VOLUME REGULARIZED NON-NEGATIVE MATRIX FACTORIZATIONS

Andersen M.S. Ang, Nicolas Gillis

University of Mons
Department of Mathematics and Operational Research
Rue de Houdain 9, 7000 Mons, Belgium

ABSTRACT

This work considers two volume regularized non-negative matrix factorization (NMF) problems that decompose a nonnegative matrix $\mathbf{X}$ into the product of two nonnegative matrices $\mathbf{W}$ and $\mathbf{H}$ with a regularization on the volume of the convex hull spanned by the columns of $\mathbf{W}$. This regularizer takes two forms: the determinant (det) and logarithm of the determinant (logdet) of the Gramian of $\mathbf{W}$. In this paper, we explore the structure of these problems and present several algorithms, including a new algorithm based on an eigenvalue upper bound of the logdet function. Experimental results on synthetic data show that (i) the new algorithm is competitive with the standard Taylor bound, and (ii) the logdet regularizer works better than the det regularizer. We also illustrate the applicability of the new algorithm on the San Diego airport hyperspectral image.

Index Terms— Non-negative matrix factorization, volume regularizer, determinant, log-determinant, coordinate descent

1. INTRODUCTION

Given a non-nonnegative matrix $\mathbf{X} \in \mathbf{R}_+^{m \times n}$ and a positive integer $r \ll \min\{m,n\}$, non-negative matrix factorization (NMF) aims to approximate $\mathbf{X}$ as the product of two nonnegative matrices $\mathbf{W} \in \mathbf{R}_+^{m \times r}$ and $\mathbf{H} \in \mathbf{R}_+^{n \times r}$ so that $\mathbf{X} \approx \mathbf{W}\mathbf{H}^\top$. In this paper, we focus on volume regularized NMF (VR-NMF) and would like to solve the following regularized optimization problem:

$$\min_{\substack{\mathbf{W}\geq\mathbf{0},\\ \mathbf{H}\geq\mathbf{0},\\ \mathbf{H}\mathbf{1}=\mathbf{1}}} F(\mathbf{W},\mathbf{H}) = f(\mathbf{W},\mathbf{H}) + \lambda g(\mathbf{W}). \tag{1}$$

where $f(\mathbf{W},\mathbf{H}) = \frac{1}{2}\|\mathbf{X} - \mathbf{W}\mathbf{H}^\top\|_F^2$ is the data fitting term, $\lambda \geq 0$ is the regularization parameter and g is the volume regularizer. The constraints $\mathbf{W} \geq \mathbf{0}$ and $\mathbf{H} \geq \mathbf{0}$ mean that $\mathbf{W}$ and $\mathbf{H}$ are component-wise nonnegative, where $\mathbf{0}$ is the matrix of zeros. The constraint $\mathbf{H}\mathbf{1}_r = \mathbf{1}_r$, where $\mathbf{1}_r$ is vector of 1's with length r, means $\mathbf{H}$ is a row-stochastic matrix, that is, the entries in each row of $\mathbf{H}$ sum to one. This implies that the columns of $\mathbf{X}$ are encapsulated inside the convex hull spanned by the columns of $\mathbf{W}$. Let $\mathbf{w}_j$ be the j^{th} column of $\mathbf{W}$ and let $\mathcal{C}(\mathbf{W})$ be the convex hull spanned by the set $\{\mathbf{w}_j\}_{j=1}^{j=r}$. Figure 1 illustrates the geometry of the VR-NMF problem: the m-dimensional data points (columns in $\mathbf{X}$) are located inside $\mathcal{C}(\mathbf{W})$ whose vertices are the $\mathbf{w}_j$'s. In this work we focus on two measures for the volume of $\mathcal{C}(\mathbf{W})$

$$\text{det}: \quad g_1(\mathbf{W}) = \frac{1}{2}\det\left(\mathbf{W}^\top\mathbf{W}\right), \tag{2}$$

$$\text{logdet}: \quad g_2(\mathbf{W}) = \frac{1}{2}\log\det\left(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r\right). \tag{3}$$

