

Convergent Projective Non-negative Matrix Factorization

Lirui Hu^{1,2,3}, Jianguo Wu^{1,3} and Lei Wang^{1,3}

¹ Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, Anhui University
Hefei, 230039, China

² School of Computer Science and Technology, Nantong University
Nantong, 226019, China

³ School of Computer Science and Technology, Anhui University
Hefei, 230039, China

Abstract

In order to solve the problem of algorithm convergence in projective non-negative matrix factorization (P-NMF), a method, called convergent projective non-negative matrix factorization (CP-NMF), is proposed. In CP-NMF, an objective function of Frobenius norm is defined. The Taylor series expansion and the Newton iteration formula of solving root are used. An iterative algorithm for basis matrix is derived, and a proof of algorithm convergence is provided. Experimental results show that the convergence speed of the algorithm is higher, however it is affected by the initial value of the basis matrix; relative to non-negative matrix factorization (NMF), the orthogonality and the sparseness of the basis matrix are better, however the reconstructed results of data show that the basis matrix is still approximately orthogonal; in face recognition, there is higher recognition accuracy. The method for CP-NMF is effective.

Keywords: Non-negative Matrix Factorization, Projective, Convergence, Face Recognition.

1. Introduction

According to the point of view which perception of the whole is based on perception of its parts, a data technology, called non-negative matrix factorization (NMF) $X \approx WH$ [1], was constructed. The method had revealed the essence of describing data, and it had been widely applied to the fields of data dimension reduction, image analysis, pattern recognition [1, 2], text mining, spectral data analysis [3], and so on. NMF is a current research focus.

Projective non-negative matrix factorization (P-NMF) $X \approx WW^T X$ [4] was proposed based on NMF. Since it was constructed from the projection angle, the basis matrix W was only computed in the algorithm for P-NMF. The computational complexity was lower for one iteration step for P-NMF, as only one matrix had to be computed instead of two for NMF. On the basis of optimization rule

$\arg \min_{W \geq 0} \frac{1}{2} \|X - WW^T X\|_F^2$, the basis matrix W for P-

NMF was forced to tend to be orthogonal. So, the orthogonality and the sparseness of the basis matrix were better in P-NMF than in NMF, and then the method for P-NMF was more beneficial to the applications of data dimension reduction, pattern recognition, and so on.

However, the proof of algorithm convergence was not given in the paper [4]. Now, we use the objective function

$$F = \frac{1}{2} \|X - WW^T X\|_F^2 \quad (W \geq 0, X \geq 0) \quad \text{and} \quad \text{the}$$

iterative formula

$$W_{ij} = W_{ij} \frac{2(XX^T W)_{ij}}{(WW^T XX^T W)_{ij} + (XX^T WW^T W)_{ij}} \quad (1)$$

in the paper [4] to do an experiment. In Eq. (1), X consists of the first five images of each person in the ORL facial image database, a total of 200 data. We set the rank of the basis matrix W 80 and initialize it with non-negative data. In order to reduce the amount of computation and improve the speed of operation, each image is reduced to a quarter of the original. After 10000 iteration steps, we will see that the varied curve of the objective function values versus iteration steps is severely concussive, and the algorithm does not converge. Here, in order to make the graphics clearly seen, we give the varied curve of objective function values versus iteration steps after 100 iteration steps, and the curve is shown in Fig. 1.

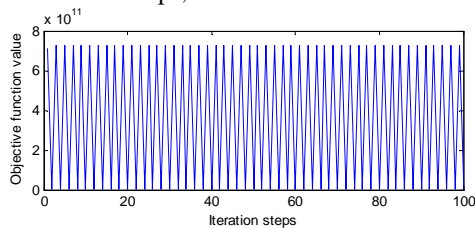


Fig. 1 Objective function values versus iteration steps

In order to solve the problem of algorithm convergence in P-NMF, a method is proposed based on P-NMF $X \approx WW^T X$ in this paper. We call it convergent projective non-negative matrix factorization (CP-NMF). In this method, another iterative algorithm for the basis matrix W is constructed, and strict proof of algorithm convergence is provided, and the convergence speed of the algorithm is higher. Like P-NMF, the orthogonality and the sparseness of the basis matrix are still better in CP-NMF. We compare this method with the methods of NMF, LNMF [5], and NMFOs [6], and the experimental results show that this method has higher recognition accuracy in face recognition.

