NEIGHBOR EMBEDDING WITH NON-NEGATIVE MATRIX FACTORIZATION FOR IMAGE PREDICTION

Christine Guillemot and Mehmet Turkan

INRIA, Campus Universitaire de Beaulieu, 35042 RENNES, FRANCE

Christine.Guillemot@inria.fr

ABSTRACT

The paper studies several non-negative matrix factorization methods with nearest neighbors constrained dictionaries for image prediction. The methods considered include the multiplicative update algorithm, the projected gradient algorithm, as well as the graph-regularized NMF solution which aims at taking into account the geometrical structure of the input data. The Intra prediction problem based on these NMF solutions amounts to a neighbor embedding problem. Both prediction and rate-distortion performances are then given in comparison with other neighbor embedding methods like locally linear embedding (LLE) and locally linear embedding with low dimensional neigborhood representation (LLE-LDNR).

Index Terms— Image compression, prediction, non-negative matrix factorization, data dimensionality reduction

1. INTRODUCTION

Non-negative Matrix Factorization (NMF) techniques have recently become popular for finding representations of non-negative high dimensional input data [8], [6]. NMF searches for two lower dimensional nonnegative matrices whose product gives a good approximation of the input matrix. One of the two component matrices can be seen as containing in its columns basis vectors which will give a good linear approximation of the input data. The underlying assumption is that two data points which are close in the geometry of the data distribution will have representations in the searched basis which will also be close to each other. This is known as the manifold assumption. The non-negativity constraints allowing only additive linear combinations lead to so-called parts-based representations.

The NMF problem is actually a least squares problem with bound constraints. Many algorithms have been proposed to solve this problem. The first method proposed in [8] was based on alternating nonnegative least squares. The most widely used method to solve this problem is the multiplicative update algorithm proposed in [6]. The alternating nonnegative least squares framework has been revisited by several researchers leading to faster convergence algorithms based on either the projected gradients method [7] or the block principal pivoting method [5]. The authors in [5] report faster convergence speed for the projected gradients method when the tolerance (or approximation error) is loose, that is for the first iterations, when compared with the block principal pivoting method, and vice versa when the tolerance is tight.

NMF can be regarded as a data dimensionality reduction method based on a low rank approximation of the input data matrix. Other data dimensionality reduction methods include linear solutions (e.g. PCA) which project the input data into a lower dimensional space via a linear transformation, non linear solutions (e.g. Locally Linear Embedding LLE [9], LLE with Low-Dimensional Neighborhood Representation(LLE-LDNR) [3], Isomap[10]) which aim at learning the underlying data manifold. The design of these techniques has indeed been guided by the assumption that the input data is sampled from a probability distribution on a low dimensional manifold embedded in the high dimensional Euclidean space [9].

One desired property of data dimensionality reduction techniques is thus that the geometrical structure of the input data distribution is preserved in the lower dimensional space. That is, if two data points are close in the data distribution geometry, their representations in the low-dimensional space should also be close. This led to the introduction of further variants of NMF algorithms, such as Locality Preserving NMF [2], graph-regularized NMF (GNMF)[1], and Neighborhood Preserving NMF [4] which add constraints between a point and its neighbors. In these methods, a nearest neighbor graph is constructed to capture the intrinsic geometry of the input data and thus model the local manifold structure.

This paper addresses the problem of Intra prediction which is a key component of image and video compression algorithms. Given observations, or known samples in a spatial neighborhood, the goal is to estimate unknown samples of the block to be predicted. We consider the NMF framework and adapt several variants of NMF algorithms (multiplicative updates, projected gradients, and graph regularization) to the targeted prediction problem. In all methods, we fix one of the component matrices which can be seen as a dictionary matrix containing in its columns basis vectors. The non-negative dictionary matrix $\mathbf{A}$ has in its columns the K nearest neighbors to the vector formed by the known samples in the neighborhood of the block to be predicted. These K nearest neighbors are texture patches of the same shape taken from the known part of the image. Only the matrix containing the weights of the linear approximation must then be found. This corresponds to a step referred to as neighbor embedding. The underlying assumption is that the corresponding uncomplete and complete patches have similar neighborhoods on some nonlinear manifolds. The neighbor embedding based on NMF for the targeted prediction problem is compared with the one given by LLE and LLE-LDNR which are widely used neighbor embedding methods.

