Hyperspectral Image Unmixing via Alternating Projected Subgradients

A. Zymnis, S.-J. Kim, J. Skaf, M. Parente, and S. Boyd

Department of Electrical Engineering, Stanford University, Stanford, CA 94305
Email: {azymnis,sjkim,jskaf,cyberey,boyd}@stanford.edu

***Abstract*—We formulate the problem of hyperspectral image unmixing as a nonconvex optimization problem, similar to nonnegative matrix factorization. We present a heuristic for approximately solving this problem using an alternating projected subgradient approach. Finally, we present the results of applying this method on the 1990 AVIRIS image of Cuprite, Nevada and show that our results are in agreement with similar studies on the same data.**

I. Introduction

We are given $Y \in \mathbf{R}_+^{m \times n}$, a hyperspectral image with n pixels and m frequency bands. This means that Y_{ij} is the reflectance of the jth image pixel to the ith frequency band.

The spectral unmixing problem consists of finding $W \in \mathbf{R}_+^{m \times k}$ (called the *endmember matrix*) and $H \in \mathbf{R}_+^{k \times n}$, where $k < \min(m, n)$ (called the *abundance matrix*), that explain the data well, *i.e.*, $Y \approx WH$. In other words, we want the columns of Y to be approximately linear combinations of the columns of H and the columns H to be similar for neighboring pixels. Furthermore the elements of W and H must be nonnegative, since the elements of the former represent reflectance, which are inherently nonnegative quantities, while the elements of the latter represent mixing coefficients. Finally, we want the columns of H to sum to 1. The matrices W and H have a very natural interpretation. The columns of W can be thought of as a nonnegative basis for the image spectra. The columns of H can be thought of as the mixing coefficients of each individual pixel. Therefore we can view spectral unmixing as a special type of blind source separation.

We formulate the spectral unmixing problem (SUP) as

$$\begin{array}{ll} \text{minimize} & \|Y - WH\|_F^2 + \lambda \sum_{i=1}^{n} \sum_{j \in \mathcal{N}(i)} ||h_i - h_j||_1 \\ \text{subject to} & W, H \geq 0, \quad \mathbf{1}^T H = \mathbf{1}^T, \end{array} \tag{1}$$

where the optimization variables are W and H, and λ is the regularization parameter. Here $\mathbf{1}$ is the vector of all ones whose dimension is clear from the context and $\|A\|_F = \sqrt{\mathbf{Tr}(A^T A)}$ denotes the matrix Frobenius norm, h_i denotes the ith column of H, $\|x\|_1 = \sum_i |x_i|$ denotes the vector ℓ_1-norm and $\mathcal{N}(i)$ denotes the set of pixels which are neighboring pixel i. The parameter $\lambda \geq 0$ controls the trade-off between data fidelity and pixel dissimilarity. The SUP (1) is not convex; it is however biconvex, meaning that for fixed W it is convex in H, while for fixed H it is convex in W.

The SUP (1) is a variation of the so-called nonnegative matrix factorization (NMF) problem which includes a penalty term for pixel dissimilarities in the objective. NMF has been the focus of an extensive body of research in a wide range of machine learning and signal processing applications [1], [2], [3], [4], [5], [6], [7]. There has been a considerable interest in developing efficient heuristics for solving NMF problems. The first method proposed in [8] consists of a set of multiplicative updates that can be shown to converge to a local minimum. A method based on alternating least squares is proposed in [9]. Methods based on alternating convex programming have also been proposed [10], [11]. Finally, some authors have considered methods based on alternating projected gradients [12], [13].

In this paper we propose a heuristic for approximately solving the SUP (1), which is based on the idea of alternating projected subgradient descent. As such our method is closely related to the work presented in [12]. The main difference is that we use a subgradient method for one of the alternations. Furthermore the objective of the problem that we want to solve has strong coupling terms between the columns of H.

The rest of this paper is organized as follows. In §II we present our method. In §III we show the performance of this method on a real hyperspectral dataset taken from Cuprite, Nevada. Finally in §IV we provide our concluding remarks.

II. Method

The method that we propose for solving problem (1) is an alternating projected gradient method. We initialize W and H and then alternate between fixing W and taking a few subgradient steps on H and fixing H and taking a gradient step on W. This method is guaranteed to converge to a local minimum of the SUP (1).