VR-NMF is asymmetrical with respect to (w.r.t.) $\mathbf{W}$ and $\mathbf{H}$ as the constraints on $\mathbf{W}$ and $\mathbf{H}$ are different and there is no regularization on $\mathbf{H}$. VR-NMF has several applications in particular to unmix hyperspectral images where the columns of $\mathbf{X}$ are the spectral signatures of the pixels present in the image, the columns of $\mathbf{W}$ are the spectral signature of the endmembers and the rows of $\mathbf{H}$ are the abundances of the endmembers in each pixel; see, e.g., [1] and the references therein.

Contributions and organization. The contributions of this work are : (1) develop a vector-wise update algorithm for the VR-NMF with logdet regularization using a simple upper bound on the logdet function, (2) analyze and compare det and logdet regularizers. The paper is organized as follows: §2 gives the algorithmic framework we consider in this paper. §3 and 4 discusses the two regularizers g_1 and g_2 and the corresponding algorithms. §5 contains the numerical experiments and §6 concludes and provides directions of future research.

2. BLOCK COORDINATE DESCENT FOR VR-NMF

The optimization problem (1) has two block of variables, $\mathbf{W}$ and $\mathbf{H}$, and this work adopts the block coordinate descent (BCD) framework with these two blocks of variables; see Algorithm 1. To update $\mathbf{H}$, which is a convex optimization problem, we use the fast gradient method (FGM) described in [2]. The update of $\mathbf{W}$ is more difficult as this subproblem is nonconvex. This work focuses on the update of $\mathbf{W}$, for which we

Thanks to the ERC starting grant no 679515.

will use cyclic BCD with projected gradient update.

Algorithm 1 Algorithm framework for VR-NMF

```
Input: $\mathbf{X} \in \mathbf{R}_+^{m \times n}$, $r \geq 3$ and $\lambda > 0$
Output: $\mathbf{W} \in \mathbf{R}_+^{m \times r}$ and $\mathbf{H} \in \mathbf{R}_+^{r \times n}$
    Initialisation : $\mathbf{W} \in \mathbf{R}_+^{m \times r}$ , $\mathbf{H} \in \mathbf{R}_+^{r \times n}$
for $k = 1, 2, ...$ do
    Update $\mathbf{W}$
    $\mathbf{H} \leftarrow$FGM$(\mathbf{W}, \mathbf{H}, \mathbf{X})$
end for
```

3. VR-NMF WITH DETERMINANT REGULARIZER

Let us first consider $g_1(\mathbf{W}) = \frac{1}{2}\det\left(\mathbf{W}^\top\mathbf{W}\right)$. Focusing on the ith column of $\mathbf{W}$, $\det\left(\mathbf{W}^\top\mathbf{W}\right)$ can be written as a quadratic function of $\mathbf{w}_i$ [3]. In fact, letting $\mathbf{W}_i \in \mathbf{R}_+^{m \times (r-1)}$ be $\mathbf{W}$ with the column $\mathbf{w}_i$ removed, we have

$$\det\left(\mathbf{W}^\top\mathbf{W}\right) = \eta_i \mathbf{w}_i^\top \mathbf{B}_i \mathbf{w}_i = \eta_i \|\mathbf{w}_i\|_{\mathbf{B}_i}^2, \tag{4}$$

where $\eta_i = \det\left(\mathbf{W}_i^\top\mathbf{W}_i\right)$, $\mathbf{B}_i = \mathbf{I}_m - \mathbf{W}_i(\mathbf{W}_i^\top\mathbf{W}_i)^{-1}\mathbf{W}_i$ is a projection matrix that projects into the orthogonal complement of the column space of $\mathbf{W}_i$. As $\mathbf{B}_i$ and $\mathbf{W}_i^\top\mathbf{W}_i$ are symmetric positive semidefinite, the right hand side of (4) is a weighted norm. This means that the determinant regularizer can be interpreted as a re-weighted ℓ_2 norm regularization. This also shows that the problem w.r.t. each column of $\mathbf{W}$ is convex. The objective function (1) w.r.t. $\mathbf{w}_i$ is

