A Discrete-Time Projection Neural Network for Sparse Signal Reconstruction With Application to Face Recognition

Bingrong Xu, Qingshan Liu, *Senior Member, IEEE*, and Tingwen Huang, *Senior Member, IEEE*

***Abstract*— This paper deals with sparse signal reconstruction by designing a discrete-time projection neural network. Sparse signal reconstruction can be converted into an L_1-minimization problem, which can also be changed into the unconstrained basis pursuit denoising problem. To solve the L_1-minimization problem, an iterative algorithm is proposed based on the discrete-time projection neural network, and the global convergence of the algorithm is analyzed by using Lyapunov method. Experiments on sparse signal reconstruction and several popular face data sets are organized to illustrate the effectiveness and performance of the proposed algorithm. The experimental results show that the proposed algorithm is not only robust to different levels of sparsity and amplitude of signals and the noise pixels but also insensitive to the diverse values of scalar weight. Moreover, the value of the step size of the proposed algorithm is close to 1/2, thus a fast convergence rate is potentially possible. Furthermore, the proposed algorithm achieves better classification performance compared with some other algorithms for face recognition.**

***Index Terms*— Convergence, face recognition, projection neural network, sparse signal reconstruction, stability.**

I. Introduction

SPARSE representation-based classification (SRC) has become one of the hottest research directions in signal processing and pattern recognition [1], [2]. The main idea of SRC is to use a sparse linear combination of the existing sets to represent a test sample, and then classify the test sample to the class which the representation error is minimum. SRC has been widely applied in several real applications, such as data clustering [3], image classification [4], and face recognition [5].

In general, the sparse solutions can be estimated by solving the L_1-minimization problem [6]. Assume that there is an unknown signal $x \in \mathbb{R}^n$, a measurement vector $b \in \mathbb{R}^m$, and a full row rank dictionary $A \in \mathbb{R}^{m\times n}$ satisfying $b = Ax$. The main purpose of compressive sensing is to obtain x from the known A and b, which can be obtained by solving the L_0-minimization problem

$$\begin{aligned} &\min \ \|x\|_0 \\ &\text{s. t. } Ax = b \end{aligned} \tag{1}$$

where $\|\cdot\|_0$ is the L_0-norm which represents the amount of nonzero elements in a vector. The problem (1) is NP-hard [7]. If the solution to problem (1) is sparse sufficiently, according to the restricted isometry property condition, problem (1) is equal to the L_1-minimization problem [8]

$$\begin{aligned} &\min \ \|x\|_1 \\ &\text{s. t. } Ax = b \end{aligned} \tag{2}$$

where $\|\cdot\|_1$ is the L_1-norm.

Many algorithms have been proposed to solve the problems mentioned above. The L_1-minimization problems can be converted into a linear program, solved by some traditional methods, such as the interior point method [9], for which computational complexity is too high. Also, many effective algorithms have been proposed based on the real applications in various fields. Homotopy method [10] is specifically designed for the context of least absolute shrinkage and selection operator [11] for solving the basis pursuit problem. Moreover, there are several methods to be efficient for L_1-minimization optimization, such as the proximal point method [12], Lagrange dual method [13], and gradient projection method [14]. Also, some algorithms that belong to iterative projection method are proposed for nonlinear large sparse problem [15] which inspires new ideas. Recently, some novel methods have come up to solve the optimization problems based on neurodynamic approach, such as continuous-time recurrent neural networks [16], [17], discrete-time recurrent neural networks [18], [19], and projection neural networks [20]–[22]. The dynamical behaviors of neural networks have been analyzed for nonlinear optimization and applied to robot motion planning [23], support vector machine learning [24], and sparse signal reconstruction [25].

Manuscript received July 24, 2016; revised February 6, 2017, July 28, 2017, November 13, 2017, and March 11, 2018; accepted May 10, 2018. Date of publication June 5, 2018; date of current version December 19, 2018. This work was supported in part by the Jiangsu Provincial Key Laboratory of Networked Collective Intelligence under Grant BM2017002, in part by the Qatar National Research Fund through the National Priority Research under Project NPRP 9-166-1-031, in part by the National Natural Science Foundation of China under Grant 61473333, and in part by the Fundamental Research Funds for the Central Universities. *(Corresponding author: Qingshan Liu.)*

B. Xu is with the School of Automation, Huazhong University of Science and Technology, Wuhan 430074, China, and also with the Key Laboratory of Image Processing and Intelligent Control, Ministry of Education, Wuhan 430074, China (e-mail: bingrongxu@hust.edu.cn).

Q. Liu is with the School of Mathematics, Southeast University, Nanjing 210096, China, and also with the Jiangsu Provincial Key Laboratory of Networked Collective Intelligence, Southeast University, Nanjing 210096, China (e-mail: qsliu@seu.edu.cn).

T. Huang is with the Department of Science Program, Texas A&M University at Qatar, Doha 23874, Qatar (e-mail: tingwen.huang@qatar.tamu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2018.2836933

2162-237X © 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 09:26:13 UTC from IEEE Xplore. Restrictions apply.

Meanwhile, a more efficient neural network, convolutional neural network (CNN), has been successfully applied to face recognition [26]–[29]. Compared with the traditional method, learn features with variations of deep CNNs through hierarchical nonlinear mappings, which are more effective for face recognition, can be learned [30], [31], and it shows obvious advantages in face verification which is performed on some large-scale data sets [32], [33]. In addition to the sparse representation, the methods based on low-rank model to solve the multitask problem have become a hot topic [34]–[36] in these days.

This paper introduces an iterative projection method based on a discrete-time projection neural network for solving the basis pursuit denoising (BPDN) problem. It aims to make the iterative algorithm converge to an optimal solution which is also an equilibrium point of the neural network. When the value of the scalar weight is relatively small, the value of the step size is close to $1/2$, which leads to a fast convergence rate. From the comparison results with other L_1-minimization algorithms, the proposed algorithm has relatively stable convergence ability and is robust to the different amplitudes of signals. In the experiments, the proposed algorithm is compared with both sparse methods and deep learning methods in face verification on Labeled Faces in the Wild (LFW) and YouTube Faces data sets. Compared with SRC and collaborative representation classification (CRC) [37], the proposed algorithm shows the robustness to various value of scalar weight. The results of face recognition on five publicly available data sets, compared with SRC, CRC, robust sparse coding (RSC) [38], and regularized robust coding (RRC) [39], illustrate that the proposed method achieves a better recognition ability for the data sets with high dimensions and images with occlusion.

The organization of the paper is as follows. The discrete-time projection neural network is described and the dynamic behaviors is analyzed in Section II. The proposed algorithm will be compared with other algorithms in Section III and applied to face recognition in Section IV. Finally, the concluding remarks are given in Section V.

II. Discrete-Time Projection Neural Network

In this section, we present a discrete-time projection neural network with piecewise linear activation function to solve the sparse signal reconstruction problem.

A. Projection Operator

Let us define $g(u)$ to be a projection operator from $\mathbb{R}^n$ to $\Omega \subseteq \mathbb{R}^n$ as

$$g(u) = \arg\min_{v \in \Omega} \|v - u\|_2$$

where Ω is a closed convex set and $\|\cdot\|_2$ is the L_2-norm.

In this paper, we only need to consider Ω to be a box set. Assume $\Omega = \{u \in \mathbb{R}^n : l_i \le u_i \le h_i, i = 1, 2, \ldots, n\}$, then

$$g(u_i) = \begin{cases} h_i, & u_i > h_i \\ u_i, & l_i \le u_i \le h_i \\ l_i, & u_i < l_i \end{cases} \tag{3}$$

which is a piecewise linear function. The projection method has been used to construct continuous-time and discrete-time optimization algorithms, such as the ones in [14], [20], and [25].

Lemma 1 [40]: The following inequality holds for a projection operator $g(\cdot)$:

$$(u - g(u))^T (g(u) - v) \ge 0, \quad \forall\, u \in \mathbb{R}^n,\ v \in \Omega.$$

B. Neural Network Modeling

To solve the optimization problem in (2), it will be converted into the BPDN problem, and a discrete-time neural network will be modeled based on the projection method.

Lemma 2 [25]: Assume there is a full row rank matrix A. For any $x \in \mathbb{R}^n$, $Ax = b$ if and only if $Px = q$, where $P = A^T(AA^T)^{-1}A$ and $q = A^T(AA^T)^{-1}b$.

