Convolutive Non-Negative Sparse Coding

Wenwu Wang

Abstract— **Non-negative sparse coding (NSC) is a powerful technique for low-rank data approximation, and has found several successful applications in signal processing. However, the temporal dependency, which is a vital clue for many realistic signals, has not been taken into account in its conventional model. In this paper, we propose a general framework, i.e., convolutive non-negative sparse coding (CNSC), by considering a convolutive model for the low-rank approximation of the original data. Using this model, we have developed an effective learning algorithm based on the multiplicative adaptation of the reconstruction error function defined by the squared Euclidean distance. The proposed algorithm is applied to the separation of music audio objects in the magnitude spectrum domain. Interesting numerical results are provided to demonstrate its advantages over both the conventional NSC and an existing convolutive coding method.**

I. INTRODUCTION

NON-NEGATIVE sparse coding (NSC) is an emerging technique for representing multivariate data with a low rank approximation. It essentially originates from linear sparse coding and non-negative matrix factorization (NMF) [1] [2] [3]. With the non-negativity constraint, NSC gives "parts" based representation as only additive combinations of basis are allowed in the representation [2]. Due to the enforcement of additional sparseness constraint, the original data is encoded with only a small number of *active* components [3]. These promising properties have made NSC an attractive signal representation method that is potentially very useful for a number of applications in signal and image processing [2] [8] [9] [6] [7].

Although NSC is shown to be powerful in image coding, e.g., [3] [5] [4], it is not as promising for audio signal processing problems [8] [9] [6]. A major reason is that no connections between the neighboring columns of the data matrix have been considered in its fundamental model. This essentially implies that the temporal dependency, a vital clue for audio signals, has been ignored in the model.

In this paper, we extend NSC to a more general framework, i.e., convolutive non-negative sparse coding (CNSC), using a convolutive data model. Due to this new formulation, the temporal dependency within many realistic signals can be successfully captured. We further develop an effective learning algorithm for the minimization of the new cost function based on the squared Euclidean distance. By applying the proposed algorithm to spectrogram separation of audio objects (repeating musical notes) with time-varying frequency patterns, we demonstrate its advantage over the standard NSC, as well as a convolutive coding algorithm that has been developed recently.

Wenwu Wang is with the Centre for Vision, Speech and Signal Processing, Department of Electronic Engineering, University of Surrey, Guildford, United Kingdom (phone: 44-1483-686039; fax: 44-1483-686031; e-mail: w.wang@surrey.ac.uk).

The remainder of the paper is organized as follows. The next section briefly reviews the standard NSC. The proposed CNSC framework and algorithm are detailed in section III. Section IV demonstrates its performance using numerical examples and section V concludes the paper.

II. NSC

Given an $M \times N$ non-negative matrix $\mathbf{X} \in \mathbb{R}_+^{M \times N}$, the goal of NSC is to encode $\mathbf{X}$ as a product of matrices $\mathbf{W} \in \mathbb{R}_+^{M \times R}$ and $\mathbf{H} \in \mathbb{R}_+^{R \times N}$ that are both nonnegative and sparse, where R is the rank of the factorization, typically smaller than M (or N). This can be achieved by the minimization of the following commonly used criterion [3] [5],

$$\begin{aligned}(\hat{\mathbf{W}}, \hat{\mathbf{H}}) &= \arg\min_{\mathbf{W},\mathbf{H}} \mathcal{L}(\mathbf{X}||\hat{\mathbf{X}}) \\ &= \arg\min_{\mathbf{W},\mathbf{H}} \left\|\mathbf{X} - \hat{\mathbf{X}}\right\|_F^2 + \lambda \sum_{ij} \mathbf{H}_{ij} \end{aligned} \quad (1)$$

where $\hat{\mathbf{W}}$ and $\hat{\mathbf{H}}$ are the estimated optimal values of $\mathbf{W}$ and $\mathbf{H}$, $\|\cdot\|_F$ denotes the Frobenius norm, $\mathbf{H}_{ij}$ is the ij-th elements of $\mathbf{H}$, the positive constant λ controls the sparsity being constrained on the criterion, and $\hat{\mathbf{X}}$ is estimated by

$$\hat{\mathbf{X}} = \mathbf{W}\mathbf{H} \quad (2)$$

In (1), we have focused on the error function based on the squared Euclidean distance. Apparently, other statistical measurements, such as the extended KL divergence used in [9] [6], can also be used to measure the accuracy of the coding algorithm.