$$F(\mathbf{w}_i) = \frac{1}{2}\mathbf{w}_i^\top \Big(\underbrace{\|\mathbf{h}_i\|_2^2\mathbf{I}_m + \lambda\eta_i\mathbf{B}_i}_{\mathbf{Q}_i}\Big)\mathbf{w}_i - \langle \mathbf{X}_i\mathbf{h}_i, \mathbf{w}_i\rangle + c. \tag{5}$$

with gradient $\nabla F(\mathbf{w}_i) = \mathbf{Q}\mathbf{w}_i - \mathbf{X}_i\mathbf{h}_i$ and Lipschitz constant $L_i = \|\mathbf{Q}\|_F$ so that we can use the update

$$\mathbf{w}_i \leftarrow \left[\Big(\mathbf{I} - \frac{\mathbf{Q}}{\|\mathbf{Q}\|_F}\Big)\mathbf{w}_i + \frac{\mathbf{X}_i\mathbf{h}_i}{\|\mathbf{Q}_i\|_F}\right]_+ . \tag{6}$$

The update (6) can be used to update $\mathbf{W}$ in Algorithm 1 and we will refer to this algorithm as Det; see Algorithm 2.

Algorithm 2 – Det [3]

```
For $i = 1 : r$
  | compute $\mathbf{Q}_i$, update $\mathbf{w}_i$ as (6)
```

4. VR-NMF WITH LOG DETERMINANT REGULARIZER

We now consider $g_2(\mathbf{W}) = \frac{1}{2}\log\det\left(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r\right)$, where $\mathbf{I}_r$ is the identity matrix of order r. The term $\delta\mathbf{I}_r$ (δ is set to be a small positive value, here we set $\delta = 1$ for simplicity) acts as a lower bound to prevent $\log\det\left(\mathbf{W}^\top\mathbf{W}\right)$ to go to $-\infty$ as $\mathbf{W}$ tends to a rank deficient matrix. The non-convex function g_2 has a tight convex upper bound (see §3.3 of [1] and the reference therein), which comes from the first-order Taylor approximation of the logdet function:

$$\log\det(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r) \leq \operatorname{tr}\big(\mathbf{F}\mathbf{W}^\top\mathbf{W}\big) + c, \tag{7}$$

where $\mathbf{F} = (\mathbf{Y}^\top\mathbf{Y} + \delta\mathbf{I}_r)^{-1}$ and $\mathbf{Y} \in \mathbf{R}^{m \times r}$ is a matrix constant, and $c = \log\det(\mathbf{F}) - r$. Equality holds when $\mathbf{Y} = \mathbf{W}$. Hence

$$F_T(\mathbf{W}) = \frac{1}{2}\|\mathbf{X} - \mathbf{W}\mathbf{H}^\top\|_F^2 + \frac{\lambda}{2}\operatorname{tr}\big(\mathbf{F}\mathbf{W}^\top\mathbf{W}\big) + c$$

is an upper bound for the objective function of (1) using g_2. The gradient and its Lipschitz constant are given by

$$\begin{aligned} \nabla_{\mathbf{W}} F_T(\mathbf{W}) &= \mathbf{W}(\mathbf{H}\mathbf{H}^\top + \lambda\mathbf{F}) - \mathbf{X}\mathbf{H}^\top, \text{ and} \\ L &= \|\mathbf{H}\mathbf{H}^\top + \lambda\mathbf{F}\|_F. \end{aligned}$$

Minimizing the upper bound instead of the original objective function using $\mathbf{Y} = \mathbf{W}$, we obtain an inexact BCD method that we refer to as Taylor; see Algorithm 3

