Commun Nonlinear Sci Numer Simulat 16 (2011) 3746–3759

Contents lists available at ScienceDirect

Commun Nonlinear Sci Numer Simulat

journal homepage: www.elsevier.com/locate/cnsns

Edge detection of noisy images based on cellular neural networks

Huaqing Li *, Xiaofeng Liao, Chuandong Li, Hongyu Huang, Chaojie Li

State Key Lab. of Power Transmission Equipment: System Security and New Technology, College of Computer Science, Chongqing University, Chongqing 400044, PR China

ARTICLE INFO

Article history:
Received 5 August 2010
Received in revised form 13 December 2010
Accepted 14 December 2010
Available online 13 January 2011

Keywords:
Cellular neural network (CNN)
Templates
Image edge detection
Noise reduction

ABSTRACT

This paper studies a technique employing both cellular neural networks (CNNs) and linear matrix inequality (LMI) for edge detection of noisy images. Our main work focuses on training templates of noise reduction and edge detection CNNs. Based on the Lyapunov stability theorem, we derive a criterion for global asymptotical stability of a unique equilibrium of the noise reduction CNN. Then we design an approach to train edge detection templates, and this approach can detect the edge precisely and efficiently, i.e., by only one iteration. Finally, we illustrate performance of the proposed methodology from the aspect of peak signal to noise ratio (PSNR) through computer simulations. Moreover, some comparisons are also given to prove that our method outperforms classical operators in gray image edge detection.

1. Introduction

It is well known that the Hopfield neural network (HNN) requires fully connected and grows exponentially with the size of the array. Thus it is very difficult to implement, even in modest array sizes, as VSLI circuits [1,2]. A novel class of information processing system called cellular neural network (CNN) was proposed by Chua and Yang in 1988, which came from the HNN and cellular automata as an effective combination of both characteristics [3,4]. Moreover, the CNN has two prominent features: real-time signal processing capability and local connection. On one hand, the characteristic of real-time signal processing has been extensively exploited in various applications such as parallel signal processing, image edge detection, connected component detection and various morphology operations (dilation, erosion and hole filling, etc.). On the other hand, the characteristic of local connection makes it applicable to VLSI implementation and allows to operate at a very high speed in real time. With deep submicron technology (0.25 um-0.33 um), an array of 100×100 large analog processors array can be implemented on a single chip, whose theoretical computation speed can be at least a thousand times faster than the current digital processor [5]. Some smaller operational test chips have also been designed [6–8]. As a result of this rapid development, the CNNs have been widely studied for practical applications in image and video signal processing, robotic and biological visions and higher brain functions [9–12].

The most important key point of CNNs applications is how to find the satisfactory feedback template "A", control template "B" and bias "I". In recent years, the problem of CNN design for image processing has attracted considerable attention [13–20] and the promising potential of CNN has resulted in the development of several templates design methods. Among studies on the templates design in known literatures, the intuitive way will lead to quick results in several simple cases, but most of the time it does not guarantee to find the desired templates and the main disadvantage is doing lots of experiments. If the desired functions are exactly determined, then an approach for the direct templates design is applied. However, the most widely studied and used way is the templates learning method. These approaches such as LMI-based method [13], classic neural networks learning [11], the heuristic ways such as genetic algorithm [21], differential evolution algorithm [14],

* Corresponding author.
E-mail address: lhq_jsack@126.com (H. Li).

1007-5704/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.cnsns.2010.12.017

simulated annealing [23] and particle swarm optimization (PSO) [16] etc., are used to find the correlation between the input and the desired output and obtain the desired templates for specific applications.

Edge in an image is the connected boundary between two different regions. Edge detection is one of the most important and difficult steps in image processing and pattern recognition systems. Its importance arises from the fact that edge often gives an indication of the physical extent of an object within the image. Edge provides sufficient information about the image such that the size of the image data is reduced to the size that is more suitable for image analysis. The performance of the tasks after the edge detection, such as image segmentation, boundary detection, object recognition and classification, and image registration are dependent on the information on the edge. However, noise is a common problem in acquisition, transmission and processing of image, which will decrease image quality seriously [13–15,21,22]. Moreover, it will lead to unexpected results when we process the images with noise by using classical edge detection operators, such as Roberts, Sobel, Prewitt and LOG operators. So in image processing, if we do not reduce the noise influence on images beforehand, the obtained results will be far from our expectation. To the best of our knowledge, edge detection of image with noise based on CNNs has never been involved. Hence, it is our intention to tackle such an important and challenging subject.

In this paper, based on the heuristic level in Ref. [13], we investigated the edge detection of noisy images by employing CNNs and LMI. Our main work focuses on designing and training the templates of noise reduction and edge detection of CNNs. Based on the Lyapunov stability theorem, a criterion for the uniqueness and global asymptotical stabile equilibrium point for noise reduction CNN model is derived. Futhermore, an approach to find the edge detection templates is proposed, which can detect the edge by only one iteration. Both the noise reduction templates and edge detection templates are designed to obtain desirable output by using the property of saturation nonlinearity. It is shown that the problem of noise reduction and edge detection can be characterized in terms of LMIs. By solving a set of LMIs, the templates can be obtained given a certain pair of input and output. Finally, noise reduction templates and edge detection templates are trained by two pairs of input and output images with smaller size, respectively. Using the obtained templates, the edge of noisy image with larger size could be detected precisely. Moreover, some comparisons between the classical edge detection operators and the proposed edge detection methodology are given in detail. The experimental results illustrate that the performance of the proposed edge detection algorithm is better than that of the classical edge detection operators.

The remainder of this paper is organized as follows. In Section 2, we give and review the model of cellular neural networks in brief. Then, we propose a training method, using LMI, to produce the templates for the noise reduction and the edge detection in Sections 3 and 4, respectively. In Section 5, we show performance of the proposed methodology from the aspect of peak signal to noise ratio (PSNR) by means of computer simulations. Moreover, comparisons have also proven that our methodology outperforms the classic edge detection operators in edge detection. Finally, some conclusions are given in Section 6.

2. Model description of cellular neural network and preliminaries

In this section, the model of a two-dimensional CNN is briefly described which is composed of basic processing units called cells. Each cell is connected to its neighboring ones, therefore only the adjacent cells can interact directly with each other. For a CNN array with M rows and N columns on a 2D grid, the dynamics of each cell can be described by the following state equations [3,4]:

$$\begin{cases} \dot{x}_{ij}(t) = -x_{ij}(t) + \sum\limits_{C(k,l)\in N_r(i,j)} A(i,j;k,l)y_{kl}(t) + \sum\limits_{C(k,l)\in N_r(i,j)} B(i,j;k,l)u_{kl} + I_{ij}, \\ y_{ij}(t) = f(x_{ij}(t)) = \frac{1}{2}\left(|x_{ij}(t)+1| - |x_{ij}(t)-1|\right), \end{cases} \tag{1}$$

where $i = 1,2,\ldots,M$, $j = 1,2,\ldots,N$; $x_{ij}(t)$, u_{ij} and $y_{ij}(t)$ are the state, the input and the output of the (i,j)-th cell in the grid. The initial condition $x_{ij}(0) = 0$ and static input $|u_{ij}| \leqslant 1$. $A(i,j;k,l)$, $B(i,j;k,l)$ denote the connection templates from cell $C(k,l)$ to cell $C(i,j)$; I_{ij} represents the bias of (i,j)-th cell in the grid. From Eq. (1), it follows that the state and the output of each cell are affected by the inputs and outputs of its neighboring cells. In Eq. (1), for each cell $C(i,j)$ the following set $N_{i,j}(r)$, named r-neighborhood, can be defined as:

$$N_r(i,j) = \{C(k,l) := \max(|k-i|, |l-j|) \leqslant r, 1 \leqslant k \leqslant M, 1 \leqslant l \leqslant N\}, \tag{2}$$

where r which denotes the neighborhood radius of each cell is a positive integer, and the pairs (i,j) and (k,l) are the indices which express the position of cells, i.e., the rows and the columns of the generic cell and its neighboring ones in the grid, respectively. From a practical point of view, the cells which belong to the r-neighborhood of $C(i,j)$ are arranged in a maximum $(2r+1) \times (2r+1)$ grid whose central element coincides with $C(i,j)$. The working principle of Eq. (1) and the output function are depicted in Fig. 1(a) and (b), respectively.