The rest of this paper is organized as follows. In Section 2, our method is introduced in detail. Firstly, an iterative formula for basis matrix W is derived strictly based on $X \approx WW^T X$. Secondly, a proof of algorithm convergence is provided. Finally, the algorithm steps are given. In Section 3, the convergence of the algorithm is validated by numerical experiments, and it is emphasized that the convergence speed of the algorithm is affected by the initial value of the basis matrix W and the basis matrix W is still approximately orthogonal. Moreover, by numerical experiments in face recognition, we compare this method with NMF and some extended methods in Section 4, and explain the effectiveness of the method. In the end, conclusions are drawn in Section 5.

2. Convergent Projective Non-negative Matrix Factorization (CP-NMF)

We consider an objective function [4]

$$F = \frac{1}{2} \|X - WW^T X\|_F^2 \quad (2)$$

where $X \geq 0, W \geq 0$. Obviously, F is a function defined in P-NMF. We may minimize F to get W .

2.1 The Iterative Rule for Basis Matrix W

For any element w_{ab} of W , let $F_{w_{ab}}$ stand for the part of F relevant to w_{ab} in Eq. (2). So, writing w instead of w_{ab} in the expression of $F_{w_{ab}}$, we may get a function $F_{w_{ab}}(w)$. Obviously, the first order derivative of $F_{w_{ab}}(w)$ at w_{ab} is the first order partial derivative of F with respect to w_{ab} . That is

$$\begin{aligned} F'_{w_{ab}}(w_{ab}) &= \frac{\partial F}{\partial w_{ab}} = \frac{\partial(\frac{1}{2} \sum_{ij} [X_{ij} - (WW^T X)_{ij}]^2)}{\partial w_{ab}} \\ &= \sum_{ij} -(X_{ij} - (WW^T X)_{ij}) \frac{\partial(WW^T X)_{ij}}{\partial w_{ab}} \\ &= \sum_{ij} (-X_{ij} + (WW^T X)_{ij}) \frac{\partial(\sum_k (WW^T)_{ik} X_{kj})}{\partial w_{ab}} \\ &= \sum_{ij} (-X_{ij} + (WW^T X)_{ij}) (\sum_k \frac{\partial(WW^T)_{ik}}{\partial w_{ab}} X_{kj}) \\ &= \sum_{ij} (-X_{ij} + (WW^T X)_{ij}) (\sum_k \frac{\partial(\sum_l W_{il} W_{kl})}{\partial w_{ab}} X_{kj}) \\ &= \sum_{ij} (-X_{ij} + (WW^T X)_{ij}) (\sum_k \frac{\partial(W_{ib} W_{kb})}{\partial w_{ab}} X_{kj}) \\ &= \sum_j \sum_i (-X_{ij} + (WW^T X)_{ij}) (\sum_k \frac{\partial(W_{ib} W_{kb})}{\partial w_{ab}} X_{kj}) \\ &= \sum_j [(-X_{aj} + (WW^T X)_{aj}) (\sum_k \frac{\partial(W_{ab} W_{kb})}{\partial w_{ab}} X_{kj}) + \\ &\quad \sum_{i \neq a} (-X_{ij} + (WW^T X)_{ij}) (\sum_k \frac{\partial(W_{ib} W_{kb})}{\partial w_{ab}} X_{kj})] \\ &= \sum_j [(-X_{aj} + (WW^T X)_{aj}) (\sum_{k \neq a} \frac{\partial(W_{ab} W_{kb})}{\partial w_{ab}} X_{kj}) + \\ &\quad \frac{\partial(W_{ab} W_{ab})}{\partial w_{ab}} X_{aj}) + \sum_{i \neq a} (-X_{ij} + (WW^T X)_{ij}) W_{ib} X_{aj}] \\ &= \sum_j [(-X_{aj} + (WW^T X)_{aj}) (\sum_{k \neq a} W_{kb} X_{kj} + 2W_{ab} X_{aj}) + \\ &\quad \sum_i (-X_{ij} + (WW^T X)_{ij}) W_{ib} X_{aj} - \\ &\quad (-X_{aj} + (WW^T X)_{aj}) W_{ab} X_{aj}] \\ &= \sum_j [(-X_{aj} + (WW^T X)_{aj}) (\sum_{k \neq a} W_{kb} X_{kj} + W_{ab} X_{aj}) + \\ &\quad \sum_i (-X_{ij} + (WW^T X)_{ij}) W_{ib} X_{aj}] \\ &= \sum_j [(-X_{aj} + (WW^T X)_{aj}) (W^T X)_{bj} + \\ &\quad \sum_i (-X_{ij} W_{ib} X_{aj}) + \sum_i (WW^T X)_{ij} W_{ib} X_{aj}] \\ &= -\sum_j X_{aj} (W^T X)_{bj} + \sum_j (WW^T X)_{aj} (W^T X)_{bj} \end{aligned}$$

