DEVELOPING HOPFIELD NEURAL NETWORK FOR COLOR IMAGE RECOGNITION

by

KUSSAY NUGAMESH MUTTER

Thesis submitted in fulfillment of the requirements
for the degree of
Doctor of Philosophy

2010

ACKNOWLEDGEMENTS

First and foremost praise and gratitude be to ALLAH (Subhanahu Wa Ta'ala), almighty, for giving me all the patience, courage, and help; without whose gracious help it would have been impossible to accomplish this work.

I would like to thank my supervisors Dr. Mohd Zubir Mat Jafri and Dr. Azlan Abul Aziz for their advice, guidance, help, ideas, insights, encouragement, and invaluable discussions, and suggestions.

Special thanks are due to my colleagues in the engineering lab, School of Physics, Universiti Sains Malaysia and to all my friends for their numerous discussions, help, and encouragement. I want also to thank the School of Physics, IPS, and Hamzah Sendut Library 1 & 2 at Universiti Sains Malaysia for their administrative help.

My sincere thanks go to my parents, family, and my wife for their never-fading love and encouragement.

TABLE OF CONTENTS

CHAPTER 8- CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER WORK

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
ANN	Artificial Neural Network
ASAP	Advanced System Analysis Programming
BMP	Bitmap type of images used by Windows operating system
BS	Beam Splitter
BTC	Block Truncation Coding
CCD	Charge-Coupled Device
CPU	Central Processing Unit
FWM	Four-Wave Mixing
GPS	Global Positioning System
HNN	Hopfield Neural Network
IC	Integrated Circuit
MHNN	The proposed Modified HNN
MZI	Mach-Zehnder Interferometer
NN	Neural Network
OCR	Optical Character Recognition
OLG	Optical Logic Gates
PD	Photo Detector
RAM	Random Access Memory
RGB	Red, Green, and Blue bands

RLD	Run-Length-Decoding
RLE	Encoding
ROI	Region-Of-Interest
SLM	Spatial light Modulator
SOA	Semiconductor Optical Amplifier
UNI	Ultrafast Nonlinear Interferometer
VLSI	Very Large Scale Integrate Circuits
XGM	Cross Gain Modulation
XNOR	Exclusive NOR gate
XOR	Exclusive OR gate
XPM	Cross Phase Modulation

LIST OF SYMBOLS

δ_A, δ_B	Delta function
$\Delta\emptyset$	Phase difference
λ	Electromagnetic wavelength
ν, ω	Electromagnetic frequency
φ	Electromagnetic phase
$\mathit{\Lambda}$	minimum spacing between fringes' planes
T_r	Relative time of processing
V_I^k	The known vector property
V_O^k	The unknown vector property
W', W_{ij}	The weight matrix, the corrected weight matrix
W_{ns}	The weight size
$V_{unknown}$	The unknown output vector
AV_{TLW}	Average of TLW
b	Number of bits required to store a digitized image
c	Speed of light
C_{new}	New image size
C_{old}	Old images size
C_r	correlation
CR	Compression Ratio

d	Number of bits in a pixel
E	Lyapunov energy function
E_e	Electromagnetic energy
E_i	Electromagnetic component of the light
F	Image filter
$g(x,y)$, s	A new output pixel value at x and y coordinates
h	Plank’s constant
$I(r)$	Irradiance at position r
$i(x,y)$	Illumination at x and y coordinates
$I(x,y)$, r	Pixel value in an image at x and y coordinates
net_i , V_{out}	The output vector number i
OPD	Optical Path Difference
P_i	Electromagnetic power
$r(x,y)$	Reflectance at x and y coordinates
T_c	The consumed time of processing
T_i , θ	Threshold of learning Hebb rule
TLW	Total Letter Weight
V	Visibility of light
V_i	Vector (pattern) number i

PEMBANGUNAN RANGKAIAN NEURO HOPFIELD UNTUK PENGECAMAN IMEJ BERWARNA

ABSTRAK

Rangkaian Neuro Hopfield (HNN) adalah suatu rangkaian auto-asosiatif berlelar yang mengandungi lapisan tunggal elemen pemproses bersambung sepenuhnya dan tertumpu di vektor sepadanan terdekat. Rangkaian ini mengubah pola masukan melalui lelaran yang berurutan sehingga vektor terlatih dilepaskan di output. Seterusnya output tidak lagi berubah dengan lelaran berikutnya. HNN menghadapi masalah nyata apabila berurusan dengan imej yang memiliki lebih daripada dua warna, hingar penumpuan, muatan terhad dan latihan perlahan serta penumpuan bilangan vektor dan saiznya. Permasalahan ini dikaji dan diuji dengan penyelesaian yang dicadangkan bagi mendapatkan prestasi optimum HNN dan menetapkan permulaan kajian masa depan. Pe nggunaan saiz vektor yang lebih kecil, iaitu kurang daripada tiga piksel dan pemogaran imej digital kepada satahbit penting sebagai sub-imej bebas digunakan untuk pemprosesan HNN. Selain itu, nilai kestabilan pemberat matriks HNN diyahtempatkan daripada sifar ke bukan-sifar. Ini akan membetulkan ralat yang mungkin muncul di vektor akhir. Namun begitu, pengubahsuaian sebelumnya masih memerlukan pengolahan data yang besar yang dihasilkan daripada memisahan imej berwarna peringkat tinggi ke satahbit. Penggunaan HNN sebagai algoritma kompresor dan Pengekod-Panjang-larian (RLE) akan membantu untuk mengurangkan jumlah data yang disimpan. HNN terubahsuai (MHNN) yang baru ini menyebakan pemprosesan kompleks dan perlahan; oleh itu, Gate Logik Optik menjadi asas yang kukuh untuk mempercepatkan proses MHNN. Untuk menguji kebolehpercayaan MHNN baru yang dicadangkan, tiga pelaksanaan berurutan dicadangkan iaitu imej binari, kelabu dan RGB. Penemuan secara ujikaji menunjukkan bahawa MHNN yang diusulan ini telah berjaya dilaksanakan untuk imej berwarna dengan hingar yang rendah dan penumpuan jelas berbanding dengan HNN tradisional. Akhirnya, teknik yang dicadangkan melalui MHNN ini secara umumnya boleh digunakan untuk setiap imej berwarna dengan penumpuan yang optimum.