A space-invariant standard CNN with a 3×3 neighborhood is defined uniformly by a string of 19 real numbers, called the CNN gene, i.e., they include a uniform threshold I, a feedback template A consists of nine feedback synaptic weights and a control cloning template B consists of nine control synaptic weights. The 19 real numbers together with initial condition and static input can completely determine the dynamical properties of the CNN (1). Suppose the templates A, B and the bias I are given by

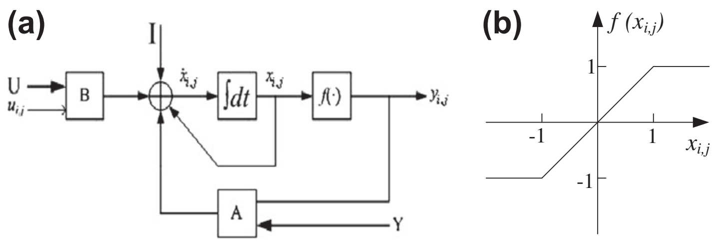

Fig. 1. System structure of Eq. (1) and its output function.

$$A = \begin{pmatrix} a_{-1,-1} & a_{-1,0} & a_{-1,1} \\ a_{0,-1} & a_{0,0} & a_{0,1} \\ a_{1,-1} & a_{1,0} & a_{1,1} \end{pmatrix}, \quad B = \begin{pmatrix} b_{-1,-1} & b_{-1,0} & b_{-1,1} \\ b_{0,-1} & b_{0,0} & b_{0,1} \\ b_{1,-1} & b_{1,0} & b_{1,1} \end{pmatrix}, \; I = I_{i,j}. \tag{3}$$

Let $n = M \times N$, through re-numbering the cells from 1 to n, Eq. (1) can be written in compact vector form by the following matrix equation:

$$\begin{cases} \dot{x}(t) = -x(t) + \bar{A}y(x(t)) + \bar{B}u + \bar{I}, \\ y(x(t)) = f(x(t)) = \frac{1}{2}(|x(t)+1| - |x(t)-1|), \end{cases} \tag{4}$$

or

$$\begin{cases} \dot{x}_i(t) = -x_i + \sum_{j=1}^{n} \bar{a}_{ij} y_j(x_j(t)) + \sum_{j=1}^{n} \bar{b}_{ij} u_j + \bar{I}_i, \\ y_i(x_i(t)) = f_i(x_i(t)) = \frac{1}{2}(|x_i(t)+1| - |x_i(t)-1|), \end{cases} \tag{5}$$

where $x(t) = [x_1(t), x_2(t), \ldots, x_n(t)]^{\mathrm{T}}$ is the state vector, $y(x(t)) = [y_1(x_1(t)), y_2(x_2(t)), \ldots, y_n(x_n(\cdot))]^{\mathrm{T}}$ is the output vector, $u = [u_1, u_2, \ldots, u_n]^{\mathrm{T}}$ is a static input vector which is independent of time, $f = [f_1(x_1), f_2(x_2), \ldots f_n(x_n)]^{\mathrm{T}}$ is the output function vector, $A = \{\bar{a}_{ij}\} \in R^{n\times n}$ and $B = \{\bar{b}_{ij}\} \in R^{n\times n}$ are feedback and control matrices respectively, describing the interaction between each cell and its neighboring ones in terms of their input and output variables. $\bar{I} = [I, I, \ldots, I]^{\mathrm{T}}$ is the vector containing the bias of each cell.

The feedback matrix $\bar{A}$ and control matrix $\bar{B}$ are different from the feedback template A and control template B, respectively, but their relation is quite close. The entries of the matrix $A = \{\bar{a}_{ij}\} \in R^{n\times n}$ and $B = \{\bar{b}_{ij}\} \in R^{n\times n}$ can be computed as [16]:

$$\bar{a}_{ij} = \begin{cases} a_{q-k,w-l}, & \text{if } |q-k| \leqslant 1 \quad \text{and} \quad |w-l| \leqslant 1, \\ 0, & \text{otherwise}, \end{cases} \tag{6}$$

and

$$\bar{b}_{ij} = \begin{cases} b_{q-k,w-l}, & \text{if } |q-k| \leqslant 1 \quad \text{and } |w-l| \leqslant 1, \\ 0, & \text{otherwise}, \end{cases} \tag{7}$$

where $i = 1,2,\ldots, M$, $j = 1,2,\ldots,N$; k, l, q and w are given by the following expressions:

$$k = \left\lceil \frac{i}{N} \right\rceil, \quad l = i - (k-1)N, \tag{8}$$

$$q = \left\lceil \frac{j}{N} \right\rceil, \quad w = j - (q-1)N, \tag{9}$$

where operator $\lceil\cdot\rceil$ denotes the ceiling function and the pairs (k,l) and (q,w) correspond to the indices providing the position of the interconnected cells in the r-neighborhood.

In literature [13], a LMI-based approach is proposed to remove the noise in noisy images. In this paper, we futher investigate edge detection of noisy images by combining CNNs and LMI technique. In order to achieve good results, we design a CNN to reduce the noise in the image waiting to be processed. After that, a CNN is designed to detect the edge of the image output by the noise reduction CNN. The flow chart is shown in Fig. 2. Both the templates of noise reduction CNN and edge detection CNN are trained by given sample with smaller size. At the same time, based on Lyapunov stability theorem and LMI approach, stability and convergence for the above CNN model are derived to guarantee the correctness and effectiveness of the training algorithm. In the next two sections, we will discuss how to design the noise reduction templates and edge detection templates, respectively.

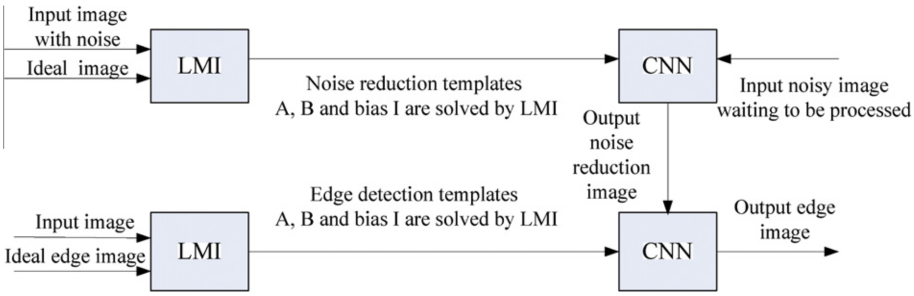

Fig. 2. Flow chart of edge detection of noisy image.

3. Noise reduction CNN design

In this section, we investigate an approach to design templates used to reduce the noise in the images. The templates is trained by a noisy image (be represented as the static input vector u) and desired output image (be represented as the static output vector y^*). The Schur Complement Lemma (please see the Appendix) will be used to derive a criterion for the uniquely and globally asymptotically stable equilibrium point for noise reduction CNN.

