SIAM J. SCI. COMPUT.
Vol. 38, No. 5, pp. B645–B684

ANALYSIS ON A NONNEGATIVE MATRIX FACTORIZATION AND ITS APPLICATIONS[*]

YAT TIN CHOW[†], KAZUFUMI ITO[‡], AND JUN ZOU[§]

Abstract. In this work we perform some mathematical analysis on a special nonnegative matrix trifactorization (NMF) and apply this NMF to some imaging and inverse problems. We will propose a sparse low-rank approximation of positive data and images in terms of tensor products of positive vectors and investigate its effectiveness in terms of the number of tensor products to be used in the approximation. A new multilevel analysis (MLA) framework is suggested to extract major components in the matrix representing structures of different resolutions but still preserve the positivity of the basis and sparsity of the approximation. We will also propose and formulate a semismooth Newton method based on primal-dual active sets for the nonnegative factorization. Numerical results are given to demonstrate the effectiveness of the proposed method at capturing features in images and structures of inverse problems under no a priori assumption on the underlying structure in the data as well as to provide a sparse low-rank representation of the data.

Key words. nonnegative matrix factorization, clustering, feature extraction, multilevel analysis, inverse problems

AMS subject classifications. 15A23, 65F22, 65F30, 65F50, 78M25

DOI. 10.1137/15M1020824

1. Introduction to nonnegative matrix factorizations.

Nonnegative matrix factorization (NMF) has attracted a great deal of attention in the last decade because of its many important applications, e.g., in extraction of principal components, features, structures, and similarities inside a large set of data or an image. In general, an NMF for a given matrix $Y \in \mathbb{R}^{N\times M}$ is to generate a factorization of the form

$$Y \approx AP\,, \quad A \in \mathbb{R}^{N\times k},\ P \in \mathbb{R}^{k\times M}\,, \tag{1.1}$$

where the matrix factors A and P are nonnegative componentwise. This was studied as early as in 1994 [48] and was used for machine learning and data mining [39, 40]. The concept of NMF as k-means clustering for principal component analysis has been widely studied theoretically and numerically (see, e.g., [6, 13, 17, 18, 30, 43, 48, 54]); and the concept of trifactorization was used as a concurrent column and row clustering of data in [19]. In order to extract desired features as well as to reduce memory complexity, sparsity is often imposed in NMF using l_0 or l_1 regularization. Effective NMF toolboxes have also been developed to provide different choices of regularizers and constraints, e.g., the nonnegative matrix factorization toolbox in MATLAB [44].

[*]Submitted to the journal's Computational Methods in Science and Engineering section May 18, 2015; accepted for publication (in revised form) June 2, 2016; published electronically September 1, 2016.

http://www.siam.org/journals/sisc/38-5/M102082.html

Funding: The work of the second author was partially supported by a direct grant for research from CUHK. The work of the third author was supported by Hong Kong RGC grants (projects 405513 and 14306814).

[†]Department of Mathematics, University of California, Los Angeles, Los Angeles, CA 90095-1555 (ytchow@math.ucla.edu).

[‡]Department of Mathematics and Center for Research in Scientific Computation, North Carolina State University, Raleigh, NC 27695 (kito@unity.ncsu.edu).

[§]Department of Mathematics, The Chinese University of Hong Kong, Shatin, Hong Kong (zou@math.cuhk.edu.hk).

A convex model for NMF was suggested in [20], where the convex $l_{1,\infty}$-norm was used as the regularizer to enforce row sparsity. In an application of this convex model to hyperspectral end-member selections, the NMF succeeded in providing abundance maps of end-members representing different structures inside an image, e.g., roofs, trees, grass, soil, and road.

One of the motivations for NMF is a pursuit of linear dimensionality reduction, which aims to obtain an approximation of the data in the low-dimensional linear space. It is very useful in various aspects, such as image processing, low-rank matrix recovery, classification of documents/data, unmixing of spectral signatures, as in applications like computational biology [16], clustering [18], music analysis [21], community detection [53], and air emission control [48], etc. In addition to the conventional Frobenius norm model for NMF, there are also other norms/divergences, such as the Kullback–Leibler divergence [7, 51] and the Itakura–Saito distance for music analysis [21]. Some of these models are motivated by statistical considerations [50]. Although NMF is NP-hard in general and ill-posed, many algorithms have been developed to realize NMF. A popular family of algorithms is the two-block coordinate descent method, where the problem becomes convex after fixing one of the matrix factors. Alternative direction algorithms are another family, which yield mostly a decrease in objective functionals, and their convergence is guaranteed and often fast once one of A and P in the NMF falls into an appropriate subspace. Multiplicative updates are also a popular family [15, 40], and they are simple to implement and scale well. Their convergence may be guaranteed [7], but it is mostly very slow [28]. Other methods include the alternating least squares, which does not generally converge, but the alternating nonnegative least squares may converge very fast in practice with the help of active sets [36, 37, 38], and its convergence is guaranteed to a stationary point (considering the fact that this is a block Gauss–Seidel update) [26]. For alternating nonnegative least squares, the matrix factors are updated alternatively using approaches such as projected gradients [45], quasi-Newton [11], or fast gradient methods [27]. The hierarchical alternating least squares is a special coordinate descent method, which updates one column at a time and can be decoupled into the problems of a single nonnegative variable [5, 10, 12, 24, 29, 42, 46]. This method converges much faster than the multiplicative updates [22] and is guaranteed to converge to a stationary point [24]. A very important class of NMFs may be the separable/near-separable NMFs, in which the matrix can be factorized into two factors, with one consisting of the columns of the original matrix. This is especially useful in test-mining and hyperspectral unmixing. The choice of ranks in nonnegative matrix factorizations is usually a crucial but rather tricky issue. Some practically used techniques are “trial and error,” “estimation using SVD,” and “experts insights” [4, 35, 21]. We refer the reader to [23] for a detailed discussion about the development of NMFs in both theories and algorithms.

For an NMF of the form (1.1), the rows of P may be viewed as the basis vectors of the information contained in matrix Y. We may further impose P to be nearly orthonormal, i.e., $PP^T \approx I$. This is similar to a partition of unity in the underlying space, and the row vectors of P are similar to some indicator functions. In order to reduce memory complexity in storing the basis P, one may further add a sparsity constraint on P. The matrix A is an assignment matrix, which gives some special weighting to the corresponding basis vectors of P. It is our aim to obtain a sparse matrix A that has a very small number of nonzero entries. Therefore, A can be interpreted as some sparse assignments of linear combinations of basis vectors in P. As our interest is the case when matrix Y is nonnegative componentwise, we shall further require $A \geq 0$ componentwise. Such a constraint may be infeasible if Y is not nonnegative, and then one should relax the nonnegativity condition for A, but this is not the focus of the current work. The sparsity constraint on A ensures more concise information extraction. Moreover, we may also have a postprocess to sort (column) vectors of A in descending order in terms of magnitude, which can yield the most important basis vectors from matrix P. Using a standard l_1 regularization to impose sparsity for A and P and near-orthogonality for P, we may formulate the NMF for a nonnegative matrix Y as the following minimization problem:

$$\min_{A\geq 0, P\geq 0} ||Y - AP||_{F,2}^2 + \alpha||A||_{F,1} + \nu||P||_{F,1} + \gamma||PP^T - I||_{F,1} \tag{1.2}$$

over nonnegative matrices $A \in \mathbb{R}^{N\times k}$ and $P \in \mathbb{R}^{k\times M}$, where $||X||_{F,2} := \sqrt{\sum_{i,j}|X_{ij}|^2}$, $||X||_{F,1} := \sum_{i,j}|X_{ij}|$, and α, ν, γ are three regularization parameters. Under this model, we regard the dimension k as the (nonnegative) rank of A under the NMF throughout the paper.

We aim in this work to investigate a different but closely related nonnegative factorization model. Before we start our discussion on the model of our interest, we first discuss a classical matrix factorization for a general matrix to motivate the development of our nonnegative factorization model. A popular classical matrix factorization is the singular value decomposition (SVD), which helps obtain the best low-rank approximation of a matrix in the l_2 sense and extracts the most important components of the matrix based on the magnitude of their corresponding singular values. An SVD is of the form

$$Y = U\Sigma V^T, \tag{1.3}$$

where we can interpret the matrices U, V as bases of information and Σ as a weighting representing the importance of the corresponding basis vectors in U and V. Although this approach gives the best low-rank approximation of matrix Y in the l_2 norm after a truncation of Σ, the SVD factorization is unstructured and usually does not respect positivity, and the basis vectors of U and V are rather oscillatory. In particular, for a matrix Y that represents an image or a probability density function, its SVD does not give very useful information about the underlying structures of Y, e.g., indicating the regions of high probability, locating objects inside the image and recognizing its sparsity, etc. The major objective of this work is to achieve an NMF that may offer a more structural decomposition of the matrix while still preserving the positivity of the basis. Now, combining the nonnegativity constraints with a factorization form similar to SVD gives rise to the idea of a nonnegative matrix trifactorization [19]. For a given nonnegative matrix Y, we shall investigate in this work a nonnegative matrix trifactorization of the following form using l_1 regularization:

$$\begin{aligned} \min_{U\geq 0,\Sigma\geq 0,V\geq 0} & ||Y - U\Sigma V^T||_{F,2}^2 + \alpha||\Sigma||_{F,1} + \nu||U||_{F,1} \\ & + \nu||V||_{F,1} + \gamma||UU^T - I||_{F,1} + \gamma||VV^T - I||_{F,1}\,, \end{aligned} \tag{1.4}$$

where we may interpret the matrices $U \in \mathbb{R}^{N\times p}$ and $V \in \mathbb{R}^{M\times p}$ as the bases of information and $\Sigma \in \mathbb{R}^{p\times p}$ as a weighting matrix. We emphasize that the matrix Σ is not required to be diagonal in our setting here, but rather to be sparse. Though we see the similarity between SVD and nonnegative matrix trifactorization by their forms, we shall emphasize several features of the NMF very different from SVD. Therefore we do not consider our trifactorization as a general version of SVD. However, the trifactorization and SVD do have some connections, and the SVD was used sometimes to provide a good initial guess for NMF algorithms in the traditional setting of NMF [25]. Under this trifactorization model, we shall regard the dimension p of the middle matrix $\Sigma \in \mathbb{R}^{p\times p}$ as the (nonnegative) rank of A under this NMF throughout the paper.

We will mainly focus on the nonnegative matrix trifactorization model (1.4) in this work, and we will analyze this factorization model and introduce some algorithms to realize this factorization. We also propose the application of the aforementioned model for nonnegative matrix trifactorizations to various sets of data and images to extract their major components, which may represent some special structures or features, and obtain an approximation of the data with low memory complexity when the rank p is small, even when the original data and images do not share any sparsity structures. This should be quite useful in applications, considering the fact that the factorization gives a low-rank sparse approximation of the matrix in term of the tensor products of column and row vectors of U and V. As p is small, it requires a small storage of the columns and rows in the matrices U and V and a much smaller memory than the original matrix does. The sparsity of Σ is also very important for the reduction of memory complexity because we only need to store the respective columns and rows of the matrices U and V, e.g., u_i and v_j, where the corresponding entry σ_{ij} of matrix Σ is significant. The sparsities of U and V are equally important because u_i and v_j will then have a small number of nonzero entries and be inexpensive to store. These reasons suggest that we apply the above NMF model to various sets of data and images. However, the choice of an NMF model and its rank is usually very tricky and affects the features and performances of the resulting factorizations directly. We shall develop a theory to analyze the number of rank p to be used. To effectively implement the NMF, we propose and formulate the semismooth Newton method based on primal-dual active sets [33] for the resulting nonlinear optimizations, instead of the classical methods [17, 19]. Using the result of an NMF from the Newton method, we shall also propose a dissection of the image into levels by its order of importance.

We then proceed to develop a new multilevel analysis (MLA) framework for the images based on an NMF, aiming at extracting major components inside the matrix Y representing structures of different resolutions and achieving sparse low-rank approximations of different levels with positive bases. At each level, we hope to extract and represent finer features of the original image with sparse approximation by positive bases, compared with the previous level. Our MLA framework is partially motivated by the multiresolution analysis (MRA) in wavelets [14], but it is quite different in nature. The MRA framework is well established to provide successive approximations of increasing resolutions of a function by a shifting and scaling of a mother wavelet, but the basis functions generated from the mother wavelet do not have the same (positive) sign of the whole space. This is a very undesirable feature in our context. Hence, we introduce a new MLA framework, which shall respect the positivity of the basis for function/matrix approximation but still provide a multiresolution property similar to that of MRA. In our MLA framework, we introduce a nested sequence of linear spaces H_s, each of which represents a level of fineness, and define interpolation operators among these spaces at coarser and finer levels. The NMF is then performed on each level to obtain a positive sparse approximation. We would like to emphasize that the main purpose of either our NMF model or our MLA framework is only to identify and represent structures (of different scales) in the images or data. But we are neither hoping to reconstruct the data in full entity nor to achieve the high-quality compression of images to defeat any available well-developed compression techniques, e.g., wavelet/curvelet compression, JPEG, etc. Instead, we aim to compare the effectiveness of feature capturing of the new factorization with other existing methods. Numerical experiments show good resolutions of images and data can be achieved by this sparse approximation using the MLA framework of the NMF model, and some major features and components in the images and data can be extracted without any a priori assumption on their structures, such as sparsity and specific patterns.

The contribution of this work includes both theory and numerical algorithms, mainly in the following three aspects: (1) We develop a theory to provide an asymptotic estimate of an optimal choice of the dimension p in the matrix Σ in sections 2 and 3. Although it is often very practical and desirable to perform such analysis when a generative model of Y is assumed, or when more specific sets of data are considered [4, 35, 21], there are many applications where no prior knowledge of the structures of the matrix Y is available. In these cases, one might not even expect a nontrivial or meaningful factorization. However, we present an asymptotic theory from the probabilistic arguments that give us the best asymptotic estimate for an optimal p in the very general case when there is no prior information available. Our theoretical analysis provides a lower bound in the probability sense; namely, p is selected such that the lower bound of the probability is maximized. This theoretical estimate of p is further justified numerically in section 5, where the rank p is selected in all numerical examples for the MLA based on the asymptotic estimate, indicating that the choice is indeed nearly optimal to balance between the necessity to include a larger basis to represent the data and the aim to reduce the basis for sparse representation. (2) We introduce an MLA framework to provide an approximation of the data at different coarse levels. Although MLA and MRA share some similarities, we shall emphasize in the later discussions that there are two fundamental differences. The first major difference is that our MLA framework utilizes a positive basis and therefore respects the positivity and structure of the data. The second major difference is that MRA provides a direct sum decomposition of the L^2 space into different resolutions, but the basis functions are not positive, while MLA cannot provide a direct sum decomposition, but it retains the positivity of the basis and is crucial to our applications of interest in this work. (3) We propose and formulate a primal-dual semismooth Newton method [34] for solving the nonlinear and nonsmooth optimizations involved in our NMF. This method combines the semismooth Newton technique with the active-set principle to handle the nonlinearity and nonsmoothness of the objective functional (1.4), converges fast, and is computationally inexpensive. Our method may fall in a category similar to the alternating nonnegative least squares, but it has some fundamental differences. We first pair up the last two matrices and perform the factorization, and then further factorize the resulting matrices. This can be regarded as a variant of a block Gauss–Seidel minimization performing only one sweep. Active sets and semismooth Newton methods are both used to speed up the convergence, with low computational efforts. Our semismooth Newton method is more advantageous than some classical methods for trifactorization, e.g., the ones in [19] where the multiplicative update is used; and its performance and structure is comparable to other quasi-Newton methods in the alternating nonnegative least squares. Furthermore, the semismooth method deals effectively with the nonsmoothness in our NMF formulation (2.1) and produces sequences that converge to the solution to the necessary optimality system. The new method can provide more desirable factorizations in our numerical tests than the standard NMF algorithms that usually do not use the regularizations or use only the 2-norm regularizations [15, 48, 40, 5, 10, 24, 29, 42, 46]. Moreover, the popular coordinate-descent-type methods may not be applicable to the current objective functional (2.1) that contains the nonsmooth and nonseparable terms, and they are likely to produce sequences which may get stuck in nonstationary points.

This paper is organized as follows. In section 2 the general mathematical framework of nonnegative matrix trifactorization using l_1 regularization is stated, and an optimal choice of the dimension of matrix Σ is investigated. An MLA framework using NMF is introduced in section 3, and a semismooth Newton method based on primal-dual active sets for NMF is formulated in section 4. Applications of our framework to imaging and inverse problems are provided in section 5, providing numerical evidence for successful feature extractions and sparse low-rank representations of the data.

2. A nonnegative matrix trifactorization using l_1 regularization. In this section we shall specify the type of matrix trifactorizations for our subsequent consideration. For the purpose, we often write $\mathbb{R}^{N\times M}$ for the set of $N \times M$ matrices and $(\mathbb{R}^{N\times M})^+ \subset \mathbb{R}^{N\times M}$ for those with positive entries. Given a matrix $Y \in (\mathbb{R}^{N\times M})^+$, we define a functional $\mathcal{J}_p^{\alpha,\nu,\gamma} : (\mathbb{R}^{N\times p})^+ \times (\mathbb{R}^{p\times p})^+ \times (\mathbb{R}^{M\times p})^+ \to \mathbb{R}$ for a fixed set of parameters p, α, γ:

$$\begin{aligned}\mathcal{J}_p^{\alpha,\nu,\gamma}(U,\Sigma,V) := ||Y - U\Sigma V^T||^2_{F,2} + \gamma||\Sigma||_{F,1} + \nu||U||_{F,1}\\ + \nu||V||_{F,1} + \alpha||UU^T - I||_{F,1} + \alpha||VV^T - I||_{F,1}.\end{aligned} \tag{2.1}$$

Let $[\tilde{U}_p, \tilde{\Sigma}_p, \tilde{V}_p]$ be a minimizer of the functional; then we define an operator $\mathcal{I}_p^{\alpha,\nu,\gamma}$: $(\mathbb{R}^{N\times M})^+ \to (\mathbb{R}^{N\times M})^+$ by

$$\mathcal{I}_p^{\alpha,\nu,\gamma}(Y) := \tilde{U}_p\tilde{\Sigma}_p\tilde{V}_p = \sum_{i,j} \sigma_{ij}(\tilde{u}_p)_i \otimes (\tilde{v}_p)_j, \tag{2.2}$$

where $(\tilde{u}_p)_i, (\tilde{v}_p)_j$ denote the column and row vectors of $\tilde{U}_p$ and $\tilde{V}_p$, respectively, and σ_{ij} is the (i,j)th entry of the matrix $\tilde{\Sigma}_p$. Compared with the standard SVD, the above trifactorization presents some essential differences: l^1 regularizations are involved, and the weighting matrix $\tilde{\Sigma}_p$ is not required to be diagonal.

