International Journal of Scientific Engineering and Technology (ISSN : 2277-1581)
www.ijset.com, Volume No.1, Issue No.2 pg:143-147 01 April 2012

A Neural Network Based Image Abstraction Technique

Trapti Sahu, Shabahat Hasan
R.K.D.F.I.S.T. Bhopal (M.P.) India
trapti.jec@gmail.com

Abstract: image abstraction is a method of enhancing or highlighting the visualization of main information contents of image and smoothing the other portion. It is quite important for many multimedia applications like cartooning and animation where symbolic presentation matters; hence it is an important part in the field of image processing. In this paper we are proposing a neural network based approach which not only abstract the image but also avoids the enhancement of noise. This particular task is performed by empirically training the neural network for noise patterns on images by mixing the plane image with noise finally the algorithm is simulated on MATLAB and results showing that the proposed technique is fast and efficient in performing the particular task.

Keywords: image enhancement, Neural Network, empirical analysis.

1. Introduction

Image abstraction is a technique of image acquisition in which except main information abstract, all other parts are filtered (smoothed) mostly this technique is used for Non-photorealistic rendering (NPR) which is an area of computer graphics that focuses on enabling a wide variety of expressive styles for digital art. In contrast to traditional computer graphics, which has focused on photorealism, NPR is inspired by artistic styles such as painting, drawing, technical illustration, and animated cartoons. The method should be in such stylistic fashion, so that the output of abstraction can serve the purpose of communicating the messages as well as pleasing the viewer's eyes. Herman and Duke [1] used the term "minimal graphics" to describe the benefit of such meaningful abstraction and stylization.

There is considerable interest in artistic rendering outside of the research community. NPR can be found more and more often in the entertainment industry. Scenes or objects are commonly composited into hand-animated cartoons with cell style shading and outline contours to emulate the style of manually drawn work simple NPR effects are typical in many modern video games. Most of the techniques used for it are fairly straightforward and are not difficult to emulate algorithmically [2]. However, this paper presents a technique which can be used for abstraction of noisy images without the enhancement of noise generated information; other important aspect of the technique is that its complexity is maintained lower with faster execution speed and lower resources requirement.

2. Related Work

Many of the existing image-based NPR techniques are intended to serve artistic purposes, that is, to elicit an aesthetic response from the viewer; some of most relevant article to our work are discussed here. Henry Kang, Seungyong Lee, and Charles K. Chui [3] presented a non-photorealistic rendering technique that automatically delivers a stylized abstraction of a photograph. Their approach is based on shape/color filtering guided by a vector field that describes the flow of salient features in the image. This flow-based filtering significantly improves the abstraction performance in terms of feature enhancement and stylization, their method is also simple, fast, and easy to implement and they verify it by Experimental results to demonstrate the effectiveness of the method in producing stylistic and feature-enhancing illustrations from photographs. Manolis Kamvysselis and Ovidiu Marina "A Cognitive Abstraction Approach to Sketch-Based Image Retrieval" [4], in their thesis is addressing the problem of retrieving from a small database a particular image previously seen by the user. This thesis combines current findings in cognitive science with the knowledge of previous image retrieval systems to present a novel approach to content based image retrieval and indexing. We focus on algorithms which abstract away information from images in the same terms that a viewer abstracts information from an image. The focus in Imagina is on the matching of regions, instead of the matching of global measures. Multiple representations, focusing on shape and color, are used for every region. The matches of individual regions are combined using a saliency metric that accounts for differences in the distributions of metrics. Region matching along with configuration determines the overall match between a query and an image. The paper [1] presents a simple algorithm for producing stylistic abstraction of a photograph. Based on mean curvature flow in conjunction with shock filter, our method simplifies both shapes and colors simultaneously while preserving important features. In particular, we develop a constrained mean curvature flow, which outperforms the original mean curvature flow in conveying the directionality of features and shape boundaries. The proposed algorithm is iterative and incremental, and therefore the level of abstraction is intuitively controlled. Optionally, simple user masking can be incorporated into the algorithm to selectively control the abstraction speed and to protect particular regions. Experimental results show that our method effectively produces highly abstract yet feature-preserving illustrations from photographs.

3. Edge Detection Techniques

Edge detection is a fundamental tool in image processing and computer vision, particularly in the areas of feature detection and feature extraction, which aim at identifying points in a digital image at which the image brightness changes sharply or, more formally, has discontinuities. The same problem of finding discontinuities in 1D signal is known as step detection.

