Learning Sparse Basis Vectors in Small-Sample Datasets through Regularized Non-Negative Matrix Factorization

T. Yamanaka, A. Perera, B. Raman, A Gutierrez-Galvez and R. Gutierrez-Osuna

Department of Computer Science, Texas A&M University, College Station, TX 77840
E-mail: yamanaka@mem.iee.or.jp, {aperera, barani, agustin, rgutier}@cs.tamu.edu

Abstract

This article presents a novel dimensionality-reduction technique, Regularized Non-negative Matrix Factorization (RNMF), which combines the non-negativity constraint of NMF with a regularization term. In contrast with NMF, which degrades to holistic representations with decreasing amount of data, RNMF is able to extract parts of objects even in the small-sample case.

Keywords: Approximate Methods, Feature Extraction, Constrained Optimization, Face Recognition

1. INTRODUCTION

Dimensionality-reduction techniques such as Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) have been successfully applied to extract meaningful dimensions from high dimensional data in many research areas. A promising dimensionality-reduction technique that emerged in recent years is Non-negative Matrix Factorization (NMF) [1,2,3]. In this method, a non-negativity constraint is imposed on the bilinear decomposition into basis vectors and encoding vectors. Since no subtractions can occur, NMF is consistent with the intuitive notion of combining parts to form a whole [1]. Thus, NMF has the ability to learn basis vectors that are the intrinsic local parts of objects (e.g., a face consists of two eyes, a nose, a mouth, etc), whereas the basis vectors with PCA and LDA are holistic (e.g., eigenfaces) [1,4].

NMF has been successfully applied to face recognition [5,6,7] and image classification [8]. For face recognition purposes, NMF provides better classification performance than PCA in the case of partial occlusions due to the locality of the NMF basis vectors [7]. NMF has been also applied to clustering gene expressions [9], and to extracting the substructure of gene expressions (metagenes) [10]. In the latter case, the ability to learn local parts leads to the extraction of meaningful metagenes. Thus, the locality of basis vectors in NMF is an important property for the purpose of feature extraction in a number of domains.

It has been shown that, in order to obtain local basis vectors, the training data has to contain a variety of combinations of the intrinsic local parts of the objects [11]. However, this condition is unfeasible in real applications due to the limited-number of available samples, which may result in basis vectors that are holistic. In order to improve the locality of NMF, a method known as Local NMF (LNMF) has been proposed [12]. Although LNMF produces basis vectors that

represent the local parts of the objects, the algorithm converges slowly and leads to relatively poor reconstructions of the data [13].

This article proposes a novel method, Regularized NMF (RNMF), to improve the locality of the basis vectors in the limited sample case. RNMF is related to a previous NMF regularization method known as Non-Negative Sparse Coding (NNSC) [14,15]. NNSC employs a regularization term that favors *sparse encoding vectors*. As a result, each object is represented by a linear combination of a few basis vectors, but the basis themselves are not sparse. In contrast, the regularization term in RNMF favors *sparse basis vectors* which, combined with the non-negativity constraint, leads to a spatially-local parts-based decomposition of objects. The performance of the proposed model is experimentally evaluated against PCA, NMF and LNMF on two facial image databases.

2. NON-NEGATIVE MATRIX FACTORIZATION

In what follows, we assume that the data matrix is expressed as an $n \times m$ matrix $\mathbf{V}$, each column being an n-dimensional sample out of a dataset with m samples. Dimensionality-reduction techniques construct approximate bilinear factorizations of the form

$$\mathbf{V}_{ij} \approx (\mathbf{WH})_{ij} = \sum_{a=1}^{r} \mathbf{W}_{ia}\mathbf{H}_{aj} , \quad (1)$$

where $\mathbf{W}$ is $n \times r$, $\mathbf{H}$ is $r \times m$ and $\mathbf{M}_{ij}$ represents the (i, j) element of the matrix $\mathbf{M}$. The r columns of $\mathbf{W}$ are called basis vectors. Each column of $\mathbf{H}$ is called an encoding vector, and is in one-to-one correspondence with a sample vector in $\mathbf{V}$. Each encoding vector represents the coefficients of a linear decomposition of the corresponding sample vector in terms of the basis vectors in $\mathbf{W}$. The

product of **WH** is a reconstructed form of the data in **V**, and the columns of **WH** are called reconstructed vectors.

PCA performs this decomposition by minimizing the reconstruction error under the constraint that the basis vectors be orthogonal. In this case, the columns of **W** are the eigenvectors of the correlation matrix of **V** [4]. In NMF, the reconstruction error is minimized under the constraint that all the elements in **W** and **H** be non-negative, which leads to a parts-based representation of the data [1].

Two iterative algorithms have been proposed for the effective computation of **W** and **H** in NMF [2]. The first algorithm employs the Euclidean distance between **V** and **WH** (Frobenius error) as an objective function:

$$J = \left\| \mathbf{V} - \mathbf{WH} \right\|^2 = \sum_{ij} (\mathbf{V}_{ij} - (\mathbf{WH})_{ij})^2 , \tag{2}$$

whereas the second algorithm uses the information-theoretic divergence:

$$J = \sum_{ij} (\mathbf{V}_{ij} \log \frac{\mathbf{V}_{ij}}{(\mathbf{WH})_{ij}} - \mathbf{V}_{ij} + (\mathbf{WH})_{ij}), \tag{3}$$

both subject to the constraint $\{\mathbf{W}_{ij}, \mathbf{H}_{ij} \geq 0; \forall i, j\}$. It can be shown that the objective functions in Eqs. (2) and (3) can be monotonically decreased with the following update rules, respectively [2].

