*Int. J. Open Problems Compt. Math., Vol. 2, No. 3, September 2009*
*ISSN 1998-6262; Copyright ©ICSRS Publication, 2009*
*www.i-csrs.org*

# Digital Image Inpainting Using Cellular Neural Network

**P.Elango [a] and K.Murugesan [b]**

[a] Department of Information Technology
Perunthalaivar Kamarajar Institute of Engineering and Technology (PKIET)
Karaikal – 609 603, Puducherry, INDIA.

[b] Department of Mathematics
National Institute of Technology
Tiruchirappalli – 620 015, INDIA.

**e-mail:** elanalin_74@yahoo.com

***Abstract***

*Digital Image inpainting methods provide a means for reconstruction of small damaged portions of an image. Image or video resources are often received in poor conditions, mostly with noise or defects making the resources difficult to read and understand. Some methods are presented that can be used for the reconstruction of damaged or partially known images. We propose an effective algorithm with CNN, that can be used to inpainting digital images or video frames with very high noise ratio. Noises inside the cell with different sizes are inpainted with different levels of surrounding information. So, the result showed that an almost blurred image or unrecognized cell can be recovered with visually good effect. The proposed method takes the possibility of direct implementation of an existing CNN chip into account, in a single step, by using 3x3 dimensional linear reaction templates. This same method can be further used for processing motion picture with high percentage of noise.*

**Keywords:** Image inpainting, Cellular Neural Network, Digital images, isophotes, PSNR ratio.

# 1 Introduction

Cellular Neural Networks (CNN) is analog, continuous time, nonlinear dynamic systems and formally belongs to the class of recurrent Neural Networks. Since their introduction in 1988 by Chua and Yang [1], they have been the subjects of intense research. The Cellular Neural Network (CNN) is an artificial neural network of the nearest neighbor interaction type. It has been widely used for image processing, pattern recognition, moving object detection, target classification, signal processing, augmented reality and solving partial differential equations etc. The Cellular Neural Network Complementary Metal-Oxide Semiconductor CMOS array was implemented by Anguita et al [2-7]. The design of a Cellular Neural Network template is an important problem, and has received wide attention [1 - 9] in the recent years. This paper reports an efficient Algorithm exploiting the latency properties of Cellular Neural Networks along with popular numerical approximation algorithms. The dynamic equation of a cell $C(i, j)$ in an M x N Cellular Neural Network is given by Chua and Yang [1][9] and is shown in Figure-1.(a)-(b).

$$C\frac{dx_{ij}(t)}{dt} = -\frac{1}{R_x}x_{ij}(t) + \sum_{C(k,l)\in N_r(i,j)} A(i,j;k,l)\,y_{kl}(t) + \sum_{C(k,l)\in N_r(i,j)} B(i,j;k,l)\,u_{kl} + I \qquad (1)$$

$$y_{ij}(t) = \tfrac{1}{2}\left[\left|x_{ij}(t)+1\right| - \left|x_{ij}(t)-1\right|\right],\ 1\le i\le M,\ 1\le j\le n \qquad (2)$$

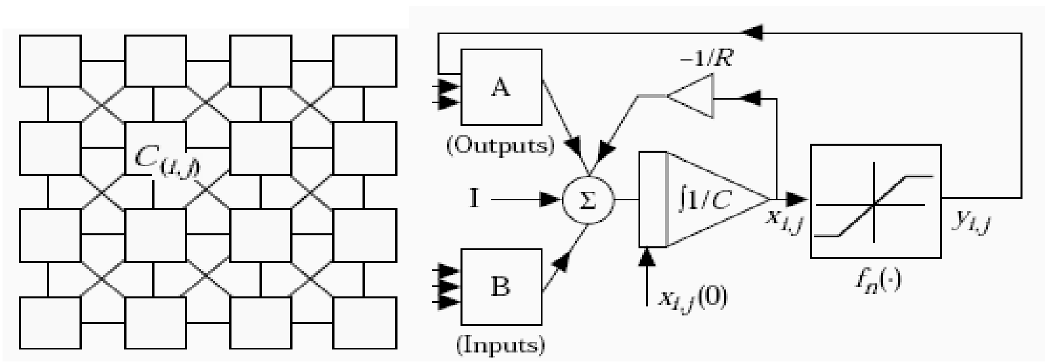


Figure 1(a) Figure 1(b)

