A Sensory System for Robots

Using

Evolutionary Artificial Neural Networks

Ann Reddipogu

A thesis submitted in partial fulfilment of the requirements of

The Robert Gordon University

for the degree of Doctor of Philosophy

August 2006

DECLARATION

I hereby declare that this thesis is a record of work undertaken by myself. That it has not been the subject of any previous application for a degree and that all sources of information have been duly acknowledged.

Ann Reddipogu

2006

To

My Parents

Dr. Deva Prasad Reddipogu

Dr. Mary Jones Saganti

ACKNOWLEDGEMENTS

Embarking on this project is one of the best things that happened in my career. Although, the scope of the project was very broad and the solution to the problem was as inconspicuous as a blind man looking for a black cat in a dark room that may not exist, I thoroughly enjoyed the search for the solution. This rewarding project would not have been possible and fulfilling without the help of a few people.

First of all, I would like to extend my gratitude to the staff at the RGU Library, who relentlessly provided me with every publication that I requested.

I thank Mr. Grant Maxwell (Director of Studies) for his support and gentle persuasion during the duration of the research project. I am extremely grateful to Prof. Jörg-Peter Ewert for his compliments regarding the accomplishments of the research project and his continued collaboration.

I am indebted to Prof. Norman Deans for his advice and finding time to read the thesis. I shall be ever grateful for all the inspiring little chats we had, despite his busy schedule. I express thanks to Mr. Ken Gow for his friendship and confidence in me.

Without the encouragement from Dr. Prathap Saganty, I could not have completed this research. I also thank Dr. Prem Saganti for his motivation. I extend appreciation to my parents, sister Eleanor, nephew Avinash and niece Sareena for reminding me (almost everyday) that I had a PhD project to complete! Words cannot express my thanks for John’s undying faith in his mummy and the frequent sacrifices he has had to make through the journey of the project.

The real spur behind the success of the project was Dr. Christopher MacLeod and our explosive discussions.

ABSTRACT

The thesis presents the research involved with developing an Intelligent Vision System for an animat that can analyse a visual scene in uncontrolled environments. Inspiration was drawn both from Biological Visual Systems and Artificial Image Recognition Systems. Several Biological Systems including the Insect, Toad and Human Visual Systems were studied alongside popular Pattern Recognition Systems such as fully connected Feedforward Networks, Modular Neural Networks and the Neocognitron.

The developed system, called the Distributed Neural Network (DNN) was based on the sensory-motor connections in the common toad, *Bufo Bufo*. The sparsely connected network architecture has features of modularity enhanced by the presence of lateral inhibitory connections. It was implemented using Evolutionary Artificial Neural Networks (EANN).

A novel method called 'FUSION' was used to train the DNN, which is an amalgamation of several concepts of learning in Artificial Neural Networks such as Unsupervised Learning, Supervised Learning, Reinforcement Learning, Competitive Learning, Self-organisation and Fuzzy Logic.

The DNN has unique feature detecting capabilities. When the DNN was tested using images that comprised of combination of features used in the training set, the DNN was successful in recognising individual features. The combinations of

features were never used in the training set. This is a unique feature of the DNN trained using Fusion that cannot be matched by any other popular ANN architecture or training method. The system proved to be robust in dealing with New and Noisy Images.

The unique features of the DNN make the network suitable for applications in robotics such as obstacle avoidance and terrain recognition, where the environment is unpredictable. The network can also be used in the field of Medical Imaging, Biometrics (Face and Finger Print Recognition) and Quality Inspection in the Food Processing Industry and applications in other uncontrolled environments.

CONTENTS

CHAPTER 1: INTRODUCTION

1.1 MOTIVATION

Animats are animal-like robots. They are currently a topic of intensive study within the Artificial Intelligence research community and several attempts have been made to make them intelligent enough to behave autonomously. Such systems may, in the future, be very useful in unconstrained or uncertain environments to fulfil various tasks. Advanced animats might find applications, for example, in sub-sea exploration, search and rescue or toxic chemical industries.

MacLeod and Maxwell proposed a framework for an Artificial Nervous System (ANS) based on the Biological Nervous System [1]. The idea of the ANS is to provide a modular, flexible and general-purpose control system for animats. The ANS consists of a hierarchy of systems, from simple control of individual actuators at the lowest levels, to complex reasoning tasks at the higher levels. The model suggested has dedicated modules for specific tasks such as reflex control, action control and object recognition. McMinn modified the model and introduced a new layer to perform the task of sequencing actions, which was absent in the original model [2]. A detailed explanation of the ANS is given in Chapter 2.

The research, which is reported in this thesis, involved the development of a vision system for the ANS model explained above. To implement such a system, a study of Artificial and Biological Visual Systems was undertaken. This included Vision Networks in higher and lower animals.

The aim of the research was to find a system, which produced the best compromise between functionality and complexity. After extensive research, which included experiments using standard Back Propagation (BP) and Advanced Neural Networks, such as Voter Systems and Modular Neural Networks (MNNs), a working system was proposed, developed and evaluated. The system that has proved to be the best compromise is a biologically inspired model based on the toad's visual system.

Much research has been done by Neuroethologists on the behavioural responses of toads and frogs to the presence of prey and predators. Research has shed light on the different classes of neurons present in the toad's brain that have particular functions with respect to recognition of objects in the environment and the generation of appropriate behaviours [3].

An artificial model of the toad's visual system was implemented using a combination of three biologically inspired techniques, namely, Artificial Neural Networks (ANNs), Evolutionary Algorithms (EAs) and Reinforcement Learning (RL). EAs are sometimes used to evolve ANNs and the field of study is now more popularly known as Evolutionary Artificial Neural Networks (EANNs) [4]. Evolutionary Algorithms for Reinforcement Learning (EARL) are being used as a new training method in ANNs [5].

1.2 AIMS AND OBJECTIVES

The overall aim of the research was to develop a modular sensory network for the ANS model mentioned in section 1.1. The following objectives were set out to accomplish the implementation of a biologically inspired vision system for an animat using ANNs and EAs.

Literature Survey:

A literature search in the fields of Vision System and Artificial Intelligence was conducted to understand the foundations on which the research is based. Insights into the various vision architectures and learning methods currently used were sought.

Study of Biological Sensory Systems:

The vision systems of humans and lower animals such as insects and frogs were studied. Considerable knowledge was gained from the methods used by these animals in acquiring, processing and interpreting the sensory information into meaningful data.

Investigation of Neural Network based Pattern Recognition Techniques:

The subjects of Vision and Neural Networks are of interest to scientists from diverse backgrounds – Neurobiologists, Computer Scientists, Engineers and Psychologists. Contributions from a number of research fields were examined and the findings incorporated into the current research programme.

Implementation of a Vision System using Parallel Modular Networks:

It was essential to understand the limitations of standard BP networks and other techniques currently being used in vision systems, in order that a better model could be developed. To achieve this, experiments were undertaken on several network architectures including MNNs, which incorporated a priority system to select between the outputs of modules of the network that were trained for different tasks.

Comparison of results against published work:

Modular Neural Networks (MNNs) were developed mainly to improve network efficiency by splitting a large task into smaller tasks and to use sub-networks called modules to process the information. In these systems, the outputs from sub-networks contribute towards the analysis of the larger task at hand. Other MNNs use a gating network to prioritise between the outputs of sub-networks, which analyse completely different tasks. Interesting results were obtained and compared with published work.

Implementation of an Evolutionary Vision System:

An Evolutionary Artificial Neural Network (EANN) based on the toad's visual system was proposed. A reinforcement based reward mechanism was selected to train the network to replicate the animal's learning behaviour in an environment containing prey and predators. After considering all of the types of Evolutionary Algorithms, an Evolutionary Strategy (ES) was chosen to evolve optimal networks in successive generations.

Interface of the Sensory System to the lower levels of the ANS:

The model developed was based on the sensori-motor connections in the toad that help the animal to differentiate between various objects in its environment and elicit appropriate behaviours. Hence, the outputs from the network are directly associated with the generation of appropriate behaviours, which constitute the lower levels of the ANS model.

Testing the network with real data input from a camera:

The proposed network was tested with real images of obstacles from a digital camera.

All the objectives mentioned have been met and the research undertaken to accomplish each objective is described in the following chapters. A brief overview of the chapters is given in section 1.4.

1.3 NOVEL ASPECTS OF THIS RESEARCH

During the course of research, several new ideas have been investigated in the field of Artificial Vision Systems. The development and implementation of an intelligent vision system called the DNN, based on the toad's visual system, is unique and is the major contribution of the research. The implementation of a biologically inspired training method called 'FUSION' that embodies aspects of Evolutionary Strategies, Supervised Learning, Unsupervised Learning, Reinforcement Learning and Fuzzy Logic. The Reinforcement based reward mechanism is also original.

The network, proposed and developed in this research was not only capable of recognising individual features in its environment, but was also extremely good at recognising combinations of these features, even though the network was not trained with these combinations. This is difficult to achieve using other techniques, including popular ANN configurations. The network is therefore useful for obstacle avoidance in robots employed in uncertain environments. The network needs to be trained only for individual obstacles to be able to handle all combinations in which the obstacles may occur.

Many of the experiments using Back Propagation (BP) networks and MNNs that are presented in the thesis have not been published elsewhere. The results obtained show that BP networks are not capable of recognising combinations of features that they are individually trained for, neither when a single large network was trained for all the features nor when modules of sub-networks are used for training similar classes of features. The results also throw some interesting light on the Grandmother Cell Theory, which has been a cause of some debate, but never through the type of experiments performed in this research.

The research also shows through artificial implementation that new learning processes can override initial genetic orientation. In other words, the results show that although the initial hard-wired networks that are genetically inherited by off-spring play a significant role, the amount of information that the animat analyses while learning to adapt to its surroundings is equally important. Hence, the networks are continuously re-wired, while maintaining the vital connections.