DEVELOPING HOPFIELD NEURAL NETWORK FOR COLOR IMAGE RECOGNITION

ABSTRACT

Hopfield Neural Network (HNN) is an iterative auto-associative network which consists of a single layer of fully connected processing elements and converges to the nearest match vector. This network alters the input patterns through successive iterations until a learned vector evolves at the output. Then the output will no longer change with successive iterations. HNN faces real problems when it deals with images of more than two colors, noisy convergence, limited capacity, and slow learning and converging according to the number of vectors and their sizes. These problems were studied and tested the proposed solutions to obtain the optimum performance of HNN and set a starting for future research. Smaller size of vectors of three pixels and dismantling the presented digital image into its essential bitplanes as independent sub-images is used for HNN's processes. In addition, the stability value of HNN's weight matrices can be re-localized from zero to non-zero. This will correct the errors which may appear in the final vector. However, the previous modifications still require processing of large data which are produced from separating high level color images into bitplanes. Using HNN as a compressor algorithm and Run-Length-Encoding will help to reduce the amount of the saved data. The final new Modified HNN (MHNN) has a complex and a slow processing; therefore, the Optical Logic Gates promotes a solid base to speed up MHNN processes. For testing the reliability of the proposed MHNN, three new sequenced implementations are suggested which are binary, gray, and RGB images. The experimental findings show that the new proposed MHNN can successfully work with color images with low noises and clear converging in comparison with the traditional HNN. Finally, the proposed technique of the MHNN can be generalized to be applied for any color images with optimum converging.

CHAPTER 1
INTRODUCTION

1.0 Overview

The last few decades witnessed rapid developments in different disciplines of information technology whereby various techniques have been used to optimize their performance, especially in the field of Artificial Intelligence (AI). Neural Networks as a discipline falls under AI and have attracted researchers because they have wide error tolerance as they deal with a wide range of data. Many Neural Networks basically revolves around pattern recognition such as Optical Character Recognition (OCR), fingerprints, and satellite images (Fausett, 1994; Zurada, 1996; Arbib, 2003; Samarasinghe, 2007).

Information processing technology has always been under investigation throughout the history of science. Recently pictorial information processing has increasingly become important. A large number of pictures that are related to science, government, and industry have been collected for the purpose of pictorial analysis. This has promoted the need for an automatic system to analyze pictures. The availability of digital computers and massive amounts of pictorial data in many fields has recently made picture recognition one of the major topics (Lippmann, 1987; Zurada, 1992; Arbib, 2003).

Hopfield Neural Network (HNN) has been investigated to be adapted according to the problem of the study. However, HNN is still facing real problems when it deals with images of more than two colors, noisy convergence, limited capacity, and slow learning and converging according to the number of vectors and their sizes (Lippmann, 1987; Zurada, 1992).

In the current work, the researcher would like to study these problems and test the proposed solutions to obtain the optimum performance of HNN and set a starting for future research. There is a problem with HNN efficiency when it deals with more than two color images; therefore, the researcher suggests dismantling the present digital image into its essential binary planes (bitplanes) (Umbaugh, 1998; Gonzalez & Woods, 2002; Sonka et al., 2008). Then the researcher will consider each layer of these bitplanes as an independent sub-image to be ready for HNN's processes. In this approach, the researcher may form and implement a general algorithm of HNN to work with color levels with almost the same efficiency as for two color images.

Nevertheless, noisy reconstructions presented through HNN converging operation are still undesired. Therefore, one may re-localize the stability value of HNN's weight matrices from zero to non-zero. This means that the neurons will have self-connection and the system will correct the errors which will appear in the final pattern (vector). In addition, the image may be considered as a set of small vectors with three pixels, which will lead to get high capacity memory, faster learning and converging, and better stability of reconstruction. Even through HNN has these advantages; it still requires processing of large data produced from separating high level color images into bitplanes by compression. Therefore, the researcher adopts two techniques to compress the data. The first one is by using HNN itself as a compressor algorithm, and the second is by hybridizing Run-Length-Encoding (RLE) approach for extra lossless compression. After accomplishing the final proof of the algorithms of this research, Optical Logic Gates (OLG) promotes a solid base to speed up both learning and converging processes of HNN. This chapter will briefly discuss relevant issues such as artificial neural networks, HNN, and optical neural networks.

1.1 Artificial Neural Networks

An Artificial Neural Network (ANN) can be configured for a specific application, such as pattern recognition, data classification, optimization, coding, and control, through a learning process. They possess the ability to solve cumbersome or intractable problems by learning directly from the data. An ANN is an information processing model that is inspired by the biological neurons. A biological neuron consists of dendrite, a cell body, and axon, (see Figure 1.1). The connections between the dendrite and the axon of other neurons are called synapses. Electrical pulses which come from other neurons are translated into chemical information at each synapse. The cell body will input these pieces of information and fires an electric pulse if the sum of the inputs exceeds a certain threshold. The network which consists of these neurons is a neural network which represents the most essential part of our brain activity. In fact, there is a close analogy between the structure of a biological neuron (i.e., a brain or nerve cell) and the processing neuron. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons which can be applied to ANNs as well (or artificial neuron) (Rao and Rao, 1993; Fausett, 1994; Zurada, 1996; Ruan, 1997; Arbib, 2003; Samarasinghe, 2007).