$$\begin{aligned}
 & -\sum_j X_{aj}(W^T X)_{bj} + \sum_j X_{aj}(W^T WW^T X)_{bj} \\
 = & -(XX^T W)_{ab} + (WW^T XX^T W)_{ab} - \\
 & (XX^T W)_{ab} + (XX^T WW^T W)_{ab} \\
 = & -2(XX^T W)_{ab} + (WW^T XX^T W)_{ab} + \\
 & (XX^T WW^T W)_{ab}. \quad (3)
 \end{aligned}$$

Similarly, in order to get the second order derivative of $F_{w_{ab}}(w)$ at w_{ab} , we can get

$$\begin{aligned}
 \frac{\partial(-2(XX^T W)_{ab})}{\partial w_{ab}} &= -2(XX^T)_{aa}, \\
 \frac{\partial(WW^T XX^T W)_{ab}}{\partial w_{ab}} &= (WW^T XX^T)_{aa} + \\
 & (W^T XX^T W)_{bb} + (XX^T W)_{ab} W_{ab},
 \end{aligned}$$

and

$$\begin{aligned}
 \frac{\partial(XX^T WW^T W)_{ab}}{\partial w_{ab}} &= (WW^T XX^T)_{aa} + \\
 & (XX^T W)_{ab} W_{ab} + (XX^T)_{aa} \sum_k W_{kb}^2.
 \end{aligned}$$

So, the second order derivative of $F_{w_{ab}}(w)$ at w_{ab} is

$$\begin{aligned}
 F_{w_{ab}}''(w_{ab}) &= \frac{\partial(-2(XX^T W)_{ab})}{\partial w_{ab}} + \frac{\partial(WW^T XX^T W)_{ab}}{\partial w_{ab}} \\
 & + \frac{\partial(XX^T WW^T W)_{ab}}{\partial w_{ab}} \\
 = & -2(XX^T)_{aa} + 2(WW^T XX^T)_{aa} + (W^T XX^T W)_{bb} \\
 & + 2(XX^T W)_{ab} W_{ab} + (XX^T)_{aa} \sum_k W_{kb}^2. \quad (4)
 \end{aligned}$$

Similarly, in order to get the third order derivative of $F_{w_{ab}}(w)$ at w_{ab} , we can get

$$\begin{aligned}
 \frac{\partial(XX^T)_{aa}}{\partial w_{ab}} &= 0, \\
 \frac{\partial(WW^T XX^T)_{aa}}{\partial w_{ab}} &= (XX^T)_{aa} W_{ab} + (XX^T W)_{ab}, \\
 \frac{\partial(W^T XX^T W)_{bb}}{\partial w_{ab}} &= 2(XX^T W)_{ab}, \\
 \frac{\partial(XX^T W)_{ab} W_{ab}}{\partial w_{ab}} &= (XX^T W)_{ab} + (XX^T)_{aa} W_{ab},
 \end{aligned}$$

and

$$\frac{\partial((XX^T)_{aa} \sum_k W_{kb}^2)}{\partial w_{ab}} = 2(XX^T)_{aa} W_{ab}.$$

So, the third order derivative of $F_{w_{ab}}(w)$ at w_{ab} is

$$F_{w_{ab}}'''(w_{ab}) = 6(XX^T W)_{ab} + 6(XX^T)_{aa} W_{ab}. \quad (5)$$

Similarly, in order to get the fourth order derivative of $F_{w_{ab}}(w)$ at w_{ab} , we can get

$$\frac{\partial(XX^T W)_{ab}}{\partial w_{ab}} = \frac{\partial(\sum_k (XX^T)_{ak} W_{kb})}{\partial w_{ab}} = (XX^T)_{aa}$$

and

$$\frac{\partial((XX^T)_{aa} W_{ab})}{\partial w_{ab}} = (XX^T)_{aa}.$$

So, the fourth order derivative of $F_{w_{ab}}(w)$ at w_{ab} is

$$F_{w_{ab}}^{(4)}(w_{ab}) = 12(XX^T)_{aa}, \quad (6)$$

and other order derivatives of $F_{w_{ab}}(w)$ with respect to w are

$$F_{w_{ab}}^{(n)}(w) = 0 \quad (7)$$

where $n \geq 5$.