1.4 STRUCTURE OF THE THESIS

The framework of the Artificial Nervous System is explained in Chapter 2. The various modular networks assigned for specific tasks like Behaviour, Vision, Co-ordination and Higher Intelligent Functions are discussed in this chapter. Attempts made by other experts in the field to develop ANSs for Animats are also discussed.

Chapter 3 gives details on the literature search conducted on Traditional Pattern Recognition Techniques. During, the project, a critical review of these methods was completed, and is detailed in this chapter.

Chapter 4 gives an account of the explorations made by researchers studying MNNs and their biological significance. The initial experiments conducted on MNNs and the results obtained are discussed.

Chapter 5 discusses the experiments conducted on standard Back Propagation networks. A hardwired feature-detecting network is also explained in this chapter. Also included in this chapter are experiments conducted on MNNs.

Chapter 6 discusses biological vision systems. Particular emphasis is given to the insect eye among the lower animals. The studies conducted on the human visual system are also included here. A comparison of the various systems is given. The chapter also presents the biological background of the toad's visual system and some of the Engineering models developed by other researchers.

Chapter 7 explains the development of the Distributed Neural Network (DNN) that was based on the toad's visual system. The design process of the network architecture is discussed here.

Chapter 8 presents the implementation of the DNN. The methods used and the reward mechanism, which is very important for the successful implementation of the DNN is given in this chapter.

Chapter 9 discusses the training of the DNN and the tests conducted to verify DNN's pattern recognition capabilities. The results obtained from the first phase of testing are presented here.

Chapter 10 gives details of the tests conducted using combinations of features and gives an evaluation of the results obtained.

Chapter 11 presents the results obtained when the DNN was tested with digital images. The pre-processing details of the images are also given here.

Chapter 12 presents an evaluation of the DNN architecture and some of the results obtained when the structure of the DNN was altered.

Chapter 13 presents an analysis of the training methods and reward mechanism used to train the DNN during the course of this research.

Chapter 14 presents the discussion of results obtained in this research programme and proposes ideas for further work that evolved during the course of the research project.

Chapter 15 gives details of the project achievements and contributions to the state of the art.

CHAPTER 2: BACKGROUND

THE ARTIFICIAL NERVOUS SYSTEM

2.1 INTRODUCTION

One of the problems in developing Robotics and Autonomous Systems is training systems to learn and adapt to changing environments [6]. Neuroethologists, Cognitive Scientists, Ecological Communities, Computational Scientists and Engineers have brought a wealth of knowledge to the construction of intelligent behaviour in robots. Two of the main aspects studied in such Robots are their locomotion and vision. For example, simple Artificial Neural Networks are used to mimic walking in insects and study emergent behaviours [7]. Also, robotic vision systems based on biological vision systems have been developed [8] [9]. Other advances in Robotics include intelligent controllers developed using Artificial Neural Networks, Evolutionary Algorithms, Fuzzy Logic and Expert Systems, which have been proven to be powerful compared to the traditional controllers [10]. Inspired by these developments, researchers in various fields embarked on synthesising Artificial Nervous Systems (ANSs) to act as complete control systems for robots [11][12] [13].

This chapter discusses the origin of the ideas behind the ANS, developed at the Robert Gordon University, for which this vision system was developed. It gives the background to the project and puts the purpose and aims of the research into context. The chapter gives a description of the ANS in section 2.2. Sections 2.3 and 2.4 give an explanation of the modules involved with the locomotion of the robot. Section 2.5 includes justification of the methods used to evolve the vision network.

2.2 FRAMEWORK OF THE ARTIFICIAL NERVOUS SYSTEM

The Computational Intelligence group at The Robert Gordon University, Aberdeen has developed an Artificial Nervous System (ANS) inspired by the Biological Nervous System of higher animals. The model was proposed by MacLeod and Maxwell [14] and modified by McMinn [15].

The aim was to develop operating system models for the control of autonomous vehicles and other robots. Its purpose was to provide a flexible, modular, general-purpose control system for robots and other electronic devices. It has a layered, hierarchical structure as shown in Figure 2.1. The ANS is designed to be a feasible engineering solution rather than a precise biological model, and aims to incorporate the elegance of nature with the efficiency of modern technology.

The uppermost layer is where intelligent thought processes take place. Although Biological Nervous Systems are not completely understood at this level, this layer of the ANS would perform tasks including task planning, decision-making and other intelligent processes.

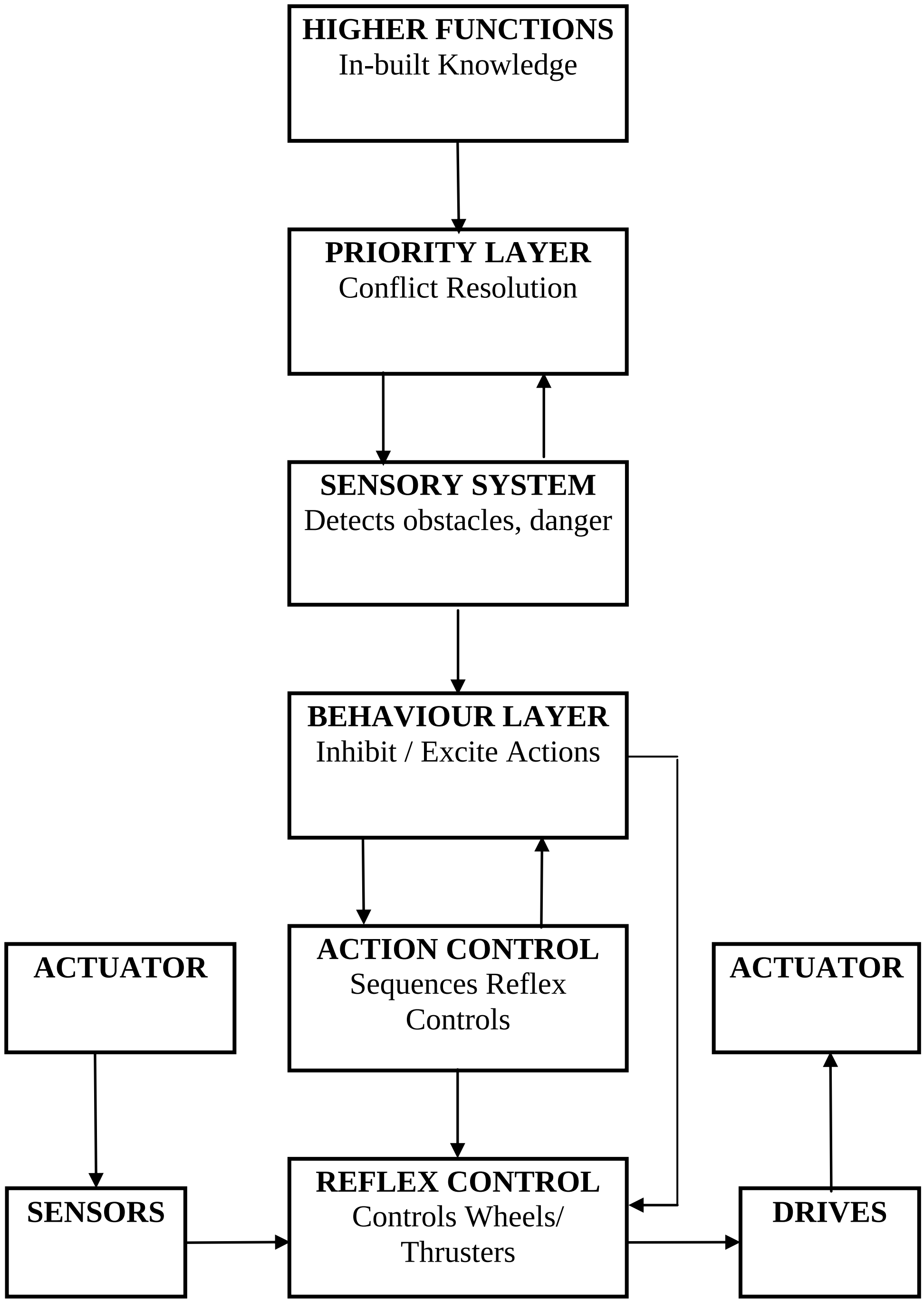


Figure 2.1: Framework of the ANS [14]

This is connected to the Priority Resolution Layer, which gives precedence to important tasks depending on the situation and internal status of the robot. For example, if the robot were “tired”, then it would need to find a location to rest or recharge. However, if the robot were in a dangerous situation, the Priority Resolution Layer would need to decide whether to give priority to a task for resting or navigating out of danger.

The task of the Sensory System is to recognise a particular element in the environment, for example, obstacles in the robot’s way, charging unit (food) or the presence of a potential danger (predator). The outputs of the Sensory System inhibit or excite particular behaviours, thus bringing the Behaviour Network into action. Behaviours are sequences of actions or combinations of actions on reflexes. Each module in this layer would activate the appropriate action modules to achieve the task. For example, in the case of danger avoidance, the behavioural module for that task might activate the actions “turn” followed by “run”. Actions such as “turn” involve co-ordinated and often repetitive control of groups of actuators that is undertaken by the Action Layer. The task of the Reflex Layer is to control individual actuators [14]. McMinn showed the implementation of the Reflex Layer and Action Layer of the ANS using Evolutionary Artificial Neural Networks [15].

Sections 2.3 and 2.4 give an explanation of McMinn’s work on the Reflex Layer and the Action Layer respectively. This implementation is presented here, as the vision network was developed to be interfaced with these lower levels of the ANS.

2.3 REFLEX LAYER

The functionality of the Artificial Reflex Layer was based on the biological reflex, which activates certain muscles or cells via motor neurons. Reflexes can operate with or without the involvement of the higher decision-making layers of the biological nervous system.