The nonlinear output function of a CNN always guarantees there exist at least one equilibrium point of system (4). For the purpose of simplifying the analysis, we shift the equilibrium $x^* = [x_1^*, x_2^*, \ldots, x_n^*]$ of (4) to the origin. Let $z(t) = x(t) - x^*$, the system (4) is transformed to

$$\begin{cases} \dot{z}(t) = -z(t) + \bar{A}\Phi(z(t)), \\ y(z(t) + x^*) = f(z(t) + x^*), \end{cases} \tag{10}$$

where $z(t) = [z_1(t), z_2(t), \ldots, z_n(t)]^{\mathrm{T}}$ is the new state vector, $\Phi(z(t)) = [\Phi_1(z_1(t)), \Phi_2(z_2(t)), \ldots, \Phi_n(z_n(t))]^{\mathrm{T}}$ represents the output vector of the transformed system and $\Phi_i(z_i(t)) = y_i(z_i(t) + x_i^*) - y_i(x_i^*), i = 1, 2, \ldots, n$. The function $\Phi_i(z_i(t))$ satisfies $|\Phi_i(z_i(t))| \leqslant |z_i(t)|$, which implies that

$$\Phi_i^2(z_i(t)) \leqslant z_i(t)\Phi_i(z_i(t)), \Phi_i(0) = 0, i = 1, 2, \ldots, n. \tag{11}$$

or equivalently

$$\Phi^{\mathrm{T}}(z(t))\Phi(z(t)) \leqslant \Phi^{\mathrm{T}}(z(t))z(t), \Phi(0) = 0. \tag{12}$$

Next we will first derive a criterion for the uniquely and globally asymptotically stable equilibrium point of the CNN based on the Lyapunov stability theorem and this criterion may be regarded as the extension of the results in [13].

Theorem 1. *If there exists a positive definite symmetric matrix $P = [p_{ij}] \in R^{n\times n}$ and a positive definite diagonal matrix $D = diag\{d_1, d_2, \ldots, d_n\} \in R^{n\times n}$, where $d_i > 0$, $i = 1,2,\ldots,n$, such that*

$$M = \begin{bmatrix} -2P & P\bar{A} \\ \bar{A}^{\mathrm{T}}P & D\bar{A} + \bar{A}^{\mathrm{T}}D \end{bmatrix} < 0, \tag{13}$$

then the origin of system (10) or equivalently the equilibrium point x^* of system (4) is uniquely and globally asymptotically stable for every $\bar{B}u + \bar{I}$.

The proof is appended in the Appendix.

So far, the criterion of the unique and global asymptotic stability of equilibrium point of CNN has been derived above. In other words, feedback template "A" of noise reduction CNN is obtained according to (13). Now, we will design the control template "B" and bias "I" of noise reduction CNN to achieve desirable output y^* at steady state. The equilibrium point equation of (5) takes the following form:

$$\begin{cases} -x_i^* + \sum_{j=1}^{n} \bar{a}_{ij} y_j(x_j^*) + \sum_{j=1}^{n} \bar{b}_{ij} u_j + I = 0, \\ y_i(x_i^*) = f_i(x_i^*) = \frac{1}{2}\left(|x_i^* + 1| - |x_i^* - 1|\right). \end{cases} \tag{14}$$

By using the property of saturation nonlinearity, Eq. (14) can be rewritten as

$$\begin{cases} \sum_{j=1}^{n} \bar{a}_{ij} + \sum_{j=1}^{n} \bar{b}_{ij} u_j + I > 1, & \text{if } x_i^* > 1 (y_i^* = 1) \\ \sum_{j=1}^{n} \bar{a}_{ij} y_j^* + \sum_{j=1}^{n} \bar{b}_{ij} u_j + I = y_i^*, & \text{if } |x_i^*| < 1 (|y_i^*| < 1) \\ -\sum_{j=1}^{n} \bar{a}_{ij} + \sum_{j=1}^{n} \bar{b}_{ij} u_j + I < -1, & \text{if } x_i^* < -1 (y_i^* = -1) \end{cases}$$

$$i = 1, 2, \ldots, n = MN \tag{15}$$

Remark 1. The image is coded such that -1 corresponds to black pixels and + 1 to white ones.

Combination of (13) and (15) forms the main result in this section and it will be employed to design the templates of noise reduction CNN. Using the obtained templates, noise in the image can be reduced to a large extent which lays a good foundation for later edge detection.

4. CNN design for edge detection

In this section, we discuss a method to train the templates which will be used to detect the noise reduced image. Firstly, we give a uncoupled CNN which can detect the edge precisely by only one iteration. So this will greatly reduce the time of algorithm implementation. We choose the following form templates:

$$A = \begin{pmatrix} 0 & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} b_{-1,-1} & b_{-1,0} & b_{-1,1} \\ b_{0,-1} & b_{0,0} & b_{0,1} \\ b_{1,-1} & b_{1,0} & b_{1,1} \end{pmatrix}, \; I = I_{i,j}. \tag{16}$$

Then the dynamics of the uncoupled CNN becomes

$$\begin{cases} \dot{x}_{ij}(t) = -x_{ij}(t) + a y_{ij}(t) + \sum_{C(k,l)\in N_r(i,j)} B(i,j;k,l) u_{kl} + I, \\ y_{ij}(t) = f(x_{ij}(t)) = \frac{1}{2}\left(|x_{ij}(t) + 1| - |x_{ij}(t) - 1|\right), \end{cases} \tag{17}$$

The first equation of (17) could be discretized into following difference equation by setting time step $h = 1$.

$$]x_{i,j}(n+1) = a y_{i,j}(n) + \sum_{C(k,l)\in N_r(i,j)} B(i,j;k,l) u_{kl} + I, \tag{18}$$

Now we will show that if $a > 1$ and the initial condition $x_{i,j}(0) = 0$, then the final state will be $|x_{i,j}(\infty)| > 1$, which implies that the CNN will decay to stabilization at last.

According to the correlation between input and nonlinear output, we obtain that $y_{i,j}(0) = 0$. In the light of (18), we know that

$$x_{i,j}(1) = \sum_{C(k,l)\in N_r(i,j)} B(i,j;k,l) u_{kl} + I. \tag{19}$$

From (19), we know that in the convergence process, the second state value of each cell in CNN is only dependent of $B(i,j;k,l)$, I and u_{kl}. So we give the following inferences.

(1) To ensure the normal operation of CNN, we need $x_{i,j}(1) \neq 0$. Suppose $x_{i,j}(1) = 0$, then $y_{i,j}(1) = 0$. According to (18), we have $x_{i,j}(2) = 0$, which implies $x_{i,j}(\infty) = 0$. So $x_{i,j}(1) = 0$ will lead that network does not work.

(2) If $x_{i,j}(1) \geqslant 1$, $y_{i,j}(1) = 1$. On the basis of (19), we have

$$x_{i,j}(2) = a + \sum_{C(k,l)\in N_r(i,j)} B(i,j;k,l) u_{kl} + I = a + x_{i,j}(1) > 1, \tag{20}$$

which implies $x_{i,j}(\infty) > 1$.

(3) If $0 < x_{i,j}(1) < 1$, $y_{i,j}(1) = x_{i,j}(1)$. Under (19), we obtain that

$$x_{i,j}(2) = a x_{i,j}(1) + \sum_{C(k,l)\in N_r(i,j)} B(i,j;k,l) u_{kl} + I = (a+1) x_{i,j}(1), \tag{21}$$

which implies

$$x_{i,j}(n) = (a^{n-1} + a^{n-2} + \cdots + a + 1)x_{i,j}(1). \tag{22}$$

Eventually, $x_{i,j}(\infty) \geqslant 1$.

(4) If $-1 < x_{i,j}(1) < 0$, $y_{i,j}(1) = x_{i,j}(1)$. According to (19), we known that

$$x_{i,j}(2) = ax_{i,j}(1) + \sum_{C(k,l)\in N_r(i,j)} B(i,j;k,l)u_{kl} + I_{i,j} = (a+1)x_{i,j}(1), \tag{23}$$

which implies that

$$x_{i,j}(n) = (a^{n-1} + a^{n-2} + \cdots + a + 1)x_{i,j}(1). \tag{24}$$

Ultimately, $x_{i,j}(\infty) \leqslant -1$.