Thus, the Taylor series expansion of $F_{w_{ab}}(w)$ at w_{ab} is

$$\begin{aligned}
 F_{w_{ab}}(w) &= F_{w_{ab}}(w_{ab}) + F'_{w_{ab}}(w_{ab})(w - w_{ab}) + \\
 & \frac{1}{2} F''_{w_{ab}}(w_{ab})(w - w_{ab})^2 + \frac{1}{6} F'''_{w_{ab}}(w_{ab})(w - w_{ab})^3 + \\
 & \frac{1}{24} F_{w_{ab}}^{(4)}(w_{ab})(w - w_{ab})^4. \quad (8)
 \end{aligned}$$

Meantime, to emphasize time of w_{ab} in numerical calculation, we write $w_{ab}^{(t)}$ instead of w_{ab} in the brackets of $F_{w_{ab}}(w)$. So, equation

$$\begin{aligned}
 F_{w_{ab}}(w) &= F_{w_{ab}}(w_{ab}^{(t)}) + F'_{w_{ab}}(w_{ab}^{(t)})(w - w_{ab}^{(t)}) + \\
 & \frac{1}{2} F''_{w_{ab}}(w_{ab}^{(t)})(w - w_{ab}^{(t)})^2 + \frac{1}{6} F'''_{w_{ab}}(w_{ab}^{(t)})(w - w_{ab}^{(t)})^3 \\
 & + \frac{1}{24} F_{w_{ab}}^{(4)}(w_{ab}^{(t)})(w - w_{ab}^{(t)})^4 \quad (9)
 \end{aligned}$$

is gotten from Eq. (8).

Now, we define a function

$$\begin{aligned}
 G_{w_{ab}}(w, w_{ab}^{(t)}) &= F_{w_{ab}}(w_{ab}^{(t)}) + F'_{w_{ab}}(w_{ab}^{(t)})(w - w_{ab}^{(t)}) + \\
 &\frac{1}{2} \left[\frac{(WW^T XX^T W)_{ab} + (XX^T WW^T W)_{ab}}{w_{ab}^{(t)}} + \right. \\
 &\quad \left. + \frac{(W^T XX^T W)_{bb} W_{ab} + 2(XX^T W)_{ab} W_{ab}^2}{w_{ab}^{(t)}} + \right. \\
 &\quad \left. \frac{((XX^T)_{aa} \sum_k W_{kb}^2) W_{ab}}{w_{ab}^{(t)}} \right] (w - w_{ab}^{(t)})^2 + \\
 &\frac{1}{6} F'''_{w_{ab}}(w_{ab}^{(t)})(w - w_{ab}^{(t)})^3 + \frac{1}{24} F^{(4)}_{w_{ab}}(w_{ab}^{(t)})(w - w_{ab}^{(t)})^4.
 \end{aligned} \tag{10}$$

Theorem 1. $G_{w_{ab}}(w, w_{ab}^{(t)})$ is an auxiliary function for $F_{w_{ab}}(w)$.

Proof: $G_{w_{ab}}(w, w_{ab}^{(t)}) = F_{w_{ab}}(w)$ is obvious when $w_{ab}^{(t)} = w$. We need show that $G_{w_{ab}}(w, w_{ab}^{(t)}) \geq F_{w_{ab}}(w)$ when $w_{ab}^{(t)} \neq w$.

