http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 9 (8): 1719-1724, 2010
ISSN 1812-5638

Online Face Recognition Algorithm via Nonnegative Matrix Factorization

[1,2]Hongli Yang and [1]Guoping He
[1]College of Information Science and Engineering,
Shandong University of Science and Technology, Qingdao, Shandong, 266510, China
[2]Department of Computer and Information Science,
Norwegian University of Science and Technology, Trondheim, 7491, Norway

Abstract: In this study, a novel algorithm is presented for dealing with the online data based on nonnegative matrix factorization model. Nonnegative Matrix Factorization (NMF) is a promising approach for face recognition in that it is capable of extracting the local features by factorizing the nonnegative matrix into two nonnegative matrices. However, there are two major weaknesses in almost all the existing NMF based methods. The first shortcoming is that the computational cost is high for large matrix factorization, it also needs more memory to save the huge data. The other is that it must conduct repetitive learning when the data samples are updated. To overcome these two limitations, a novel Online Nonnegative Matrix Factorization (ONMF) algorithm for online face recognition is presented in this study. The ONMF algorithm can not only deal with the incremental nonnegative matrix factorization, but also can deal with the decremental nonnegative matrix factorization which has never been considered in other study. Two face databases, namely ORL and FERET face database, are selected for evaluation. Compared with the conventional standard NMF, the method in this study gives the better performance both in computational costs and other aspects.

Key words: NMF, feature extraction, algorithm, incremental, decremental, online data

INTRODUCTION

Nonnegative Matrix Factorization (NMF) can extract local features by factorizing the nonnegative matrix into two nonnegative matrixes (Lee and Seung, 1999, 2000). It has been used for static data analysis and pattern recognition in the past. For the reason that the results of the factorization have good interpretable properties and also because of the sparseness of the factor matrixes, it has been widely used in many fields including face recognition and spectral data analysis (Guillamet and Vitria, 2002, 2003; Lee and Seung, 1999). However, for its low convergence speed and the huge memory demands for the online data processing in conventional NMF, it is impossible to apply the model to online data analysis. The purpose of the online NMF algorithm is to perform rapid NMF analysis so that the data in reality such as recommendation systems or online text mining can be produced in real time. In such applications, systems need to exclude participation of the old samples from their representations and also need to increase influence of the latest samples. For instances, in a real face recognition database, the new samples have to be added into the database and oppositely, the old samples will be excluded from the database.

Nonnegative matrix factorization was first used in face recognition to extract the local features in 1999 (Lee and Seung, 1999), it aims to perform nonnegative matrix decomposition on the training image matrix A such that $A \approx UV^T$, where, U, V are the base matrix and the coefficient matrixes, respectively. The local image features are learned and included in U as column vectors. In the latest years, several algorithms for dealing with the incremental learning were developed. In (Wen-Sheng *et al*., 2008; Pan *et al*., 2008), a supervised Incremental NMF (INMF) scheme was given base on the class of the new samples, with the orthogonal constraints, the author declares that their factors are unique and their algorithm is convergent , experiments given in their study show that their algorithm have a good performance, but the influence of the new samples only on the class while not the whole database which will affect the local features extraction. Bucak and Gunsel (2009), an incremental subspace learning method was proposed by a weighted cost function which allows controlling the memorylessness of the factorization and incrementally updating its factors by reflecting the influence of each observation, the method was tested on video data sequences features extraction. Cao *et al*. (2007), a new model was presented by adding some

Corresponding Author: Hongli Yang, College of Information Science and Engineering,
Shandong University of Science and Technology, Qingdao, Shandong, 266510, China

orthogonal constraint and sparseness constraints in the objective function, it also approximate the relation between the old base matrix and the new base matrix. Rebhan *et al.* (2008), an incremental learning algorithm was given, it can incrementally and continuously adapt to the new data. Several other new algorithms for the incremental NMF algorithm also are given such as by Li *et al.* (2010) and Bucak and Gunsel (2007). All the above algorithms have taken account into the incremental case of the online data while have not considered the decremental situations which often happens in reality. To the best of our knowledge, there is no studies available dealing the decremental case based on NMF.

In this study, we give an algorithm which will consider the two cases: incremental and decremental cases. The study is organized as follows: first, we derive the algorithms for the two cases and then analysis the computational complexity of the algorithm. After that, experiments are given based two face databases. The conclusions and the future work were given in the end.

