Discrete-Time Neural Network for Fast Solving Large Linear L_1 Estimation Problems and Its Application to Image Restoration

Youshen Xia, *Senior Member, IEEE*, Changyin Sun, and Wei Xing Zheng, *Senior Member, IEEE*

***Abstract*— There is growing interest in solving linear L_1 estimation problems for sparsity of the solution and robustness against non-Gaussian noise. This paper proposes a discrete-time neural network which can calculate large linear L_1 estimation problems fast. The proposed neural network has a fixed computational step length and is proved to be globally convergent to an optimal solution. Then, the proposed neural network is efficiently applied to image restoration. Numerical results show that the proposed neural network is not only efficient in solving degenerate problems resulting from the nonunique solutions of the linear L_1 estimation problems but also needs much less computational time than the related algorithms in solving both linear L_1 estimation and image restoration problems.**

***Index Terms*— Discrete-time neural network, global convergence, image restoration, large linear L_1 estimation problem.**

I. Introduction

It is well known that an important problem in linear regression models is to estimate a set of model parameters in the presence of background noise [1]. The least squares (LS) estimation method is considered the most suitable method for estimating the model parameters if the noise is Gaussian-distributed [2]. But, non-Gaussian noise usually arises in real application environments, and thus the LS estimator may be very poor in such cases. An attractive alternative method is the least absolute deviation (L_1) estimation method [3]–[5]. The L_1 estimator is efficient for non-Gaussian error distribution such as Laplace or Cauchy distribution. Moreover, it does not overemphasize large errors and is thus more robust than the LS estimator when the signal data contains outlier non-Gaussian noise errors. In recent years, another growing interest in L_1 norm is due to the sparsity properties of the solution. The estimator is in the form of a shrinkage estimator. The nonsmooth nature of the L_1 norm (singularity at the origin) restricts the estimation to a polyhedral region which ensures that the penalized likelihood has the sparsity property [6], [7]. Because of its excellent properties, the linear L_1 estimation method has been widely applied to many engineering applications, especially in the field of signal and image processing ([5], [8], [12], [13], [20]–[22]). Unlike the LS estimation problem, it is more difficult to solve the L_1 estimation problem since its objective function is nonsmooth. Thus, developing efficient algorithms for the linear L_1 estimation is desirable.

There are several types of numerical algorithms for solving unconstrained or constrained L_1 estimation problems. The first type is the descent methods [8]. It has a complex computational procedure. The least absolute shrinkage and selection operator algorithm [9]–[11] was introduced to overcome the numerical problem. The second type is the approximate method [12], [13]. This method needs to choose a scaling parameter for the estimator accuracy. The third type is to transform the L_1 estimation problem into a linear programming (LP) problem and then solve it by using the modified Karmarkar's algorithm and the disciplined convex programming algorithm [14], [15]. They are suitable for nondegenerate LP problems but may not work well when the L_1 estimation problem has nonunique solutions. The perturbation method [9], the modified descent method [16], and the projected iterative algorithm [17], [18] can handle the degenerate case but they involve a lot of computational cost for solving image restoration problems. The latest type is based on recurrent neural networks [19]. A continuous-time recurrent neural network for solving an L_1 estimation problem with special linear constraints was first presented in [20]. A continuous-time recurrent neural network for solving an L_1 estimation problem with general linear constraints was presented in [21] and a continuous-time compact recurrent neural network for solving an L_1 estimation problem with general linear constraints was presented in [22]. It was shown that they can deal with degeneracy problems arising in the constrained L_1 estimation methods and have low computational complexity. Yet, because of the continuous-time feature, these recurrent neural networks still involve a lot of computational cost and result in slow convergence rate when solving large-size L_1 estimation problems.

It is well known that image restoration problems are usually of large-size [23], [24]. The conventional regularization method can quickly obtain the regularization solution by the efficient use of fast Fourier transform. But the

Manuscript received January 28, 2011; revised January 8, 2012; accepted January 8, 2012. Date of publication April 5, 2012; date of current version May 2, 2012. This work was supported by the National Natural Science Foundation of China under Grant 61179037 and Grant 60875085, in part by the National Natural Science Foundation of China under Grant 61125306 and Grant 61034002, and in part by a Research Grant from the Australian Research Council.

Y. Xia is with the College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350108, China (e-mail: ysxia@fzu.edu.cn).

C. Sun is with the School of Automation, Southeast University, Nanjing 210096, China (e-mail: cysun@seu.edu.cn).

W. X. Zheng is with School of Computing, Engineering and Mathematics, University of Western Sydney, Sydney NSW 2751, Australia (e-mail: W.Zheng@uws.edu.au).

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.2012.2184800

regularization solution may be a good image estimate only under the assumption of both the optimal regularization parameter and the Gaussian noise environment [25]–[30]. To handle non-Gaussian noises, the high-order statistical method and the entropy estimation method were presented [18], [31], [32], where the optimal regularization parameter is still required. In order to relax the need of the optimal regularization parameter to be estimated, a generalized least absolute deviation (GLAD) estimation method and one cooperative recurrent neural network (CRNN) algorithm for image restoration were developed [20]. It was shown that the GLAD method can obtain the optimal image estimation with a nonoptimal regularization parameter and is robust against non-Gaussian noise. On the other hand, because the CRNN algorithm has a continuous-time feature, it has slow convergence rate since the image restoration problem to be estimated is of large size. Thus it is very desirable to develop a fast algorithm for image restoration [33].

An iterative method for solving large linear L_1 problem without constraints was developed in paper [38]. This paper proposes a discrete-time neural network algorithm for solving fast constrained linear L_1 estimation problems with large size. The proposed neural network is an iterative algorithm with a fixed and large step length and is shown to converge globally to an optimal solution under a fixed step length. Compared with the related algorithms, i.e., the iterative algorithm with varying-step length and the continuous-time neural network algorithm, the proposed iterative algorithm has a very fast convergence rate because of the large step length [34]. Furthermore, the proposed neural network is applied satisfactorily to an image restoration model. Numerical results show that the proposed neural network is not only efficient in solving degenerate problems resulting from the nonunique solutions of the large linear L_1 estimation problems with much less computational time than similar algorithms but also is efficient in solving large image restoration problems.

This paper is organized as follows. In Section II, the linear L_1 estimation problem and the solution algorithm are introduced and proposed. In Section III, the global convergence of the proposed discrete-time neural network is proven. In Section IV, the proposed neural network algorithm is applied for image restoration. Finally, some concluding remarks are given in Section V.

II. Linear L_1 Estimation Problem and Algorithm

A. Problem and Existing Algorithms

Consider the general linear L_1 estimation problem

$$\begin{aligned} \min \quad & \|\mathbf{Dx}-\mathbf{d}\|_1 \\ \text{s.t} \quad & \mathbf{Bx}=\mathbf{b}, \\ & \mathbf{Ax}\le \mathbf{p}, \quad \mathbf{x}\in X \end{aligned} \tag{1}$$

where $\|\cdot\|_1$ denotes l_1 norm given by $\|\mathbf{d}\|_1 = \sum_{i=1}^{m}|d_i|$, $X=\{\mathbf{x}\in R^n |\, \mathbf{l}\le\mathbf{x}\le\mathbf{h}\}$, $\mathbf{D}\in R^{m\times n}$, $\mathbf{B}\in R^{l\times n}$, $\mathbf{A}\in R^{r\times n}$, $\mathbf{x}\in R^n$, $\mathbf{d}\in R^m$, $\mathbf{b}\in R^l$, and $\mathbf{p}\in R^r$.