A. Abundance matrix update

We first look at the problem that we have to solve when W is fixed. In this case, for each pixel i we need to solve

$$\begin{array}{ll} \text{minimize} & \|y_i - Wh_i\|_2^2 + \lambda \sum_{j \in \mathcal{N}(i)} ||h_i - h_j||_1 \\ \text{subject to} & h_i \geq 0, \quad \mathbf{1}^T h_i = 1, \end{array} \tag{2}$$

This material is based on work supported by JPL award I291856, NSF award 0529426, DARPA award N66001-06-C-2021, NASA award NNX07AEIIA, and AFOSR award FA9550-06-1-0312.

with variable h_i. Problem (2) is a convex nondifferentiable optimization problem. This problem can be easily transformed into a smooth problem and solved in a variety of ways such as interior point methods [14]. However, we use instead a subgradient method on this problem. Specifically let

$$f_i(h) = \|y_i - Wh\|_2^2 + \lambda \sum_{j \in \mathcal{N}(i)} ||h - h_j||_1.$$

Suppose $s \in \mathbf{R}^k$ has elements $s_l = \mathbf{sgn}(h_l - H_{lj})$, where $\mathbf{sgn}$ is the sign function. Then we have that

$$g = 2W^TWh - 2W^Ty_i + \lambda s \in \partial f,$$

i.e., g is a subgradient of f at h. By this we mean that

$$f(h') \geq f(h) + g^T(h' - h),$$

for all h'.

A projected subgradient method for solving problem (2) proceeds by iteratively taking a step in a negative subgradient direction and then projecting onto the feasible set. That is, if at iteration t the value of h_i is $h_i^{(t)}$, then at $t+1$ h_i is updated according to

$$h_i^{(t+1)} := \left(h_i^{(t)} - \gamma^{(t)} g\right)_{\mathcal{P}}, \tag{3}$$

where $\gamma^{(t)} > 0$ is the step size and $(\cdot)_{\mathcal{P}}$ denotes projection on the probability simplex, *i.e.*,

$$(x)_{\mathcal{P}} = \underset{y \geq 0,\ \mathbf{1}^T y = 1}{\operatorname{argmin}} \|x - y\|_2.$$

The projection on the probability simplex can be carried out very efficiently by a technique similar to waterfilling. Using Lagrange duality arguments, we can show that finding y, the projection of $x \in \mathbf{R}^k$ on the probability simplex, is equivalent to finding $\lambda \in \mathbf{R}_+^k$ and $\mu \in \mathbf{R}$ such that

$$x - y + \mu \mathbf{1} - \lambda = 0.$$

These can be found up to a tolerance $\epsilon > 0$ by the following algorithm:

```
given: $x \in \mathbf{R}^k$, $\epsilon > 0$
initialize: $\mu = 0$, $\lambda = 0$, $y = x$
while: $\mathbf{1}^T y - 1 > \epsilon$ or $y_l < -\epsilon$ for some $1 \leq l \leq k$
    $d\mu := (\mathbf{1}^T y - 1)/k$
    $\mu := \mu + d\mu$
    $y := y - d\mu \mathbf{1}$
    $d\lambda := (y)_-$
    $\lambda := \lambda - d\lambda$
    $y := y - d\lambda$
end
```

Here $(y)_-$ denotes the negative part of y.

The projected subgradient update (3) will converge to the solution of the SUP (2) as long as the step sizes γ_t satisfy the so-called diminishing step size rules. For a convergence proof, see [15, §3.4] or [16, §7.2.2].

We want to point out that the projected subgradient method is not a descent method. Therefore at each iteration, there is no guarantee for a reduction of the objective. However, the objective is reduced eventually since the method converges.

In fact we can update all columns of H simultaneously, despite the coupling between different columns in the objective of problem (1). This corresponds to each pixel i having a local copy of h_j for all neighboring pixels j, which is updated after each subgradient step. This technique is known as primal decomposition in the optimization literature.