PROPOSED ONMF

The basic model for nonnegative matrix factorization is to factorize the nonnegative matrix A into two nonnegative matrix factors U, V, where, the matrix U is the base matrix and V is the coefficient matrix. Basing on the Euclidean error distance, the NMF problem has the following optimization model:

$$\begin{aligned}&\min\left\|A-UV^{T}\right\|_{F}^{2}\\&\text{s.t.}\\&U\geq 0, V\geq 0\end{aligned}\tag{1}$$

The updates formula for standard NMF (Lee and Seung, 2000) is:

$$\begin{aligned}U_{ij}&=U_{ij}\cdot\frac{(AV)_{ij}}{(UV^{T}V)_{ij}}\\V_{ij}&=V_{ij}\cdot\frac{(A^{T}U)_{ij}}{(VU^{T}U)_{ij}}\end{aligned}\tag{2}$$

The factors are U*, V* and we have the approximations $A\approx U^{*}(V^{*})^{T}$.

Incremental learning: The basic NMF method has been proven to be an effective method for extracting the local features by seeking the nonnegative factorizations. However, the basic NMF method and its available existing variations assume that the data are static. In reality, we need considering the dynamic nature of the online data. In order to apply the NMF based method on online data, we have to repetitively factorize the data when it updates, this procedure is clearly time consuming. Thus, our direction is to aim at improving the basic NMF based method by developing an online version. From the point of the application in face recognition, the task of NMF is to find a set of nonnegative bases to represent the input data by a linear combination, when the new data arrives and the old data is deleted, the data base need to be updated to represent the data. So, we mainly consider the changes in the base matrix, this is the main idea of the online NMF method given in this study.

For the new incremental samples $\alpha\epsilon R^{m\times k}$, the coefficient based on the old bases is:

$$v=\left(U^{*}\right)^{+}a=\left(\left(U^{*}\right)^{T}U^{*}\right)^{-1}\left(U^{*}\right)^{T}a\tag{3}$$

In Eq. 3, the coefficient of the incremental samples is not necessarily nonnegative due to the generalized inverse of matrix U*, so in the iteration, we project the coefficient onto the nonnegative orthant when it is not nonnegative.

Assumptions: The column vectors of matrix U is linearly independent.

This assumption is modest in that the column vectors of the base vectors are often linear independent. For the incremental samples, we have the following model:

$$\min_{U_1,V_1\geq 0}\left\|\tilde{A}-U_1V_1^{T}\right\|_{F}^{2}\approx\min_{U,V_1\geq 0}\left\|(A,a)-UV_1^{T}\right\|_{F}^{2}\tag{4}$$

where, matrix A is the old data and a is the new samples, under the condition of assumption, the above model then is changed into the following form:

$$\begin{aligned}&\min_{U,V_1\geq 0}\left\|\tilde{A}-UV_1^{T}\right\|_{F}^{2}=\min_{U,V_1\geq 0}\left\|(A,a)-UV_1^{T}\right\|_{F}^{2}\\&=\min_{U,V\geq 0}\left\|\tilde{A}-U\begin{pmatrix}V\\ \mathbf{v}^{T}\end{pmatrix}^{T}\right\|_{F}^{2}=\min_{U,V\geq 0}\left\|\tilde{A}-U\begin{pmatrix}V\\ \left(\left(U^{*T}U^{*}\right)^{-1}U^{*T}a\right)^{T}\end{pmatrix}^{T}\right\|_{F}^{2}\\&=\min_{U,V\geq 0}\left\|(A,a)-U\left(V^{T},\left(U^{*T}U^{*}\right)^{-1}U^{*T}a\right)\right\|_{F}^{2}=\min_{U,V\geq 0}\left\|(A,a)-\left(UV^{T},U\left(U^{*T}U^{*}\right)^{-1}U^{*T}a\right)\right\|_{F}^{2}\\&=\min_{U,V\geq 0}\left\|\left(A-UV^{T},a-U\left(U^{*T}U^{*}\right)^{-1}U^{*T}a\right)\right\|_{F}^{2}\end{aligned}\tag{5}$$