In vertebrates and invertebrates, the position of the body is controlled by the extent to which the muscles contract and extend. This allows for exactly the right amount of movement necessary to perform a certain action. Hence, animals are capable of slow, strong, fast and gentle movements. The number of muscles and their arrangement around different types of joints facilitate many useful movements that help in the survival of the animal in its environment [15].

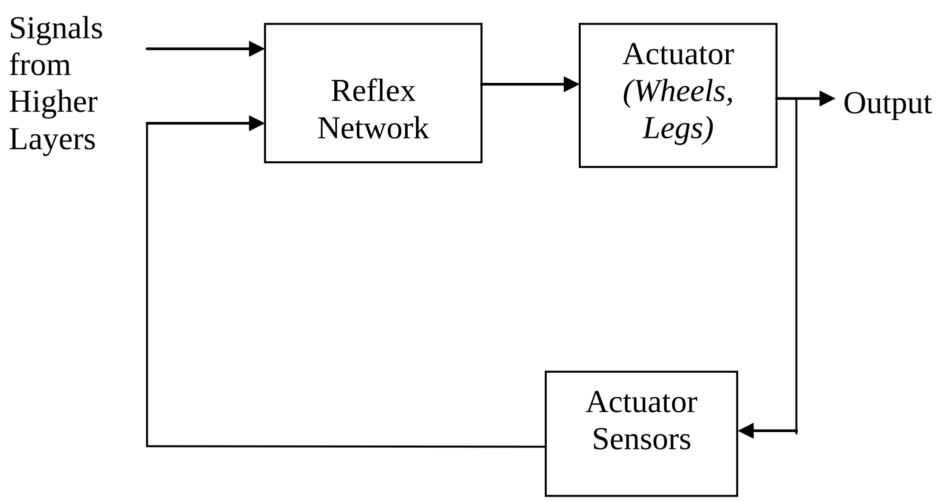


Figure 2.2: Interactions of the Reflex Network with other components of the ANS [15]

The reflex layer is the lowest layer in the ANS and is responsible for controlling the physical movements of the robot. The reflexes are connected directly to the actuators of the animat. Figure 2.2 shows the basic model of the modules involved in the reflex.

In McMinn's work, an Artificial Reflex System was developed using Evolutionary Artificial Neural Networks. A controlled supply DC motor was used as the actuator. On completion of one complete revolution, the actuator's movement was represented as a movement from position 0 to position 1. Reflexes involved movement of the actuator from an initial position to the desired position and then back to the initial position [15].

The network size was kept conveniently small for the various tests conducted on the desired reflexes. It usually consisted of 2 or 3 input neurons, a similar number of hidden neurons and an output neuron for the reflex signal [15]. The neuron model used was a standard perceptron type with a sigmoid squashing function [16].

The reflex network was trained using an Evolutionary Strategy. The weights were encoded as genes in a chromosome. The fitness of each chromosome was the average of fitness achieved for a series of tasks.

The fitness of each movement depended on 3 factors [15]. They are:

1. Offset: The difference between the position reached and the desired position.

2. Overshoot: The difference between the desired position and the first peak value of the position reached by the actuator while being trained to achieve its target.
3. Rise Time: The time taken to reach the desired position.

All the above factors were minimised during evolution of the optimal reflex network for various tasks.

Including feedback modelled on traditional controllers such as the PID (Proportional, Integral and Differential) Controller increased the complexity of the Reflex Layer [15].

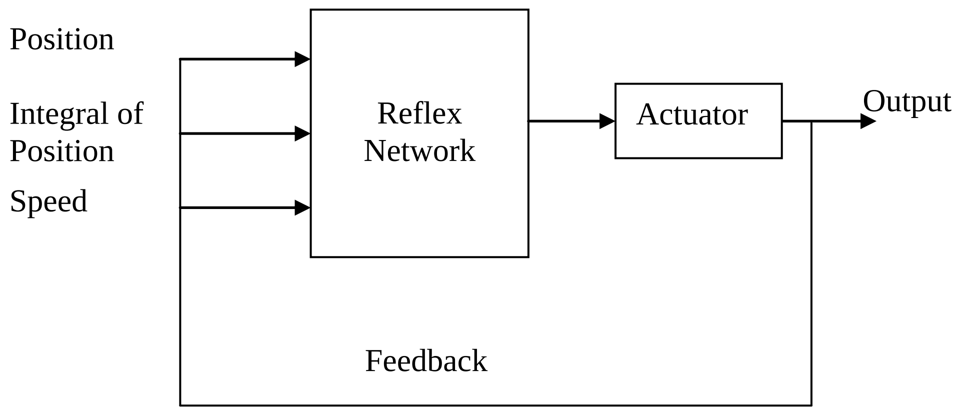


Figure 2.3: Feedback similar to PID Controllers incorporated in the the Reflex Layer [15]

The inputs to the Reflex Network were the Position, Integral of Position and the Speed of the actuator as shown in the Figure 2.3. Hence, the Reflex System of the robot could be improved by including appropriate sensory inputs available to the system from the surroundings, various feedback factors available from the actuator and increasing the type and number of reflex tasks performed by the system [15]. This could improve the versatility of the robot in adapting to changing terrains.

2.4 ACTION LAYER

In Biology, co-ordinated activation of several muscle reflexes produce repetitive actions like walking, running, swimming, etc. Dedicated neural networks called the Central Pattern Generators (CPGs) produce the required signals to accomplish these tasks. In a similar way to the Reflex Layer, the functions of the CPG can be automated or controlled by the higher layers. Figure 2.4 depicts a biological CPG and its interactions with other layers in the Nervous System.

McMinn generated Rhythmic Gait Patterns such as walk in a Bipedal Robot and trot and pronk in a Quadruped using Artificial Neural Networks trained using Evolutionary Strategies. The Artificial CPG had only one ON/OFF input signal, which represented the activation or inactivation of the CPG from higher layers of the ANS.

In the case of Biped Locomotion, the CPG consisted of four neurons of which two were dedicated output neurons for the two legs. The network used a spiky neuron model developed by McMinn [15].

The network used to produce the Quadruped gaits consisted of 8 neurons, where 4 were output neurons corresponding to the four legs of the animat. A single actuator was used as an output device for each leg [15].

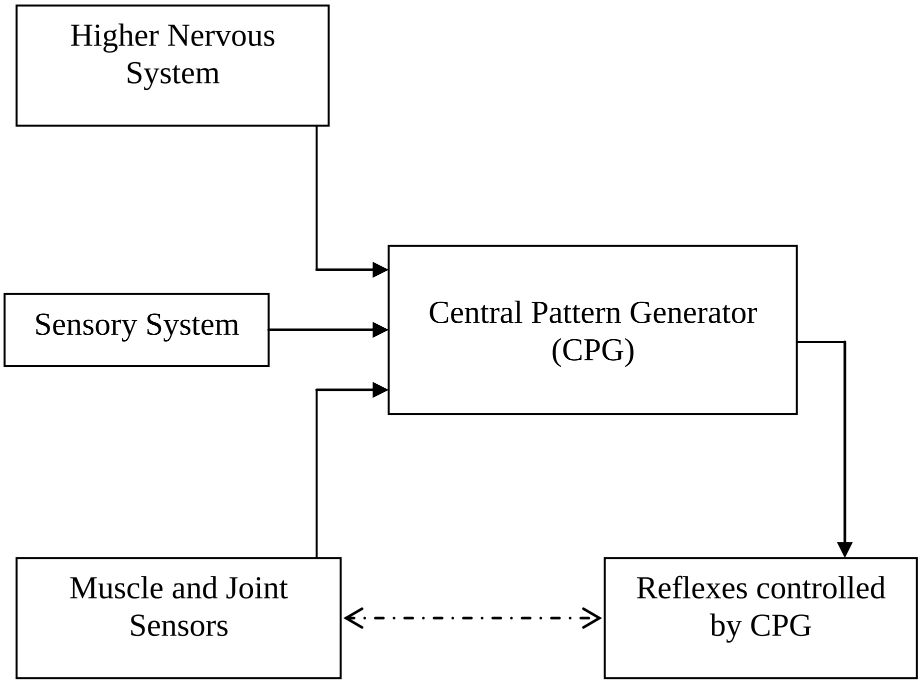


Figure 2.4: The Biological Architecture of CPGs [15]

Evolutionary Strategies were used to encode information related to the neurons in the CPG. Each gene had the following information regarding the neuron [15].

1. The Connection: The neurons to which it is connected
2. The Weight: The synaptic strength of the connection between the two neurons
3. The Delay: The time taken by the pulse to reach the post-synaptic neuron
4. The Type: The type of pulse (long /short) to be produced as output of the post-synaptic neuron

The output of the CPG was fed to the Reflex Layer (described in section 2.3) via a Leaky Integrator neuron, which integrated all the output pulses of the CPG and transmitted the signal to the appropriate reflex networks [15]. The leaky integrator is necessary to convert the spiky outputs from the CPG network to a form suitable for the perceptron types in the reflex as shown in Figure 2.5 [15].

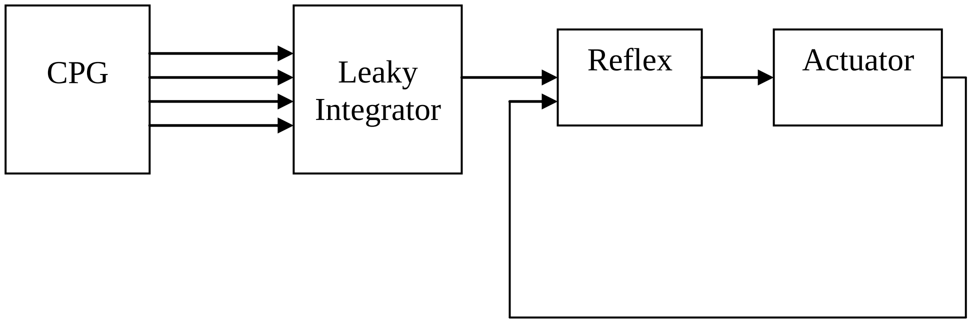


Figure 2.5: The CPG connected to the Reflex Layer [10]

2.5 THE VISION NETWORK

The main aim of the research presented in this thesis is to develop a Sensory System for the ANS, as mentioned in Chapter 1. Examination of the architecture of the lower layers of the ANS (explained under Sections 2.3 and 2.4) was undertaken to establish the design specifications of the Sensory Layer, which are as follows:

1. Implementation of a simple robotic vision system that would perform basic operations such as obstacle avoidance.
2. Development of a system that could be easily interfaced with the Action and Reflex Networks.
3. Conception of the intelligent vision system, preferably using Evolutionary Neural Networks, since this would allow the entire ANS to be built using a common training method.