(5) If $x_{i,j}(1) \leqslant -1$, $y_{i,j}(1) = -1$. Under (19), we get that

$$x_{i,j}(2) = -a + \sum_{C(k,l)\in N_r(i,j)} B(i,j;k,l)u_{kl} + I_{i,j} = -a + x_{i,j}(1) < -1 \tag{25}$$

which implies $x_{i,j}(\infty) < -1$.

From the above analysis, we can determine the final output of each cell in the edge detection CNN by determining the state's first iteration value $x_{i,j}(1)$. It can be specifically expressed as follows:

if $x_{i,j}(1) > 0$ ($x_{i,j}$ corresponds to edge point), then $y_{i,j}(\infty) = 1$;
if $x_{i,j}(1) < 0$ ($x_{i,j}$ corresponds to non-edge point), then $y_{i,j}(\infty) = -1$.

Given the initial condition $x_{i,j}(0) = 0$ and parameter constraint $a > 1$, the CNN will be stable, i.e., $|x_{i,j}(\infty)| > 1$. In the following, the approach to obtain the edge detection will be investigated. The edge detection templates are trained by a image which is represented as the static input vector u and the corresponding desired image which is represented as the output vector y^*. Similar to Eq. (15), we have the following results.

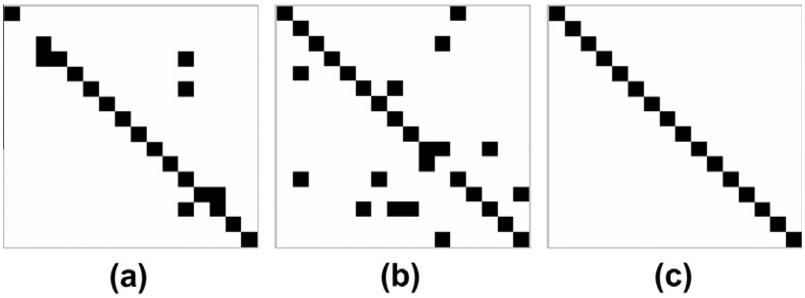

Fig. 3. The training images. (a) Corrupted image with 5% noise. (b) Corrupted image with 10% noise. (c) Desired image.

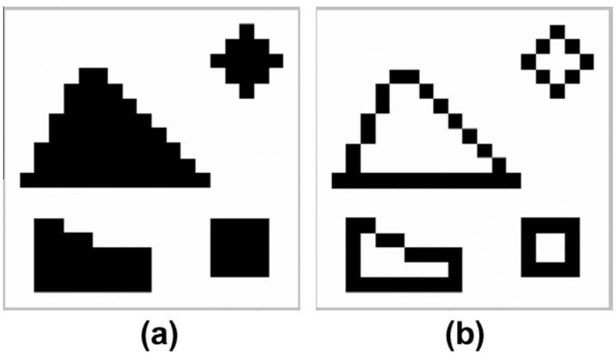

Fig. 4. The training images. (a) Input image. (b) Ideal edge image.

Theorem 2. *If a > 1 and the following LMIs are solvable*

$$\begin{cases} a+\sum\limits_{j=1}^{n}\bar{b}_{ij}u_j+I>1, & \text{if } x_i^*>1(y_i^*=1) \\ ay_i^*+\sum\limits_{j=1}^{n}\bar{b}_{ij}u_j+I=y_i^*, & \text{if } |x_i^*|<1(|y_i^*|<1) \\ -a+\sum\limits_{j=1}^{n}\bar{b}_{ij}u_j+I<-1, & \text{if } x_i^*<-1(y_i^*=-1) \end{cases}$$
$$i=1,2,\cdots,n=MN. \tag{26}$$

Then, the edge detection templates can be obtained.

The proof is similar to the derivation of (15), thus is omitted here. The templates of the CNN are solved such that its output converges to the output of the ideal edge detector which, by definition, can completely extract the edges from the image. The ideal edge detector that is needed for template learning and this is represented by the ideal edge image.

Remark 2. In image edge detection, we always require the feedback parameter $a > 1$, which determines the convergence rate of the CNN. There are two advantages by choosing $a > 1$. The one is a plays a positive feedback role, which makes the value of cell state corresponding to edge point (non-edge point) in an image increase (decrease), and output 1 (-1) finally. The other is one can determine the final output only through judging the second state value. Therefore, this method greatly reduces the time complexity of program.

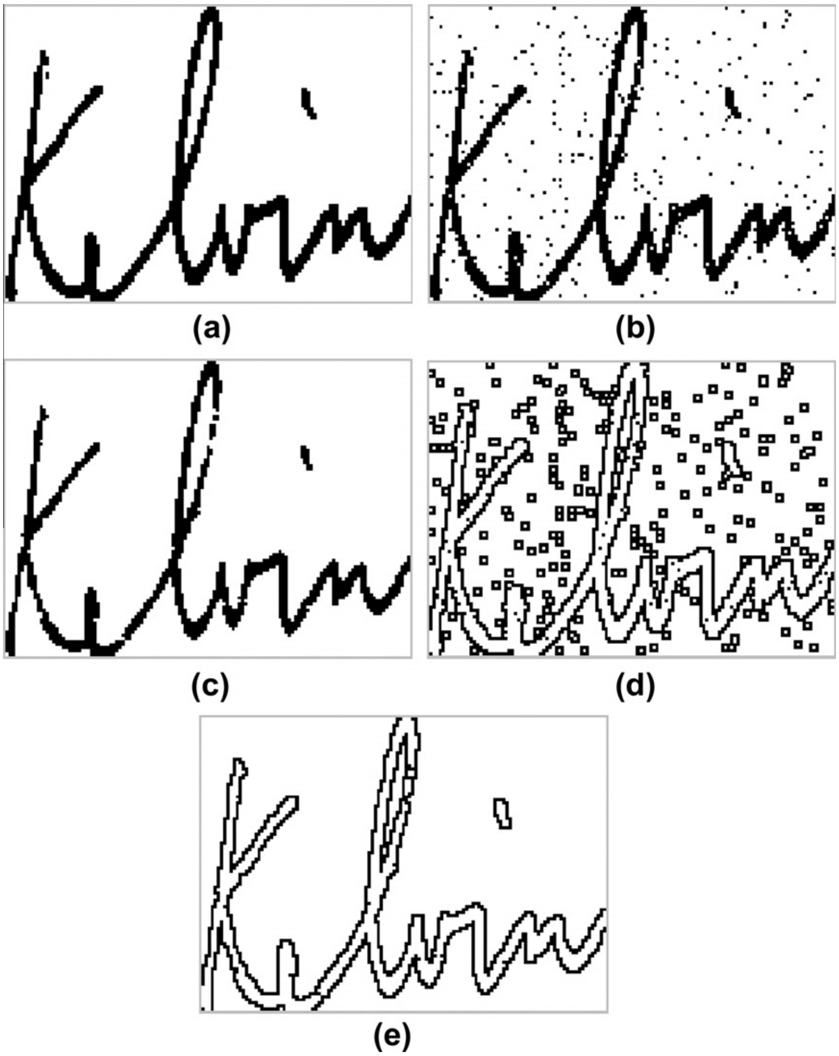

Fig. 5. Edge detection of noisy image, where the noise ratio is 5%. (a) Original image. (b) The noisy image. (c) Image after CNN processing with templates in (27). (d) Edge detection results of noisy image. (e) Edge detection results of noise reduction image.