Algorithm 3 – Taylor [1]

```
$\mathbf{W} \leftarrow \left[\mathbf{W} - \frac{1}{L}\nabla_{\mathbf{W}} F_T(\mathbf{W})\right]_+$ .
```

As $\mathbf{F}$ is a dense matrix, Taylor updates $\mathbf{W}$ matrix-wise, which is different from algorithm Det that updates $\mathbf{W}$ column-wise. In the following, we obtain a column-wise update based on the logdet regularizer by using a simple upper bound of g_2. Given a matrix $\mathbf{A}$ with rank r, let us denote $\mu_i, i = 1, 2, ..., r$ the eigenvalues of $\mathbf{A}$ and $\mu(\mathbf{A})$ the vector containing these eigenvalues. We will assume that they are arranged in descending order $|\mu_1| \geq |\mu_2| \geq ... \geq |\mu_r|$.

Theorem (logdet-trace inequality). Let $\mathbf{A} \in \mathbf{R}^{r \times r}$ be a positive definite (pd) matrix. Then

$$\log\det(\mathbf{A}) \leq \nu\operatorname{tr}(\mathbf{A}) + c, \tag{8}$$

where $\nu = \big(\mu_r(\mathbf{Y}^\top\mathbf{Y} + \delta\mathbf{I}_r)\big)^{-1}$, $c = \sum\log\mu_i(\mathbf{Y}^\top\mathbf{Y}) - r$.

Proof. Recall the log inequality $\log x \leq x - 1$ for $x \geq 0$. Equality holds when $x = 1$. Generalizing the inequality to an arbitrary point $x_0 > 0$, we get $\log x \leq x_0^{-1}x + \log x_0 - 1$. To prove the logdet-trace inequality (8), let $x = \mu_i(\mathbf{A})$ and use the facts that $\det(\mathbf{A}) = \prod\mu_i$, that $\operatorname{tr}(\mathbf{A}) = \sum\mu_i$ and that μ_r is the smallest eigenvalue of $\mathbf{A}$. □

Letting $\mathbf{A} = \mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r$, Equation (8) becomes

$$\log\det(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r) \leq \nu\operatorname{tr}(\mathbf{W}^\top\mathbf{W}) + c. \tag{9}$$

With (9), function (1) using $g = g_2$ w.r.t. $\mathbf{w}_i$ can be upper bounded using

$$F_E(\mathbf{w}_i) = \frac{1}{2}\mathbf{w}_i^\top \underbrace{(\|\mathbf{h}_i\|_2^2 + \lambda\nu)\mathbf{I}_m}_{\mathbf{Q}_i}\mathbf{w}_i - \langle \mathbf{X}_i\mathbf{h}_i, \mathbf{w}_i\rangle + c. \tag{10}$$

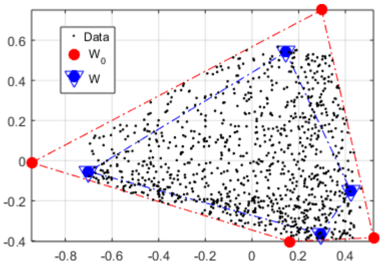

Fig. 1. The geometry of VR-NMF as convex hull fitting. Here $(m, n, r, \theta) = (10, 1500, 4, 0.8)$. For visualization, the data points were projected onto a 2-dimensional space using PCA.

We update $\mathbf{W}$ by taking a gradient step, that is, use (6) with $\mathbf{Q}_i$ defined in (10), and we refer to this algorithm as Eigen; see Algoritm 4.

```
Algorithm 4 – Eigen
1: For $i = 1 : r$
2:   | compute $\mathbf{Q}_i$ as (10), update $\mathbf{w}_i$ as (6)
```

Comparing (9) with (7) reveals that $\nu\text{tr}(\mathbf{W}^\top\mathbf{W})$ is an approximation of $\text{tr}(\mathbf{F}\mathbf{W}\mathbf{W}^T)$ so that (9) is an approximation of (7). The advantage of (9) over (7) is its separable structure, which allows the logdet regularizer to have a column-wise update, similarly as for Det.