However, unlike the standard NMF, there is a scaling problem with (1). If $\mathbf{W}$ is scaled up to $\alpha\mathbf{W}$ and $\mathbf{H}$ scaled down to $\mathbf{H}/\alpha$, where $\alpha > 1$, then although the value of the first term in (1) does not change, the second term is decreased [3]. As a result, the minimization of (1) leads to an unboundedly growing $\mathbf{W}$ while $\mathbf{H}$ approaches to zero, so that the solution found actually does not depend on the sparsity term any more. To prevent this, an additional normalization step has to be taken to restrict the variance of either $\mathbf{W}$ or $\mathbf{H}$ [3] [4]. If we adopt the same procedures as in [2], and further enforce the unit norm constraint on the columns of $\mathbf{W}$, we can develop a multiplicative algorithm for minimizing criterion (1). In compact forms, the updating rules for $\mathbf{H}$ and $\mathbf{W}$ can be derived as,

$$\mathbf{H}^{q+1} = \mathbf{H}^q \odot ((\mathbf{W}^q)^T\mathbf{X}) \oslash ((\mathbf{W}^q)^T\mathbf{W}^q\mathbf{H}^q + \lambda\Xi) \quad (3)$$

$$\mathbf{W}^{q+1} = \mathbf{W}^q \odot (\mathbf{X}(\mathbf{H}^{q+1})^T) \oslash (\mathbf{W}^q\mathbf{H}^{q+1}(\mathbf{H}^{q+1})^T) \quad (4)$$

978-1-4244-1821-3/08/$25.00©2008 IEEE

where $\odot$ and $\oslash$ denote the Hadamard (element-wise) product and division respectively, q is the iteration index, $(\cdot)^T$ is the matrix transpose operator, Ξ is a matrix whose elements are all unity, and the columns of $\mathbf{W}$ should be normalized to have constant variance at each iteration.

III. CNSC

To consider the temporal dependency in the original data matrix $\mathbf{X}$, we extend the NSC using a convolutive model for $\mathbf{X}$, as defined in [9]. Effectively, we replace $\hat{\mathbf{X}}$ in (2) by a sum of shifted matrix products, i.e.

$$\hat{\mathbf{X}} = \sum_{p=0}^{P-1} \mathbf{W}(p) \overrightarrow{\mathbf{H}}^{p} \tag{5}$$

where $\mathbf{W}(p) \in \mathbb{R}_+^{M\times R}$, $p = 0, \cdots, P-1$, are a set of bases, $\mathbf{H} \in \mathbb{R}_+^{R\times N}$ is a weighting matrix, and $\overrightarrow{\mathbf{H}}^{p}$ shifts the columns of $\mathbf{H}$ by p spots to the right, with the columns shifted in from outside the matrix set to zero. Analogously, $\overleftarrow{\mathbf{H}}^{p}$ shifts the columns of $\mathbf{H}$ by p spots to the left. These notations will also be used for the shifting operations of other matrices throughout the paper. Note that, $\overrightarrow{\mathbf{H}}^{0} = \overleftarrow{\mathbf{H}}^{0} = \mathbf{H}$.

As compared with (2), $\hat{\mathbf{X}}$ in (5) is a linear combination of the columns of $\mathbf{H}$. Therefore, the gradient of $\mathcal{L}(\mathbf{X}||\hat{\mathbf{X}})$ with respect to $\mathbf{H}$, can be derived as a collection of a group of standard NSC problems, i.e.,

$$\frac{\partial \mathcal{L}(\mathbf{X}||\hat{\mathbf{X}})}{\partial \mathbf{H}} = (\mathbf{W}(p))^T \overleftarrow{\hat{\mathbf{X}}}^{p} - (\mathbf{W}(p))^T \overleftarrow{\mathbf{X}}^{p} + \lambda\Xi. \tag{6}$$

Set the element-related step size to

$$\mu_{\mathbf{H}} = \mathbf{H} \oslash ((\mathbf{W}(p))^T \overleftarrow{\hat{\mathbf{X}}}^{p} + \lambda\boldsymbol{\Xi}) \tag{7}$$

Substituting (6) and (7) into the following equation,

$$\mathbf{H} = \mathbf{H} - \mu_{\mathbf{H}} \odot \frac{\partial \mathcal{L}(\mathbf{X}||\hat{\mathbf{X}})}{\partial \mathbf{H}}, \tag{8}$$

we can easily derive the following multiplicative update equation (marked with the iteration number),

$$\mathbf{H}^{q+1} = \mathbf{H}^{q} \odot (((\mathbf{W}^{q+1}(p))^T \overleftarrow{\mathbf{X}}^{p}) \oslash ((\mathbf{W}^{q+1}(p))^T \overleftarrow{\hat{\mathbf{X}}^{q}}^{p} + \lambda\boldsymbol{\Xi}) \tag{9}$$

Note that the operator $\odot$ in (8) denotes an element-wise step-size adaptation.