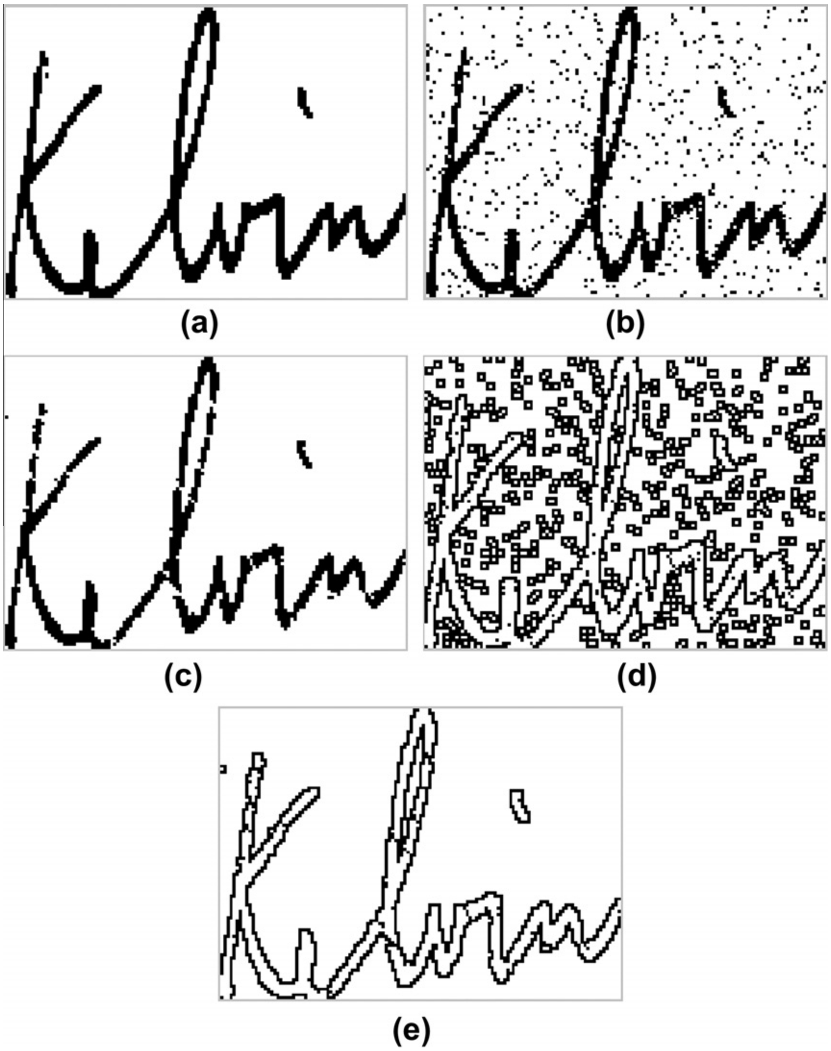

Fig. 6. Edge detection of noisy image, where the noise ratio is 10%. (a) Original image. (b) The noisy image. (c) Image after CNN processing with templates in (28). (d) Edge detection results of noisy image. (e) Edge detection results of noise reduction image.

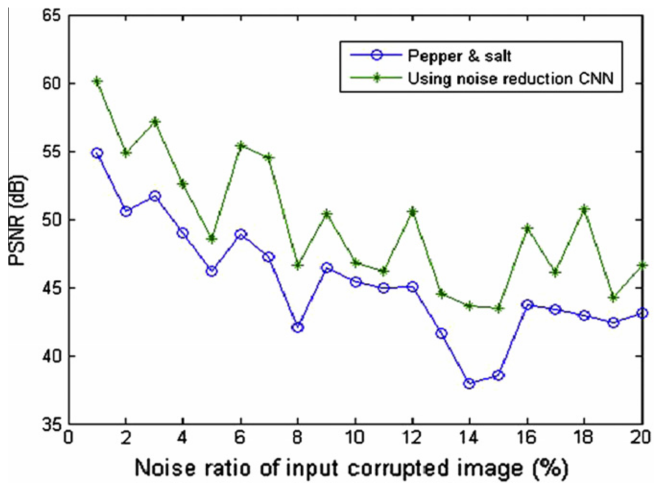

Fig. 7. The PSNR of pepper & salt and noise reduction CNN under different ratios.

Remark 3. This training method is not only suit able for the binary image, but also suit able for the gray level image.

5. Simulations

In this section, an example is presented to illustrate the effectiveness of the proposed method. Herein we will use the different binary training samples (with smaller size 16 by 16) shown in Fig. 3 to train the templates of noise reduction CNN. The image in Fig. 3 (a) is corrupted by the pepper and salt noise with noise radio 5% and the image in Fig. 3 (b) is corrupted by the pepper and salt noise with noise radio 10%. Fig. 3 (c) is the desired image. Fig. 4 shows the training images for edge detection CNN. Fig. 4. (a) shows the original training image, which is a 20 by 20 sized synthetic image generated in personal computer. Fig. 4. (b) is the ideal edge detected image which is obtained using intensity differences in the training image. As discussed before, templates of the proposed CNNs are solved using LMI.

Using the training samples in Fig. 3 (a) and (b), by solving the LMIs in (15), the following templates used to reduce the noise in images corrupted with 5% noise are obtained.

$$A = \begin{pmatrix} 291.6553 & -118.9228 & -106.4406 \\ -253.9071 & -251.7907 & -203.3817 \\ -197.7633 & -195.9248 & 273.8371 \end{pmatrix}, \quad B = \begin{pmatrix} 135.9598 & 139.5516 & 145.6166 \\ 229.4109 & 379.7530 & 212.4300 \\ 160.3459 & 182.4772 & 153.2460 \end{pmatrix},$$

$$I = 223.3609. \tag{27}$$

Similarly, using the training samples in Fig. 3 (a) and (c), by solving the LMIs in (15), the following templates used to reduce the noise in images corrupted with 10% noise are obtained.

$$A = \begin{pmatrix} 81.2522 & -138.6389 & -55.0413 \\ -64.4410 & -150.7432 & -238.3880 \\ -229.1651 & -76.0652 & 60.3057 \end{pmatrix}, \quad B = \begin{pmatrix} 178.2492 & 142.5289 & 0.3584 \\ -0.0220 & 294.5108 & 85.8291 \\ 135.7233 & 95.3915 & 126.5357 \end{pmatrix},$$

$$I = 300.9400. \tag{28}$$

Using the training samples in Fig. 4 (a) and (b), the following templates used for edge detection are solvedd by LMI.

$$A = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 3.9899 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} -0.9804 & -0.9804 & -0.9804 \\ -0.9804 & 8.0025 & -0.9804 \\ -0.9804 & -0.9804 & -0.9804 \end{pmatrix}, I = -0.5387. \tag{29}$$

By combining the noise reduction templates and edge detection templates, the results images series are as follows. (See Figs. 5 and 6)

In order to assess the performance of noise reduction CNN under different levels of noise ratio, we introduce the peak signal to noise ratio (PSNR) as:

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(\tilde{y}_{ij} - \hat{y}_{ij})^2, \tag{30}$$

$$PSNR = 10\log_{10}\frac{255^2}{MSE}dB, \tag{31}$$

where $\tilde{y}_{ij}$ is the pixel of the original ideal image, $\hat{y}_{ij}$ is the pixel of the reconstruction image at the output of noise reduction CNN. The PSNR of pepper & salt and noise reduction CNN under different noise ratios is shown in Fig. 7 and Table 1. From Fig. 7 and Table 1, we know that the noise in the image was greatly reduced quantitatively, which shows the effectiveness of proposed method.

Table 1
PSNR (dB) at different levels of noise ratio of input corrupted image.