It is easy to see that a smaller p means a smaller memory for storing the matrix triple $[\tilde{U}_p, \tilde{\Sigma}_p, \tilde{V}_p]$. If $\tilde{\Sigma}_p$ is a sparse matrix, the memory complexity can be further reduced, as we only need to store the vectors $(\tilde{u}_p)_i$ and $(\tilde{v}_p)_j$ with nonzero σ_{ij}. In fact, for a generic matrix Y, if p can be chosen to be small such that $||Y - \mathcal{I}_p^{\alpha,\nu,\gamma}(Y)||_{F,2}$ is also small in some sense, then $[\tilde{U}_p, \tilde{\Sigma}_p, \tilde{V}_p]$ may serve as our desired sparse low-rank approximation of Y. However, it is clear that the smaller the value of p is, the poorer the approximation of Y by $\mathcal{I}_p^{\alpha,\nu,\gamma}(Y)$ will be. With a smaller p, the error $||Y - \mathcal{I}_p^{\alpha,\nu,\gamma}(Y)||_F$ is larger, and so is the objective $\mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p, \tilde{\Sigma}_p, \tilde{V}_p)$. Therefore, it is interesting and practically important to balance these two effects, and this will be analyzed in the next section.

We shall derive a lower bound for the probability of the error functional (2.1), which measures the discrepancy of the data from a possible nonnegative matrix trifactorization, being controlled by a threshold δ (see (2.17)). This lower bound suggests that we follow an asymptotic relation to choose an optimal rank p (see (2.18)). Then similar analysis is performed to exploit the possibility of dropping some basis vectors where the corresponding σ_{ij} in Σ is small. We remark that although a lower bound of the probability may not be sufficient to justify the maximization in the probability, it strongly suggests the optimality of the asymptotic estimate. This choice of the rank p will be further and fully justified numerically in section 5.

We would also like to point out that we shall develop our theoretical asymptotic estimate of an optimal choice of the dimension in the matrix Σ without any assumption on the generative model of Y. It will be very practical and desirable if a special form of the generative model is given, which may help us achieve a sharper bound and better description of the situations [4, 35, 21]. However, there are many applications where no prior knowledge of the structure of the matrix is available. Our theory shall give us a best asymptotic estimate to help us choose an optimal rank p when a generative model is lacking.

2.1. An optimal choice of the dimension p of the matrix Σ. In this section, we aim to find an optimal choice of the dimension p of the matrix $\tilde{\Sigma}$ in the decomposition (2.2) with respect to N, M by means of a probabilistic argument, under no prior assumption on the structures of Y. In our analysis we shall assume that Y is entrywise independent and identically distributed (i.i.d.) for simplicity of presentation. We first derive a lower bound in terms of p, N, M, δ of the probability that there exists a triple $[U, \Sigma, V]$ such that $\mathcal{J}_p^{\alpha,\nu,\gamma}(U, \Sigma, V) < \delta$ for a given small δ (Lemma 2.4). Then we maximize the lower bound over p to obtain an asymptotic choice of p (cf. (2.18)).

The objective value $\mathcal{J}_p^{\alpha,\nu,\gamma}(U_p, \Sigma_p, V_p)$ reflects the deviations of matrices U_p, V_p from being orthogonal, the sparsity of U_p, Σ_p, V_p, and the error of the approximation of Y by $\mathcal{I}_p^{\alpha,\nu,\gamma}(Y)$. Thus if we have $\mathcal{J}_p^{\alpha,\nu,\gamma}(U, \Sigma, V) < \delta$ for some $[U, \Sigma, V]$, then

$$||Y - \mathcal{I}_p^{\alpha,\nu,\gamma}(Y)||_{F,2}^2 \le \mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p, \tilde{\Sigma}_p, \tilde{V}_p) \le \mathcal{J}_p^{\alpha,\nu,\gamma}(U, \Sigma, V) < \delta\,.$$

This strongly suggests that our optimal choice of the rank p provided by our analysis is legitimate.

We begin by showing the following several simple yet important lemmas concerning a set of i.i.d. random vectors. Using these results, one can derive a lower bound for the probability of the error functional (2.1) being controlled by a threshold δ (see (2.17)). This lower bound suggests that we follow an asymptotic estimate (cf. (2.18)) to choose the optimal p, which is very important to our subsequent analysis.

Now we are ready to present the following few auxiliary but important results for our further development.

Lemma 2.1. *Consider a set of i.i.d. random vectors $\{X_i\}_{i=1}^N \in [0,1]^d$, where the probability distribution $d\mathbb{P}_X = f dx$ with dx denoting the standard Lebesgue measure and $0 < C_1 < f < C_2 < \infty$. Then the probability of the vectors $\omega_i := X_i/||X_i||_2$ that can be approximated by p points $\{P_i\}_{i=1}^p \in \mathbb{S}^{d-1} \bigcup [0,1]^d$ within an error of small $\varepsilon > 0$ can be bounded by*

$$p^N (C_3\varepsilon)^{(d-1)N} \le \mathbb{P}\left(\exists \{P_i\}_{i=1}^p \ s.t.\ \{\omega_i\}_{i=1}^N \subset \bigcup_{1\le i \le P} B_\varepsilon(P_i)\right) \le p^N (C_4\varepsilon)^{(d-1)N} \tag{2.3}$$

for two positive constants C_3 and C_4 depending on the distribution $f dx$.

Proof. By the assumption on the i.i.d. random vectors $\{X_i\}_{i=1}^N \in [0,1]^d$, it is direct to see that the random variables $\{\omega_i\}_{i=1}^N \in \mathbb{S}^{d-1}$ have a probability density $d\mathbb{P}_\omega = g d\omega$, where $d\omega$ is the standard surface measure and $\frac{\tilde{C}_1}{||\omega||_\infty} \le g \le \frac{\tilde{C}_2}{||\omega||_\infty}$ for some constants $\tilde{C}_1, \tilde{C}_2$ which depend on C_1 and C_2. Then using the fact that for small

$\varepsilon > 0$, $C\varepsilon < \sin\varepsilon < \varepsilon$ for some $C > 0$ and the binomial theorem, we derive

$$\begin{aligned}
&\mathbb{P}\left(\exists\{P_i\}_{i=1}^{p} \text{ s.t. } \{\omega_i\}_{i=1}^{N} \subset \bigcup_{1\leq i\leq P} B_\varepsilon(P_i)\right)\\
&= \sum_{\sum_{i=1}^{p} N_i = N} \frac{N!}{\prod_i N_i!} \frac{1}{|\mathbb{S}^{d-1}\bigcap[0,1]^d|} \prod_i \int_{\mathbb{S}^{d-1}\bigcap[0,1]^d} \mathbb{P}(||\omega_i - K||_2 < \varepsilon)^{N_i} dK\\
&\geq \sum_{\sum_{i=1}^{p} N_i = N} \frac{N!}{\prod_i N_i!} (C_3\varepsilon)^{(d-1)\sum_i N_i} \geq p^N (C_3\varepsilon)^{(d-1)N}
\end{aligned}$$

for some $C_3 > 0$. The other inequality is similar. □

Lemma 2.2. *Consider a set of i.i.d. random vectors $\{P_i\}_{i=1}^{p} \in [0,1]^d$, where the probability distribution $d\,\mathbb{P}_\omega = f d\omega$ with $d\omega$ denoting the standard surface measure and $0 < C_1 < f < C_2 < \infty$. Then for $p \leq d$ the probability of the set of vectors P_i being almost mutually orthogonal within an error of small $\varepsilon > 0$ can be bounded by*

$$p!\, d\, (C_3\varepsilon)^{\frac{(p)(p-1)}{2}+(d-1)} \leq \mathbb{P}\left(|\langle P_i, P_j\rangle - \delta_{ij}| < \varepsilon\, \forall i,j\,\right) \leq p!\, d\, (C_4\varepsilon)^{\frac{(p)(p-1)}{2}+(d-1)} \tag{2.4}$$

for two positive constants C_3 and C_4 depending on the distribution fdx.

Proof. By a direct counting and the half angle formula, we obtain for $p \leq d$ that

$$\begin{aligned}
&\mathbb{P}\left(\langle|P_i, P_j\rangle - \delta_{ij}| < \varepsilon\, \forall i,j\right)\\
&\geq p!\, d\, (C_3\varepsilon)^{d-1} \prod_{1\leq i\leq p} (C_3\varepsilon)^i |(\mathbb{B}_1^i \times \mathbb{B}_1^{n-i}) \bigcap [0,1]^d|\\
&\geq p!\, d\, (C_3\varepsilon)^{\frac{(p)(p-1)}{2}+(d-1)}
\end{aligned}$$

for some $C_3 > 0$, where we have used the fact that $||P_i - P_j||^2 = 2 - 2\langle P_i, P_j\rangle$. The other inequality is similar. □

The next lemma follows directly from the previous two lemmas and will be very helpful later for us to derive an important inequality (cf. (2.17)).

Lemma 2.3. *Consider a set of i.i.d. random vectors $\{X_i\}_{i=1}^{N} \in [0,1]^d$, where the probability distribution $d\mathbb{P}_X = fdx$ with dx denoting the standard Lebesgue measure and $0 < C_1 < f < C_2 < \infty$. Then for $p \leq N$ the probability of the event $E_{p,\varepsilon}$ representing the existence of $\{P_i\}_{i=1}^{p}$ such that $\{\omega_i\}_{i=1}^{N} \subset \bigcup_{1\leq i\leq P} B_\varepsilon(P_i)$ and $|\langle P_i, P_j\rangle - \delta_{ij}| < \varepsilon$ for all i,j for a small $\varepsilon > 0$ can be bounded by*

$$\begin{aligned}
&\left(p^N - (p-1)^N\right) pl!\, d\, (C_3\varepsilon)^{\frac{p(p-1)}{2}+(d-1)(N+1)}\\
&\leq \mathbb{P}\left(E_{p,\varepsilon} \backslash E_{p-1,\varepsilon}\right) \leq \left(p^N - (p-1)^N\right) pl!\, d\, (C_4\varepsilon)^{\frac{p(p-1)}{2}+(d-1)(N+1)}
\end{aligned}$$

for two positive constants C_3 and C_4 depending on the distribution fdx, and therefore

$$\begin{aligned}
&\sum_{l=1}^{p} \left(l^N - (l-1)^N\right) l!\, d\, (C_3\varepsilon)^{\frac{l(l-1)}{2}+(d-1)(N+1)}\\
&\leq \mathbb{P}\left(E_{p,\varepsilon}\right) \leq \sum_{l=1}^{p} \left(l^N - (l-1)^N\right) l!\, d\, (C_4\varepsilon)^{\frac{l(l-1)}{2}+(d-1)(N+1)}.
\end{aligned}$$

Moreover, the following lower bound holds:

$$\mathbb{P}\left(E_{p,\varepsilon}\right) \geq dp^N (C_3\varepsilon)^{(d-1)(N+1)+\frac{(p)(p-1)}{2}}.$$

Proof. The following inequality follows directly from the arguments of the previous two lemmas:

$$\sum_{\substack{\sum_1^p N_i=N,\\ N_i>0}} \frac{N!}{\prod_i N_i!} p!\, d\, (C_3\varepsilon)^{\frac{p(p-1)}{2}+(d-1)(N+1)}$$
$$\leq \mathbb{P}\left(E_{p,\varepsilon}\backslash E_{p-1,\varepsilon}\right) \leq \sum_{\substack{\sum_1^p N_i=N,\\ N_i>0}} \frac{N!}{\prod_i N_i!} p!\, d\, (C_4\varepsilon)^{\frac{p(p-1)}{2}+(d-1)(N+1)} .$$

The last term can be readily simplified by the following summation:

$$\sum_{\sum_1^p N_i=N\,,\, N_i>0} \frac{N!}{\prod_i N_i!} = \sum_{\sum_1^p N_i=N} \frac{N!}{\prod_i N_i!} - \sum_{\sum_1^{p-1} N_i=N} \frac{N!}{\prod_i N_i!} = p^N - (p-1)^N\,,$$

so the first inequality in the lemma follows. The second inequality is a direct consequence of the first after taking a summation over p. The last inequality comes readily from the second one. ◻

We may notice from the arguments of the previous lemmas that all the probability estimates there involve two constants C_3 and C_4, which depend heavily on the constants C_1 and C_2 that control the probability distribution $f dx$. It would be very interesting to explore how these constants depend on f more explicitly, and the results will help us understand the dependence of the constant in the subsequent estimate of an optimal choice of the rank p (see (2.18)) on the distribution f.

Next, we wish to connect what we have proved in Lemma 2.3 to the probability of the error functional (2.1) being controlled by a threshold δ, which will help us derive a very important asymptotic estimate of the optimal rank p. For this purpose, we consider a general image $Y = \sum_{i,j} Y_{ij}\, e_i \otimes e_j$ comprised of nonnegative entries. Without loss of generality, we may assume $\max_{i,j}|Y_{ij}| = 1$. If we write $Y_i := \sum_j Y_{ij}\, e_j$, and $\omega_i = Y_i/||Y_i||_2$, then $Y = \sum_i ||Y_i||_2\, e_i \otimes \omega_i$. If there exists a set of $\{P_i\}_{i=1}^p$ such that $\{\omega_i\}_{i=1}^N \subset \bigcup_{1\leq i\leq P} B_\varepsilon(P_i)$ and $|\langle P_i, P_j\rangle - \delta_{ij}| < \varepsilon$ for all i, j, we can write $\{\omega_{k_j}\}_{j=1}^{K_j} \in B_\varepsilon(P_j)$ for some K_j with $1 \leq j \leq p$. Then we should have

$$Y = \sum_i ||Y_i||_2 e_i \otimes \omega_i \approx \sum_{j=1}^p \sum_{k_j=1}^{K_j} ||Y_{k_j}||_2\, e_{k_j} \otimes P_j\,.$$

Writing $Q_j = (\sum_{k_j=1}^{K_j} ||Y_{k_j}||_2 e_{k_j})/\sqrt{\sum_{k_j=1}^{K_j} ||Y_{k_j}||_2}$ and $\sigma_{ij} = \delta_{ij}\sqrt{\sum_{k_j=1}^{K_j} ||Y_{k_j}||_2}$, then

$$Y \approx \sum_{i,j} \sigma_{ij}\, Q_i \otimes P_j,$$

where $|\langle P_i, P_j\rangle - \delta_{ij}| < \varepsilon$ and $|\langle Q_i, Q_j\rangle - \delta_{ij}| = 0$ for any i, j. By setting $\Sigma = (\sigma_{ij})$, $P = (P_i)^T$, $Q = (Q_j)$, we derive directly that

$$\left|\left| Y - \sum_{i,j} \sigma_{ij}\, Q_i \otimes P_j \right|\right|_{F_2} \leq \sum_{j=1}^p \sum_{k_j=1}^{K_j} ||Y_{k_j}||_2 |\omega_{k_j} - P_j| \leq ||Y||_{F,2}\varepsilon \leq NM\varepsilon\,,$$

which implies

$$\begin{aligned}\mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p,\tilde{\Sigma}_p,\tilde{V}_p) &\leq \mathcal{J}_p^{\alpha,\nu,\gamma}(Q,\Sigma,P)\\ &\leq ||Y||_{F,2}\varepsilon + \gamma\sum_j \sqrt{\sum_{k_j=1}^{K_j} ||Y_{k_j}||_2}\\ &\quad + \nu\sum_j ||Q_j||_1 + \nu\sum_i ||P_i||_1 + \alpha p(p-1)\varepsilon\\ &\leq NM\varepsilon + NM(\gamma+2\nu) + \alpha\, p(p-1)\varepsilon\,.\end{aligned}$$

Now if we assume Y is an entrywise i.i.d. random matrix, then both the columns and rows are i.i.d. vector-valued random variables, and so are the normalized columns and rows. Therefore we may apply Lemma 2.3 to get that the probability of the event $E_{p,\varepsilon}$, such that the above bound holds, can be bounded below by

$$\begin{aligned}\mathbb{P}\left(E_{p,\varepsilon}\right) &\geq \sum_{l=1}^{p}\left(l^N-(l-1)^N\right) l!\, M\,(C_3\varepsilon)^{\frac{l(l-1)}{2}+(M-1)(N+1)}\\ &\geq M\,p^N (C_3\varepsilon)^{(M-1)(N+1)+\frac{(p)(p-1)}{2}}\,.\end{aligned}$$

Similarly, switching the columns and rows of the image, we may follow the above argument again to conclude the same with N, M swapped. Combining the above two statements, we come to

$$\begin{aligned}&\mathbb{P}\left(\mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p,\tilde{\Sigma}_p,\tilde{V}_p) < NM\varepsilon + NM(\gamma+2\nu) + \alpha\, p(p-1)\varepsilon\right)\\ &\geq \sum_{l=1}^{p}\left(l^{\max(N,M)}-(l-1)^{\max(N,M)}\right) l!\,\min(N,M)\,(C_3\varepsilon)^{\mu(N,M,l)}\\ &\geq \min(N,M)p^{\max(N,M)}(C_3\varepsilon)^{\mu(N,M,p)}\,,\end{aligned}$$

where the function $\mu(\,\cdot\,,\,\cdot\,,\,\cdot\,)$ is defined for all $N, M, l \in \mathbb{N}$ by

$$\mu(N,M,l) := \frac{l(l-1)}{2} + NM - |N-M| - 1\,. \tag{2.5}$$

If we further choose the parameter $\gamma + 2\nu \leq (K-1)\varepsilon$ for some $K > 1$, we can deduce the following lemma.

Lemma 2.4. *For any small $\varepsilon > 0$ and for all $N, M \in \mathbb{N}$, it holds that*

$$\mathbb{P}\left(\mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p,\tilde{\Sigma}_p,\tilde{V}_p) < \left(KNM + \min(N,M)^2\right)\varepsilon\right)$$

$$\geq \sum_{l=1}^{p}\left(l^{\max(N,M)}-(l-1)^{\max(N,M)}\right) l!\,\min(N,M)\,(C_3\varepsilon)^{\mu(N,M,l)} \tag{2.6}$$

$$\geq \min(N,M)p^{\max(N,M)}(C_3\varepsilon)^{\mu(N,M,p)}, \tag{2.7}$$

where the function $\mu(\,\cdot\,,\,\cdot\,,\,\cdot\,)$ is defined as in (2.5) *and γ is such that $\gamma+2\nu \leq (K-1)\varepsilon$ for some $K > 1$.*

Before we derive a sharp bound of an optimal choice for p from (2.6), let us consider a rough lower bound introduced in the last inequality (2.7). Clearly, for the function

$$F(p) := \min(N,M) p^{\max(N,M)} \, (C_3\varepsilon)^{\mu(N,M,p)}$$

for $p \geq 1$, it is easy to see that

$$F'(p) = \frac{F(p)}{p}\left(\max(N,M) + \frac{|\log(C_3\varepsilon)|}{16} - |\log(C_3\varepsilon)|\left(p - \frac{3}{4}\right)^2\right)\left\{\begin{matrix} > \\ = \\ < \end{matrix}\right\} 0 \ ,$$

namely

$$p \left\{\begin{matrix} < \\ = \\ > \end{matrix}\right\} \frac{3}{4} + \sqrt{\frac{1}{16} + \frac{\max(N,M)}{|\log(C_3\varepsilon)|}} \, .$$

Therefore we can propose a primitive optimal choice of p to maximize the lower bound of the possibility $\mathbb{P}\big(\mathcal{J}_p^{\alpha,\nu,\gamma}([\tilde{U}_p, \tilde{\Sigma}_p, \tilde{V}_p]) < \big(KNM + \min(N,M)^2\big)\varepsilon\big)$, i.e., to choose

$$p = \sqrt{\frac{\max(N,M)}{|\log(C_3\varepsilon)|}} \tag{2.8}$$

for large N, M. Following some basic substitutions, we obtain the following theorem.