Similarly, we can derive the update equation for $\mathbf{W}(p)$ by fixing $\mathbf{H}$.

$$\mathbf{W}^{q+1}(p) = \mathbf{W}^{q}(p) \odot (\mathbf{X}(\overrightarrow{\mathbf{H}}^{p})^T) \oslash (\hat{\mathbf{X}}^{q}(\overrightarrow{\mathbf{H}}^{p})^T)) \tag{10}$$

where $p = 0, \cdots, P-1$. Note, that the above equations may lead to a biased estimate of $\mathbf{H}$, as all $\mathbf{W}(p)$ share the same $\mathbf{H}$. In order to mitigate this effect, we can update all $\mathbf{W}(p)$ first, and then take the average of all the updates for $\mathbf{H}$, that is

$$\mathbf{H}^{q+1} = \frac{1}{P} \sum_{p=0}^{P-1} \mathbf{H}^{q}(p) \tag{11}$$

where $\mathbf{H}^{q}(p)$ is given by

$$\mathbf{H}^{q}(p) = \mathbf{H}^{q} \odot (((\mathbf{W}^{q+1}(p))^T \overleftarrow{\mathbf{X}}^{p}) \oslash ((\mathbf{W}^{q+1}(p))^T \overleftarrow{\hat{\mathbf{X}}^{q}}^{p} + \lambda\boldsymbol{\Xi}) \tag{12}$$

Instead of computing the loop in (5), the update of $\hat{\mathbf{X}}^{q}$ in (9) and (10) can be implemented efficiently using

$$\hat{\mathbf{X}}^{q} = \hat{\mathbf{X}}^{q} - \mathbf{W}^{q}(p) \overrightarrow{\mathbf{H}^{q}}^{p} + \mathbf{W}^{q+1}(p) \overrightarrow{\mathbf{H}^{q}}^{p} \quad (p = 0, \cdots, P-1) \tag{13}$$

where $\hat{\mathbf{X}}^{q}$ is updated recursively to accommodate the new values of each $\mathbf{W}(p)$ (inside the P loops), and the initial value of $\hat{\mathbf{X}}^{q}$ ($q > 1$) in the right hand side (RHS) of equation (13) is obtained at the end of $(q-1)$-th iteration (outside the P loops), when the recursions are completed. For $q = 1$, $\hat{\mathbf{X}}^{q}$ in the RHS of equation (13) is still calculated via equation (5). In practice, we found that the non-negative property of $\hat{\mathbf{X}}^{q}$ may not be guaranteed, due to the subtraction operation and small numerical errors. The small negative values can be prevented by using the projection operation:

$$\hat{\mathbf{X}}^{q} = \max(\epsilon, \hat{\mathbf{X}}^{q}) \tag{14}$$

where $\max(\cdot)$ takes the maximum value of its arguments, and ϵ is a trivial constant, typically, $\epsilon = 10^{-9}$ in our implementation. The algorithm stops iterations when the following criterion is satisfied,

$$\frac{\left\| \hat{\mathbf{X}}^{q+1} - \hat{\mathbf{X}}^{q} \right\|_F}{\left\| \hat{\mathbf{X}}^{q} \right\|_F} < \zeta \tag{15}$$

where ζ is a small constant. At the end of each iteration, we normalize $\mathbf{W}(p)$ such that the the unbounded scaling problem described in the above section can be approximately controlled.

In summary, the adaptation of equations (10), (13), (14), (12), (11) and (15) in order represents our proposed algorithm. We denote $\mathbf{H}$ as $\mathbf{H}^o$, and $\mathbf{W}(p)$ as $\mathbf{W}^o(p)$ when the algorithm satisfies the stopping criterion (15). If $P = 1$, it basically reduces to the NSC algorithm represented by (3) and (4). If $P > 1$, the computational load of the proposed algorithm is approximately P times that of the NSC.

IV. Numerical Experiments

Two music audio signals with each containing repeating musical notes G4 and A3 played by a guitar are mixed together. The mixed signal is approximately 6.8s sampled at $f_s = 22050$Hz. The factorization rank R is set to 2, i.e., exactly the same as the total number of the signals in the mixture. The matrices $\mathbf{W}(p)$ and $\mathbf{H}$ are initialized as the absolute values of random matrices. P is set[1] to 105. All tests were running on a computer whose CPU speed is 1.8GHz. Spectrogram matrix $\mathbf{X}$ is generated using T-point windowed DFT [10], and is visualized in Figure 1, where T has been set to 2048 samples. The resulted $\mathbf{H}^o$ and $\mathbf{W}^o(p)$ by applying the proposed CNSC are plotted in Figure 2 and 3 respectively,

[1]In order for the object to be separated, P should be big enough to cover the length of the object in the audio signal.