Let $S^{(t)} \in \mathbf{R}^{k \times n}$ have elements

$$S_{li}^{(t)} = \sum_{j \in \mathcal{N}(i)} \mathbf{sgn}(H_{li}^{(t)} - H_{lj}^{(t)}).$$

Then the projected subgradient update for H is

$$H^{(t+1)} := \left(H^{(t)} - \gamma^{(t)}(2W^TWH^{(t)} - 2W^TY + \lambda S^{(t)})\right)_{\mathcal{P}}.$$

Here $(H)_{\mathcal{P}}$ means projecting each column of H on the probability simplex. This update rule is guaranteed to converge to the solution of (2); the reader is referred to [15, §4.1] for a convergence proof.

B. Endmember matrix update

For a fixed abundance matrix H, finding the endmember matrix reduces to solving the convex problem

$$\begin{array}{ll} \text{minimize} & \|Y - WH\|_F^2 \\ \text{subject to} & W \geq 0. \end{array} \tag{4}$$

The objective is separable in the rows of W. The gradient update rule for this problem is

$$W^{(t+1)} := \left(W^{(t)} + \gamma(W^{(t)}H - Y)H^T\right)_+,$$

where $\gamma > 0$ is the step size and $(\cdot)_+$ denotes taking the positive part. It can be shown that this update will converge to the solution of problem (4) as long as

$$\gamma < 2/\|HH^T\|,$$

where $\|A\|$ denotes the spectral norm, *i.e.*, the maximum singular value. For a proof of this result see [16, §7.2.1].

C. Overall algorithm

To approximately solve problem (1), we first initialize the columns of W from k randomly chosen columns of Y and initialize $H_{ij} = 1/k$ for all i and j. We then proceed to iteratively take P subgradient updates on H and a single gradient step on W. We alternate the steps for T times. The overall algorithm is described below:

```
given: $T$, $P$, $\lambda$
initialize: $H = H^{(0)}$, $W = W^{(0)}$
for: $t = 1$ to $T$
    for: $p = 1$ to $P$
        $H := \left(H - (1/\sqrt{p})(2W^TWH - 2W^TY + \lambda S\right)_{\mathcal{P}}$
    end
    $W := \left(W + (1/\|HH^T\|)(WH - Y)H^T\right)_+$
end
```

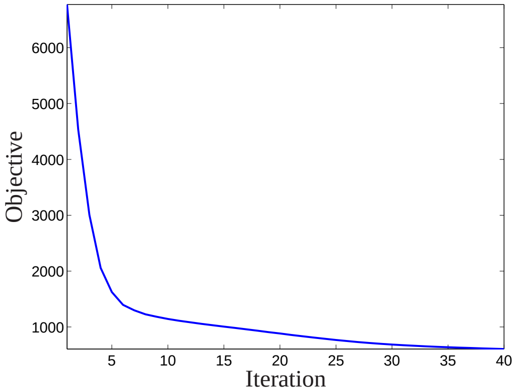

Fig. 1. Objective versus iteration for a single run of our method.

III. Numerical results

In this section we give the result of applying our method to the data described in [17]. This dataset is a hyperspectral image of a mining site at Cuprite, Nevada, which was taken using the airborne AVIRIS imaging system. It is a $350 \times 400 \times 50$ sized image, with each pixel corresponding to a 20m by 20m area. Each pixel spectrum consists of 50 wavelength bands, linearly spaced in the 2.0μm to 2.5μm range. This dataset has been used in a number of studies related to spectral unmixing [18], [19], [20], which allows us to compare our method to other methods.

We applied our method to this dataset for a number of different values of k and λ. We found that setting $\lambda = 0.01$ gives a good tradeoff between data fidelity and abundance correlation. We set the number of endmembers to $k = 10$ Finally, we found that for this value of λ, 10 subgradient steps are enough to ensure that our method converges and that the algorithm converges in about 50 iterations (*i.e.*, we set $P = 10$ and $T = 50$). Each iteration takes about 10s so, in total, the algorithm requires about 10 minutes to reach a local minimum of this problem. Figure 1 shows the objective value versus iteration for a single run of our method.