5. NUMERICAL EXPERIMENTS

We first conduct experiments on synthetic data to compare the performance of the three algorithms: Det, Taylor and Eigen for comparing $g_1(\mathbf{W})$ and $g_2(\mathbf{W})$. Then, we apply Eigen on the real San Diego airport hyperspectral image.

5.1. Synthetic data sets

Given integers (m, n, r), each entry of the ground truth matrix $\mathbf{W}_0 \in \mathbf{R}_+^{m\times r}$ is generated using the uniform distribution in $[0\ 1]$. The matrix $\mathbf{H}_0 \in \mathbf{R}_+^{n\times r}$ is generated in the form $\mathbf{H} = \mathbf{\Pi}\begin{bmatrix}\mathbf{I}_r \\ \mathbf{H}'_0\end{bmatrix}$ where $\mathbf{H}'_0 \in \mathbf{R}_+^{(n-r)\times r}$ is a row-stochastic matrix that is randomly generated using the Dirichlete distribution (with parameters equal to 1) and the permutation matrix $\mathbf{\Pi}$ shuffles the order of the rows of $\mathbf{H}$. The clean nonnegative matrix $\mathbf{X}_0 = \mathbf{W}_0\mathbf{H}_0$ is then corrupted by additive white Gaussian noise (scaled to fit standard signal to noise ratio) $\mathbf{N} \in \mathbf{R}^{m\times r}$ to form $\mathbf{X}$ as $\mathbf{X} = \mathbf{X}_0 + \mathbf{N}$. Note that in the generation process, the rows of $\mathbf{H}$ with element $\mathbf{H}_{ij}$ larger than a threshold $\theta \geq 0$ are removed and resampled so that all the points in $\mathbf{X}$ are away from the generating vertices $\mathbf{W}_0$; see Figure 1 for an illustration.

We set λ as a constant $\dfrac{5f(\mathbf{W}^{(0)}, \mathbf{H}^{(0)})}{g(\mathbf{W}^{(0)})}$ where $\mathbf{W}^{(0)}$ is the initialization. To initialize the variables, $\mathbf{W}$ is generated using the successive nonnegative projection algorithm (SNPA) from [2], and $\mathbf{H}$ is generated using the FGM from [2].

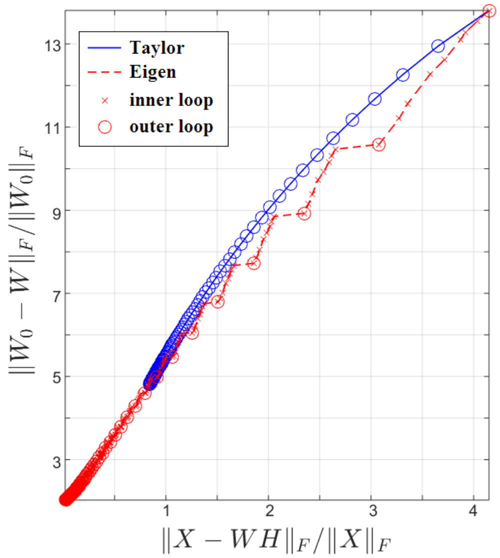

Fig. 2. Error curves of Taylor and Eigen on fitting $\mathbf{X}$ and $\mathbf{W}_0$. The maximum iteration is 200 (recall Eigen has r inner iterations to update each column of $\mathbf{W}$). Both curves start with the same initialization and move towards the bottom left corner. Eigen makes larger progress in every outer iteration than Taylor.The unit of the axes is percentage. The final value of Taylor and Eigen are (0.84, 4.82) and (0.01, 2.03) respectively.