Where $x_{ij}$ , $y_{ij}$ and $u_{ij}$ are the state voltage, output voltage and input voltage respectively and they are functions of time t. $R_x$ is a linear resistance, C is a linear capacitor, and $A(i, j; k, l)$ and $B(i, j; k, l)$ are the transconductances of the output and input voltages of $C(k,l)$ with respect to $C(i,j)$ called the cloning templates of CNN. $N_r(i,j)$ denotes the $r^{th}$ - neighbor of $C(i,j)$ and I is an independent current source. From equation (2) one can discern that the output voltage is nonlinear. Now let us rewrite the cell equation (1) as follows:

$$C\frac{dx_{ij}(t)}{dt} = -f\left[x_{ij}(t)\right] + g(t) \tag{3}$$

Where $f\left[x_{ij}(t)\right] = \frac{1}{R_x} x_{ij}(t),$ (4)

$$g(t) = \sum_{\substack{C(k,l)\in N_r(i,j)\\ C(k,l)\neq C(i,j)}} A(i,j;k,l)\,y_{kl}(t) + \sum_{C(k,l)} B(i,j;k,l)\,u_{kl} + I \tag{5}$$

In this Paper, we describe the digital image inpainting techniques and existing methods in section 2, Reconstruction of damaged images using CNN methods in section 3, Testing and Experimental results in section 4 and finally conclusions are presented in Section 5.

## 2 Digital Image Inpainting Techniques

Reconstruction of missing or damaged portions of images is an ancient practice used extensively in artwork restoration. This activity, also known as inpainting or retouching, consists of filling in the missing areas or modifying the damaged ones in a manner non-detectable by an observer not familiar with the original images. The goal of inpainting algorithms varies, depending on the application, from making the inpainted parts look consistent with the rest of the image, to making them as close as possible to the original image, restoration of photographs, films and paintings, to removal of occlusions, such as text, subtitles, stamps and advertisements from images. In addition, inpainting can also be used to produce special effects. While, traditionally skilled artists have performed image inpainting manually, currently digital techniques are used, e.g. for automatic restoration of scratched films.

Most inpainting methods work as follows.

As a first step the user manually selects the portions of the image that will be restored. Then image restoration is done automatically, by filling these regions in with new information coming from the surrounding or cell in our case. In order to produce a perceptually plausible reconstruction, an inpainting technique must attempt to continue the isophotes (line of equal gray value) as smoothly as possible inside the reconstruction region. In other words the missing region should be inpainted so that inpainted gray value and gradient extrapolate the gray value and gradient outside the region. Several inpainting methods are based on the above ideas. Bertalmio et al. [12-13] first introduced the notion of digital image inpainting and used third order partial differential equations (PDE) [10] to diffuse the known image information into the missing regions. Later, this inpainting approach was modified to take into account the direction of the level lines, called isophotes, [1] and to relate it to the Navier-Stokes flow [11].

This operation propagates information into the masked region while preserving the edges. In [19] anisotropic diffusion is used to preserve edges across the inpainted regions. For further discussion of various methods, see the recent survey articles [16-25]. The algorithms proposed in the literature differ depending on the assumptions made about the properties of the image. For example, the total variation (TV) inpainting model proposed in [9], based on the Euler–Lagrange equation, employs anisotropic diffusion based on the contrast of the isophotes inside the inpainting domain. This model, designed for inpainting small regions, does a good job at removing noise, but does not connect broken edges (single lines embedded in a uniform background). The Curvature-Driven Diffusion (CDD) model, extends the TV algorithm to also take into account geometric information of isophotes when defining the 'strength' of the diffusion process, thus allowing the inpainting to proceed over larger areas. Although some of the broken edges are connected by the CDD approach, the resulting criteria for stopping the inpainting, the process is constantly applied to all masked pixels, regardless of the local smoothness of the region. As a result, computationally expensive operations might be unnecessarily performed, resulting in lengthy processing time. Thus, although non-linear PDE-based image restoration methods have the potential of systematically preserving edges, fast numerical implementations are difficult to design.