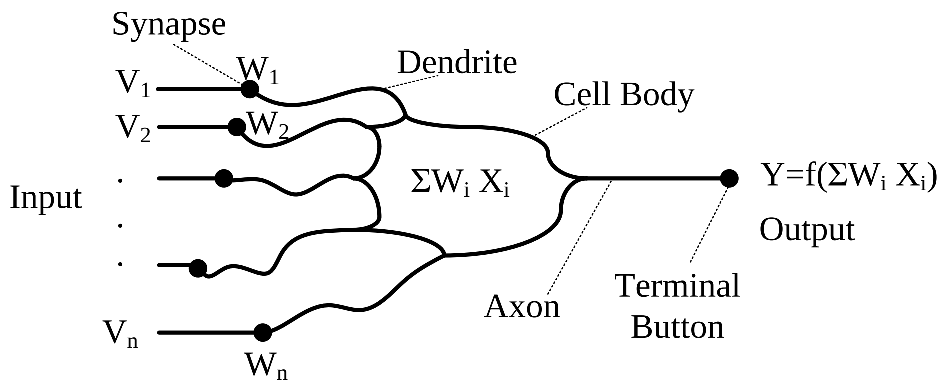

Figure 1.1: Biological Neuron. V_n is an input signal, W_n is a weight, and Y is an activation function

An artificial neuron model simulates multiple inputs and one output, the switching (activation) function of input-output relation, and the adaptive synaptic weights, (see Figure 1.2). It is composed of a large number of highly interconnected processing neurons working in unison to solve specific problems.

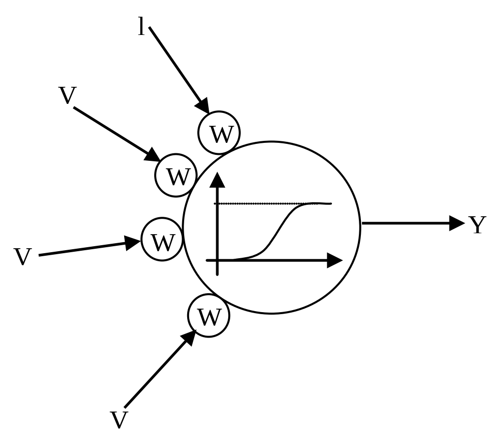

Figure 1.2: An artificial neuron model [Ruan, 1997]

According to Fausett (1994) and Zurada (1996), ANNs, which have been developed as generalization derived from mathematical models of biological neuron, are based on the following considerations:

1. Information processing occurs at many neurons.
2. Signals are passed between neurons over connection links.
3. Each connection link has an associated weight which multiplies the signal transmitted in a typical neural net.
4. Each neuron applies an activation function to its net input (sum of the weighted input signals) to determine its output signal.

The future of neural networks lies in the development of hardware. Efficient neural networks depend on hardware specified for its eventual use. Due to the limitations of processors, neural networks take longer time to learn. Therefore, many studies were interested in developing optical neural networks since the 80 th's.

1.1.1 Architecture

ANNs can be classified into single-layer and multi-layer. A single-layer net has one layer of connection weights. Often, the units can be distinguished as input units, which receive signals from the outside world and output units from which the response of the net can be read. In the typical single-layer net shown in Figure 1.3, the input units are fully connected to other units. HNN architecture is an example of a single-layer net in which all units function as both input and output units (Samarasinghe, 2007).

A multi-layer net is a net with one or more layers of hidden units between the input units and the output units. Typically, there is a layer of weights between the two adjacent levels of units (input, hidden, or output). Multi-layer nets can solve more complicated problems than single-layer nets can, yet learning in this case will be more difficult as shown in Figure 1.4 (2007).