5.2. An illustrative example

We first compare the use of inequalities (9) and (7) in minimizing (1) with g_2. Here $(m, n, r, \theta) = (20, 1000, 8, 0.8)$ and SNR=100dB (noiseless). Figures 2 shows the relative errors of data fitting ($\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F/\|\mathbf{X}\|_F$) and vertex fitting ($\|\mathbf{W}_0 - \hat{\mathbf{W}}\|_F/\|\mathbf{W}_0\|_F$, where $\hat{\mathbf{W}}$ is the matrix $\mathbf{W}$ scaled and matched to $\mathbf{W}_0$ using the Hungarian algorithm). Note that Taylor uses matrix-wise update and Eigen uses column-wise update. Hence, to make a fair comparison, we plot the iteration on the outer loop (every r iteration for Eigen).

Figures 2 and 3 show that, in this example, algorithm Eigen performs better as it achieves lower data fitting error both on $\mathbf{X}$ and $\mathbf{W}$.

5.3. Comparing Det, Taylor and Eigen

Tables 1 gives the statistics of the relative errors in percent over 100 trials in the format of 'average ± standard deviation' for the three algorithms, in all the experiments, we use $(m, n, r) = (20, 1000, 8)$ and a maximum number of iterations of 200. The results show that Eigen performs significantly better than Taylor and Det in the noiseless cases. In noisy settings, Taylor is better than Eigen to identify the

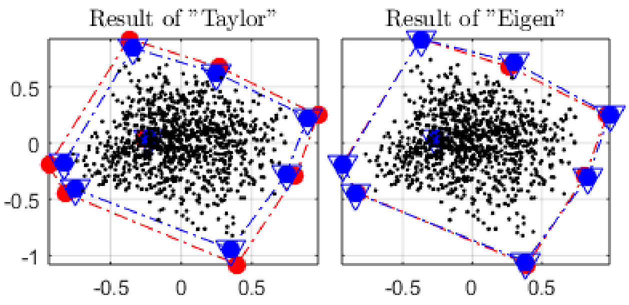

Fig. 3. The geometry of the fittings (PCA projected).

Table 1. Comparison of Det, Taylor and Eigen on synthetic data sets. The table reports the average and standard deviation of the relative errors in percent over 100 trials. The first (resp. second) column is the error on fitting $\mathbf{X}$ (resp. $\mathbf{W}$).

	$\theta = 0.9$, no noise	
Det	2.49 ± 0.51	9.79 ± 1.49
Taylor	0.46 ± 0.12	3.29 ± 0.64
Eigen	$\mathbf{0.01 \pm 0.00}$	$\mathbf{1.19 \pm 0.40}$
	$\theta = 0.9$, 10% noise	
Det	27.18 ± 0.45	36.64 ± 3.45
Taylor	27.76 ± 0.33	$\mathbf{25.43 \pm 2.37}$
Eigen	$\mathbf{23.64 \pm 0.14}$	33.21 ± 5.25
	$\theta = 0.7$, no noise	
Det	3.36 ± 0.62	11.74 ± 2.05
Taylor	1.76 ± 0.34	8.63 ± 1.13
Eigen	$\mathbf{0.02 \pm 0.01}$	$\mathbf{2.80 \pm 1.50}$
	$\theta = 0.7$, 10% noise	
Det	27.17 ± 0.42	39.03 ± 3.51
Taylor	28.00 ± 0.34	$\mathbf{27.97 \pm 2.10}$
Eigen	$\mathbf{23.58 \pm 0.14}$	37.43 ± 4.10

ground truth $\mathbf{W}_0$. In all cases, Det perform poorly.

This experiment shows that the log determinant models produce more accurate solutions. This can be explained as follows: consider the singular value expression of the regularizers, we have $\log\det(\mathbf{W}^\top\mathbf{W}+\delta\mathbf{I}_r) = \sum_i \log(\sigma_i(\mathbf{W})^2+\delta)$ while $\det(\mathbf{W}^\top\mathbf{W}) = \prod_i \sigma_i^2(\mathbf{W})$. Hence the det regularizer is more sensitive to the large singular values. For logdet regularizer, the log operator reduces the effect of large singular values, and thus making a better fit. For example, the singular values of $\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r$ in a trial are : 37.98, 4.26, 3.47, 3.29, 2.60, 2.36, 1.73 and 1.50.