Then, by using the method by Lee and Seung (1999), we can derive the iteration formula of the algorithm. The objective function of the above model is:

$$\begin{aligned} f(U,V) &= \left\| \left(A - UV^T, a - U(U^{*T}U^*)^{-1}U^{*T}a\right) \right\|_F^2 \\ &= \mathrm{trace}\left(\left(A - UV^T, a - U(U^{*T}U^*)^{-1}U^{*T}a\right)\left(A - UV^T, a - U(U^{*T}U^*)^{-1}U^{*T}a\right)^T \right) \end{aligned} \quad (6)$$

For simplicity, we can denote b = a-U(U*^{T}U*)$^{-1}$U*^{T}a and then we have:

$$\begin{aligned} f(U,V) &= \mathrm{trace}\left(\left(A - UV^T, b\right) \begin{pmatrix} \left(A - UV^T\right)^T \\ b^T \end{pmatrix} \right) \\ &= \mathrm{trace}\left(\left(A - UV^T\right)\left(A - UV^T\right)^T + bb^T \right) \\ &= \mathrm{trace}(AA^T - AVU^T - UV^TA^T + UV^TVU^T + bb^T) \\ &= \mathrm{trace}(AA^T) - 2\mathrm{trace}(AVU^T) + \mathrm{trace}(UV^TVU^T) + \mathrm{trace}(bb^T) \end{aligned} \quad (7)$$

The lagrangian function of the above model Eq. 5 is:

$$L(U,V) = f(U,V) - \mathrm{sum}(\mathrm{sum}(\lambda \otimes U)) - \mathrm{sum}(\mathrm{sum}(\mu \otimes V)) \quad (8)$$

The KKT conditions of the optimization problem Eq. 8 is:

$$\begin{aligned} &U \geq 0, && V \geq 0 \\ &\nabla_U L \geq 0, && \nabla_V L \geq 0 \\ &\lambda \geq 0, && \mu \geq 0 \\ &\lambda \otimes U = 0, && \mu \otimes V = 0 \end{aligned} \quad (9)$$

where, matrixes λ, μ have the similar dimensions as U,V, respectively. And the symbol ⊗ denotes elmentwise product of the two matrixes, sum( ) has the same mean with that in Matlab. Matrix λ,μ≥0 mean all the entries in the matrixes are nonnegative.

According to the KKT conditions, we have the following results:

$$\begin{aligned} &UV^TV - AV + U\left(U^*\right)^+ aa^T\left(U^{*+}\right)^T - a\left(U^{*+}a\right)^T - \frac{\lambda}{2} = 0 \\ &U^TUV^T - U^TA - \frac{\mu}{2} = \end{aligned} \quad (10)$$

Then, we have:

$$\begin{aligned} &\left(UV^TV - AV + UU^{*+}aa^T\left(U^{*+}\right)^T\right) \otimes U = 0 \\ &\left(U^TUV^T - U^TA\right) \otimes V^T = 0 \end{aligned} \quad (11)$$

So, the iterates of the algorithm is:

$$U_{ij} = U_{ij} \cdot \frac{(AV + a(U^{*+}a)^T)_{ij}}{(UV^TV + UU^{*+}aa^T(U^{*+})^T)_{ij}}$$

$$V_{ij} = V_{ij} \frac{(A^TU)_{ij}}{(VU^TU)_{ij}} \quad (12)$$

In order to avoid division by zero, we have a small positive value ε,then we have:

$$U_{ij} = U_{ij} \cdot \frac{(AV + a(U^{*+}a)^T)_{ij}}{(UV^TV + UU^{*+}aa^T(U^{*+})^T)_{ij} + \varepsilon}$$

$$V_{ij} = V_{ij} \frac{(A^TU)_{ij}}{(VU^TU)_{ij} + \varepsilon} \quad (13)$$

Decremental learning: Suppose matrix a are the old samples that will be excluded, first we rearrange the column of the samples matrix as:

$$\tilde{A} = AP = (A_1, a) \quad (14)$$

where, matrix P is the permutation matrix, then we have:

$$\tilde{A} = AP = (U^*P)(P^TV^{*T}P) \approx \tilde{U}\left(\tilde{V}\right)^T \quad (15)$$