Considerable computing power is necessary to solve the image processing task described by variational computing. For the time being serial processing does not provide us with methods implementable in real-time. The cellular neural network proved to be very useful regarding real-time image processing. The reduction of computing time, due to parallel processing, can be obtained only if the processing algorithm can be implemented on CNN-UC.

Even if variational methods are used, the determination of templates ensuring the gray-scale image and the desired processing remains a difficult problem, since the fact that the actually existing CNN chip can use only linear templates having 3*3 dimension has to be taken into consideration. In some cases templates satisfy these conditions by using nonlinear templates. Effective CNN implementation is still possible in CNN algorithms.

# 3 The Reconstructions Of Damaged Images Using Cellular Neural Network Model

Considering an image or a video frame as a large Damaged Image Block (DIB) is placed in the CNN as shown in Fig.2. Each DIB cell can be further subdivided into different Image Blocks (IBs), each of which may or may not contain damaged pixels. Furthermore, each Image Block is subdivided into several Pixel Blocks (PBs) which are elementary objects to be inpainted. The recursive algorithm is shown in Fig.3

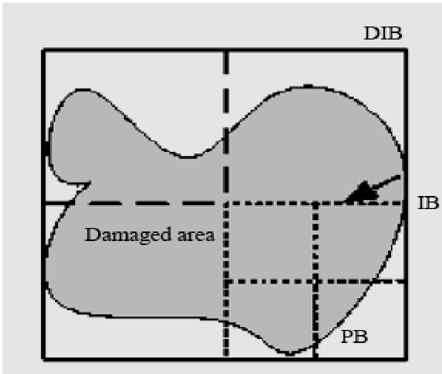


Figure.2 Damaged Image Block (DIB), Image Block (IB) and Pixel Block (PB) with CNN cell.

$$SR\_Inpaint(DIB) \leftrightarrow$$
$$DIB = \in \text{ or}$$
$$\forall IB \in DIB \cdot (var(IB) > \alpha \rightarrow SR\_Inpaint(IB) \text{ or}$$
$$\forall PB \in IB \cdot (DamagePercent(PB) > \beta_2 \rightarrow Fill(PB, MeanColor(IB))$$
$$\beta_2 \geq DamagePercent(PB) > \beta_1 \rightarrow Fill(PB, MeanColor(PB))$$
$$\beta_1 \geq DamagePercent(PB) \rightarrow Interpolate(PB))$$

Figure. 3 Procedure for CNN based Inpainting method

The basic assumption is that color difference is a strong indication of the degree of details in an Image Block. The threshold value alpha (α) sets the criterion on whether a recursive call to the CNN based inpainting algorithm is required (the algorithm terminates when the DIB is small). In our implementation, the value of α is a percentage of color variance of an IB (i.e., the maximum var(IB) is 100). If the color variance of IB is greater than the threshold value α, the algorithm is called recursively to handle the next level of details. Otherwise, the algorithm further divides an IB into several pixel blocks in the CNN cell (i.e., PBs). Another criterion is the percentage of damaged pixels. We argue that if the percentage is too high, using surrounding color information to fix a pixel is less realistic as compared to using a global average color. In some severe cases, it is very difficult to use neighboring cell colors. Note that, both thresholds are adjustable for the sake of analysis. The algorithm iterates through each of the PBs in an IB. If the percentage of damaged pixels in a PB is too high (i.e., greater than $\beta2$), the mean IB color is used [i.e., Mean Color (IB)]. One example is that the entire PB is damaged, so we must use the mean IB color. The function Damage Percent (PB) simply counts the number of damaged pixels in a PB. And, the function Fill (PB, C) takes a Pixel Block and a color C, and fills the Pixel Block with the color. Alternatively, if the percentage is still high (i.e., greater than $\beta1$), the mean PB color is used. Note that, the computation of mean colors does not take damaged pixels into account. If the percentage is low enough (i.e., less than $\beta1$), neighbor pixels and cells are used for interpolation. The function Interpolate (PB) implemented in our algorithm uses a bi-linear interpolation technique. We further take an optimization step to improve the algorithm. When the filling function is called, we add noise on the boundaries of Pixel Blocks. Our proposed CNN based image inpainting algorithm is called again to remove these bounding boxes. Thus, block effect is reduced. There are three thresholds in the above algorithm, $\alpha$, $\beta1$ and $\beta2$. We use all combinations of the following values:

