Cellular Nonlinear Network Model for Image Denoising

Angela Slavova, Victoria Rashkova

1 Introduction

Recently many mathematical models for image processing have been widely applied in computer visualization. The nonlinear diffusion partial differential equation have been broadly applied in image processing since the first model was introduced in 1987 [3]. Through the time evolution, the diffusion can effectively remove the noise as well as having edge enhancement simultaneously. Since then various nonlinear diffusion filters have been widely proposed in implementing the image denoising / enhancement, edge detection and flow filed visualization . The common feature for nonlinear diffusion model is that the diffusion coefficient is small as the gradient of image is large. However the diffusion coefficient is a function of the convolution of the Gaussian kernel and solution such that this requires an extra cost in computing the nonlinear diffusion coefficient. In the numerical experiments we find that when using the nonlinear diffusion model in denoising the noise is not quite good [6]. Hence we propose a convection- diffusion filter by adding a convection term in the modified diffusion equation as a physical interpretation for removing the noise. The aim of this paper is to focus on the noise removal algorithm for extracting the target information (image) precisely. The main idea of our model algorithm is to diffuse the noise by following the convection direction during time evolution. To prevent the numerical layer in the discontinuities on the relative coarse grids we use the Cellular Nonlinear Network (CNN) approach [1,2,4].

2 Perona- Malik type nonlinear diffusion equation and its CNN model

Dynamic properties of Perona-Malik based filters are summarized in [3]. It is well known that it converges toward a constant steady state solution, representing the average value of the initial image. In order to obtain a non trivial output image, the system evolution has to be stopped after a finite time (usually called scale). In the most general case, the scale depends on the object and on the characteristics of the input image, and hence is no *a priori* known time for stopping the image processing. Let the noisy image be a given scaled intensity map $u_0(x) : \Omega \to [0, 255]$ for the image domain $\Omega \in \Re^2$. The nonlinear diffusion equation was first proposed by Perona and Malik in filtering the noise. They built a sequence of continuous images $u(x, t)$ on the abstract scale t and through the nonlinear diffusion equation to remove the noise during the scaling (ï¿$\frac{1}{2}$timeï¿$\frac{1}{2}$) revolution. Many of such nonlinear diffusion filtering models have been implemented. We briefly review this model as follows. The Perona-Malik type nonlinear isotropic diffusion equation (ND) is:

$$\begin{cases} u_t(x,t) - div(g(|\nabla G_\sigma * u|)\nabla u(x,t)) = 0 & \text{in} \quad \Omega \times I \\ \frac{\partial u}{\partial n}(x,t) = 0 & \text{on} \quad \partial\Omega \times I \\ u(x,0) = u_0(x) & \text{on} \quad \Omega \end{cases} \tag{1}$$

where the initial value $u_0(x)$ is the given noisy image in the gray level, $I = [0,T]$ is the scaling (time) interval for some $T > 0, \Omega$ is a simply bounded rectangular domain with boundary $\partial\Omega$ and n is the outward unit normal vector to $\partial\Omega; g$ is a given non-increasing function. There are several choices for $g(s)$. Select a monotonic decreasing function

$$g(s) = \frac{1}{1+s^2}$$

and in [3] it is introduced the Gaussian kernel,$G_\sigma * u$,for the existence and uniqueness of (1). Thus, the diffusion coefficient $g(|\nabla G_\sigma * u|)$ is inhibited as the gradient of image intensity is big i.e. the diffusion coefficient is small around the image edge. Hence ND preserves the edges of image and protects the brightness of the image simultaneously. Perona and Malik considered the Galerkin finite element method for the discretization of (1). We shall apply polynomial CNN in order to study its dynamics.
The following diffusion functions were proposed by Perona and Malik:

$$g(\|\nabla u\|^2) = e^{-(\frac{\|\nabla u\|}{k})^2}, g(\|\nabla u\|^2) = [1 + (\frac{(\|\nabla u\|}{k})^2]^{-1} \tag{2}$$

Let us suppose that the continuous space domain is composed by $M \times M$ points arranged on a regular grid, u_{ij} represents the pixel value.
In order to implement a general polynomial CNN architecture, let us consider the following basis function:

$$f(z) = 1 - |\frac{1}{2}(|\frac{z}{m} + 1| - |\frac{z}{m} - 1|)|$$

and let approximate the $g(.)$ functions (2) with the following expression:

$$\gamma(z) = \sum_{p=1}^{Q} C_p f^p(z)$$

We obtain a general nonlinear PDE based polynomial CNN model:

$$\frac{du_{ij}(t)}{dt} = \sum_{(kl)\in N_{ij}} \Gamma_{kl}(u_{kl} - u_{ij}) \tag{3}$$

$$\Gamma_{kl} = \sum_{p=1}^{Q} \frac{C_p}{2h^2}[f^p(\|\widetilde{\nabla} u_{k,l}\|^2) + f^p(\|\widetilde{\nabla} u_{i,j}\|^2)]$$

The polynomial CNN model (PCNN) (3) approximate the functions $g(.)$ with the expression $\gamma(.)$ that turns out to be different from zero only for $|z| = \|\widetilde{\nabla} u_{i,j}\|^2 < m$ This practical approximation presents the advantage of stopping the evolution of the image when the approximated gradient magnitude $\|\widetilde{\nabla} u_{i,j}\|^2$ is greater than the threshold m. Hence, the output image exhibits a segmented structure. The above behavior is possible because the PCNN system (3) presents ore than one equilibrium point and for each initial image, the output corresponds to one of these equilibria.