In terms of computational time, the three methods have the same computational complexity, running in $O(mnr)$ operations per iteration. For instance, on average on the synthetic data sets, the matrix-wise method Taylor takes 2.70 seconds, while the column-wise methods take about 2.63 seconds.

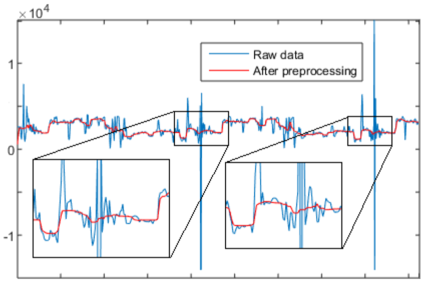

Fig. 4. A portion of data San Diego (band no. 35, data points no. 59000 to 60000) before and after preprocessing.

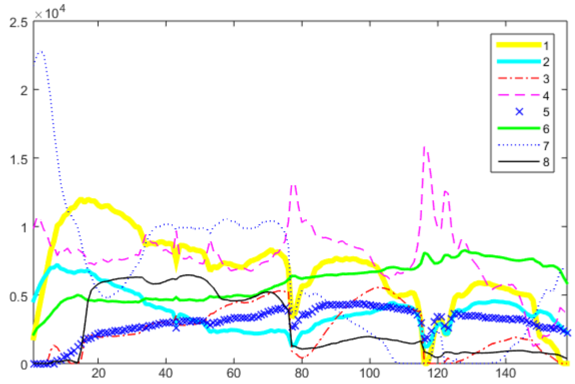

Fig. 5. The spectra of the San Diego airport image obtained by Eigen. The x-axis is the wavelength band number and the y-axis the reflectance. The relative percentage error on fitting the data is 1.76%.

5.4. San Diego airport hyperspectral image

For illustration, we apply the method Eigen on the San Diego airport image (see §3.4.3 of [4] for details) with parameters $r = 8$ and maximum iteration is 100 (the initialization and λ are chosen as for the synthetic data sets). The raw data with sizes $(m, n) = (158, 400^2)$ is preprocessed by replacing all negative values (caused by camera shaking) to zero, spikes are corrected by median filter with window length 20. Figure 4 shows the data before and after preprocessing. Figures 5 and 6 show the spectra and the corresponding abundance map extracted by Eigen, respectively.

Eigen successfully decomposes the data into meaningful components: components 1 and 2 correspond to roof tops, components 3 and 8 corresponds to trees and grass, the remaning ones correspond to different road surfaces [4].

6. CONCLUSION

In this paper, we have studied two VR-NMF problems: one with det and one with logdet regularizer. For the logdet case, we have proposed a new column-wise update of the columns of $\mathbf{W}$ called Eigen, and showed that it has a better numerical performance than the matrix-wise update algorithm from [1] (Taylor) and the vector-wise update with det regularizer from [3] (Det). We have also illustrated the ability of the method Eigen to decompose data into meaningful components on the San Diego airport image.

Future directions include: (1) Compare Det, Taylor and Eigen on real data. (2) Design faster algorithms, e.g., the update of $\mathbf{w}_i$ in Eigen contains many implicit steps and repeated computation, which can be improved [5]. Furthermore, as both Det and Eigen are BCD algorithm with PGD update, it will be interesting to apply randomized acceleration of BCD [6]. (3) Connect to rank minimumization and nuclear norm minimization. The equations (4), (7) and (8) show that there is strong connections between the different regularizations in terms of the singular values of the matrix $\mathbf{W}$. Therefore it will be interesting to study the connection between the volume regularizer and the (convex) nuclear norm regularizer (which is the sum of the singular values of $\mathbf{W}$).