THEOREM 2.5. *For any small $\delta > 0$, we have*

$$\mathbb{P}\left(\min_p \mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p, \tilde{\Sigma}_p, \tilde{V}_p) < \delta\right) \geq \min(N,M) \, p_{N,M,\delta}^{\max(N,M)} \, (C_3\varepsilon)^{\mu(N,M,p_{N,M,\delta})} \tag{2.9}$$

whenever $\gamma + 2\nu \leq (K-1)\,\varepsilon$, where $\varepsilon = \delta\left(KNM + \min(N,M)^2\right)^{-1}$ for some $K > 1$, the function $\mu(\,\cdot\,,\,\cdot\,,\,\cdot\,)$ is defined as in (2.5)*, and $p_{N,M,\delta}$ is the following constant:*

$$p_{N,M,\delta} := \sqrt{\frac{\max(N,M)}{|\log(C_3\varepsilon)|}} = \sqrt{\frac{\max(N,M)}{\log(KNM + \min(N,M)^2) - |\log\delta| - \log C_3}} \, . \tag{2.10}$$

When $M = N$, it is obvious that the above optimal choice of p for a fixed $\delta > 0$ is of the form

$$p = p_{N,N,\delta} = \sqrt{\frac{N}{2\log N - |\log\delta| - \log C_3 + \log(K+1)}} \sim \sqrt{\frac{N}{2\log N}} \tag{2.11}$$

as N goes to infinity. The last asymptotic relation actually gives a precise approximation and

$$\sqrt{\frac{N}{2\log N}} \leq p_{N,N,\delta} \leq \sqrt{\frac{N}{\log N}} \tag{2.12}$$

if N is large enough such that $N > C_3\delta^{-1}$. Hence (2.11) serves as an optimal choice of p for large N. Furthermore, with this choice of p, the memory complexity is asymptotically $\sqrt{\frac{2N^3}{\log N}}$ as N goes to infinity.

However, we note that the optimal choice of p obtained above is only based on a rough lower bound (2.7). In what follows, we deduce a sharper bound by using (2.6).

Since the summation in (2.6) always increases with p, we get an optimal choice of p by controlling the increment of (2.6) with respect to p. In order to do so, we investigate the ratio of the terms

$$a_l := \left(l^{\max(N,M)} - (l-1)^{\max(N,M)}\right) l! \,\min(N,M)\,(C_3\varepsilon)^{\mu(N,M,l)}\,,$$

explicitly given by

$$\frac{a_{l+1}}{a_l} = \frac{(l+1)^{\max(N,M)} - l^{\max(N,M)}}{l^{\max(N,M)} - (l-1)^{\max(N,M)}}\, l\, e^{-|\log(C_3\varepsilon)|(l+1)}\,.$$

From L'Hôpital's rule, we can directly see that for a fixed pair of N, M the ratio $a_{l+1}/a_l \to 0$ as $l \to \infty$. Therefore, given a small $\eta < 1$, there is always a $\hat p_{N,M,\eta,\varepsilon}$ such that $a_{l+1}/a_l \le \eta$ whenever $l > \hat p_{N,M,\eta,\varepsilon}$. Then for all $p > \hat p_{N,M,\eta,\varepsilon}$ we have that

$$\begin{aligned}
&\mathbb{P}\left(\mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde U_p, \tilde\Sigma_p, \tilde V_p) < \left(KNM + \min(N,M)^2\right)\varepsilon\right)\\
&\ge \sum_{l=1}^{\hat p_{N,M,\eta,\varepsilon}-1} \left(l^{\max(N,M)} - (l-1)^{\max(N,M)}\right) l!\,\min(N,M)\,(C_3\varepsilon)^{\mu(N,M,l)}\\
&\quad+ \frac{1}{1-\eta}\left((\hat p_{N,M,\eta,\varepsilon})^{\max(N,M)} - (\hat p_{N,M,\eta,\varepsilon}-1)^{\max(N,M)}\right)(\hat p_{N,M,\eta,\varepsilon})!\,\min(N,M)\,(C_3\varepsilon)^{\mu(N,M,\hat p_{N,M,\eta,\varepsilon})}
\end{aligned}$$

whenever $\gamma + 2\nu \le (K-1)\,\varepsilon$ and that the increment of p from $\hat p_{N,M,\eta,\varepsilon}$ onward brings insignificant increment to the summation in (2.6). Now we aim to find an explicit $\hat p_{N,M,\eta,\varepsilon}$ in terms of N, M, thus obtaining an optimal choice of p. By Hölder's inequality we readily derive

$$\begin{aligned}
a_{p+1}/a_p &= \frac{\sum_{i=0}^{\max(N,M)-1}(1+1/p)^i}{\sum_{i=0}^{\max(N,M)-1}(1-1/p)^i}\, p\, e^{-|\log(C_3\varepsilon)|(p+1)}\\
&\le \frac{p(p+1)^{\max(N,M)-1}}{(p-1)^{\max(N,M)-1}}\, e^{-|\log(C_3\varepsilon)|(p+1)}\,.
\end{aligned} \tag{2.13}$$

Now if we consider the smooth function

$$G(N_0,p) := \frac{p\,(p+1)^{N_0-1}}{(p-1)^{N_0-1}}\, e^{-|\log(C_3\varepsilon)|(p+1)}$$

for $p > 1$ and $N_0 > \frac{1}{2}|\log(C_3\varepsilon)| + 1 + \frac{1}{6|\log(C_3\varepsilon)|}$, then we see that

$$\begin{aligned}
&\frac{\partial}{\partial p} G(N_0,p)\\
&= G(N_0,p)\left(\frac{1}{p} + \frac{N_0-1}{p+1} - \frac{N_0-1}{p-1} - |\log(C_3\varepsilon)|\right)\\
&= -\left(|\log(C_3\varepsilon)|p^3 - p^2 - (|\log(C_3\varepsilon)| - 2N_0 + 2)p + 1\right)\frac{(p+1)^{N_0-2}}{(p-1)^{N_0}}\, e^{-|\log(C_3\varepsilon)|(p+1)}\\
&< 0\,.
\end{aligned}$$

Together with the fact that $G(N_0,p) \to +\infty$ as $p \to 1^+$, whereas $G(N_0,p) \to 0$ as $p \to +\infty$, one directly obtains that $G(N_0,\cdot\,)$ is monotonically decreasing for $p > 1$ from the value $+\infty$ down to 0. Fixing N_0, we then have a well-defined smooth monotone function $G(N_0,\cdot)^{-1} : (0,\infty) \to (1,\infty)$ by the inverse function theorem.

The implicit function $g : (\frac{1}{2}|\log(C_3\varepsilon)| + 1 + \frac{1}{6|\log(C_3\varepsilon)|}, \infty) \to (1, \infty)$ defined by $G(N_0, g(N_0)) = \eta$ is now well-defined and smooth by the implicit function theorem as $g(N_0) = [G(N_0, \cdot)]^{-1}(\eta)$. Moreover,

$$\begin{aligned} g' &= -\frac{\frac{\partial G}{\partial N_0}(N_0, g(N_0))}{\frac{\partial G}{\partial p}(N_0, g(N_0))} \\ &= -\log\left(\frac{g+1}{g-1}\right)\left(\frac{1}{g} + \frac{N_0 - 1}{g+1} - \frac{N_0 - 1}{g-1} - |\log(C_3\varepsilon)|\right)^{-1} \\ &= \log\left(\frac{g+1}{g-1}\right)\frac{g(g+1)(g-1)}{|\log(C_3\varepsilon)|g^3 - g^2 - (|\log(C_3\varepsilon)| - 2N_0 + 2)g + 1}\,. \end{aligned}$$

Now note that with $N_0 > \frac{1}{2}|\log(C_3\varepsilon)| + 1 + \frac{1}{6|\log(C_3\varepsilon)|}$ and $g > 1$, we have $|\log(C_3\varepsilon)|g^3 - g^2 - (|\log(C_3\varepsilon)| - 2N_0 + 2)g + 1 > |\log(C_3\varepsilon)|$ and $0 < g'(N_0) < \infty$ for all N_0. Moreover, putting these inequalities back into the expression of g', we see that g satisfies the following differential inequality for large N_0:

$$g' \leq \log\left(\frac{g+1}{g-1}\right)\frac{2}{|\log(C_3\varepsilon)|} \leq \frac{4}{(g-1)|\log(C_3\varepsilon)|}\,.$$

Now, using the Gronwall–Bellman–Bihari inequality, we directly infer that

$$g \leq H^{-1}(H(a(\eta)) + N_0) \tag{2.14}$$

for some constant $a(\eta)$ depending only on η, where the function H is defined by

$$H(s) := \frac{|\log(C_3\varepsilon)|}{4}\int (s-1)ds = \frac{|\log(C_3\varepsilon)|(s-1)^2}{8} + \widehat{K_0}(\eta) \tag{2.15}$$

for some $\widehat{K_0}(\eta)$. Therefore the following inequality holds for g and some constants $\widehat{K_1}(\eta), \widehat{K_2}(\eta), \widehat{K_3}(\eta)$:

$$g \leq \sqrt{\frac{\widehat{K_1}(\eta)N_0 - \widehat{K_2}(\eta)}{|\log(C_3\varepsilon)|}} + \widehat{K_3}(\eta)\,.$$

Using $p_{N,M,\delta}$ defined in (2.10), we can choose $\hat{p}_{N,M,\eta,\varepsilon}$ such that

$$\hat{p}_{N,M,\eta,\varepsilon} = K_\eta\sqrt{\frac{\max(N,M)}{|\log(C_3\varepsilon)|}} = K_\eta p_{N,M,\delta} \tag{2.16}$$

for some K_η depending on η; then for all

$$p > \hat{p}_{N,M,\eta,\varepsilon} \geq g\left(\max(N,M)\right) = \left[G\left(\max(N,M), \cdot\right)\right]^{-1}(\eta),$$

we have

$$\frac{p(p+1)^{\max(N,M)-1}}{(p-1)^{\max(N,M)-1}}\,e^{-|\log(C_3\varepsilon)|(p+1)} < \eta\,.$$

Hence the growth of the probability $\mathbb{P}\big(\mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p, \tilde{\Sigma}_p, \tilde{V}_p) < \big(KNM + \min(N,M)^2\big)\varepsilon\big)$ with respect to p becomes insignificant for $p > \hat{p}_{N,M,\eta,\varepsilon}$. This gives another optimal choice of p. Surprisingly, we notice that $\hat{p}_{N,M,\eta,\varepsilon} \sim p_{N,M,\delta}$, i.e., the two choices of p are of the same order. This leads to the following results.

THEOREM 2.6. *The following probability bound holds for any small $\delta > 0$:*

$$\begin{aligned}&\mathbb{P}\left(\mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde U_p, \tilde\Sigma_p, \tilde V_p) < \delta\right)\\ &\geq \sum_{l=1}^{p} \left(l^{\max(N,M)} - (l-1)^{\max(N,M)}\right) l!\, \min(N,M)\ (C_3\varepsilon)^{\mu(N,M,l)}\end{aligned} \tag{2.17}$$

whenever $\gamma + 2\nu \leq (K-1)\,\varepsilon$, where $\varepsilon = \delta\left(KNM + \min(N,M)^2\right)^{-1}$ for some $K > 1$ and the function $\mu(\,\cdot\,,\,\cdot\,,\,\cdot\,)$ is defined as in (2.5)*. For a given small constant η, the growth of the summation above with respect to p can be controlled by η when $p > K_\eta\, p_{N,M,\delta}$ for some K_η depending only on η, where $p_{N,M,\delta}$ is defined as* (2.10)*.*

The above theorem is crucial for suggesting our optimal choice of an asymptotic p. Indeed, we can easily see that $||Y - \mathcal{I}_p^{\alpha,\nu\gamma}(Y)||_{F,2}^2 < \delta$ if $\mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde U_p, \tilde\Sigma_p, \tilde V_p) < \delta$. And in the particular case when $M = N$, the following asymptotic order for p,

$$p \sim \sqrt{\frac{N}{\log N}}, \tag{2.18}$$

is basically an optimal choice of p, and they are equivalent up to a multiplicative constant whenever $N > C_3\delta^{-1}$. Following this optimal choice, the memory complexity grows in the order $\sqrt{\frac{N^3}{\log N}}$ as N goes to infinity.

We may notice that the multiplicative asymptotic estimate (2.18) has its multiplicative constant that depends strongly on the distribution $f dx$. A thorough investigation of the dependence is important for a better understanding of the asymptotic estimate.

2.2. Effects of magnitudes of entries in matrix $\boldsymbol{\Sigma}$.

In this subsection, we discuss a further reduction of the memory complexity by truncating the matrix $\tilde\Sigma_p = (\sigma_{ij})$ and dropping the less important components $(\tilde u_p)_i \otimes (\tilde v_p)_j$ in (2.2) in a way that it still serves as a good approximation of the original matrix Y.

For doing so, we rearrange σ_{ij} from the largest value to the smallest one as $\sigma_{i_1j_1} \geq \sigma_{i_2j_2} \geq \cdots \geq \sigma_{i_{p^2}j_{p^2}}$, and we write $\tilde\sigma_l = \sigma_{i_lj_l} e_l \otimes e_l$ and $\tilde\Sigma_{p,\tilde p} = \sum_{l=1}^{\tilde p} \tilde\sigma_l$ as the truncated matrix for all $\tilde p \leq p^2$. The sequence $\{\tilde\sigma_l\}_{l=1}^{p^2}$ represents the components of $\tilde\Sigma_p$ in descending order by the importance of their magnitudes. Let $[\tilde U_p, \Sigma_p, \tilde V_p]$ be a minimizer of the functional (2.1); then we define a convenient operator $\mathcal{I}_{p,\tilde p}^{\alpha,\nu,\gamma}$: $\mathbb{R}^{N\times M} \to (\mathbb{R}^{N\times M})^+$ by

$$\mathcal{I}_{p,\tilde p}^{\alpha,\nu,\gamma}(Y) := \tilde U_p \tilde\Sigma_{p,\tilde p} \tilde V_p = \sum_{l=1}^{\tilde p} \sigma_{i_lj_l} (\tilde u_p)_{i_l} \otimes (\tilde v_p)_{j_l}\,. \tag{2.19}$$

The approximation $Y \approx \mathcal{I}_{p,\tilde p}^{\alpha,\nu,\gamma}(Y) = \tilde U_p \tilde\Sigma_{p,\tilde p} \tilde V_p$ is a truncation of the approximation (2.2) of Y up to $\tilde p$. This truncated approximation drops those less important components, and hence we need only save the vectors $(\tilde u_p)_{i_l}$ and $(\tilde v_p)_{j_l}$ for $1 \leq l \leq \tilde p$. This further reduces the memory complexity and serves as our desired sparse low-rank approximation of Y. Next, we give a brief analysis for this truncated approximation of Y. We obtain directly from the pigeon-hole principle that

$$\begin{aligned}\mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p, \tilde{\Sigma}_{p,\tilde{p}}, \tilde{V}_p) &< \mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p, \tilde{\Sigma}_p, \tilde{V}_p) + ||I||_1 \sum_{i=0}^{p^2-\tilde{p}} \frac{1}{p^2-i} \\ &< \mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p, \tilde{\Sigma}_p, \tilde{V}_p) + ||I||_1 \int_{\frac{\tilde{p}}{p^2}}^{1} 1/x dx \\ &< \mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p, \tilde{\Sigma}_p, \tilde{V}_p) + NM \log\left(\frac{p^2}{\tilde{p}}\right) \\ &< \left((K+T)NM + \min(N,M)^2\right)\varepsilon\end{aligned}$$

whenever $\mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p, \tilde{\Sigma}_p, \tilde{V}_p) < \left((KNM + \min(N,M)^2\right)\varepsilon$ for some K and $\tilde{p} > e^{-T\varepsilon}p^2$ for some T. Now the following corollary is a direct consequence of this estimate combined with Theorem 2.5.

COROLLARY 2.7. *For some given constants K and L, let $\varepsilon = \delta((K+T)NM + \min(N,M)^2)^{-1}$ and $p_{N,M,\delta}$ be given by (2.10). Then the following inequality holds for any small $\delta > 0$, $\gamma + 2\nu \leq (K-1)\varepsilon$, and $\tilde{p} > e^{-T\varepsilon}p^2$:*

$$\mathbb{P}\left(\min_p \mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p, \tilde{\Sigma}_{p,\tilde{p}}, \tilde{V}_p) < \delta\right) \geq \min(N,M)\, p_{N,M,\delta}^{\max(N,M)} \left(C_3\varepsilon\right)^{\mu(N,M,p_{N,M,\delta})} .$$

3. Multilevel analysis (MLA) of nonnegative trifactorizations.

In this section, we introduce an MLA framework based on the trifactorization addressed in the previous section. We notice that, for a matrix Y, especially when it represents an image or an inhomogeneous medium inclusion, there are features of different scales in Y, which may usually represent different objects or several different parts of one object in the image. We aim at extracting these features of different scales and represent them in a sparse low-rank approximation in terms of tensor products. Therefore we introduce an MLA framework to NMF which helps us achieve a sparse representation of the features of multiple scales, ranging from the coarsest scale to the finest scale in the image Y. This MLA framework aims to identify the major components in the matrix Y which represent structures at multiple scales/levels of the image so that structures from large scales up to small scales in the image can be separately identified and sparsely represented. Our MLA framework is partially motivated by the MRA in wavelet analysis, which is widely used to capture different resolutions of a function or image as well as for compression purposes. However, an essential difference of our MLA framework from the MRA lies in its unique feature that the positivity of the basis for the function/matrix approximation is respected, while a multiresolution property similar to that in MRA is still achieved.

The most primitive idea of MRA is to successively approximate an L^2-function by dyadic shifts and dilations of a wavelet function ψ (a.k.a. the mother wavelet), which results in a multiple resolution of the L^2-function. But the mother wavelet ψ has a vanishing mean [14, 47], and it cannot have the same sign in the whole space. So the MRA approximation fails to represent an L^2-function f by positive basis functions. This is also true for higher dimensions. Therefore the MRA may not be desirable if we intend to approximate a positive function by a positive basis. This is often the case when the function/matrix represents an image or a probability density function, and it motivates us for a nonnegative version of a similar multilevel approximation of the function based on the NMF technique. We shall call it the MLA, in the hope that each finer level of approximation of the function by a positive basis shall represent an increasing resolution in some sense.

