# Non-negative Matrix Factorization for Images with Laplacian Noise

Edmund Y. Lam
Imaging Systems Laboratory
Department of Electrical and Electronic Engineering
The University of Hong Kong
Pokfulam Road, Hong Kong
Email: elam@eee.hku.hk

***Abstract*— This paper is concerned with the design of a non-negative matrix factorization algorithm for image analysis. This can be used in the context of blind source separation, where each observed image is a linear combination of a few basis functions, and that both the coefficients for the linear combination and the bases are unknown. In addition, the observed images are commonly corrupted by noise. While algorithms have been developed when the noise obeys Gaussian or Poisson statistics, here we take it to be Laplacian, which is more representative for other leptokurtic distributions. It is applicable for cases such as transform coefficient distributions and when there are insufficient noise sources for the Central Limit Theorem to apply. We formulate the problem as an $L_1$ minimization and solve it via linear programming.**

## I. Introduction

In many applications, the signals that we acquire are often mixtures of a certain small set of prototype signals, and the problem at hand is to unravel the mixture. For example, a classical problem in biomedical signal processing is to extract the fetal electrocardiogram (ECG) from a mixture of bioelectric signal of the pregnant woman, such as maternal heart activity [1]. Similar problem also exists for images [2]. Consider the human face recognition problem. The goal is to represent each face by a small selection of individual parts (such as eyes, ears, mouth, and nose). The representation, however, would seldom be perfect, and the "error" may be attributed to the imperfection in restricting to a small number of bases, or noise in the data acquisition.

Mathematically we can formulate the above problem, commonly known as blind source separation (BSS), as follows. Assume we have $n$ observed images, each with $m$ pixels, so that by a raster scan we can represent each of them with a length-$m$ vector. We then seek to approximate each one by a different linear combination of $r$ basis vectors. Often, $r \ll n$, so that we are limiting ourselves to a few bases in representing the often large amount of data. In terms of equations, we seek to write

$$\begin{bmatrix} d_{1j} \\ \vdots \\ d_{mj} \end{bmatrix} \approx \sum_{k=1}^{r} w_{kj} \begin{bmatrix} b_{1k} \\ \vdots \\ b_{mk} \end{bmatrix}, \qquad j = 1, \ldots, n, \tag{1}$$

where $[d_{1j} \ldots d_{mj}]^{\mathrm{T}}$ denotes the $j$th observation data, $[b_{1k} \ldots b_{mk}]^{\mathrm{T}}$ denotes the basis vector, and $w_{kj}$ are the weights applied to the $k$th basis vector to form data $j$. In many image analysis problems, the observation and the basis functions should be non-negative, as are the weights concerned [3]. Thus, we require that $d_{ij} \geq 0$, $b_{ir} \geq 0$, and $w_{rj} \geq 0$ for $i = 1, \ldots, m$, $j = 1, \ldots, n$, and $k = 1, \ldots, r$. Note that we can compactly write Equation (1) using matrix notations as

$$\mathsf{D} = \mathsf{BW} + \mathsf{N}, \tag{2}$$

where $\mathsf{D}$, $\mathsf{B}$, $\mathsf{W}$, $\mathsf{N}$ are $m \times n$, $m \times r$, $r \times n$, and $m \times n$ matrices respectively. $\mathsf{N}$ is used to incorporate the error between the observed data $\mathsf{D}$ and the weighted combination of the bases $\mathsf{BW}$. The non-negativity constraints can also be written as $\mathsf{D} \succeq 0$, $\mathsf{B} \succeq 0$, and $\mathsf{W} \succeq 0$, using curled inequality sign to denote componentwise inequality. The BSS problem formulated this way is called non-negative matrix factorization of the observation $\mathsf{D}$ [3].

It is common to model $\mathsf{N}$ as a multivariate Gaussian distribution. In that case, one can show that the non-negative matrix factorization problem can be tackled with a minimization of a cost function $C_{\text{Gaussian}}$, where

$$C_{\text{Gaussian}} = \|\mathsf{D} - \mathsf{BW}\|_F^2. \tag{3}$$

$\|\cdot\|_F$ is used to denote the Frobenius norm [4], i.e.

$$\|\mathsf{D}\|_F = \left( \sum_{i=1}^{m} \sum_{j=1}^{n} |d_{ij}|^2 \right)^{\frac{1}{2}}. \tag{4}$$

On the other hand, additive non-Gaussian noise also arises in many important scenarios, such as image transform coefficient distributions [5] and when there is an insufficient number of identical noise sources for the Central Limit Theorem to apply [6]. In these instances, Laplacian distribution is a more appropriate model, given its leptokurtic nature. Thus in this paper, we develop a non-negative matrix factorization algorithm where the noise statistics is assumed to follow the Laplacian statistics. The derivation is given in Section II. Simulation results are given in Section III, with some concluding remarks in Section IV.