The rest of the paper is organized as follows. Section 2 formulates the image prediction problem. Section 3 reviews NMF methods and Section 4 describes the dictionary constrained NMF algorithms considered here for prediction. Section 5 briefly recalls the compression algorithm with which the performances of the different methods have been assessed. Section 6 gives performance illustrations in the context of prediction and compression.

2. THE PREDICTION PROBLEM

Let X be a texture patch which comprises a known part X_k (of a given shape) formed by the pixels located in a causal neighborhood and of an unknown part X_u formed by the block to be predicted (see Fig.1). For each input texture patch X, we constitute a training set of patches by taking all patches in a search window SW within the coded-decoded causal part of the image. Each patch of the training set is also formed by a so-called "*known*" part (set of N_1 pixels at the same positions as the known pixels of X, also referred to as the template) and an "*unknown*" part (set of N_2 pixels at the same positions as the unknown pixels of X). We assume that the set of data points formed by the known template pixels and the set of data points formed by the complete patches (template plus block to be predicted) belong to two related manifolds. The goal of the proposed methods is to "*connect*" these two manifolds, and this in order to make sure that a good approximation of the known part of the input patch (i.e. of the template) will also lead to a good approximation of the complete patch (template plus block to be predicted).

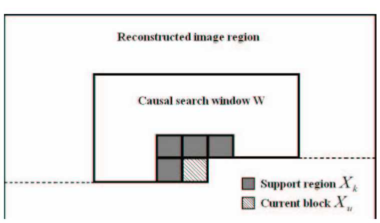

Fig. 1. Prediction problem statement and notations: $\mathbf{X_k}$ is the approximation support (known pixels), $\mathbf{X_u}$ is the current block to be predicted, and $\mathbf{SW}$ is the search window from which texture patches are taken to construct the dictionary.

3. NON-NEGATIVE MATRIX FACTORIZATION

NMF is a data dimensionality reduction method based on a lower rank input data approximation in which the factors are non negative [6]. Given a non-negative input data matrix $\mathbf{X} \in \mathbf{R}^{m\times n}$, in which each column is an input data point (or vector), the aim is thus to find non-negative matrix factors $\mathbf{A} \in \mathbf{R}^{m\times k}$ and $\mathbf{V} \in \mathbf{R}^{k\times n}$ which minimize $||\mathbf{X}-\mathbf{A}.\mathbf{V^T}||_F^2$, subject to $\mathbf{A} \geq 0$ and $\mathbf{V} \geq 0$, where $||.||_F$ denotes the matrix Frobenius norm.

3.1. Multiplicative Update

Many algorithms exist to solve the above problem, the most widely used solution being the multiplicative update procedure [6], where the NMF problem is solved by iteratively updating the elements of each matrix $\mathbf{A}$ and $\mathbf{V}$ as

$$V_{a\mu} \leftarrow V_{a\mu}\frac{\left(\mathbf{A}^{\mathrm{T}}\mathbf{X}\right)_{a\mu}}{\left(\mathbf{A}^{\mathrm{T}}\mathbf{A}.\mathbf{V}\right)_{a\mu}} \qquad \mathbf{A}_{ia} \leftarrow \mathbf{A}_{ia}\frac{\left(\mathbf{X}\mathbf{V}^{\mathrm{T}}\right)_{ia}}{\left(\mathbf{A}.\mathbf{V}\mathbf{V}^{\mathrm{T}}\right)_{ia}}. \quad (1)$$

Here, $a\mu$ (or ia) represents the a^{th} (or i^{th}) row and μ^{th} (or a^{th}) column elements of the corresponding matrices respectively. The Euclidean distance $||\mathbf{X} - \mathbf{A}.\mathbf{V}||_2^2$ is decreasing under the above update rules as shown in [6]. One can interpret that a column vector X_i of $\mathbf{X}$ is approximated by a linear combination of the columns of dictionary $\mathbf{A}$, the weights being given by the column vectors of $\mathbf{V}$.