(Rule I) $$\mathbf{H}_{aj} \leftarrow \mathbf{H}_{aj} \frac{(\mathbf{W}^T \mathbf{V})_{aj}}{(\mathbf{W}^T \mathbf{WH})_{aj}} \qquad \mathbf{W}_{ia} \leftarrow \mathbf{W}_{ia} \frac{(\mathbf{VH}^T)_{ia}}{(\mathbf{WHH}^T)_{ia}} \tag{4a,b}$$

(Rule II) $$\mathbf{H}_{aj} \leftarrow \mathbf{H}_{aj} \frac{\sum_i \mathbf{W}_{ia} \mathbf{V}_{ij} / (\mathbf{WH})_{ij}}{\sum_k \mathbf{W}_{ka}} \quad \mathbf{W}_{ia} \leftarrow \mathbf{W}_{ia} \frac{\sum_j \mathbf{H}_{aj} \mathbf{V}_{ij} / (\mathbf{WH})_{ij}}{\sum_\nu \mathbf{H}_{a\nu}} \tag{5a,b}$$

Note that, since these update rules are multiplicative, initial non-negative matrices remain non-negative for all future iterations. Following each iterative step, the basis vectors are normalized with the L_1 or L_2 norm.

(L_1 Normalization) $$\mathbf{W}_{ia} \leftarrow \mathbf{W}_{ia} \Big/ \sum_j \mathbf{W}_{ja} \tag{6}$$

(L_2 Normalization) $$\mathbf{W}_{ia} \leftarrow \mathbf{W}_{ia} \Big/ \sqrt{\sum_j \mathbf{W}_{ja}{}^2} \tag{7}$$

In order to increase the locality of the basis vectors, LNMF uses a modified divergence:

$$J = \sum_{ij} (\mathbf{V}_{ij} \log \frac{\mathbf{V}_{ij}}{(\mathbf{WH})_{ij}} - \mathbf{V}_{ij} + (\mathbf{WH})_{ij}) + \alpha \sum_{ij} (\mathbf{W}^T \mathbf{W})_{ij} - \beta \sum_i (\mathbf{H}\mathbf{H}^T)_{ii} , \tag{8}$$

where α, β are non-negative constants [12]. It must be noted that in [12] the term $\alpha \sum_{ij} (\mathbf{W}^T \mathbf{W})_{ij}$ is omitted when driving an update rule of **W** for Eq. (8). Therefore, LNMF seeks basis vectors that minimize the divergence and maximize the diagonal elements of $\mathbf{HH}^\mathbf{T}$ (i.e., the autocorrelation matrix of the encoding vectors). The resulting update rule for **W** is the same as Eq. (5b), followed by L_1 normalization, whereas the update rule for **H** is given by

$$\mathbf{H}_{aj} \leftarrow \sqrt{\mathbf{H}_{aj} \sum_i \mathbf{W}_{ia} \mathbf{V}_{ij} / (\mathbf{WH})_{ij}} . \tag{9}$$

In the original LNMF, the resulting encoding matrix **H** does not yield to the minimum reconstruction error for the basis vectors **W**. This is due to the simplification made by the authors when deriving the update rule for **H** (refer to [12] for details). Therefore, for the purpose of evaluating the basis vectors **W** in terms of the reconstruction error, **H** needs to be recalculated after the fixed-point iteration (5b), (9) has converged. In our article, the recalculation of **H** was

performed by iteratively calculating **H** with Eq. (5a) until convergence, using the final value of **W** obtained from Eqs. (5b), (9).

3. REGULARIZED NON-NEGATIVE MATRIX FACTORIZATION

This article proposes a regularization method to obtain a sparse representation for the basis vectors in NMF. Regularization encourages smooth regression by adding a penalty term $\lambda f(\mathbf{W})$ to the objective function [16], or sparse encoding by adding a penalty term $\lambda f(\mathbf{H})$ [17], where $f(\bullet)$ is a complexity-penalty function, and λ is the regularization parameter. As mentioned earlier, regularization was employed in NNSC [14,15] to obtain *sparse encoding vectors* through a regularization term $\lambda \sum_{ij} \mathbf{H}_{ij}$. In contrast, RNMF uses regularization to encourage *sparse basis vectors* with the following objective function:

$$J = \frac{1}{2}\sum_{ij}(\mathbf{V}_{ij} - (\mathbf{WH})_{ij})^2 + \lambda \sum_{ij} \mathbf{W}_{ij}, \tag{10}$$

where λ is a non-negative constant. For simplicity, only Euclidean distance is considered in this article, though RNMF can also be applied to divergence-based objective functions.

The parameter λ weights the relative importance of the reconstruction error and the sparseness. The appropriate range of values for the parameter λ depends on the relative magnitude of the two terms. To decrease this data-dependency, a normalized parameter λ' is instead specified by the user:

$$\lambda = \lambda' \frac{\frac{1}{2}\sum_{ij}(\mathbf{V}_{ij} - (\mathbf{WH})_{ij})^2\Big|_{\lambda=0}}{\sum_{ij}\mathbf{W}_{ij}\Big|_{\lambda=0}}, \tag{11}$$

where $\sum_{ij}(\bullet)\big|_{\lambda=0}$ is the value of $\sum_{ij}(\bullet)$ when λ = 0. These values are calculated once at the beginning of the RNMF algorithm using λ = 0, and then used to normalize the regularization parameter.