Our method correctly identified a number of endmembers that are actually present in the dataset. Specifically our method identified two types of alunite (figures 2 and 3), a type of kaolinite (figure 4), a type of halloysite (figure 5), as well a type of jarosite (figure 6). These figures show the corresponding mineral spectra (*i.e.*, columns of W) on the left and the pixel abundances (*i.e.*, rows of H) for each identified mineral. We also plot the spectra taken from [21] which are closest to the estimated endmembers. Our results are in good agreement with similar studies performed on this data [18], [19], [20], especially for the estimated abundances of alunite and kaolinite.

IV. Conclusions

In this paper we have formulated the hyperspectral image unmixing problem as a nonconvex optimization problem. We have described a heuristic for obtaining a good approximate solution to the problem using an alternating projected gradient approach. We demonstrated the heuristic on a real hyperspectral image. The techniques presented in this paper should be well suited for other nonnegative matrix factorization problems with different penalty functions added to the original objective.

References

[1] D. Lee and S. Seung, “Learning the parts of objects by non-negative matrix factorization,” *Nature*, vol. 401, pp. 788–791, 1999.

[2] F. Sha, L. Saul, and D. Lee, “Multiplicative updates for nonnegative quadratic programming in support vector machines,” in *Advances in Neural Information Processing Systems 15*, S. Becker, S. Thrun, and K. Obermayer, Eds. Cambridge, MA: MIT Press, 2003, pp. 1065–1073.

[3] D. Guillamet, J. Vitrià, and B. Schiele, “Introducing a weighted non-negative matrix factorization for image classification,” *Pattern Recognition Letters*, vol. 24, no. 14, pp. 2447–2454, 2003.

[4] Y. Gao and G. Church, “Improving molecular cancer class discovery through sparse non-negative matrix factorization,” *Bioinformatics*, vol. 21, pp. 3970–3975, 2005.

[5] Y. Lin, D. Lee, and L. Saul, “Nonnegative deconvolution for time of arrival estimation,” in *Proceedings of the 2004 International Conference of Speech, Acoustics, and Signal Processing (ICASSP)*, 2004, pp. 377–380.

[6] F. Sha and L. Saul, “Real-time pitch determination of one or more voices by nonnegative matrix factorization,” in *Advances in Neural Information Processing Systems 17*, L. Saul, Y. Weiss, and L. Bottou, Eds. Cambridge, MA: MIT Press, 2005, pp. 1233–1240.

[7] M. Berry, M. Browne, A. Langville, V. Pauca, and R. Plemmons, “Algorithms and applications for approximate nonnegative matrix factorization,” 2006, elsevier Preprint.

[8] D. Lee and S. Seung, “Algorithms for non-negative matrix factorization,” in *Advances in Neural Information Processing Systems 13*, T. Leen, T. Dietterich, and V. Tresp, Eds. Cambridge, MA: MIT Press, 2001, pp. 556–562.

[9] P. Paatero and U. Tapper, “Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values,” *Environmetrics*, vol. 5, pp. 111–126, 1994.

[10] M. Heiler and C. Schnorr, “Learning non-negative sparse image codes by convex programming,” in *Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV)*, 2005, pp. 1667–1674.

[11] R. Zdunek and A. Cichocki, “Nonnegative matrix factorization with constrained second-order optimization,” *Signal Processing*, vol. 87, pp. 1904–1916, 2007.

[12] C.-J. Lin, “Projected gradient methods for nonnegative matrix factorization,” *Neural Computation*, vol. 19, pp. 2756–2779, 2007.

[13] P. Hoyer, “Non-negative matrix factorization with sparseness constraints,” *Journal of Machine Learning Research*, vol. 5, pp. 1457–1469, 2004.

[14] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004, available from http://www.stanford.edu/~boyd/cvxbook.

[15] N. Z. Shor, *Minimization Methods for Non-Differentiable Functions.* Springer-Verlag, 1985.

[16] B. Polyak, *Introduction to Optimization*. Optimization Software, Inc., 1987.

[17] G. A. Swayze, R. N. Clarke, S. Sutley, and A. Gallagher, “Ground-truthing AVIRIS mineral mapping at Cuprite, Nevada,” in *Proceedings of the 3rd JPL Airborne Geoscience Workshop*, 1992.

[18] F. A. Kruse, “Comparison of AVIRIS and Hyperion for hyperspectral mineral mapping,” in *Proceedings of the 11th JPL Airborne Geoscience Workshop*, Mar. 2002.