3.2. Projected Gradients

The above problem can also be solved by using a projected gradient method [7], which updates the two matrix components at each iteration l as

$$\mathbf{V}^{(l+1)} \leftarrow P_\Omega[V^{(l)} - \eta_V^{(l)} P_V^{(l)}] \quad (2)$$

$$\mathbf{A}^{(l+1)} \leftarrow P_\Omega[\mathbf{A}^{(l)} - \eta_A^{(l)} P_A^{(l)}]. \quad (3)$$

where $P_\Omega(x)$ is a projection of x onto the convex set defining the subspace of nonnegative real numbers, $P_V^{(l)}$ and $P_A^{(l)}$ are descent directions for V and A, and $\eta_V^{(l)}$ and $\eta_A^{(l)}$ are learning rules. One simple method for the projection is to zero all negative values in x.

3.3. Graph-regularized NMF

With the above NMF algorithms, the data representation is learned in the Euclidean space. A variant, called *Graph-regularized NMF* (GNMF) is proposed in [1] which better takes into account the geometrical structure of the input data. A graph of the nearest neighbors among the input data points is first constructed, and the corresponding edge weight matrix is defined with each element $w_{i,j} = 1$ if the data point X_j is in the set of K neighors of X_i and 0 otherwise. The problem is then formulated as the minimization of the objective function

$$||\mathbf{X} - \mathbf{A}.\mathbf{V}||_F^2 + \lambda Tr(\mathbf{V}^T.\mathbf{L}.\mathbf{V}), \quad (4)$$

where $Tr(.)$ denotes the trace of a matrix, and where L is the Graph Laplacian computed as $\mathbf{L} = \mathbf{D} - \mathbf{W}$ with $\mathbf{D}$ being a diagonal matrix whose entries are column sums of $\mathbf{W}$. The multiplicative update rule for the dictionary matrix $\mathbf{A}$ remains unchanged with respect to the initial NMF algorithm of [6], but the update rule for the coefficient matrix becomes:

$$\mathbf{A}_{ia} \leftarrow \mathbf{A}_{ia} \frac{\left(\mathbf{X}\mathbf{V}^{\mathrm{T}} + \lambda \mathbf{W}.\mathbf{V}\right)_{ia}}{\left(\mathbf{A}.\mathbf{V}\mathbf{V}^{\mathrm{T}} + \lambda \mathbf{D}.\mathbf{V}\right)_{ia}}.$$

4. NMF WITH KNN-BASED DICTIONARY

Let us now come back to the problem of image prediction. Let X_u be the input block to be predicted and X_k its causal neighborhood (also called template). Both unknown X_u and known X_k pixels form the input patch X (put in a vector form) to be completed. For each input patch X, we fix the non-negative dictionary $\mathbf{A}$. The columns of the dictionary matrix $\mathbf{A} = [\frac{\mathbf{P}}{\mathbf{Q}}]$ are formed by the K-closest texture patches taken from a search window SW within the previously coded/decoded causal part of the image. The distance is computed between the known samples X_k (template pixels) and the pixels at the same positions in the patches taken from SW. The idea is to first obtain an NMF representation V_{opt} of the known set of pixels X_k and keep the same representation vector to approximate the unknown pixels X_u of the block to be predicted. Thus only the rules for updating the coefficient matrices (given in Eq.1, Eq.2 and Eq.4) for the three NMF methods are applied, the dictionary matrix being kept fixed thru all iterations.

The update equation can be iterated until the residual energy on the known part X_k is under a given threshold or until a pre-defined iteration number is reached. However, a low appoximation error of the known samples X_k does not induce a low prediction error and good RD performance for the block to be predicted X_u. To improve the prediction and RD performances, the best number K of nearest patches can be searched in order to minimize an MSE or and RD cost function on the block to be predicted. This optimum value of K must then be signalled to the decoder. Once the best approximation vector V_{opt} has been found, the estimated pixels $\hat{X}_u$ of the block to be predicted are then obtained by multiplying the sub-matrix $\mathbf{Q}$ by the vector V_{opt} found, as $\hat{X}_u = \mathbf{Q}.V_{opt}$.