Because $W \geq 0, X \geq 0$,

$$\begin{aligned}
 (WW^T XX^T W)_{ab} &= \sum_k (WW^T XX^T)_{ak} W_{kb}^{(t)} \\
 &\geq (WW^T XX^T)_{aa} W_{ab}^{(t)}
 \end{aligned}$$

and

$$\begin{aligned}
 (XX^T WW^T W)_{ab} &= \sum_k (XX^T WW^T)_{ak} W_{kb}^{(t)} \\
 &\geq (XX^T WW^T)_{aa} W_{ab}^{(t)} \\
 &= (WW^T XX^T)_{aa} W_{ab}^{(t)}.
 \end{aligned}$$

So,

$$\begin{aligned}
 (WW^T XX^T W)_{ab} + (XX^T WW^T W)_{ab} \\
 \geq 2(WW^T XX^T)_{aa} W_{ab}^{(t)}.
 \end{aligned}$$

When $W_{ab}^{(t)} > 0$,

$$\begin{aligned}
 \frac{(WW^T XX^T W)_{ab} + (XX^T WW^T W)_{ab}}{W_{ab}^{(t)}} \\
 \geq 2(WW^T XX^T)_{aa}.
 \end{aligned}$$

In fact, $W_{ab} = W_{ab}^{(t)} = w_{ab}$. So

$$\begin{aligned}
 \frac{(WW^T XX^T W)_{ab} + (XX^T WW^T W)_{ab}}{w_{ab}} + \\
 \frac{(W^T XX^T W)_{bb} W_{ab} + 2(XX^T W)_{ab} W_{ab}^2}{w_{ab}} +
 \end{aligned}$$

$$\begin{aligned}
 &\frac{((XX^T)_{aa} \sum_k W_{kb}^2) W_{ab}}{w_{ab}^{(t)}} \\
 &\geq -2(XX^T)_{aa} + 2(WW^T XX^T)_{aa} + (W^T XX^T W)_{bb} \\
 &\quad + 2(XX^T W)_{ab} W_{ab} + (XX^T)_{aa} \sum_k W_{kb}^2 \\
 &= F''_{w_{ab}}(w_{ab}^{(t)}),
 \end{aligned}$$

and then

$$G_{w_{ab}}(w, w_{ab}^{(t)}) \geq F_{w_{ab}}(w).$$

Thus, $G_{w_{ab}}(w, w_{ab}^{(t)})$ is an auxiliary function for $F_{w_{ab}}(w)$ according to the definition 1 of reference [7].

Theorem 2. $F_{w_{ab}}(w)$ is nonincreasing under the update

$$w_{ab}^{(t+1)} = \arg \min_w G_{w_{ab}}(w, w_{ab}^{(t)})$$

Proof: Because

$$\begin{aligned}
 F_{w_{ab}}(w_{ab}^{(t+1)}) &\leq G_{w_{ab}}(w_{ab}^{(t+1)}, w_{ab}^{(t)}) \leq G_{w_{ab}}(w_{ab}^{(t)}, w_{ab}^{(t)}) \\
 &= F_{w_{ab}}(w_{ab}^{(t)}),
 \end{aligned}$$

$F_{w_{ab}}(w)$ is nonincreasing.

Using the definition of auxiliary function and Theorem 2, we can get the local minimum of $F_{w_{ab}}(w)$ if only the

local minimum of $G_{w_{ab}}(w, w_{ab}^{(t)})$ is gotten. To get a local minimum of $F_{w_{ab}}(w)$, we may calculate the first order derivative of $G_{w_{ab}}(w, w_{ab}^{(t)})$ with respect to w , and have

$$\begin{aligned}
 G'_{w_{ab}}(w, w_{ab}^{(t)}) &= F'_{w_{ab}}(w_{ab}^{(t)}) + \\
 &\left[\frac{(WW^T XX^T W)_{ab} + (XX^T WW^T W)_{ab}}{w_{ab}^{(t)}} + \right. \\
 &\quad \left. \frac{(W^T XX^T W)_{bb} W_{ab} + 2(XX^T W)_{ab} W_{ab}^2}{w_{ab}^{(t)}} + \right. \\
 &\quad \left. \frac{((XX^T)_{aa} \sum_k W_{kb}^2) W_{ab}}{w_{ab}^{(t)}} \right] (w - w_{ab}^{(t)}) +
 \end{aligned}$$

$$\begin{aligned}
 &\frac{1}{6} F'''_{w_{ab}}(w_{ab}^{(t)})(w - w_{ab}^{(t)})^2 + \frac{1}{6} F^{(4)}_{w_{ab}}(w_{ab}^{(t)})(w - w_{ab}^{(t)})^3.
 \end{aligned} \tag{11}$$