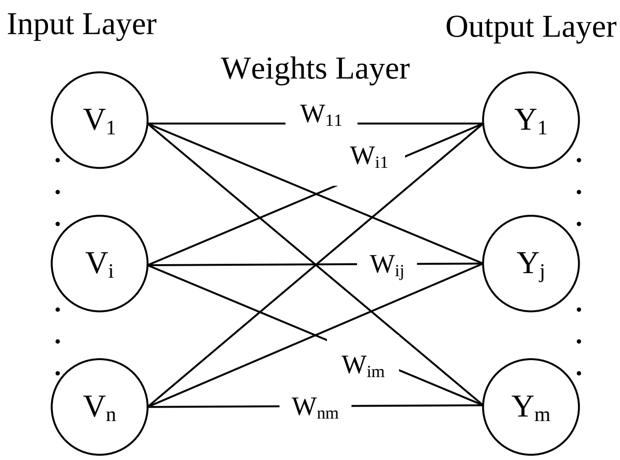

Figure 1.3: A singl-layer neural net

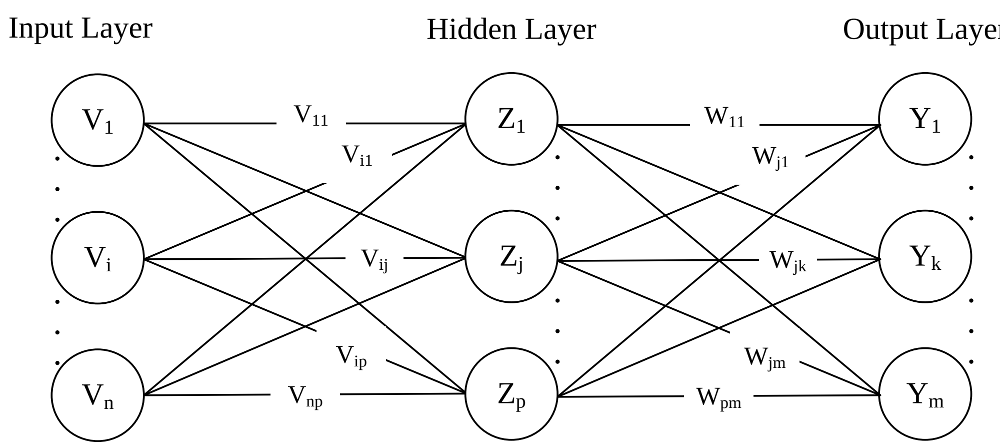

Figure 1.4: A multi-layer neural net

According to Zurada (1996), Arbib (2003), and Samarasinghe (2007), generally, a neural network is characterized by:

1. Pattern of connections among the neurons (called architecture)

2. Method of determining the weights on the connections (called training, or learning algorithm)
3. Activation function (threshold function) and recalling (converging)

An associative memory is a simple single-layer neural network that can learn a set of pattern pairs (or associations). An efficient associative memory can store a large set of patterns as memories. During recall, the memory is excited with a key pattern which contains a portion of information about a particular member of a stored pattern set. This particular stored prototype can be recalled through association of the key pattern and the memorized information. In other words, associative memories provide one way to the computer-engineering problem of storing and retrieving data based on content rather than on storage address. The information is stored in a neural net and is distributed throughout the system (in the net's weights). Before training an associative memory of a neural net, the original patterns must be converted to an appropriate representation for computation. For example, the original pattern might consist of on and off signals, and the conversion can be on→ +1, off→ 0 (binary representation) or on→ +1, off→ -1 (bipolar representation) (Arbib, 2003; Samarasinghe, 2007).

Thus, the purpose of an auto associative memory is to store and retrieve an individual pattern where the pattern which is applied to retrieve may be incomplete or distorted by noise. In this case the input and output vectors are identical and called as auto-associative memory, while they are called hetero-associative memory when they are different (Rao and Rao, 1993; Kinnebrock, 1995; Arbib, 2003; Samarasinghe, 2007).

1.1.2 Learning

Like people, ANNs learn through given examples. The method of setting the values of the weights (learning) is an important distinguished characteristic of different neural nets. Generally, there are two types of learning. The first type is the supervised training, in which the training is accomplished by presenting a sequence of training vectors or patterns each with an associated target output vector. The weights are then adjusted according to a learning algorithm. The second type is the unsupervised training which is self-organizing neural nets. It performs grouping to similar input vectors together without the use of training data to specify what a typical member of each group looks like. An ANN usually consists of a large number of simple processing units, called neurons, via mutual interconnections. It learns to solve problems by adequately adjusting the strength of the interconnections according to the input data. Moreover, the neural network is adapted easily to new environments by learning; and it can deal with information that is noisy, inconsistent, vague, or probable. These features have motivated extensive research and developments in ANNs.

1.1.3 Threshold Functions

The threshold function is used to determine the output of a neuron in the output layer. The value obtained from the threshold function characterizes the neuron to fire or not. Common activation functions are shown in Figure 1.5 (Rao & Rao, 1993; Fausett, 1994; Samarasinghe, 2007).