978-1-4244-2342-2/08/$25.00 ©2008 IEEE. 

## II. Non-negative Matrix Factorization Algorithm

We tackle Equation (2) using a maximum likelihood approach [7]. Since the probability distribution of D is the same as that of N, we can maximize the likelihood D by choosing the various possible parameters B and W. Therefore, we have

$$(\mathsf{B}_{\text{ML}}, \mathsf{W}_{\text{ML}}) = \underset{\mathsf{B}\succeq 0, \mathsf{W}\succeq 0}{\arg\max}\, \mathcal{P}(\mathsf{D}|\mathsf{B},\mathsf{W}), \tag{5}$$

where $\mathcal{P}$ denotes probability. Provided $\mathcal{P} \neq 0$, using the fact that logarithm is a monotonic transformation, we can rewrite the equation above as

$$(\mathsf{B}_{\text{ML}}, \mathsf{W}_{\text{ML}}) = \underset{\mathsf{B}\succeq 0, \mathsf{W}\succeq 0}{\arg\max}\, \log \mathcal{P}(\mathsf{D}|\mathsf{B},\mathsf{W}). \tag{6}$$

Substituting with the Laplacian distribution equation, we have

$$\mathcal{P}([\mathsf{D}]_{ij}|\mathsf{B},\mathsf{W}) = \frac{\lambda}{2} \mathrm{e}^{-\lambda|[\mathsf{D}-\mathsf{BW}]_{ij}|}, \tag{7}$$

where $\lambda$ is the parameter of the Laplacian distribution [8]. Therefore,

$$\begin{aligned} \log \mathcal{P}(\mathsf{D}|\mathsf{B},\mathsf{W}) &= \log \prod_{i=1}^{m} \prod_{j=1}^{n} \mathcal{P}([\mathsf{D}]_{ij}|\mathsf{B},\mathsf{W}) & (8)\\ &= \sum_{i=1}^{m} \sum_{j=1}^{n} \log \mathcal{P}([\mathsf{D}]_{ij}|\mathsf{B},\mathsf{W}) & (9)\\ &= mn \log\left(\frac{\lambda}{2}\right) - \sum_{i=1}^{m}\sum_{j=1}^{n} \lambda|[\mathsf{D}-\mathsf{BW}]_{ij}|. & (10) \end{aligned}$$

The first term above is a constant. Thus, to maximize the likelihood, we can tackle the following optimization problem:

$$\begin{array}{ll} \text{minimize} & \|\mathsf{D}-\mathsf{BW}\|_1 \\ \text{subject to} & \mathsf{B} \succeq 0 \\ & \mathsf{W} \succeq 0 \end{array}, \tag{11}$$

where we use the $L_1$ norm on the matrix as if it has been put in lexicographic ordering, i.e.

$$\|\mathsf{D}\|_1 = \sum_{i=1}^{m}\sum_{j=1}^{n} |d_{ij}|. \tag{12}$$

Unfortunately, the optimization problem in Equation (11) is not jointly convex in B and W, making it very difficult to find a global optimal solution. Instead, as with other algorithms in non-negative matrix factorization, we use an iterative scheme to find a local minimum instead. This is achieved by alternating minimization, where at each iteration either B or W is taken as fixed and the other as variable for the minimization. The resulting problem is then convex and can be solved efficiently [9]. For $L_1$ norm minimization, we can even cast it as linear programming, as shown below.

We first assume B is fixed and we seek to minimize the objective function in Equation (11) taking W as the variable. Because of symmetry between B and W, holding W fixed and minimizing with respect to B will be analogous. If we use $d_{*j}$ and $w_{*j}$ to denote the $j$th column of D and W respectively, Equation (11) can be decomposed into optimization problems:

$$\begin{array}{ll} \text{minimize} & \|d_{*j} - \mathsf{B}w_{*j}\|_1 \\ \text{subject to} & w_{*j} \succeq 0 \end{array} \tag{13}$$

for every column of D and W. To convert it to a linear program, we introduce an auxiliary variable $t$ [10], which is a vector of length $m$. The above problem is equivalent to:

$$\begin{array}{ll} \text{minimize} & \|t\|_1 \\ \text{subject to} & d_{*j} - \mathsf{B}w_{*j} \preceq t \\ & d_{*j} - \mathsf{B}w_{*j} \succeq -t \\ & w_{*j} \succeq 0 \end{array} \tag{14}$$