5. APPLICATION TO PREDICTION AND COMPRESSION

The compression algorithm used for validating the prediction approach based on the above NMF algorithms with KNN constrained dictionary matrices is the same as in [11]. The top 4 rows and left 4 columns of blocks of size 4x4 are intra predicted with the H.264/AVC Intra prediction modes. The algorithm then proceeds with the prediction based on NMF, using 7 template shapes as in [11]. Once a block has been predicted, the DCT transformed residue is quantized with a uniform quantizer, zig-zag scanned, and encoded with an algorithm similar to JPEG. A uniform quantization matrix with $\Delta = 16$ is weighted by a quality factor. The quality factor (q_f) is increased from 10 to 90 with a step size of 10, and the corresponding weight $\mathbf{w}_{q_f}$ is calculated as

$$\mathbf{w}_{q_f} = \begin{cases} 50/q_f & \text{if } q_f \leq 50 \\ 2 - 0.02q_f & \text{if } q_f > 50 \end{cases}. \quad (5)$$

Image blocks are processed in a raster scan order, and the reconstructed image is obtained by adding the quantized residue to the prediction. The training patches are collected from the search region which is located in a causal coded/decoded neighborhood of the unknown block to be predicted. The optimum value of K as well as the best template shape are also Huffman encoded.

6. PERFORMANCE ILLUSTRATION

Prediction and RD performances obtained are compared with those given by other neighbor embedding methods, in particular by LLE [9] and LLE-LDNR [3]. The LLE method with low-dimensional neighborhood representation is a variant of LLE in which the embedding weights are computed using a low-dimensional representation of the input point neighborhood. They are also compared with those obtained with template matching (TM) and the H.264 Intra prediction modes.

Fig.2 (left top and bottom) show the PSNR of the predicted Barbara and Cameraman images when the best value of K (between 1 and 8) is searched in order to minimize the prediction error and is then signalled to the decoder. For fixed K, simulation results have shown that the multiplicative update and GNMF algorithms outperform all the other methods, however, when optimizing the value of K with the MSE criterion, best prediction performances have been obtained with the projected gradient method, as shown in Fig.2.

Fig.2 (right top and bottom) shows the Rate-PSNR curves obtained for the Barbara and Cameraman images with the different methods, when the best value of K (between 1 and 8) is searched in order to minimize an RD cost function, and is then signalled to the decoder. The PSNR values here are thus those of the reconstructed image when adding the coded and decoded prediction residue to the predicted image. All 3 NMF algorihms lead to comparable results and outperform LLE-based and TM solutions. They outperform the H264 Intra prediction modes on textured images, which are images on which the simple H.264 Intra prediction modes tend to fail. These methods are obviously less advantageous on images (like Cameraman) with large smooth and uniform areas on which the H.264 Intra prediction modes already give good