4. Consideration of biologically inspired systems as a basis as this would blend into the biologically inspired ANS.

It was thought that the system should be able to recognise various obstacles in the environment of the robot. Typical representations of obstacles that the network could be trained for are shown in the Figure 2.6.

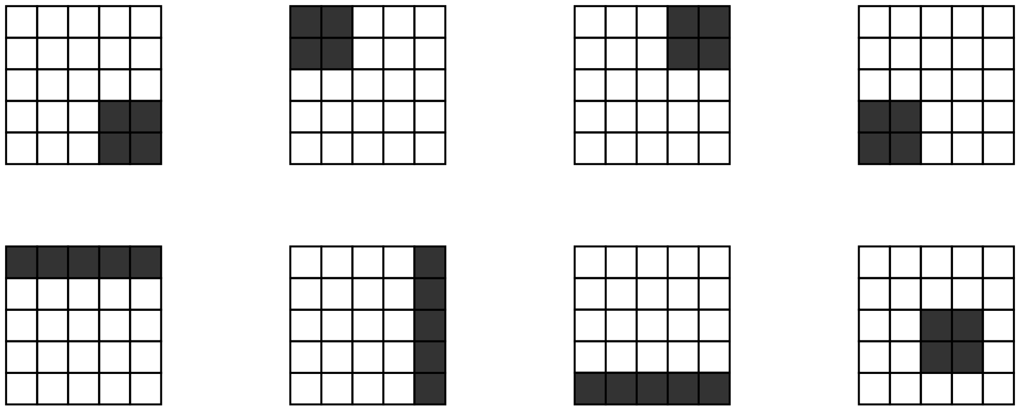


Figure 2.6: Obstacles in the Environment of the Robot

In a real environment, the robot would encounter combinations of such obstacles as shown in Figure 2.7.

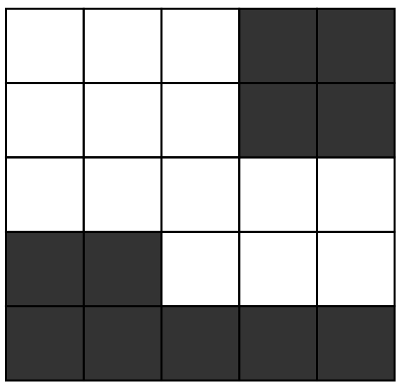


Figure 2.7: Combinations of Obstacles in an Uncertain Environment

The network training should also include some provision to deal with changing environments and noisy sensory data.

Obstacle Avoidance in Robotics is a well-known problem. Most of the obstacle avoidance systems rely on sensors for navigation [17]. Research on visual information being provided to robots to tackle the problem has led to some fruitful results [18]. Such systems usually employ AI methods to analyse the visual information [19][20][21][22].

Other systems developed are biologically-inspired [23][24]. Most systems are developed for specific applications and hence have limitations in adapting in other areas [25][26][27]. The reasons for developing application specific obstacle avoidance systems is that there are no known methods currently available that can be employed in changing environments.

The aim of the research programme is to study current methodologies used for processing visual information gathered by robots and develop new techniques that will enhance the robot's obstacle avoidance capabilities in changing environments.

2.6 SUMMARY

The Artificial Nervous System is a control system for Robots. It is a modular, hierarchical system designed to overcome the problems inherent in robot software design. The structure of the ANS has been described in this chapter. Also, the chapter gives an explanation of McMinn’s Reflex and Action Layer, which form the lower levels of the ANS.

This project investigates the vision system associated with the Artificial Nervous System. The chapter has presented the requirements of the Sensory Layer of the ANS, which are derived based on the overall structure of the system and previous work undertaken on its lower levels. A detailed investigation was carried out to implement a vision system for the ANS that fulfils these requirements.

The chapters that follow describe the methodical examination of Biological and Artificial Vision Systems necessary to identify an appropriate candidate system, the identification of problems in current methods and the gradual development of ideas in the implementation of the vision system.

CHAPTER 3: LITERATURE SEARCH

TRADITIONAL PATTERN RECOGNITION TECHNIQUES

3.1 INTRODUCTION

Pattern Recognition is the scientific discipline involving the classification of objects or any type of data into several categories. Since the aim of the project is to develop a vision system for the ANS that is capable of recognising various classes of patterns or features, a review and study of contemporary Pattern Recognition techniques was carried out.

The main aim of the literature search was to understand the capabilities and limitations of some of the most popular methods used for classification. Artificial Neural Networks and Mathematical Techniques are the most widely used methods for Pattern Recognition. As one of the requirements of the vision system being developed for the ANS was its compatibility with other modules of the system, the vision system was to be built based on Artificial Neural Networks, which form the basic building blocks of the ANS. Hence the emphasis of the literature review presented in this chapter is on Artificial Neural Networks. However, a brief overview of mathematical techniques used for Pattern Recognition is also presented here to give an overall view of the literature in this area.

In this chapter a general introduction to Artificial Neural Networks and their training methods is given in Sections 3.2 and 3.3 respectively. This is followed by an explanation of the supervised and unsupervised neural network training methods in Sections 3.4 and 3.5. Some of their applications in classification and artificial vision systems are dealt in Sections 3.6 and 3.7. Section 3.8 presents a brief review of the mathematical techniques. A discussion of these methods, considering their advantages and disadvantages and their suitability to the vision system being developed under this research is given in Section 3.9. A summary of the material presented in the chapter is given in Section 3.10.

3.2 ARTIFICIAL NEURAL NETWORKS

Artificial Neural Networks are based on Biological Neural Networks. Simple processing units called neurons are interconnected to form networks of neurons as shown in the Figure 3.1. McCulloch and Pitts proposed the basic neuron in 1943 [28].

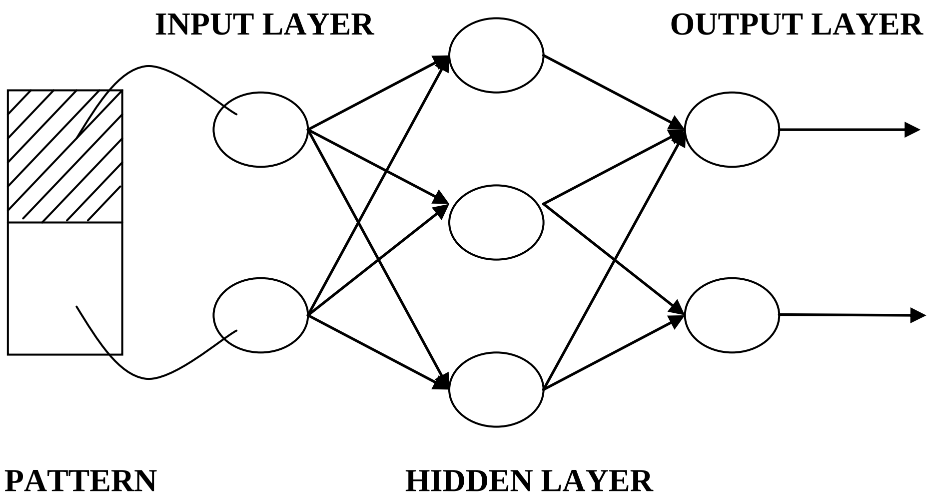


Figure 3.1: General Structure of an Artificial Neural Network

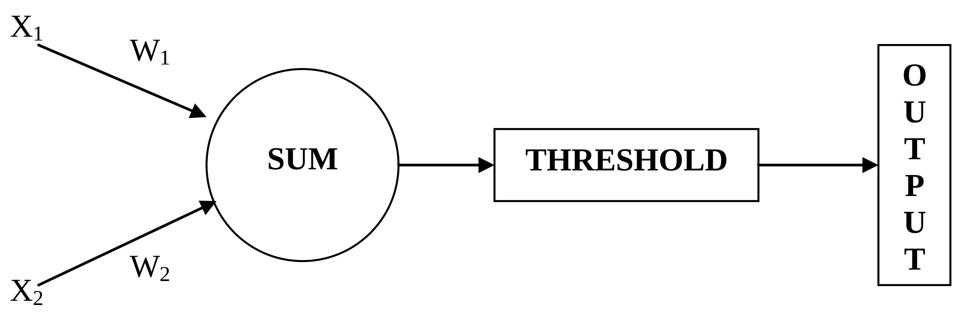


Figure 3.2: Illustration of the Operation of an Artificial Neuron

In the Figure 3.2, X_1 and X_2 are the inputs to the neuron, which are fed through the connection weights, W_1 and W_2. The weighted sum of the inputs is calculated as,

$$SUM = W_1X_1 + W_2X_2$$

Thus may be generalised to n inputs as,

$$SUM = \sum_{m=1}^{n} W_m X_m$$

The combined sum of the inputs, SUM, is passed through a threshold or squashing function to obtain the neuron's output. The most widely used threshold functions are the binary threshold, where the output is either 0 or 1, and the sigmoid function, where the output lies between 0 and 1 [28].