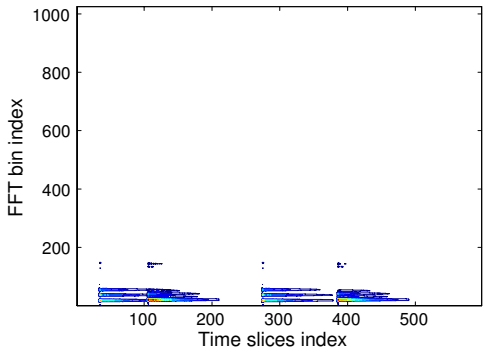

Fig. 1. The contour plot of the magnitude spectrum matrix $\mathbf{X}$ of the real music audio signal.

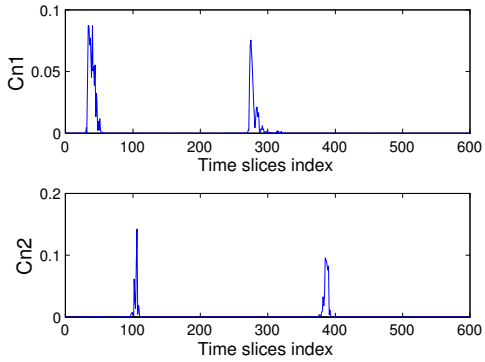

Fig. 2. Visualization of the factorized $\mathbf{H}^o$. "Cn1" and "Cn2" denote the first and second rows of $\mathbf{H}^o$ respectively.

for which $\lambda = 0.6$. From these two figures, we see that the audio objects have been successfully separated with $\mathbf{W}^o(p)$ being the time-frequency representation of the repeating patterns, and $\mathbf{H}^o$ containing the temporal structure of these patterns, i.e., the happening time of individual patterns. The standard NSC described by the learning rules (3) and (4), however, totally fails for separating the audio objects in these tests since the single basis learned by NSC is not sufficient for covering the temporal features of the audio objects. We have extensively tested the algorithm for different set-ups of the parameters, including other randomly initialized matrices $\mathbf{W}$ and $\mathbf{H}$, and found such similar separation performance.

To evaluate the performance more accurately, we use two performance indices. One is the rejection ratio (RR). If we denote $\hat{\mathbf{X}} = \sum_{i=1}^{R} \hat{\mathbf{X}}(i)$, i.e., splitting $\hat{\mathbf{X}}$ into R factorized components, we can define RR as

$$RR(\text{dB}) = 10\log_{10}\left[\sum_{\forall j \neq i} cor(\hat{\mathbf{X}}(i), \hat{\mathbf{X}}(j))\right] \qquad (16)$$

where cor denotes the correlation. This index can measure

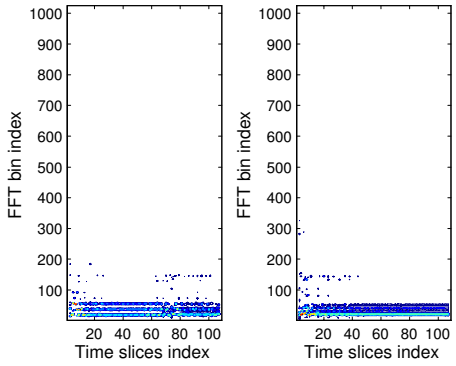

Fig. 3. Visualization of the factorized $\mathbf{W}^o(p)$, $p = 0, ..., 104$. The left plot represents note G4, and the right denotes note A3.

how accurate the separation performance is, and a lower value represents a better performance. The other index is the relative estimation error (REE) defined as

$$REE(\text{dB}) = 10\log_{10}\left(\left\|\mathbf{X} - \hat{\mathbf{X}}\right\|_F / \|\mathbf{X}\|_F\right) \qquad (17)$$

which measures the accuracy of the factorization, a lower value representing a better performance.