Use the same philosophy as the incremental case, we mainly take account into the base matrix of the factorization, then we have:

$$\min_{U,V \geq 0} \left\| \tilde{A} - UV^T \right\|_F^2 \approx \min_{U,V \geq 0} \left\| (A_1, a) - U\begin{pmatrix} V_1 \\ (U^{*+}a)^T \end{pmatrix} \right\|_F^2 \quad (16)$$

where, V$_1$ is the new coefficient matrix and (V$_1$^{T}, U$^+$a) has the same size as $\tilde{V}$. With the same division as above, we have the iterations for the decremental case:

$$U_{ij} = U_{ij} \cdot \frac{(A_1V_1 + a(U^{*+}a)^T)_{ij}}{(UV_1^TV_1 + UU^{*+}aa^T(U^{*+})^T)_{ij}}$$

$$\left(V_1\right)_{ij} = \left(V_1\right)_{ij} \cdot \frac{(U^TA_1)_{ij}}{(U^TUV_1^T)_{ij}} \quad (17)$$

In order to avoid division by zero, like what we have done for Eq. 11 , we have a small positive value ε, then we have:

$$U_{ij} = U_{ij} \cdot \frac{(A_1V_1 + a(U^{*+}a)^T)_{ij}}{(UV_1^TV_1 + UU^{*+}aa^T(U^{*+})^T)_{ij} + \varepsilon}$$

$$\left(V_1\right)_{ij} = \left(V_1\right)_{ij} \cdot \frac{(U^TA_1)_{ij}}{(U^TUV_1^T)_{ij} + \varepsilon} \quad (18)$$

The online NMF algorithm:

- Step 1 input the initializations U$^{(0)}$,V$^{(0)}$ the rank r≤min(m,n), ε and the termination criterion for the algorithm, carry out the following iterations until convergent

- Step 2 update U according to the iterates formula Eq. 13 or 18:

$$U_{ij} = \frac{U_{ij}}{\sqrt{\sum_{i=1}^{n} U_{ij}^2}}, \text{ for } i = 1, 2, \cdots, m; j = 1, 2, \cdots, k$$

- Step 3 update V according to the iterates formula Eq. 13 or 18

Extra computational complexity of the above algorithm: Compared with the conventional standard NMF updates formula, there are more operations needed to be carried out in advance. In the iterations, we mainly take account into the multiplication and division operations. Generally, for a matrix with k×k size, the operations for the calculations of the inverse matrix are $O(k^3)$ based on the Gaussian Elimination methods. Suppose the number of the updating data (incremental/decremental) is k, which means $a \in R^{m \times k}$, the base matrix $U \in R^{m \times r}$, the extra operations of the ONMF algorithms in an iteration compared with the conventional NMF are:

- $U^{*T}U$ has nr^2 times operations
- $(U^{*T}U^*)^{-1}$ has $O(r^3)$ times operations;
- The production of $(U^{*T}U)^{-1}U^T$ has mr^2 times operations and the total operations for the calculation of $(U^*)^+$ are $O(r^3) + (m+n)r^2$
- The operations for the $(a(U^{*+}a)^T)_{ij}$ is m^2r+mr
- the operations for the $UU^{*+}aa^T (U^{*+})^T$ is $m^2r +mr$

In the real experiments, we can calculate V^TV and after that we calculate $U(V^TV)$ which will save the computational costs because the former operations are O(mnr) and the operations for latter are $O(\max(m,n) \cdot r^2)$.

EXPERIMENTAL RESULTS

Here, Feret and ORL face databases are selected to evaluate the performance of our ONMF method along with conventional NMF method. All the images in the two databases are 112×92 size. The original images are reduced to 28×23 size for the ORL database and 20×20 size for the Feret database by the DCT transformation. We stop the iteration if the stopping condition is met or if it exceeds 1000 times iterations. The standard NMF means the NMF iterations formula used by Lee and Seung (2000), the results with the standard NMF means the results obtained by using the standard NMF algorithm on the samples with incremental/decremental case.

Incremental learning

Experiment 1: The database is feret face recognition database, we evaluate r = 20, ε = 1e-006, the number of the samples is 300 and the number of incremental sample is 30. We initialize all the algorithms randomly. The factorization results is in Table 1.

Experiment 2: The database is ORL face recognition database, we evaluate r = 20, ε = 1e-006, the number of the samples is 300 and the number of incremental sample is 30. We initialize all the algorithms randomly. The factorization results is in Table 2.

Decremental learning

Experiment 1: The database is feret face database, we evaluate r = 20, ε = 1e-006, the number of the samples is 300 and the number of decremental sample is 30. We initialize all the algorithms randomly. The factorization results is in Table 3.