The update of **H** for the objective function (10) is the same as that in the original NMF (Eq. (4a)) since the regularization term $\lambda\sum_{ij}\mathbf{W}_{ij}$ does not depend on **H**. However, a multiplicative update rule for **W** as derived in [2] is not guaranteed to converge for an objective function that includes a regularization term (Eqs. (8) and (10)), since the normalization of **W** rescales the regularization term. In the original NMF, this normalization does not prevent the convergence since this algorithm compensates for it by rescaling **H** in subsequent iterations. A solution to the problem of normalization in regularization methods has been proposed by Hoyer for NNSC [14] using gradient descent combined with the non-negativity constraint. A similar strategy is employed in our RNMF algorithm, which results in the following steps.

1. Initialize **W** and **H** to random positive matrices, and normalize each column of W by L_2 norm.

2. Iterate until convergence:

   (a) Update $\mathbf{W}_{ia} \leftarrow \mathbf{W}_{ia} - \mu\frac{\partial J}{\partial \mathbf{W}_{ia}} = \mathbf{W}_{ia} + \mu\left((\mathbf{V} - \mathbf{WH})\mathbf{H}^T\right)_{ia} - \lambda$ $\quad (\mu > 0)$.

   (b) Set to zero any negative values in **W**.

(c) Normalize each column of **W** by L_2 norm.

(d) Update $\mathbf{H}_{aj} \leftarrow \mathbf{H}_{aj} (\mathbf{W}^T \mathbf{V})_{aj} / (\mathbf{W}^T \mathbf{W} \mathbf{H})_{aj}$.

This algorithm decreases the objective function under the normalization constraint if the learning parameter μ is small enough [14]. Note that L_1 normalization ($\sum_i \mathbf{W}_{ij} = 1$) cannot be used since the regularization term $\lambda \sum_{ij} \mathbf{W}_{ij}$ would then become constant.

4. EXPERIMENTAL

Two facial image databases were used to characterize RNMF against NMF, LNMF and PCA. The first database, obtained from CBCL (Center for Biological and Computational Learning, MIT), contained 2,429 hand-aligned frontal faces with several facial expressions [18]. Each image was 19×19 pixels in size, as shown in Fig. 1(a). The second set was the ORL database (AT&T Laboratories Cambridge) [19], which consisted of 400 images (40 people × 10 images). Each image, 112×92 pixels in size, was a frontal view in an upright position with a slight left-right rotation, as shown in Fig. 1(b). The CBCL database was used to investigate small-sample effects on the locality of the basis vectors extracted by NMF. The ORL database, which contains significantly larger images, was used to show that the results obtained with RNMF are not specific to a particular database.

Each column in the data matrix **V** was obtained by sampling the corresponding image in raster-scan fashion. Subsequently, each row in **V** was normalized by its standard deviation in order to weight equally all features in the data vector (i.e., all pixels in the image).

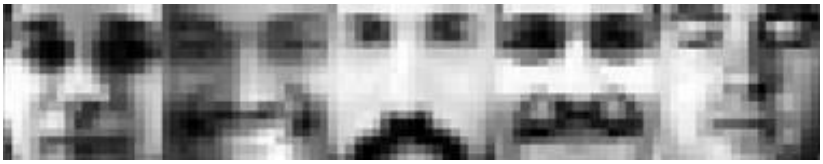
Three measures were used to characterize RNMF, NMF, LNMF and PCA: relative Frobenius error *e*, sparseness *s*, and collinearity ϕ.

$$e = \sqrt{\sum_{ij} (\mathbf{V} - \mathbf{WH})_{ij}^{\ 2}} \Big/ \sqrt{\sum_{ij} \mathbf{V}_{ij}^{\ 2}} \tag{12}$$

$$s = \text{Fraction of zero entries (less than } 10^{-5}\text{) in W} \tag{13}$$

$$\phi = \sum_{i \neq j} (\mathbf{W}^T \mathbf{W})_{ij} \Big/ \sum_{i = j} (\mathbf{W}^T \mathbf{W})_{ij} \tag{14}$$

The relative Frobenius error *e* is a measure of the reconstruction error. The sparseness *s* is a measure of locality, and is large for parts-based representations. The collinearity ϕ is a measure of overlap among the basis vectors, and is low for orthogonal representations.



(a)

(b)

Fig. 1 Examples of face images in databases. (a) CBCL database [18] (19×19 pixels, 2,429 images), (b) ORL database [19] (112×92 pixels, 400 images).