According to Lemma 2, it is obvious that the problem (2) is equivalent to the following L_1-minimization problem:

$$\begin{aligned} &\min \|x\|_1, \\ &\text{s. t. } Px = q. \end{aligned} \tag{4}$$

Another form of this problem is described as the BPDN problem [41] with a positive scalar weight λ

$$\text{minimize } \frac{1}{2}\|Px - q\|_2^2 + \lambda\|x\|_1. \tag{5}$$

Theorem 1: $x^* \in \mathbb{R}^n$ is an optimal solution to problem (5) if and only if there exists $y^* \in \mathbb{R}^n$ such that x^* and y^* satisfy

$$\begin{cases} Px^* - q + \lambda y^* = 0 \\ y^* = g(y^* + x^*). \end{cases} \tag{6}$$

Proof: x^* is an optimal solution to problem (5), if and only if there exists $y^* \in \mathbb{R}^n$ such that x^* and y^* satisfy the following condition:

$$Px^* - q + \lambda y^* = 0 \tag{7}$$

where $y^* = (y_1^*, y_2^*, \ldots, y_n^*)^T$ is a subgradient of $\|x\|_1$ with its component defined as $(j = 1, 2, \ldots, n)$

$$y_j^* \begin{cases} = 1, & x_j^* > 0 \\ \in [-1, 1], & x_j^* = 0 \\ = -1, & x_j^* < 0. \end{cases}$$

Then, y^* satisfies

$$y^* = g(y^* + x^*)$$

where g is a projection operator from $\mathbb{R}^n \to [-1, 1]^n$, which is defined in (3) with $l_i = -1$ and $h_i = 1$. ■

Based on the equations in (6), the discrete-time projection neural network is described as follows:

$$\begin{cases} x(k+1) = x(k) - \alpha(Px(k) - q + \lambda g(y(k) + x(k))) \\ y(k+1) = g(y(k) + x(k+1)) \end{cases} \tag{8}$$

where $\alpha > 0$ is the gain of the network, $x(k)$ is the state vector, $y(k)$ is the output vector, P is the connection weight matrix among different neurons, and q is the input vector.

For simplicity, we use the subscript to denote the iterations, i.e., $x_k = x(k)$ and $y_k = y(k)$, the equations in (8) can be rewritten as

$$\begin{cases} x_{k+1} = x_k - \alpha(Px_k - q + \lambda g(y_k + x_k)) \\ y_{k+1} = g(y_k + x_{k+1}). \end{cases} \tag{9}$$

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 09:26:13 UTC from IEEE Xplore. Restrictions apply.

C. Convergence Analysis

According to Theorem 1, let $x^* \in \mathbb{R}^n$ be an optimal solution to problem (5), then there exists $y^* \in \mathbb{R}^n$ such that the equations in (6) hold. To prove the convergence of the proposed neural network, we first define that

$$V_1(x_k) = \|x_k - x^*\|_2^2, \quad V_2(y_k) = \|y_k - y^*\|_2^2.$$

Then, the properties for $V_1(x_k)$ and $V_2(y_k)$ are presented in following lemma.

Lemma 3: The $V_1(x_k)$ and $V_2(y_k)$ satisfy the following inequalities along the system in (9).

(i) $V_1(x_{k+1}) - V_1(x_k) \le (2\alpha - 1)\|x_{k+1} - x_k\|_2^2 - 2\alpha\lambda x_{k+1}^T(g(y_k + x_k) - y^*)$.
(ii) $V_2(y_{k+1}) - V_2(y_k) \le -\|y_{k+1} - y_k\|_2^2 + 2x_{k+1}^T(y_{k+1} - y^*)$.
(iii) $V_1(x_{k+1}) - V_1(x_k) + \alpha\lambda(V_2(y_{k+1}) - V_2(y_k)) \le (2\alpha - 1 + 2\alpha\lambda)\|x_{k+1} - x_k\|_2^2 - \alpha\lambda(\|y_{k+1} - g(y_k + x_k)\|_2^2 + \|y_k - g(y_k + x_k)\|_2^2)$.

The proof is given in the Appendix.

Next, the stability and convergence of the proposed neural network are addressed in the following theorem.

Theorem 2: For any initial value $x_0 \in \mathbb{R}^n$ and $y_0 \in \mathbb{R}^n$, the vectors x_k and y_k in (9) are stable in the sense of Lyapunov and globally convergent to an equilibrium point if $\alpha < 1/[2(1 + \lambda)]$.

Proof: Let $x^* \in \mathbb{R}^n$ be an optimal solution to problem (5). According to Theorem 1, there exists $y^* \in \mathbb{R}^n$ such that the equations in (6) hold.

Construct a Lyapunov function as follows:

$$V(x_k, y_k) = V_1(x_k) + \alpha\lambda V_2(y_k) \tag{10}$$

where $V_1(x_k)$ and $V_2(y_k)$ are defined earlier.

According to the inequalities and equalities in Lemma 3, we have

$$\begin{aligned} &V(x_{k+1}, y_{k+1}) - V(x_k, y_k) \\ &\quad = V_1(x_{k+1}) - V_1(x_k) + \alpha\lambda(V_2(y_{k+1}) - V_2(y_k)) \\ &\quad \le (2\alpha - 1 + 2\alpha\lambda)\|x_{k+1} - x_k\|_2^2 - \alpha\lambda\|y_{k+1} \\ &\qquad - g(y_k + x_k)\|_2^2 - \alpha\lambda\|y_k - g(y_k + x_k)\|_2^2. \end{aligned} \tag{11}$$

If $\alpha < 1/[2(1 + \lambda)]$, we have $2\alpha - 1 + 2\alpha\lambda < 0$. Then

$$V(x_{k+1}, y_{k+1}) - V(x_k, y_k) \le 0. \tag{12}$$

From (12), $V(x_k, y_k)$ is a Lyapunov function of system (8) indicating its Lyapunov stability.

As for discrete-time system, according to the LaSalle's invariance principle [42], x_k and y_k will converge to the largest invariant subset of the following set:

$$\Theta = \{x_k, y_k \in \mathbb{R}^n : V(x_{k+1}, y_{k+1}) - V(x_k, y_k) = 0\}.$$

Note that from (11), if $V(x_{k+1}, y_{k+1}) - V(x_k, y_k) = 0$, we have

$$\begin{aligned} &(2\alpha - 1 + 2\alpha\lambda)\|x_{k+1} - x_k\|_2^2 \\ &\quad = \alpha\lambda(\|y_{k+1} - g(y_k + x_k)\|_2^2 \\ &\qquad + \|y_k - g(y_k + x_k)\|_2^2). \end{aligned} \tag{13}$$

Substituting it with (9), we have

$$\begin{aligned} &(2\alpha - 1 + 2\alpha\lambda)\|\alpha(Px_k - q + \lambda g(y_k + x_k))\|_2^2 \\ &\quad = \alpha\lambda(\|y_{k+1} - g(y_k + x_k)\|_2^2 \\ &\qquad + \|y_k - g(y_k + x_k)\|_2^2). \end{aligned} \tag{14}$$

As mentioned earlier, $2\alpha - 1 + 2\alpha\lambda < 0$, and it is obvious that $\|y_{k+1} - g(y_k + x_k)\|_2^2 + \|y_k - g(y_k + x_k)\|_2^2 \ge 0$. We can obtain that

$$Px_k - q + \lambda g(y_k + x_k) = 0. \tag{15}$$

Substituting it with (14), we have

$$y_k - g(y_k + x_k) = 0. \tag{16}$$

Thus, x_k is an optimal solution to problem (5).

From (12), we have

$$\begin{aligned} &\|x_{k+1} - x^*\|_2^2 + \alpha\lambda\|y_{k+1} - y^*\|_2^2 \\ &\quad \le \|x_k - x^*\|_2^2 + \alpha\lambda\|y_k - y^*\|_2^2 \\ &\quad \le \cdots \\ &\quad \le \|x_0 - x^*\|_2^2 + \alpha\lambda\|y_0 - y^*\|_2^2. \end{aligned}$$

Then, x_k and y_k are bounded in $\mathbb{R}^n$. There exists an increasing subsequence $\{k_l\}_{l=1}^{\infty}$ such that

$$\lim_{l\to\infty} x_{k_l} = x^*, \quad \lim_{l\to\infty} y_{k_l} = y^*.$$

From above analysis, for any k, there exists k_l such that

$$\|x_k - x^*\|_2^2 + \alpha\lambda\|y_k - y^*\|_2^2 \le \|x_{k_l} - x^*\|_2^2 + \alpha\lambda\|y_{k_l} - y^*\|_2^2.$$

Let $l \to \infty$, $k \to \infty$, then

$$\begin{aligned} &\lim_{k\to\infty} \|x_k - x^*\|_2^2 + \alpha\lambda\|y_k - y^*\|_2^2 \\ &\quad = \lim_{l\to\infty} \|x_{k_l} - x^*\|_2^2 + \alpha\lambda\|y_{k_l} - y^*\|_2^2. \\ &\quad = 0 \end{aligned}$$

That is, $\lim_{k\to\infty} x_k = x^*$, $\lim_{k\to\infty} y_k = y^*$. Consequently, for any initial point $x_0 \in \mathbb{R}^n$ and $y_0 \in \mathbb{R}^n$, the vector (x_k, y_k) is convergent to an equilibrium point of the neural network in (9). ■

From Theorem 1, if (x^*, y^*) is an equilibrium point of neural network (9), x^* is also an optimal solution to problem (5). Combining with Theorem 2, the state vector x_k of the discrete-time projection neural network in (9) is convergent to an optimal solution of problem (5) if a and λ satisfy the constraint condition $\alpha < 1/[2(1 + \lambda)]$. Obviously, different choices for λ will yield different optimal solutions to problem (5), and we do not have a clear method to choose the optimal λ to get a lower reconstruction error. Thus, in the following algorithms which are proposed to solve the sparse signal reconstruction problems, the value of λ will be confirmed during the experiments.