In order to get the root of the equation

$$G'_{w_{ab}}(w, w_{ab}^{(t)}) = 0, \tag{12}$$

we have known that the function $G'_{w_{ab}}(w, w_{ab}^{(t)})$ is a Taylor series expansion with respect to w from Eq. (11), and then may use the Newton iteration formula of solving root to get the root of Eq. (12). That is

$$w = w_{ab}^{(t)} - \frac{G'_{w_{ab}}(w_{ab}^{(t)}, w_{ab}^{(t)})}{G''_{w_{ab}}(w_{ab}^{(t)}, w_{ab}^{(t)})}. \quad (13)$$

In Eq. (13),

$$G'_{w_{ab}}(w_{ab}^{(t)}, w_{ab}^{(t)}) = F'_{w_{ab}}(w_{ab}^{(t)})$$

and

$$G''_{w_{ab}}(w_{ab}^{(t)}, w_{ab}^{(t)}) = \frac{(WW^T XX^T W)_{ab}}{w_{ab}^{(t)}} + \frac{(XX^T WW^T W)_{ab} + (W^T XX^T W)_{bb} W_{ab}}{w_{ab}^{(t)}} + \frac{2(XX^T W)_{ab} W_{ab}^2 + ((XX^T)_{aa} \sum_k W_{kb}^2) W_{ab}}{w_{ab}^{(t)}}.$$

Using Eq. (3), we simplify the Eq. (13) to

$$w = \frac{M}{D} \quad (14)$$

where

$$M = [2(XX^T W)_{ab} + (W^T XX^T W)_{bb} W_{ab} + 2(XX^T W)_{ab} W_{ab}^2 + ((XX^T)_{aa} \sum_k W_{kb}^2) W_{ab}] w_{ab}^{(t)} \quad (15)$$

and

$$D = (WW^T XX^T W)_{ab} + (XX^T WW^T W)_{ab} + (W^T XX^T W)_{bb} W_{ab} + 2(XX^T W)_{ab} W_{ab}^2 + ((XX^T)_{aa} \sum_k W_{kb}^2) W_{ab}. \quad (16)$$

So, the iterative rule of w_{ab} is

$$w_{ab}^{(t+1)} = \frac{M}{D}. \quad (17)$$

Because the Newton iteration formula of solving root is convergent, we may use this iterative rule Eq. (17) and make the auxiliary function $G_{w_{ab}}(w, w_{ab}^{(t)})$ local minimum, and thus make the objective function $F_{w_{ab}}(w)$ local minimum. If all elements of W are updated by Eq. (17), the local minimum of the objective function F may be gotten. Therefore, the algorithm converges.

The Eq. (17) is the iterative update rule for the basis matrix W .

2.2 Algorithm Steps

Using Eq. (17), we may get an algorithm to compute the basis matrix W . As follows:

- Step1:** initialize W and X with non-negative data;
- Step2:** update W by Eq. (15), Eq. (16) and Eq. (17);
- Step3:** repeat step2 until algorithm converges.

W_n and W_{n+1} are respectively used to denote the n th and $n+1$ th iterative result of the basis matrix W . The condition of algorithm convergence is that there is

$$\|W_{n+1} - W_n\|_F^2 < \varepsilon, \forall \varepsilon > 0 \quad (18)$$

for an arbitrarily small positive number ε . In inequality (18), F stands for Frobenius norm.

3. Experiments and Analysis

In the following experiments, X consists of the first five images of each person in the ORL facial image database, a total of 200 data. We set the rank of the basis matrix W 80. In order to reduce the amount of computation and speed up the operating speed, every image is reduced to half.

3.1 Algorithm Convergence

In order to make the process of algorithm convergence seen more clearly in the graph, we set the larger ε 0.001, and randomly initialize W with non-negative data. The set precision of algorithm convergence is obtained after 60 iterations. In this case, the varied curve of the objective function values versus iteration steps is shown in Fig. 2. We can see that the convergence speed of the algorithm is higher. In addition, we initialize W with the first two images of each person in the ORL, and do an experiment again. The same precision of algorithm convergence is obtained after 39 iterations. The varied curve is shown in Fig. 3. We can see faster convergence of the algorithm. This shows the initial value of the basis matrix affects the convergence speed of the algorithm. The reason is that the convergence speed of Newton iteration formula is dependent on initial value, and the initial value is close to the root convergence faster.