and, in turn, using $s = \begin{bmatrix} w_{*j} \\ t \end{bmatrix}$, can be recast as:

$$\begin{array}{ll} \text{minimize} & [0 \;\; 1]s \\ \text{subject to} & \begin{bmatrix} -\mathsf{B} & -\mathsf{I} \end{bmatrix} s \preceq -d_{*j} \\ & \begin{bmatrix} \mathsf{B} & -\mathsf{I} \end{bmatrix} s \preceq d_{*j} \\ & \begin{bmatrix} -\mathsf{I} & 0 \end{bmatrix} s \preceq 0 \end{array} \tag{15}$$

which is a standard linear program in $s$. The solution $w_{*j}$ can be extracted as the first part of this vector.

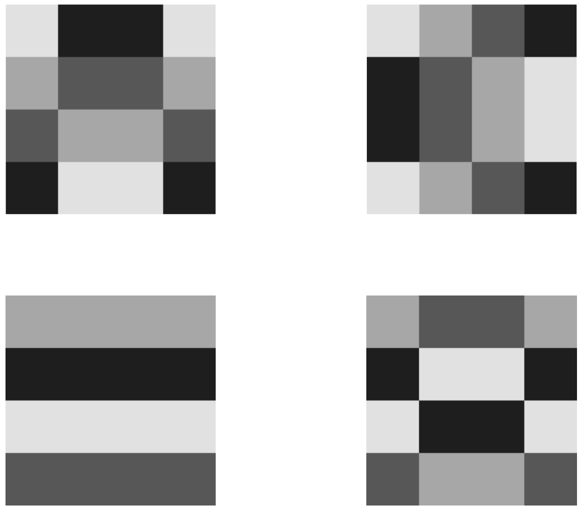

Fig. 1. The four basis images used in simulation.

## III. Simulation

In this section, we experiment the linear programming-based non-negative matrix factorization algorithm above with simple images to test the efficacy of the approach. Four basis images are generated as shown in Figure 1. They are randomly selected among the sixteen $4 \times 4$ discrete cosine transform (DCT) basis images, given by the equation [11]

$$I_{u,v}(x,y) = \alpha(u)\alpha(v)\cos\left(\frac{(2x+1)\pi u}{8}\right)\cos\left(\frac{(2y+1)\pi v}{8}\right) \tag{16}$$

where $0 \leq x, y \leq 3$ are the pixels, for values of $u$ and $v$ between 0 and 3 inclusive. To preserve the unitary nature of

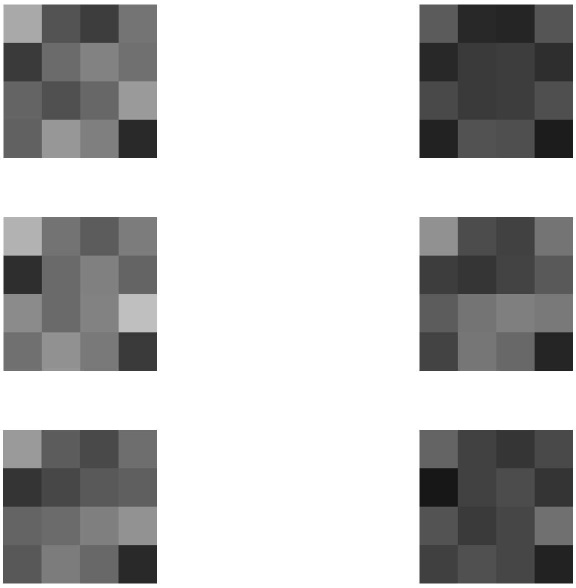

Fig. 2. Six out of the 36 observed images used in simulation.

DCT, we take

$$\alpha(\mu) = \begin{cases} \sqrt{\frac{1}{n}} & : \quad \mu = 0 \\ \sqrt{\frac{2}{n}} & : \quad \text{otherwise.} \end{cases} \tag{17}$$

We then generate 36 images that are linear combinations of these basis images, with the coefficients for each image being uniformly distributed between 0 and 0.25, and noise is then added to generate the observed images. Six of these observed images are given in Figure 2.

We run the non-negative matrix factorization algorithm above with no knowledge of the basis images and mixing coefficients, and track the cost function $\mathcal{C}_{\text{Laplacian}} = \|\mathsf{D} - \mathsf{BW}\|_1$. This is given in Figure 3. Note that we do not expect it to decrease to zero because of the existence of noise. As can be seen, the cost decreases rapidly in the initial iterations, but much slower soon afterwards. This trend is similar to several other non-negative matrix factorization schemes, such as in [12].