[19] D. Stein, “Application of the normal compositional model to the analysis of hyperspectral imagery,” in *Proceedings of the IEEE Workshop on Advances in Techniques for Analysis of Remotely Sensed Data*, 2003.

[20] R. N. Clarke, G. A. Swayze, K. E. Livo, R. F. Kokaly, S. Sutley, J. B. Dalton, R. R. McDougal, and C. A. Gent, “Imaging spectroscopy: Earth and planetary remote sensing with the USGS Tetracorder and expert systems,” *Journal of Geophysical Research*, vol. 108, no. 12, pp. 5–44, 2003.

[21] R. N. Clark, , G. A. Swayze, R. Wise, K. Livo, T. M. Hoefen, R. F. Kokaly, and S. J. Sutley, “USGS digital spectral library,” 2006, USGS Open File Report 03-395.

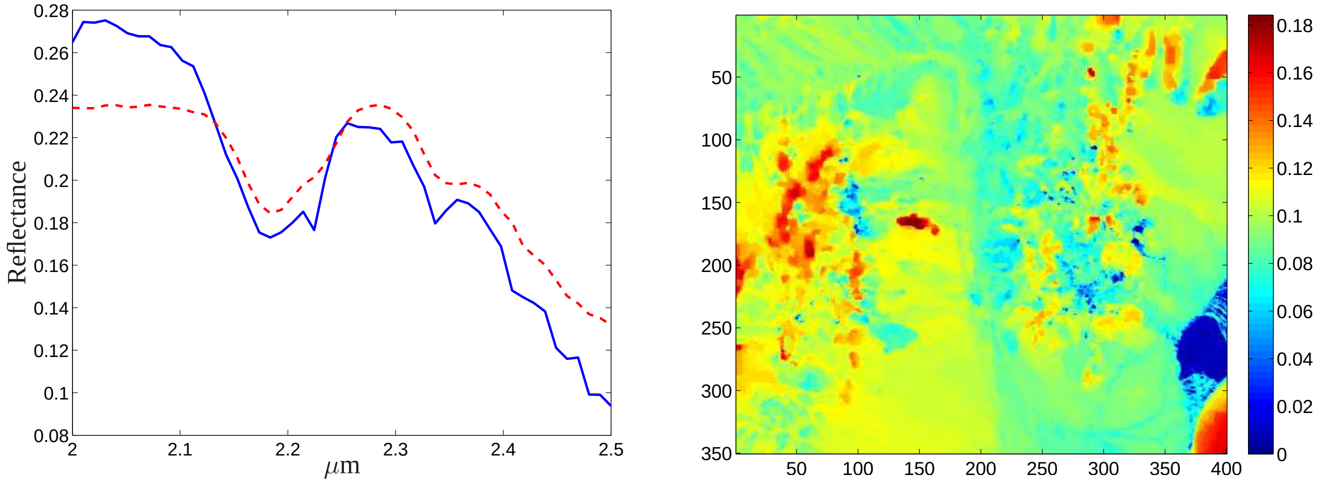

Fig. 2. *Left.* Endmember identified as Alunite type 1 (solid) and the mathced mineral from the USGS library (dashed). *Right.* Estimated mineral abundance.