The considered L_1 problem is general because the several constrained L_1 estimation problems, discussed in [21] and [22], are its special cases. Many numerical algorithms for solving these unconstrained or constrained L_1 estimation problems have been developed, such as the descent method, the perturbation method, the modified descent method, the Huber M-estimator method, the LP method-based Karmarkar's approach, and the continuous-time recurrent neural networks. In particular, a compact cooperative neural network for solving (1) was recently presented as follows [22]:

$$\frac{d\mathbf{x}(t)}{dt} = \lambda\left\{E_1(t)-\mathbf{D}^T E_2(t)-\mathbf{B}^T E_3(t)-\mathbf{A}^T E_4(t)\right\} \tag{2a}$$
$$\frac{d\mathbf{y}(t)}{dt} = \lambda\left\{\mathbf{D}E_1(t)+E_2(t)\right\} \tag{2b}$$
$$\frac{d\mathbf{z}_I(t)}{dt} = \lambda\left\{\mathbf{B}E_1(t)+E_3(t)\right\} \tag{2c}$$
$$\frac{d\mathbf{z}_{II}(t)}{dt} = \lambda\left\{\mathbf{A}E_1(t)+E_4(t)\right\} \tag{2d}$$

where $\lambda > 0$ is a design constant,$\mathbf{x}(t), \mathbf{y}(t), \mathbf{z}_I(t), \mathbf{z}_{II}(t)$ are the state vector, $E_1(t) = g_X(\mathbf{x}(t) - \mathbf{D}^T\mathbf{y}(t) - \mathbf{B}^T\mathbf{z}_I(t) - \mathbf{A}^T\mathbf{z}_{II}(t)) - \mathbf{x}(t)$, $E_2(t) = g_{X_1}(\mathbf{y}(t)+\mathbf{Dx}(t)-\mathbf{d})-\mathbf{y}(t)$, $E_3(t) = \mathbf{Bx}(t)$, $E_4(t) = g_{X_2}(\mathbf{z}_{II}(t)+\mathbf{Ax}(t)-\mathbf{p})-\mathbf{z}_{II}(t)$, $X_2 = \{\mathbf{z}_{II}\in R^r \mid \mathbf{z}_{II}\ge 0\}$, $g_{X_2}(\mathbf{z}) = [g_{X_2}(z_1),\ldots,g_{X_2}(z_r)]^T$,$g_{X_2}(z_i) = \max\{0, z_i\}$,$X_1 = \{\mathbf{y}\in R^m \mid \max_i |y_i| \le 1\}$, $g_{X_1}(\mathbf{y}) = [g_{X_1}(y_1),\ldots,g_{X_1}(y_m)]^T$

$$g_{X_1}(z_i) = \begin{cases} -1 & z_i < -1 \\ z_i & -1\le z_i\le 1 \\ 1 & z_i > 1 \end{cases} \tag{3}$$

$g_X(\mathbf{x}) = [g_X(x_1),\ldots,g_X(x_n)]^T$ and

$$g_X(x_i) = \begin{cases} l_i & x_i < l_i \\ x_i & l_i\le x_i\le h_i \\ h_i & x_i > h_i. \end{cases} \tag{4}$$

It was shown that the CRNN can efficiently solve linearly constrained L_1 estimation problems, especially for degeneracy cases. Yet, because of the continuous-time feature, the CRNN will have a slow convergence rate when solving large-size L_1 estimation problems.

B. Proposed Discrete-Time Neural Network

In order to speed up convergence, we propose a discrete-time neural network for solving (1) as follows:

$$\begin{aligned}\mathbf{x}(k+1) = \mathbf{x}(k) + h\Big\{&F_1(k)-\hat{\mathbf{D}}^T F_2(k) \\ &-\hat{\mathbf{B}}^T F_3(k)-\hat{\mathbf{A}}^T F_4(k)\Big\}\end{aligned} \tag{5a}$$
$$\mathbf{y}(k+1) = \mathbf{y}(k) + h\left\{\hat{\mathbf{D}}F_1(k)+F_2(k)\right\} \tag{5b}$$
$$\mathbf{z}_I(k+1) = \mathbf{z}_I(k) + h\left\{\hat{\mathbf{B}}F_1(k)+F_3(k)\right\} \tag{5c}$$
$$\mathbf{z}_{II}(k+1) = \mathbf{z}_{II}(k) + h\left\{\hat{\mathbf{A}}F_1(k)+F_4(k)\right\} \tag{5d}$$

where $h > 0$ is a fixed step length, $F_1(k) = g_X(\mathbf{x}(k) - \hat{\mathbf{D}}^T\mathbf{y}(k) - \hat{\mathbf{B}}^T\mathbf{z}_I(k) - \hat{\mathbf{A}}^T\mathbf{z}_{II}(k)) - \hat{\mathbf{x}}(k)$, $F_2(t) = g_{X_1}(\mathbf{y}(k) + \hat{\mathbf{D}}\mathbf{x}(k) - \hat{\mathbf{d}}) - \mathbf{y}(k)$, $F_3(k) = \hat{\mathbf{B}}\mathbf{x}(k) - \hat{\mathbf{b}}$, $F_3(k) = g_{X_2}(\mathbf{z}_{II}(k) + \hat{\mathbf{A}}\mathbf{x}(k) - \mathbf{p}) - \mathbf{z}_{II}(k)$, $(\mathbf{x}(k), \mathbf{y}(k), \mathbf{z}_I(k), \mathbf{z}_{II}(k)) \in R^n\times R^m\times R^l\times R^r$, $\hat{\mathbf{D}} = \mathbf{D}/\|\mathbf{D}\|_2$, $\hat{\mathbf{B}} = \mathbf{B}/\|\mathbf{B}\|_2$, $\hat{\mathbf{A}} = \mathbf{A}/\|\mathbf{A}\|_2$, $\hat{\mathbf{d}} = \mathbf{d}/\|\mathbf{D}\|_2$, $\hat{\mathbf{b}} = \mathbf{b}/\|\mathbf{B}\|_2$, $\hat{\mathbf{p}} = \mathbf{p}/\|\mathbf{A}\|_2$. It is easy to see that the proposed discrete-time neural network has the same

normalization of weights

computational complexity as the CRNN. More importantly, computed examples will show that the proposed neural network can speed up the CRNN greatly.

It should be pointed out that the proposed algorithm can be viewed as an iterative algorithm with a fixed step length. Thus the computational step length may be chosen as the varying-step-length-like project-gradient algorithms [35]

$$h(k) = \frac{\|F_1(k)\|_2^2 + \|F_2(k)\|_2^2 + \|F_3(k)\|_2^2 + \|F_4(k)\|_2^2}{2\|F(k)\|_2^2} \quad (6)$$

where

$$F(\mathbf{u(k)}) = \begin{pmatrix} F_1(k) - \hat{\mathbf{D}}^T F_2(k) - \hat{\mathbf{B}}^T F_3(k) - \hat{\mathbf{A}}^T F_4(k) \\ \hat{\mathbf{D}} F_1(k) + F_2(k) \\ \hat{\mathbf{B}} F_1(k) + F_3(k) \\ \mathbf{A} F_1(k) + F_4(k) \end{pmatrix}$$

and $\mathbf{u}(k) = [\mathbf{x}(k), \mathbf{y}(k), \mathbf{z}_I(k), \mathbf{z}_{II}]^T$. However, the varying-step-length-based iterative algorithm has a slow convergence speed because the varying step length could be very small for the L_1 estimation problem. Computed results will show that the proposed iterative algorithm with the fixed step length has higher convergence speed than the iterative algorithm with varying step length.

Remark 1: In terms of the nature of connectivity, an artificial neural network (ANN) can be the feedforward network or the recurrent network, in term of how the synaptic weights are obtained, an ANN can be classified into fixed-weight networks, unsupervised networks, and supervised networks. The proposed neural network can be viewed as a fixed-weight recurrent network. The LAD problem is reformulated into a set of piecewise equations and is then mapped into the proposed neural network where the optimal solution corresponds to the equilibrium point $F(\mathbf{u}^*) = 0$ of the proposed neural network. The optimal solution of the LAD problem can be obtained by tracking the solution trajectory (recurrent iteration).

III. Global Convergence of the Proposed Algorithm

To show the global convergence of the proposed neural network, we first introduce two lemmas.