Binary Threshold

OUTPUT = 0 If SUM < Threshold Value

OUTPUT = 1 If SUM >= Threshold Value

Sigmoid Function

$$OUTPUT = \frac{1}{1 + e^{-SUM}}$$

The detailed operation of neural networks is not discussed further here as it is well covered in the literature [29].

3.3 TRAINING METHODS

Artificial Neural Networks are widely used in various Pattern Recognition and Classification applications. Traditional Artificial Neural Network paradigms use two main types of training techniques. These are known as Supervised and Unsupervised Learning methods [30] [31].

Supervised Learning is a method where the desired output of the network is specified prior to commencing the training. Quantitative feedback, usually in the form of an error is provided to the network on its performance and the network uses the feedback to create an accurate mapping between the input and desired output. Back Propagation is an example of a Supervised training method [30][32].

In Unsupervised Learning, the network has no target output to aim for. The networks that use unsupervised training methods are capable of identifying common features or patterns in a set of data. Such networks self-organise to enable groups of data (which have similarities) to form clusters [33]. An example of an Unsupervised Learning method is Competitive Learning, which is used in networks such as Kohonen's Self-Organising Maps and the Neocognitron [28][34].

To these two basic types of learning may be added a third type – Reinforcement Learning [35][36][37]. Reinforcement Learning could be considered a subset of supervised learning, but instead of a quantitative feedback signal, it operates using a qualitative signal (usually an indication of whether the network is performing well or poorly). Reinforcement learning is not widely implemented in practice, but is important in this research and the concepts therein are incorporated into the learning method used to train the final network discussed in Chapters 7 and 8.

All the Artificial Neural Networks presented in this chapter use the model presented above to calculate the outputs of their neurons. The sections that follow explain the training methods used to train networks of neurons that are useful in various applications.

3.4 BACK PROPAGATION ALGORITHM

Back Propagation is a Supervised Learning method that is used in many Artificial Vision Systems. It was developed independently at various times by Widrow, Parker and others [28].

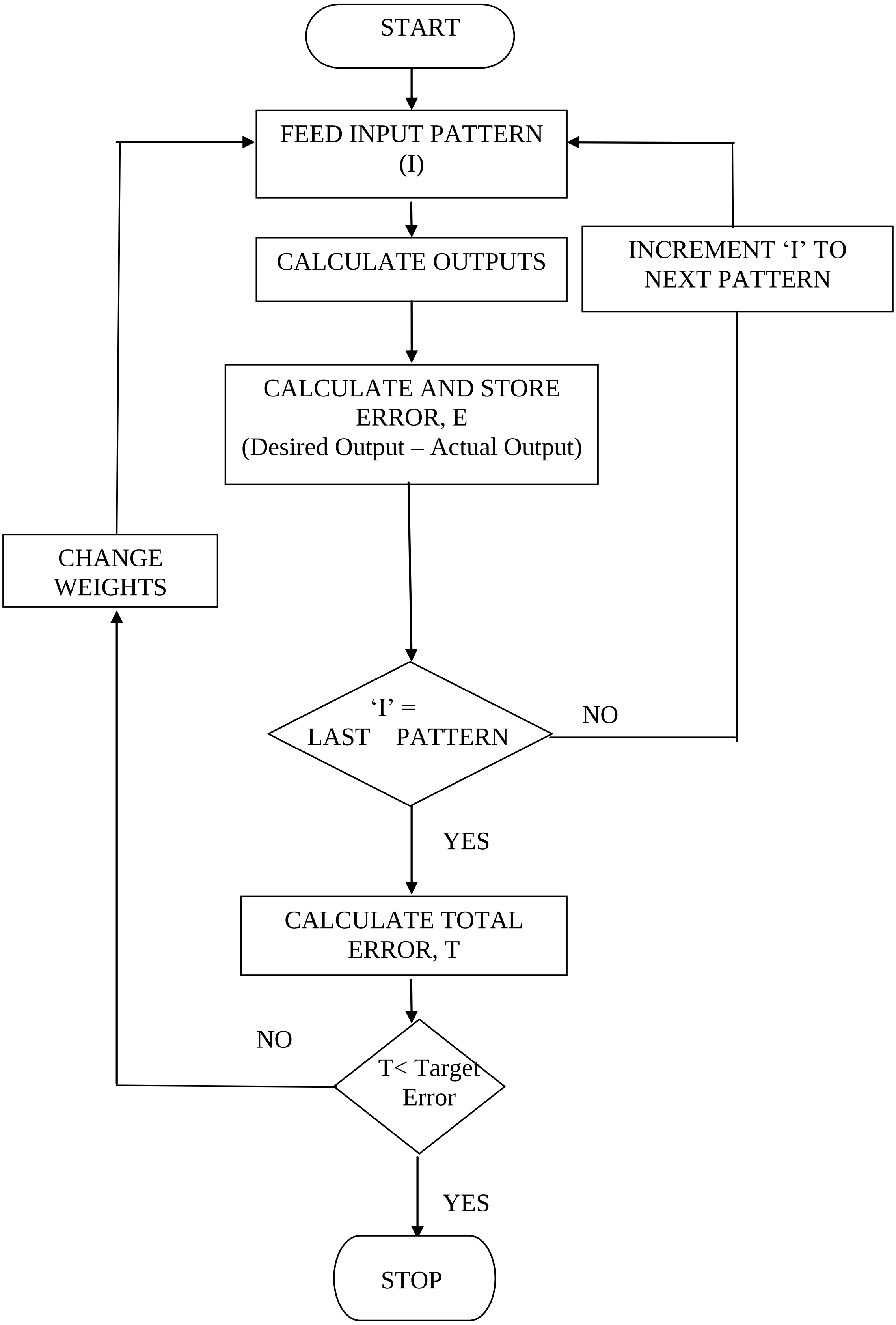


Figure 3.3: The Back Propagation Algorithm

In this method, there is the notion of an abstract entity similar to a teacher (supervisor), who specifies the desired output of the network provides instructive feedback to the network and monitors the progress of the training. The operation of the Back Propagation algorithm is illustrated in Figure 3.3. This method is mainly used to train fully-connected multi-layered feed forward networks of neurons.

The training data consists of several pairs of input patterns and corresponding desired output patterns. When an input pattern is applied to the network, it produces an output, which may not be the desired output. The error (i.e. the difference between the actual and the desired output) is calculated for all patterns. The total error, which is generally the Root Mean Square (rms) value, is used to change the weights of the network. The training continues until the error reaches an acceptable minimum [34].

3.5 COMPETITIVE LEARNING

Rosenblatt proposed the method of Competitive learning. It is an unsupervised method based on the principle of 'Winner takes all' [34]. There are several different ways in which it is implemented. Networks that employ Competitive Learning are made up of two layers of units, which are fully interconnected as shown in the Figure 3.4.

The first layer of cells simply passes the inputs directly to the second layer and no processing occurs. The second layer is where competition between neurons occurs and the output is produced. Neurons in this layer compete for dominance and one of the neurons wins i.e. succeeds in producing the highest activation [34]. The training process involves making the connections to the winning neuron stronger, while the other connections are made weaker.

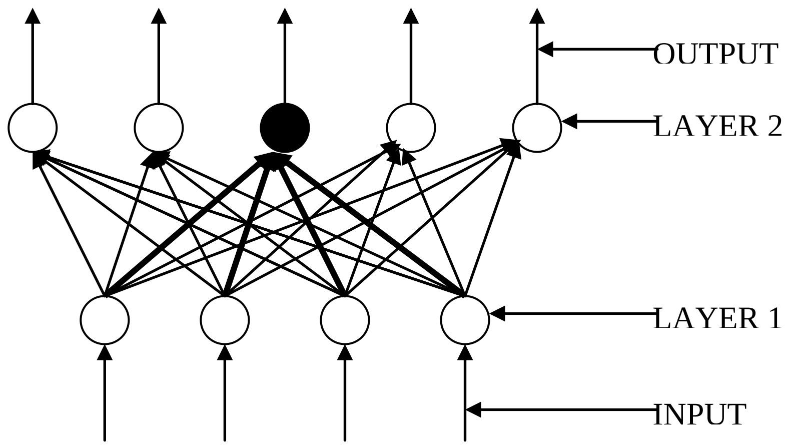


Figure 3.4: Illustration of Competitive Learning

As the sum of the connection weights of the network is kept constant, the training method allows for the winning neuron to become more sensitive to the winning input pattern. In this manner, various neurons become more or less selective to input data and hence succeed in classification [34].

3.6 KOHONEN'S SELF ORGANISING MAPS

Self-Organising Maps were invented by Kohonen. They use competitive learning methods for categorisation [38] [39] [40]. Unlike the network discussed in Section 3.4, neurons in the second layer are arranged in a two dimensional array. The network is fully connected [41].

The training method consists of not only changing the connection weights of the winning neuron, but also changing the connection weights of some of the neighbouring neurons to a lesser extent.

This creates ordered maps of neurons that are selective to related classes of input patterns and thus results in the classification of data, as shown in Figure 3.5 [41].