Based on (9), the Algorithm 1 for solving the BPDN problem (5) is stated as following:

Remark 1: Considering the constraint condition (2) with noise, $Ax + e = b$. According to Lemma. 2, we can get $P = A^T(AA^T)^{-1}A$ and $q = A^T(AA^T)^{-1}(b - e)$ (where e represents the noise pixel, block occlusion or face disguise,

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 09:26:13 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1 Projection Neural Network-Based Iterative Method

Initialization:

Set $k = 0$, any $x_0 \in \mathbb{R}^n$ and $y_0 \in \mathbb{R}^n$.

Iteration:

while x_k is not converged **do**

$x_{k+1} \leftarrow x_k - \alpha(Px_k - q + \lambda g(y_k + x_k))$;

$y_{k+1} \leftarrow g(y_k + x_{k+1})$.

end while

return $x^* \leftarrow x_k$.

$(b - e)$ represents the test images with noises). Taking the obtained P and q into Algorithm 1, the noises will be reduced on a certain degree. The experiments in Section IV will illustrate that the projection neural network-based iterative method (PNNBIM) is validated to identify the faces with noises.

Remark 2: The constraint condition of original problem $Ax = b$ is transformed into $Px = q$, and the norm of matrix P is 1, which can avoid the effect of sample values in classification and improve the algorithm's robustness. Moreover, the gradient of $\|x\|_1$ is transformed into continuous projection function, which can enhance the convergence of the algorithm.

III. Reconstruction Experimental Results

In this section, the performance of the proposed algorithm is benchmarked in two experiments. To show the advantages of the proposed PNNBIM, the convergence rate is compared with several other L_1-minimization algorithms, including fast interior soft-thresholding algorithm, primal augmented Lagrangian method, truncated Newton interior-point method (L1LS), sparse reconstruction by separable approximation, and dual-augmented Lagrangian method (DALM) (see [43], [44] for more details).

Experiment 1: In the experiment, the full row rank matrix $A \in \mathbb{R}^{m\times n}(m < n)$ is set as every column normalized to L_2-norm, and each entry is independent and identically distributed Gaussian. We generate the observation b set as $b = Ax$, where x's sparsity level is K, and the nonzero elements in a vector in x are random. Set $n = 1000$, $m = 800$, and $K = 30$. The reconstruction results are shown in the Figs. 1 and 2.

Fig. 1 describes the original signal and its reconstruction signals which is obtained by Algorithm 1. Fig. 1(a) shows the original signal, and Fig. 1(b) is the sparse signal reconstruction result in which the error $e = \|Ax - b\|_2 \leq 10^{-3}$. Fig. 2 illustrates the convergence results of the proposed method. Fig. 2(a) demonstrates the results with the fixed value of step size $\alpha = 0.4$ and different values of scalar weight λ ($\lambda = 5 \times 10^{-4}$, 5×10^{-3}, 0.01, 0.1, and 0.5). We can observe that with the value of λ increasing, the PNNBIM converges faster and the reconstruction error of x increases. The most ideal result appears at $\lambda = 0.01$ taking both convergence speed and reconstruction error into consideration. Fig. 2(b) depicts the convergences with the fixed value of λ and various values of α. Based on the above result, we set $\lambda = 0.01$ and take $\alpha = 0.1$, 0.2, 0.3, 0.4, and 0.499. It is obvious that the PNNBIM converges faster with bigger value of the step size α.

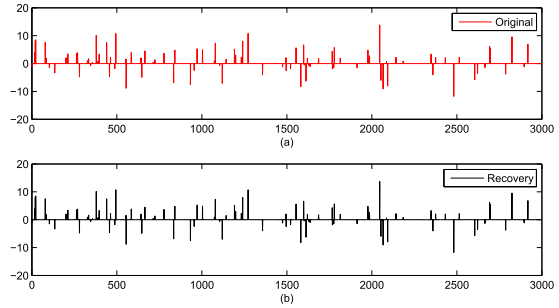
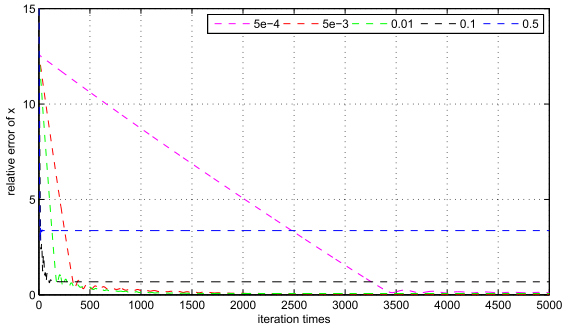
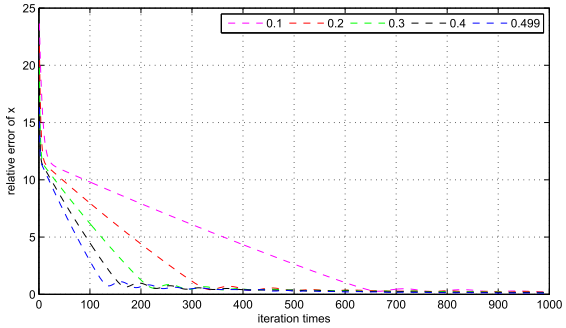


Fig. 1. (a) Original signal. (b) Sparse signal reconstruction using Algorithm 1.



(a)



(b)

Fig. 2. Convergence of PNNBIM with (a) $\alpha = 0.4$ and different values of λ and (b) $\lambda = 0.01$ and different values of α.

Experiment 2: The signals are produced with the same parameter setting as in the previous experiment, but with different sparsity K. Since most signals, in many practical applications, are in multimodel, the random signals are combined with 100 models with their Gaussian distribution in $[-50, 50]^m$. Set $n = 1000$, $m = 800$, and $K = 20, 100$. Fig. 3 describes the average relative error of 10 trials regarding the convergence time, in which (a) and (b) represent the results of normalized signals with unit L_2-norm, and (c) and (d) represent the results of unnormalized signals.

In Fig. 3, we can obtain the conclusion that the proposed PNNBIM is not only robust to the signals with normalized or unnormalized form but also to the sparsity levels of signals, which is better than most of the other algorithms shown in the experiment. Fig. 3 shows that DALM is the fastest among the six algorithms, but the proposed PNNBIM has relatively stable convergence ability, especially when the sparsity level is large.

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 09:26:13 UTC from IEEE Xplore. Restrictions apply.