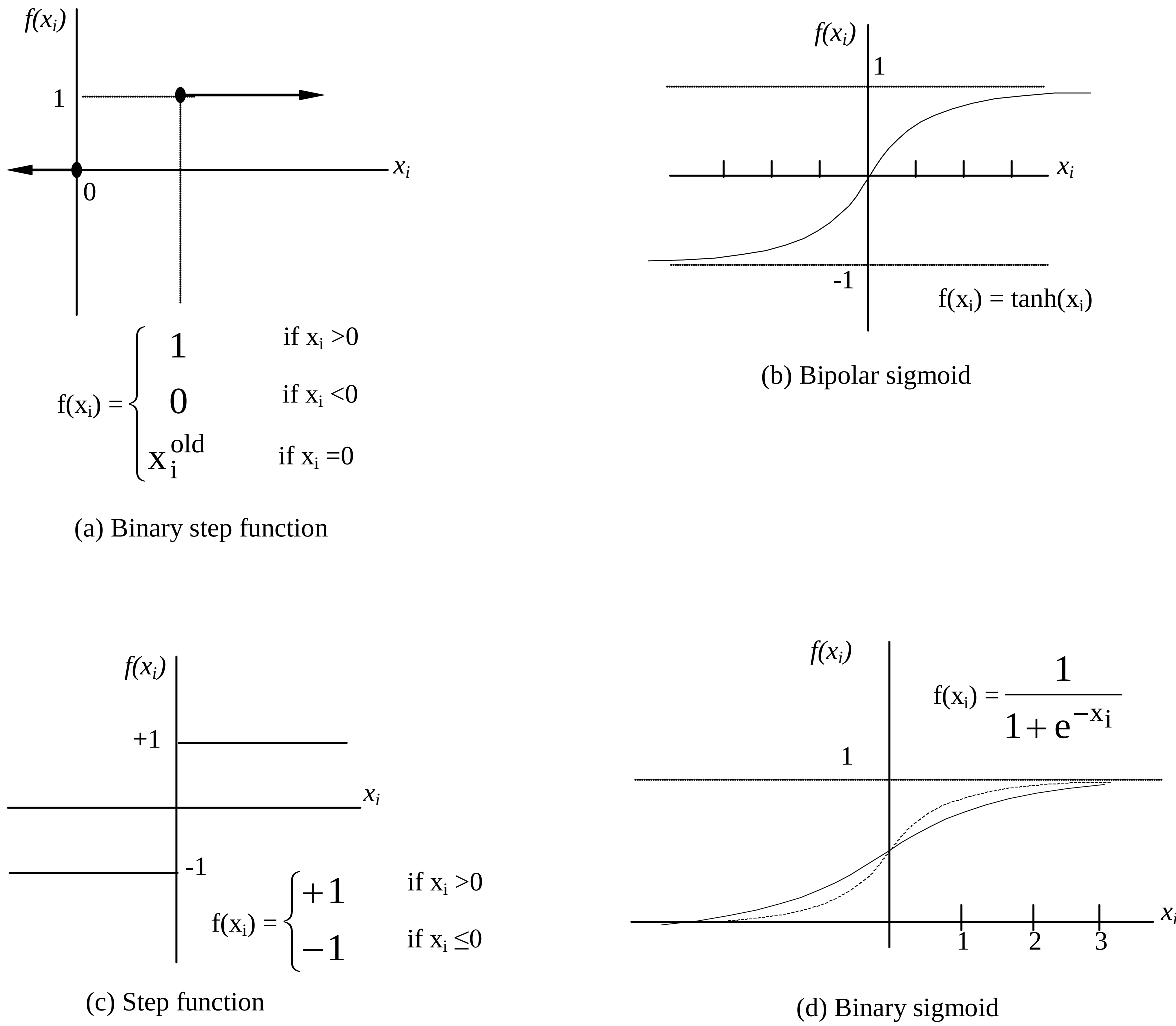

Figure 1.5: A common activation functions

1.1.4 Characteristics of Neural Networks

Neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. Neural networks take a different approach to problem solving than that of conventional computers.

Conventional computers use an algorithmic approach, i.e., the computer follows a set of instructions in order to solve a problem. This restricts problem solving capability of the conventional computers to the problems that already understand and know how to solve.

Because neural networks learn via examples, they cannot be programmed to perform a specific task. The examples must be selected carefully; otherwise there will be a lot of wasted time and also the network may function incorrectly. The disadvantage is that the network operation can be unpredictable. Neural networks and conventional algorithmic computers are not in competition, but they complement each other. There are tasks that are more suitable to an algorithmic approach like arithmetic operations and tasks that are more suitable to neural networks. Furthermore, a large number of tasks requires systems that combine two approaches in order to perform at maximum efficiency. Other advantages include:

1. Adaptive learning: An ability to learn how to do tasks based on the data given for training or initial experience.
2. Self-Organisation: An ANN can create its own organisation or representation of the information which is received during learning time.
3. Real Time Operation: ANN computations may be carried out in parallel and special hardware devices are designed and manufactured which take advantage of this capability.
4. Fault tolerance via redundant information coding: Partial destruction of a network leads to the corresponding degradation of performance. However, some network capabilities may be retained even with major network damage.

1.2 Hopfield Neural Network

A large number of neural network paradigms have been developed and used in the last four decades. One of these widely used paradigms is the HNN. This type of network was described by J.J. Hopfield (1982). He is a physicist and was working on the magnetic behavior of solids (spin glasses). This is essentially determined by the Ising-spin, a property of magnetic atoms, and is described by two states (+1 and –1). The interesting part is the magnetic mutual exchange between the atoms which can be described by a mathematical formula; and this has led to create HNN (Arbib, 2003; Samarasinghe, 2007). HNN is considered as a network which converges to the nearest match pattern. This network alters the input patterns through successive iterations until a learned pattern evolves at the output and the output will not change with successive iterations. However, HNN can be used for constrained optimization problems, such nets are called fixed-weight nets (Fausett, 1994; Jain & Jain, 1997; Leondes, 1998).

HNNs are iterative auto-associative networks which consist of a single layer of fully connected processing elements, which can be categorized as an associative memory. An expanded form of a common representation of HNN is shown in Figure 1.6. All the processing neurons are connected through a feedback architecture along with connection weights. HNNs are fully interconnected; that is to say, each unit (neuron) is connected to other units, and has feedback connections among the units (Fausett, 1994; Zurada, 1996; Arbib, 2003).

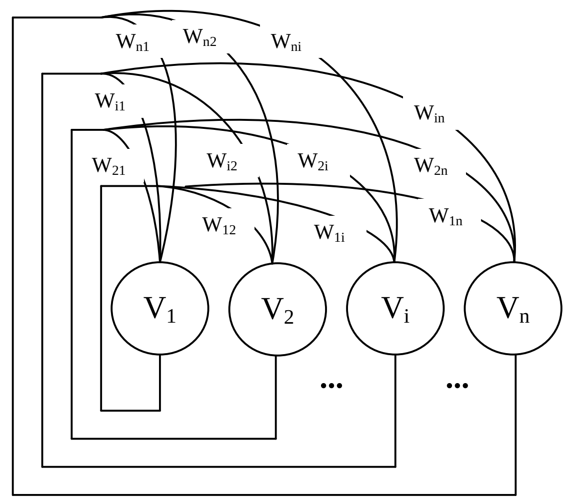

Figure 1.6: Hopfield Neural Network Architecture, [Fausett, 1994]