Lemma 1: Let

$$\mathbf{C} = \begin{pmatrix} \mathbf{I_1} & -\hat{\mathbf{D}}^\mathbf{T} & -\hat{\mathbf{B}}^\mathbf{T} & -\hat{\mathbf{A}}^\mathbf{T} \\ \hat{\mathbf{D}} & \mathbf{I_2} & \mathbf{O_{23}} & \mathbf{O_{24}} \\ \hat{\mathbf{B}} & \mathbf{O_{23}^T} & \mathbf{I_3} & \mathbf{O_{34}} \\ \hat{\mathbf{A}} & \mathbf{O_{24}^T} & \mathbf{O_{34}^T} & \mathbf{I_4} \end{pmatrix} \quad (7)$$

where $\hat{\mathbf{A}}$, $\hat{\mathbf{B}}$, and $\hat{\mathbf{D}}$ are defined in (5), $\mathbf{O_{23}} \in \mathbf{R^{m\times l}}$, $\mathbf{O_{24}} \in \mathbf{R^{m\times r}}$ and $\mathbf{O_{34}} \in \mathbf{R^{l\times r}}$ are zero matrices, and $\mathbf{I_1} \in \mathbf{R^{n\times n}}$, $\mathbf{I_2} \in \mathbf{R^{m\times m}}$, $\mathbf{I_3} \in \mathbf{R^{l\times l}}$, and $\mathbf{I_4} \in \mathbf{R^{r\times r}}$ are unit identity matrices. Then

$$\|\mathbf{C}\|_2 \le 2.$$

Proof: Let $\mathbf{W} = \hat{\mathbf{D}}^\mathbf{T}\hat{\mathbf{D}} + \hat{\mathbf{B}}^\mathbf{T}\hat{\mathbf{B}} + \mathbf{A}^\mathbf{T}\mathbf{A}$. Then

$$\mathbf{CC^T} = \begin{pmatrix} \mathbf{I_1} + \mathbf{W} & \mathbf{O_1} & \mathbf{O_2} & \mathbf{O_2} \\ \mathbf{O_1^T} & \mathbf{I_2} + \hat{\mathbf{D}}\hat{\mathbf{D}}^\mathbf{T} & \hat{\mathbf{D}}\hat{\mathbf{B}}^\mathbf{T} & \hat{\mathbf{A}}^\mathbf{T} \\ \mathbf{O_2^T} & \hat{\mathbf{B}}\hat{\mathbf{D}}^\mathbf{T} & \mathbf{I_3} + \hat{\mathbf{B}}\hat{\mathbf{B}}^\mathbf{T} & \hat{\mathbf{B}}\hat{\mathbf{A}}^\mathbf{T} \\ \mathbf{O_3^T} & \hat{\mathbf{A}}\hat{\mathbf{D}}^\mathbf{T} & \hat{\mathbf{A}}\hat{\mathbf{B}}^\mathbf{T} & \mathbf{I_4} + \hat{\mathbf{A}}\hat{\mathbf{A}}^\mathbf{T} \end{pmatrix}$$

$$\begin{pmatrix} \hat{\mathbf{D}}\hat{\mathbf{D}}^\mathbf{T} & \hat{\mathbf{D}}\hat{\mathbf{B}}^\mathbf{T} & \hat{\mathbf{D}}\hat{\mathbf{A}}^\mathbf{T} \\ \hat{\mathbf{B}}\hat{\mathbf{D}}^\mathbf{T} & \hat{\mathbf{B}}\hat{\mathbf{B}}^\mathbf{T} & \hat{\mathbf{B}}\hat{\mathbf{A}}^\mathbf{T} \\ \hat{\mathbf{A}}\hat{\mathbf{D}}^\mathbf{T} & \hat{\mathbf{A}}\hat{\mathbf{B}}^\mathbf{T} & \hat{\mathbf{A}}\hat{\mathbf{A}}^\mathbf{T} \end{pmatrix} = \begin{bmatrix} \hat{\mathbf{D}} \\ \hat{\mathbf{B}} \\ \hat{\mathbf{A}} \end{bmatrix} \left[\hat{\mathbf{D}}^\mathbf{T}, \hat{\mathbf{B}}^\mathbf{T}, \hat{\mathbf{A}}^\mathbf{T}\right]$$

where $\mathbf{O_1} \in \mathbf{R^{n\times m}}$, $\mathbf{O_2} \in \mathbf{R^{n\times l}}$, $\mathbf{O_3} \in \mathbf{R^{m\times n}}$ are zero matrices, and

$$\left\| \begin{bmatrix} \hat{\mathbf{D}} \\ \hat{\mathbf{B}} \\ \hat{\mathbf{A}} \end{bmatrix} \left[\hat{\mathbf{D}}^\mathbf{T}, \hat{\mathbf{B}}^\mathbf{T}, \hat{\mathbf{A}}^\mathbf{T}\right] \right\|_2 = \left\| \hat{\mathbf{D}}^\mathbf{T}\hat{\mathbf{D}} + \hat{\mathbf{B}}^\mathbf{T}\hat{\mathbf{B}} + \hat{\mathbf{A}}^\mathbf{T}\hat{\mathbf{A}} \right\|_2 = \|\mathbf{W}\|_2.$$

Then

$$\|\mathbf{C}\|_2^2 = \left\|\mathbf{CC^T}\right\|_2 \le 1 + \|\mathbf{W}\|_2 = 4.$$

∎

Lemma 2: Let $\{\mathbf{x}(k), \mathbf{y}(k), \mathbf{z}_I(k), \mathbf{z}_{II}(k)\}$ be the sequence generated by the proposed neural network. Then

$$\begin{aligned} &(\mathbf{x}(k) - \mathbf{x}^*)^T F_1(k) + (\mathbf{y}(k) - \mathbf{y}^*)^T \hat{\mathbf{D}} F_1(k) \\ &\quad + (\mathbf{z}_I(k) - \mathbf{z}_I^*)^T \hat{\mathbf{B}} F_1(k) + (\mathbf{z}_{II}(k) - \mathbf{z}_{II}^*)^T \hat{\mathbf{A}} F_1(k) \\ &\le -\|F_1(k)\|_2^2 - (\mathbf{x(k)} - \mathbf{x}^*)^T \hat{\mathbf{D}}^T (\mathbf{y}(k) - \mathbf{y}^*) \\ &\quad - F_3(k)^T (\mathbf{z}_I(k) - \mathbf{z}_I^*) - (\mathbf{x(k)} - \mathbf{x}^*)^T \hat{\mathbf{A}}^T (\mathbf{z}_{II}(k) - \mathbf{z}_{II}^*) \quad (8) \\ &(\mathbf{x}(k) - \mathbf{x}^*)^T \hat{\mathbf{D}}^T (-F_2(k) + (\mathbf{y}(k) - \mathbf{y}^*)^T F_2(k) \\ &\le -\|F_2(k)\|_2^2 + (\mathbf{x}(k) - \mathbf{x}^*)^T \hat{\mathbf{D}}^T (\mathbf{y}(k) - \mathbf{y}^*) \quad (9) \\ &(\mathbf{x}(k) - \mathbf{x}^*)^T \hat{\mathbf{A}}^T (-F_3(k)) + (\mathbf{z}_{II}(k) - \mathbf{z}_{II}^*)^T F_3(k) \\ &\le -\|F_3(k)\|_2^2 + (\mathbf{x}(k) - \mathbf{x}^*)^T \hat{\mathbf{A}}^T (\mathbf{z}_{II}(k) - \mathbf{z}_{II}^*) \quad (10) \end{aligned}$$

where $F_1(k)$, $F_2(k)$, $F_3(k)$, and $F_4(k)$ are defined in (3), $(\hat{\mathbf{x}}^*, \hat{\mathbf{y}}^*, \hat{\mathbf{z}}_I^*, \hat{\mathbf{z}}_{II}^*) \in R^n \times R^m \times R^l \times R^r$ satisfies

$$\begin{cases} \hat{\mathbf{B}}\mathbf{x} = \mathbf{b}, \\ \mathbf{y} = g_{X_1}(\mathbf{y} + \hat{\mathbf{D}}\mathbf{x} - \hat{\mathbf{d}}), \\ \mathbf{z}_{II} = g_{X_2}(\mathbf{z}_{II} + \hat{\mathbf{A}}\mathbf{x} - \hat{\mathbf{p}}), \\ \mathbf{x} = g_X(\mathbf{x} - \hat{\mathbf{D}}^T\mathbf{y} - \hat{\mathbf{B}}^T\mathbf{z}_I - \hat{\mathbf{A}}^T\mathbf{z}_{II}). \end{cases} \quad (11)$$

Moreover, $\hat{\mathbf{x}}^*$ is an optimal solution of (1).

