Image Denoising Using Stochastic Chaotic Simulated Annealing

Lipo Wang, Leipo Yan, and Kim-Hui Yap

School of Electrical and Electronic Engineering, Nanyang Technological University Block S1, 50 Nanyang Avenue, Singapore 639798

Abstract. In this Chapter, we present a new approach to image denoising based on a novel optimization algorithm called stochastic chaotic simulated annealing. The original Bayesian framework of image denoising is reformulated into a constrained optimization problem using continuous relaxation labeling. To solve this optimization problem, we then use a noisy chaotic neural network (NCNN), which adds noise and chaos into the Hopfield neural network (HNN) to facilitate efficient searching and to avoid local minima. Experimental results show that this approach can offer good quality solutions to image denoising.

Keywords: image denoising, neural networks, chaos, noise

1 Introduction

Image denoising is to estimate the original image from a noisy image with some assumptions or knowledge of the image degradation process. There exist many approaches for image denoising [1] [2] [3] [4]. Here we adopt a Bayesian framework because it is highly parallel and it can decompound a complex computation into a network of simple local computations [3], which is important in hardware implementation of neural networks. This approach computes the maximum *a posteriori* (MAP) estimation of the original image given a noisy image. The MAP estimation requires the prior distribution of the original image and the conditional distribution of the data. The prior distribution of the original images imposes contextual constraints and can be modeled by Markov random field (MRF) or, equivalently, by Gibbs distribution. The MAP-MRF principle centers on applying MAP estimation on MRF modeling of images.

Li incorporated augmented Lagrange multipliers into the Hopfield neural network (HNN) for solving optimization problems [5]. He transformed a combinatorial optimization problem into real constrained optimization using the notion of continuous relaxation labeling. The HNN was then used to solve the real constrained optimization.

The neural network approaches have been shown to be a powerful tool for solving the optimization problems [6] [7]. The HNN is a typical model of neural network with symmetric connection weights. It is capable of solving quadratic optimization problems. However, it suffers from convergence to local minima [8]. To overcome the weakness, different simulated annealing techniques have been combined with the HNN to solve optimization problems [8] [9] [10] [11] [12]. Kajiura et al [11] proposed the gaussian machine which combines stochastic simulated annealing (SSA) with neural network for solving assignment problems. Convergence to globally optimal solutions is guaranteed if the cooling schedule is sufficiently slow, i.e., no faster than logarithmic progress [3]. SSA searches the entire solution space, which is time consuming. Chen and Aihara [9] proposed a transiently chaotic neural network (TCNN) which adds a large negative self-coupling with slow damping in the Euler approximation of the continuous HNN so that neurodynamics eventually converge from strange attractors to an equilibrium point. Chaotic simulated annealing (CSA) can search efficiently because of its reduced search spaces. The TCNN showed good performance in solving traveling salesman problem. However CSA is deterministic and is not guaranteed to settle into a global minimum. In view of this, Wang and Tian [12] proposed a novel algorithm called stochastic chaotic simulated annealing (SCSA) which combines both stochastic manner of SSA and chaotic manner of CSA. In this paper the NCNN, which performs SCSA algorithm, is applied to solve the constrained optimization in the MAP-MRF formulated image denoising. Experimental results show that the NCNN outperforms the HNN and the TCNN.

The rest of the chapter is organized as follows: Section 2 introduces the MAP-MRF framework in image restoration and the transformation of the combinatorial optimization to a real unconstrained optimization. Section 3 presents the NCNN and the derivation of the neural network dynamics. The experimental results are shown in Section 4. Section 5 concludes the paper.

2 MAP-MRF Image Restoration

Let the original image, the restored image and the degraded image be denoted by $x = \{x_i \mid i \in S\}$, $\hat{x} = \{\hat{x}_i \mid i \in S\}$ and $y = \{y_i \mid i \in S\}$ respectively, where $S = \{1, \ldots, N\}$ indexes the set of sites corresponding to the image pixels and N is the number of the image pixels.

When the original image is degraded by identical independently distributed (i.i.d.) Gaussian noise, the degraded image is modeled by

$$y_i = x_i + e_i \tag{1}$$

where $e_i \sim N(0, \sigma^2)$ is the zero mean Gaussian distribution with standard deviation σ . The objective of image denoising is to find an \hat{x} that approximates x.