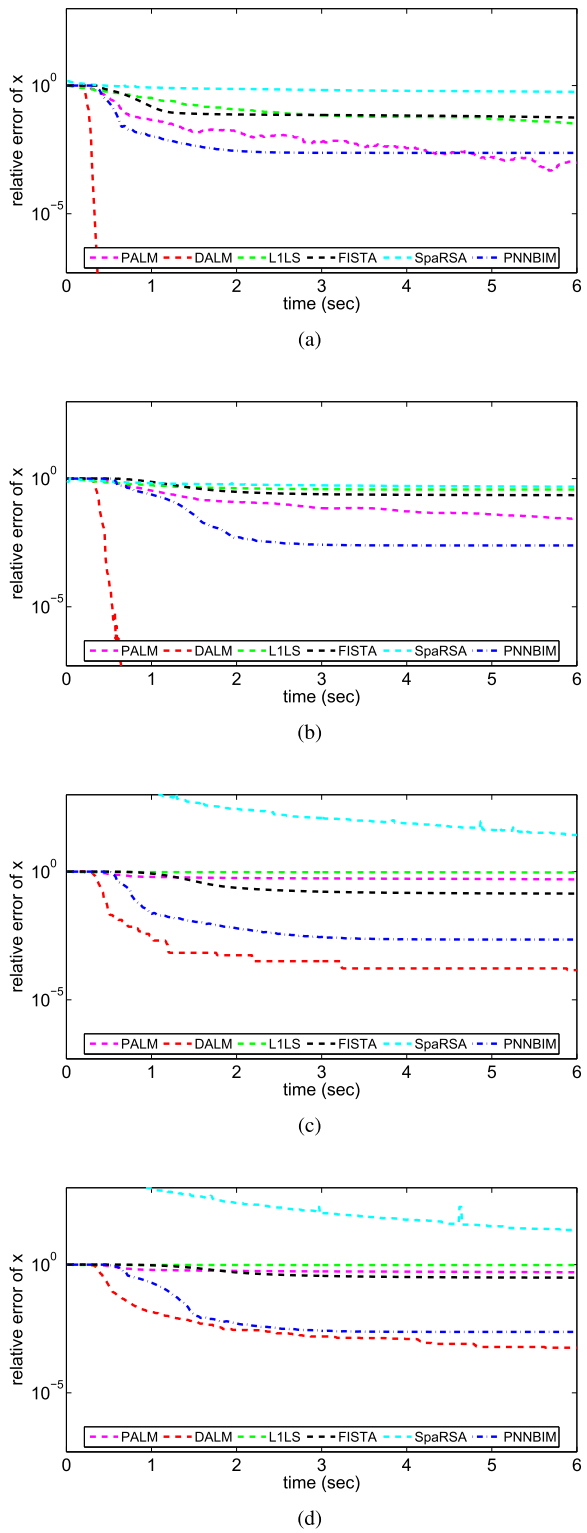


Fig. 3. Relative error of x in Experiment 2. (a) and (b) Result of normalized signals for different sparsity levels. (c) and (d) Result of unnormalized signals for different sparsity levels.

IV. Application to Face Recognition

Face recognition is composed of four parts: face detection, face alignment, face verification, and face classification (identification). Face verification is to measure the similarity of two images, and face classification is to identify which category the test image belongs to. In this section, the proposed PNNBIM is applied for face recognition, experiments are conducted to demonstrate the verification ability and classification ability of the algorithm, compared with deep learning methods (Facenet [45], Deepface [46], Normface [47], FVSR [48], and MSSRC [49]) and traditional sparse algorithms (SRC using DALM method, CRC, RSC, and RRC). Experiments are conducted on widely used face data sets, the Extended Yale B (EYB), AR, Carnegie Mellon University (CMU) PIE, CAS-PEAL, LFW and YouTube Faces (YTF) DB, to demonstrate the recognition ability. And then, we compare our method with robust regression models-based approach (RSC and RRC) on the EYB data set with pixel corruption, block occlusion, and AR data set with face disguise. Finally, we study the influence of different scalar weights λ on SRC, CRC and PNNBIM to test their performance and robustness. For each data set, the training samples are chosen randomly to guarantee that the classification results will not be affected by any particular choice of training data, and use the rest samples for testing.

As for face recognition, assuming the frontal face images can be expressed by a low-dimensional subspace spanned by

$$A_i = \{a_{i1}, a_{i2}, \ldots, a_{in_i}\}$$

where a_{ij} $(j = 1, 2, \ldots, n_i)$ is the vector form stacked from the ith training image. Assume that there are c kinds of subjects provided for recognition. Let all the training images compose the matrix $A \in \mathbb{R}^{m\times n}$ which is defined as

$$A = \{A_1, A_2, \ldots, A_c\}.$$

The procedure of obtaining the most sparse linear representation of the test face image b can be expressed as the following form:

$$b = Ax \tag{17}$$

where $x = (x_1, x_2, \ldots, x_n)^T \in \mathbb{R}^n$. In brief, if b is a valid test image which belong to the c kinds of training set, the solution x in (17) can be expressed as a sparse representation $x = (0, \ldots, 0, x_{i1}, x_{i2}, \ldots, x_{in_i}, 0, \ldots, 0)^T \in \mathbb{R}^n$, in which every entries are zeros, but only those related to the ith training image.

Based on the PNNBIM, the algorithm for face verification and classification problem are stated as Algorithm 2. Step 8 to 9 are for verification task and step 10 to 11 are for classification task.

The residuals in the Algorithm 2 contain $\|x^i\|_0$ and $\|x^i\|_2$, indicating that the correlation between the test image and the training iterations is taken into consideration from both quantity and amplitude, which are different from SRC [44] or CRC [37] methods. In experiments, the value of the optimal parameter λ is set as 0.01, and according to the constraint condition $\alpha < 0.495$. On the basis of the actual experimental results, we set $\alpha = 0.4$ to get the best recognition results, and set the threshold value $th = 0.4$.

A. Face Verification

1) Experiment on LFW Data set: This experiment is conduct on LFW data set [50] which contain 13 233 faces with visual variations, due to pose, facial appearance, age, lighting,

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 09:26:13 UTC from IEEE Xplore. Restrictions apply.

Algorithm 2 PNNBIM for Face Recognition

Initialization:

Set $k = 0$, any $x_0 \in \mathbb{R}^n$ and $y_0 \in \mathbb{R}^n$, reconstruction error θ and sparsity level K, threshold value th. Input the training sets $A_1, A_2, \ldots, A_c$ and the test image b.

Iteration:

while $\|Ax - b\|_2 > \theta$ **do**

  $x_{k+1} \leftarrow x_k - \alpha[Px_k - q + \lambda g(y_k + x_k)]$;

  $y_{k+1} \leftarrow g(y_k + x_{k+1})$;

end while

return $x^* \leftarrow x_k$;

For verification task, go to Step 8; for classification task, go to Step 10.

Verification:

Compute the cosine similarity (CS) of the two spars codes

$$CS(x^i, x^j) = \frac{(x^i)^T x^j}{\|x^i\|_2 \|x^j\|_2}$$

where x^i and x^j are the sparsity representation vectors related to classes i and j respectively;

If $CS(x^i, x^j) > th$, the images correspond to the same person; else the images to different persons.

Classification:

Compute the residuals

$$e_i(b) = \frac{\|b - A_i x^i\|_2}{(\|x^i\|_0 \|x^i\|_2)}$$

where x^i is the sparsity representation vector related to class i;

The classification of b is

$$\text{class}(b) = \arg\min_i \{e_i\}.$$

occlusion, scale, misalignment, gender, etc. The test is based on the standard unrestricted with labeled outside data protocol, which is evaluated on 6000 image pairs.

Inspired by the feature extraction procedure in [46], the images in the LFW data set are first detected with bounding boxes and five facial points using multitask cascaded CNNs [51], and later aligned. Then, we use the Facenet to extract the 128-D features, and the principal component analysis (PCA) is used to reduce the dimension to 100, which leads to a better verification result.

In the experiments, we compare the PNNBIM not only with the sparse methods, MMSRC and FVSR, but also with the-state-of-the-art methods using deep learning, such as Facenet, Deepface, and Normface, which show the great advantages in face recognition. The MMSRC and FVSR adopt hybrid HoG, local binary pattern, and Gabor features, and we modify the MMSRC (used for classification in the original paper) to use the same CS distance method as PNNBIM for verification. Also, we compare the results processing the features with [100 dimensions, named PNNBIM(PCA) in Table I] and without (128 dimensions, named PNNBIM in Table I) PCA method. Moreover, we compare the results acquired by CS distance method dealing with the original features obtained by Facenet (128 dimensions, not calculated by PNNBIM, called CSF in Table I), with the sparsity representation vector obtained by PNNBIM, to testify whether the algorithm is efficient in the verifying process.

TABLE I

VERIFICATION RESULTS ON LFW DATA SET (%)

Algorithms	Facenet	Deepface	Normface	MMSRC
Accuracy	99.20	98.35	98.78	78.75
Algorithms	FVSR	CSF	PNNBIM	PNNBIM(PCA)
Accuracy	70.18	77.35	97.85	99.30

TABLE II

VERIFICATION RESULTS ON YTF DATA SET (%)

Algorithms	Facenet	Deepface	Normface	PNNBIM
Accuracy	95.12	91.44	94.72	95.14

The verification results are shown in Table I. It is obvious that the deep learning methods achieve much higher verification accuracies than the traditional SRC based methods using hybrid feature extraction methods. Based on the features obtained by deep learning method and reduced by PCA, the PNNBIM(PCA) gets the highest accuracy among the methods, reached 99.30%. From the comparison of CSF and PNNBIM, the accuracy using the original features (CSF) is only 77.35%, we could get a conclusion that the effectiveness of the PNNBIM is significant. By comparing the results of PNNBIM and PNNBIM(PCA), it can be concluded that the proposed algorithm has advantages to deal with the recognition tasks with low dimensions.