Next, we formulate a mathematical framework for the MLA in NMF. For the sake of exposition, we introduce several convenient operators for the subsequent discussion. We first define an interpolation operator $\iota_s : \mathbb{R}^{N\times M} \to \mathbb{R}^{\frac{N}{r^s}\times\frac{M}{r^s}}$ as the following averaging operator:

$$\iota_s(Y) := \sum_{1\leq i\leq N/r^s, 1\leq j\leq M/r^s} \frac{1}{r^{2s}} \sum_{k,l\in Q_{I_iJ_j}} Y_{kl} e_i \otimes e_j, \tag{3.1}$$

where $Q_{I_iJ_j}$ contains the entries with indices (i,j) satisfying $iN/r^s \leq k < (i+1)N/r^s, jM/r^s \leq l < (j+1)M/r^s$. We note that this operator gives an interpolation from a fine space $H_0 := \mathbb{R}^{N\times M}$ to a coarse space $H_s := \mathbb{R}^{\frac{N}{r^s}\times\frac{M}{r^s}}$, and the spaces H_s actually form a nested sequence of spaces, i.e., $H_s \subset H_l$ if $s > l$. Certainly one may consider more general nested sequences of spaces and interpolation operators. Then we define $\mathcal{I}^{\alpha,\nu,\gamma}_{s,p} : (\mathbb{R}^{N\times M})^+ \to (\mathbb{R}^{N\times M})^+$ by

$$\mathcal{I}^{\alpha,\nu,\gamma}_{s,p} := \iota_s^T \circ \mathcal{I}^{\alpha,\gamma}_{p} \circ \iota_s\,. \tag{3.2}$$

Let $\lfloor\cdot\rfloor$ be the floor function, and let $s_{\max}$ be an integer such that

$$s_{\max} \leq \lfloor \log(\min(N,M))/\log(r) \rfloor;$$

then the action $\mathcal{I}^{\alpha,\nu,\gamma}_{s,p}(Y)$ represents the approximation of the $(s_{\max}-s)$th level of the image Y by NMF. Similarly, we define $\mathcal{I}^{\alpha,\nu,\gamma}_{s,p,\tilde{p}} : (\mathbb{R}^{N\times M})^+ \to (\mathbb{R}^{N\times M})^+$ as a truncated approximation of the $(s_{\max}-s)$th level of Y by

$$\mathcal{I}^{\alpha,\nu,\gamma}_{s,p,\tilde{p}} := \iota_s^T \circ \mathcal{I}^{\alpha,\nu,\gamma}_{p,\tilde{p}} \circ \iota_s\,. \tag{3.3}$$

We may notice that our new MLA and the existing MRA share some similarities: they both present a dissection of the image into different slices (referred to as levels and scales in the respective MLA and MRA frameworks), with each level providing a specific coarse/fine level of information about the data. But there are some fundamental differences between MLA and MRA: the basis in an MRA framework cannot hold a same sign in the whole space, while the MLA approximation is represented by a positive basis. On the other hand, the L^2 space is decomposed into a direct sum in an MRA framework, and hence a summation over different layers gives back the original image; but in the MLA framework, the direct-sum structure is not maintained in order to enforce the positivity of the basis, and therefore the results from different levels cannot be combined directly or linearly into one image.

Now we are ready to investigate and analyze the error of the approximation given by this MLA framework. For the sake of simplicity, we write the summation $\sum_{I\in\{I_1,\dots,I_{N/r^2}\},J\in\{J_1,\dots,J_{M/r^2}\}} a(Y_{I,J})$ as $\sum_{I,J} a(Y_{I,J})$, where Y_{IJ} is the (I,J)th block of the matrix Y and $a(\cdot)$ is any function acting on these block matrices. Then it is easy to see by combining the arguments in the previous sections and the Poincaré inequality that

$$
\begin{aligned}
&||Y-\iota_s^T\circ\mathcal{I}_{p,\tilde{p}}^{\alpha,\nu,\gamma}\circ\iota_s(Y)||_{F,2}^2\\
&\le r^{2s}||\iota_s(Y)-\mathcal{I}_{p,\tilde{p}}^{\alpha,\nu,\gamma}\circ\iota_s(Y)||_{F,2}^2+\sum_{I,J}||\nabla_\delta Y_{IJ}||_{F,2}^2\\
&\le r^{2s}\mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p,\tilde{\Sigma}_{p,\tilde{p}},\tilde{V}_p)+\sum_{I,J}||\nabla_\delta Y_{IJ}||_{F,2}^2\\
&\le r^{2s}\mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p,\tilde{\Sigma}_{p,},\tilde{V}_p)+r^{2s}\left(r^{-2s}NM\log\left(\frac{p^2}{\tilde{p}}\right)\right)+\sum_{I,J}||\nabla_\delta Y_{IJ}||_{F,2}^2,
\end{aligned}
$$

where ∇_δ is the difference gradient operator defined as

$$(\nabla_\delta X)_{i,j}=(X_{i+1,j}-X_{i,j},X_{i,j+1}-X_{i,j})$$

for any matrix X, $[\tilde{U}_p,\tilde{\Sigma}_p,\tilde{V}_p]$ is an argument minimum of (2.1) with Y replaced by $\iota_s(Y)$, and $\tilde{\Sigma}_{p,\tilde{p}}$ is the truncation of $\tilde{\Sigma}_p$ up to $\tilde{p}$ (see section 2.2). Therefore if we can choose $[\tilde{U}_p,\tilde{\Sigma}_p,\tilde{V}_p]$ such that $\mathcal{J}_p^{\alpha,\nu,\gamma}(\tilde{U}_p,\tilde{\Sigma}_p,\tilde{V}_p)<r^{-2s}\big((KNM+\min(N,M)^2\big)\varepsilon$ and $\tilde{p}>e^{-T\varepsilon}p^2$ for some K and T, then

$$||Y-\mathcal{I}_{s,p,\tilde{p}}^{\alpha,\nu,\gamma}(Y)||_{F,2}^2\le\big((K+T)NM+\min(N,M)^2\big)\varepsilon+\sum_{I,J}||\nabla_\delta Y_{IJ}||_{F,2}\,.$$

Let $p_{r^{-s}N,r^{-s}M,\delta}$ be defined as in (2.11). We recall from the discussions in the previous section that the probability of the above event, denoted as $E_{p,\tilde{p},\delta}$, is bounded below by

$$\mathbb{P}(E_{p,\tilde{p},\delta})\ge r^{-s}\min(N,M)\,p_{r^{-s}N,r^{-s}M,\delta}^{r^{-s}\max(N,M)}\,(C_3\varepsilon)^{\mu(r^{-s}N,r^{-s}M,p_{r^{-s}N,r^{-s}M,\delta})}$$
$$\text{for}\quad\gamma+2\nu\le(K-1)\,\varepsilon\,.$$

In general, we may not expect that either $||\nabla_\delta Y||_{F,2}^2$ or $\sum_{I,J}||\nabla_\delta Y_{IJ}||_{F,2}^2$ can be controlled, since we have not imposed any regularity conditions for Y. However, if we further assume that Y has some regularity, for instance, $\sum_{I,J}||\nabla_\delta Y_{IJ}||_{F,2}^2<\hat{K}MN\varepsilon$, then

$$||Y-\mathcal{I}_{s,p,\tilde{p}}^{\alpha,\nu,\gamma}(Y)||_{F,2}^2\le\Big((K+T+\hat{K})NM+\min(N,M)^2\Big)\varepsilon\,.$$

Combining all the above arguments, we come to the following conclusion.

THEOREM 3.1. *Let* $K,T,\hat{K}$ *be given, let*

$$\varepsilon=-r^{2s}\delta\left((K+T+\hat{K})NM+\min(N,M)^2\right)^{-1},$$

and for any small $\delta>0$*, let* $E_{p,\tilde{p},\delta}$ *be the event that the following inequality holds:*

$$||Y-\mathcal{I}_{s,p,\tilde{p}}^{\alpha,\nu,\gamma}(Y)||_{F,2}^2\le\big((K+T)NM+\min(N,M)^2\big)\varepsilon+\sum_{I,J}||\nabla_\delta Y_{IJ}||_{F,2}^2\,;$$

then if $\tilde{p}$ *is chosen such that* $\tilde{p}>e^{T\varepsilon}p^2$*, we have for any* s *and* $\gamma+2\nu\le(K-1)\,\varepsilon$ *that*

$$\mathbb{P}\left(\bigcup_p E_{p,\tilde{p},\delta}\right)\ge r^{-s}\min(N,M)\,p_{r^{-s}N,r^{-s}M,\delta}^{r^{-s}\max(N,M)}\,(C_3\varepsilon)^{\mu\left(r^{-s}N,r^{-s}M,p_{r^{-s}N,r^{-s}M,\delta}\right)}\,, \tag{3.4}$$

where the functions $\mu(\cdot,\cdot,\cdot)$ and $p_{r^{-s}N,r^{-s}M,\delta}$ are defined as in (2.5) *and* (2.10), *respectively. For all s and $p < r^{-s}\min(N,M)$, we have*

$$\mathbb{P}\left(E_{p,\tilde{p},\delta}\right) \geq \sum_{l=1}^{p} \left(l^{r^{-s}\max(N,M)} - (l-1)^{r^{-s}\max(N,M)}\right) l!\, r^{-s}\min(N,M)\; (C_3\varepsilon)^{\mu(r^{-s}N,r^{-s}M,l)} \tag{3.5}$$

whenever $\tilde{p} > e^{T\varepsilon}p^2$ and $\gamma+2\nu \leq (K-1)\,\varepsilon$. For a given small constant η, the growth of the summation above with respect to p can be controlled by η when $p > K_\eta\, p_{r^{-s}N,r^{-s}M,\delta}$ for some K_η depending only on η. Furthermore, if the event $E_{p,\tilde{p},\delta}$ occurs and the inequality $\sum_{I,J} ||\nabla_\delta Y_{IJ}||_{F,2}^2 < \hat{K}MN\varepsilon$ holds, then

$$||Y - \mathcal{I}^{\alpha,\nu,\gamma}_{s,p,\tilde{p}}(Y)||_{F,2}^2 \leq \delta\,. \tag{3.6}$$

Now we can see from the above theorem that for a given threshold δ and $M = N$, if Y has the regularity such that $||\nabla_\delta Y||_{F,2}^2 < \tilde{K}\delta$ for some $\tilde{K} < 1$, then the lower bound of the probability of $||Y - \mathcal{I}^{\alpha,\gamma}_{s,p,\tilde{p}}(Y)||_{F,2}^2 < \delta$ is higher than that of $||Y - \mathcal{I}^{\alpha,\nu,\gamma}_{p,\tilde{p}}(Y)||_{F,2}^2 < \delta$ with an appropriately selected $\tilde{p}$. Furthermore, for each s, the optimal choice of p has the same order as $p_{r^{-s}N,r^{-s}M,\delta}$, which behaves asymptotically like

$$p \sim r^{-s/2}\sqrt{\frac{N}{\log N - 2s\log r}}\,, \tag{3.7}$$

with the memory complexity of $\mathcal{I}^{\alpha,\nu,\gamma}_{s,p,\tilde{p}}$ growing in the order $r^{-3s/2}\sqrt{\frac{N^3}{\log N - 2s\log r}}$ as N goes to infinity. This indicates that, by increasing s, the probability of a valid approximation by the sth level in MLA of NMF is increased and the memory complexity is decreased. Moreover, we observe from our subsequent numerical experiments that the resulting approximations $\mathcal{I}^{\alpha,\nu,\gamma}_{s,p,\tilde{p}}$ from larger values of s capture the coarser features of Y, and they achieve finer and finer features as s decreases.

4. Semismooth Newton method for nonnegative factorizations.

In this section, we propose and formulate an efficient and cost-effective numerical algorithm to realize the NMF for a given image or data Y, as we discussed in the previous sections. Instead of finding the optimal solution $[\tilde{U}_p, \Sigma_p, \tilde{V}_p]$ of the functional (2.1), we shall propose performing the following alternative two-stage NMF to obtain an approximation of $\mathcal{I}^{\alpha,\nu,\gamma}_p(Y)$:

$$Y \approx AV^T\,, \quad A^T \approx \Sigma^T U^T\,;\ \text{then form } Y \approx U\Sigma V^T\,. \tag{4.1}$$

In each of the above two NMFs, we minimize the functional (1.2) via a semismooth Newton method based on primal-dual active sets [34], which will be derived below.

As we recall, a big class of NMF algorithms falls in a category called the multiplicative updates [15, 40], which has a variant to guarantee convergence [7]. Although it is simple to implement and scales well, its convergence is very slow [28]. Another method is the alternating least squares, which does not converge generally, but the alternating nonnegative least squares can be fast in practice with the use of active sets [36, 37, 38], and it is guaranteed to converge to a stationary point [26]. For the alternating nonnegative least squares, the matrix factors are updated alternatively, e.g., using projected gradients [45], or accelerated by the quasi-Newton [11] or fast gradient methods [27]. One may also use the hierarchical alternating least squares, which is a coordinate descent method that updates one column at a time and can be decoupled into the problems of a single nonnegative variable [5, 10, 12, 24, 29, 42, 46].

This method converges to a stationary point [24] and is much faster than the multiplicative updates [22]. We refer the reader to [23] for a detailed discussion on the development of NMFs in both theories and algorithms.

Our method may fall in a category similar to the alternating nonnegative least square, but it has some fundamental differences. We first pair up the last two matrices, perform the factorization, and then further factorize the resulting matrices. This can be regarded as a variant of a block Gauss–Seidel minimization performing only one sweep. Active sets and semi-smooth Newton methods are both used to speed up the convergence, with low computational efforts. The semismooth Newton method is more advantageous than some classical methods for trifactorization, e.g., the multiplicative updates [17, 19]; and its performance and structure are comparable to some other quasi-Newton methods in the alternating nonnegative least squares. Moreover, the semismooth method deals effectively with the nonsmoothness in our NMF formulation (2.1), converges to the solutions to the necessary optimality system (4.8), and may provide a more desirable factorization in our numerical tests than the standard NMF algorithms which do not take the regularizations or take the 2-norm regularizations [15, 48, 40, 5, 10, 24, 29, 42, 46]. Furthermore, some existing methods, e.g., the coordinate descent method, may not be applicable to our functional (2.1) involving nonsmooth and nonseparable 1-norm terms, and they are likely to produce sequences that may get stuck in some nonstationary points.

Our two-stage NMF may not yield the optimal solution $[\tilde{U}_p, \Sigma_p, \tilde{V}_p]$ of the functional (2.1), but it generates a sufficiently fine approximation of $\mathcal{I}_p^{\alpha,\nu,\gamma}(Y)$, as we shall observe from our numerical experiments. More importantly, this two-stage process is more user-friendly and less expensive computationally, since the linearized systems of the functional (2.1) involved in the semismooth Newton iteration are much more convenient to evaluate numerically than the systems encountered when (2.1) is minimized directly.

4.1. Semismooth Newton method based on primal-dual active sets for NMF.

Before we present a two-stage NMF for an approximation of $\mathcal{I}_p^{\alpha,\nu,\gamma}(Y)$, we first discuss some mathematical properties of the important nonconvex minimization problem (1.2). The semismooth Newton method based on primal-dual active sets was studied in [34] to solve either convex or nonconvex nonsmooth optimization problems effectively by combining the ideas of active sets and Newton-type update. In this section, we formulate this method for solving the nonsmooth nonconvex optimization (1.2):

$$\min_{A\geq 0, P\geq 0} J(A,P) := ||Y - AP||_{F,2}^2 + \alpha||A||_{F,1} + \nu||P||_{F,1} + \gamma||PP^T - I||_{F,1}. \tag{4.2}$$

4.1.1. Complementary conditions.

We first recall two complementary conditions for the characterization of some constraint conditions from [34], which are crucial for the development of the algorithm in the subsequent analysis. For this purpose, we will need the subdifferential of the function $|\cdot| : \mathbb{R} \to \mathbb{R}$, which is the set-valued signum function defined by

$$\partial|\cdot|(x) = \begin{cases} 1 & \text{if } x > 0\,, \\ [-1,1] & \text{if } x = 0\,, \\ -1 & \text{if } x < 0\,. \end{cases} \tag{4.3}$$

With this definition, we are now ready to introduce the first complementarity condition which characterizes the set-valued subdifferential $\partial|\cdot|$ based on the following equivalence [34].

Lemma 4.1. *For any given constant $c > 0$, it holds that*

$$\lambda = \frac{\lambda + cx}{\max(1, |\lambda + cx|)} \Leftrightarrow \lambda \in \partial |\cdot|(x)\,. \tag{4.4}$$

The above theorem follows directly from a pointwise comparison of the corresponding set-valued functions. The condition $\lambda = (\lambda + cx)/\max(1, |\lambda + cx|)$ for a given $c > 0$ is regarded as a complementary condition characterizing the subdifferential $\partial|\cdot|$ [34], where the choice of c is arbitrary. However, in a practical implementation using the complementary condition, c is often chosen as a fixed constant that acts as a stabilization parameter.

Now, for any matrix A, we note that $||A||_{F,1} = \sum_{i,j} |A_{i,j}|$. Then, using a dual variable λ and the complementarity condition (4.1), the set-valued subdifferential function $\partial || \cdot ||_{F,1}(A)$ can be characterized by

$$\lambda_{i,j} = \frac{\lambda_{i,j} + cA_{i,j}}{\max(1, |\lambda_{i,j} + cA_{i,j}|)} \Leftrightarrow \lambda \in \partial || \cdot ||_{F,1}(A)\,. \tag{4.5}$$

We may often write this simply as $\lambda = \frac{\lambda + cA}{\max(1, |\lambda + cA|)}$, where the division, the maximum, and the absolute value are all taken pointwise.

Next we introduce a second complementary condition that is used to characterize an inequality constrain $x \geq 0$ [34]. For a functional $F : \mathbb{R}^N \to \mathbb{R}$, the constrained optimization

$$\min F(x) \quad \text{subject to} \quad x \geq 0 \tag{4.6}$$

can be reformulated into an equivalent augmented Lagrangian formulation [34], yielding the following result.

Theorem 4.2. *The necessary optimality conditions for the minimization problem* (4.6) *are given by*

$$0 \in \partial F(x) + \mu \quad \textit{and} \quad \mu = \min(\mu + cx, 0)\,. \tag{4.7}$$

The proof of this theorem follows from the same arguments in [34]. The condition $\mu = \min(\mu + cx, 0)$ for the dual variable μ is regarded as a complementary condition in [34], which serves as a characterization of the constraint $x \geq 0$. This complementary condition may also be regarded as a project of the solution to the convex set as the epigraph defined by the constraint.

4.1.2. Necessary optimality conditions for the optimization (4.2). By applying Theorem 4.2 and Lemma 4.1 and calculating the subdifferentials involved, we come to the following necessary optimality conditions for the optimization (4.2) using the primal-dual and other auxiliary variables.

Theorem 4.3. *The necessary optimality conditions for the optimization* (4.2) *can be given in terms of the primal-dual variables* $(A, P, R, L, \mu_A, \lambda_A, \mu_P, \lambda_P, \lambda_L)$ *and two* *constants c_1, c_2 by*

$$
\begin{cases}
0 &= 2APP^T - 2YP^T + \mu_A + \alpha\lambda_A, \\
\lambda_A &= \frac{\lambda_A + c_2 A}{\max(1, |\lambda_A + c_2 A|)}, \\
\mu_A &= \min(\mu_A + c_1 A, 0), \\
0 &= -2A^T Y + 2A^T AP + \mu_P + \nu\lambda_P + \gamma\lambda_L R, \\
\lambda_P &= \frac{\lambda_P + c_2 P}{\max(1, |\lambda_P + c_2 P|)}, \\
L &= PP^T - I, \\
R &= P \circ T + T \circ P^T \circ T, \\
\lambda_L &= \frac{\lambda_L + c_2 L}{\max(1, |\lambda_L + c_2 L|)}, \\
\mu_P &= \min(\mu_P + c_1 P, 0),
\end{cases}
\tag{4.8}
$$

where $T : \mathbb{R}^{M\times N} \to \mathbb{R}^{N\times M}$ is the transpose operator that maps A to A^T.