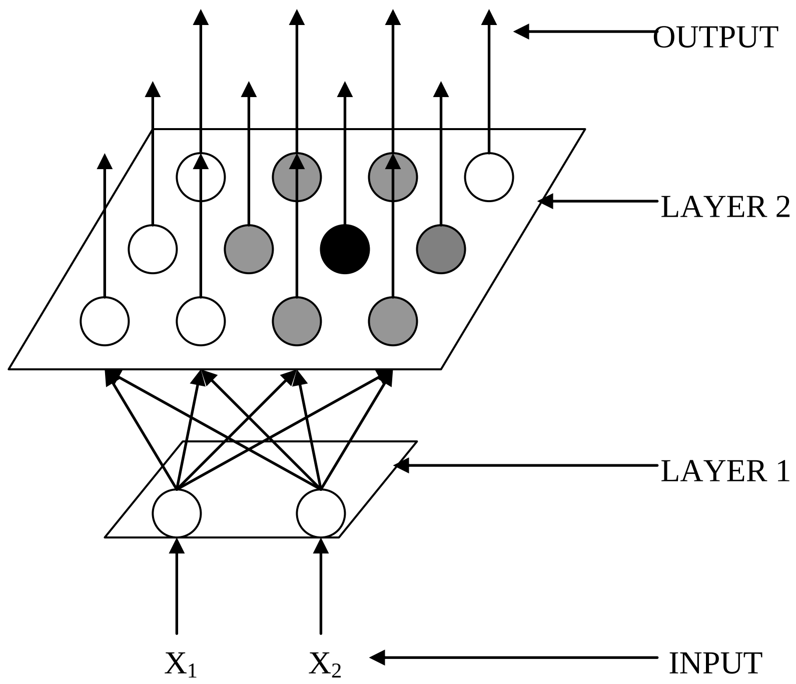


Figure 3.5: Self-Organising Maps

As the training progresses, the learning rate and the size of the neighbourhood are decreased to increase the stability of the network and to allow a finer classification of the input patterns [41].

3.7 NEOCOGNITRON

The Neocognitron invented by Fukushima is a multi-layered network that embodies the methods of competitive learning and self-organising maps discussed in sections 3.5 and 3.6 respectively. The system is based on the hierarchical structure of the primate visual cortex [42].

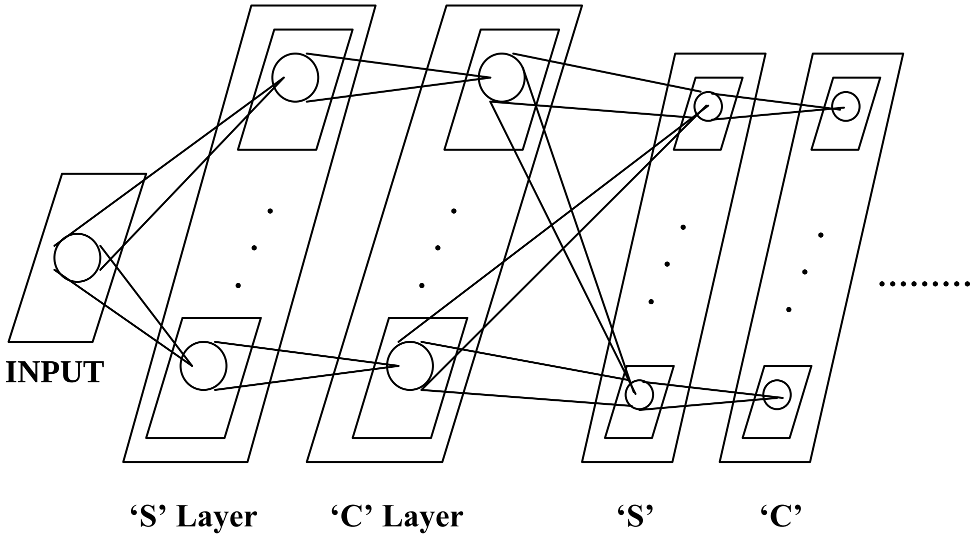


Figure 3.6: General Architecture of the Neocognitron. Redrawn from [42].

The concept around which the architecture was developed was that the main features of an image are recognised in the first few layers and integrated in the layers that occur deeper in the visual system [43][44].

Figure 3.6 shows the general architecture of the Neocognitron, which is made up of interconnected two-dimensional layers of neurons, each split into several modules called ‘Cell Planes’. There is a reduction in the number of cells in layers that occur at the latter stages of the visual process. Each stage is made up of two layers [42].

The first layer consists of S-cells, which extract features from various positions of the image. The second layer consists of C-cells, which have a larger receptive field than the S-cells and allows some tolerance of the spatial displacement of features to be accommodated. The C-cells respond to any S-cells (connected to that C-cell), which are activated due to the presence of a feature in its territory. Only the input connections to the S-cells can be modified. Thus, the competition between neurons takes place in the layer of S-cells [42].

The C-cells of each layer are connected to the S-cells in the next layer. Connections also run from the C-cells to an inhibitory cell called the V-cell, which stimulates a neighbouring S-cell with an inhibitory value that is equal to the average of the output of the C-cells that are connected to it as shown in Figure 3.7 [42].

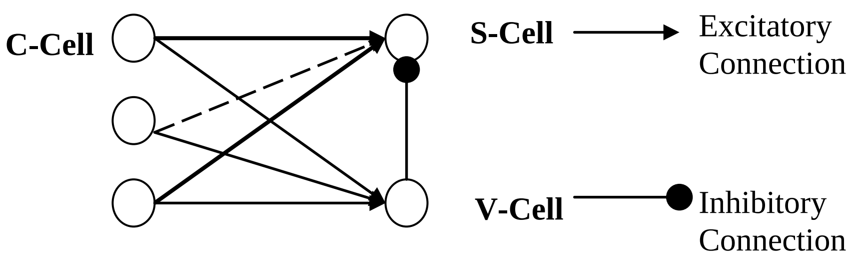


Figure 3.7: Interaction between C, V and S-Cells in the Neocognitron. Redrawn from [7].

Initially, before training starts, all connection weights are set to a very low value, shown by the dashed arrow. When an input pattern is presented, the connections of the winning S-neuron are strengthened, as is the connection to the V-cell. During training, the V-cell's connections are adapted such that its output is high for irrelevant features and is low for relevant features, while the appropriate reinforcement of S-cell connections are carried out. This process is continued with the repeated presentation of input patterns. At the end of the training, specific S-cells show a preference to certain features [42].

3.8 OTHER TECHNIQUES

Alongside the study of learning methods used in Artificial Neural Networks, the literature search into Pattern Recognition and Classification included a survey of other mathematical techniques to judge their suitability for implementing a vision system as described in Chapter 2, which is the main aim of the research project.

Mathematicians have developed several statistical methods that fall under the headings of Supervised and Unsupervised Classifiers. In all the supervised methods of classification, training data or reference data, which consists of feature vectors that belong to individual classes, act as prototypes. Supervised classification is achieved using statistical techniques such as Discriminant Functions, Minimum Distance Classifiers and Bayes Classifier [8][46][47][48]. Other methods such as K-means Clustering and Isodata Algorithms are used for unsupervised classification [49].

A typical example of supervised classification system uses Discriminant Functions. Discriminant Functions are linear combinations of a feature vector that are derived in terms of the respective reference vectors. The boundaries between the various regions are obtained using these discriminant functions [45].

Another method of supervised classification uses Minimum Distance Classifiers, which are based on the Euclidean Distance between a feature vector and the reference vectors. The feature vector is classified under a certain class, if the corresponding Euclidean distance is less than its distance from other reference vectors [45].

The last common method of supervised classification worthy of mention in this section is the Bayes Classifier. In this method, the classification is based on the Bayes rule, which states that a feature vector belongs to a certain class if the probability of the feature vector belonging to that class is greater than its probability of belonging to any other class. Hence, classification is performed using the probability density functions [45].

K-means Clustering is a type of unsupervised classification method. In this technique, a number of cluster centres are randomly chosen. The feature vector belongs to a class if the square of its Euclidean distance from the corresponding cluster centre is a minimum compared to the others. The process is repeated with new cluster centres until the data converges, which happens when all cluster centres remain the same and do not undergo any further changes [49].

The Isodata Algorithm, another unsupervised classifier, is an extension of the K-means clustering. To speed up the classification process and also improve the quality of classification, small clusters are discarded and large clusters are split, during training [49].

3.9 DISCUSSION

The Back Propagation algorithm has many successful applications in Image Recognition, Prediction and Control. It is estimated that Back Propagation is used in 70% of real world applications, despite some drawbacks [50]. Research continues to improve the Generalisation, Convergence Rate and Fault Tolerance of networks that use the algorithms [51]. However, if the aim is to develop an animal-like intelligent vision system, the back propagation algorithm fails badly. Networks trained for the recognition of certain images, when presented with the same set of images in a different spatial orientation or spatial displacement do not recognise the image. This is because the Back Propagation algorithm is capable of processing whole images and not identifying the individual features that make up the image. To confirm this, a series of experiments were conducted, the results of which are presented in Chapter 5.

Unlike the Back Propagation algorithm, Competitive Learning, which is based on competition between neurons in the biological brain, is biologically plausible and has found many applications in Clustering and Classification. Advances in this field include Soft Competition, where neurons in a layer are interconnected by other neurons that provide lateral inhibition. Here, the winning neurons inhibit only the adjacent neurons and not all other neurons in that layer [52]. It was shown that switching off the inhibition causes failure in extraction of many features. This shows that the networks that use Competitive Learning have certain advantages over other vision networks, which do not employ lateral inhibition [52].

Since Competitive Learning is unsupervised, there is very little external control that can be exerted on the network. The network classifies features or patterns that may be present in a given data set, which are difficult to identify and are unknown to the user. The classification is dependent on the initial weights of the network, the input stimulus and the number of neurons in the competing layer. As there is no general rule available in setting these parameters, classification is based on trial and error. This type of learning is extremely useful in applications such as Stock Market Prediction, where a system is required to identify and classify various features from a large data set that makes no apparent sense.

If the data consists of simple features that occur frequently, then classification using Competitive Learning is stable. However, if the features are more random and do not occur frequently, the classification process becomes highly unstable and may result in the misclassification of features [53]. This method of classification cannot identify rare but important features in the data and there is no simple way of altering the design of the network to accommodate this [54].

Also, as networks that use competitive learning have only one layer of neurons, they are only good at extracting the lower level features, but not the high level features (e.g. faces or combination of several obstacles), which some networks employing Back Propagation algorithms are good at. Both learning algorithms are completely incapable of dealing with new patterns, unless re-trained from an initial starting point.

Unsupervised and supervised learning methods can be integrated to capture the best of both of their capabilities. For example, as the unsupervised learning is capable of extracting features from a given data set, it can be used for pre-processing data before it is passed onto a supervised pattern recognition system, which can recognise higher order features [55][56].