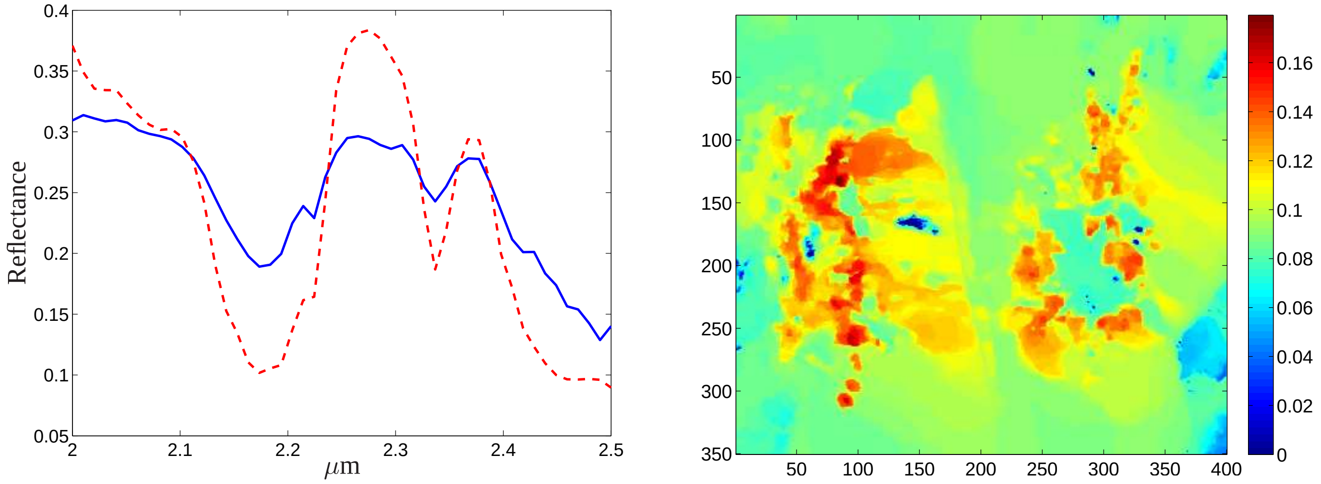

Fig. 3. *Left.* Endmember identified as Alunite type 2 (solid) and the matched mineral from the USGS library (dashed). *Right.* Estimated mineral abundance.

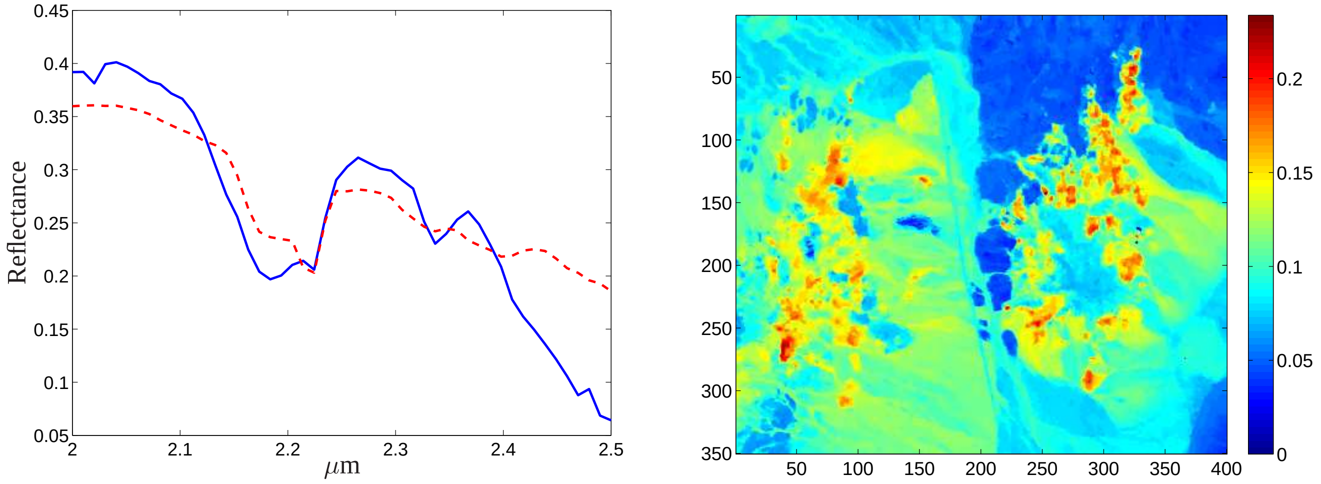

Fig. 4. *Left.* Endmember identified as Kaolinite (solid) and the matched mineral from the USGS library (dashed). *Right.* Estimated mineral abundance.