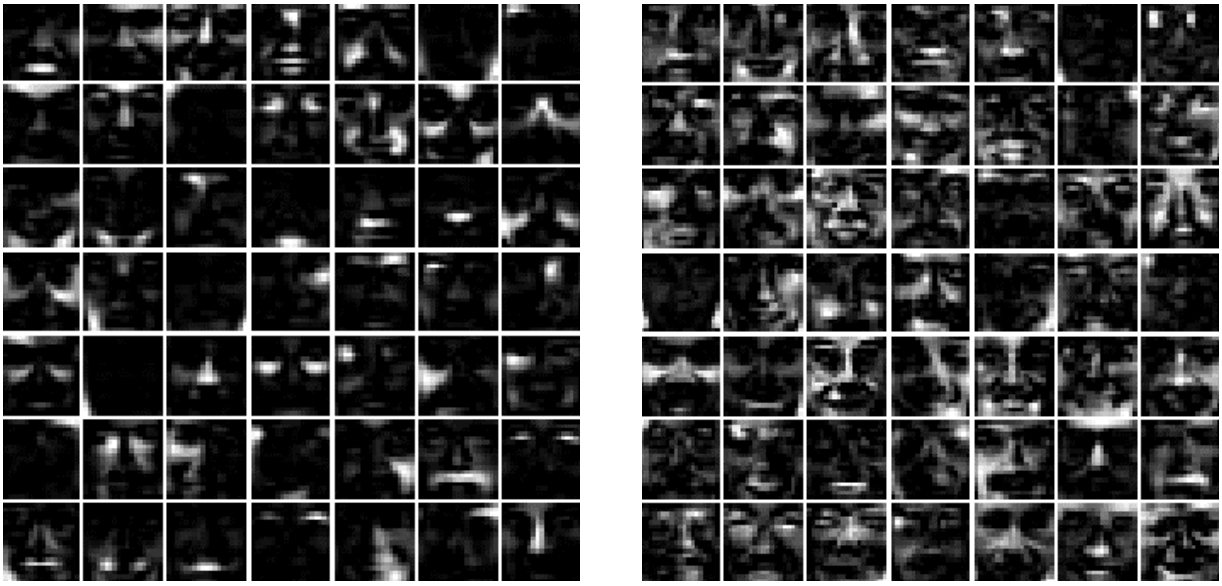
5. RESULTS AND DISCUSSION

5.1 Small-sample Effects on Locality of NMF, LNMF and PCA

The dependence of NMF on sample size was first investigated using the CBCL database. For this purpose, three datasets with different sample sizes were constructed, with 2,429 images (i.e., the complete database), 500 randomly-selected images, and 100 randomly-selected images. The update rule in Eq. (4) and L_1 normalization were used. Random initial values for **H** and **W** were obtained from a uniform distribution (0, 1), and the algorithm was allowed to run for 4,000 iterations.

Performance results are summarized in Table 1 and Fig. 2. As expected, the sparseness degraded as the number of samples decreased. Collinearity also degraded with fewer samples, since the basis vectors became increasingly holistic, as illustrated in Fig. 2(b). Thus, NMF fails to capture the intrinsic local parts when training samples are limited.

Results for NMF, LNMF and PCA on the 100-sample dataset are summarized in Table 2 and Fig. 3. The performance of NMF was largely insensitive to the objective function (D=Divergence, E=Euclidean), in agreement with [1]. However, L_1 normalization improved sparseness, as compared with L_2 normalization. This is due to the fact that L_1-normalization constraint provides more opportunities for elements to be zero in regression models [20]. As expected, LNMF produced highly local representations with large reconstruction error, whereas PCA generated holistic representations with the lowest reconstruction error. It must be noted that, in our implementation of LNMF, **H** was recalculated with Eq. (5a) to minimize the reconstruction error, as described in Section 2. The original LNMF produced a much larger reconstruction error ($e \approx 35\%$) than the value reported in Table 2.



(a) (b)

Fig. 2 Basis vectors obtained by NMF on CBCL database (19×19 pixels), *r*=49, 4,000 iterations. (a) 2,429-sample dataset, (b) 100-sample dataset. The basis vectors became holistic as the number of samples decreased.

Table 1 Characteristics change in basis vectors obtained by NMF as function of sample size. Sparseness is a measure of locality in the basis vectors.

Number of Samples	Error e (%)	Sparseness s (%)	Collinearity ϕ (arb.unit)
2429	8.70	38.82	8.23
500	8.41	36.15	10.17
100	6.12	26.56	18.73