We can also observe the estimated basis images given in Figure 4. While these images are not identical to those in Figure 1, which would be very difficult to achieve given the existence of noise, they nevertheless bear certain similarity, such as the location of bright and dark pixels.

## IV. Conclusions

In this paper we propose a non-negative matrix factorization algorithm for blind source separation of images, using linear programming as the computational core. Preliminary simulation results show that this approach is viable to recover characteristics of the basis images. As with the case for Gaussian noise, further restrictions such as sparsity can be added to the bases and mixing coefficients [13], [14]. We expect that such additional prior knowledge can help to enhance the quality of the factorization.

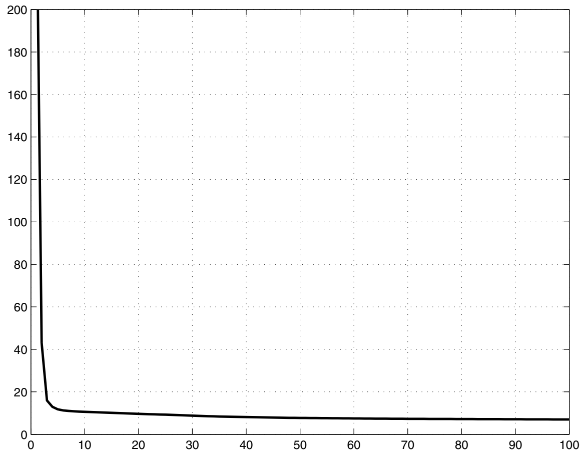


Fig. 3. The value of the cost function during each iteration.

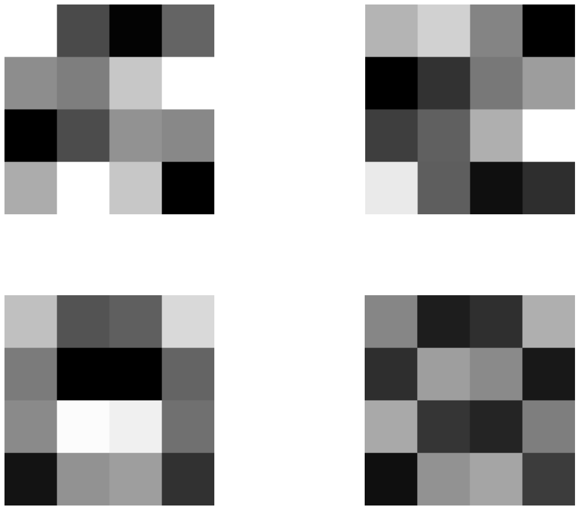

Fig. 4. The estimated image bases generated from non-negative matrix factorization of the observed images.

## References


[1] L. De Lathauwer, B. De Moor, and J. Vandewalle, "Fetal electrocardiogram extraction by blind source subspace separation," *IEEE Transactions on Biomedical Engineering*, vol. 47, no. 5, pp. 567–572, May 2000.

[2] A. Cichocki and S. Amari, *Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications*. Wiley, 2002.

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

[4] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Johns Hopkins University Press, 1996.

[5] E. Y. Lam and J. W. Goodman, "A mathematical analysis of the DCT coefficient distributions for images," *IEEE Transactions on Image Processing*, vol. 9, no. 10, pp. 1661–1666, October 2000.

[6] D. W. Stein, "Detection of random signals in gaussian mixture noise," *IEEE Transactions on Information Theory*, vol. 41, no. 6, pp. 1788–1801, November 1995.

[7] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data," *Journal of the Royal Statistical Society Series B*, vol. 39, pp. 1–38, 1977.

[8] A. Leon-Garcia, *Probability and Random Processess for Electrical Engineering*, 2nd ed. Reading, Massachusetts: Addison Wesley, 1994.

[9] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge: Cambridge University Press, 2004.

[10] E. Y. Lam, "Robust minimization of lighting variation for real-time defect detection," *Real-Time Imaging*, vol. 10, no. 6, pp. 365–370, December 2004.
[11] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed. New Jersey: Prentice Hall, 2008.
[12] P. Sajda, S. Du, and L. C. Parra, "Recovery of constituent spectra using non-negative matrix factorization," in *Wavelets: Applications in Signal and Image Processing X*, ser. Proceedings of the SPIE, vol. 5207, November 2003, pp. 321–331.
[13] P. O. Hoyer, "Non-negative sparse coding," in *12th IEEE Workshop on Neural Networks for Signal Processing*, September 2002, pp. 557–565.
[14] ——, "Non-negative matrix factorization with sparseness constraints," *Journal of Machine Learning Research*, vol. 5, pp. 1457–1469, December 2004.