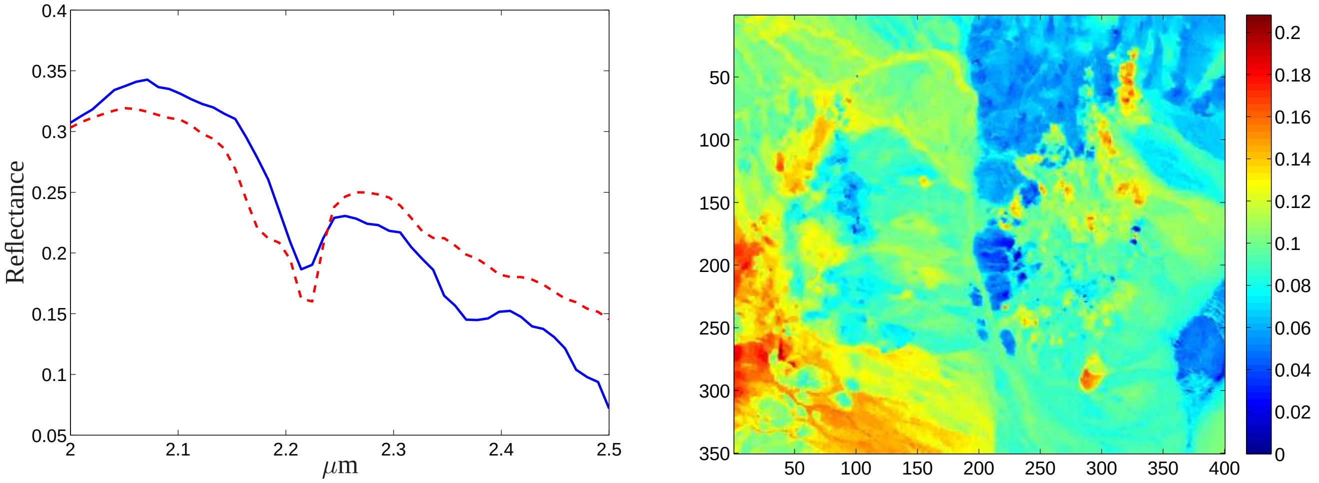

Fig. 5. *Left.* Endmember identified as Halloysite (solid) and the matched mineral from the USGS library (dashed). *Right.* Estimated mineral abundance.

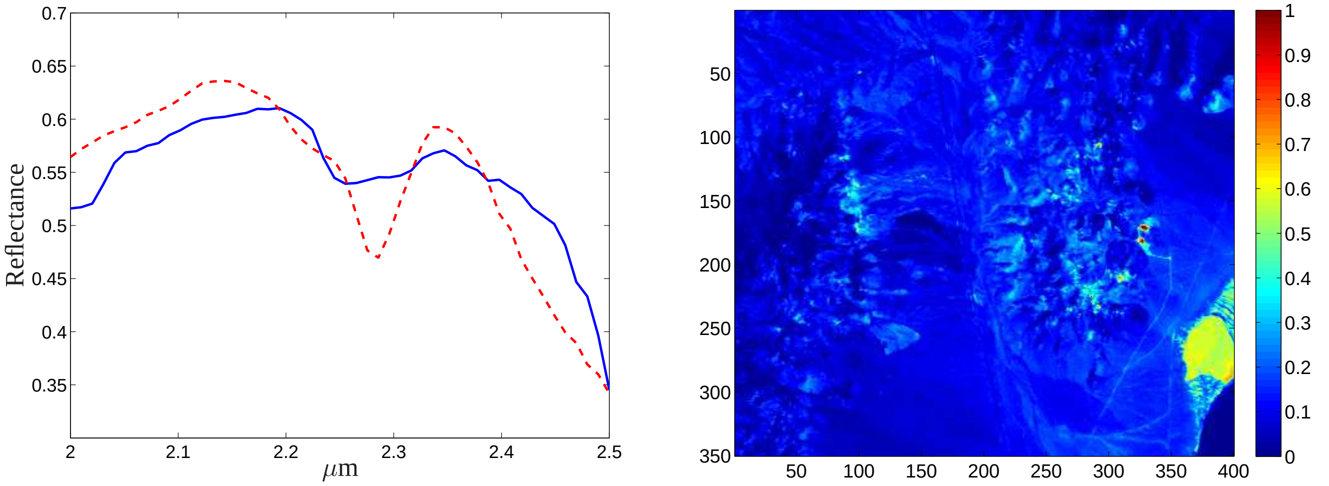

Fig. 6. *Left.* Endmember identified as Jarosite (solid) and the matched mineral from the USGS library (dashed). *Right.* Estimated mineral abundance.