3.1 Sobel Operator

The Sobel operator is used in image processing, particularly within edge detection algorithms. Technically, it is a discrete differentiation operator, computing an approximation of the gradient of the image intensity function. At each point in the image, the result of the Sobel operator is either the corresponding gradient vector or the norm of this vector. The Sobel operator is based on convolving the image with a small, separable, and integer valued filter in horizontal and vertical direction and is therefore relatively inexpensive in terms of computations. On the other hand, the gradient approximation which it produces is relatively crude, in particular for high frequency variations in the image.

Mathematically, the operator uses two 3×3 kernels which are convolved with the original image to calculate approximations of the derivatives - one for horizontal changes, and one for vertical. If we define A as the source image, and G_x and G_y are two images which at each point contain the horizontal and vertical derivative approximations, the computations are as follows:

$$\mathbf{G}_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * \mathbf{A} \quad \text{and} \quad \mathbf{G}_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * \mathbf{A}$$

Where * here denotes the 2-dimensional convolution operation.
The x-coordinate is here defined as increasing in the "right"-direction, and the y-coordinate is defined as increasing in the "down"-direction. At each point in the image, the resulting gradient approximations can be combined to give the gradient magnitude, using:

$$\mathbf{G} = \sqrt{\mathbf{G}_x{}^2 + \mathbf{G}_y{}^2}$$

Using this information, we can also calculate the gradient's direction:

$$\mathbf{\Theta} = \operatorname{atan2}(\mathbf{G}_y, \mathbf{G}_x)$$

Where, for example, θ is 0 for a vertical edge which is darker on the right side.

3.2 Roberts Operator

The Roberts' Cross operator is used in image processing and computer vision for edge detection. It was one of the first edge detectors and was initially proposed by Lawrence Roberts in 1963. As a differential operator, the idea behind the Robert's Cross operator is to approximate the gradient of an image through discrete differentiation which is achieved by computing the sum of the squares of the differences between diagonally adjacent pixels.

In order to perform edge detection with the Roberts operator we first convolve the original image, with the following two kernels:

$$\begin{bmatrix} +1 & 0 \\ 0 & -1 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 0 & +1 \\ -1 & 0 \end{bmatrix}.$$

Let I(x, y) be a point in the original image and $G_x(x, y)$ be a point in an image formed by convolving with the first kernel and $G_y(x, y)$ be a point in an image formed by convolving with the second kernel. The gradient can then be defined as:

$$\nabla I(x,y) = G(x,y) = \sqrt{G_x^2 + G_y^2}.$$

The direction of the gradient can also be defined as follows:

$$\Theta(x,y) = \arctan\left(\frac{G_y(x,y)}{G_x(x,y)}\right).$$

3.3 Canny Operator

The Canny edge detector is an edge detection operator that uses a multi-stage algorithm to detect a wide range of edges in images. It was developed by John F. Canny in 1986. Most importantly, Canny also produced a computational theory of edge detection explaining why the technique works.

The Canny edge detector uses a filter based on the first derivative of a Gaussian, because it is susceptible to noise

International Journal of Scientific Engineering and Technology (ISSN : 2277-1581)
www.ijset.com, Volume No.1, Issue No.2 pg:143-147 01 April 2012

present on raw unprocessed image data, so to begin with, the raw image is convolved with a Gaussian filter. The result is a slightly blurred version of the original which is not affected by a single noisy pixel to any significant degree. Here is an example of a 5x5 Gaussian filter, used to create the image to the right, with σ = 1.4:

$$\mathbf{B} = \frac{1}{159}\begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} * \mathbf{A}.$$

The Canny algorithm uses four filters to detect horizontal, vertical and diagonal edges in the blurred image. The edge detection operator (Roberts, Prewitt, Sobel for example) returns a value for the first derivative in the horizontal direction (G_x) and the vertical direction (G_y). From this the edge gradient and direction can be determined:

$$\mathbf{G} = \sqrt{\mathbf{G}_x{}^2 + \mathbf{G}_y{}^2}$$

$$\mathbf{\Theta} = \arctan\left(\frac{\mathbf{G}_y}{\mathbf{G}_x}\right).$$

The edge direction angle is rounded to one of four angles representing vertical, horizontal and the two diagonals (0, 45, 90 and 135 degrees for example).