Proof: From [21], we know that $\mathbf{x}^* \in R^n$ is an optimal solution to (1) if and only if there exists $(\mathbf{y}^*, \mathbf{z}_I^*, \mathbf{z}_{II}^*) \in R^m \times R^l \times R^r$ such that $(\mathbf{x}^*, \mathbf{y}^*, \mathbf{z}_I^*, \mathbf{z}_{II}^*)$ satisfies the following four equations:

$$\begin{cases} \mathbf{B}\mathbf{x} = \mathbf{b}, \\ \mathbf{y} = g_{X_1}(\mathbf{y} + \mathbf{D}\mathbf{x} - \mathbf{d}), \\ \mathbf{z}_{II} = g_{X_2}(\mathbf{z}_{II} + \mathbf{A}\mathbf{x} - \mathbf{p}), \\ \mathbf{x} = g_X(\mathbf{x} - \mathbf{D}^T\mathbf{y} - \mathbf{B}^T\mathbf{z}_I - \mathbf{A}^T\mathbf{z}_{II}). \end{cases} \quad (12)$$

Thus, we have that $(\hat{\mathbf{x}}^*, \hat{\mathbf{y}}^*, \hat{\mathbf{z}}_I^*, \hat{\mathbf{z}}_{II}^*)$ satisfies (11) if and only if $\hat{\mathbf{x}}^*$ is an optimal solution of the following estimation problem

$$\begin{aligned} \min \quad & \|\hat{\mathbf{D}}\mathbf{x} - \hat{\mathbf{d}}\|_1 \\ \text{s.t} \quad & \hat{\mathbf{B}}\mathbf{x} = \hat{\mathbf{b}}, \\ & \hat{\mathbf{A}}\mathbf{x} \le \hat{\mathbf{p}}, \quad \mathbf{x} \in X. \end{aligned} \quad (13)$$

It is easy to see that (13) is equivalent to (1). So, $\hat{\mathbf{x}}^*$ is also an optimal solution of (1). Furthermore, similar to analysis given in [22], based on (11) we can obtain inequalities (8)–(10), respectively. ∎

Now we give the global convergence of the proposed neural network.

Theorem 1: Let $0 < h < 1/2$. Then the proposed algorithm will converge globally to the optimal solution of (1).

Proof: Let $\{\mathbf{x}(k), \mathbf{y}(k), \mathbf{z}_I(k), \mathbf{z}_{II}(k)\}$ be the sequence generated by the proposed algorithm. Then (5) can be rewritten as

$$\begin{pmatrix} \mathbf{x}(k+1) \\ \mathbf{y}(k+1) \\ \mathbf{z}_I(k+1) \\ \mathbf{z}_{II}(k+1) \end{pmatrix} = \begin{pmatrix} \mathbf{x}(k) \\ \mathbf{y}(k) \\ \mathbf{z}_I(k) \\ \mathbf{z}_{II}(k) \end{pmatrix} + h \begin{pmatrix} F_1(k) + \hat{\mathbf{D}}^T F_2(k) - \hat{\mathbf{B}}^T F_3(k) - \hat{\mathbf{A}}^T F_4(k) \\ \hat{\mathbf{D}} F_1(k) + F_2(k) \\ \hat{\mathbf{B}} F_1(k) + F_3(k) \\ \hat{\mathbf{A}} F_1(k) + F_4(k) \end{pmatrix}.$$

For convenience of discussion, we denote $\mathbf{u}(k) = [\mathbf{x}(k), \mathbf{y}(k), \mathbf{z}_I(k), \mathbf{z}_{II}(k)]^T$ and

$$F(\mathbf{u}(\mathbf{k})) = \begin{pmatrix} F_1(k) - \hat{\mathbf{D}}^T F_2(k) - \hat{\mathbf{B}}^T F_3(k) - \hat{\mathbf{A}}^T F_4(k) \\ \hat{\mathbf{D}} F_1(k) + F_2(k) \\ \hat{\mathbf{B}} F_1(k) + F_3(k) \\ \mathbf{A} F_1(k) + F_4(k) \end{pmatrix}. \quad (14)$$

Then $\mathbf{u}(k+1) = \mathbf{u}(k) + hF(\mathbf{u}(\mathbf{k}))$ and

$$\begin{aligned}
&(\mathbf{u}(k) - \mathbf{u}^*)^T F(\mathbf{u}(k)) = \\
&(\mathbf{x}(k) - \mathbf{x}^*)^T (F_1(k) - \hat{\mathbf{D}}^T F_2(k) - \hat{\mathbf{B}}^T F_3(k) - \hat{\mathbf{A}}^T F_4(k)) \\
&\quad + (\mathbf{y}(k) - \mathbf{y}^*)^T (\hat{\mathbf{D}} F_1(k) + F_2(k)) \\
&\quad + (\mathbf{z}_{II}(k) - \mathbf{z}_{II}^*)^T (\hat{\mathbf{A}} F_1(k) + F_4(k) \\
&\quad + (\mathbf{z}_I(k) - \mathbf{z}_I^*)^T (\hat{\mathbf{B}} F_1(k) + F_3(k))
\end{aligned}$$

where $\mathbf{u}^* = (\mathbf{x}^*, \mathbf{y}^*, \mathbf{z}_I^*, \mathbf{z}_{II}^*)$ is a solution of (11). Note that $\hat{\mathbf{B}}(\mathbf{x}(k) - \mathbf{x}^*) = \hat{\mathbf{B}}\mathbf{x}(k) - \hat{\mathbf{b}}$. Adding (8)–(10) we obtain

$$\begin{aligned}
&(\mathbf{x}(k) - \mathbf{x}^*)^T F_1(k) + (\mathbf{y}(k) - \mathbf{y}^*)^T \hat{\mathbf{D}} F_1(k) \\
&\quad + (\mathbf{z}_I(k) - \mathbf{z}_I^*)^T \mathbf{B} F_1(k) + (\mathbf{z}_{II}(k) - \mathbf{z}_{II}^*)^T \mathbf{A} F_1(k) \\
&\quad + (\mathbf{z}_I(k) - \mathbf{z}_I^*)^T (-F_3(k)) \\
&-(\mathbf{x}(k) - \mathbf{x}^*)^T \mathbf{D}^T F_2(k) - (\mathbf{y}(k) - \mathbf{y}^*)^T (-F_2(k)) \\
&-(\mathbf{x}(k) - \mathbf{x}^*)^T \mathbf{A}^T F_3(k) - (\mathbf{z}_{II}(k) - \mathbf{z}_{II}^*)^T (-F_3(k)) \\
&\le -\|F_1(k)\|_2^2 - \|F_2(k)\|_2^2 - \|F_3(k)\|_2^2 - \|F_4(k)\|_2^2. \quad (15)
\end{aligned}$$

Thus

$$\begin{aligned}
&(\mathbf{u}(k) - \mathbf{u}^*)^T F(\mathbf{u}(k)) \le \\
&\quad - \|F_1(k)\|_2^2 - \|F_2(k)\|_2^2 - \|F_3(k)\|_2^2 - \|F_4(k)\|_2.
\end{aligned}$$

Hence

$$\begin{aligned}
\left\|\mathbf{u}(k+1) - \mathbf{u}^*\right\|_2^2 &= \left\|\mathbf{u}(k) - \mathbf{u}^*\right\|_2^2 + 2h\left(\mathbf{u}(k) - \mathbf{u}^*\right)^T F(k) \\
&+ h^2 \|F(\mathbf{u}(k))\|_2^2 \le \left\|\left(\mathbf{u}(k) - \mathbf{u}^*\right)\right\|^2 + h^2 \|F(\mathbf{u}(k))\|^2 \\
&-2h\left\{\|F_1(k)\|_2^2 + \|F_2(k)\|_2^2 + \|F_3(k)\|_2^2 + \|F_4(k)\|_2\right\}.
\end{aligned}$$

By Lemma 1, we get

$$\begin{aligned}
&\|F(\mathbf{u}(k))\|_2^2 \\
&\quad \le 2\left\{\|F_1(k)\|_2^2 + \|F_2(k)\|_2^2 + \|F_3(k)\|_2^2 + \|F_4(k)\|_2\right\}.
\end{aligned}$$

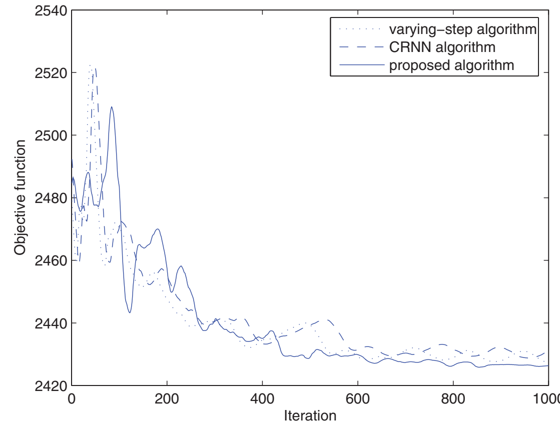

Fig. 1. Behavior of objective function as iterative number for Example 1.