Each pixel takes on a discrete value in the label set $\mathcal{L} = 1, \ldots, M$. The spatial relationship of the sites is determined by a neighborhood system $\mathcal{N} = \{\mathcal{N}_i \mid i \in \mathcal{S}\}$ where \mathcal{N}_i is the set of sites neighboring *i*. A single site or a set of neighboring sites form a clique denoted by *c*. \mathcal{C} is the set of all cliques.

There are many different ways the pixels can influence each other through contextual interactions. Such contextual interactions can be modeled as MRFs. According to Markov-Gibbs equivalence [13], an MRF is equivalent to a Gibbs distribution

$$P(x) = Z^{-1} \times e^{-\frac{1}{T} \sum_{c \in \mathcal{C}} V_c(x)}$$

$$\tag{2}$$

where $V_c(x)$ is the clique potential function, T is a temperature constant, and Z is a normalizing constant.

In the MAP-MRF labeling, the posterior distribution can be computed by using [14]

$$P(x|y) = \frac{p(y|x)P(x)}{p(y)}$$
(3)

p(y) is a constant when y is given. P(x) is the prior probability and p(y|x) is the conditional distribution. The prior distribution of the original image which imposes the contextual constraints can be expressed in terms of the MRF clique potentials

$$P(x) = Z^{-1} \times e^{-\frac{1}{T} \sum_{c \in C} U(x)}$$

$$\tag{4}$$

where

$$U(x) = \sum_{\{i \in \mathcal{C}\}} V_1(x) + \sum_{\{i,i'\} \in \mathcal{C}} V_2(x_i, x_{i'}) + \cdots$$
(5)

In this chapter only pair-site cliques are considered.

In (1) the noise is independent Gaussian noise. The conditional distribution can be expressed in terms of y, x and σ .

$$p(y|x) = \frac{1}{\prod_{i}^{N} \sqrt{2\pi\sigma^2}} e^{-U(y|x)}$$
(6)

where

$$U(y|x) = \sum_{i \in \mathcal{S}} \frac{(x_i - y_i)^2}{2\sigma^2} \tag{7}$$

Knowing the prior distribution and the conditional distribution, the energy in the posterior distribution is given by [4]

$$E(x) = \sum_{i \in S} \frac{(x_i - y_i)^2}{2\sigma^2} + \sum_{\{i, i'\} \in \mathcal{C}} V_2(x_i, x_{i'})$$
(8)

in which $\sum_{\{i,i'\}\in \mathcal{C}} V_2(x_i, x_{i'})$ is the pair-site clique potential of the MRF model. Maximizing a *posteriori* is equivalent to find an \hat{x} such that $E(\hat{x})$ is the global minimum.

In order to minimize E(x) in (8), proper MRF model has to be chosen so

that appropriate contextual constraints can be posed. Among various MRFs, the multi-level logistic (MLL) model is a simple mechanism for encoding a large class of spatial patterns [14]. In MLL, the pair-site clique potentials take the form

$$V_2(x_i, x_{i'}) = \begin{cases} -\beta_c \text{ if } x_i = x_{i'} \\ \beta_c \text{ otherwise} \end{cases}$$
(9)

where β_c is the parameter for cliques of type $c = \{i, i'\}$. However we found that the pair-site potentials are strong constraints. Instead of using (9) we use

$$V_2(x_i, x_{i'}) = -\beta_c * g(x_i, x_{i'})$$
(10)

$$g(x_i, x_{i'}) = 2e^{-|x_i - x_{i'}|/\tau} - 1$$
(11)

where τ is a positive constant.



Fig. 1. Graphical representation of $g(x_i, x_{i'})$

 $g(x_i, x_{i'})$ in the modified potential function is an exponential function in (-1,1]. Compared to the potential function in the MLL model, the modified potential function allows the pixels to be slightly different from the neighboring pixels. This is logical as most real images have smooth non-uniform regions.

Since image pixels can only take discrete values, the minimization in (8) is combinatorial. The combinatorial optimization can be converted into a constrained optimization in a real space using the continuous relaxation labeling. Let $p_i(I) \in [0, 1]$ represent the strength with which label I is assigned to i, the energy with the p variables is given by

$$E(p) = \sum_{i \in \mathcal{S}} \sum_{I \in \mathcal{L}} p_i(I) +$$

$$\sum_{i \in \mathcal{S}} \sum_{I \in \mathcal{L}} \sum_{i' \in \mathcal{S}, i' \neq i} \sum_{I' \in \mathcal{L}} r_{i,i'}(I, I') p_i(I) p_{i'}(I')$$
(12)

418

where $I = x_i$ and $I' = x_{i'}$, $r_i(I) = V_1(I|y) = (I - y_i)^2/2\sigma^2$ is the singlesite clique potential function and $r_{i,i'}(I, I') = V_2(I, I'|y) = V_2(I, I')$ is the pair-site clique potential function in the posterior distribution P(x|y).