4. Neural Network

The term neural network was traditionally used to refer to a network or circuit of biological neurons [5]. The modern usage of the term often refers to artificial neural networks, which are composed of artificial neurons or nodes. Thus the term has two distinct usages:

P_{11}	P_{12}	P_{13}	P_{14}
P_{21}	P_{22}	P_{23}	P_{24}
P_{31}	P_{32}	P_{33}	P_{34}
P_{41}	P_{42}	P_{43}	P_{44}

E_{11}	E_{12}	E_{13}	E_{14}
E_{21}	E_{22}	E_{23}	E_{24}
E_{31}	E_{32}	E_{33}	E_{34}
E_{41}	E_{42}	E_{43}	E_{44}

1. Biological neural networks are made up of real biological neurons that are connected or functionally related in a nervous system. In the field of neuroscience, they are often identified as groups of neurons that perform a specific physiological function in laboratory analysis.
2. Artificial neural networks are composed of interconnecting artificial neurons (programming constructs that mimic the properties of biological neurons). Artificial neural networks may either be used to gain an understanding of biological neural networks, or for solving artificial intelligence problems without necessarily creating a model of a real biological system. The real, biological nervous system is highly complex: artificial neural network algorithms attempt to abstract this complexity and focus on what may hypothetically matter most from an information processing point of view. Good performance (e.g. as measured by good predictive ability, low generalization error), or performance mimicking animal or human error patterns, can then be used as one source of evidence towards supporting the hypothesis that the abstraction really captured something important from the point of view of information processing in the brain. Another incentive for these abstractions is to reduce the amount of computation required to simulate artificial neural networks, so as to allow one to experiment with larger networks and train them on larger data sets.

5. Proposed Algorithm

The proposed algorithm works on detection and enhancing the important information of an image while suppressing the noise generated false information contents this is particularly useful where the original image having possibility of being distorted by noise.

The proposed algorithm can be divided into two parts described in following steps.

Part 1: Training of Neural Network

Step1. Generate the initial noisy patterns of selected size (like 16X16, 4X4 etc.) by Gaussian distribution of different mean (from 0 to 0.8) and variance (0 to 0.8).

Step2. Generate the edge image from above generated image by using 'Canny' operator.

Step3. Divide the original image into non-overlapping blocks of selected size (like 3X3, 5X5, 7X7 etc.).

Figure 1: Example image left with Pixel values P_{xy} and their corresponding Edge values E_{xy} in right Matrix.

Step4. Now train the neural network using image pixels of selected block as training vectors (by converting that block into row vector) and corresponding edge value (0 or 1) as label. For example if in fig1 we take a block of 3X3 and placed it on upper left corner then corresponding edge value will be E_{22}.

Part 2: Generating Abstracted Image

Step1. Generate the smoothed image by filtering the given image using median filter.

Step2. Find out the edges using 'canny' operator.

Step3. Now from the edge image find out the points of edges and extract the pixels around that pixel location from given image according to windows size. Example let the E_{12} be the edge then extracts the pixels from upper left corner by 3X3 windows size here windows size is considered 3X3.

Step4. Now pass this vector to previously trained neural network.

Step5. If the neural network predicts an edge then remove the edge information from edge image else keep it intact.

6. Simulation Results

For better analysis of proposed algorithm it is simulated in MATLAB with different images and results are presented here

Figure 1: shows the original image.

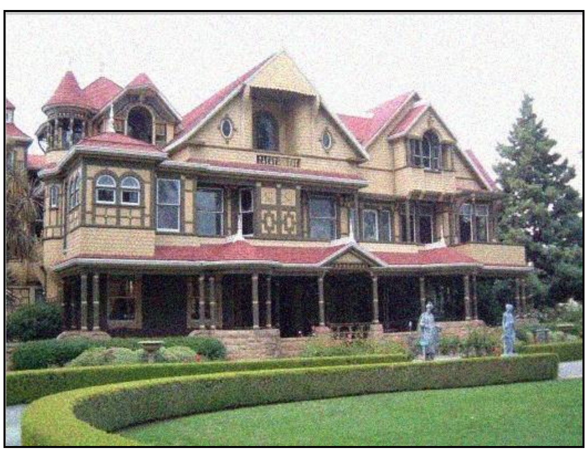

Figure 2: image with noise

Figure 3: edges without neural network.