Experiment 2: The database id ORL face database, we evaluate r = 20, ε = 1e-006, the number of the samples is

Table 1: Numerical results of incremental NMF for feret database

Iterate No.	Error for standard NMF(1e+004)	Time cost (sec)	Error for ONMF (1e+004)	Time cost (sec)
5	5.1787	0.14	2.245	0.125
10	5.0003	0.187	2.2437	0.219
15	4.8641	0.234	2.2431	0.25
20	4.6616	0.312	2.2427	0.312
25	4.2335	0.406	2.2425	0.405
30	3.9604	0.499	2.2423	0.468
50	3.3552	0.733	2.2417	0.749
80	2.879	1.186	2.241	1.124
100	2.7341	1.467	2.2406	1.435
200	2.4175	2.902	2.239	2.839
300	2.3497	4.274	2.2378	4.197
400	2.3186	5.678	2.2364	5.632
500	2.2975	7.114	2.2352	6.973
1000	2.2358	14.165	2.2314	14.118

Table 2: Numerical results of incremental NMF for ORL database

Iterate No.	Error for standard NMF(1e+004)	Time cost (sec)	Error for ONMF (1e+004)	Time cost (sec)
5	2.2821	0.188	1.1745	0.156
10	2.2364	0.281	1.1729	0.265
15	2.1353	0.39	1.1722	0.375
20	2.0508	0.484	1.1717	0.50
25	1.9071	0.577	1.1713	0.593
30	1.8062	0.718	1.1711	0.717
50	1.502	1.186	1.1703	1.17
80	1.3546	1.81	1.1694	1.825
100	1.3109	2.262	1.169	2.23
200	1.2139	4.431	1.1675	4.431
300	1.192	6.693	1.1664	6.584
400	1.1821	8.814	1.1657	8.83
500	1.1741	11.044	1.165	10.952
1000	1.1654	22.074	1.1629	22.183

Table 3: Numerical results of decremental NMF for feret database

Iterate No.	Error for standard NMF(1e+004)	Time cost (sec)	Error for ONMF (1e+004)	Time cost (sec)
5	4.7906	0.125	2.0364	0.093
10	4.6622	0.125	2.0362	0.14
15	4.4593	0.218	2.036	0.203
20	4.2198	0.234	2.0358	0.296
25	3.8056	0.343	2.0356	0.359
30	3.5399	0.406	2.0354	0.437
50	3.0031	0.624	2.0346	0.702
80	2.6399	0.95	2.0335	1.045
100	2.4492	1.186	2.0328	1.326
200	2.1403	2.355	2.0295	2.496
300	2.0919	3.479	2.027	3.744
400	2.0837	4.648	2.0249	5.038
500	2.0461	5.803	2.0231	6.271
1000	2.0241	11.544	2.017	12.48

Table 4: Numerical results of decremental NMF for ORL database

Iterate No.	Error for standard NMF(1e+004)	Time cost (sec)	Error for ONMF (1e+004)	Time cost (sec)
5	2.0514	0.156	1.0265	0.125
10	1.9969	0.249	1.0264	0.203
15	1.9268	0.328	1.0264	0.359
20	1.7923	0.421	1.0264	0.452
25	1.6921	0.531	1.0263	0.577
30	1.5581	0.624	1.0262	0.64
50	1.3452	0.951	1.026	1.029
80	1.2164	1.513	1.0257	1.591
100	1.1352	1.857	1.0255	2.044
200	1.0576	3.666	1.0245	4.009
300	1.0451	5.507	1.0237	6.037
400	1.0332	7.363	1.023	7.94
500	1.0198	9.141	1.0223	9.922
1000	1.0091	18.314	1.02	19.795

300 and the number of decremental sample is 30. We initialize all the algorithms randomly. The factorization results is in Table 4.

The sparseness of the base matrix U: For a vector $u \in R^n$, we define the sparseness as that by Hoyer (2004),

$$\mathrm{Sparseness}(u) = \frac{\sqrt{n} - \|u\|_1 / \|u\|_2}{\sqrt{n} - 1}$$

The sparseness of the base matrix is defined as:

$$\mathrm{Sparseness}(U) = \frac{\sum_{i=1}^{n} \mathrm{sparseness}(u_i)}{n}$$

The left side iteration number is the number for Incremental NMF and decremental NMF, the right side of the iterations number is the number for the standard NMF. The sparseness is for the base matrix of the factors.