With such a representation, the combinatorial minimization is reformulated as the following constrained minimization

$$\min_{p} E(p) \tag{13}$$

subject to
$$C_i(p) = 0$$
 $i \in \mathcal{S}$ (14)

$$p_i(I) \ge 0 \qquad \forall i \in \mathcal{S}, \ \forall I \in \mathcal{L}$$
 (15)

where $C_i(p) = \sum_I p_i(I) - 1 = 0.$

The augmented Lagrange Hopfield (ALH) method can be used to solve the above constrained minimization problem [4]. It uses the augmented Lagrange technique to satisfy the equality constraints of (14) and the Hopfield encoding to impose the inequality constraints of (15). The augmented Lagrange function takes the form

$$L_{\beta}(p,\gamma) = E(p) + \sum_{k \in \mathcal{S}} \gamma_k C_k(p) + \frac{\beta}{2} \sum_{k \in \mathcal{S}} [C_k(p)]^2$$
(16)

where γ_k are the Lagrange multipliers and $\beta > 0$ is the weight for the penalty term. The final solution p^* is subject to additional constraints: $p_i^*(I) \in \{0, 1\}$.

The ALH method uses the HNN to optimize the energy function. However the HNN is prone to trappings at local minima. In view of this, we propose a new network NCNN to perform the optimization.

3 Noisy Chaotic Neural Network

Let $u_i(I)$ denote the internal state of the neuron (i, I) and $p_i(I)$ denote the output of the neuron (i, I). $p_i(I) \in [0, 1]$ represents the strength that the pixel at location *i* takes the value *I*. The NCNN is formulated as follows [12]:

$$p_i^{(t)}(I) = \frac{1}{1 + e^{-u_i^{(t)}(I)/\epsilon}}$$
(17)

$$u_{i}^{(t+1)}(I) = ku_{i}^{(t)}(I) - z^{(t)}(p_{i}^{(t)}(I) - I_{o}) + n^{(t)} + \alpha(\sum_{i'=1,i'\neq i}^{N} \sum_{I'=1}^{M} T_{iI;i'I'}p_{i'}^{(t)}(I') + \mathcal{I}_{i}(I))$$
(10)

(18)

$$z^{(t+1)} = (1 - \beta_z) z^{(t)} \tag{19}$$

$$A[n^{(t+1)}] = (1 - \beta_n) A[n^{(t)}]$$
(20)

where

 $T_{iI;i'I'}$: connection weight from neuron (i', I') to neuron (i, I);

 $\mathcal{I}_i(I)$: input bias of neuron (i, I);

k: damping factor of nerve membrane ($0 \le k \le 1$)

 α : positive scaling parameter for inputs;

- ϵ : steepness parameter of the output function ($\epsilon \geq 0$);
- z : self-feedback connection weight or refractory strength $(z \ge 0)$;
- I_o : positive parameter;

n : random noise injected into the neurons;

 β_z : positive parameter ($0 < \beta_z < 1$);

 β_n : positive parameter ($0 < \beta_n < 1$);

A[n]: the noise amplitude.

When $n^{(t)}$ in (18) equals to zero, the NCNN becomes TCNN. When $z^{(t)}$ equals to zero, the TCNN becomes similar to the HNN with stable fixed point dynamics. The basic difference between the HNN and the TCNN is that a nonlinear term $z^{(t)}(p_i^{(t)}(I) - I_o)$ is added to the HNN. Since the "temperature" $z^{(t)}$ tends toward zero with time evolution, the updating equations of the TCNN eventually reduce to those of the HNN. In (18) the variable $z^{(t)}$ can be interpreted as the strength of negative self-feedback connection of each neuron, the damping of $z^{(t)}$ produces successive bifurcations so that the neurodynamics eventually converge from strange attractors to a stable equilibrium point [8].

CSA is deterministic and is not guaranteed to settle into a global minimum. In view of this, Wang and Tian [12] added a noise term $n^{(t)}$ in (18). The noise term continues to search for the optimal solution after the chaos of the TCNN disappears.