1.3 Shortcomings of Hopfield Neural Network

HNN basically revolves around the notion of pattern recognition; that is the ability to recognize correctly unclear (noisy) pictures (Kinnebrock, 1995). However, because HNN is an associative memory network, the net can save all the trained patterns in its memory and make convergence if the input pattern is correlated with one of these patterns. On the other hand, there are many limitations associated with HNN that can be listed down as follows (Fausett, 1994; Zurada, 1996; Lippmann, 1987; Rao & Rao, 1993; Arbib, 2003):

1. The number of patterns that can be stored and accurately recalled is almost limited. If too many patterns are stored, the net may converge to a novel spurious pattern different from all the stored patterns.
2. It is desirable during the converging process to reach a global minimum rather than settling down at a local minimum. Figure (1.7) clarifies the distribution

between a local minimum and a global minimum. In this figure, one can see the graph of an energy function and two points, A and C. These points show that the energy levels are smaller than the energy levels at any point in the vicinity, which means, they represent the points of minimum energy. The overall or global minimum is at point C where the energy level is smaller than that even at point A. This means that A only corresponds to a local minimum. It is supposed to get B and not to stop at A. Yet, when point C is reached, further movement will be toward B and not toward A. Similarly, if a point near A is reached, the subsequent movement should avoid reaching or settling at A and proceed to B.

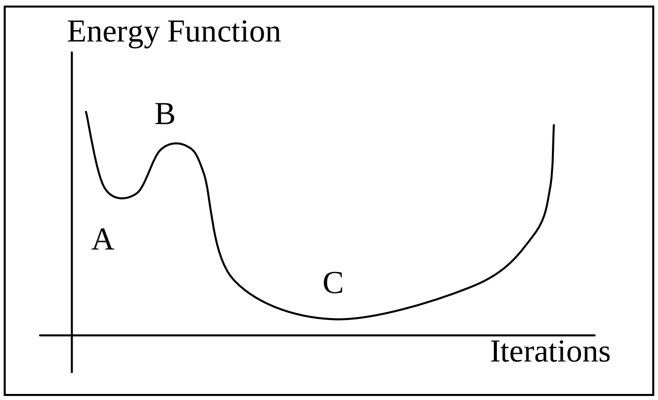

Figure 1.7: Local and global minima

3. The stored pattern will be unstable if it shares many bits with other stored patterns. Here is a pattern which is considered unstable if it is applied at time zero and the net converges to some other patterns. Hence, all the saved patterns must be orthogonal with each other.
4. HNN cannot retrieve the stored pattern when it enters the network with shifting, scaling, or rotating.

5. The weight size is an important factor that determines the number of iterations that needed to finish successive repetitions. The large size of weight is also affect the net efficiency to create true patterns and have consuming time.
6. It is known that HNN deals with only two states of bipolar (or binary) which make the net useless when color images are presented.
7. One more limitation of HNN is the large number of patterns which force the net to be slow with consuming time.

1.4 Optical Neural Networks

Electronic implementations of neural networks are limited by the number of weighted connections between neurons that can be practically achieved. Once the problem under investigation gets more complex, the number of neurons and the geometric fashion will also increase. Optical networks offer an inherent parallelism which allows a high capacity connection between neurons in different layers of network (McAulay, 1991; Alvaro; 1998; Fernandes, 1995; Von, 2004).

The most basic and very important approach to design an ANN is via simulations using software programs which can be run on conventional digital computers. This powerful tool provides valuable information when developing new and innovative neural computer designs. Electronic implementation of neural networks includes systems that are assembled using conventional computer chips and standard passive and active electrical components. A clear disadvantage of software implementation is the slowdown of operation speed. This is due to the fact that the microprocessor of a computer, on which the neural network program runs calculates the neuron states one by one. Considerable acceleration can be achieved by implementing neural networks in dedicated and parallel hardware (Fernandes, 1995; Von, 2004).

When implementing a neural network on a chip the number of connections between the neurons will be physically limited. This limitation is essentially due to the fact that connections cannot cross each other within the same layer on a chip, and consequently they need to be separated in one dimension. Because the number of layers on a chip is limited, the number of crossing connections is also limited (Fernandes, 1999; Von, 2004).

Optics can help to solve this connectivity problem in two ways. Because photons only interact with matter and not with each other, the light beams which are emitted from the connections in an optical neural network can cross each other without problems. Furthermore, the rays of light do not need to be guided in free space. As a consequence no predefined paths or wires are necessary in an optical neural network. This means that the three dimensions in space can be used without soldering. For these reasons, there have been many studies on optical neural networks since the mid-1980's. Nevertheless, since the recognized pattern is often received optically, it is perhaps most natural and straightforward to recognize an optical pattern by using optics (Yu & Jutamulia, 1998; Mos, 1999; Fernandes, 1995; Von, 2004; Borundiya, 2008).

1.4.1 Advantages of Optical Computing

The potential advantage of optics is the capability of high-speed parallel transmission and processing of structured data. It appears that investigation directed toward the development of digital optical computers is one of the most promising and ambitious areas in optical computing (Athale, 1990; McAulay, 1991; Boffi et al., 2003).

Optics is rapidly developing as it has several advantageous features that listed below which can overcome the limitations in the current electronic computers (Athale, 1990; Senior, 1992; Boone, 1998; Boffi et al., 2003; Von, 2004).