Thus

$$\begin{aligned}
&\|\mathbf{u}(k+1) - \mathbf{u}^*\|^2 \le \|\mathbf{u}(k) - \mathbf{u}^*\|^2 - 2(1-h)h\{\|F_1(k)\|_2^2 \\
&+ \|F_2(k)\|_2^2 + \|F_3(k)\|_2^2 + \|F_4(k)\|_2^2\} < \|\mathbf{u}(k) - \mathbf{u}^*\|^2 \\
&\lim_{k\to\infty} \mathbf{u}(k) = \hat{\mathbf{u}} = [\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}_I, \hat{\mathbf{z}}_{II}]^T. \quad (16)
\end{aligned}$$

Then $\lim_{k\to\infty} \|F_1(k)\|_2^2 = \lim_{k\to\infty} \|F_2(k)\|_2^2 = \lim_{k\to\infty} \|F_3(k)\|_2^2 = \lim_{k\to\infty} \|F_4(k)\|_2^2 = 0$. It implies that $[\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}_I, \hat{\mathbf{z}}_{II}]^T$ satisfies

$$\begin{cases}
\hat{\mathbf{B}}\mathbf{x} = \mathbf{b}, \\
\mathbf{y} = g_{X_1}(\mathbf{y} + \hat{\mathbf{D}}\mathbf{x} - \hat{\mathbf{d}}), \\
\mathbf{z}_{II} = g_{X_2}(\mathbf{z}_{II} + \hat{\mathbf{A}}\mathbf{x} - \hat{\mathbf{p}}), \\
\mathbf{x} = g_X(\mathbf{x} - \hat{\mathbf{D}}^T\mathbf{y} - \hat{\mathbf{B}}^T\mathbf{z}_I - \hat{\mathbf{A}}^T\mathbf{z}_{II}).
\end{cases}$$

So, $\hat{\mathbf{x}}$ is an optimal solution of (11) and is also an optimal solution of (1). Therefore, the proposed algorithm is globally convergent to an optimal solution of (1). ■

Finally, it should be pointed out that the condition of Theorem 1 is sufficient. So, the proposed neural network may converge when $h > 0.5$. In order to demonstrate the effectiveness and efficiency of the proposed algorithm, we first give an example. The simulation is conducted in MATLAB. Because the continuous-time neural network algorithm was shown to be advantageous over the LP-based algorithm in solving degenerate problems resulting from the nonunique solutions of the L_1 estimation problem [21], we here compare the proposed algorithm with the related algorithms, i.e., the iterative algorithm with varying-step length in (6) and the continuous-time neural network algorithm.

Example 1: Consider the linearly constrained L_1 estimation problem (1), where $\mathbf{D}$ is the $m \times n$ random matrix, $\mathbf{B}$ is the $l \times n$ random matrix, $\mathbf{A}$ is the $r \times n$ random matrix, $\mathbf{d}$ is the m dimensional random vector, and $\mathbf{p}$ is the r-dimensional random vector. We choose a n-dimensional random integer vector as a feasible solution. We construct a 1-D vector $\mathbf{b}$ satisfying linear constraints, and construct the inequality constants by adding an uniformly distributed random vector. For comparison, we employ the proposed neural network with a fixed step length $h = 0.7$, the neural network with varying-step length defined in (6), and the continuous-time neural

TABLE I

COMPUTED RESULTS OF THE THREE ALGORITHMS IN EXAMPLE 1

Algorithm	Problem size	Objective function value	CPU time (s)
CRNN algorithm	$m = 400,\ n = 300,\ l = 100,\ r = 50$	2.395×10^3	227
Proposed algorithm	$m = 400,\ n = 300,\ l = 100,\ r = 50$	2.395×10^3	63.3
Varying-step algorithm	$m = 400,\ n = 300,\ l = 100,\ r = 50$	2.395×10^3	180.3

Algorithm	Problem size	Objective function value	CPU time (s)
CRNN algorithm	$m = 300,\ n = 200,\ l = 50,\ r = 20$	1.4338×10^3	86.2
Proposed algorithm	$m = 300,\ n = 200,\ l = 50,\ r = 20$	1.4334×10^3	38.3
Varying-step algorithm	$m = 300,\ n = 200,\ l = 50,\ r = 20$	1.434×10^3	103.9

Algorithm	Problem size	Objective function value	CPU time (s)
CRNN algorithm	$m = 300,\ n = 100,\ l = 50,\ r = 20$	1.637×10^3	131.5
Proposed algorithm	$m = 300,\ n = 100,\ l = 50,\ r = 20$	1.637×10^3	25.96
Varying-step algorithm	$m = 300,\ n = 100,\ l = 50,\ r = 20$	1.637×10^3	75.75

Algorithm	Problem size	Objective function value	CPU time (s)
CRNN algorithm	$m = 300,\ n = 100,\ l = 50,\ r = 20$	1.637×10^3	131.5
Proposed algorithm	$m = 300,\ n = 100,\ l = 50,\ r = 20$	1.637×10^3	25.96
Varying-step algorithm	$m = 300,\ n = 100,\ l = 50,\ r = 20$	1.637×10^3	75.75

Algorithm	Problem size	Objective function value	CPU time (s)
CRNN algorithm	$m = 200,\ n = 200,\ l = 50,\ r = 20$	727.76	65.81
Proposed algorithm	$m = 200,\ n = 200,\ l = 50,\ r = 20$	727.72	32.63
Varying-step algorithm	$m = 200,\ 2 = 100,\ l = 50,\ r = 20$	727.73	84.4

Algorithm	Problem size	Objective function value	CPU time(sec.)
CRNN algorithm	$m = 200,\ n = 40,\ l = 10,\ r = 6$	463.1	26.1
Proposed algorithm	$m = 200,\ n = 40,\ l = 10,\ r = 6$	463	9.9
Varying-step algorithm	$m = 200,\ n = 40,\ l = 10,\ r = 6$	463	28.9

network (CRNN) defined in (2). The computed results show that all three algorithm are all efficient in solving degenerate problems resulting from the nonunique solutions of the linear L_1 estimation problems. Furthermore, the computed results are listed in Table I and are plotted in Fig.1. From Table I and Fig. 1, we can see that the proposed neural network has a higher convergence speed than both the iterative algorithm with varying-step length and the CRNN.

IV. Application to Image Restoration Model

In this section, we apply the proposed neural network to an image model for real-time image restoration.

Consider the standard image restoration observation model

$$\mathbf{g} = H\mathbf{f} + \mathbf{n} \tag{17}$$

where $\mathbf{f} \in R^N$ and $\mathbf{g} \in R^N$ are the lexicographically ordered original and degraded images, H is the degradation matrix of size $N \times N$, and $\mathbf{n} \in R^N$ is the additive noise which may be non-Guassian. The purpose of image restoration is to remove those degradations so that the restored image becomes as close to the original as possible. Since H is usually ill-conditioned, the regularization method is often used to find an optimal solution of the following optimization problem:

$$\text{minimize } \lambda\|Q\mathbf{f}\|_2^2 + \|\mathbf{g} - H\mathbf{f}\|_2^2 \tag{18}$$

where Q is the regularization operator, $\lambda > 0$ is a regularization parameter, and $\|\cdot\|_2$ is the Euclidean norm. The regularization solution can be given by

$$\mathbf{f}_\lambda = \left(H^T H + \lambda Q^T Q\right)^{-1} H^T \mathbf{g}. \tag{19}$$

In order to handle non-Guassian noise, the following l_1 regularization optimization problem was studied

$$\text{minimize } \lambda\|Q\mathbf{f}\|_1 + \|\mathbf{g} - H\mathbf{f}\|_1. \tag{20}$$

In [17], Fu *et al.* proposed converting the L_1-norm optimization problem into a very large LP problem and then solved it by using a numerical optimization technique. Because their algorithm implementation is based on 1-D image model, it has to be restricted to very small images. Furthermore, both the L_2 regularization approach and L_1 regularization approach require the optimal regularization parameter for a perfect image estimate.

In order to relax the need of the optimal regularization parameter to be estimated, a GLAD estimation method for image restoration was developed in [20]. Let $\{\lambda_k\}_1^K$ be a group of regularization parameters which are not optimal, and let the corresponding regularization solution be $\mathbf{f}_{\lambda_k}$. A new image estimate is then given by

$$\mathbf{f}^* = \sum_{k=1}^{K} w_k^* \mathbf{f}_{\lambda_k} \tag{21}$$

where the weight vector $\mathbf{w}^* = [w_1^*, \ldots, w_K^*]^T$ is an optimal solution of one L_1 estimation problem

$$\text{minimize } \left\|\sum_{k=1}^{K} w_k^* \mathbf{f}_{\lambda_k} - \mathbf{f}\right\|_1$$

$$\text{s.t } \sum_{k=1}^{K} w_k = 1,\ 0 \le \mathbf{f} \le L\mathbf{e}$$

Fig. 2. Original image in Example 2.