2) Experiment on YTF Data set: This experiment is validated on the YTF data set [52] which consists of 3425 videos of 1595 different people. We follow the unrestricted protocol with labeled outside data, which takes 5000 video pairs to evaluate the performance. Given a test pair, all of the first 100 frames are used for verification.

The features of each images are extracted through the method mentioned above, and reduced dimensions to 100 by PCA method. The results obtained by Facenet, Deepface, Normface, and PNNBIM are given in Table II, and the proposed PNNBIM achieves 95.14%, slightly higher than the other methods.

B. Face Classification Without Occlusion

1) Experiment on Extended Yale B Data set: The EYB data set [53] contains over 2000 face images of 38 individuals under different illumination circumstances. Fig. 4 depicts some image samples of data set. For each class, we randomly select parts of the images to consist the dictionary, and use the others for classification. We use the PCA method to reduce the images' dimension to 200, and the number of each class's training images is $N = 10$, 15, and 20. The recognition rates of the proposed PNNBIM, SRC using DALM method, and CRC are compared in Table III.

From Table III, we can obtain that the classification accuracy of the proposed PNNBIM is higher than typical SRC

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 09:26:13 UTC from IEEE Xplore. Restrictions apply.

Fig. 4. Some image samples of EYB data set.

TABLE III
RECOGNITION RATE ON EYB DATA SET (%)

Algorithms	SRC	CRC	RSC	RRC	PNNBIM
N=10	87.17	89.38	73.45	74.20	90.75
N=15	90.78	91.86	83.46	83.68	93.49
N=20	94.80	94.68	94.38	94.68	95.10

Fig. 5. Some image samples of AR data set.

TABLE IV
RECOGNITION RATE ON AR DATA SET (%)

Algorithms	SRC	CRC	RSC	RRC	PNNBIM
N=4	78.75	84.75	81.75	83.00	84.25
N=5	80.00	85.17	83.5	85.20	85.83
N=6	81.40	85.83	84.6	86.00	87.00

Fig. 6. Some image samples of CMU PIE data set.

using DALM and CRC, especially for small training samples. For example, when $N = 10$, the recognition rate of PNNBIM is 90.75% whereas that of SRC is only 87.17%.

2) Experiment on AR Data set: The AR data set [54] includes over 4000 color face images consisted by 126 subjects, each one includes the faces with different facial expressions, illumination circumstances, and occlusions. Some image samples of the AR face data set are given in Fig. 5. First, we convert all the images into gray version, and then reduce the images' dimension to 100 obtained by PCA. The number of training samples chosen for each class is N ($N = 4, 5, 6$), and the rest subsets are for testing. The comparison of recognition rates is shown in Table IV. It shows that the proposed PNNBIM achieves a higher recognition rate comparing with SRC, CRC, RSC, and RRC, especially has greater advantages over SRC.

3) Experiment on CMU PIE Data set: The CMU PIE data set [55] contains 68 subjects with images captured by 13 poses and 43 illumination conditions and 4 different positions of each subject. Fig. 6 shows some sample faces of this data set. The dimension of each image is reduced to 100 by using PCA. Then, for each subject, 5 or 10 images are randomly selected for training, the rests are for classification. The comparison of recognition rate is described in Table V. From Table. V, we can observe that the recognition accuracy of the proposed method PNNBIM outperforms than the other four algorithms.

TABLE V
RECOGNITION RATE ON CMU PIE DATA SET (%)

Algorithms	SRC	CRC	RSC	RRC	PNNBIM
N=5	80.56	89.19	80.26	79.09	89.60
N=10	93.39	94.83	94.22	95.02	95.05

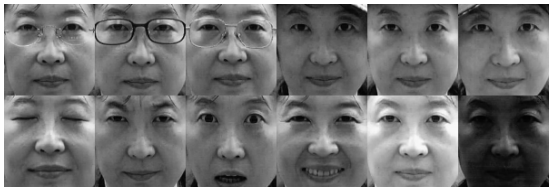


Fig. 7. Some image samples of CAS-PEAL data set. Images with accessories for training (first row). Parts of images for testing (second row).

TABLE VI
RECOGNITION RATE ON CAS-PEAL DATA SET (%)

Algorithms	SRC	CRC	RSC	RRC	PNNBIM
rate	92.25	90.75	88.25	92.25	93.50

Fig. 8. Some image samples of LFW data set.

4) Experiment on CAS-PEAL Data set: The CAS-PEAL data set [56] contains 99 594 faces with different expressions and accessories under various illuminations situations; it can be utilized for face verification, face classification, and age detection. In the experiment, all the images are cropped to size 100×82, and the eyes are located at certain position [here we choose (19, 28) and (63, 28)]. The 434 individuals from the accessory category are for training, and the rest in aging category, illumination category, and expression category are used for testing. Some images of the data set are shown in Fig. 7. The comparison of recognition rates is described in Table VI. It is obvious that the recognition rate of the PNNBIM outperforms is slightly higher than the other algorithms.

5) Experiment on LFW Data set: In the experiments, we choose 143 individuals with more than 12 pictures per person. Fig. 8 shows some sample faces of this data set. The first 10 images of each individual are used for training, and the rest are for classification. The face detection, alignment, and feature extraction proceedings are the same as the verification experiment, and then we follow the Step 10 and 11 in Algorithm 2. And we compared the classification results of the tested algorithms with deep convolutional neural network (DCNN) features and PCA features (features obtained only using the PCA method), the comparison results are shown in Table VII. We can see that the PNNBIM have the advantage over the other methods, and it demonstrates that the tested algorithms using the DCNN features made great progress compared with the ones using the PCA features only.

C. Face Classification With Occlusion

In this section, we compare the proposed PNNBIM with RSR and RRC which are proved to be effective in face

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 09:26:13 UTC from IEEE Xplore. Restrictions apply.

TABLE VII
RECOGNITION RATE ON LFW DATA SET WITH DCNN FEATURES AND PCA FEATURES(%)

Algorithms	SRC	CRC	RSC	RRC	PNNBIM
rate (PCA fea.)	73.07	74.15	71.58	73.29	74.88
rate (DCNN/PCA fea.)	92.24	92.22	91.35	91.92	92.53

Fig. 9. Some EYB image samples under different levels of random noise.

TABLE VIII
RECOGNITION RATES UNDER DIFFERENT NOISE LEVELS (%)

Algorithms	0	0.2	0.4	0.6
RSC	93.89	91.72	88.87	78.96
RRC	94.68	92.68	90.08	82.29
PNNBIM	95.10	92.74	91.00	83.97

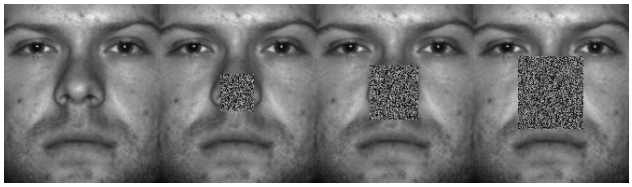


Fig. 10. Some EYB image samples under different levels of block occlusion.

recognition with random pixel corruption, block occlusion on EYB data set, and face disguise on AR data set. Then, we analyze the recognition results and compare the running time of three algorithms.

1) Face Classification With Noise Pixel: The EYB data set is used to demonstrate the efficiency of PNNBIM, RSC, and RRC. The setting of the algorithms is the same as in the previous experiments, but for each testing image, a certain percentage of pixels are added with random Gaussian noise. Fig. 9 shows the original image and the corrupted images with the percentage of 20, 40, and 60. In Table VIII, we can obtain the conclusion that the PNNBIM method performs better than RSC and RRC, which is robust to various types of outliers (i.e., occlusion, noises, expression, etc.), especially with the growth percentage of the noised pixel.

2) Face Classification With Block Occlusion: The performance that the three algorithms process the images with the block occlusion is validated on the EYB data set as in previous experiments. Each test images' middle block is occluded by random pixels, and Fig. 10 shows the example of the original image and the images under the 20%, 30%, and 40% block occlusion. The classification results are given in Table IX. When the block occlusion is 20%, the recognition rate of RRC is slightly higher than PNNBIM, but with the percentage of the occlusion increased, the PNNBIM enjoys the advantage

TABLE IX
RECOGNITION RATES UNDER DIFFERENT BLOCK OCCLUSION LEVELS (%)

Algorithms	0.2	0.3	0.4	0.5
RSC	86.75	77.93	67.47	55.86
RRC	87.24	80.47	72.31	56.80
PNNBIM	87.12	84.76	80.13	67.77

Fig. 11. Some AR image samples with sunglasses and scraf.