	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%	11%
Pepper & salt	55.36	50.39	52.16	48.49	46.16	48.56	47.86	43.65	52.42	46.23	44.52
Noise reduction CNN	61.23	55.02	58.26	52.96	48.63	56.63	56.96	47.86	47.52	47.96	46.35

	12%	13%	14%	15%	16%	17%	18%	19%	20%
Pepper & salt	45.63	38.32	38.56	39.95	43.56	45.36	44.86	43.26	44.72
Noise reduction CNN	53.26	45.62	45.36	43.72	52.49	47.85	51.23	44.25	47.21

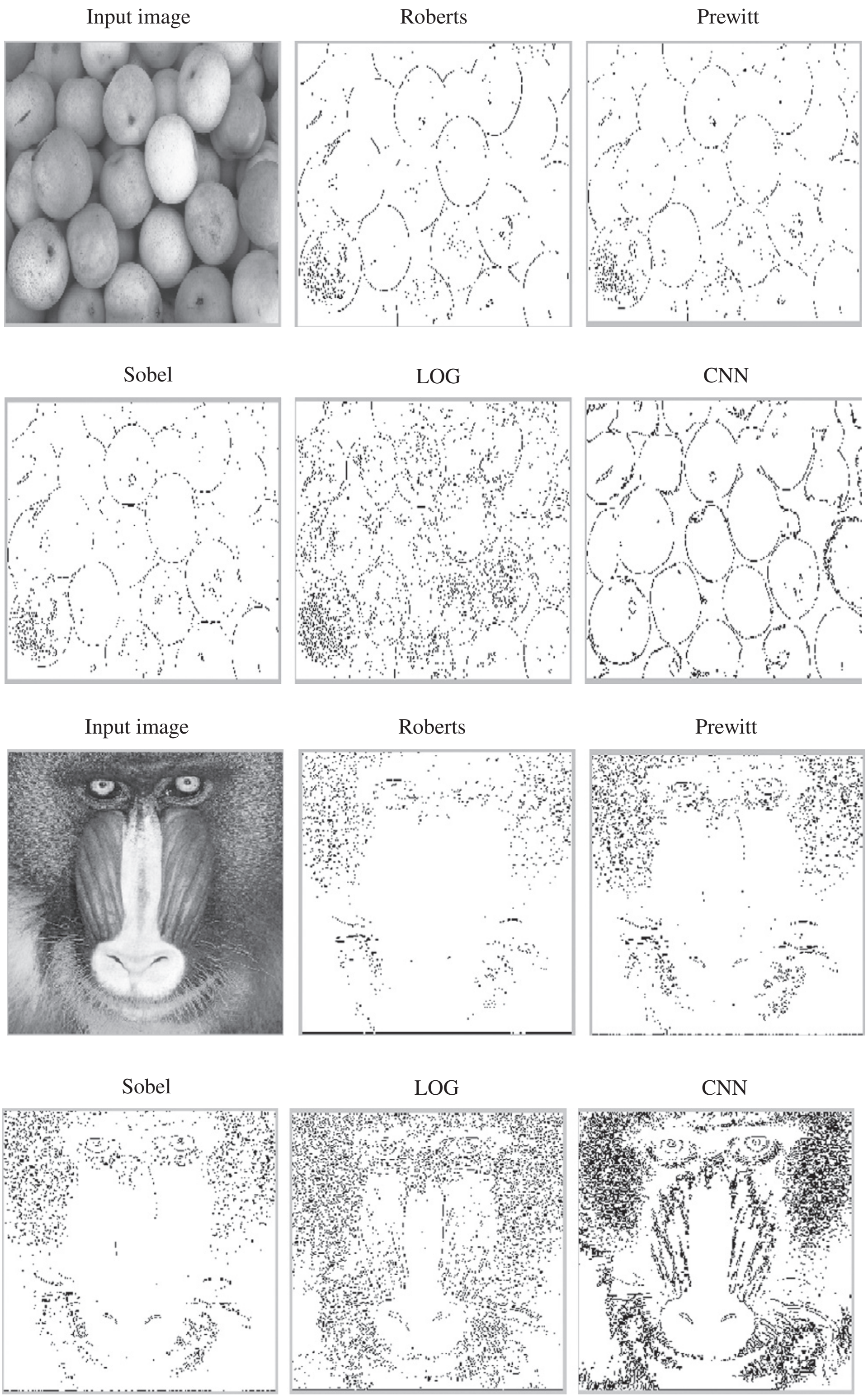

Fig. 8. Comparison of edge detection performance using Roberts, Prewitt, Sobel, LOG and CNN.

For evaluating performance of proposed CNN edge detector, detailed edge detection experiments are performed with a binary image and four gray level images. To compare the performance of proposed CNN edge operator, other four popular competitive edge detectors (Roberts, Sobel, Prewitt and LOG) are used as references. The simulation results are

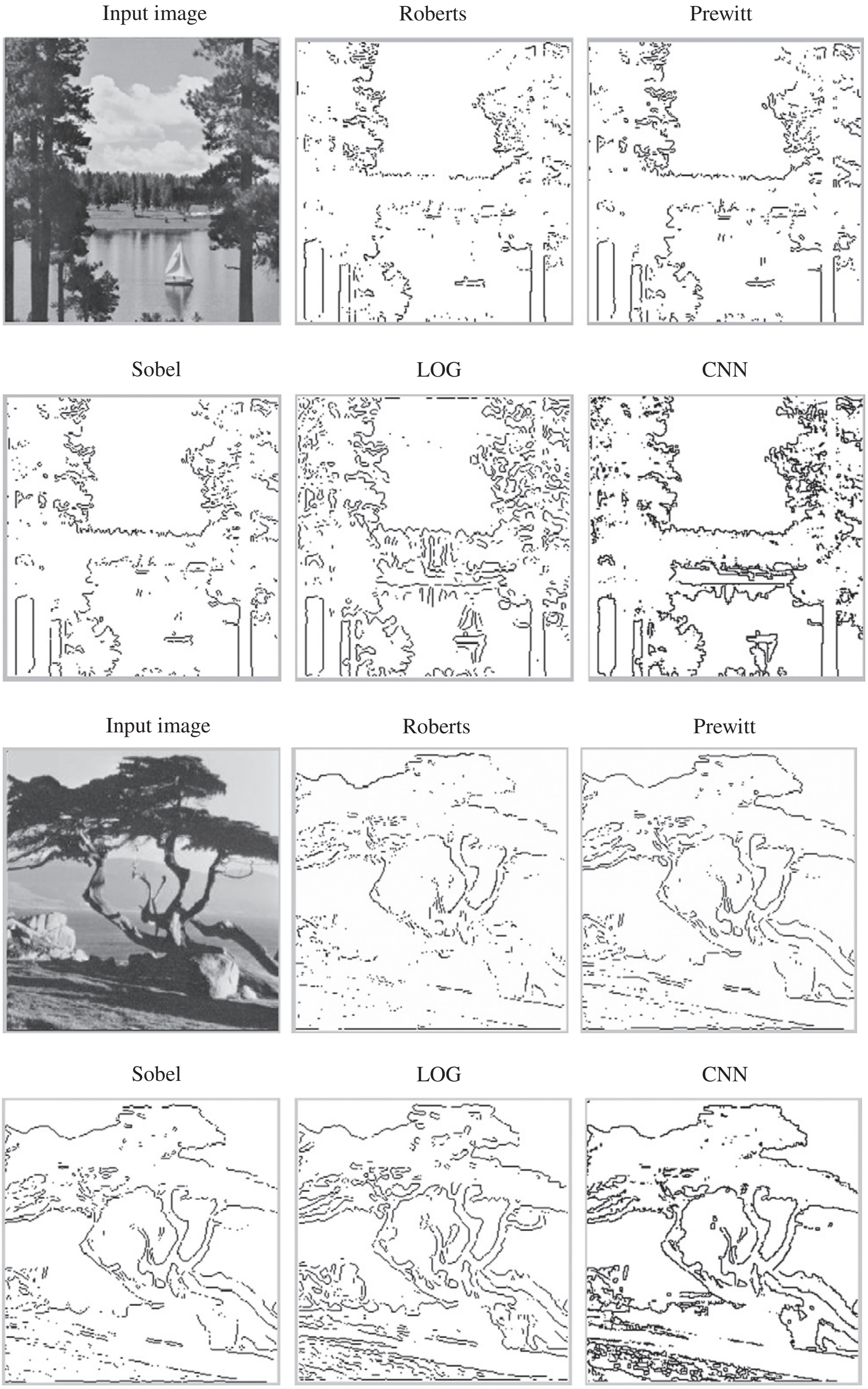

Fig. 8 (*continued*)

shown in Fig. 8. It can be seen from Fig. 8 that the edge detection results using CNN are more precise than above classic operators.