1. One advantage of optics over electronics is the higher bandwidth that enables more information to be carried. This is because electronic communication via wires requires charging capacitors that depends on length. In contrast, optical signals in optical fibers, optical integrated circuits and free space do not have to charge a capacitor and are, therefore, faster. Faster transmission with optics is important because transmission time between units is often the limiting factor for the performance of high speed machines. Faster transmission allows faster computational elements to be processed.
2. Very high speed machines use additional power to provide speed and have elements located close to one another to limit transmission time. In this case, the technology of transferring heat out of the system is a limiting factor. Faster optical links allow the computer elements to be spread farther which, in turn, relieve the difficulty and cost heat transfer.
3. Photons are uncharged and do not interfere with one another as readily as electrons. Consequently, light beams may pass through one another without distorting the information carried. This suggests that optical memory is capable

of avoiding the difficulties of memory contention. Moreover, loops of connections are difficult to avoid in massively parallel systems. In the case of electronics, loops generate noise voltage spikes whenever the electromagnetic fields through the loop change. Further, high frequency or fast switching pulses can cause interference in neighboring wires. Signals in adjacent fibers or in optical integrated channels do not interfere with each other nor do they pick up noise due to loops.

4. Images or arrays of pixels may be handled in parallel. Thus, it is conceivable to process a million elements or more in parallel by formulating a problem as a sequence of steps on a 2-D array.

1.4.2 Optical Logic Gates

The fundamental building block of modern electronic computers is the transistor. To replace electronic components with optical ones, an equivalent optical transistor or optical logic gate is required. This is achieved by using materials with a linear and/or non-linear refractive index. The optical transistor can be used to create logic gates, which in turn are assembled into the higher level components of the computer's CPU. The optical logic gate controls one light beam with another. It is (on) when the device transmits light, and (off) when it blocks the light. Optical bistable devices and logic gates such as these are the equivalent of electronic transistors are operate as very high speed on-off switches and are also useful as optical cells for information storage. Optical logic is the use of photons light in logic gates (NOT, AND, OR, NAND, NOR, XOR, and XNOR). However, switching is obtained by using nonlinear optical effects when two or more signals are combined. In addition, using interferometers to perform optical logic gates is widely used by researchers especially for Mach-Zehnder interferometer (Athale, 1990; Boone, 1998; Lee et al., 2002; Von, 2004).

It is possibly true that most attempts at optical logic devices have fundamentally been defective in their real potential usefulness in digital systems. The main reason of using digital systems is that they are essentially infinitely extensible; they must be cascadable (the output of device can drive the input of the next). They must be capable of at least simple complete logic functionality; and they must have fan-out (the output of one device must be able to drive the inputs of at least two subsequent gates). To make large systems, any critical biasing requirements for individual gates should be avoided. That is, it is not allowed to adjust each gate individually. They must also have good input/output isolation. Moreover, the devices must run sufficiently fast and with sufficiently low energy (McAulay, 1991; Boone, 1998; Lee et al., 2002; Von, 2004).

1.5 Statement of the Problem

The shortcomings of HNN, which were mentioned previously, could be real obstacles that may lead to the failure of meeting certain standard requirements of this model. The problem of using HNN with color images for pattern recognition is the main scope of this research. Thus, HNN cannot function with images that have more than 1-bit level (two colors). This will restrict the implementation of the net with a wide range of applications. Therefore, most images' information will be lost while preparing and adapting such images for HNN. Normally the preparing is a preprocessing stage for image binarization.

Moreover, the crosstalk effect depends on the relationship among the learned patterns. The high rate of crosstalk is caused from a less orthogonality among patterns and vice versa. This will conduct the net to produce noises on the output patterns. Further, HNN have problems of limited capacity, and slow learning and converging according to the number of vectors and their sizes.

1.6 Objectives

This research will investigate the optimum improvements that make it possible to use HNN with color images. These improvements are:

1. To study the ability of simplifying any image of any color into a fundamental structure that enable HNN to achieve all processes efficiently.
2. It is reliable to perform segmentation on input images to get an array of small patterns rather than the whole image.
3. Further, it is convenience to test and analyze the likelihood of changing the diagonal of weights' matrices to be non-zero diagonal to reduce the noises at the output patterns.
4. The compressing of weights that produced by HNN will be tested by using HNN indexing and Run-Length-Encoding.
5. A design of an array of optical logic gates can be useful to suggest optical processors that execute HNN operations in a very fast performance. The aim in this case, is to design an optical implementation of the proposed

HNN technique based on array of XNOR gates working with cascaded Mach-Zehnder Interferometer as an optical gate.

6. Throughout the research, the researcher may prove the efficiency of each proposal by providing some examples such as Arabic OCR, Fingerprint recognition, and Satellite image of water pollution detection. In addition, the researcher will compare the results obtained with the previous work in terms of capacity, and clear converging.

1.7 Thesis Organization

An overview of ANN, HNN, limitations of HNN, and optical neural network has been discussed in this chapter. Chapter Two reviews the literature related to the use of HNN with color images and covers the modifications made on this network in terms of vector size, network capacity, and self-connection architecture. In addition, Run Length Encoding algorithm of compression and Optical Logic Gates are also discussed from the previous studies' point of view to overcome the weaknesses that may arise in HNN for better optimization.

The essential bases and background of theoretical aspects will be discussed in Chapter Three. The chapter is divided into three main parts. The first part is a concentrated material on digital images acquiring and representation which are considered the main samples of testing the reliability of the proposed system in this work. The second part is a detailed explanation about HNN. Optical HNN, optical logic gates, and a mathematical representation of Mach-Zehnder Interferometer will be demonstrated in the third part. In Chapter Four, the design of developing HNN to process color images will be examined. The researcher will present the problem of HNN with color images because the information is highly reduced when the images are converted into black and white. Furthermore, a solution for using HNN with high color images will be presented by dealing with the depth of the pixel in these images.

The experimental results are devoted to prove that the modified HNN is able to work at any level of color of images, and this will be discussed in Chapter Five, Chapter Six, and Chapter Seven.