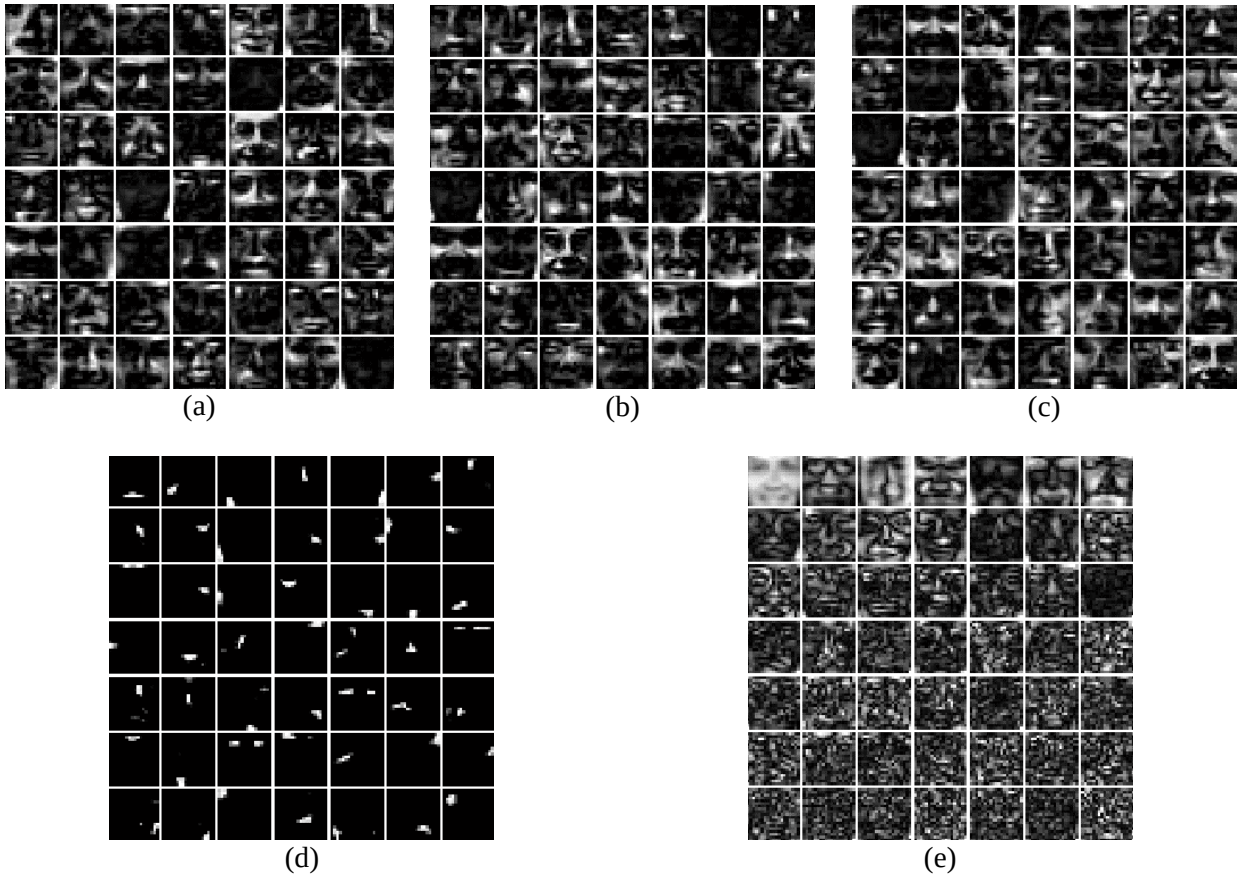


Fig. 3 Basis vectors obtained by NMF, LNMF and PCA. 100-sample CBCL database (19×19 pixels), *r*=49, 4,000 iterations. (a) NMF-D-L_1, (b) NMF-E-L_1, (c) NMF-E-L_2, (d) LNMF (e) PCA (absolute values).

Table 2 Characteristics of basis vectors extracted by NMF (D=Divergence, E=Euclidean), LNMF and PCA.

Fig. 3		Error *e* (%)	Sparseness *s* (%)	Collinearity ϕ (arb.unit)
(a)	NMF(D, L1)	6.40	27.46	18.50
(b)	NMF(E, L1)	6.12	26.56	18.73
(c)	NMF(E, L2)	6.02	21.31	22.78
(d)	LNMF	13.74	95.37	0.26
(e)	PCA	4.87	0.028	0.00

5.2 Evaluation of RNMF

The performance of RNMF was evaluated on the 100-sample CBCL dataset. The evolution of the Frobenius error during iterative calculation is shown in Fig. 4 along with those of NMF and LNMF. NMF and RNMF converged after 1,000 iterations, whereas LNMF required an additional 2,000 iterations. The Frobenius error, sparseness and collinearity for RNMF as a function of the regularization parameter are shown in Fig. 5, whereas the basis vectors obtained for λ'=8 and λ'=20 are illustrated in Fig. 6. An increase in the regularization parameter led to improved sparseness and orthogonality, though at the expense of higher reconstruction errors. Altogether, these results clearly show that RNMF produced more local representations than NMF in the small-sample case (compare Fig. 6 with Fig. 3(a-c)), as well as lower reconstruction errors than LNMF with comparable sparseness.