TABLE X
RECOGNITION RATES WITH FACE DISGUISE (%)

Algorithms	RSC	RRC	PNNBIM
Sunglasses	83.33	83.67	87.33
Scarf	61.33	66.33	68.00

TABLE XI
RUNNING TIME OF THREE ALGORITHMS (s)

Algorithms	RSC	RRC	PNNBIM
30% noise	430.64	7391.04	16.86
50% occlusion	513.61	12683.88	29.29

of face recognition. When the percentage of occlusion reaches 50, the recognition rate of the PNNBIM is much higher than RSC and RRC.

3) Face Classification With Disguise: The face recognition with disguise is a challenging and meaningful task which attracts many researchers' interest. In this section, we choose the AR data set as experimental object, since it contains the face images with sunglasses or scarf. For each individual, the 14 images without disguise are used for training, and six images with sunglasses and six images with scarf are used for testing separately. The testing samples are shown in Fig. 11, and the recognition rates are described in Table X, which shows the effectiveness and superiority of the PNNBIM on face disguise recognition.

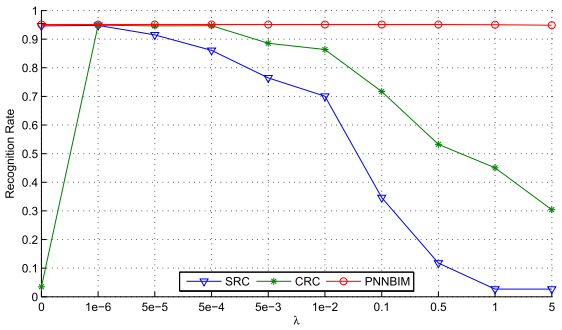
D. Comparison of Running Time

From previous experiments, the PNNBIM is proven to have good performance with face occlusion. In addition to the classification accuracy, computing time consumption is also an important aspect to measure the capability of the algorithms. We compare the running time of the PNNBIM, RSC, and RRC in the experiments of EYB images with 30% noise and 50% block occlusion. The results are described in Table XI. Combining with the conclusions in earlier experiments, it is obvious that the PNNBIM is not only the most effective but also the fastest algorithm among the three algorithms.

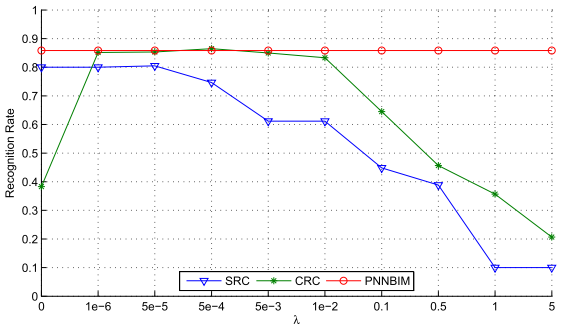
E. Influence of Parameter λ

The regularization parameter λ plays an important role in the BPDN problem (5), and small change of the value may

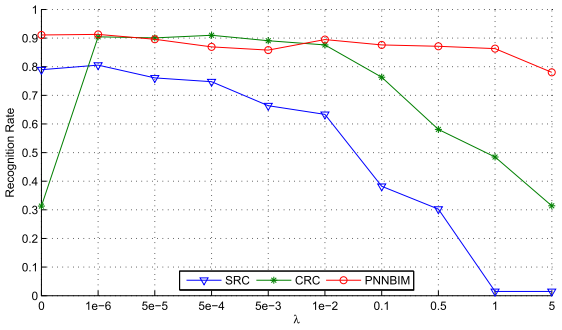
Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 09:26:13 UTC from IEEE Xplore. Restrictions apply.



(a)



(b)



(c)

Fig. 12. Recognition rates of SRC, CRC, and PNNBIM versus the different values of λ on three data sets. (a) EYB. (b) AR. (c) CMU PIE.

lead to large difference in the results. In this experiment, we study the robustness of three algorithms to different values of regularization parameter λ in face recognition. We compare the proposed PNNBIM with SRC using the dual-augment Lagrangian method and CRC for face recognition.

The experiments are conducted on three face databases, the EYB, AR, and CMU PIE. In the EYB, we randomly choose 20 images of each class for training and reduce the images' dimension to 200; in the AR, the number of training images is set as six and the dimension of each image is 100; in the CMU PIE, five images are used for training and the dimension is set as 100.

The results on the EYB, AR, and CMU PIE are shown in Fig. 12. We can observe that the proposed algorithm is robust to the values of weight λ for a large range, though the recognition rate of the PNNBIM fluctuates slightly in the CMU PIE data set. Fig. 12 shows that all the three algorithms achieve good results when λ is assigned to a relatively small positive value. With the increasing values of $\lambda(> 0.01)$, the recognition rates of SRC and CRC drop. However, the PNNBIM maintains the ideal rate, which shows the robustness to scalar weight λ.

Remark 3: According to the results from Figs. 2 and 12, the value of λ has little effect on the recognition rate, but the convergence speed has, obviously, positive correlation with the value of α. Therefore, it is recommended to set a larger value of α and a proper value of λ to obtain a faster convergence speed and an ideal recognition rate.

V. Conclusion

In this paper, we propose a projection neural network-based iterative method for sparse signal reconstruction with application to face recognition. The stability and convergence of the neural network is proven by using Lyapunov method. Experimental results on the signal reconstruction show that the proposed algorithm is robust to different sparsity levels and amplitudes of signals. Furthermore, six public data sets are used to test the performance of the proposed algorithm for face recognition, which demonstrate that the PNNBIM not only achieves better recognition results comparing with other algorithms but also be robust to the different values of the scalar weight λ in the optimization problem.

Appendix

Proof of Lemma *3:*

(i) We have

$$\begin{aligned}
&V_1(x_{k+1}) - V_1(x_k) \\
&\quad = \|x_{k+1} - x^*\|_2^2 - \|x_k - x^*\|_2^2 \\
&\quad = (x_{k+1} - x_k)^T (x_{k+1} + x_k - 2x^*) \\
&\quad = -\|x_{k+1} - x_k\|_2^2 + 2(x_{k+1} - x_k)^T (x_{k+1} - x^*) \\
&\quad = -\|x_{k+1} - x_k\|_2^2 \\
&\qquad -2\alpha(Px_k - q + \lambda g(y_k + x_k))^T (x_{k+1} - x^*) \\
&\quad = -\|x_{k+1} - x_k\|_2^2 - 2\alpha(x_{k+1} - x^*)^T (Px_k - q) \\
&\qquad - 2\alpha\lambda(x_{k+1} - x^*)^T g(y_k + x_k). \qquad (18)
\end{aligned}$$

Let $f(x) = \|Px - q\|_2^2/2$, then $f(x)$ is convex. We have

$$(x_k - x^*)^T (Px_k - q) \geq f(x_k) - f(x^*)$$

and

$$(x_{k+1} - x_k)^T (Px_{k+1} - q) \geq f(x_{k+1}) - f(x_k)$$

which implies

$$\begin{aligned}
&(x_{k+1} - x^*)^T (Px_k - q) \\
&\quad = (x_{k+1} - x_k)^T (Px_k - q) + (x_k - x^*)^T (Px_k - q) \\
&\quad \geq (x_{k+1} - x_k)^T (Px_k - q) + f(x_k) - f(x^*) \\
&\quad = (x_{k+1} - x_k)^T (Px_k - Px_{k+1}) \\
&\qquad + (x_{k+1} - x_k)^T (Px_{k+1} - q) + f(x_k) - f(x^*) \\
&\quad \geq (x_{k+1} - x_k)^T P(x_k - x_{k+1}) + f(x_{k+1}) - f(x^*).
\end{aligned}$$

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 09:26:13 UTC from IEEE Xplore. Restrictions apply.

Substituting it into (18) results that

$$\begin{aligned}
&V_1(x_{k+1}) - V_1(x_k) \\
&\quad \le -\|x_{k+1}-x_k\|_2^2 + 2\alpha(x_{k+1}-x_k)^T P(x_{k+1}-x_k) \\
&\qquad + 2\alpha(f(x^*) - f(x_{k+1})) \\
&\qquad -2\alpha\lambda(x_{k+1}-x^*)^T g(y_k+x_k) \\
&\quad \le (2\alpha-1)\|x_{k+1}-x_k\|_2^2 + 2\alpha(f(x^*)-f(x_{k+1})) \\
&\qquad - 2\alpha\lambda(x_{k+1}-x^*)^T g(y_k+x_k) \qquad (19)
\end{aligned}$$

where the last inequality holds since $\|P\|_2 \le 1$.

Since x^* is a minimum point of $f(x)+\lambda\|x\|_1$, we get that $0 \in \nabla f(x^*) + \lambda\partial(\|x^*\|_1)$. That is,

$$Px^* - q + \lambda y^* = 0 \qquad (20)$$

where $y^* = g(y^*+x^*) \in \partial(\|x^*\|_1)$.