MATLAB code is avaliable in matlab1.com

6. Conclusion

Edge detection is one of the most important and difficult steps in image processing and pattern recognition systems. Its importance arises from the fact that edge often gives an indication of the physical extent of an object within the image. Edge provides sufficient information about the image such that the size of the image data is reduced to the size that is more suitable for image analysis. The performance of the tasks after the edge detection, such as image segmentation, boundary detection, object recognition and classification, and image registration are dependent on the information on the edge. However, noise is a common problem in acquisition, transmission and processing of image, which will decrease image quality seriously. Moreover, it will lead to unexpected results when we process the images with noise by using classical edge detection operators, such as Roberts, Sobel, Prewitt and LOG operators.

In this paper, a methodology employing both CNN and LMI for edge detection of noisy images is investigated. The work is achieved in two steps. In the first step, a CNN used for reducing the noise in image waiting to be processed is designed. Then, a CNN used for detecting the edge is designed in the second step. It is shown that the templates design problem in the two steps can be transformed into LMIs, then it is straightforward to obtain the solution by recently developed LMI Toolbox. The satisfactory results can be achieved by choosing the best templates among a few experiments. From the simulation results, the CNN shows excellent performance in detecting the details in images. Therefore, it is particularly suitable for high-definition image processing. In the future, the templates design should be considered by combining the CNN and intelligent optimization algorithms such as differential evolution algorithm (DEA), genetic algorithm (GA), particle swarm optimization (PSO) and simulated annealing algorithm (SAA), ect.

Acknowledgment

The work described in this paper is funded by the Project No. CDJXS11182239 Supported by the the Fundamental Research Funds for the Central Universities, the National Natural Science Foundation of China (Nos. 60973114 and 61003247), the Natural Science Foundation project of CQCSTC (2009BA2024), and the State Key Laboratory of Power Transmission Equipment & System Security and New Technology, Chongqing University (2007DA10512709207).

Appendix A

Lemma (see [23]). *Given constant matrices X_1, X_2 and X_3 of appropriate dimensions, where $X_1^T = X_1, X_3^T = X_3$ and $0 < X_2^T = X_2$, then $X_1 + X_3^T X_2^{-1} X_3 < 0$ if and only if*

$$\begin{bmatrix} X_1 & X_3^T \\ X_3 & -X_2 \end{bmatrix} < 0, \quad \text{or} \begin{bmatrix} -X_2 & X_3 \\ X_3^T & X_1 \end{bmatrix} < 0,$$

where the notation $P > 0$ implies that P is a positive definite matrix.

Proof. The proof of this theorem is divided in two steps: in the first step the uniqueness of the equilibrium point is discussed and in the second step global asymptotic stability of the equilibrium point is proven.

Step 1: We will now prove the uniqueness of the equilibrium point by the contradiction method. The equilibrium point z^* of system (10) satisfies the following equation

$$z^* - \overline{A}\Phi(z^*) = 0 \tag{A1}$$

Eq. (A1) implies that if $\Phi(z^*) = 0$, then $z^* = 0$. In the following, we assume that $\Phi(z^*) \neq 0$. From (12), we have that

$$2\Phi^{\mathrm{T}}(z^*)z^* \geqslant 2\Phi^{\mathrm{T}}(z^*)\Phi(z^*) \geqslant 0. \tag{A2}$$

If there exists a positive definite diagonal matrix $D = diag\{d_1, d_2, \ldots, d_n\} \in R^{n\times n}$, where $d_i > 0$, $i = 1,2,\ldots, n$, then we have that

$$2\Phi^{\mathrm{T}}(z^*)Dz^* \geqslant 0. \tag{A3}$$

Eq. (A3) can be equivalently rewrite as:

$$\Phi^{\mathrm{T}}(z^*)Dz^* + (z^*)^{\mathrm{T}}D\Phi(z^*) \geqslant 0. \tag{A4}$$

By (A1), we have that

$$\Phi^{\mathrm{T}}(z^*)D\overline{A}\Phi(z^*) + \Phi^{\mathrm{T}}(z^*)\overline{A}^{\mathrm{T}}D\Phi(z^*) \geqslant 0. \tag{A5}$$

Adding and subtracting $2(z^*)^{\mathrm{T}}Pz^*$ on the left hand of (A5) yields

$$-2(z^*)^{\mathrm{T}}Pz^* + 2(z^*)^{\mathrm{T}}Pz^* + \Phi^{\mathrm{T}}(z^*)D\bar{A}\Phi(z^*) + \Phi^{\mathrm{T}}(z^*)\bar{A}^{\mathrm{T}}D\Phi(z^*) \geqslant 0. \tag{A6}$$

Substituting (A1) into (A6), we obtain that

$$-2(z^*)^{\mathrm{T}}Pz^* + 2(z^*)^{\mathrm{T}}P\bar{A}\Phi(z^*) + \Phi^{\mathrm{T}}(z^*)D\bar{A}\Phi(z^*) + \Phi^{\mathrm{T}}(z^*)\bar{A}^{\mathrm{T}}D\Phi(z^*) \geqslant 0. \tag{A7}$$

Eq. (A7) can be expressed as the following inequality:

$$\left[(z^*)^{\mathrm{T}}, \quad \Phi^{\mathrm{T}}(z^*)\right]\begin{bmatrix} -2P & P\bar{A} \\ \bar{A}^{\mathrm{T}}P & D\bar{A}+\bar{A}^{\mathrm{T}}D \end{bmatrix}\begin{bmatrix} z^* \\ \Phi(z^*) \end{bmatrix} \geqslant 0. \tag{A8}$$

On the other hand, if $\Phi(z^*) \neq 0$, then by inequality (13) in Theorem 1, it implies that

$$\left[(z^*)^{\mathrm{T}}, \quad \Phi^{\mathrm{T}}(z^*)\right]\begin{bmatrix} -2P & P\bar{A} \\ \bar{A}^{\mathrm{T}}P & D\bar{A}+\bar{A}^{\mathrm{T}}D \end{bmatrix}\begin{bmatrix} z^* \\ \Phi(z^*) \end{bmatrix} < 0. \tag{A9}$$

Clearly, (A9) contradicts with (A8), which implies that $\Phi(z^*) = 0$ and $z^* = 0$. Thus, we have proven that the origin of Eq. (10) is the unique solution.