Conclusion 1 *The PCNN model (3) exhibit the coexistence of the constant average value equilibrium point (that is only admissible for the Perona-Malik discretized models) and of an infinite set of equilibrium points. As a consequence, the correct output is obtained without stopping the evolution of the system. This represents a significant advantage from the algorithmic point of view.*

We obtain the following simulation results for different values of the cell parameters:

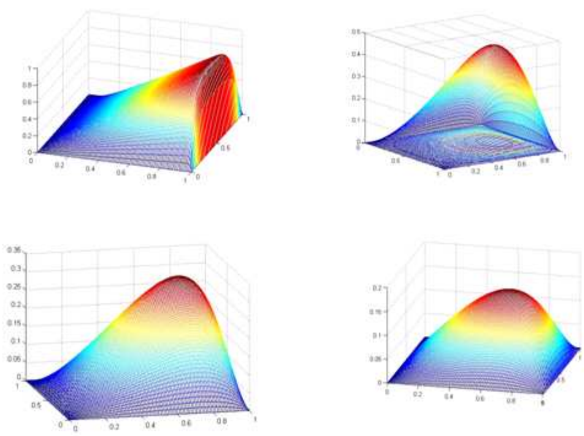

Fig.1. Simulations of the CNN algorithm for the problem (1).

Example 1. Consider the following singularly perturbed boundary value problem

$$-\varepsilon \Delta u + b u_x + c u = 0, \quad \text{in} \quad \Omega \equiv (0,1)^2 \tag{4}$$
$$u = 0, \quad \text{on} \quad \partial\Omega.$$

In order to construct a robust numerical method for the considered problem, it is of key interest to have information on a behavior of the solution. The state equation of the CNN model of (4) is:

$$-\varepsilon A_1 * u_i + b * A_1 * u_i + c * u_i = 0 \tag{5}$$

Applying CNN algorithm we obtain the following simulation results:

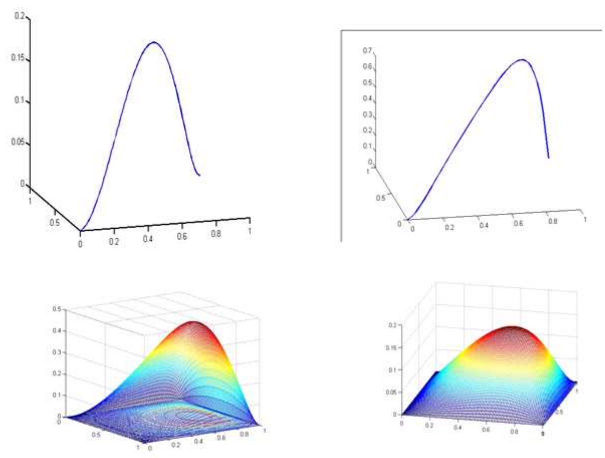

Fig.2. Simulation of the CNN algorithm of (4).

Example 2. We consider the following example on the layer-adapted mesh:

$$-\varepsilon \Delta u + (1 + x^3)u_x + (1 + xy)u = 0, in\Omega \tag{6}$$

$$u = 0, on\partial\Omega$$

CNN model of the above system is:

$$-\varepsilon A_1 * u_i + (1 + x^3)A_1 * u_i + (1 + xy)u_i = 0 \tag{7}$$

$$1 \leq i \leq N$$

We obtain simulation results of (6) on Fig.3.

Remark 1. We consider a CNN programmable realization allowing the calculation of all necessary processing steps in real time. The network parameter values of CNN models are determined in a supervised optimization process.During the optimization process the mean square error is minimized using Powell method and Simulated Annealing [5]. The results are obtained by the CNN simulation system MATCNN applying 4th order Runge-Kutta integration.

Acknowledgments. This work is supported by the Bulgarian National Science Fund under grant DID 02/15.

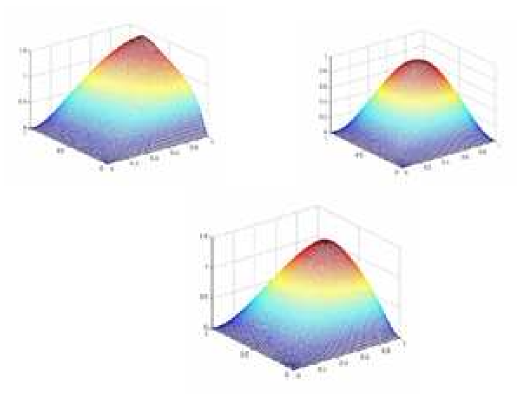

Fig.3. Simulation of the CNN algorithm of (6).

References

[1] Chua L.O., Yang L., Cellular Neural Network: Theory and Applications, *IEEE Trans. CAS*, vol. 35, pp. 1257-1290, Oct. 1988.

[2] Chua L.O., Hasler M., Moschytz G.S., Neirynsk J., Autonomous cellular neural networks: a unified paradigm for pattern formation and active wave propagation, *IEEE Trans. CAS-I*, vol. 42, N 10, pp. 559-577, Oct. 1995.

[3] P. Perona, J. Malik, Scale space and edge detection using anisotropic diffusion. In: Proceedings of IEEE Computer Soc. Workshop and Computer Vision, 1987, pp. 16-22.

[4] A.Slavova,Cellular Neural Networks: Dynamics and Modelling, Kluwer Academic Publishers, 2003.

[5] R. Tetzlaff, F. Gollas, Modeling complex systems by reaction-diffusion Cellular Nonlinear Networks with polynomial Weight-Functions,Proc. IEEE CNNA2005, 2005.