From the convexity of $f(x)$, one gets that

$$(x_{k+1}-x^*)^T(Px^*-q) \le f(x_{k+1}) - f(x^*).$$

Then, we have $\lambda(x_{k+1}-x^*)^T y^* \ge f(x^*) - f(x_{k+1})$ from (20). Combining with (19) follows that

$$\begin{aligned}
&V_1(x_{k+1}) - V_1(x_k) \\
&\quad \le (2\alpha-1)\|x_{k+1}-x_k\|_2^2 \\
&\qquad - 2\alpha\lambda(x_{k+1}-x^*)^T(g(y_k+x_k)-y^*). \qquad (21)
\end{aligned}$$

Since $y^* = g(y^*+x^*)$, then y^* satisfies the following variational inequality

$$(y-y^*)^T(-x^*) \ge 0 \quad \forall y \in [-1,1]^n.$$

Since $g(y_k+x_k) \in [-1,1]^n$, we have

$$(g(y_k+x_k)-y^*)^T x^* \le 0.$$

Then

$$\begin{aligned}
&(x_{k+1}-x^*)^T(g(y_k+x_k)-y^*) \\
&\quad = x_{k+1}^T(g(y_k+x_k)-y^*) - (x^*)^T(g(y_k+x_k)-y^*) \\
&\quad \ge x_{k+1}^T(g(y_k+x_k)-y^*).
\end{aligned}$$

Substituting the previous inequality into (21) follows that

$$\begin{aligned}
&V_1(x_{k+1}) - V_1(x_k) \\
&\quad \le (2\alpha-1)\|x_{k+1}-x_k\|_2^2 - 2\alpha\lambda x_{k+1}^T(g(y_k+x_k)-y^*).
\end{aligned}$$

(ii) For $V_2(y_k)$, we have

$$\begin{aligned}
&V_2(y_{k+1}) - V_2(y_k) \\
&\quad = \|y_{k+1}-y^*\|_2^2 - \|y_k-y^*\|_2^2 \\
&\quad = (y_{k+1}-y_k)^T(y_{k+1}+y_k-2y^*) \\
&\quad = -\|y_{k+1}-y_k\|_2^2 + 2(y_{k+1}-y_k)^T(y_{k+1}-y^*) \\
&\quad = -\|y_{k+1}-y_k\|_2^2 + 2(y_{k+1}-y_k-x_{k+1})^T(y_{k+1}-y^*) \\
&\qquad + 2x_{k+1}^T(y_{k+1}-y^*).
\end{aligned}$$

Note that $y_{k+1} = g(y_k+x_{k+1})$. Let $u = y_k+x_{k+1}$ and $v = y^*$. From the inequality in Lemma 1, we have

$$(y_{k+1}-y_k-x_{k+1})^T(y_{k+1}-y^*) \le 0.$$

Then

$$V_2(y_{k+1}) - V_2(y_k) \le -\|y_{k+1}-y_k\|_2^2 + 2x_{k+1}^T(y_{k+1}-y^*).$$

(iii) From the inequalities in (i) and (ii), we have

$$\begin{aligned}
&V_1(x_{k+1}) - V_1(x_k) + \alpha\lambda(V_2(y_{k+1}) - V_2(y_k)) \\
&\quad \le (2\alpha-1)\|x_{k+1}-x_k\|_2^2 - 2\alpha\lambda x_{k+1}^T(g(y_k+x_k)-y^*) \\
&\qquad -\alpha\lambda\|y_{k+1}-y_k\|_2^2 + 2\alpha\lambda x_{k+1}^T(y_{k+1}-y^*) \\
&\quad = (2\alpha-1)\|x_{k+1}-x_k\|_2^2 - \alpha\lambda\|y_{k+1}-y_k\|_2^2 \\
&\qquad +2\alpha\lambda x_{k+1}^T(g(y_k+x_{k+1}) - g(y_k+x_k))
\end{aligned}$$

where the last equality holds since $y_{k+1} = g(y_k+x_{k+1})$.

Furthermore, we have

$$\begin{aligned}
&-\|y_{k+1}-y_k\|_2^2 + 2x_{k+1}^T(g(y_k+x_{k+1}) - g(y_k+x_k)) \\
&\quad = -\|y_{k+1}-y_k\|_2^2 \\
&\qquad + 2(x_{k+1}-x_k)^T(g(y_k+x_{k+1}) - g(y_k+x_k)) \\
&\qquad + 2x_k^T(g(y_k+x_{k+1}) - g(y_k+x_k)) \\
&\quad \le -\|y_{k+1}-y_k\|_2^2 + 2\|x_{k+1}-x_k\|_2^2 \\
&\qquad + 2x_k^T(y_{k+1} - g(y_k+x_k)) \\
&\quad = -\|y_{k+1}-y_k\|_2^2 + 2\|x_{k+1}-x_k\|_2^2 \\
&\qquad + 2(y_k+x_k-y_{k+1})^T(y_{k+1} - g(y_k+x_k)) \\
&\qquad + 2(y_{k+1}-y_k)^T(y_{k+1} - g(y_k+x_k)) \\
&\quad \le -\|y_{k+1}-y_k\|_2^2 + 2\|x_{k+1}-x_k\|_2^2 \\
&\qquad - 2\|y_{k+1} - g(y_k+x_k)\|_2^2 \\
&\qquad + 2(y_{k+1}-y_k)^T(y_{k+1} - g(y_k+x_k)) \\
&\quad = 2\|x_{k+1}-x_k\|_2^2 - \|y_{k+1} - g(y_k+x_k)\|_2^2 \\
&\qquad - \|y_{k+1}-y_k\|_2^2 - \|y_{k+1} - g(y_k+x_k)\|_2^2 \\
&\qquad + 2(y_{k+1}-y_k)^T(y_{k+1} - g(y_k+x_k)) \\
&\quad = 2\|x_{k+1}-x_k\|_2^2 - \|y_{k+1} - g(y_k+x_k)\|_2^2 \\
&\qquad - \|y_{k+1} - y_k - y_{k+1} + g(y_k+x_k)\|_2^2 \\
&\quad = 2\|x_{k+1}-x_k\|_2^2 - \|y_{k+1} - g(y_k+x_k)\|_2^2 \\
&\qquad - \|y_k - g(y_k+x_k)\|_2^2
\end{aligned}$$

where the first inequality holds due to $(x_{k+1}-x_k)^T(g(y_k+x_{k+1}) - g(y_k+x_k)) = (y_k+x_{k+1}-y_k-x_k)^T(g(y_k+x_{k+1}) - g(y_k+x_k)) \le \|y_k+x_{k+1}-y_k-x_k\|_2^2 = \|x_{k+1}-x_k\|_2^2$, and the second inequality holds due to $(y_{k+1}-y_k-x_k)^T(y_{k+1} - g(y_k+x_k)) \ge \|y_{k+1} - g(y_k+x_k)\|_2^2$.

Then, we have

$$\begin{aligned}
&V_1(x_{k+1}) - V_1(x_k) + \alpha\lambda(V_2(y_{k+1}) - V_2(y_k)) \\
&\quad \le (2\alpha - 1 + 2\alpha\lambda)\|x_{k+1}-x_k\|_2^2 \\
&\qquad - \alpha\lambda\|y_{k+1} - g(y_k+x_k)\|_2^2 - \alpha\lambda\|y_k - g(y_k+x_k)\|_2^2.
\end{aligned}$$

This completes the proof.

REFERENCES

[1] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[2] Y. Xu, Z. Zhong, J. Yang, J. You, and D. Zhang, "A new discriminative sparse representation method for robust face recognition via l_2 regularization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2233–2242, Oct. 2017.

[3] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, Nov. 2013.

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 09:26:13 UTC from IEEE Xplore. Restrictions apply.

[4] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 3360–3367.

[5] A. Wagner, J. Wright, A. Ganesh, Z. Zhou, H. Mobahi, and Y. Ma, "Toward a practical face recognition system: Robust alignment and illumination by sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 2, pp. 372–386, Feb. 2012.

[6] M. A. Khajehnejad, W. Xu, A. S. Avestimehr, and B. Hassibi, "Analyzing weighted ℓ_1 minimization for sparse recovery with nonuniform sparse models," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 1985–2001, May 2011.

[7] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comput.*, vol. 24, no. 2, pp. 227–234, 1995.

[8] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, 2001.

[9] R. H. Byrd, M E. Hribar, and J. Nocedal, "An interior point algorithm for large-scale nonlinear programming," *SIAM J. Optim.*, vol. 9, no. 4, pp. 877–900, 1999.

[10] D. M. Malioutov, M. Cetin, and A. S. Willsky, "Homotopy continuation for sparse signal representation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 5, Mar. 2005, pp. 733–736.