However, this cannot be fully accomplished because the supervised learning methods, such as Back Propagation cannot recognise images from individual features. This aspect of Back Propagation algorithms is further explained in Chapters 4 and 5.

Systems developed for real time image recognition, such as that developed in this research, should have a simple and elegant architecture and training process to reduce the training and response time as much as possible to allow for online capture of new information and training. In addition to this, if the system can achieve its objectives with the minimum number of internal operations (mathematical calculations, etc.), adaptability of the system to machines with varied processing capabilities becomes possible.

The Statistical Techniques discussed in Section 3.8 and the Neocognitron discussed in Section 3.7, even though make good classifiers, they fail to meet the requirements mentioned above of an intelligent robotic vision system. Due to the restrictions caused by the single layer of processing neurons in competitive networks and little control over its performance and noise tolerance, these networks are also less suitable for the purpose of this research. Hence, despite the limitations, considering merits (e.g. ease of training) and its success in application areas such as Pattern Recognition, the networks that employ Back Propagation were chosen for further investigation as a first step towards building a vision system for the ANS. The results are presented in Chapter 5.

3.10 SUMMARY

This chapter has outlined the results of a study of different methods used for Classification and Image Recognition. A critical appraisal of the various techniques presented in this chapter was carried out to evaluate the advantages and disadvantages of using these techniques to develop the vision system of the ANS as discussed in Chapter 2. The findings were discussed in Section 3.9.

It can be gathered from the literature presented in this chapter that the Neocognitron is relatively better than the other networks in complex classification tasks. This is mainly because of its complex hierarchical architecture. Such architectures allow for a sophisticated integration of image elements and results in a better performance in these tasks.

However, employing large and complex networks for these tasks increases training time. This can be avoided by decomposing the problem at hand into several smaller tasks that can be handled by modules of simpler networks. Such networks are more popularly known as Modular Neural Networks. Such networks are inspired by the nervous system, which has been shown to be modular at all levels of its structure.

The ANS is modular and many Artificial Neural Networks employ modularity in their structure. Therefore, conducting a literature search on Modular Neural Networks was considered to be essential for a complete review of Artificial Neural Networks used in Pattern Recognition. This is presented in Chapter 4.

CHAPTER 4: REVIEW OF LITERATURE

MODULAR NEURAL NETWORKS (MNNs)

4.1 SIGNIFICANCE OF MNNs

The Artificial Nervous System (ANS) proposed by MacLeod and Maxwell for applications in Robotics is modular (as mentioned earlier in Chapter 1 and described in Chapter 2), like many engineering systems for convenience of design, maintenance and repair. Modular structures also provide the flexibility to add additional modules that can cater for new tasks to improve performance. Therefore, modularity was the core element around which the ANS is modelled with the aim of deriving an elegant engineering solution to solve problems faced by Animats.

Modularity is also evident in the evolution and development of various biological systems from a single-celled amoeba to human beings. Studies throughout the animal kingdom suggest modular development of the embryo and foetus [57]. Research is being undertaken on modularity in Artificial Neural Networks (ANNs) that has lead to the implementation of several architectures such as Gating Systems [58]. Such research is also proving beneficial to both Engineers and Biologists [59].

The modular evolution of systems in biology and the practical benefits of such systems in engineering were considered to be very significant in developing an

Artificial Vision System for the ANS based on ANNs. Hence, a literature search was conducted on modularity in biological systems and ANNs.

This chapter presents some of the findings of the literature search on Biological and Artificial MNNs. Section 4.2 on Biological MNNs includes a description of modularity in the development of the brain (Section 4.2.1), which forms functionally specialised neuronal clusters (Section 4.2.2). Sections 4.2.3 and 4.2.4 discuss the inter-dependence between such modules and the communication between neurons respectively. Examples of modular architectures in ANNs are presented in Section 4.3.

4.2 BIOLOGICAL MNNs

Scientists have long studied the vision systems of animals to try and understand the mechanisms, which give the animal the ability to recognise various elements in its environment. Although the vision system is the most studied sensory system and several advances have been made, many unanswered questions remain to this day.

However, the research done so far has shown that the vision system in the biological brain has a highly modular structure. This allows a systematic method to be followed to process information. The subjects used for this area of research are usually primates such as macaque monkeys and human beings [60][61][62]. One of the disadvantages of conducting research on higher animals is the difficulty in identifying the finer details of neuronal interactions due to the presence of millions or billions of highly specialised neurons. Despite the complexity of the problem, several interesting aspects regarding the modularity of the biological visual system have been identified. These are discussed in Sections 4.2.2, 4.2.3 and 4.2.4.

Modularity is not simply confined to the visual areas of the brain, but has an important part in the embryological development of the brain as mentioned in Section 4.1 [57]. This is briefly discussed in Section 4.2.1.

Research shows that modular structures are not entirely the result of learning during the developmental stages of an animal. Modularity is at least partly due to the genetic make-up, which is responsible for hard-wired modular structures. Chapter 5 presents some of the experiments performed on modularity in ANNs with particular emphasis on learning and hard wiring - aspects which are thought to exist in biological neural networks [63][64][65].

4.2.1 DEVELOPMENT OF THE BRAIN

Studies on the embryological development of the brain have revealed many interesting facts regarding biological MNNs. A few major stages have been identified in the development of the brain. This is illustrated in Figure 4.1.

The first stage in the development of the brain is the induction of the neural plate, which is followed by localised cell proliferation in different regions, i.e., the division of these cells in different regions of the brain. The cells then migrate from their birth regions of generation and aggregate with other similar cells to form identifiable parts or modules of the brain.

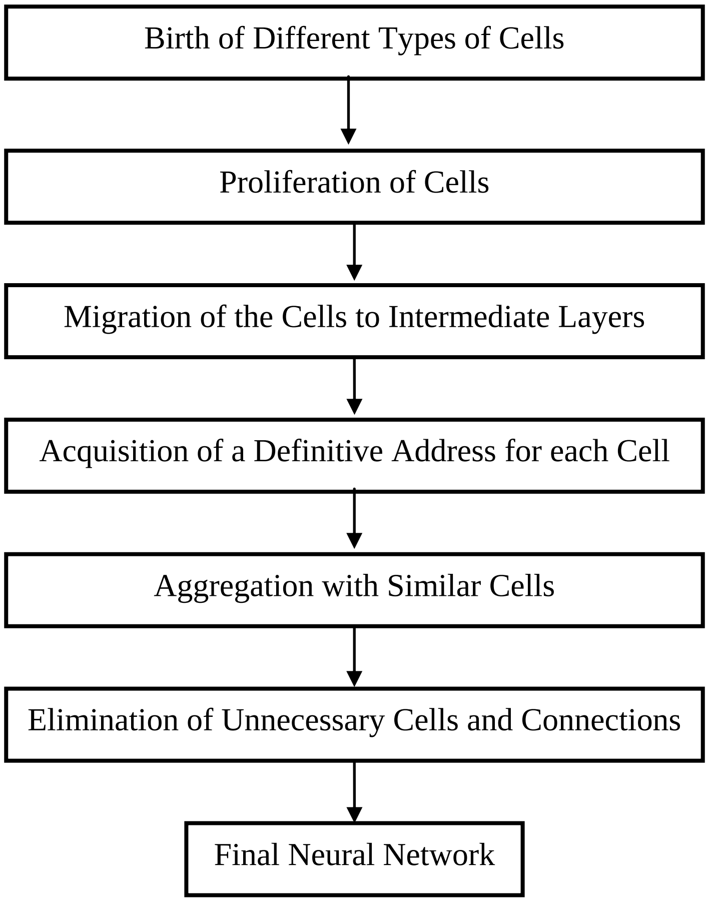


Figure 4.1: Development of Biological Neural Networks

The final stages involve development of connections between various neurons, selective death of some cells and elimination of some of the unnecessary connections. This process determines the ultimate network pattern that a neuron forms [66].

In 1912, Richard Goldschmidt proposed that neurons are unique. His findings were based on the examination of the nervous system of a primitive intestinal parasitic worm, Ascaris. The brain of this worm consists of exactly 162 cells, which were organised in several clusters and never varied from animal to animal. The spatial position of the cells and the connections formed between cells were consistent in these worms [67].

In fact, the consistency of these connections was so strong that when the various cells were disassociated and mixed in an artificial medium, they formed the same connections that were originally observed in the animal's brain [66].

The fact that cells and connections between cells are invariant was reinforced by research done by Wesley T Frazier, Irving Kerpfermann and Eric R Kandel on the large sea snail Aplysia [67].

The mounting proof on the consistency of different types of neurons (in various species of animals), the connections and clusters that the neurons ultimately form convinced researchers that such cell assemblies were functionally specialised, i.e., a group of similar cells aggregated together are usually dedicated to perform a certain task such as recognition of objects[68][69][70]. Section 4.2.2 presents a review of the research being done on functionally specialised neuron groups.

4.2.2 FUNCTIONALLY SPECIALISED NEURONAL CLUSTERS

Research on Pattern Recognition in humans included the identification and classification of various objects such as letters, digits, objects (chairs, tables, etc.) and human faces. Functional imaging techniques were used to study the activities of brain areas in human beings and higher intelligent animals (macaque monkeys) during various recognition tasks to identify the neuronal clusters involved in carrying out the task. Results from such experiments have contributed to the understanding of the neuroanatomy, neurophysiology, and the modularity of the human visual processes [71]. The knowledge of biological modular neural clusters was the main influence behind the design of Artificial MNNs.

The sections that follow present studies, which suggest the presence of functionally specialised neuronal clusters in the human brain.

4.2.2.1 HUMAN FACES vs. OBJECTS

Martha J Farah attempted to confirm whether face and non-face object recognition could be accomplished by different systems. For systems to be considered different they must be functionally independent, i.e., each system must be able to operate without the other [71] [72].