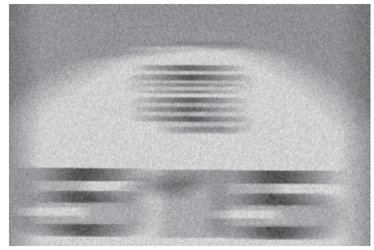

Fig. 3. Noisy blurred image in Example 2.

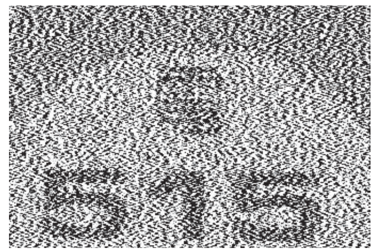

Fig. 4. Image restored by the regularization solution in Example 2.

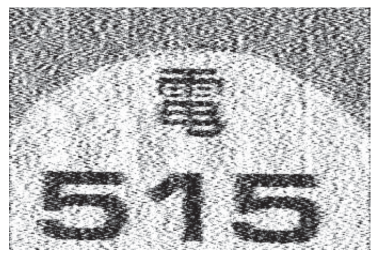

Fig. 5. Image restored by the proposed neural network in Example 2.

where $\|\cdot\|_1$ denotes l_1 norm. Furthermore, it can be reformulated into the following constrained least absolute deviation regression problem

$$\begin{aligned} &\text{minimize } f(\mathbf{w},\mathbf{z}) = \|A_\lambda \mathbf{w} - \mathbf{z} - \mathbf{b}\|_1 \\ &\text{s.t } \sum_{k=1}^{K} w_k = 1,\ \mathbf{z} \in X_0 \end{aligned} \tag{22}$$

where regression vector $(\mathbf{w},\mathbf{z})$ consists of parameter vector $\mathbf{w} \in R^K$ and noise error vector $\mathbf{z} \in R^N$, $A_\lambda = [\mathbf{f}_{\lambda_1},\ldots,\mathbf{f}_{\lambda_K}]$, $\mathbf{b} = 1/K\sum_{k=1}^{K}\mathbf{f}_{\lambda_k}$, $X_0 = \{\mathbf{z} \in R^N | \min(-\gamma_1\mathbf{b}, -\gamma_2\mathbf{b} + L\mathbf{e}) \le \mathbf{z} \le -\gamma_1\mathbf{b}\}$, $\mathbf{e} = [1,\ldots,1]^T \in R^N$, L is the upper bound of the gray value of image $\mathbf{f}$, and $\gamma_1 > 0$ and $\gamma_2 > 0$ are design constants. The GLAD estimation method for image restoration is to find an optimal solution $(\mathbf{w}^*,\mathbf{z}^*)$ of the constrained LAD regression problem (24) and the optimal estimate of image restoration is given by $\mathbf{f}^* = \sum_{k=1}^{K} w_k^*\mathbf{f}_{\lambda_k}$. It was shown that the CRNN based on the GLAD method can obtain the optimal image restoration estimate under the nonoptimal regularization parameter and non-Gaussian noise environments. Yet, because of the continuous-time feature of the CRNN and the large size of the image restoration problems, CRNN has very slow convergence rate.

For fast image restoration based on the GLAD method, we here use the proposed discrete-time neural network to solve (22). To do that, we first let $\alpha = \|[A_\lambda, -I]\|_2$, $\hat{A}_\lambda = A_\lambda/\alpha$, $\hat{b} = b/\alpha$, $X = \{\mathbf{x} \in R^n \mid \hat{\mathbf{l}} \le \mathbf{x} \le \hat{\mathbf{h}}\}$, $\hat{\mathbf{l}} = \min(-\mathbf{b}, -\mathbf{b} + L\mathbf{e})/\alpha$, $\hat{\mathbf{h}} = (-\mathbf{b})/\alpha$, and $I \in R^{N\times N}$ is a unit identity matrix. Then (22) can be equivalently rewritten as

$$\begin{aligned} &\text{minimize } f_1(\mathbf{w},\mathbf{z}) = \|\hat{A}_\lambda \mathbf{w} - \mathbf{z} - \hat{\mathbf{b}}\|_1 \\ &\text{s.t } \mathbf{e}_K^T\mathbf{w} = 1/K,\ \mathbf{z} \in X \end{aligned} \tag{23}$$

where $\mathbf{e}_K = [1,\ldots,1]/K \in R^K$. It is one special case of (1).

By the proposed neural network defined in (5), we have the following algorithm for image restoration.

Step 1: Compute the corresponding regularization solution given by

$$\mathbf{f}_{\lambda_k} = \left(H^T H + \lambda_k I\right)^{-1} H^T\mathbf{g},\ (k = 1,\ldots,K).$$

Step 2: Compute

$$\mathbf{w}(k+1) = \mathbf{w}(k) + h(\phi_1(k) - \hat{A}_\lambda^T\phi_3(k)) \tag{24a}$$

$$\mathbf{y}(k+1) = \mathbf{y}(k) + h(\hat{A}_\lambda\phi_1(k) + \phi_2(k) - \phi_3(k)) \tag{24b}$$

$$\mathbf{z}(k+1) = \mathbf{z}(k) + h(\phi_2(k) + \phi_3(k)) \tag{24c}$$

where $h > 0$ is a computational step length. $\phi_1 = (I_K - \mathbf{e} * \mathbf{e}_K^T/K)(\mathbf{w} - \hat{A}_\lambda^T\mathbf{y}) + \mathbf{e}_K/K - \mathbf{w}$, $\phi_2 = g_{X_1}(\mathbf{y} - \mathbf{z} + \hat{A}_\lambda\mathbf{w} - \hat{\mathbf{b}}) - \mathbf{y}$, $\phi_3 = g_{X_2}(\mathbf{y}+\mathbf{z}) - \mathbf{z}$, $X_1 = \{\mathbf{y} \in R^N |\ \max_i |y_i| \le 1\}$, $g_{X_1}(\mathbf{y}) = [g_{X_1}(y_1),\ldots,g_{X_1}(y_N)]^T$, $g_X(\mathbf{z}) = [g_X(z_1),\ldots,g_X(z_N)]^T$, and $g_X(z_i)$ and $g_{X_1}(z_i)$ are defined in (2) and (3), respectively.

Step 3: Calculate $\mathbf{f}(t) = \sum_{k=1}^{K} w_k(t)\mathbf{f}_{\lambda_k}$.

Step 4: Compute $Error = \|\mathbf{w}(k) - \mathbf{w}(k-1)\|_2 + \|\mathbf{y}(k) - \mathbf{y}(k-1)\|_2 + \|\mathbf{z}(k) - \mathbf{z}(k-1)\|_2$. If $Error < \epsilon$, output image estimate $\mathbf{f}^* = \mathbf{f}(k)$. Otherwise, let $k := k+1$, go to step 2.

As a direct corollary of Theorem 1, we have the following result.

Corollary 1: If $h < 0.5$, then the proposed neural network algorithm will converge globally to the optimal image estimate.

Remark 2: There exist the echo state network (ESN) [36] and the extreme learning machine (ELM) algorithm [37].We should point out the difference in their applications. The proposed discrete-time neural network can be applied to image restoration under nonoptimal regularization parameters, where a linearly constrained L_1 estimation problem defined in (23) is solved. In contrast, the ESN is used in various power system applications such as power system nonlinear load modeling and true harmonic current detection and intelligent control of an active power filter. The extreme learning machine is used in classification and and prediction. Also, it was reported that they may solve unconstrained regression problems fast.

Example 2: Consider a 227 by 339 pixel image acquired by a camera, as shown in Fig. 2. The original image is blurred

TABLE II
COMPARISON OF THE COMPUTATIONAL TIME OF THE THREE ALGORITHMS IN EXAMPLE 2

Algorithm	Problem size	Iterative number	CPU time (s)
CRNN algorithm	$N = 339$	7254	30.5
Varying-step algorithm	$N = 339$	15000	34.4
Proposed algorithm	$N = 339$	6000	6.8
Algorithm	Problem size	Iterative number	CPU time (s)
CRNN algorithm	$N = 1695$	7254	474.5
Varying-step algorithm	$N = 1695$	35000	326.8
Proposed algorithm	$N = 1695$	7000	35.2
Algorithm	Problem size	Iterative number	CPU time (s)
CRNN algorithm	$N = 3390$	9481	981.7
Varying-step algorithm	$N = 3390$	40000	515.4
Proposed algorithm	$N = 3390$	8500	86.5

Fig. 6. Original image in Example 3.