[11] M. R. Osborne, B. Presnell, and B. A. Turlach, "A new approach to variable selection in least squares problems," *IMA J. Numer. Anal.*, vol. 20, no. 3, pp. 389–403, 2000.

[12] S. Becker, J. Bobin, and E. J. Candès, "NESTA: A fast and accurate first-order method for sparse recovery," *SIAM J. Imag. Sci.*, vol. 4, no. 1, pp. 1–39, 2011.

[13] Y. Wang, G. Zhou, L. Caccetta, and W. Liu, "An alternative Lagrange-dual based algorithm for sparse signal reconstruction," *IEEE Trans. Signal Process.*, vol. 59, no. 4, pp. 1895–1901, Apr. 2011.

[14] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 4, pp. 586–597, Dec. 2007.

[15] Z. Jia, "Refined iterative algorithms based on Arnoldi's process for large unsymmetric eigenproblems," *Linear Algebra Appl.*, vol. 259, pp. 1–23, Jul. 1997.

[16] Q. Liu and J. Wang, "A one-layer recurrent neural network with a discontinuous hard-limiting activation function for quadratic programming," *IEEE Trans. Neural Netw.*, vol. 19, no. 4, pp. 558–570, Apr. 2008.

[17] X. He, C. Li, T. Huang, C. Li, and J. Huang, "A recurrent neural network for solving bilevel linear programming problem," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 4, pp. 824–830, Apr. 2014.

[18] M. J. Pérez-Ilzarbe, "Convergence analysis of a discrete-time recurrent neural network to perform quadratic real optimization with bound constraints," *IEEE Trans. Neural Netw.*, vol. 9, no. 6, pp. 1344–1351, Nov. 1998.

[19] X. Hu and J. Wang, "Solving the assignment problem using continuous-time and discrete-time improved dual networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 5, pp. 821–827, May 2012.

[20] Q. Liu and J. Wang, "A one-layer projection neural network for non-smooth optimization subject to linear equalities and bound constraints," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 5, pp. 812–824, May 2013.

[21] Q. Liu, T. Huang, and J. Wang, "One-layer continuous- and discrete-time projection neural networks for solving variational inequalities and related optimization problems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 7, pp. 1308–1318, Jul. 2014.

[22] Q. Liu and J. Wang, "A projection neural network for constrained quadratic minimax optimization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 11, pp. 2891–2900, Nov. 2015.

[23] Y. Zhang, J. Wang, and Y. Xu, "A dual neural network for bi-criteria kinematic control of redundant manipulators," *IEEE Trans. Robot. Autom.*, vol. 18, no. 6, pp. 923–931, Dec. 2002.

[24] Y. Xia and J. Wang, "A one-layer recurrent neural network for support vector machine learning," *IEEE Trans. Syst., Man, Cybern. B. Cybern.*, vol. 34, no. 2, pp. 1261–1269, Apr. 2004.

[25] Q. Liu and J. Wang, "L_1-minimization algorithms for sparse signal reconstruction based on a projection neural network," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 3, pp. 698–707, Mar. 2016.

[26] D. K. Pal, F. Juefei-Xu, and M. Savvides, "Discriminative invariant kernel features: A bells-and-whistles-free approach to unsupervised face recognition and pose estimation," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 5590–5599.

[27] Y. Wen, Z. Li, and Y. Qiao, "Latent factor guided convolutional neural networks for age-invariant face recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 4893–4901.

[28] Y. Sun, X. Wang, and X. Tang, "Sparsifying neural network connections for face recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 4856–4864.

[29] M. Iacopo, R. Stephen, M. Gerard, and N. Prem, "Pose-aware face recognition in the wild," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 4838–4846.

[30] G. B. Huang, H. Lee, and E. Learned-Miller, "Learning hierarchical representations for face verification with convolutional deep belief networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2518–2525.

[31] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10,000 classes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1891–1898.

[32] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1988–1996.

[33] J. Hu, J. Lu, and Y.-P. Tan, "Discriminative deep metric learning for face verification in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1875–1882.

[34] Z. Ding and F. Yun, "Robust multiview data analysis through collective low-rank subspace," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 5, pp. 1986–1997, May 2018.

[35] S. Li, K. Li, and F. Yun, "Self-taught low-rank coding for visual learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 3, pp. 645–656, Mar. 2018.

[36] Y. Kong, M. Shao, K. Li, and F. Yun, "Probabilistic low-rank multitask learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 3, pp. 670–680, Mar. 2018.

[37] L. Zhang, M. Yang, and X. Feng, "Sparse representation or collaborative representation: Which helps face recognition?" in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 471–478.

[38] Y. Meng, Z. Lei, Y. Jian, and D. Zhang, "Robust sparse coding for face recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 625–632.

[39] M. Yang, L. Zhang, J. Yang, and D. Zhang, "Regularized robust coding for face recognition," *IEEE Trans. Image Process.*, vol. 22, no. 5, pp. 1753–1766, May 2013.

[40] D. Kinderlehrer and G. Stampacchia, *An Introduction to Variational Inequalities and Their Applications*, vol. 31. Philadelphia, PA, USA: SIAM, 1980.

[41] E. J. Candés, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Commun. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, 2006.

[42] J. P. LaSalle, *The Stability of Dynamical Systems*, vol. 25. Philadelphia, PA, USA: SIAM, 1976.

[43] A. Y. Yang, Z. Zhou, A. G. Balasubramanian, S. S. Sastry, and Y. Ma, "Fast ℓ_1-minimization algorithms for robust face recognition," *IEEE Trans. Image Process.*, vol. 22, no. 8, pp. 3234–3246, Aug. 2013.

[44] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2479–2493, Jul. 2009.

[45] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 815–823.

[46] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1701–1708.

[47] F. Wang, X. Xiang, J. Cheng, and A. L. Yuille, "NormFace: L_2 hypersphere embedding for face verification," in *Proc. ACM Conf. Multimedia*, Oct. 2017, pp. 1041–1049.

[48] H. Guo, R. Wang, J. Choi, and L. S. Davis, "Face verification using sparse representations," in *Proc. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2012, pp. 37–44.

[49] E. G. Ortiz, A. Wright, and M. Shah, "Face recognition in movie trailers via mean sequence sparse representation-based classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3531–3538.

[50] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Univ. Massachusetts Amherst, Amherst, MA, USA, Tech. Rep., 2007, pp. 7–49, vol. 1.

[51] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1499–1503, Oct. 2016.

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 09:26:13 UTC from IEEE Xplore. Restrictions apply.

[52] L. Wolf, T. Hassner, and I. Maoz, "Face recognition in unconstrained videos with matched background similarity," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 529–534.
[53] A. S. Georghiades, P. N. Belhumeur, and D. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 643–660, Jun. 2001.
[54] A. M. Martínez and A. C. Kak, "PCA versus LDA," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 2, pp. 228–233, Feb. 2001.
[55] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression (PIE) database," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recognit.*, May 2002, pp. 46–51.
[56] W. Gao *et al.*, "The CAS-PEAL large-scale Chinese face database and baseline evaluations," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 38, no. 1, pp. 149–161, Jan. 2008.

Bingrong Xu received the B.S. degree from the Wuhan University of Technology, Wuhan, China, in 2011. She is currently pursuing the Ph.D. degree with the School of Automation, Huazhong University of Science and Technology, Wuhan.

Her current research interests include sparse representation, computer vision, machine learning, pattern recognition.

Qingshan Liu (S'07–M'08–SM'15) received the B.S. degree in mathematics from Anhui Normal University, Wuhu, China, in 2001, the M.S. degree in applied mathematics from Southeast University, Nanjing, China, in 2005, and the Ph.D. degree in automation and computer-aided engineering from the Chinese University of Hong Kong, Hong Kong, in 2008.

He is currently a Professor with the School of Mathematics, Southeast University, Nanjing. His current research interests include optimization theory and applications, artificial neural networks, computational intelligence, and multiagent systems.

Dr. Liu serves as an Associate Editor for the IEEE TRANSACTIONS ON CYBERNETICS and the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, and a member of the Editorial Board of *Neural Networks*.

Tingwen Huang (SM' 13) received the B.S. degree from Southwest University, Chongqing, China, in 1990, the M.S. degree from Sichuan University, Chengdu, China, in 1993, and the Ph.D. degree from Texas A&M University, College Station, TX, USA, in 2002.

He is currently a Professor of mathematics with Texas A&M University at Qatar, Doha, Qatar. His current research interests include dynamical systems, memristor, neural networks, complex networks, optimization and control, and traveling wave phenomena.

Authorized licensed use limited to: BEIHANG UNIVERSITY. Downloaded on June 15,2020 at 09:26:13 UTC from IEEE Xplore. Restrictions apply.