Figure 4: edges with neural network

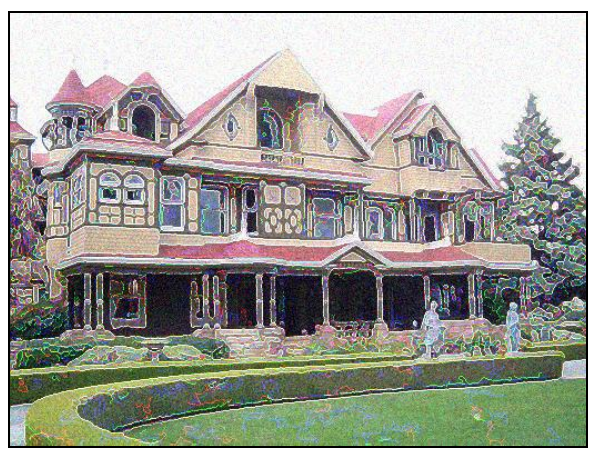

Figure 5: Abstracted Image without neural network

Figure 6: Abstracted Image with neural network

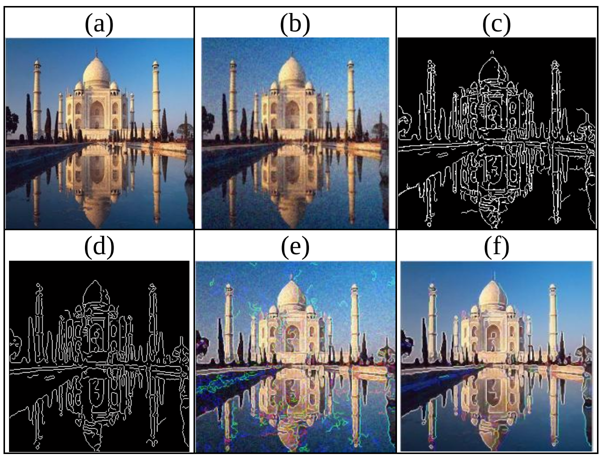

Figure 7: results for Tajmahal image (a) original image (b) image with noise (c) edges without NN (d) edges with NN (e) Abstracted image without NN (f) Abstracted image with NN.

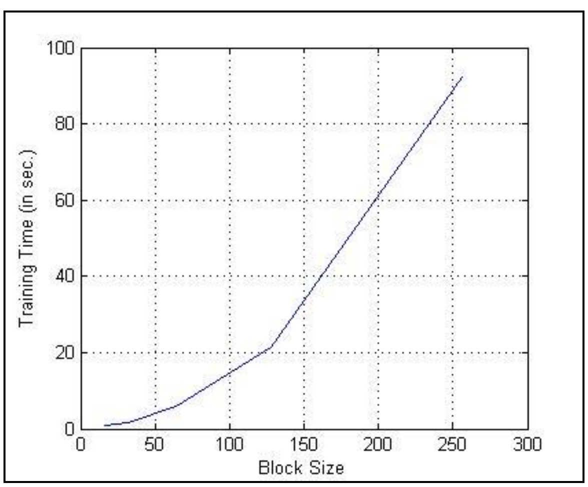

Figure 8: Plot of block size vs. training time

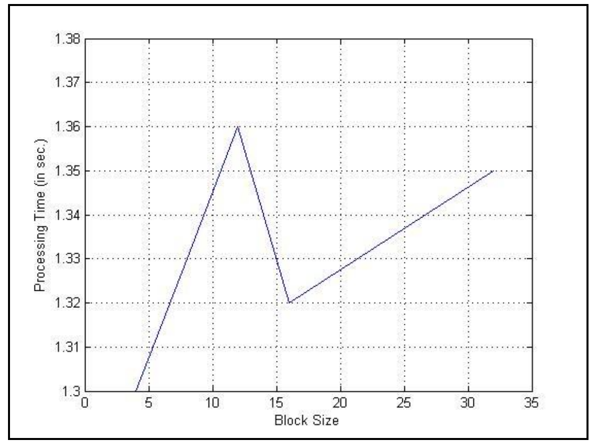

Figure 8: Plot of block size vs. processing time image size is fixed to 256X256

7. Conclusion

After simulating the algorithm with different images and analyzing results it's clear that proposed algorithm sufficiently reduces the effect of noise patterns in the abstracted image, (the figure 3 and 4 shows the clear difference between the edges abstracted by NN & without NN) also the proposed technique takes almost same time for processing even with different block size although the training time grows exponentially hence in future it is needed to minimize.

References

[1] Henry Kang and Seungyong Lee "Shape-simplifying Image Abstraction", Pacific Graphics 2008 Volume 27 (2008), Number 7.

[2] Alexander Kolliopoulos "Image Segmentation for Stylized Non-Photorealistic Rendering and Animation" University of Toronto 2005.

[3] Henry Kang, Seungyong Lee, and Charles K. Chui "Flow-Based Image Abstraction" in IEEE transactions on visualization and computer graphics, vol. 15, no. 1, january/february 2009.

[4] Manolis Kamvysselis and Ovidiu Marina "Imagina: A Cognitive Abstraction Approach to Sketch-Based Image Retrieval" MASSACHUSETTS INSTITUTE OF TECHNOLOGY May 1999.

[5] J. J. HOPFIELD Neural networks and physical systems with emergent collective computational abilities. Proc. NatL Acad. Sci. USA Vol. 79, pp. 2554-2558, April 1982 Biophysics