From (13)-(15) and (18), we obtain the dynamics of the NCNN:

$$u_i^{(t+1)}(I) = k u_i^{(t)}(I) - z^{(t)}(p_i^{(t)}(I) - I_o) + n^{(t)} + \alpha q_i^{(t)}(I)$$
(21)

where

$$q_i(I) = -\frac{\partial L_{\beta}(p,\gamma)}{\partial p_i(I)}$$

Note that the Lagrange multipliers are updated with neural outputs according to $\gamma_k^{(t+1)} = \gamma_k^{(t)} + \beta C_i(p^{(t)}).$

4 Experimental Results

Both artificial image and real image have been used to demonstrate the performance of the NCNN on image denoising.

The artificial image is a circle image of size 256×256 with M = 4 gray levels. Its label set is $\mathcal{L} = \{0, 1, 2, 3\}$. Three noisy circle images were generated by adding zero-mean i.i.d. Gaussian noise with standard deviation $\sigma = 0.5$, $\sigma = 0.75$ and $\sigma = 1$ respectively.

420

The noisy images were set to be the input of the neural networks. After the neural networks were initialized, each neuron was updated using (21) and (17). After all neurons in the neural networks were updated once, γ_k , z and n are updated. The updating scheme is cyclic and asynchronous. When all the neurons are updated once, we call it one iteration. Once the state of a neuron is updated, the new state information is immediately available to other neurons in the network (asynchronous).

The parameters that we used for the NCNN are: k = 1, $\epsilon = 0.01$, $\alpha = 0.005$, $I_0 = 0.65$, $z^{(0)} = 0.05$, $n^{(0)} = 0.01$. β is increased from 1 to 50 according to $\beta \leftarrow 1.01\beta$. The decreasing rate of z and n, β_z and β_n are 0.005. For the TCNN and the HNN we use the same parameters as the NCNN except that n = 0 for the TCNN, n = 0 and z = 0 for the HNN. The MRF pair-site clique potential parameter $\beta_c = 1$.

Table 1 shows the required iteration numbers and the peak signal-to-noise ratio (PSNR) of the denoised images. The higher the PSNR, the better the image quality. The denoised images are shown in Fig. 2-4.

Noise Level		Iterations	PSNR_r	PSNR_d
$\sigma = 0.5$	HNN	23	35.8127	18.7018
	TCNN	75	38.935	18.7018
	NCNN	199	38.9675	18.7018
$\sigma = 0.75$	HNN	121	30.6347	15.8694
	TCNN	125	33.0543	15.8694
	NCNN	206	33.7814	15.8694
$\sigma = 1$	HNN	226	27.3437	14.7626
	TCNN	137	28.4625	14.7626
	NCNN	151	29.4505	14.7626

Table 1. Numerical denoising results of the circle image (PSNR_r: PSNR of the denoised image, PSNR_d: PSNR of the noisy image)

The next experiment was conducted on the real image. The real image is the Lena image of size 128×128 with M = 256 gray levels. Its label set is $\mathcal{L} = \{0, 1, 2, \dots, 255\}$. Three noisy Lena images were generated by adding zero-mean i.i.d. Gaussian noise with standard deviation of $\sigma = 8$, $\sigma = 16$ and $\sigma = 24$ respectively.

The denoising process of the noisy Lena images is the same as the circle image denoising process. The parameters that we used for the NCNN of the Lena image denoising process are: k = 1, $\epsilon = 0.01$, $\alpha = 0.0001$, $I_0 = 0.65$, $z^{(0)} = 0.05$, $n^{(0)} = 0.01$. β is increased from 1 to 50 according to $\beta \leftarrow 1.01\beta$.



(e)

Fig. 2. Denoising of the circle image with noise level $\sigma = 0.5$: (a) Original image. (b) Noisy image. (c)-(e) Denoised images using the HNN, the TCNN and the NCNN, respectively



(e)

Fig. 3. Denoising of the circle image with noise level $\sigma = 0.75$: (a) Original image. (b) Noisy image. (c)-(e) Denoised images using the HNN, the TCNN and the NCNN, respectively





(a)











Fig. 4. Denoising of the circle image with nose level $\sigma = 1$: (a) Original image. (b) Noisy image. (c)-(e) Denoised images using the HNN, the TCNN and the NCNN, respectively

The decreasing rate of z and n, β_z and β_n are 0.005. For the TCNN and the HNN we use the same parameters as the NCNN except that n = 0 for the TCNN, n = 0 and z = 0 for the HNN. The MRF pair-site clique potential parameter $\beta_c = 8$.