4.1.3. Semismooth Newton strategy. We introduced the necessary optimality conditions for solving the optimization problem (4.2) in the previous subsection. We shall now develop a semismooth Newton method for solving these optimality systems, which can be readily shown to be Newton differentiable [34]. To further develop our algorithm, we separate the variables $(A, P, R, L, \mu_A, \lambda_A, \mu_P, \lambda_P, \lambda_L)$ into three sets, i.e., (A, μ_A, λ_A), (P, μ_P, λ_P), and (L, R, λ_L), and solve for each set of variables independently. Clearly, the separated systems are easier for us to perform active-set techniques and greatly reduce the computational costs, and more importantly, each separated nonlinear system consists of many fewer variables and is therefore much more stable when performing semismooth Newton iterations. Together with the introduction of the active and inactive sets

$$
\begin{aligned}
\mathcal{A}_{A,1} &= \{(i,j) \,:\, (\mu_A)_{i,j} + c_1 A_{i,j} > 0\}\,, & \mathcal{I}_{A,1} &= \{(i,j) \,:\, (\mu_A)_{i,j} + c_1 A_{i,j} \le 0\}\,, \\
\mathcal{A}_{A,2} &= \{(i,j) \,:\, |(\lambda_A)_{i,j} + c_2 A_{i,j}| \le 1\}\,, & \mathcal{I}_{A,2} &= \{(i,j) \,:\, |(\lambda_A)_{i,j} + c_2 A_{i,j}| > 1\}\,, \\
\mathcal{A}_{P,1} &= \{(i,j) \,:\, (\mu_P)_{i,j} + c_1 P_{i,j} > 0\}\,, & \mathcal{I}_{P,1} &= \{(i,j) \,:\, (\mu_P)_{i,j} + c_1 P_{i,j} \le 0\}\,, \\
\mathcal{A}_{P,2} &= \{(i,j) \,:\, |(\lambda_P)_{i,j} + c_2 P_{i,j}| \le 1\}\,, & \mathcal{I}_{P,2} &= \{(i,j) \,:\, |(\lambda_P)_{i,j} + c_2 P_{i,j}| > 1\}\,, \\
\mathcal{A}_{L} &= \{(i,j) \,:\, |(\lambda_L)_{i,j} + c_2 L_{i,j}| \le 1\}\,, & \mathcal{I}_{L} &= \{(i,j) \,:\, |(\lambda_L)_{i,j} + c_2 L_{i,j}| > 1\}\,,
\end{aligned}
$$

we can separate (4.8) into three simple systems thanks to direct substitutions and pointwise comparisons of the complementary conditions:

(1) For a fixed P, we have $A = 0$ on $\mathcal{A}_{A,1} \bigcup \mathcal{A}_{A,2}$, while (A, λ_A) on $\mathcal{I}_{A,1} \bigcap \mathcal{I}_{A,2}$ satisfies

$$
2APP^T - 2YP^T + \alpha\lambda_A = 0\,, \quad \lambda_A|\lambda_A + c_2 A| - (\lambda_A + c_2 A) = 0\,. \tag{4.9}
$$

(2) For the fixed A, L, R, λ_L, we have $P = 0$ on $\mathcal{A}_{P,1} \bigcup \mathcal{A}_{P,2}$, while (P, λ_P) on $\mathcal{I}_P$ satisfies

$$
-2A^T Y + 2A^T AP + \nu\lambda_P + \gamma\lambda_L R = 0\,, \quad \lambda_P|\lambda_P + c_2 P| - (\lambda_P + c_2 P) = 0\,. \tag{4.10}
$$

(3) For a fixed P, we have $L = 0$ on $\mathcal{A}_L$, while (L, R, λ_L) on $\mathcal{I}_L$ satisfies

$$
L = PP^T - I\,, \quad R = P \circ T + T \circ P^T \circ T\,, \quad \lambda_L|\lambda_L + c_2 L| - (\lambda_L + c_2 L) = 0\,. \tag{4.11}
$$

For the nonlinear constraints with $\lambda_A, \lambda_P.$ and λ_L, we propose a semismooth Newton-step update as in [33] to solve the corresponding equations. To solve the first system (4.9), the following semismooth Newton update from (A, λ_A) to (A^+, λ_A^+) involving damping and regularization can be derived as in [33, 34]:

$$2A^+PP^T - 2YP^T + \alpha\lambda_A^+ = 0\,, \quad \lambda_A^+ - \frac{c_2}{d_A - 1}\left(I - a_A b_A^T\right)A^+ + a_A = 0\,, \tag{4.12}$$

where $a_A = \frac{\lambda_A}{\max(1,|\lambda_A|)}$, $b_A = \frac{\lambda_A + c_2 A}{|\lambda_A + c_2 A|}$, and $d_A = |\lambda_A + c_2 A|$. Similarly we can linearize the constraints for the variables λ_P and λ_L.

On the other hand, although the second equation in the third system (4.11) is linear, it is computationally expensive, as the transpose operator T is involved. We therefore suggest a semismooth Newton update for R from L and P instead of a direct substitution. The fact that $(L^h - L) = R^h(P^h - P) + O(h^2)$ holds when $L = PP^T - I$ and $(L^h, R^h, P^h) = (L, R, P) + O(h)$, together with the aforementioned linearization strategy for λ_L, provides us with the following semismooth Newton update from (L, R, P) to (L^+, R^+, P^+):

$$\begin{aligned} L^+ = P^+(P^+)^T - I\,, \quad R^+(P^+ - P) = (L^+ - L)\,, \quad \lambda_L^+ \\ = \frac{c_2}{d_L - 1}\left(I - a_L b_L^T\right)L^+ - a_L, \end{aligned} \tag{4.13}$$

where a_L, b_L, and d_L are given, respectively, by $a_L = \frac{\lambda_L}{\max(1,|\lambda_L|)}$, $b_L = \frac{\lambda_L + c_2 L}{|\lambda_L + c_2 L|}$, and $d_L = |\lambda_L + c_2 L|$.

4.1.4. Numerical algorithms. Combining all the techniques and results from the previous two subsections, we are ready to propose the semismooth Newton method based on primal-dual active sets for solving the optimality system (4.8) to tackle the minimization problem (4.2).

Semismooth Newton Algorithm 1. Given two constants c_1, c_2 and the initial guess $(A^0, P^0, \mu_A^0, \lambda_A^0, \mu_P^0, \lambda_P^0, \lambda_L^0)$.

For $k = 0, 1, \ldots, K$, do the following steps:

1. Compute $\mu_A^{(k)} := -2A^{(k)}P^{(k)}{P^{(k)}}^T + 2Y(P^{(k)})^T - \alpha\lambda_A^{(k)}$.
2. Set the active and inactive sets $\mathcal{A}_{A,i}^k$ and $\mathcal{I}_{A,i}^k$ for $i = 1, 2$:

$$\begin{aligned} \mathcal{A}_{A,1}^{(k)} &= \{(i,j) : (\mu_A)_{i,j}^{(k)} + c_1 A_{i,j}^{(k)} > 0\}\,, \quad & \mathcal{I}_{A,1}^{(k)} &= \{(i,j) : (\mu_A)_{i,j}^{(k)} + c_1 A_{i,j}^{(k)} \le 0\}\,, \\ \mathcal{A}_{A,2}^{(k)} &= \{(i,j) : |(\lambda_A)_{i,j}^{(k)} + c_2 A_{i,j}^{(k)}| \le 1\}\,, \quad & \mathcal{I}_{A,2}^{(k)} &= \{(i,j) : |(\lambda_A)_{i,j}^{(k)} + c_2 A_{i,j}^{(k)}| > 1\}\,. \end{aligned}$$

3. Compute $a_A^{(k)}, b_A^{(k)}, d_A^{(k)}$:

$$a_A^{(k)} := \frac{\lambda_A^{(k)}}{\max(1, |\lambda_A^{(k)}|)}\,, \quad b_A^{(k)} := \frac{\lambda_A^{(k)} + c_2 A^{(k)}}{|\lambda_A^{(k)} + c_2 A^{(k)}|}, \quad d_A^{(k)} := |\lambda_A^{(k)} + c_2 A^{(k)}|\,.$$

4. Set $A^{(k+1)} := 0$ on $\mathcal{A}_{A,1}^{(k)} \bigcup \mathcal{A}_{A,2}^{(k)}$; solve the system for $(A^{(k+1)}, \lambda_A^{(k+1)})$ on $\mathcal{I}_{A,1}^{(k)} \bigcap \mathcal{I}_{A,2}^{(k)}$:

$$\begin{cases} 0 &= 2A^{(k+1)}P^{(k)}{P^{(k)}}^T - 2Y(P^{(k)})^T + \alpha\lambda_A^{(k+1)}\,, \\ 0 &= \lambda_A^{(k+1)} - \frac{c_2}{d_A^{(k)} - 1}\left(I - a_A^{(k)}[b_A^{(k)}]^T\right)A^{(k+1)} + a_A^{(k)}\,. \end{cases}$$

5. Compute $\mu_P^{(k)} := 2(A^{(k+1)})^T Y - 2(A^{(k+1)})^T A^{(k+1)} P^{(k)} - \nu\lambda_P^{(k)} - \gamma\lambda_L^{(k)} R^{(k)}$.

6. Set the active and inactive sets $\mathcal{A}_{P,i}^k$ and $\mathcal{I}_{P,i}^k$ for $i = 1, 2$:

$$\mathcal{A}_{P,1}^{(k)} = \{(i,j) : (\mu_P)_{i,j}^{(k)} + c_1 P_{i,j}^{(k)} > 0\}\,, \quad \mathcal{I}_{P,1}^{(k)} = \{(i,j) : (\mu_P)_{i,j}^{(k)} + c_1 P_{i,j}^{(k)} \le 0\}\,,$$
$$\mathcal{A}_{P,2}^{(k)} = \{(i,j) : |(\lambda_P)_{i,j}^{(k)} + c_2 P_{i,j}^{(k)}| \le 1\}\,, \quad \mathcal{I}_{P,2}^{(k)} = \{(i,j) : |(\lambda_P)_{i,j}^{(k)} + c_2 P_{i,j}^{(k)}| > 1\}\,.$$

7. Compute $a_P^{(k)}, b_P^{(k)}, d_P^{(k)}$:

$$a_P^{(k)} := \frac{\lambda_P^{(k)}}{\max(1, |\lambda_P^{(k)}|)}\,, \quad b_P^{(k)} := \frac{\lambda_P^{(k)} + c_2 P^{(k)}}{|\lambda_P^{(k)} + c_2 P^{(k)}|}, \quad d_P^{(k)} := |\lambda_P^{(k)} + c_2 P^{(k)}|\,.$$

8. Set $P^{(k+1)} := 0$ on $\mathcal{A}_{P,1}^{(k)} \bigcup \mathcal{A}_{P,2}^{(k)}$; solve the system for $(P^{(k+1)}, \lambda_P^{(k+1)})$ on $\mathcal{I}_{P,1}^{(k)} \bigcap \mathcal{I}_{P,2}^{(k)}$:

$$\begin{cases} 0 &= -2(A^{(k+1)})^T Y + 2(A^{(k+1)})^T A^{(k+1)} P^{(k+1)} + \nu \lambda_P^{(k+1)} + \gamma \lambda_L^{(k)} R^{(k)}, \\ 0 &= \lambda_P^{(k+1)} - \frac{c_2}{d_P^{(k)} - 1} \left(I - a_P^{(k)} [b_P^{(k)}]^T \right) P^{(k+1)} + a_P^{(k)}\,. \end{cases}$$

9. Set the active and inactive sets $\mathcal{A}_L^{(k)}$ and $\mathcal{I}_L^{(k)}$:

$$\mathcal{A}_L^{(k)} = \{(i,j) : |(\lambda_L)_{i,j}^{(k)} + c_2 L_{i,j}^{(k)}| \le 1\}\,, \quad \mathcal{I}_L^{(k)} = \{(i,j) : |(\lambda_L)_{i,j}^{(k)} + c_2 L_{i,j}^{(k)}| > 1\}\,.$$

10. Compute $a_L^{(k)}, b_L^{(k)}, d_L^{(k)}$:

$$a_L^{(k)} := \frac{\lambda_L^{(k)}}{\max(1, |\lambda_L^{(k)}|)}\,, \quad b_L^{(k)} := \frac{\lambda_L^{(k)} + c_2 L^{(k)}}{|\lambda_L^{(k)} + c_2 L^{(k)}|}, \quad d_L^{(k)} := |\lambda_L^{(k)} + c_2 L^{(k)}|\,.$$

11. Set $L^{(k+1)} = 0$ on $\mathcal{A}_L^{(k)}$; evaluate $(L^{(k+1)}, R^{(k+1)}, \lambda_L^{(k+1)})$ on $\mathcal{I}_L^{(k)}$:

$$\begin{cases} L^{(k+1)} &= P^{(k+1)} (P^{(k+1)})^T - I\,, \\ R^{(k+1)} (P^{(k+1)} - P^{(k)}) &= \left(L^{(k+1)} - L^{(k)} \right)\,, \\ \lambda_L^{(k+1)} &= \frac{c_2}{d_L^{(k)} - 1} \left(I - a_L^{(k)} [b_L^{(k)}]^T \right) L^{(k+1)} - a_L^{(k)}\,. \end{cases}$$

A natural choice of the stopping criterion is based on the changes of the active sets: if the active sets from two consecutive iterations are the same, we may stop the iteration [34]. As the iteration goes on, A, P, L become more and more sparse, and the sizes of the linear systems involved drop drastically, so the inversions of the linear systems are more stable and less expensive computationally.

Finally, a few remarks are in order for effective implementations of the algorithm:

1. With the enforcement of the constraints $A, P \ge 0$ by the dual variables μ_A, μ_P, the algorithm ensures naturally $A^{(k)}, P^{(k)} \ge 0$ for all k if the initial guesses $A^{(0)}$ and $P^{(0)}$ are set to be nonnegative. Thus the algorithm can be simplified by setting the dual variables $\lambda_A^{(k)}$ and $\lambda_P^{(k)}$ to be $\lambda_A^{(k)} = \lambda_P^{(k)} = 1$ and drop the active/inactive sets $\mathcal{A}_{A,2}^{(k)}$, $\mathcal{I}_{A,2}^{(k)}$, $\mathcal{A}_{P,2}^{(k)}$, and $\mathcal{I}_{P,2}^{(k)}$.
2. In order to further simplify the algorithm, we may normalize the row vectors of P after step 8 so that PP^T has unitary diagonal entries. If this normalization is added, then $L^{(k)} \ge 0$ for all k. In this case, $\lambda_L^{(k)}$ can simply be set to be $\lambda_L^{(k)} = 1$, while $\mathcal{A}_L^{(k)}$ and $\mathcal{I}_L^{(k)}$ can be dropped.

3. In the development of our algorithm above, we assume $Y \geq 0$ entrywise; therefore it is natural to enforce the constraint $A \geq 0$. This nonnegativity condition for A is, however, infeasible and shall be dropped if Y is not nonnegative entrywise. In this case, nonetheless, we can still utilize the above algorithm for a nonnegative factorization with the following minor modification: drop the dual variable μ_A and the active/inactive sets $\mathcal{A}_{A,1}^{(k)}$ and $\mathcal{I}_{A,1}^{(k)}$.

4.2. Nonnegative matrix factorization of an image. With Semismooth Newton Algorithm 1 to minimize the functional (4.2), we are ready to propose an algorithm to approximate $\mathcal{I}_p^{\alpha,\nu,\gamma}(Y)$ in (2.2) and $\mathcal{I}_{p,\tilde{p}}^{\alpha,\nu,\gamma}(Y)$ in (2.19) for the NMF of an image Y.

Nonnegative Matrix Factorization Algorithm 2. Specify five parameters α, ν, γ, p, $\tilde{p}$.

1. Apply Semismooth Newton Algorithm 1 to find a minimizer $[A_0, V_0]$ of the problem

$$\min_{A\geq 0, V\geq 0} ||Y - AV^T||_{F,2}^2 + \alpha||A||_{F,1} + \nu||V||_{F,1} + \gamma||V^TV - I||_{F,1}.$$

2. Apply Semismooth Newton Algorithm 1 to find a minimizer $[\Sigma_0, U_0]$ of the problem

$$\min_{\Sigma\geq 0, U\geq 0} ||A_0^T - \Sigma^T U^T||_{F,2}^2 + \alpha||\Sigma||_{F,1} + \nu||U||_{F,1} + \gamma||U^TU - I||_{F,1}.$$

3. Form $\mathcal{I}_p^{\alpha,\nu,\gamma}(Y) := U_0\Sigma_0V_0^T$ from $[U_0, \Sigma_0, V_0]$.
4. Sort the entries of Σ_0 from the largest to the smallest as $\sigma_{i_1j_1} \geq \sigma_{i_2j_2} \geq \cdots \geq \sigma_{i_{p^2}j_{p^2}}$.
5. Compute $\tilde{\sigma}_l := \sigma_{i_lj_l} e_{i_l} \otimes e_{j_l}$; then form $\Sigma_{0,\tilde{p}} := \sum_{l=1}^{\tilde{p}} \tilde{\sigma}_l$.
6. Form the factorization $\mathcal{I}_{p,\tilde{p}}^{\alpha,\nu,\gamma}(Y) := U_0\Sigma_{0,\tilde{p}}V_0^T$.

4.3. Multilevel analysis algorithm based on NMF. Based on the results from an NMF, we can propose a multi-level analysis algorithm.

Multilevel Analysis Algorithm 3. Specify a scaling parameter r and a constant s_{max} such that $s_{max} < \log(\min(N,M))/\log r$; set parameters α, ν, γ and two arrays of parameters $[p(1), \ldots, p(s_{max})]$, $[\tilde{p}(1), \ldots, \tilde{p}(s_{max})]$.

For $s = 1, 2, \ldots, s_{max}$, do the following steps:

1. Compute $\iota_s(Y)$ as in (3.1).
2. Calculate $\mathcal{I}_{p(s),\tilde{p}(s)}^{\alpha,\nu,\gamma}[\iota_s(Y)]$ by Nonnegative Matrix Factorization Algorithm 2.
3. Calculate $\mathcal{I}_{s,p(s),\tilde{p}(s)}^{\alpha,\nu,\gamma}(Y) := \iota_s^T \circ \mathcal{I}_{p(s),\tilde{p}(s)}^{\alpha,\nu,\gamma} \circ \iota_s(Y)$.

5. Applications to photo and EIT images.

In this section we shall apply both the NMF and the MLA framework of an NMF suggested in section 4 to some photo images and several EIT images reconstructed by some direct sampling methods. We shall investigate two applications, the first one being an MLA for photo images using NMF, and the second one being an NMF over the images from an inversion algorithm for a broad class of coefficient determination inverse problems. In the first application, we aim at capturing features of different scales in an image and obtain a sparse low-rank representation of these features; in the second application, we hope to identify the principal components in the image, which correspond to the signals coming from the inhomogeneity in the corresponding inverse problems, and remove artifacts and noise from the images.

5.1. Applications to photo images. We perform now an MLA using NMF for several grey-scaled images Y. In view of the fact that an image can be represented by a positive function, and so are the major structures/objects inside these images, we are naturally motivated to use the NMF to identify the principal components of the image corresponding to these major objects in the figure and obtain a sparse representation of these objects and structures. MLA is employed to obtain these corresponding principal components representing structures/objects at multiple scales/levels of the image so that structures of large and small scales in the image can be separately identified and sparsely represented. We shall also aim to achieve a sparse representation which is robust to noise during transmission of data through channels. But we emphasize that we are neither aiming at reconstructing the image in full entity from all the NMF components in terms of tensor products nor hoping to obtain a very high compression ratio of memory complexity to defeat any well-developed compression techniques, e.g., wavelet/curvelet compression, JPEG, etc., since they are surely better candidates for compression. Rather, we aim to compare the ability of feature capturing of our newly introduced factorization with other existing methods.

