795744.