Fig. 7. Noisy blurred image in Example 3.

by 1×60 uniform motion blur and the contaminated Gaussian noise with distribution of the form

$$f = \frac{1}{2} N\left(0, \sigma^2\right) + \frac{1}{2} N\left(0, 4\sigma^2\right) \tag{25}$$

was added to the blurred image, where variance $\sigma^2 = 10$. Fig. 3 plots the noisy image. We consider the case in which the regularization operator is a unit operator. For another case in which the regularization operator is not a unit operator, we may have a similar discussion. We choose a group of nonoptimal regularization parameters in the following sequence:

$$\lambda_k = (0.15 + 0.0165 * k)/100, \quad (k = 1, \ldots, 10). \tag{26}$$

Using them, we compute the regularization solutions $\mathbf{f}_{\lambda_k}$ by (21). Fig. 4 plots the restored image by the regularization solution with λ_1, which is better than other λ_i. Clearly, it needs to be improved. We now use the proposed neural network with the computational step length $h = 0.75$ to further enhance the quality of the restored image. Note that $0 \leq s(t) \leq 255$. We take the parameters, defined in the set X, to be $L = 255$. The proposed neural network with zero initial point converges to the following optimal weight vector

$$\mathbf{w}^* = [0.0636, 0.1771, 0.0122, 0.0930, 0.0969, 0.0536, 0.1280, 0.2051, -0.0056, 0.1761]^T$$

and the resulting restored image is shown in Fig. 5, whose quality is much better than the one by the regularization algorithm. For a comparison, we employ the proposed algorithm, the varying-step algorithm, and the CRNN. The computed results on the computation time are listed in Table II. From Table II, it is clearly seen that the proposed algorithm has a much faster convergence speed than the varying-step algorithm and the CRNN.

Example 3: Consider a 256 by 256 pixels *Lena* image shown in Fig. 6. The original image is blurred by 5×5 uniform blur, and both Laplace noise and contaminated Gaussian noise of the form

$$f = 10\frac{1}{\sqrt{2\sigma^2}} e^{-\sqrt{2/\sigma^2}|\epsilon|} + \frac{1}{2} N(0, 10)$$

Fig. 8. Image restored by the average algorithm in Example 3.

Fig. 9. Image restored by the proposed neural network in Example 3.

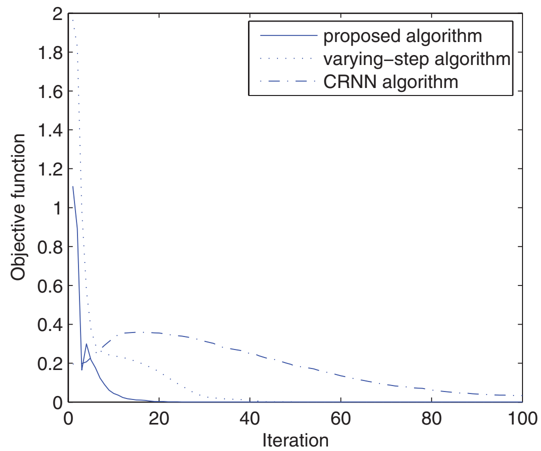

Fig. 10. Behavior of objective function as iterative number for Example 3.

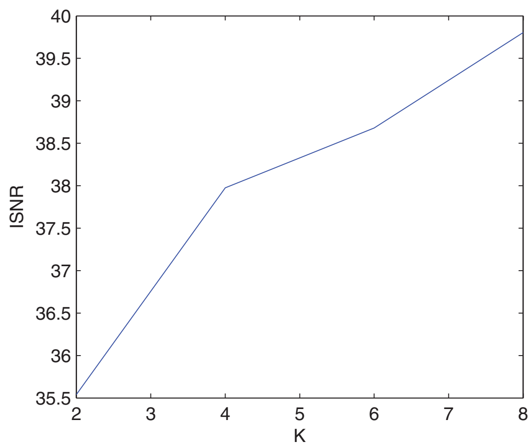

Fig. 11. ISNR curve as fusion number based on the proposed neural network for Example 3.

are added to the blurred image, where ϵ is a white noise uniformly distributed in the interval $[-0.5, 0.5]$ and $\sigma^2 = 0.5$. Fig. 7 plots the noisy image. We study the case in which the regularization operator is a unit operator. We choose a group of nonoptimal regularization parameters in the following sequence

$$\lambda_k = (1 + 0.1 * k) \times 10^{-8} \ (k = 1, \ldots, 4), \\ \lambda_k = 0.36 + k \ (k = 5, \ldots, 8). \quad (27)$$

Using them, we compute the regularization solutions $\mathbf{f}_{\lambda_k}$. Fig. 8 plots the restored image by the average fusion solution which is much better than the noisy image and has an ISNR $= 28.41$. Now we use the proposed neural network with the computational step length $h = 0.75$ to further enhance the quality of the restored image. Note that $0 \leq s(t) \leq 255$. We take the parameters, defined in the set X, be $L = 255$. The proposed neural network with zero initial point converges to the following optimal weight vector

$$\mathbf{w}^* = [-0.0004, -0.0004, 0.0199, 0.0411, 0.2337, \\ 0.2953, 0.2242, 0.2826]^T$$

and the resulting restored image is shown in Fig. 9 with an ISNR $= 39.5496$, whose quality is better than the one by the average method. Also, the varying-step algorithm defined in (6) and the CRNN defined in (2) obtain similar restoration results whose quality is much better than the one by the average method. Furthermore, we compare computation time based on the proposed algorithm, the varying-step algorithm defined in (6), and the CRNN defined in (2). Fig. 10 plots the change of the objective function value as the number of iterations. We can see that the proposed algorithm has a much faster convergence than the varying-step algorithm and the CRNN.

Finally, we plot ISNR curves as fusion number K, shown in Fig. 11. It shows that the larger the value of K, the better the image estimate.

Remark 3: The restoration result is somewhat sensitive to the K value, yet it is difficult to determine the optimal value of K for a good image estimate due to the unknown nonlinear relationship between K and the image estimate. In addition, the quality of the image estimate is also dependent on the sparsity in the weight vector.

V. Conclusion

This paper proposed a discrete-time neural network for calculating large linear L_1 estimation problems. The proposed neural network has the feature of a large computational step length and has been proven to be of global convergence. The proposed neural network was applied satisfactorily to an image model for real-time image restoration. The experimental results demonstrated that the proposed algorithm could obtain a better solution and had a much better convergence speed than the iterative algorithm with varying step length and the continuous-time neural network in solving the large linear L_1 estimation problems and image restoration problems.

Acknowledgment

Y. Xia would like to thank C. Tianping for providing an academic visiting position with the Department of Mathematics, Fudan University, Fudan, China.

References

[1] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1991.

[2] D. Erdogmus and J. C. Principe, "An error-entropy minimization algorithm for supervised training of nonlinear adaptive systems," *IEEE Trans. Signal Process.*, vol. 50, no. 7, pp. 1780–1786, Jul. 2002.

[3] Y. Dodge and J. Jana, *Adaptive Regression*. New York: Springer-Verlag, 2000.

[4] J. M. Chen and B. S. Chen, "System parameter estimation with input/output noisy data and missing measurements," *IEEE Trans. Signal Process.*, vol. 48, no. 6, pp. 1548–1558, Jun. 2000.

[5] S. S. Kuo and J. J. Mammone, "Image restoration by convex projections using adaptive constraints and the ℓ_1 norm," *IEEE Trans. Signal Process.*, vol. 40, no. 1, pp. 159–168, Jan. 1992.

[6] D. L. Donoho, "For most large underdetermined systems of equations, the minimal ℓ_1-norm near-solution approximates the sparsest near-solution," *Commun. Pure Appl. Math.*, vol. 59, no. 7, pp. 907–934, 2006.

[7] A. M. Ruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Rev.*, vol. 51, no. 1, pp. 34–81, 2009.

[8] R. H. Bartels, A. R. Conn, and J. W. Sinclair, "Minimization techniques for piecewise differentiable functions: The $\ell_1 1$ solution to an overdetermined linear system," *SIAM J. Numer. Anal.*, vol. 15, no. 2, pp. 224–241, Apr. 1978.

[9] M. R. Obsborne, *Finite Algorithm in Optimal and Data Analysis*. Chichester, U.K.: Wiley, 1985.