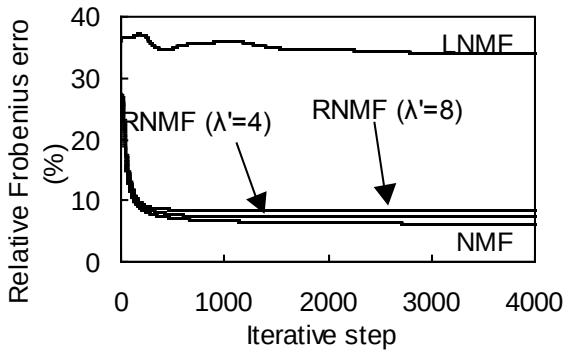
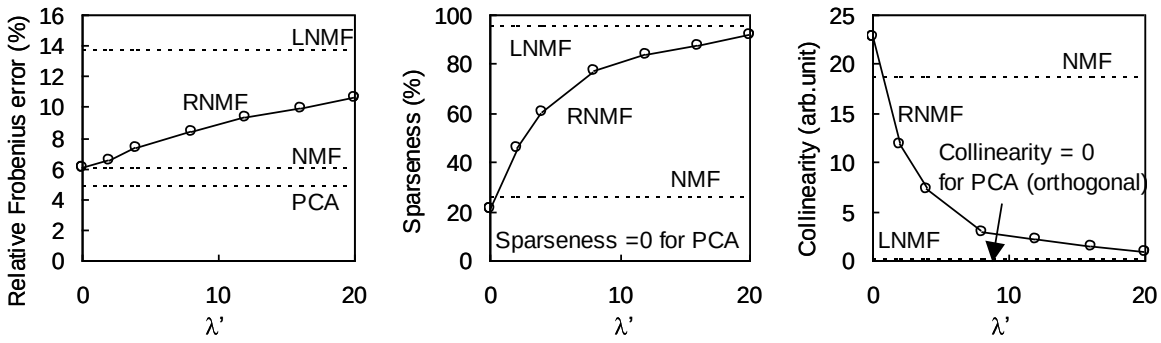
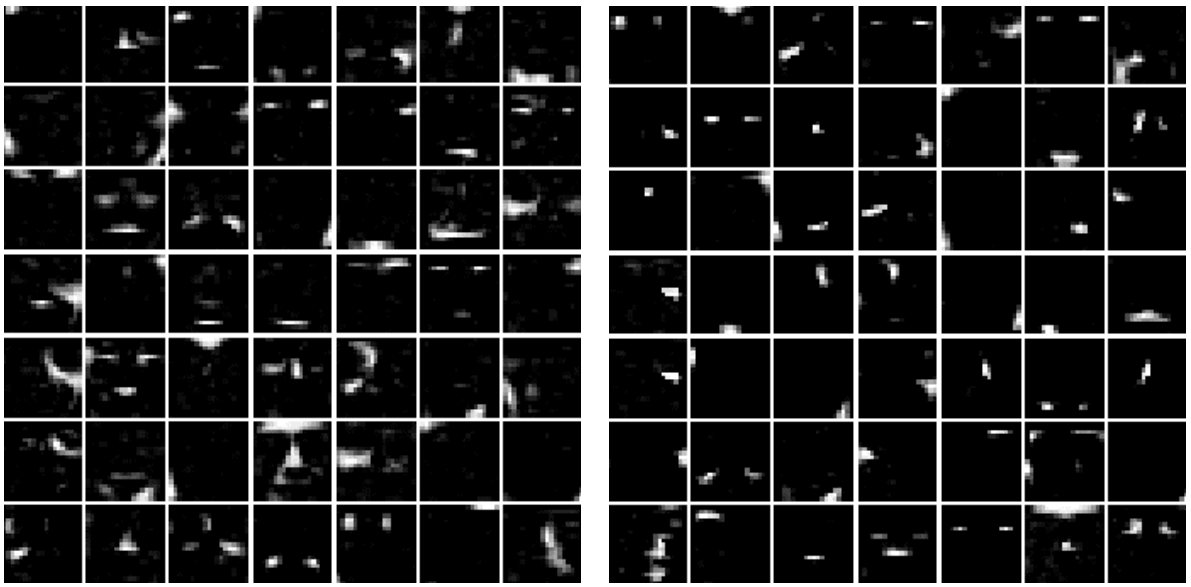


Fig. 4 Relative Frobenius error during calculation of basis vectors and encoding vectors using NMF (Euclidean, L_1 normalization), LNMF (before recalculation of H) and RNMF (λ'=4 and 8).



(a) (b) (c)

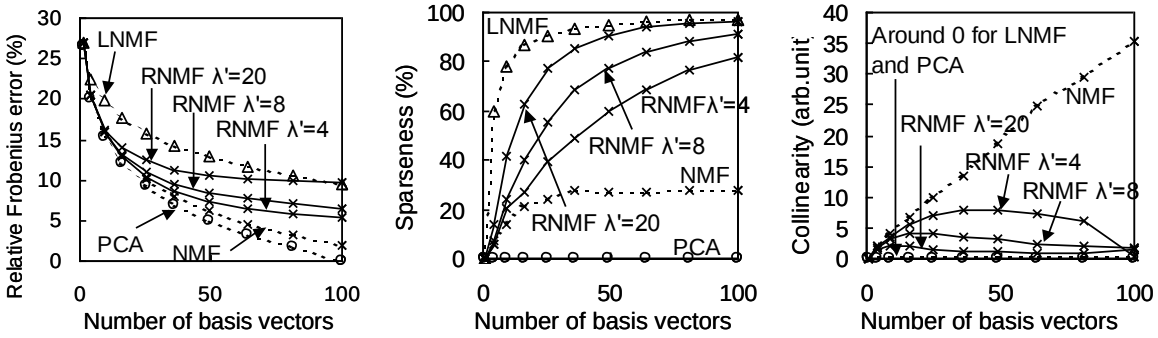
Fig. 5 Performance of RNMF as function of regularization parameter λ', compared with that of NMF (Eucledian, L_1 normalization), LNMF and PCA. 100-sample CBCL database (19×19 pixels), r=49, 4,000 iterations.



(a) (b)

Fig. 6 Basis vectors obtained by RNMF. 100-sample CBCL database (19×19 pixels), r=49, 4,000 iterations. (a) λ'=8, (b) λ'=20.

The performance of the different models as a function of the number of basis vectors *r* is illustrated in Fig. 7. The results observed in Fig. 5 for *r* = 49 generalized for different numbers of basis vectors, indicating that RNMF provided a tradeoff between low reconstruction errors (best for PCA and NMF) and sparseness (best for LNMF). This tradeoff can be controlled by means of the regularization parameter. Moreover, the results in Fig. 7(c) also show that the collinearity of NMF increased in a linear fashion with the number of basis vectors, whereas it remained relatively small in the case of RNMF.