Chapter Five is for implementing the modified HNN for binary images of printed Arabic OCR, while Chapter Six is for fingerprint gray image identification. Chapter Seven, on the other hand, is about applying the proposed technique for monitoring the polluted water in Penang strait in Malaysia. The sequence of these chapters are arranged gradually where each sample of images indicates a level of image color starting from 1-bit images of Arabic letters, 8-bit gray images of fingerprints, and RGB color images of satellite. Conclusions, future work, and recommendations to improve the final results are illustrated in Chapter Eight.

CHAPTER 2
LITERATURE REVIEW

2.0 Introduction

This chapter reviews the literature related to the use of Hopfield Neural Network (HNN) with color images. Besides, the literature review will cover the modifications made on this network in terms of vector size, network capacity, and self-connection architecture. As mentioned in Chapter One, the current study aims to develop HNN for color images by making some modifications in the architecture. Even though, these modifications may lead to optimal performance, two main weaknesses may arise. The first one is producing a large amount of weights for each learning process, and the second one is the consuming time of the processing. Hence, Run Length Encoding (RLE) technique of compression and Optical Logic Gates (OLGs) are studied to overcome these weaknesses for better optimization.

These topics provide the background to the current study to highlight the development of HNN. The proposed technique will be tested to the following applications: 1) Arabic character pattern recognition, 2) Fingerprint identification, and 3) Water pollution detection in Chapter Five, Chapter Six, and Chapter Seven respectively. Therefore, the literature review will also cover the use of HNN wth these applications.

2.1 The Traditional Hopfield Neural Network

As mentioned in Chapter One, HNN is considered as an associative memory, a single layer network, and an interconnected network which have a symmetric synaptic weight. This net has been introduced for the first time by J.J. Hopfield in 1982. He showed that the energy, defined by both the state of a neuron and the synaptic weight, recurrently decreases at every state change and reaches at a minimum point. The network converges almost the entire pattern from a part of it when the state of the network is at a minimum energy which represents the desired pattern. Since then, HNN has attracted some researchers to modify and use it in three general scopes by developing the architecture to optimize the performance. They used it for pattern recognition, hardware realization and implementation (Hopfield, 1984; Hopfield & Tank, 1985; Sudo, Sato, & Hasegawa, 2009; Wena, Lan, & Shih, 2009).

Hopfield neural network has many interesting features and applications in two ways, digital and analog. HNN can also be used either as an associative memory or to solve optimization problems through the use of energy function minimization. Although, HNN has extensively been studied, there is still scope for research. In some applications, it may be used in conjunction with other neural network models (Sulehria & Zhang, 2007).

However, the current work differs from the traditional HNN in several ways as it will be discussed in the next sections. The proposed modifications will focus concentrated on vector size, HNN capacity, self-connection architecture, and HNN for color images.

2.1.1 Vector Size and HNN Capacity

The capacity of HNN memory refers to the ability of storing an amount of information and converging the required information with a minimum cross-talking. Basically, the information capacity depends on the pattern size (Abu-Mostafa et al., 1985; Lowe, 1999). Within this sense, many trends have been suggested to improve HNN capacity of memory (Hopfield & Tank, 1985; Abu-Mostafa et al., 1985; McEliece et al., 1987; Kuh & Dickinson, 1989). It was shown that the total information stored in HNN model is a constant times the number of connections in the network, independent of a particular model, the order of the model, or whether clipped weights are used or not (Keeler, 1987).

According to Kuo and Zhang (1994), the capacity drops to zero when the stored vectors and probe vectors have non-uniform distributions. Therefore, it is necessary to explore the effect of these distributions on the capacity and to improve the capacity of the associative memory. As an alternative of Hopfield associative memory, they introduced a multi order polynomial approximation of the Projection Rule and proved that its storage capacity is higher than that of Hopfield associative memory with the same implementation complexity. Thus far, the researcher has noticed that these two scholars worked on another neural network that is different from HNN. On the other hand, some other researchers have updated all or part of HNN architecture. For instance, Liwanag (1997) proposed an approach to improve the capacity of simple Hebbian pattern associators by adding hidden units. The proposed algorithm was structured to choose potential targets for the hidden layer. The latter helped to protect the network from cross talking effects when the memory was overloaded. While, Lowe (1999) suggested a form of HNN with N neurons that can store (N log N) biased patterns. The quantity increases when the bias of the patterns increases; but, it decreases when the bias gets larges.

Moreover, Sulehria and Zhang (2008) made some improvements in the capacity of HNN and in the methods to achieve higher capacity. Their improvements were based on the best possible percentage of patterns per number of neurons needed.

Recently, Sudo, Sato, and Hasegawa (2009) proposed an associative memory that satisfies the requirements of the memory size that adaptively increases with learning patterns. Their proposal is neither redundant nor insufficient with regard to memory size even in an environment which has available maximum number of associative pairs to that is unknown before learning.

In conclusion, the current study uses a different approach from these scholar's works. The approach depends on utilizing minimum size of vectors (patterns) of three elements with the feature of converging the right vectors. Hence, a vector size is considered as a small block with only eight states that can be found in any binary image.

2.1.2 Self-Connection in HNN

Originally, the discrete-valued neural network architecture proposed by Hopfield (1982) requires zero-diagonal elements in the weight matrix (non-self connection) in which all the neurons connect with each other but not with themselves so that the net changes to a local minimum of energy function. In the case of non-zero-diagonal elements of the weight matrix, it was found in the literature that the self-connection can change the stability of the minimum energy into a point depending on the new values of