$\alpha$ = 50, 60, 70, 80

$\beta1$= 65, 70, 80, 85, 90

$\beta2$= 95

The selection of $\beta2$ is aimed at testing the usage of mean color. Unless a pixel block is completely damaged, the mean color should be used. Thus, the selection of $\beta2$ should be high. Since $\beta1< \beta2$, we select the values of $\beta1$ accordingly. The threshold $\alpha$ is used to check the color variance. We try to cover a wide spectrum. The combinations of the above thresholds are all tested using more than 1000 pictures. The values of $\alpha$, $\beta1$ and $\beta2$ have great impact on the outcome. In general, if $\alpha$ is less than 75, the average PSNR values of repaired pictures with respect to other parameters are stable. So we use $\alpha$ =80 in our implementation of an automatic inpainting tool. We chose $\beta2$=95 through our analysis. This means that unless the percentage of damaged pixels in a pixel block is higher than 95, the mean color of an outside big block should not be used. The value of $\beta1$ is critical. If $\beta1$ is less than 60, the result is not as good as expected. However, the PSNR values of the fixed pictures become stable when the value of $\beta1$ becomes 80, 85, or 90. Conclusively, we choose $\alpha$ =80, $\beta1$=85, and $\beta2$=95.

# 4 Experimental Results

In this section the experimental results obtained by using the "CadetWin" (CNN Application Development Environment and Toolkit under Windows) are presented with Pentium Core2duo Processor. Whatever the chosen image restoring method is, the precision of the restoration should be as good as possible, but at the same time, it is desirable that the dimensions of the image's holes that are restored should be as large as possible, that is the interpolation propagation distance should be great. The above is the evaluation basis of the methods proposed for the restoration of the damaged image. An example of this is shown in Figure.4. In Figure 4 a) many small areas are missing. The result in Figure 4 b) is nearly perfect. The only problem is that the edge of her shoulder has some dark spots.

(a) (b)

Figure 4. Experimental results CNN Approach (a) Original image (40% defects) (b) Restored image.

Another example of a player action is given in Figure.5. In this example Figure. 5(a) is the original image also it is a perfect image Figure 5(b) is the inpainting region to be marked by violet and cyan colors. Figure 5(c) is the final restored image by using our algorithm.

(a) (b) (c)

Figure. 5 Experimental results by CNN based inpainting algorithm. (a) Original image (b) The image with inpainting region marked (c) Final Restored image.

The analysis takes another perspective on the noise percentage of damaged pixels. Figure 6 presents our experimental results. The first row shows the damaged images and the percentages of noise pixels. The second row presents the inpainted images and PSNR values. The results showed that even with very high percentage of noise, the picture is still visible and it can be shown in the Figure 6 and Table-1.

(a) (b) (c) (d)

Figure6. Experimental results in different noise conditions with CNN.

(a) Noise ratio=51.6%, PSNR=30.1 dB;
(b) Noise ratio=70.1%, PSNR=28.16 dB;
(c) Noise Ratio=79.6%, PSNR=30.9 dB;
(d) Noise Ratio=88.3%, PSNR=22.1 dB

| Figure | Noise Ratios | Peak Signal to Noise Ratio (PSNR) values | |
|---|---|---|---|
| | | Damaged | Recovered |
| 6(a) | 51.6% | 5.66 | 25.31 |
| 6(b) | 70.1% | 6.10 | 24.76 |
| 6(c) | 79.6% | 7.1 | 23.76 |
| 6(d) | 88.3% | 8.5 | 21.90 |

Table 1. Experimental results of our proposed method with different Noise %.