(a) (b) (c)

Fig. 7 Performance of RNMF, NMF (Euclidean, L_1 normalization), LNMF and PCA as function of number of basis vectors. 100-sample CBCL database (19×19 pixels), 4,000 iterations.

In order to characterize the generalization capability of the models, the reconstruction error was calculated for an independent set of 100 images randomly selected from the CBCL database. Encoding vectors **H** for these test images were obtained using the update rule in Eq. (4a) for NMF and RNMF, and Eq. (5a) for LNMF, without updating the basis vectors **W** obtained from training data. The results, summarized in Fig. 8, show that RNMF provided lower reconstruction errors than NMF and LNMF on the independent test set. Since NMF had lower training error (see Fig. 5(a) and Fig. 7(a)), these results indicate that RNMF has improved generalization properties, a result that can be expected from a regularization technique.

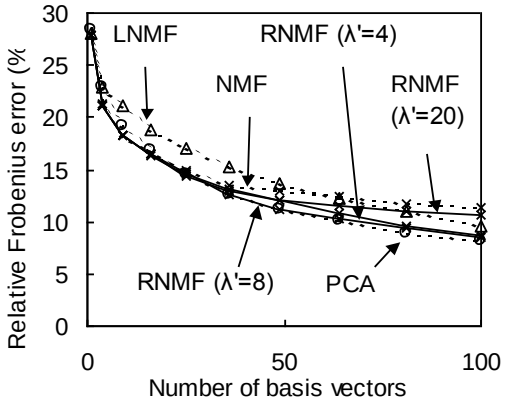
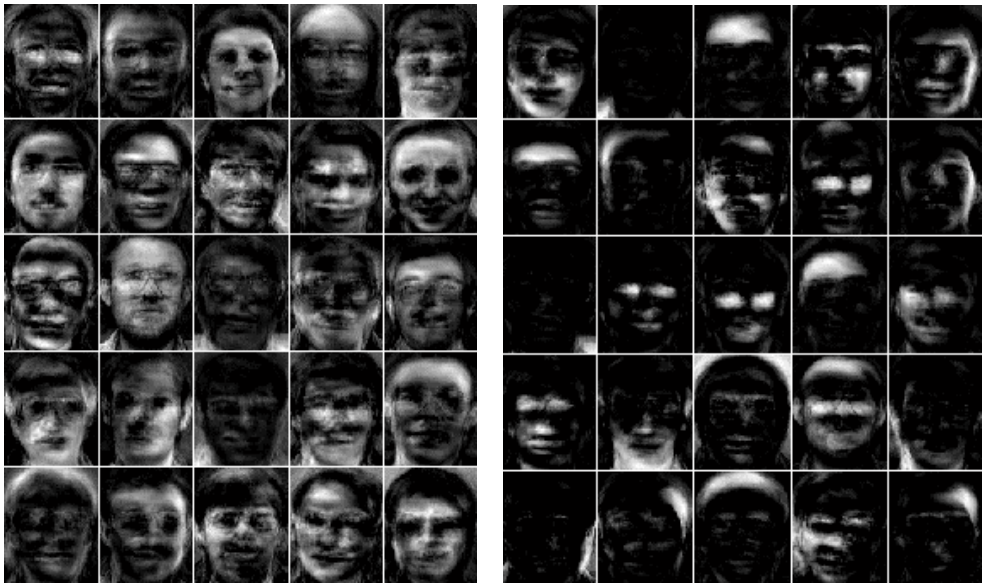


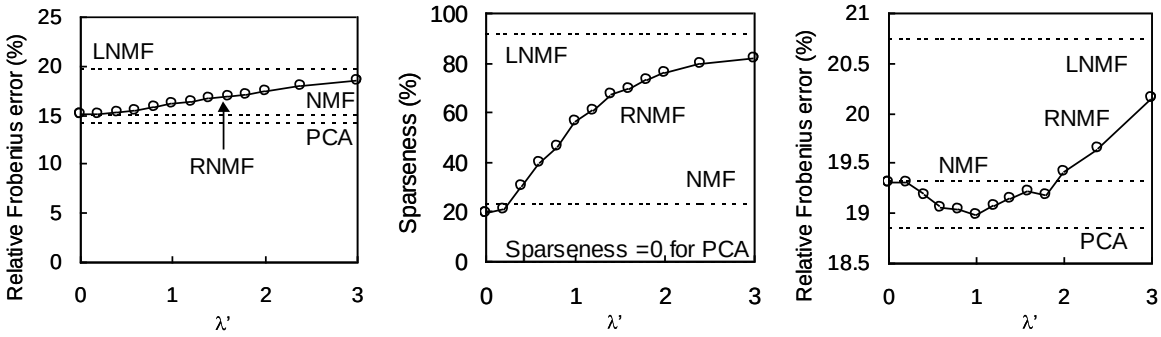
Fig. 8 Relative Frobenius errors of RNMF, NMF, LNMF and PCA on independent test set.

Finally, the performance of RNMF was evaluated using the ORL database. 120 images (40 people × 3 images) were selected to extract basis vectors, and the remaining 280 images were used as test data. Training results, summarized in Figs. 9, 10(a) and 10(b), reinforced the view of RNMF as a parameterized tradeoff between low reconstruction error and high sparseness. Validation results on the test set, shown in Fig. 10(c), also indicate that the generalization capabilities of RNMF were higher than LNMF and NMF for a wide range of values of the regularization parameter λ'. Appropriate values for λ' can be obtained through a separate cross-validation loop.