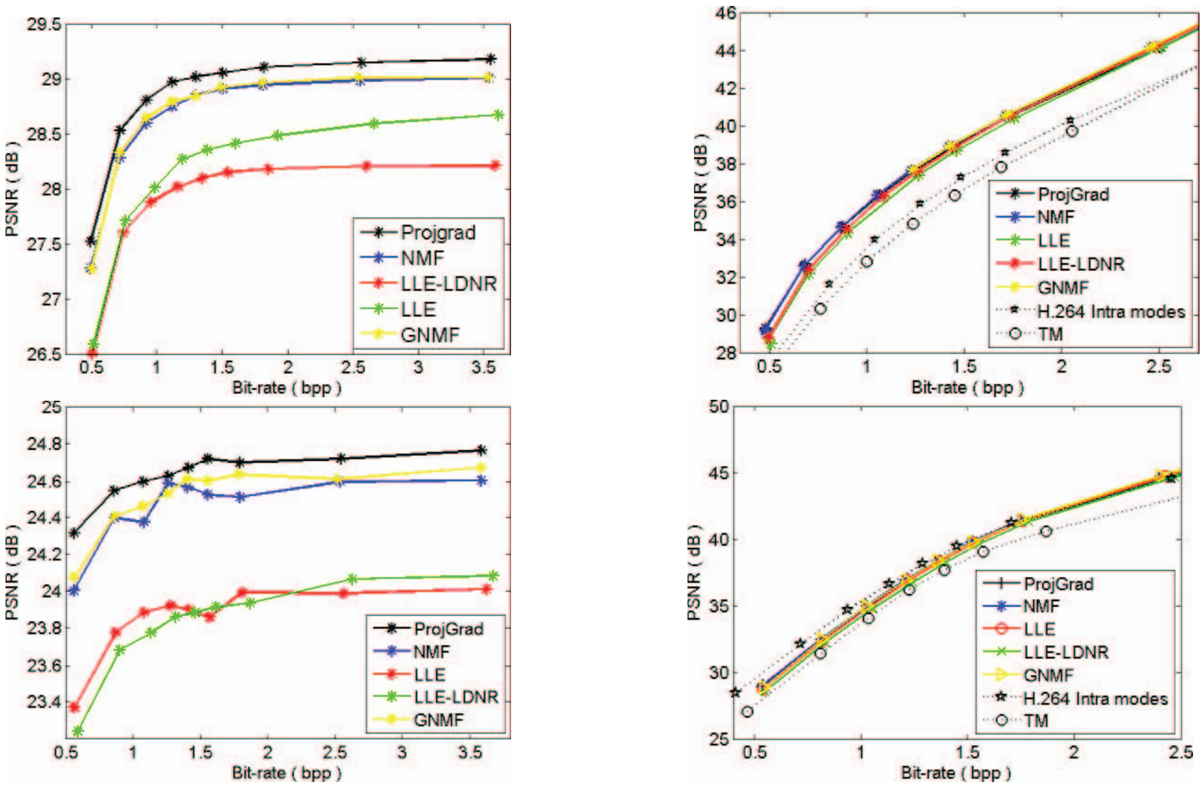

Fig. 2. (Left) Prediction and (right) rate-distortion (RD) performances for Barbara (top) and Cameraman (bottom), with NMF (multiplicative update), with projected gradients and with GNMF in comparison with LLE and LLE-LDNR. The best k value is searched in order to minimize the prediction error (for the prediction curves) or the RD cost function (for the RD curves).

results. These methods are thus good candidates as extra prediction modes, which the encoder can select according to an RD criterion.

7. REFERENCES

[1] D. Cai, X. He, J. Han, and T. S. Huang. Graph regularized non-negative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011, to appear.

[2] D. Cai, X. He, X. Wang, H. Bao, and J. Han. Locality preserving nonnegative matrix factorization. In *Proc. Int. Joint Conference on Artificial Intelligence (IJCAI'09)*, 2009.

[3] Y. Goldberg and Y. Ritov. Ldr-lle: Lle with low-dimensional neighborhood representation. In *Proceedings of the 4th International Symposium on Advances in Visual Computing, Part II*, ISVC '08, pages 43–54, 2008.

[4] Q. Gu and J. Zhou. Neighborhood preserving nonnegative matrix factorization. In *Proc. 20th British Machine Vision Conference*, 2009.

[5] J. Kim and H. Park. Toward faster nonnegative matrix factorization: A new algorithm and comparisons. In *Proc. Eighth IEEE International Conference on Data Mining*, pages 353–362, 2008.

[6] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. *Advances in Neural Information Process. Syst. (NIPS)*, 2000.

[7] C. Lin. Projected gradient methods for non-negative matrix factorization. *Neural Computation.*, 19:2756–2779, Oct. 2007.

[8] P. Paatero and U. Tapper. Positive matrix factorization: A non-negative factor model with optimal utilization of error. *Environmetrics*, 5:111–126, 1994.

[9] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, Dec. 2000.

[10] J. Tenenbaum, V. de Silva, and J. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323, Dec. 2000.

[11] M. Turkan and C. Guillemot. Image prediction: Template matching vs. Sparse approximation. In *IEEE Int. Conf. Image Process. (ICIP)*, pages 789–792, 2010.