# 5 Conclusion

This paper presents an effective algorithm for image inpainting, which achieves visually good result, even with very high percentage of defects in the pictures. We have tested more than 1000 pictures of different categories, including Chinese and western paintings, photos, and cartoon drawings. The experimental results proved that Cellular Neural Network (CNN) based inpainting approach is especially powerful for stilled photos. This result encourages us to take a further step to extend the proposed mechanism for video inpainting. Due to parallel processing, huge computing power is achieved, regardless of the dimensions of the images. Increasing the efficiency is possible through combining these methods with a nonlinear template, which ensures local properties spreading area growth, making it possible to process even images with large holes. Our objective is not to detect spikes or long vertical lines commonly reported in the literature. We aim to develop a technique to recover video signal with very high percentage of noise and achieve visually good output.

# 6 Open Problems

In this paper, we have surveyed all the recent inpainting techniques, based on that we deduced a model for combining Cellular Neural Network Ideas with different images. The classical CNN approach for Digital Image Inpainting framework has proven to be very effective in designing and improving image inpainting.

We have explained that the fundamental techniques for inpaintings such as Isophotes (line of equal gray value), PDE, Navier – Stokes equation, Total Variation (TV) methods, and Curvature Driven Diffusion (CCD) ideas in the Literature Survey. As a result, it increases the execution time due to performing computationally expensive operations again and again. Since the serial processing methods are not suitable for real-time operations of image, our proposed CNN technique is the best one for real time image inpainting due to its parallel computation.

Finally we post two interesting open problems.

(i) **Video inpainting**. Video inpainting is a crucial and challenging area of image processing. It has profound applications in the movie industry, surveillance analysis, and dynamic vision analysis. The first open problem is: How to calculate and analyze the Peak Signal Noise Ratio (PSNR) values for video images with Cellular Neural Network as tool.

(iii) D**igital realization**. Throughout our problem formulation, we found Numerical PDE has been a core computational tool for all the geometric inpainting models. The second open problem concerns fast and efficient digital implementation of the associated PDE's, especially for the high order ones and also we can utilize Cellular Neural Network Techniques for efficient digital realization.

## References

[1] L.O. Chua, and L. Yang, "Cellular Neural Networks: Theory and Applications", IEEE Trans. on Circuits and Systems, Vol.35, 1998, pp. 1257-1290.

[2] M. Anguita, F. J. Pelayo, E. Ros, D. Palomar and A. Prieto, “VLSI implementations of CNNs for image processing and vision tasks: single and multiple chip approaches”, IEEE International Workshop on Cellular Neural Networks and their Applications, 1996, pp. 479 - 484.

[3] M. Anguita, F. J. Pelayo, F. J. Fernandez and A. Prieto, “A low-power CMOS implementation of programmable CNN's with embedded photo sensors”, IEEE Transactions on Circuits Systems I: Fundamental Theory and Applications, Vol. 44(2), 1997, pp. 149 - 153.

[4] M. Anguita, F. J. Pelayo, E. Ros, D. Palomar and A. Prieto, “Focal-plane and multiple chip VLSI approaches to CNNs”, Analog Integrated Circuits and Signal Processing, Vol. 15, No. 3, 1998, pp. 263 - 275.

[5] M. Anguita, F. J. Pelayo, I. Rojas and A. Prieto, “Area efficient implementations of fixed template CNN's”, IEEE Transactions on Circuits and Systems Fundamental Theory and Applications, Vol. 45(9), 1997, pp. 968 - 973.

[6] M. Anguita, F. J. Fernandez, A. F. Diaz, A. Canas and F. J. Pelayo, “Parameter configurations for hole extraction in cellular neural networks”, Analog Integrated Circuits and Signal Processing, Vol. 32(2), 2002, pp.149 - 155.