[10] M. R. Osborne, B. Presnell, and B. A. Turlach, "Knot selection for regression splines via the lasso," in *Dimension Reduction, Computational Complexity, and Information* (Computing Science and Statistics), vol. 30, S. Weisberg, Ed. Fairfax Station, VA: Interface Foundation North America, 1998, pp. 44–49.

[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] J. A. Cadzow, "Minimum ℓ_1, ℓ_2, and ℓ_∞ norm approximate solutions to an overdetermined system of linear equations," *Digital Signal Process.*, vol. 12, no. 4, pp. 524–560, Oct. 2002.

[13] W. Li and J. J. Swetits, "The linear ℓ_1 estimator and the Huber M-estimator," *SIAM J. Optim.*, vol. 8, no. 2, pp. 457–475, 1998.

[14] S. A. Ruzinsky and E. T. Olsen, "ℓ_1 and ℓ_∞ minimization via a variant of Karmarkar's algorithm," *IEEE Trans. Signal Process.*, vol. 37, no. 2, pp. 245–253, Feb. 1989.

[15] M. Grant, S. Boyd, and Y. Y. Ye, "Disciplined convex programming," in *Global Optimization: From Theory to Implementation*, L. Liberti and N. Maculan, Eds. Dordrecht, The Netherlands: Kluwer, 2005.

[16] M. Shi and M. A. Lukas, "An ℓ_1 estimation algorithm with degeneracy and linear constraints," *Comput. Stat. Data Anal.*, vol. 39, no. 1, pp. 35–55, Mar. 2002.

[17] H. Fu, M. K. Ng, M. Nikolova, and J. Barlow, "Efficient minimization methods of mixed ℓ_2-ℓ_1 and ℓ_1-ℓ_1 norms for image restoration," *SIAM J. Sci. Comput.*, vol. 27, no. 6, pp. 1881–1902, 2006.

[18] X. X. Guo, F. Li, and M. K. Ng, "A fast ℓ_1-TV algorithm for image restoration, *SIAM J. Sci. Comput.*, vol. 31, no. 10, pp. 2322–2341, 2009.

[19] A. Cichocki and R. Unbehauen, "Neural networks for solving systems of linear equations. II. Minimax and least absolute value problems," *IEEE Trans. Circuits Syst. II: Analog Digital Signal Process.*, vol. 39, no. 9, pp. 619–633, Sep. 1992.

[20] Y. S. Xia and M. S. Kamel, "Novel cooperative neural fusion algorithms for image restoration and image fusion," *IEEE Trans. Image Process.*, vol. 16, no. 2, pp. 367–381, Feb. 2007.

[21] Y. S. Xia and M. S. Kamel, "A cooperative recurrent neural network for solving ℓ_1 estimation problems with general linear constraints," *Neural Comput.*, vol. 20, no. 3, pp. 844–872, Mar. 2008.

[22] Y. S. Xia, "A compact cooperative recurrent neural network for computing general constrained ℓ_1 norm estimators," *IEEE Trans. Signal Process.*, vol. 57, no. 9, pp. 3693–3697, Sep. 2009.

[23] B. S. Bhattacharya and S. B. Furber, "Biologically inspired means for rank-order encoding images: A quantitative analysis," *IEEE Trans. Neural Netw.*, vol. 21, no. 7, pp. 1087–1099, Jul. 2010.

[24] Y. Chen, S.-K. Park, Y. Ma, and R. Ala, "A new automatic parameter setting method of a simplified PCNN for image segmentation," *IEEE Trans. Neural Netw.*, vol. 22, no. 6, pp. 880–892, Jun. 2011.

[25] A. M. Thompson, J. C. Brown, J. W. Kay, and D. M. Titterington, "A study of methods of choosing the smoothing parameter in image restoration by regularization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 4, pp. 326–339, Apr. 1991.

[26] N. P. Galatsanos and A. K. Katsaggelos, "Methods for choosing the regularization parameter and estimating the noise variance in image restoration and their relation," *IEEE Trans. Image Process.*, vol. 1, no. 3, pp. 322–336, Jul. 1992.

[27] S. J. Reeves and R. M. Mersereau, "Blur identification by the method of generalized cross-validation," *IEEE Trans. Image Process.*, vol. 1, no. 3, pp. 301–311, Jul. 1992.

[28] L. Guan, J. A. Anderson, and J. P. Sutton, "A network of networks processing model for image regularization," *IEEE Trans. Neural Netw.*, vol. 8, no. 1, pp. 169–174, Jan. 1997.

[29] A. K. Seghouane, "A note on image restoration using C_p and MSE," *IEEE Signal Process. Lett.*, vol. 15, pp. 61–64, 2008.

[30] A. K. Seghouane, "Model selection criteria for image restoration," *IEEE Trans. Neural Netw.*, vol. 20, no. 8, pp. 1357–1363, Aug. 2009.

[31] H. Kong and L. Guan, "A neural network adaptive filter for the removal of impulse noise in digital images," *Neural Netw.*, vol. 9, no. 3, pp. 373–378, Apr. 1996.

[32] M. C. Hong, T. Stathaki, and A. K. Katstaggelos, "Iterative regularized least-mean mixed-norm image restoration," *Opt. Eng.*, vol. 41, pp. 2515–2524, Oct. 2002.

[33] S. Kim, J. Shin, and J. Paik, "Real-time iterative framework of regularized image restoration and its application to video enhancement," *Real-Time Imaging*, vol. 9, no. 1, pp. 61–72, Feb. 2003.

[34] P. D. Mandic and J. A. Chambers, "Recurrent neural networks for prediction: Learning algorithms, architectures and stability," in *Adaptive and Learning Systems for Signal Processing, Communications and Control*. New York: Wiley, 2001.

[35] M. V. Solodov and P. Tseng, "Modified projection-type methods for monotone variational inequalities," *SIAM J. Control Optim.*, vol. 34, no. 5, pp. 1814–1830, 1996.

[36] H. Jaeger and H. Haas, "Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication," *Science*, vol. 304, no. 5667, pp. 78–80, Apr. 2004.

[37] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, nos. 1–3, pp. 489–501, Dec. 2006.

[38] A. Dax, "A row relaxation method for large ℓ_1 problems," *Linear Algebra Appl.*, vols. 154–156, pp. 793–818, Aug.–Oct. 1991.

Youshen Xia (M'96–SM'01) received the B.S. and M.S. degrees in computational mathematics and applied software from Nanjing University, Nanjing, China, in 1982 and 1989, respectively, and the Ph.D. degree in computational intelligence from the Chinese University of Hong Kong, Pokfulam, Hong Kong, in 2000.

He has held various research and visiting positions with Fudan University, Fudan, China, the University of Western Sydney, New South Wales, Australia, the University of Calgary, Canada, the University of Waterloo, Waterloo, ON, Canada, the City University of Hong Kong, and the Chinese University of Hong Kong. He is currently a Professor and the Chair of the Software Engineering Department, Fuzhou University, Fuzhou, China. His current research interests include the areas of speech signal and image processing, intelligence optimization algorithms, analysis, and their engineering applications.

Changyin Sun photograph and biography not available at the time of publication.

Wei Xing Zheng (M'93–SM'98) received the Ph.D. degree in electrical engineering from Southeast University, Nanjing, China, in 1989.

He has held various faculty, research, and visiting positions with Southeast University, Nanjing, China, the Imperial College of Science, Technology and Medicine, London, U.K., the University of Western Australia, Crawley, Australia, the Curtin University of Technology, Bentley Western, Australia, the Munich University of Technology, Munich, Germany, the University of Virginia, Charlottesville, Virginia, and the University of California-Davis, Davis. He is currently a Full Professor with the University of Western Sydney, New South Wales, Australia.

Prof. Zheng has served as an Associate Editor for five flagship journals, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS PART–I: FUNDAMENTAL THEORY AND APPLICATIONS from 2002 to 2004, the IEEE TRANSACTIONS ON AUTOMATIC CONTROL from 2004 to 2007, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS PART–II: EXPRESS BRIEFS from 2008 to 2009, the IEEE SIGNAL PROCESSING LETTERS from 2007 to 2010, and *Automatica* from 2011 to present. He was a Guest Editor of the special issue on "*Blind Signal Processing and Its Applications*" for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS PART—I: REGULAR PAPERS from 2009 to 2010. He has served as the Chair of the IEEE Circuits and Systems Society's Technical Committee on Neural Systems and Applications and the IEEE Circuits and Systems Society's Technical Committee on Blind Signal Processing.