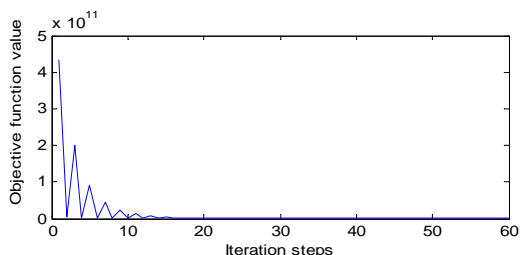


Fig. 2 Objective function values versus iteration steps when the basis matrix is initialized randomly with non-negative data

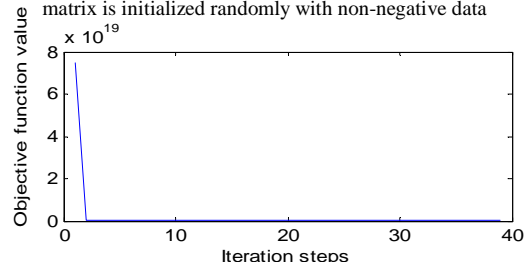


Fig. 3 Objective function values versus iteration steps when the basis matrix is initialized with the known non-negative data

3.2 Analysis of the basis matrix

In the experiment of obtaining Fig. 2, we set the convergent precision ε 0.00001 for the base matrix W while the other set data are unchanged, and do an experiment again. After the algorithm converges, the basis matrix image is shown in Fig. 4. We respectively take the vector $W^T x$ and $(W^T W)^{-1} W^T x$ as the feature vector of the data x and reconstruct x , and reconstructed results are respectively shown in Fig. 6 and Fig. 7.

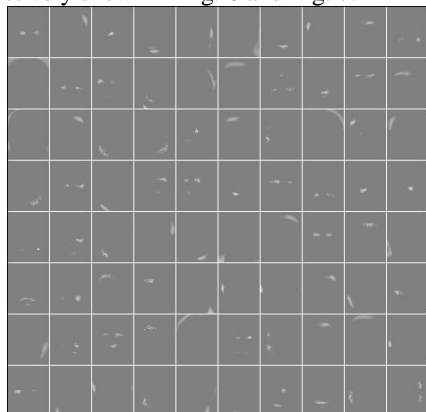


Fig. 4 Basis matrix image



Fig. 5 Original image x ; Fig. 6 $W(W^T x)$; Fig. 7 $(W^T W)^{-1} W^T x$

From the basis matrix image, we can see that the basis matrix is very sparse. This shows that the basis matrix W is forced to tend to be orthogonal by optimizing the objective function F .

From the reconstructed images, we can see that two reconstructed images are all effective, and this shows that the basis matrix W is effective; getting the reconstructed image of x is better by $W(W^T W)^{-1} W^T x$ than by $W(W^T x)$, and this shows that the basis matrix W is still approximately orthogonal, therefore getting the feature vector of data x is better using $(W^T W)^{-1} W^T x$ than using $W^T x$.

The orthogonality and the sparseness of the basis matrix may be computed quantitatively [8, 9]. Without doubt, because this method is still based on the objective function in Eq. (2) for optimization, the orthogonality and the sparseness of the basis matrix are still better. Here, we don't repeat them.

4. Results of Face Recognition and Analysis

In learning phase, X consists of the first five images of each person in the ORL facial image database, a total of 200 data. In order to reduce the amount of computation, and speed up the operating speed, each image is reduced to a quarter of the original. We set ε 0.00002, and initialize randomly the basis matrix W with non-negative data. After the algorithm converges, we get the basis matrix W and feature matrix $(W^T W)^{-1} W^T X$, and take the feature matrix as a template library.

In the pattern recognition test phase, we take the after five images of each person in the ORL facial image database, a total of 200 data, as test data, and reduce every image to a quarter of the original, use $(W^T W)^{-1} W^T x$ to compute the feature vector of test image x by the basis matrix W obtained in the learning phase, and use the nearest neighbor rule for face recognition. We compare this method with the methods of NMF, LNMf, and NMFOS. When the ranks (i.e., the feature subspace dimensions) of the basis matrix are set different values, the results of the face recognition are shown in Fig. 8.