[7] Anguita, M., Pelayo, F.J., Rojas, I., and Prieto, A., “Area efficient implementations of fixed template CNN’s”. IEEE Transactions on Circuits and Systems Fundamental Theory and Applications vol. 45(9), 1997, pp.968–973.

[8] Fajfar, F., Bratkovic, T., Tuma, and T., Puhan, J., “A rigorous design method for binary cellular neural networks”, International Journal of Circuit Theory and Applications” vol 26(4), 1998, pp. 365–373.

[9] T. Roska, and L.O. Chua, "The CNN universal machine an analogic array computer", IEEE Trans. on Circuits and Systems, Vol. 40, 1993, pp. 163-173.

[10] L.C.Evans, “Partial Differential Equations (PDE).” , Graduate Studies in Mathematics. 19. Providence RI: American Mathematical Society (AMS),1998.

[11] V. Casellas, J.M. Morel, G. Sapiro, and A. Tannenbaum, "Introduction to the special issue on partial differential equations and geometry-driven diffusion in image processing and analysis", IEEE Trans. on Image Processing, Vol. 7, 1998, pp. 269-273.

[12] Bertalmio, M., Vese, L., Sapiro, G., and Osher, S., "Simultaneous structure and texture image inpainting". IEEE Transactions on Image Processing, vol. 12(8), 2003, pp.882-889.

[13] Bertalmio, M., Sapiro, G., Caselles, V., and Ballester, C., "Image Inpainting". ACM SIGGRAPH Conference on Computer Graphics, SIGGRAPH 2000, pp.417-424.

[14] Bornard, R., Lecan, E., Laborelli, L., and Chenot, J.H., "Missing Data Correction in Still Images and Image Sequences". Proc. ACM International Conference on Multimedia, 2002, pp.355-361.

[15] Oliveira, M.M., Bowen, B., McKenna, R., and Chang, Y.S., "Fast Digital Image Inpainting". International Conference on Visualization, Imaging and Image Processing, 2001, pp. 261-266.

[16] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J.Verdera, "Filling-in by joint interpolation of vector fields and gray levels", IEEE Trans. Image Processing, vol. 10(1), 2001, pp. 1200-1211.

[17] T. Chan, and J. Shen, "Mathematical Models for Local Non texture Inpaintings", SIAM Journal of Applied Mathematics, vol. 62(3), 2001, pp. 1019-1043.

[18] Shen, J. "Inpainting and the fundamental problem of image processing," SIAM News, vol. 36(5), 2003, pp. 110-121.

[19] Murugesh, V., and Murugesan, K.: "Comparison of Numerical Integration in Raster CNN Simulation". LNCS, vol. 3285, 2004, pp. 115–122.

[20] Chan, T., and Shen, J. "Non-texture inpainting by Curvature-Driven Diffusions (CCD)", Journal of Visual Communication. Vol. 12, 2001, pp. 436–449.

[21] J. Kaipio, and E. Somersalo, "Computational and Statistical Inverse Problems", Springer, Berlin vol.160, 2005.

[22] Gacsadi, and P. Szolgay, "Interpolation of 2D signals using CNN", Proceedings of the European Conference on Circuit Theory and Design, (ECCTD'01), Espoo, 2001, pp. 349-352.

[23] Jian-Feng Cai, Raymond H. Chan, Zuowei Shen "A framelet-based image inpainting algorithm" Applied and Computational Harmonic Analysis, Volume 24, Issue 2, March 2008, pp. 131-149.

[24] Charles K. Chui "An MRA approach to surface completion and image inpainting" Applied and Computational Harmonic Analysis, Volume 26, Issue 2, March 2009, pp. 270-276.

[25] Lucia Maddalena,Alfredo Petrosino and Giuliano Laccetti "A fusion-based approach to digital movie restoration" Pattern Recognition, Volume 42, Issue 7, July 2009, pp.1485-1495.