Step 2: It will now be shown that the conditions in Theorem 1 also implies the origin of Eq. (10) in Section 3 is globally asymptotically stable. To this end, we will use the following positive definite Lyapunov functional as candidate:

$$V(z(t)) = z^{\mathrm{T}}(t)Pz(t) + 2\sum_{i=1}^{n} d_i \int_0^{z_i(t)} \Phi_i(s)ds, \tag{A10}$$

where $P > 0$, $D = diag\{d_1, d_2, \ldots, d_n\} > 0$, $d_i > 0$, $i = 1, 2, \ldots, n$. The time derivative of $V(z(t))$ along the trajectories of the system (10) in Section 3 is

$$\begin{aligned}
\dot{V}(z(t)) &= 2z^{\mathrm{T}}(t)P\dot{z}(t) + 2\Phi^{\mathrm{T}}(z(t))D\dot{z}(t) \\
&= 2z^{\mathrm{T}}(t)P\left[-z(t) + \bar{A}\Phi(z(t))\right] + 2\Phi^{\mathrm{T}}(z(t))D\left[-z(t) + \bar{A}\Phi(z(t))\right] \\
&= -2z^{\mathrm{T}}(t)Pz(t) + 2z^{\mathrm{T}}(t)P\bar{A}\Phi(z(t)) - 2\Phi^{\mathrm{T}}(z(t))Dz(t) + 2\Phi^{\mathrm{T}}(z(t))D\bar{A}\Phi(z(t)) \\
&= -2z^{\mathrm{T}}(t)Pz(t) + 2z^{\mathrm{T}}(t)P\bar{A}\Phi(z(t)) - 2\Phi^{\mathrm{T}}(z(t))Dz(t) \\
&\quad + 2\Phi^{\mathrm{T}}(z(t))D\bar{A}\Phi(z(t)) - 2\Phi^{\mathrm{T}}(z(t))D\Phi(z(t)) + 2\Phi^{\mathrm{T}}(z(t))D\Phi(z(t)) \\
&= -2z^{\mathrm{T}}(t)Pz(t) + 2z^{\mathrm{T}}(t)P\bar{A}\Phi(z(t)) + 2\Phi^{\mathrm{T}}(z(t))D\bar{A}\Phi(z(t)) \\
&\quad - 2\Phi^{\mathrm{T}}(z(t))D\Phi(z(t)) - 2\Phi^{\mathrm{T}}(z(t))Dz(t) + 2\Phi^{\mathrm{T}}(z(t))D\Phi(z(t)) \\
&= -2z^{\mathrm{T}}(t)Pz(t) + 2z^{\mathrm{T}}(t)P\bar{A}\Phi(z(t)) + 2\Phi^{\mathrm{T}}(z(t))D\Phi(z(t)) \\
&\quad - 2\Phi^{\mathrm{T}}(z(t))D\Phi(z(t)) - 2\Phi^{\mathrm{T}}(z(t))Dz(t) + 2\Phi^{\mathrm{T}}(z(t))D\Phi(z(t)) \\
&= -2z^{\mathrm{T}}(t)Pz(t) + 2z^{\mathrm{T}}(t)P\bar{A}\Phi(z(t)) + 2\Phi^{\mathrm{T}}(z(t))D\Phi(z(t)) \\
&\quad - 2\Phi^{\mathrm{T}}(z(t))D\Phi(z(t)) + 2\Phi^{\mathrm{T}}(z(t))D(\Phi(z(t)) - z(t)) \\
&\leqslant \left[z^{\mathrm{T}}(t), \Phi^{\mathrm{T}}(z(t))\right]\begin{bmatrix} -2P & P\bar{A} \\ \bar{A}^{\mathrm{T}}P & D\bar{A}+\bar{A}^{\mathrm{T}}D \end{bmatrix}\begin{bmatrix} z(t) \\ \Phi(z(t)) \end{bmatrix}
\end{aligned} \tag{A11}$$

The condition (13) in Theorem 1 ensures that $\dot{V}(z(t)) < 0$ for all $\Phi(z(t)) \neq 0$. Now let $\Phi(z(t)) = 0$ and $z(t) \neq 0$. In this case, we have

$$\dot{V}(z(t)) = -2z^{\mathrm{T}}(t)Pz(t). \tag{A12}$$

Obviously, $\dot{V}(z(t)) < 0$ for all $z(t) \neq 0$. Thus, we have proven that $\dot{V}(z(t)) \leqslant 0$. Moreover, $\dot{V}(z(t)) = 0$ if and only if $z(t) = \Phi(z(t)) = 0$. On the other hand, $V(z(t))$ is radially unbounded since $V(z(t)) \to \infty$ as $\|z(t)\| \to \infty$. Hence, under the conditions of Theorem 1, the origin of (10) or equivalently the equilibrium x^* of (4) in the Section 3 is globally asymptotically stable for every $\bar{B}u + \bar{I}$. □

References

[1] Chua LO. CNN: a version of complexity. Int J Bifurcat Chaos 1997;7(10):2219–425.
[2] Min LQ, Zhang XJ. Robust Designs of a kind of uncoupled CNNs with Nonlinear Templates. In: Proceedings of international conference on communications, circuits and systems;2008. p. 978–981.
[3] Chua LO, Yang L. Cellular neural networks: theory. IEEE Trans Circ Syst 1988;35(10):1257–72.
[4] Chua LO, Yang L. Cellular neural networks: applications. IEEE Trans Circ Syst 1988;35(10):1273–90.
[5] Zarandy A. The art of CNN template design. Int J Circ Theory Appl 1999;27(1):5–23.
[6] Harrer H, Nossek JA, Roska T, Chua LO. A current-mode DTCNN universal chip. In: Proceedings of IEEE international symposium on circuits and systems;1994. p. 135–138.
[7] Cruz JM, Chua LO, Roska T. A fast, complex and efficient test implementation of the CNN universal machine. In: Proceedings of 3th IEEE international workshop on cellular neural networks and their application;Rome, December 1994. p. 61–66.

[8] Espejo S, Dominguez-Castro R, Rodriguez-Vázquez A, Carmona R. CNNUC2 User's guide, Centro Nacional de Microelectrónica, Seville;1995.
[9] Roska T, Chua LO. The CNN universal machine: an analogic array computer. IEEE Trans Circ Syst II 1993;40(3):163–73.
[10] Anguita M, Pelayo FJ, Fernandez FJ, Prieto A. A low-power CMOS implementation of programmable CNN's with embedded photosensors. IEEE Trans Circ Syst I: Fundam Appl 1997;44(2):149–53.
[11] Liu D. Cloning template design of cellular neural networks for associative memories. IEEE Trans Circ Syst I: Fundam Theory Appl 1997;44(7):646–50.
[12] Fajfar I, Bratkovic F, Tuma T, Puhan J. A rigorous design method for binary cellular neural networks. Int J Circ Theory Appl 1998;26(4):365–73.
[13] Su TJ, Wei CP, Huang SC, Hou CL. Image noise cancellation using linear matrix inequality and cellular neural networks. Opt Commun 2008;281(23):5706–12.
[14] Bastürk A, Günay E. Efficient edge detection in digital images using a cellular neural networks optimized by differential evolution algorithm. Exp Syst Appl 2009;36(2):2645–50. part 2.
[15] Lin YJ, Hou CL, Su TJ. Cellular neural networks for noise cancellation of gray image based on hybrid linear matrix inequality and particle swarm optimization. In: Proceedings of the 2009 international conference on new trends in information and service science;2009. p. 613–617.
[16] Fornarelli G, Giaquinto A. Adaptive particle swarm optimization for CNN associative memories design. Neurocomputing 2009;72(16-18):3851–62.
[17] He QB, Chen FY. Designing cnn genes for binary images edge smoothing and noise removing. Int J Bifurcat Chaos 2006;16(10):3007–13.
[18] Chen FJ, Chen FY, He GL. Image processing via CNN genes with von neumann neighborhoods. Int J Bifurcat Chaos 2005;15(12):3999–4006.
[19] Hänggi M, Moschytz GS. An exact and direct analytical method for the design of optimally robust CNN templates. IEEE Trans Circ Syst I, Fundam Theory Appl 1999;46(2):304–11.
[20] Zamparelli M. Genetically trained cellular neural networks. Neural Netw 1997;10(6):1143–51.
[21] Chua LO, Roska T. Cellular neural networks and visual computing: foundations and applications. Cambridge, United Kingdom: University Press; 2002.
[22] De Souza SX, Suykens ME, Johan JAK, Vandewalle J. Automatic chip-specific CNN template optimization using adaptive simulated annealing. In: Proceedings of the 2003 European conference circuit theory and design;2003. p. 329–332.
[23] Liao XF, Wong KW, Wu ZF, Chen GR. Novel robust stability criteria for interval delayed Hopfield neural networks. IEEE Trans Circ Syst-I 2001;48(11):1355–8.