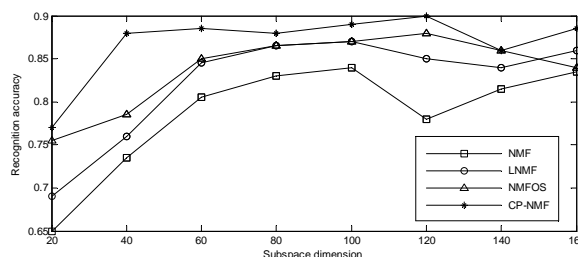


Fig. 8 Comparison of the results of face recognition in the ORL

As can be seen from the Fig. 8, the recognition accuracy is obviously higher using CP-NMF than using NMF. The cause is that the basis matrix W is forced to tend to be orthogonal by the objective function for CP-NMF in Eq. (2) so that the basis matrix is more orthogonal in CP-NMF than in NMF. So the discriminative power of the feature vector $(W^T W)^{-1} W^T x$ for CP-NMF is better. Meantime, when the rank of the basis matrix is greater than or equal to 60, the recognition accuracy is slightly higher using CP-NMF than using LNMF or NMFOS. This is because there are also approximately orthogonal constraints for the basis matrixes in the objective functions for LNMF and NMFOS so that the discriminative power of the feature vectors is also good. But the discriminative power of the feature vector $(W^T W)^{-1} W^T x$ for CP-NMF is better.

In addition, when the rank of the basis matrix for CP-NMF is between 40 and 160, the recognition accuracy becomes more stable. This is because the orthogonality and the sparseness of the basis matrix for CP-NMF are always better so that the recognition accuracy is less affected by the number of the rank of basis matrix.

5. Conclusion

In this paper, we propose a method, called convergent projective non-negative matrix factorization (CP-NMF). In CP-NMF, the algorithm steps are given. The convergence speed of the algorithm is higher. Relative to NMF, the orthogonality and the sparseness of the basis matrix are better. Relative to NMF and some extended NMF methods with orthogonal constraints for the basis matrixes in the objective functions, there is higher recognition accuracy in face recognition.

Acknowledgment

This work was supported by Key Technologies Research and Development Program of Chinese Anhui Province under grant No. 07010202057.

References

- [1] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, 1999, vol. 401, pp. 788–791.
- [2] L. Y. Ma, N. Z. Feng and Q. Wang, "Non-negative matrix factorization and support vector data description based one class classification," *International Journal of Computer Science Issues*, 2012, Vol. 9, No. 5, pp. 36–42.
- [3] M. W. Berry, M. Browne, A. N. Langville, et al., "Algorithms and applications for approximate non-negative matrix factorization," *Computational Statistics & Data Analysis*, 2007, vol. 52, pp. 155–173.
- [4] Z. J. Yuan and E. Oja, "Projective nonnegative matrix factorization for image compression and feature extraction," In: *Proceedings of the fourteenth Scandinavian Conference on Image Analysis*, 2005, pp. 333–342.
- [5] S. Z. Li, X. W. Hou, H. J. Zhang, et al., "Learning spatially localized, parts-based representation," In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2001, pp. 1–6.
- [6] Z. Li, X. Wu and H. Peng, "Non-negative matrix factorization on orthogonal subspace," *Pattern Recognition Letters*, 2010, vol. 31, pp. 905–911.
- [7] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," In *Advances in Neural Information Processing Systems 13*, MIT Press, 2001.
- [8] L. Li and Y. J. Zhang, "Linear projection-based non-negative matrix factorization," *Acta Automatica Sinica*, 2010, vol. 36, pp. 23–39.
- [9] Z. R. Yang, Z. J. Yuan and J. Laaksonen, "Projective nonnegative matrix factorization with applications to facial image processing," *International Journal of Pattern Recognition and Artificial Intelligence*, 2007, vol. 21, pp. 1353–1362.

Lirui Hu was born in Xiangtan, China, in November 1966. He received his Master degree in 2000 in mathematics from Guizhou University, Guizhou China. Presently, he is a doctoral candidate in computer application technology at the Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, Anhui University, Hefei China. He is an associate professor at School of Computer Science and Technology at Nantong University, Nantong China. His research interests include image processing, pattern recognition and machine learning.

Jianguo Wu was born in Suzhou, China, in August 1954. He received his Doctor degree in 1998 from Beijing Institute of Technology, Beijing China. He is a professor at School of Computer Science and Technology at Anhui University, Anhui China. His research interests include Chinese information processing, image processing and pattern recognition.

Lei Wang was born in Xuancheng, China, in March 1987. Presently, he is a master's candidate in computer application technology at the Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, Anhui University, Hefei China. His research interests include image processing and pattern recognition.