(a) (b)

Fig. 9 Basis vectors obtained by NMF and RNMF on 120-sample ORL database (112×92 pixels). r=25, 4,000 iterations. (a) NMF, (b) RNMF (λ'=1.0).



(a) (b) (c)

Fig. 10 Performance of RNMF as compared with NMF (Euclidean, L1 normalization), LNMF and PCA. ORL database (112×92 pixels), 120 images for training, 280 images for testing, r=25, 4,000 iterations.

6. CONCLUSIONS

This paper has introduced a novel dimensionality-reduction method, Regularized Non-negative Matrix Factorization (RNMF). The method combines the non-negativity constraint in conventional NMF with a regularization term to obtain sparse representations of objects. The performance of RNMF was systematically evaluated against NMF, LNMF and PCA on two facial databases. Our results show that RNMF leads to sparser basis vectors than NMF, particularly in the small-sample case, and lower training reconstruction errors than LNMF. Thus, the regularization parameter in RNMF provides a tradeoff between spatially local representation and accurate data compression. More importantly, the lower reconstruction errors obtained on independent test data indicate that RNMF has improved generalization properties than both NMF and LNMF, a result that is consistent with the properties of regularization methods. Since RNMF provides high generalization capability and sparse basis vectors, even in small-sample scenarios, it can be concluded that the method is able to extract the intrinsic parts of objects.

ACKNOWLEDGEMENTS

This research was supported by a Postdoctoral Fellowship for Research Abroad from Japan Society for the Promotion of Science.

REFERENCES

[1] D. D. Lee and H. S. Seung, “Learning the Parts of Objects by Non-negative Matrix Factorization,” Nature, vol. 401, pp. 788-791, 1999.

[2] D. D. Lee and H. S. Seung, “Algorithms for Non-negative Matrix Factorization,” Advances in Neural Information Processing System, vol. 13, pp. 556-562, 2001.

[3] P. Paatero and U. Tapper, "Positive Matrix Factorization: a Non-negative Factor Model with Optimal Utilization of Error Estimates of Data Values," Environmetrics, vol. 5, pp. 111-126, 1994.

[4] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 711-720, 1997.

[5] W. Liu and N. Zheng, "Non-negative Matrix Factorization Based Methods for Object Recognition," Pattern Recognition Letters, vol. 25, pp. 893-897, 2004.

[6] D. Guillamet and J. Vitria, "Evaluation of Distance Metrics for Recognition based on Non-negative Matrix Factorization," Pattern Recognition Letters, vol. 24, pp. 1599-1605, 2003.

[7] D. Guillamet and J. Vitria, "Non-negative Matrix Factorization for Face Recognition," Lecture Notes in Computer Science, vol. 2504, pp. 336-344, 2003.

[8] D. Guillamet, J. Vitria, and B. Schiele, "Introducing a Weighted Non-negative Matrix Factorization for Image Classification," Pattern Recognition Letters, vol. 24, pp. 2447-2454, 2003.

[9] J. P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov, "Metagenes and Molecular Pattern Discovery using Matrix Factorization," Proceedings of the National Academy of Sciences of the United States of America, vol. 101, no. 12, pp. 4164-4169, 2004.

[10] P. M. Kim and B. Tidor, "Subsystem Identification Through Dimensionality Reduction of Large-scale Gene Expression Data," Genome Research, vol. 13, pp. 1706-1718, 2003.

[11] D. Donoho and V. Stodden, "When Does Non-negative Matrix Factorization Give a Correct Decomposition into Parts?," Advances in Neural Information Processing System, vol. 16, pp. 1141-1148, 2004.

[12] S. Z. Li, X. W. How, H. J. Zhang, and Q. S. Cheng, "Learning Spatially Localized, Parts-based Representation," IEEE Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii, pp. 207-212, Dec. 2001.

[13] S. Wild, J. Curry, and A. Dougherty, "Improving Non-negative Matrix Factorizations through Structured Initialization," Pattern Recognition, vol. 37, pp. 2217-2232, 2004.

[14] P. O. Hoyer, "Non-Negative Sparse Coding," IEEE Workshop on Neural Networks for Signal Processing, Martigny, Valais, Switzerland, pp. 557-565, Sep. 2002.

[15] P. O. Hoyer, "Modeling Receptive Fields with Non-Negative Sparse Coding," Neurocomputing, vol. 52-54, pp. 547-552, 2003.

[16] C. M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, 1995, Chapter 9, pp. 332-384.

[17] B. A. Olshausen and D. J. Field, "Emergence of Simple-cell Receptive Field Properties by Learning a Sparse Code for Natural Images," Nature, vol. 381, pp. 607-609, 1996.

[18] CBCL Face Database #1, MIT Center For Biological and Computation Learning, http://www.ai.mit.edu/projects/cbcl, Training set: faces.

[19] F. Samaria and A. Harter, "Parameterisation of a Stochastic Model for Human Face Identification," 2nd IEEE Workshop on Applications of Computer Vision, Sarasota, Florida, Dec. 1994.

[20] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning, Springer, 2001, Chapter 3, pp. 59-72.