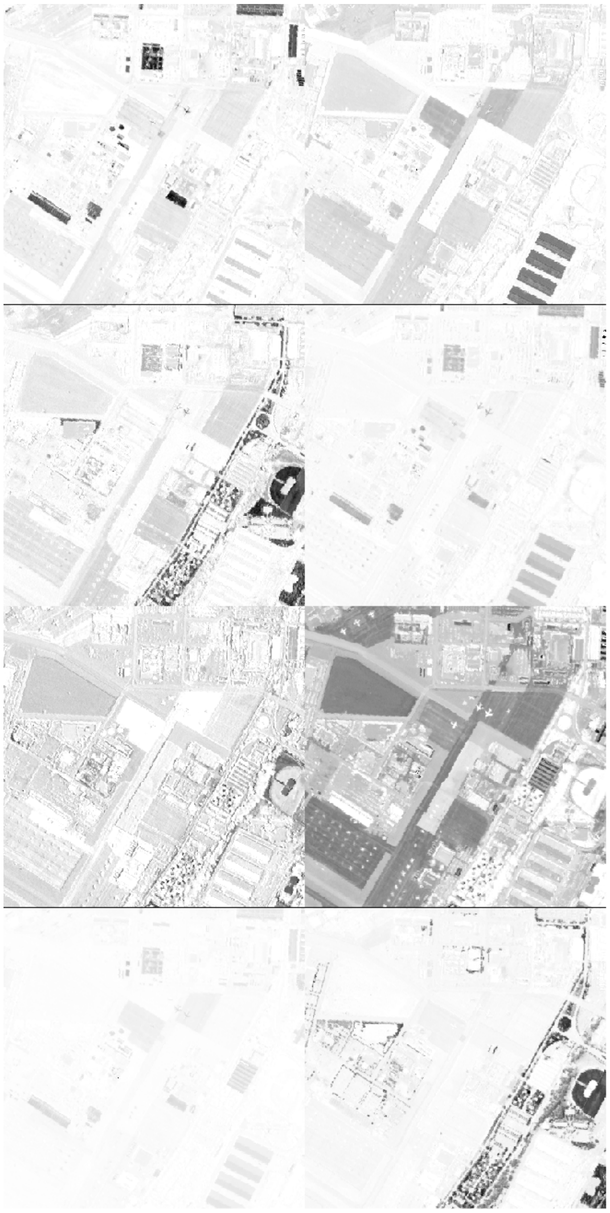

Fig. 6. The abundance map (matrix $\mathbf{H}$ computed with Eigen) for the San Diego airport image.

7. REFERENCES

[1] X. Fu, K. Huang, B. Yang, W.-K. Ma, and N.D. Sidiropoulos, "Robust volume minimization-based matrix factorization for remote sensing and document clustering," *IEEE Trans. on Signal Processing*, vol. 64, no. 23, pp. 6254–6268.

[2] N. Gillis, "Successive nonnegative projection algorithm for robust nonnegative blind source separation," *SIAM J. on Imaging Sciences*, vol. 7, no. 2, pp. 1420–1450, 2014.

[3] G. Zhou, S. Xie, Z. Yang, J.-M. Yang, and Z. He, "Minimum-volume-constrained nonnegative matrix factorization: Enhanced ability of learning parts," *IEEE Trans. on Neural Networks*, vol. 22, no. 10, pp. 1626–1637, 2011.

[4] N. Gillis, D. Kuang, and H. Park, "Hierarchical clustering of hyperspectral images using rank-two nonnegative matrix factorization," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 53, no. 4, pp. 2066–2078, 2015.

[5] N. Gillis and F. Glineur, "Accelerated multiplicative updates and hierarchical als algorithms for nonnegative matrix factorization," *Neural computation*, vol. 24, no. 4, pp. 1085–1105, 2012.

[6] Yu. Nesterov, "Efficiency of coordinate descent methods on huge-scale optimization problems," *SIAM J. on Optimization*, vol. 22, no. 2, pp. 341–362, 2012.