In the subsequent three examples, we shall utilize the Multilevel Analysis Algorithm 3 to approximate $\mathcal{I}^{\alpha,\nu,\gamma}_{s,p(s),\tilde{p}(s)}(Y)$, in which the Nonnegative Matrix Factorization Algorithm 2 is used to calculate $\mathcal{I}^{\alpha,\nu,\gamma}_{p(s),\tilde{p}(s)}[\iota_s(Y)]$ and the Semismooth Newton Algorithm 1 is used to minimize (4.2) for the NMF. In all the following examples, the parameters in Algorithm 3 are set to $r = 2$, $\alpha = 0.2$, $\nu = 0.02$, $\gamma = 0.02$, whereas s_{max} is set differently in each example. Considering the theoretical optimal choice of p in (3.7), the array of parameters $p(s)$ is set to

$$p(s) = \left\lfloor T_1 \sqrt{\frac{\max(N, M)}{\max(1, \log \max(N, M) - 2s \log r)}} r^{-s/2} \right\rfloor \tag{5.1}$$

in all our examples, where $\lfloor \cdot \rfloor$ is the floor function and T_1 is a given constant. We observe from numerical experiments that this asymptotic formula (3.7) is, on one hand, necessary for good approximation of the desirable structures we hope to identify and, on the other hand, grows fairly slowly as the value $s_{max} - s$ increases and henceforth is a practical choice and very desirable for feature identifications and sparse representation. To ensure that the fidelity of the most important features in the image can be kept after dropping the less important components from $\widetilde{\Sigma}_{p,\tilde{p}}$, the parameter $\tilde{p}(s)$ is chosen by a threshold based on the l_1-norm of $\widetilde{\Sigma}_p$, i.e., as the first integer such that

$$\sum_{l=1}^{\tilde{p}(s)} \sigma_{i_l j_l} > T_2 \sum_{l=1}^{p(s)^2} \sigma_{i_l j_l} ,$$

where T_2 is a threshold which is smaller than 1. In all the following examples, T_1 and T_2 are always chosen as $T_1 = 3.5$ and $T_2 = 0.95$. A quantization process $\mathbb{Q}$ is performed on all the three matrices $[\tilde{U}_p, \widetilde{\Sigma}_{p,\tilde{p}}, \tilde{V}_p]$ which we get by Algorithm 2 as $\mathbb{Q}(A_{ij}) := \lfloor \frac{A_{ij}}{0.01} \rfloor$ for any matrix (A_{ij}). This is to minimize the number of possible choices of values in the matrix entries in order to embrace the possibility for an efficient entropy coding postprocessing after the NMF process and minimize memory complexity. The parameters c_1, c_2 in Algorithm 1 are always set to 1.

For the sake of comparisons between feature extraction, sparsity of representation, and robustness against noise in the transmission channel, we shall also compare the performance of NMF with those by the SVD and the JPEG compression. For a given image Y, the SVD with the level parameter s, $I_{SVD,s}$, is taken as

$$I_{SVD,s} := \iota_S^T(U\Sigma V^T) \qquad \text{with } \iota_s(Y) = U\Sigma V^T\,. \tag{5.2}$$

Again, the same quantization process $\mathbb{Q}$ is performed on the three matrices $[U, \Sigma, V]$ as described above to embrace the possibility for efficient entropy coding. Meanwhile, for the JPEG compression format, we follow the standard routine as in [52]. Namely we first perform a discrete cosine transform (DCT) on 8×8 pixel-blocks to give the DCT coefficients (D_{ij}) on each block and then perform the standard JPEG quantization process $C_{ij} = \lfloor \frac{D_{ij}}{(Q_{50})_{ij}} \rfloor$ with the given standard JPEG quantization matrix Q_{50} [52]. A level parameter s is introduced to define the image $I_{JPG,s}$ as the reconstruction of the JPEG from only the first 2^{3-s} Fourier coefficients in each 8×8 pixel-block for $s = 0, 1, 2, 3$. Note that, with this definition, only four levels are available for JPEG.

In order to test the robustness of the algorithms for feature preservation during the transmission process of data through channel, multiplication noise is added to simulate the scenario of data transmission through a noisy cable for each of the aforementioned algorithms, i.e., NMF, SVD, and JPEG. For the NMF process, multiplicative noise is added to the three matrices $[\tilde{U}_p, \tilde{\Sigma}_{p,\tilde{p}}, \tilde{V}_p]$ after quantization as

$$\begin{aligned} &(\tilde{U}_p^\zeta)_{ij} = (\tilde{U}_p)_{ij}(1 + \sigma\zeta_{ij})\,, \quad (\tilde{\Sigma}_{p,\tilde{p}}^\zeta)_{ij} = (\tilde{\Sigma}_{p,\tilde{p}})_{ij}(1 + \sigma\zeta_{ij}) \\ &(\tilde{V}_p)_{ij}^\zeta = (\tilde{V}_p)_{ij}(1 + \sigma\zeta_{ij})\,, \end{aligned} \tag{5.3}$$

where $\mathcal{I}^{\alpha,\nu,\gamma}_{p(s),\tilde{p}(s)}[\iota_s(Y)] := \tilde{U}_p\tilde{\Sigma}_{p,\tilde{p}}\tilde{V}_p^T$, $\mathcal{I}^{\alpha,\nu,\gamma}_{s,p(s),\tilde{p}(s)}(Y) := \iota_s^T \circ \mathcal{I}^{\alpha,\nu,\gamma}_{p(s),\tilde{p}(s)} \circ \iota_s(Y)$, σ is the noise level, and ζ is uniformly distributed between $[-1, 1]$. Multiplicative noise is used to preserve the positivity in the perturbed data, which is a necessary feature in NMF. Noisy reconstruction from the NMF is then given by

$$\left[\mathcal{I}^{\alpha,\nu,\gamma}_{s,p(s),\tilde{p}(s)}\right]^\zeta (Y) := \iota_s^T \tilde{U}_p^\zeta \tilde{\Sigma}_{p,\tilde{p}}^\zeta (\tilde{V}_p^\zeta)^T\,. \tag{5.4}$$

Similarly, for the SVD process, multiplicative noise is added in $[U, \Sigma, V]$ after quantization such that

$$U_{ij}^\zeta = U_{ij}(1 + \sigma\zeta_{ij})\,, \quad \Sigma_{ij}^\zeta = \Sigma_{ij}(1 + \sigma\zeta_{ij})\,, \quad V_{ij}^\zeta = V_{ij}(1 + \sigma\zeta_{ij})\,, \tag{5.5}$$

where $I_{SVD,s} := \iota_s^T(U\Sigma V^T)$ and $\iota_s(Y) := U\Sigma V^T$. The noisy reconstruction $I^\zeta_{SVD,s}$ is then taken as

$$I^\zeta_{SVD,s} := \iota_S^T(U^\zeta\Sigma^\zeta(V^\zeta)^T). \tag{5.6}$$

For the JPEG process, multiplicative noise is added in DCT coefficients on each 8×8 pixel-block after quantization:

$$C_{ij}^\zeta = C_{ij}(1 + \sigma\zeta_{ij})\,, \tag{5.7}$$

and the noisy reconstruction $I^\zeta_{JPG,s}$ comes as the dequantization of C^ζ by multiplication by Q_{50} followed by an inverse DCT. In all our numerical examples, we always set the noise level to be $\sigma = 25\%$.

The relative error of the reconstruction image I_{reconst} from each reconstruction method is quantified in the following manner on the quotient space of L^2 after taking an affine equivalence:

$$\varepsilon(I_{\text{reconst}}) := \frac{\min_{a,b\in\mathbb{R}} ||aI_{\text{reconst}} + b - Y||_{L^2}}{||Y||_{L^2}}.$$

This measurement of error is adopted because all the reconstructed images are shown such that the color scale gives only the relative contrast of the gray scale, and therefore an affine equivalence is taken for an appropriate measure of relative error. For each image, we shall also measure the memory complexity ratio of a given method, which is given as the ratio between the memory size of the data after performing the corresponding method and that of the original data. We would like to remark that the memory complexities for all the three methods (including JPEG) in our examples are computed based on its size before entropy coding; meanwhile, a same entropy coding technique can be applied to all the three methods considering the fact that all of them have undergone a quantization process.

Example 1. In this example, we set Y as the grey-scale image presented in Figure 1(left). The parameter s_{max} is chosen as $s_{max} = [\log(\min(N, M))/\log(r) - 3]$. The resulting images from MLA without noise are shown in Figure 2, whereas reconstructions with 25% noise are given in Figure 3. The memory complexity ratios for the $(s_{max} - s)$th level of the three methods and their respective relative L^2 errors with and without noise are shown in Table 1.

TABLE 1

$s_{max} - s$	:	1	2	3	4	5	6
p	:	20	24	24	28	34	42
$\tilde{p}$	:	142	177	152	153	195	332
Memory complexity ratio of NMF	:	0.0017	0.0033	0.0061	0.0116	0.0271	0.0573
Memory complexity ratio of SVD	:	0.0015	0.0036	0.0072	0.0168	0.0409	0.1011
Memory complexity ratio of JPEG	:	NA	NA	0.0154	0.0497	0.0982	0.1048
Relative L^2 error in NMF (with 0% noise)	:	0.2723	0.2567	0.2350	0.1878	0.1630	0.1561
Relative L^2 error in SVD (with 0% noise)	:	0.2733	0.2584	0.2342	0.1875	0.1591	0.1594
Relative L^2 error in JPEG (with 0% noise)	:	NA	NA	0.1855	0.0974	0.0689	0.0535
Relative L^2 error in NMF (with 25% noise)	:	0.2768	0.2631	0.2462	0.2029	0.1770	0.1689
Relative L^2 error in SVD (with 25% noise)	:	0.2755	0.2629	0.2456	0.2029	0.1754	0.1704
Relative L^2 error in JPEG (with 25% noise)	:	NA	NA	0.1941	0.1089	0.0711	0.0673

We can see from Figures 2 and 3 that in the absence of noise, although it is true that the NMF does not outperform SVD and JPEG of the same level, many reasonable details of different scales can already be captured in different levels of NMF, starting from the coarser image of the horse, then finer details, and afterwards the clear black-and-white strips on the horse. In each level, JPEG gives the best image of the three; however, it also needs a relatively high memory complexity in the same level. Meanwhile the NMF provides a representation of a relatively low memory complexity of the same layer. It is especially interesting to note that a memory complexity ratio of about 0.01 (before entropy coding) at level 4 can already give us many details of the horse. With the presence of noise, we can see that although the relative L^2 errors of both NMF and SVD are more or less the same, many coarser layers of SVD are not free from the contamination of noise in the form of vertical and horizontal strips in the background, and that the NMF gives a better shape of the horse. The NMF layers are affected by noise, but most of the nice details of the horse can still be kept. The JPEG stays the most robust against the noise; nonetheless, the

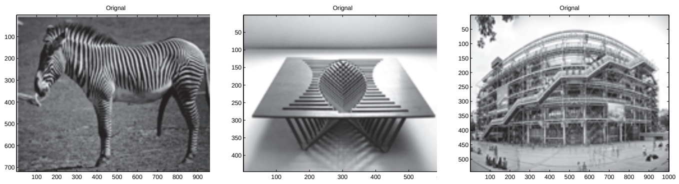


FIG. 1. *Original images in Example* 1 *(left), Example* 2 *(middle), and Example* 3 *(right).*

performance of NMF is also quite reasonable, considering the fact that NMF of the same layer usually requires less than half of the memory as JPEG.

Example 2. In this example, we set Y as the image presented in Figure 1(middle). The parameters are the same as in Example 1. The resulting images are shown in Figure 4. The memory complexity ratios for the $(s_{max}-s)$th level of the three methods and their respective relative L^2 errors with and without noise are shown in Table 2.

TABLE 2

$s_{max} - s$	:	1	2	3	4	5
p	:	22	21	23	28	34
$\tilde{p}$	:	143	113	118	146	208
Memory complexity ratio of NMF	:	0.0047	0.0075	0.0140	0.0309	0.0711
Memory complexity ratio of SVD	:	0.0053	0.0103	0.0225	0.0548	0.1330
Memory complexity ratio of JPEG	:	NA	0.0155	0.0364	0.0619	0.0716
Relative L^2 error in NMF (with 0% noise)	:	0.2945	0.2749	0.2503	0.1966	0.1693
Relative L^2 error in SVD (with 0% noise)	:	0.3024	0.2770	0.2553	0.2075	0.1717
Relative L^2 error in JPEG (with 0% noise)	:	NA	0.2487	0.1827	0.0884	0.0663
Relative L^2 error in NMF (with 25% noise)	:	0.3104	0.2842	0.2614	0.2225	0.1899
Relative L^2 error in SVD (with 25% noise)	:	0.3040	0.2867	0.2536	0.2298	0.2024
Relative L^2 error in JPEG (with 25% noise)	:	NA	0.2664	0.2025	0.1250	0.1082

From Figures 4 and 5, finer and finer details are reasonably captured as the level number of the NMF layers increases, while a reasonably low compression ratio is attained. This time the memory complexity of JPEG becomes comparable to NMF. At each level, JPEG still gives the best image of the three on the same layer; however, we notice that with the same level of memory complexity, some of the NMF images can provide a finer layer of details than the other two methods. With the presence of noise, we can see that the figures of all the three methods seem to be seriously contaminated, but the relative L^2 errors of NMF actually outperform those of the SVD in some layers. However, to our surprise, it seems that the figures of NMF seem more robust to keep the background clean, while the figures of the SVD are contaminated by random strips whereas the JPEG by random squares. In the coarsest level, the SVD does not give the shape of a table, but the NMF still generates a recognizable shape. Moreover, the most detail of the table in the finer level is still reasonably kept by the NMF in the presence of noise.

Example 3. In this last imaging example, we use the same set of parameters as for Examples 1 and 2 except that we now set $s_{max} = [\log(\min(N, M))/\log(r) - 4]$. Y is set as the image in Figure 1(right), and the resulting images are shown in Figure 6. The memory complexity ratios for the $(s_{max} - s)$th level of the three methods and their respective relative L^2 errors with and without noise are shown in Table 3.