Table 2 shows the required iteration numbers and the peak signal-to-noise ratio (PSNR) of the denoised images. The higher the PSNR, the better the image quality. It can be seen from the table that the NCNN offers the best performance. The PSNR of the restored image using the NCNN is higher than those of the restored images using the HNN and the TCNN. In addition, the NCNN use less iterations to converge than the HNN and the TCNN. The denoised images are shown in Fig. 5-7.

Noise Level		Iterations	PSNR_r	PSNR_d
$\sigma = 8$	HNN	950	31.2448	30.0532
	TCNN	835	31.5933	30.0532
	NCNN	807	32.0016	30.0532
$\sigma = 16$	HNN	1489	25.6227	23.0549
	TCNN	1173	25.9077	23.0549
	NCNN	1114	26.7554	23.0549
$\sigma = 24$	HNN	2054	22.8195	20.5695
	TCNN	1571	22.9979	20.5695
	NCNN	1862	23.1777	20.5695

Table 2. Numerical denoising results of the Lena image (PSNR_r: PSNR of the denoised image, PSNR_d: PSNR of the noisy image)

5 Conclusion

A new neural network, called noisy chaotic neural network (NCNN), is used to address the MAP-MRF formulated image denoising problem. SCSA effectively overcomes the local minima problem. We have shown that the NCNN gives better quality solutions compared to the HNN and the TCNN.

References

1. Andrews, H. C., Hunt, B. R. (1977) Digital Image Restoration. Englewood Cliffs, N J, Prentice-Hall



(e)

Fig. 5. Denoising of the Lena Image with noise level $\sigma = 8$: (a) Original image. (b) Noisy image. (c)-(e) Denoised images using the HNN, the TCNN and the NCNN, respectively





Fig. 6. Denoising of the Lena Image with noise level $\sigma = 16$: (a) Original image. (b) Noisy image. (c)-(e) Denoised images using the HNN, the TCNN and the NCNN, respectively





Fig. 7. Denoising of the Lena Image with noise level $\sigma = 24$: (a) Original image. (b) Noisy image. (c)-(e) Denoised images using the HNN, the TCNN and the NCNN, respectively

- Rosenfeld, A., Kak, A. C. (1982) Digital Picture Processing, 1, Academic Press, 2nd edition
- Geman ,S., Geman, D. (1984) Stochastic relaxation, gibbs distributions and the bayesian restoration of images. IEEE Transactions on Pattern Analysis and Machine Intelligence, 66, 721-741
- Li, S. Z. (1998) Map image restoration and segmentation by constrained optimization. IEEE Transactions on Image Processing, 712, 1730-1735
- 5. Wang, H., Li, S. Z., Petrou, M. (1990) Relaxation labeling of markov random fields. IEEE Transactions on Systems, Man and Cybernetics, **20**, 709-715
- 6. Peterson, C., Soderberg, B. (1992) Combinatorial Optimization with Neural Networks. Blackwell
- Hopfield, J. J., Tank, D. W. (1985) Neural computation of decisions in optimization problems. Biol. Cybern., 52, 141-152
- Chen, L., Aihara, K. (1995) Chaotic simulated annealing by a neural network model with transient chaos. Neural Networks, 86, 915-930
- 9. Chen, L., Aihara, K. (1994) Transient chaotic neural networks and chaotic simulated annealing. Towards the Harness of Chaos, 347-352
- Aihara, K., Tokuda, I., Nagashima, T. (1998) Adaptive annealing for chaotic optimization. Phys. Rev. E, 58, 5157-5160
- Kajiura, M., Anzai, Y., Akiyama, Y., Yamashira, A., Aiso, H. (1991) The gaussian machine: a stochastic neural network for solving assignment problems. J. Neural Network Comput., 2, 43-51
- Wang, L., Tian, F. (2000) Noisy chaotic neural networks for solving combinatorial optimization problems. Proc. International Joint Conference on Neural Networks, 4, 4037-4040
- Hammersley, J. M., Clifford, P. (1971) Markov field on finite graphs and lattices. unpublished Manuscript
- Li, S. Z. (1995) Markov Random Field Modeling in Computer Vision. Springer-Verlag