We compare the proposed CNSC algorithm with another convolutive coding algorithm in [11] (without sparseness constraint). We run both algorithms three times for each T, where T is set to be 256, 512, 1024, 2048, and 4096 respectively. Both $\mathbf{H}$ and $\mathbf{W}(p)$ are randomly initialized for CNSC, with the same initialization used for the algorithm in [11], allowing a fair comparison. The results of these three tests, together with their average are shown in Figure 4. From the plot of RR, it is clear that the proposed CNSC algorithm is consistently better than the algorithm [11] in terms of separation quality for both the individual tests and their average performance. The plot of REE implies that the proposed algorithm is less accurate as compared with [11]. However, for audio object extraction, the separation quality is the most important factor. According to the RR measurements, the CNSC has considerably better separation performance.

V. Conclusions

The concept and algorithm of CNSC have been presented, based on the extension of the standard NSC using a convolutive model for the representation of the data matrix. Its advantages over the standard NSC and an existing convolutive coding method have been demonstrated in the context of audio object and feature separation. The proposed technique is especially useful for the analysis of signals whose temporal features are considered to be critical, although it is applicable to a wide range of applications including the analysis of potentially more complex auditory scenes.

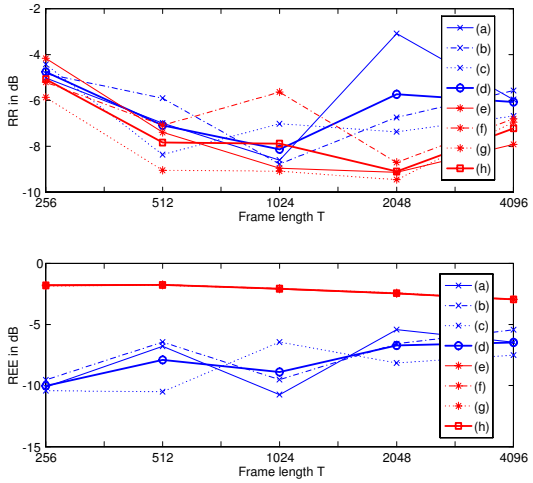

Fig. 4. The comparison of the performance indices (i.e., RR and REE) varying with T between the two algorithms. (a), (b), (c) and (d) are the results of the three random tests and their average respectively by applying the algorithm in [11]. (e), (f), (g) and (h) are the corresponding results obtained by applying the proposed CNSC algorithm.

REFERENCES

[1] P. Paatero, "Least squares formulation of robust non-negative factor analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 37, pp. 23-35, 1997.

[2] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," *Advances in Neural Information Processing 13* (in Proc. NIPS 2000), MIT Press, 2001.

[3] P. O. Hoyer, "Non-negative sparse coding," in *Neural Networks for Signal Processing XII (Proc. IEEE Workshop on Neural Networks for Signal Processing)*, pp. 557-565, Martigny, Switzerland, 2002.

[4] P. O. Hoyer, "Non-negative matrix factorization with sparseness constraints," *Journal of Machine Learning Research*, no. 5, pp. 1457-1469, 2004.

[5] J. Kggert and E. Korner, "Sparse coding and NMF," in *Proc. IEEE Int. Joint Conf. on Neural Networks*, pp. 2529-2533, 2004.

[6] P. D. O'Grady and B. A. Pearlmutter, "Convolutive non-negative matrix factorisation with a sparseness constraint," in *Proc. IEEE Int. Workshop on Machine Learning for Signal Processing*, pp. 427-432, Maynooth, Ireland, Sept.6-8, 2006.

[7] A. Cichocki, R. Zdunek, and S. Amari, "Csiszar's divergences for non-negative matrix factorization: family of new algorithms," *Spinger Lecture Notes in Computer Science*, vol. 3889, pp. 32-39, 2006.

[8] P. Smaragdis, "Non-negative matrix factor deconvolution, extraction of multiple sound sources from monophonic inputs," in *Proc. 5th Int. Conf. on Independent Component Analysis and Blind Signal Separation*, Granada, Spain, Sept. 22-24, Lecture Notes on Computer Science (LNCS 3195), pp.494-499, 2004.

[9] P. Smaragdis, "Convolutive speech bases and their application to supervised speech separation," *IEEE Trans. Audio Speech and Language Processing*, vol. 15, no. 1, pp. 1-12, 2007.

[10] W. Wang, Y. Luo, S. Sanei, and J. A. Chambers, "Non-negative matrix factorization for note onset detection of audio signals," in *Proc. IEEE Int. Workshop on Machine Learning for Signal Proces.*, pp. 447-452, Maynooth, Ireland, Sept.6-8, 2006.

[11] W. Wang, "Squared Euclidean distance based convolutive non-negative matrix factorization with multiplicative learning rules for audio pattern separation," in *Proc. IEEE Int. Symp. on Signal Proces. and Info. Tech.*, Cairo, Egypt, Dec. 15-18, 2007.