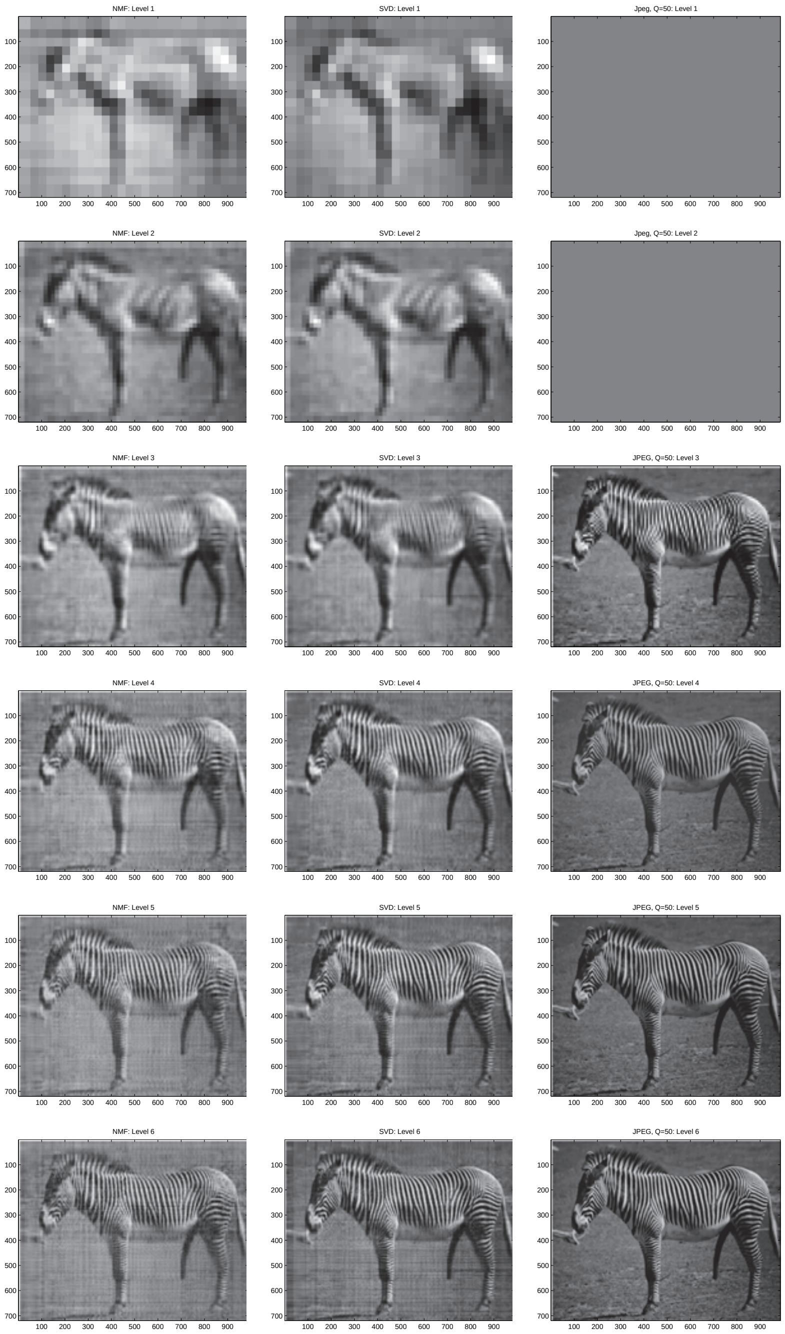


FIG. 2. *MLA for the image in Example* 1 *using NMF without noise.*

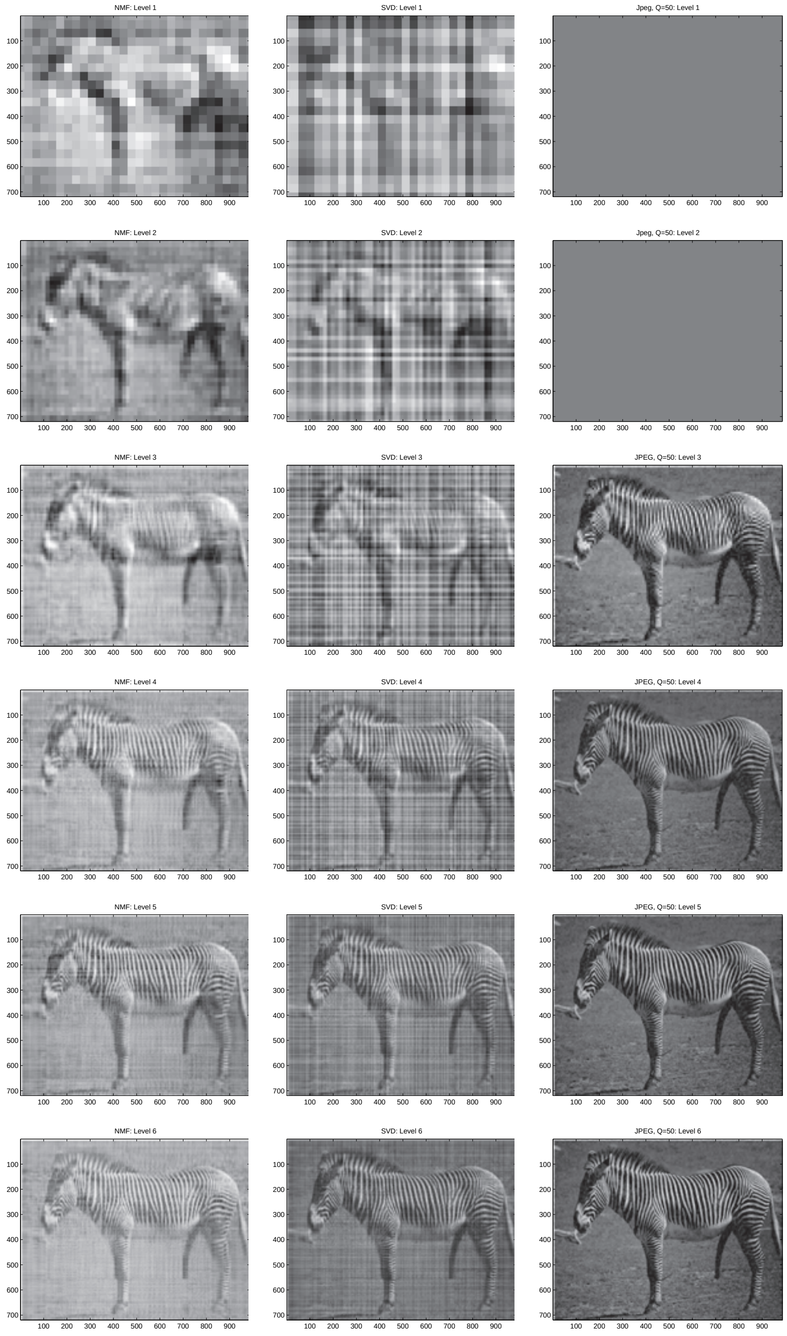


Fig. 3. *MLA for the image in Example* 1 *using NMF with* 25% *noise.*

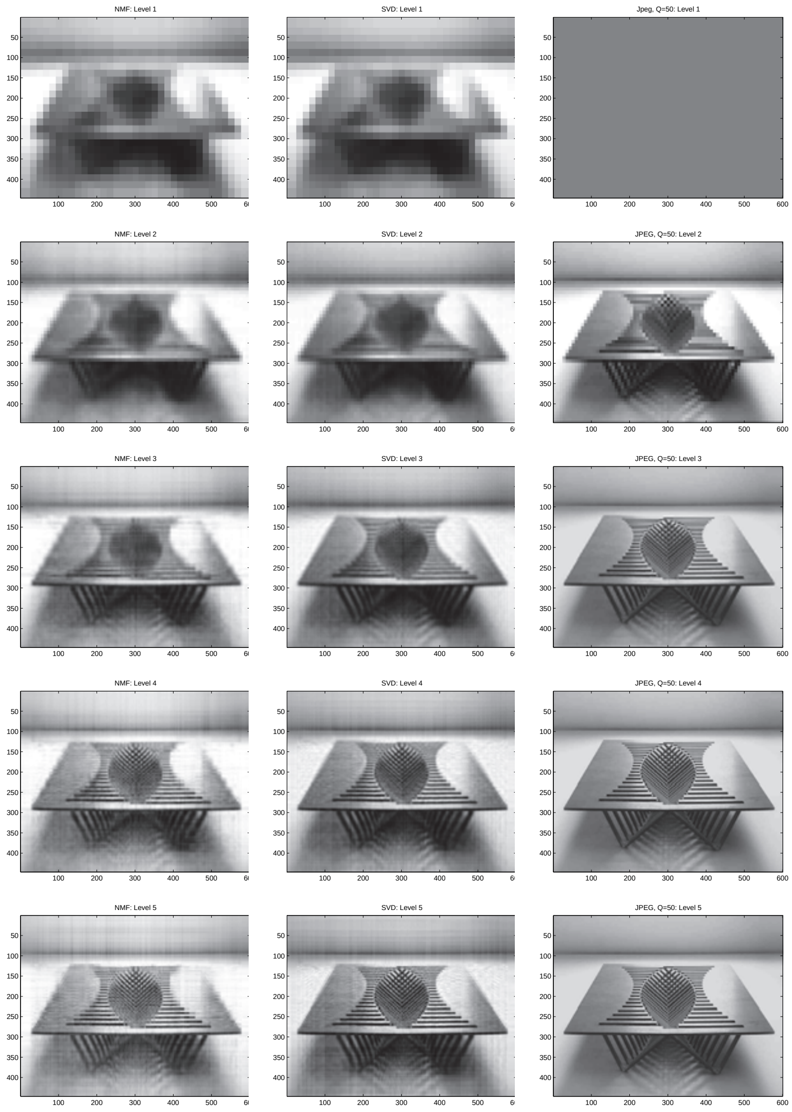


Fig. 4. *MLA for the image in Example* 2 *using NMF without noise.*

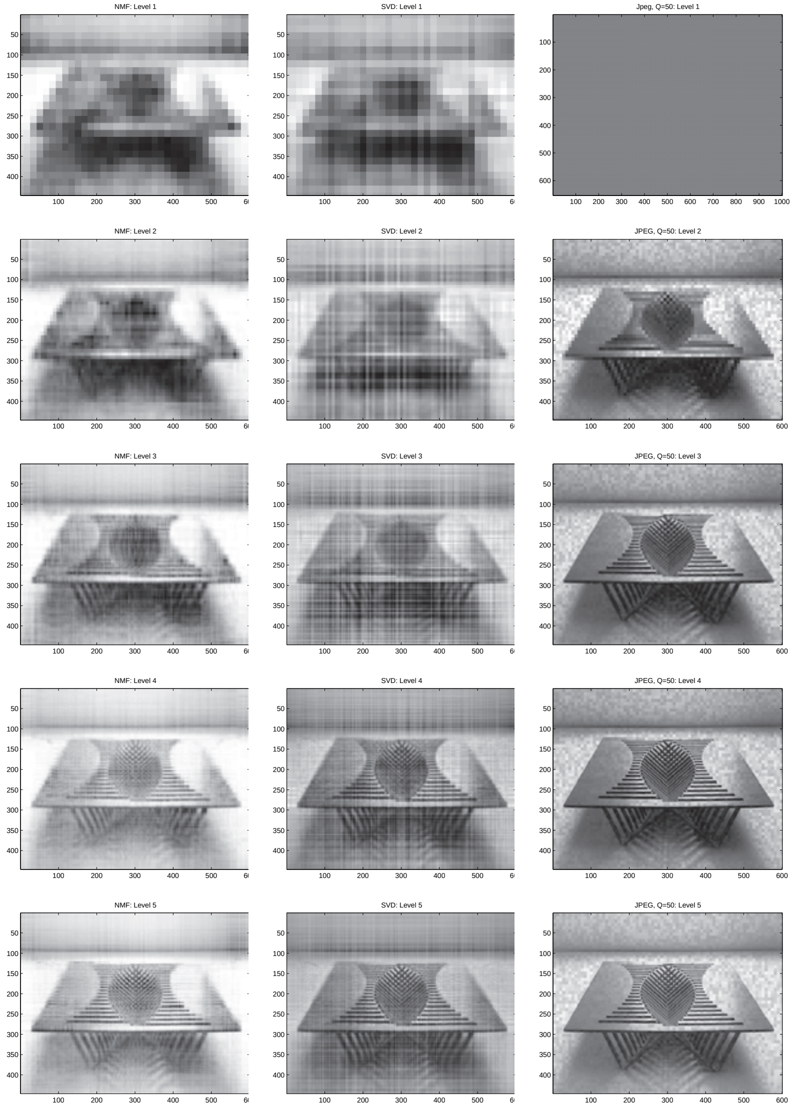


Fig. 5. *MLA for the image in Example* 2 *using NMF with* 25% *noise.*

Table 3

$s_{max} - s$	:	1	2	3	4	5
p	:	24	24	28	34	43
$\tilde{p}$	:	173	74	52	34	73
Memory complexity ratio of NMF	:	0.0033	0.0059	0.0116	0.0283	0.0600
Memory complexity ratio of SVD	:	0.0038	0.0076	0.0177	0.0430	0.1089
Memory complexity ratio of JPEG	:	NA	0.0156	0.0297	0.0609	0.0753
Relative L^2 error in NMF (with 0% noise)	:	0.4766	0.4283	0.3673	0.3109	0.2808
Relative L^2 error in SVD (with 0% noise)	:	0.4763	0.4239	0.3638	0.3133	0.2813
Relative L^2 error in JPEG (with 0% noise)	:	NA	0.3994	0.2753	0.1472	0.1079
Relative L^2 error in NMF (with 25% noise)	:	0.4813	0.4344	0.3769	0.3311	0.3011
Relative L^2 error in SVD (with 25% noise)	:	0.4832	0.4362	0.3791	0.3311	0.3038
Relative L^2 error in JPEG (with 25% noise)	:	NA	0.4221	0.3172	0.2203	0.1967

From Table 3 we can see that, on the same layer, SVD always needs about double the memory of the NMF to just have a similar performance. Again, from Figure 6, we infer that JPEG outperforms the other two methods at the same layer in the absence of noise. Nevertheless, if we choose the same memory complexity ratio, e.g., 1.5%, we can actually get a third layer of the NMF but only a 2nd layer of JPEG, and the relative error of the smaller-sized third layer of NMF is actually smaller than the larger-sized second layer of JPEG. Moreover, as we can see from Figures 6 and 7, when the layers increase and finer details are revealed, a level 4 of NMF is enough to read the Chinese characters, which requires less than 0.03% of memory complexity. With the presence of noise, the relative error of the fourth layer of NMF where the Chinese characters are recognizable becomes comparable with the third layer of JPEG, while their memory complexity is the same. Many of the NMF figures have fewer errors than the SVD figures on the same layers, while the memory complexities of SVD are actually larger. Again, in Figure 7, the SVD and the JPEG images are obviously contaminated, respectively, by straight strips and random squares, whereas the noise contamination in the NMF layers seem less obvious.

5.2. Images reconstructed by direct sampling methods. In this subsection, we shall apply the NMF to the images reconstructed by some recently developed inversion algorithms, namely the direct sampling methods (DSMs). The DSMs are a family of simple and efficient inversion methods which provide a good estimate of the locations of inhomogeneities inside a homogeneous background representing various physical media from a single or a small number of boundary data in both the full and the limited aperture cases. They were developed for inverse acoustic medium scattering in [41, 49, 31] and were later extended to the diffusive optical tomography (DOT) [8], EIT [9], and the electromagnetic inverse scattering problem [32]. In each of these tomographies, a family of probing functions is constructed and an indicator function is defined as a duality product between the observed data and the probing function. The index function, which we shall denote as a general image Y, represents the likelihood of whether a given sampling point sits inside an inhomogeneous inclusion. The DSMs are very inexpensive and robust against noise in the data, and they work with very limited measurement data.

However, from our numerical experiments in the aforementioned references, we notice that, in exchange for their robustness and cost-effectiveness, the images reconstructed by DSMs often contain some artifacts. These artifacts come mainly from the fact that the images are generated by applying a kernel $K(x, y)$ on a function with its support sitting inside the inhomogeneous inclusions, where the kernel $K(x, y)$ results

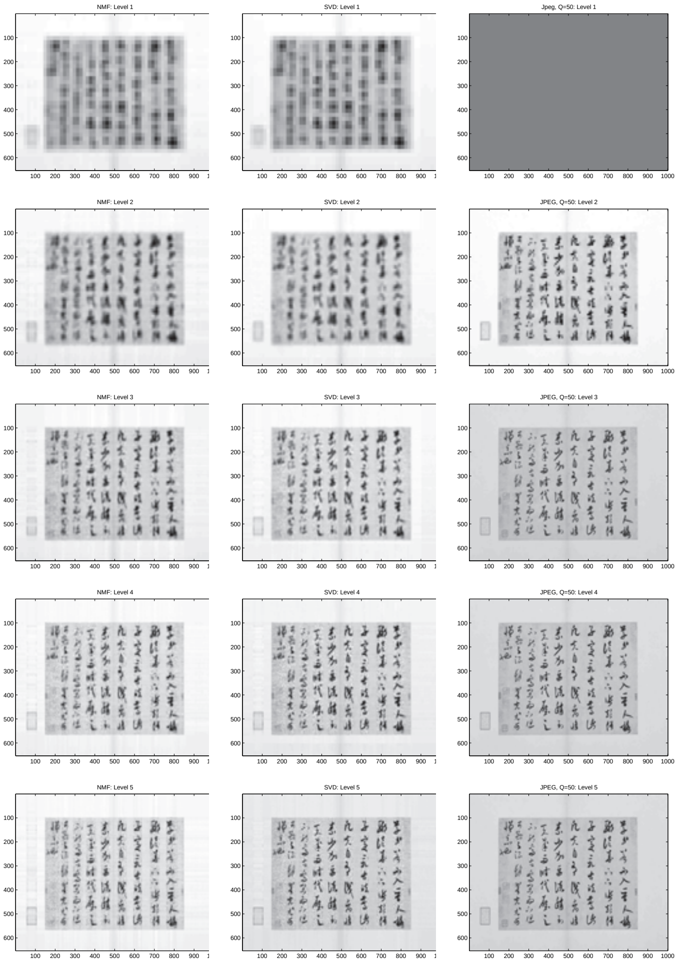


Fig. 6. *MLA for the image in Example* 3 *using NMF without noise.*

from a duality product between the probing function centered at x and the fundamental solution centered at y of the corresponding forward problem. $K(x, y)$ is expected to reach its maximum at $x = y$ and decays quickly when x moves away from y; hence the DSM provides a good estimate of the inclusions. However, we notice in many practical situations that although the kernel $K(x, y)$ attains its maximum at $x = y$, some regions corresponding to the nondiagonal part of the kernel are not negligible, and are quite diffusive, leading to shadows and tails in the DSM images. Moreover, one shall expect the information we obtain from the measurement to provide a sharper image of the inclusions [1, 2, 3]. Therefore we like to reduce the artifacts in the DSM images. From the above discussions we know that a DSM image Y consists of three parts: the first part from the signals of the inhomogeneous inclusions, the second from the contamination of the image by the nondiagonal part of the kernel, and the third part from the noise in the measurement data. Considering the fact that the DSM image and a likelihood function are both positive, it is natural for us to apply the NMF to the DSM images, in the hope of identifying the principal components of the image corresponding to the signal from the inhomogeneous inclusions. But we emphasize that we are not aiming to reconstruct the original DSM image from all

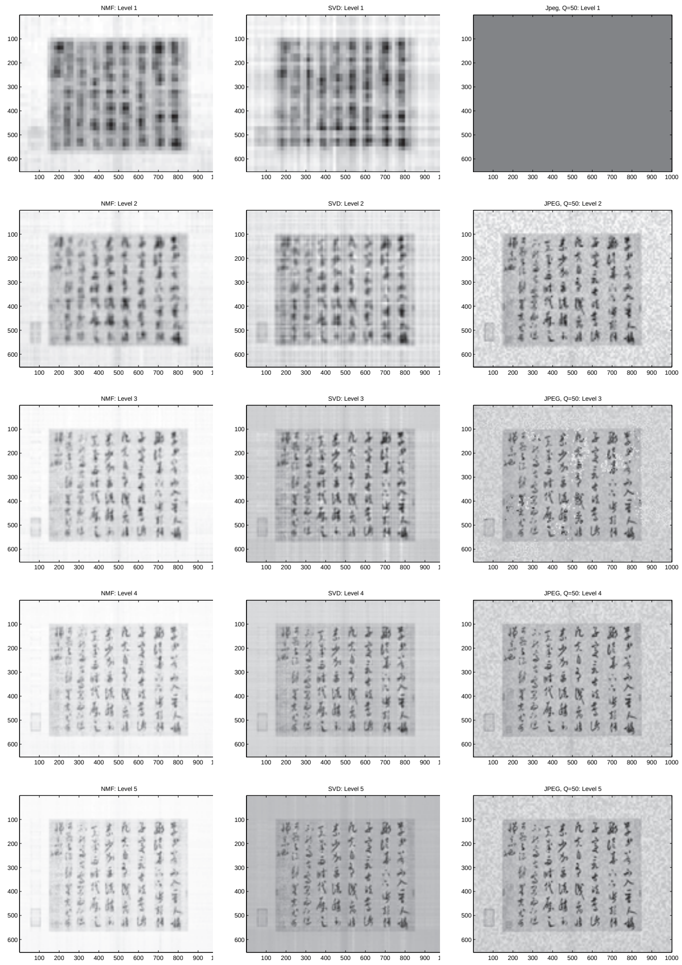
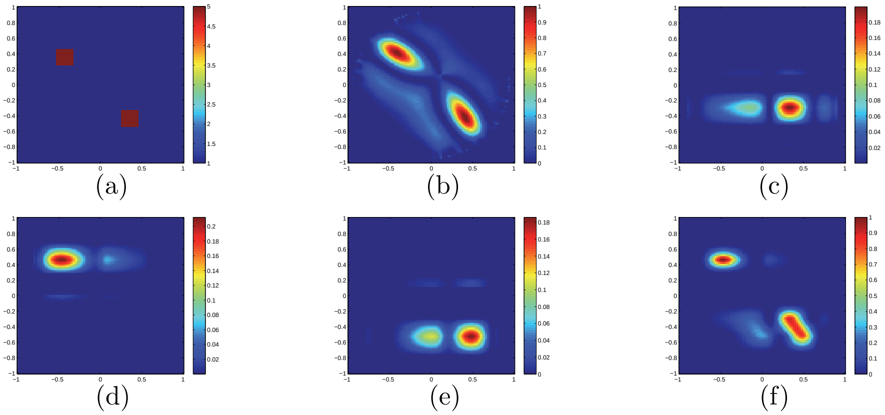


Fig. 7. *MLA for the image in Example* 3 *using NMF with* 25% *noise.*

the components (in terms of tensor products) obtained by NMF, but only to look for principal components of the image containing signals from inhomogeneous inclusions.

Now we shall apply the NMF to the EIT images reconstructed by the DSM. EIT is an effective noninvasive technique to recover the electrical conductivity of an inhomogeneous medium by applying currents at a number of electrodes on the boundary and measuring the corresponding voltages. It has wide applications in many areas, such as oil and geophysical prospection, medical imaging, physiological measurement, early diagnosis of breast cancer, monitoring of pulmonary functions, and detection of leaks from buried pipes [9]. We consider the same numerical setting as in the numerical experiments of EIT for a circular domain using DSM described in [9, section 6]. The physical coefficients of the inhomogeneous inclusions are all set to $\sigma = 5$. The images generated from the scattered potential field using the DSM algorithm are then applied to Algorithm 2 for NMF, with parameters set to $\alpha = 0.2$, $\nu = 0$, $\gamma = 0.02$, $p = 5$, $\tilde{p} = 3$, and $c_1 = c_2 = 1$ in all the examples.