Table 5: The sparseness of the base matrix

	Incremental NMF		Decremental NMF		Standard NMF		
Iteration No.	ORL	Feret	ORL	Feret	ORL	Feret	Iterations No.
5	0.4012	0.4204	0.3823	0.4211	0.3322	0.3712	200
15	0.4048	0.4274	0.3825	0.4223	0.3497	0.3623	400
20	0.4061	0.4292	0.3833	0.4227	0.3638	0.3879	500

The experimental results are given from Table 1-5. From the experimental results, we can find that the ONMF algorithm can approximate the local minimum with less iteration than the conventional standard NMF algorithm. The time costs are a little more than the conventional NMF with the same iterations but our algorithm can achieve the local minimum in several iterations while the conventional NMF can't. With the same iterations for the two algorithms, the error of the objective functions for our algorithm is less. The sparseness of the base matrix of ONMF algorithm is higher than the conventional NMF, which means that it can extract the local features successfully. So, the ONMF algorithm has good performance compared with the conventional algorithm.

CONCLUSIONS AND FUTURE WORK

A novel ONMF algorithm for face recognition based on online updating is proposed in this study. Compared with the conventional NMF based methods, our algorithm have the following properties: (1) It only need storing the base matrix of the former factorization without storing the original data matrix ,which will save a huge storage space; (2) It can deal with not only the incremental online data, but also the decremental case, which have not been considered in other studies. (3) The algorithm have a better convergence speed, it can convergent to the approximate local minimum with fewer iterations. Experimental results on ORL and FERET face database show that the ONMF algorithm has good performance.

The future research is the online update scheme under different error distance criterion such as KL divergence. We also plan to test the effectiveness of our approach on more data sets and more fields such as text mining, image processing, video surveillance and so on.

ACKNOWLEDGMENTS

This study is partly supported by National Nature Science Foundation of China (NNSFC) under the grant No. 10571109; the Nature Science Foundation of Shandong Province under grant No. Y2008A01; the Spring Bud project 2009 of SDUST.

REFERENCES

Bucak, S.S. and B. Gunsel, 2007. Video content representation by incremental nonnegative matrix factorization. IEEE Int. Conf. Image Process., 2: 113-116.

Bucak, S.S. and B. Gunsel, 2009. Incremental subspace learning via non-negative matrix factorization. Pattern Recognition, 42: 788-797.

Cao, B., D. Shen, J.T. Sun, X. Wang, Q. Yang and Z. Chen, 2007. Detect and track latent factors with online nonnegative matrix factorization. Proceedings of the 20th International Joint Conference on Artificial Intelligence, Jan. 6-12, Hyderabad, India, pp: 2689-2694.

Guillamet, D. and J. Vitria, 2002. Non-negative Matrix Factorization for Face Recognition. In: Lecture Notes in Computer Science, Escrig-Monferrer, M.T., F. Toledo and E. Golobardes (Eds.). Springer, Berlin, Heidelberg, ISBN: 978-3-540-00011-2. pp: 336-344.

Guillamet, D. and J. Vitria, 2003. Evaluation of distance metrics for recognition based on non-negative matrix factorization. Pattern Recognition Lett., 4: 1599-1605.

Hoyer, P.O., 2004. Nonnegative matrix factorization with sparseness constraints. J. Machine Learning Res., 5: 1457-1469.

Lee, D. and H. Seung, 1999. Learning the parts of objects by non-negative matrix factorization. Nature, 401: 788-791.

Lee, D.D. and H.S. Seung, 2000. Algorithms for nonnegative matrix factorization. Proceedings of the Neural Information Systems, (NIS'00), MIT Press, pp: 556-562.

Li, G., S. Zhang, W. Wang and B. Shi, 2010. Incremental nonnegative matrix factorization algorithm. Comput. Eng., 36: 66-68.

Pan, B.B., W.S. Chen and C. Xu, 2008. Incremental learning of face recognition based on block nonnegative matrix factorization. Comput. Appl. Res., 26: 117-120.

Rebhan, S., W. Sharif and J. Eggert, 2008. Incremental learning in nonnegative matrix factorization. Proceedings of the 15th International Conference, ICONIP, Nov. 25-28, Auckland, New Zealand, USA., pp: 960-969.

Wen-Sheng, C., B. Pan, B. Fang, M. Li and J. Tang, 2008. Incremental nonnegative matrix factorization for face recognition. Math. Problems Eng., 2008: 17-17.