Example 4. We investigate an example with two inclusions of size 0.1×0.1,


(a) (b) (c)

(d) (e) (f)

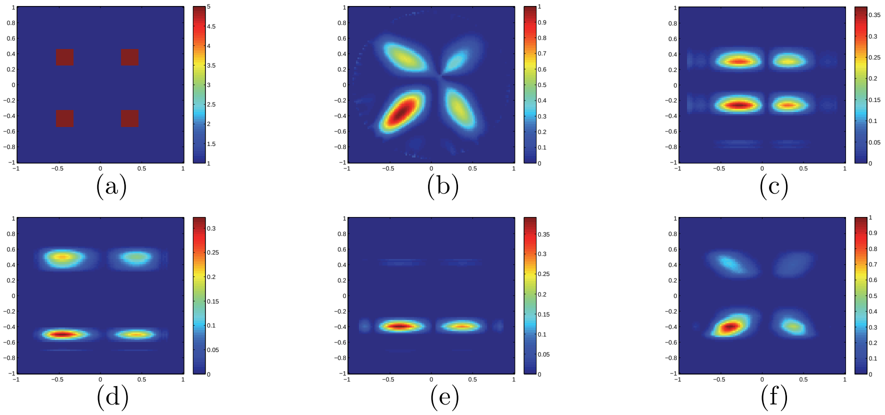
FIG. 8. *NMF decomposition of the DSM images from EIT in Example* 4*, with* $\{\sigma_{i_l j_l}\}_{l=1}^3 = \{2.3712, 2.3548, 2.2904\}$.

respectively, at the positions $(-0.44, 0.36)$ and $(0.36, -0.44)$; see Figure 8(a). The squared reconstructed images from the indices Y after normalization as described in [9] are presented in Figure 8(b). The components $\sigma_{i_l j_l}\,(\tilde{u}_p)_{i_l} \otimes (\tilde{v}_p)_{j_l}$ for $l = 1, 2, 3$ obtained from NMF using Algorithm 2 over the DSM image are shown in Figures 8(c)–(e). The values in the entries of Σ are, respectively, given as $\{\sigma_{i_l j_l}\}_{l=1}^3 = \{2.3712, 2.3548, 2.2904\}$ in this example. The squared image of the approximation to $\mathcal{I}_{p,\tilde{p}}^{\alpha,\nu,\gamma}(Y)$ after normalization is shown in Figure 8(f). The components of inhomogeneous inclusions sitting inside the original medium are decomposed into different components from the NMF.

Example 5. In this example, we consider the case of four inclusions of the same size as in Example 4 sitting inside the sampling region, which are placed at positions of $(0.36, 0.36)$, $(0.36, -0.44)$, $(-0.44, 0.36)$, and $(-0.44, -0.44)$; see Figure 9(a). The squared reconstructed images from the indices Y after normalization are shown in Figure 9(b). Figures 9(c)–(e) present the images of $\sigma_{i_l j_l}\,(\tilde{u}_p)_{i_l} \otimes (\tilde{v}_p)_{j_l}$ for $l = 1, 2, 3$ after NMF over the image Y. The values in the entries of Σ are, respectively, given as $\{\sigma_{i_l j_l}\}_{l=1}^3 = \{5.9647, 4.2460, 3.8970\}$ in this example. The squared image of the approximation to $\mathcal{I}_{p,\tilde{p}}^{\alpha,\nu,\gamma}(Y)$ after normalization is shown in Figure 9(f). We can see that we can obtain fairly nicely the principal components of the image coming from signals from the inclusions.

Example 6. In this example, two inclusions of the same size as in Example 4 are introduced in the homogeneous background, and they are, respectively, placed at the positions $(-0.36, 0.36)$ and $(0.36, 0.36)$ inside the domain; see Figure 10(a). The squared reconstructed images from the indices Y after normalization are given in Figure 10(b). The images of $\sigma_{i_l j_l}\,(\tilde{u}_p)_{i_l} \otimes (\tilde{v}_p)_{j_l}$ for $l = 1, 2, 3$ after NMF over the image Y are shown in Figures 10(c)–(e). The values in the entries of Σ are, respectively, given as $\{\sigma_{i_l j_l}\}_{l=1}^3 = \{3.9194, 0, 0\}$ in this example. Figure 10(f) presents the squared image of the approximation to $\mathcal{I}_{p,\tilde{p}}^{\alpha,\nu,\gamma}(Y)$ after normalization. From the figures, we can see that the principal components coming from the inclusions can be nicely obtained, both the sizes and the locations of inhomogeneities can be reasonably obtained, and the artifacts in the DSM image are effectively removed.

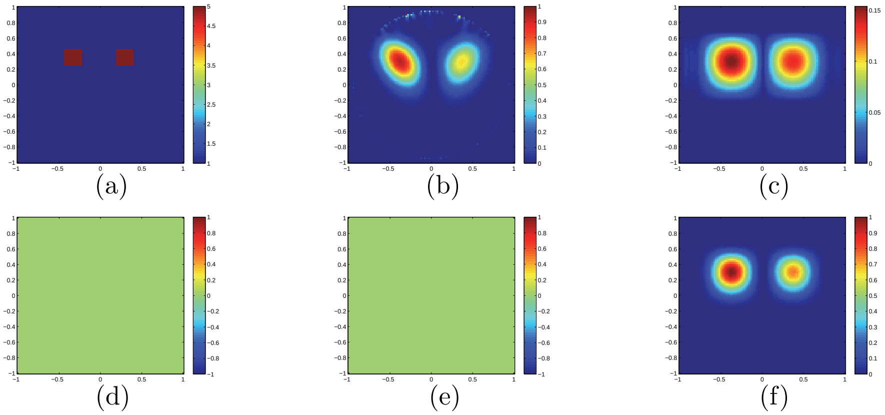
We have also tested the NMF for the DSM images from the DOT [8], and quite



(a) (b) (c)

(d) (e) (f)

FIG. 9. *NMF decomposition of the DSM images from EIT in Example* 5*, with* $\{\sigma_{i_l j_l}\}_{l=1}^3 = \{5.9647, 4.2460, 3.8970\}$.



(a) (b) (c)

(d) (e) (f)

FIG. 10. *NMF decomposition of the DSM images from EIT in Example* 6*, with* $\{\sigma_{i_l j_l}\}_{l=1}^3 = \{3.9194, 0, 0\}$.

similar results are observed.

6. Concluding remarks. We have proposed a special framework of nonnegative matrix trifactorization using l_1 regularization, and studied the probability of its existence and an optimal choice of the dimension in the factorization. The new trifactorization offers a more structural decomposition of positive data and images in terms of tensor products of positive bases. A primal-dual semismooth Newton method has been derived for the nonlinear optimizations involved in the trifactorization. Then we have developed a new multilevel analysis (MLA) framework for the images based on a nonnegative matrix trifactorization, aiming at extracting major components inside an image representing structures of different resolutions and achieving sparse low-rank approximations of different levels with positive bases. The factorization method and the MLA framework have been applied to several imaging and inverse problems. There are, however, several open problems related to nonnegative matrix factorization and its applications to imaging and inverse problems. First, our analysis on the optimal choice of dimension in the nonnegative matrix factorization assumes no prior information on a generative model for an image. But in many practical situations, a generative model of the image is known, and then our asymptotic estimate of the optimal choice of dimension may be improved, and the multiplicative constant in the asymptotic estimate should be more explicitly given in terms of the generative model. Second, it will be interesting to investigate the possibility of combining the data achieved from different levels of our new MLA to resume the true image. Finally, it will be a nice direction to analyze the ill-posed nature of different inverse problems using nonnegative matrix factorization so that one may find substantial improvements in numerical reconstructions.

Acknowledgments. The authors are very grateful to the two anonymous referees for their many insightful and constructive comments and suggestions that have helped us improve the results and presentation of our work substantially.

REFERENCES

[1] H. Ammari, J. Garnier, W. Jing, H. Kang, M. Lim, K. Solna, and H. Wang, *Mathematical and Statistical Methods for Multistatic Imaging*, Lecture Notes in Math. 2098, Springer, Cham, 2013, doi:10.1007/978-3-319-02585-8.

[2] H. Ammari, J. Garnier, H. Kang, M. Lim, and K. Sølna, *Multistatic imaging of extended targets*, SIAM J. Imaging Sci., 5 (2012), pp. 564–600, doi:10.1137/10080631X.

[3] H. Ammari, J. Garnier, and K. Solna, *Resolution and stability analysis in full-aperture, linearized conductivity and wave imaging*, Proc. Amer. Math. Soc., 141 (2013), pp. 3431–3446, doi:10.1090/S0002-9939-2013-11590-X.

[4] J. Bioucas-Dias and J. Nascimento, *Estimation of Signal Subspace on Hyperspectral Data*, Proc. SPIE 5982, SPIE, Bellingham, WA, 2005, doi:10.1117/12.620061.

[5] R. Bro, *Multi-way Analysis in the Food Industry: Models, Algorithms, and Applications* Ph.D. thesis, University of Copenhagen, Copenhagen, Denmark, 1998.

[6] J.-P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov, *Metagenes and molecular pattern discovery using matrix factorization*, Proc. Natl. Acad. Sci. USA, 102 (2004), pp. 4164–4169, doi:10.1073/pnas.0308531101.

[7] E. C. Chi and T. G. Kolda, *On tensors, sparsity, and nonnegative factorizations*, SIAM J. Matrix Anal. Appl., 33 (2012), pp. 1272–1299, doi:10.1137/110859063.

[8] Y. T. Chow, K. Ito, K. Liu, and J. Zou, *Direct sampling method for diffusive optical tomography*, SIAM J. Sci. Comput., 37 (2015), pp. A1658–A1684, doi:10.1137/14097519X.

[9] Y. T. Chow, K. Ito, and J. Zou, *A direct sampling method for electrical impedance tomography*, Inverse Problems, 30 (2014), 095003, doi:10.1088/0266-5611/30/9/095003.

[10] A. Cichocki and A. Phan, *Fast local algorithms for large scale non-negative matrix and tensor factorizations*, IEICE T. Fund. Electr., E92-A (2009), pp. 708–721, doi:10.1587/transfun.E92.A.708.

[11] A. Cichocki, R. Zdunek, and S. I. Amari, *Non-negative matrix factorization with quasi-Newton optimization*, in Artificial Intelligence and Soft Computing, Lecture Notes in Artificial Intelligence 4029, Springer, Berlin, 2006, pp. 870–879, doi:10.1002/9780470747278.

[12] A. Cichocki, R. Zdunek, and S. I. Amari, *Hierarchical ALS algorithms for non-negative matrix and 3D tensor factorization*, in Independent Component Analysis and Signal Separation, Lecture Notes in Comput. Sci. 4666, Springer, Berlin, Heidelberg, 2007, pp. 169–176, doi:10.1007/978-3-540-74494-8_22.

[13] M. Cooper and J. Foote, *Summarizing video using non-negative similarity matrix factorization*, in Proceedings of the IEEE Workshop on Multimedia Signal Processing, 2002, pp. 25–28, doi:10.1109/MMSP.2002.1203239.

[14] I. Daubechies, *Orthonormal bases of compactly supported wavelets*, Comm. Pure Appl. Math., 41 (1988), pp. 909–996, doi:10.1137/0524031.

[15] M. Daube-Witherspoon and G. Muehllehner, *An iterative image space reconstruction algorithm suitable for volume ECT*, IEEE Trans. Med. Imag., 5 (1986), pp. 61–66, doi:10.1109/TMI.1986.4307748.

[16] K. Devarajan, *Non-negative matrix factorization: An analytical and interpretive tool in computational biology*, PLoS Comput. Biol., 4 (2008), e1000029, doi:10.1093/bioinformatics/btp009.

[17] C. Ding and X. He, *K-means clustering via principal component analysis*, in Proceedings of the

International Conference on Machine Learning, 2004, pp. 225–232, doi:10.1145/1015330.1015408.

[18] C. Ding, X. He, and H. D. Simon, *On the equivalence of non-negative matrix factorization and spectral clustering*, in Proceedings of the SIAM Data Mining Conference, 2005, doi:10.1137/1.9781611972757.70.

[19] C. Ding, T. Li, W. Peng, and H. Park, *Orthogonal non-negative matrix tri-factorizations for clustering*, in Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2006, pp. 126–135, doi:10.1145/1150402.1150420.

[20] E. Esser, M. Muller, S. Osher, G. Sapiro, and J. Xin, *A convex model for non-negative matrix factorization and dimensionality reduction on physical space*, IEEE Trans. Image Process., 21 (2012), pp. 3239–3252, doi:10.1109/TIP.2012.2190081.

[21] C. Fevotte, N. Bertin, and J. L. Durrieu, *Non-negative matrix factorization with the Itakura-Saito divergence: With application to music analysis,* Neural Comput., 21(2009), pp. 793–830, doi:10.1162/neco.2008.04-08-771.

[22] N. Gillis, *Non-negative Matrix Factorization: Complexity, Algorithms and Applications*, Ph.D. thesis, Universite Catholique de Louvain, Louvain, Belgium, 2011.

[23] N. Gillis, *The Why and How of Non-negative Matrix Factorization*, preprint, arXiv:1401.5226, 2014.

[24] N. Gillis and F. Glineur, *Accelerated multiplicative updates and hierarchical ALS algorithms for non-negative matrix factorization*, Neural Comput., 24 (2012), pp. 1085–1105, doi:10.1162/NECO_a_00256.

[25] G. Golub and C. Van Loan, *Matrix Computation*, 3rd ed., The Johns Hopkins University Press, Baltimore, MD, 1996.

[26] L. Grippo and M. Sciandrone, *On the convergence of the block nonlinear Gauss-Seidel method under convex constraints*, Oper. Res. Lett., 26 (2000), pp. 127–136, doi:10.1016/S0167-6377(99)00074-7.

[27] N. Guan, D. Tao, Z. Luo, and B. Yuan, *NeNMF: An optimal gradient method for non-negative matrix factorization*, IEEE Trans. Signal Process., 60 (2012), pp. 2882–2898, doi:10.1109/TSP.2012.2190406.

[28] J. Han, L. Han, M. Neumann, and U. Prasad, *On the rate of convergence of the image space reconstruction algorithm*, Oper. Matrices, 3 (2009), pp. 41–58, doi:10.7153/oam-03-02.

[29] N. D. Ho, *Non-negative Matrix Factorization - Algorithms and Applications*, Ph.D. thesis, Universite Catholique de Louvain, Louvain, Belgium, 2008.

[30] P. O. Hoyer, *Non-negative matrix factorization with sparseness constraints*, J. Mach. Learn. Res., 5 (2003/04), pp. 1457–1469.

[31] K. Ito, B. Jin, and J. Zou, *A direct sampling method to an inverse medium scattering problem*, Inverse Problems, 28 (2012), 025003, doi:10.1088/0266-5611/28/2/025003.

[32] K. Ito, B. Jin, and J. Zou, *A direct sampling method for inverse electromagnetic medium scattering*, Inverse Problems, 29 (2013), 095018, doi:10.1088/0266-5611/29/9/095018.

[33] K. Ito, B. Jin, and J. Zou, *A two-stage method for inverse medium scattering*, J. Comput. Phys., 237 (2013), pp. 211–223, doi:10.1016/j.jcp.2012.12.004.

[34] K. Ito and K. Kunisch, *Lagrange Multiplier Approach to Variational Problems and Applications*, SIAM, Philadelphia, 2008, doi:10.1137/1.9780898718614.

[35] B. Kanagal and V. Sindhwani, *Rank selection in low-rank matrix approximations*, in Proceedings of the Twenty-Fourth Annual Conference on Neural Information Processing Systems, 2010, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.185.1337.

[36] H. Kim and H. Park, *Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis*, Bioinformatics, 23 (2007), pp. 1495–1502, doi:10.1093/bioinformatics/btm134.

[37] J. Kim, Y. He, and H. Park, *Algorithms for nonnegative matrix and tensor factorizations: A unified view based on block coordinate descent framework*, J. Global Optim., 58 (2014), pp. 285–319, doi:10.1007/s10898-013-0035-4.

[38] J. Kim and H. Park, *Fast nonnegative matrix factorization: An active-set-like method and comparisons*, SIAM J. Sci. Comput., 33 (2011), pp. 3261–3281, doi:10.1137/110821172.

[39] D. D. Lee and H. S. Seung, *Learning the parts of objects by non-negative matrix factorization*, Nature, 401 (1999), pp. 788–791, doi:10.1038/44565.

[40] D. D. Lee and H. S. Seung, *Algorithms for non-negative matrix factorization*, in Advances in Neural Information Processing Systems, Vol. 13, MIT Press, Cambridge, MA, 2001, pp. 556–562, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.31.7566.

[41] J. Li and J. Zou, *A direct sampling method for inverse scattering using far-field data*, Inverse Probl. Imaging, 7 (2013), pp. 757–775, doi:10.3934/ipi.2013.7.757.

[42] L. Li and Y. J. Zhang, *FastNMF: Highly efficient monotonic fixed-point non-negative matrix*

factorization algorithm with good applicability, J. Electron. Imaging, 18 (2009), 033004, doi:10.1117/1.3184771.

[43] S. Z. Li, X. Hou, H. Zhang, and Q. Cheng, *Learning spatially localized, parts-based representation*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2001, pp. 207–212, doi:10.1109/CVPR.2001.990477.

[44] Y. Li and A. Ngom, *The non-negative matrix factorization toolbox for biological data mining*, BMC Source Code Bio. Med., 8 (2013), pp. 10–25, doi:10.1186/1751-0473-8-10.

[45] C.J. Lin, *Projected gradient methods for non-negative matrix factorization*, Neural Comput., 19 (2007), pp. 2756–2779, doi:10.1162/neco.2007.19.10.2756.

[46] J. Liu, J. Liu, P. Wonka, and J. Ye, *Sparse non-negative tensor factorization using columnwise coordinate descent*, Pattern Recogn., 45 (2012), pp. 649–656, doi:10.1016/j.patcog.2011.05.015.

[47] Y. Meyer, *Ondelettes et functions splines*, Seminaire EDP, Ecole Polytechnique, Paris, France, 1986.

[48] P. Paatero and U. Tapper, *Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values*, Environmetrics, 5 (1994), pp. 111–126, doi:10.1002/env.3170050203.

[49] R. Potthast, *A study on orthogonality sampling*, Inverse Problems, 26 (2010), 074015, doi:10.1088/0266-5611/26/7/074015.

[50] P. Smaragdis, C. Fevotte, G. Mysore, N. Mohammadiha, M. Hoffman, and M., *A unified view of static and dynamic source separation using non-negative factorizations*, IEEE Signal Process. Mag., 31 (2014), pp. 66–75, doi:10.1109/MSP.2013.2297715.

[51] V. Tan and C. Fevotte, *Automatic relevance determination in non-negative matrix factorization*, in Proceedings of the Workshop on Signal Processing with Adaptive Sparse Structured Representations, 2009, doi:10.1109/TPAMI.2012.240.

[52] G. K. Wallace, *The JPEG still picture compression standard,* Comm. ACM, 34 (1991), pp.30–44, doi:10.1109/30.125072.

[53] F. Wang, T. Li, X. Wang, S. Zhu, and C. Ding, *Community discovery using non-negative matrix factorization*, Data Min. Knowl. Disc., 22 (2011), pp. 493–521, doi:10.1007/s10618-010-0181-y.

[54] W. Xu, X. Liu, and Y. Gong, *Document clustering based on non-negative matrix factorization*, in Proceedings of the ACM Conference on Research and Development in IR (SIRGIR), 2003, pp. 267–273, doi:10.1145/860435.860485.