Prosopagnosia, the inability to recognise human faces after brain damage, is a neuropsychological impairment. People suffering from severe conditions of this disorder cannot recognise faces of even close friends and family members. However, people with this condition have no problem recognising objects made of different shapes and colours [71].

The unique condition of prosopagnosia suggests the loss of a face recognition system in the visual areas of the brain. On the other hand if face recognition is a specific type of object recognition, then the hypothesis is that the system might have lost the ability to recognise complex objects such as faces. Hence, many scientists assumed that the areas of the brain involved in face recognition would be more or less involved in object recognition [71].

Experiments were conducted on prosopagnosics and normal subjects to test the hypotheses of the presence of just one object recognition system, where faces constitute a specific category of objects and the presence of a separate face recognition system. The results show the existence of a specialised system, which is anatomically distinct for recognising human faces in the human brain. This system or network of neurons is not necessary for recognising other objects such as sheep faces. The results also suggest that the specialised face recognition system processes information in parallel with the object recognition system [71].

Another observation from the experiments conducted was that faces were processed as whole images, while shapes were processed based on the features that made up the object [71].

Section 4.2.2.2 explains the study conducted using categories of objects such as letters and digits that are very similar to each other unlike faces and general objects, which are quite distinct.

4.2.2.2 LETTERS vs. DIGITS

Thad A Polk, et al, conducted research to assess the response of the visual areas of the brain in recognising similar patterns such as letters and digits. For the purpose of the study, both letters and digits were chosen to be line patterns and no effort was made to differentiate between the two classes of patterns and letter-digit pairs such as O-0 and I-1 had very similar features. Functional neuro-imaging techniques were used to map the area of the brain involved in letter and digit recognition respectively [72][73].

The results suggest that an area in the left inferior occipitotemporal cortex responded significantly more to letters than digits. Therefore, the fact that certain brain areas are more activated by letters than digits indicates that individual systems of neurons process these patterns as specific categories [73].

4.2.2.3 SHAPE (FROM SHADING) vs. SHAPE (FROM EDGES)

Humphrey, et al, studied functionally specialised clusters of neurons in the visual areas of the human brain by choosing images of three-dimensional objects represented with and without shading. The pattern of shading on the object depends on number of factors including the type of curvature of the object's surface relative to the viewer and the reflectance of the surface.

The conclusion of the research was that neural pathways that compute shape from shading may be independent of those that compute shape based on edges

[74].

Thus, from the case studies presented in Section 4.2.2, it appears that separate modules are associated with the recognition of various categories or classes of patterns. Section 4.2.3 deals with the research undertaken to understand the interaction between such modules.

4.2.3 COMMUNICATION BETWEEN MODULES

An analysis of the concepts and schemes identified during the literature search indicated that, when the brain processes an image, the various functionally specialised neuronal clusters analyse the tasks that they are dedicated for. Hence, some neurons respond to faces, while others may respond to finer details of the image like shape, colour, orientation, etc. Unless these specialised modules interact with each other, a meaningful interpretation of the image cannot be achieved. This problem of collating information together is known as the 'Binding Problem' [75].

Reported theory suggests that when the different elements of the image are identified, areas deep within the brain, which form the ventral visual pathway, integrate the information to recognise the entire picture (See Section 5.5 of chapter 5 for more information) [76]. However, it is still unclear how, for example, the colour 'red' is associated with the shape of an apple in the case of a red apple and not to a banana (which is present in the same basket of fruits and avoids misinterpretation of the fruit as a red banana!). A number of researchers believe that the answer to this query may relate to brain waves called ‘Gamma Rhythms’.

Electrodes, placed on the scalp have been used to record the overall activity of the cells in the brain [75]. Several brain wave patterns have been observed during various conscious and unconscious tasks. The brain waves that oscillate 40 times per second, called the Gamma Rhythm seem to act as a background template, whose peak values at a given period of time synchronised with the firing of various specialised neurons associated with a certain object [75].

There is evidence of participation of the parietal and frontal cortical areas during the performance of a visual task [77]. Also, studies show that inactivation of the non-visual cortex, eliminates visual behaviour, although the neurons in the visual cortex are active [78].

Despite the evidence of functionally specialised neural clusters and the communication between them, it is questionable whether a category (shape, colour, size, etc,) specific organisation is possible, given the limited amount of cortex available [77].

Although, biological MNNs may not be fully understood at their fundamental level, advances have been made in the field of Artificial MNNs based on the biological knowledge available so far. Section 4.3 presents a broad classification of the Artificial MNNs and discusses some popular modular architectures.

4.3 ARTIFICIAL MNNs

The incorporation of modularity in ANNs has brought about several advantages such as a reduction in model complexity, the ability to learn different tasks simultaneously, robustness (the deletion of a module will not damage the recognition of other modules) and incrementality [79].

Most artificial MNNs can be broadly classified based on the following three main operations that they employ to carry out any recognition task. They are [80][81],

1. Partitioning of Input Space
2. Successive Processing
3. Combination of Decisions

However, this classification is coarse. Many models have been developed with slight variations and combinations of these basic functions. Sections 4.3.1, 4.3.2 and 4.3.3 explain the operations mentioned in more detail.

4.3.1 PARTITIONING OF INPUT SPACE

In this method of employing MNNs for a recognition task, the input data space is divided into several smaller data sets called sub-spaces. Each sub-space is then assigned to a system for identification. Here, the modules of the MNN are dedicated to specific tasks. A priori knowledge of the data set is assumed in this type of MNNs [80]. The “Mixture of Experts” system (discussed in the section 4.4.1) falls into this category [82][83].

A typical example of a MNN based on partitioning of input space was used for speaker identification. The input space consisting of utterances by several speakers was broadly classified according to their vocal characteristics. These were then fed to neural networks or expert systems to train for fine identification of individual speakers. The probability that a certain utterance belongs to a speaker was used to factor in additional knowledge along with the activation of the outputs of individual systems. Over a period of a sentence, the successive activations were combined to arrive at a final decision successfully [80].

4.3.2 SUCCESSIVE PROCESSING

In MNNs based on successive processing, the problem is divided to smaller, manageable tasks. These sub-tasks are handled by various systems successively to analyse the problem [80].

Hybrid networks that use a combination of Neural Networks and Hidden Markov Models or Linear Vector Quantisation are based on this method. Kari Torkkola and Teuvo Kohonen have proposed a hybrid network for speech recognition where the processing takes place in successive layers [80].

The ANS model described in Chapter 2 also falls under this category, as the networks in several layers handle the problem successively in a hierarchy.

4.3.3 COMBINING DECISIONS

Some MNNs are successful in generalisation, when modules in the network are trained for a task in parallel and an average of their outputs was taken. It was found that the rate of classification had a direct influence on reducing the mean squared error when such a combination of outputs has been used [80].

Many other methods for combining outputs such as fuzzy aggregation with ordered weighted averaging operators have been proposed to overcome problems faced by MNNs, where the outputs of the modules vary drastically and the application of simple voting or averaging may not yield good results [84].

4.4 ARCHITECTURES OF MNNs

4.4.1 GATING SYSTEMS OR MIXTURE OF EXPERTS

The 'Mixture of Experts' system is sometimes also called the 'Gating Network'. A brief description of the system proposed by Jordan and Jacobs is given in this section [85]. Many other variations of the system have been tested successfully for various recognition tasks.

Figure 4.2 depicts the mixture of experts' architecture. It consists of network modules called experts. Each network maps inputs x to outputs μ_i as it is trained for a particular task. As mentioned earlier, this system decomposes input data space into sub-data to be handled by each network. This introduces ′a priori knowledge into the system. The gating network, based on the knowledge of the input, produces probabilities, g_i [85].

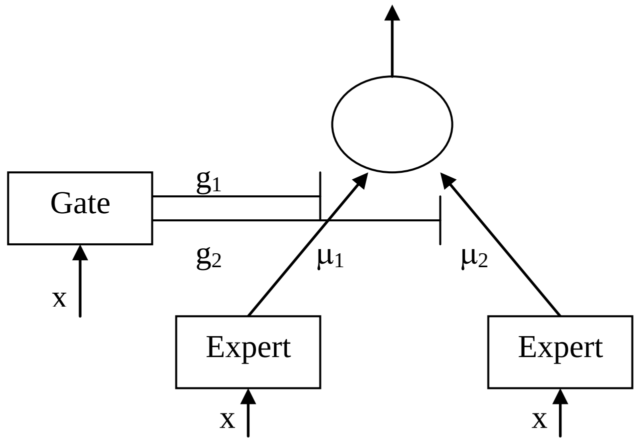


Figure 4.2: Mixture of Experts. Redrawn from [85]

The g_i values represent the probability of a particular input belonging to a certain class. The probabilities are expressed as non-negative, scalar coefficients whose sum is unity. The coefficients vary as a function of the input and hence do not stay fixed. The gating coefficient is used to weight the outputs of the expert networks [85].

In more complex problems, the expert networks can be sub-divided further into sub-expert networks. This approach helps in refining the process of classification, increasing the success rate, while reducing the training time and complexity of the problem [86].

4.4.2 MIN-MAX NETWORK

In this type of network, a K-class classification problem (a data set consisting of 'K' different classes) is divided into a number of two-class problems. This is achieved by training each module of the network for two classes using any learning algorithm (usually supervised learning) [87].

Consider an example of a three-class classification problem. When the three-class classification problem is decomposed into two-class problems, there exists the subsets 1-1,1-2,1-3,2-1,2-2,2-3,3-1,3-2 and 3-3, where 1-2 is a two-class problem involving the classification of classes 1 and 2 and so on.

Also subsets such as 1-1 are made redundant, as they do not play any role in the classification process. Subsets 1-2 and 2-1 are classifying the same classes 1 and 2. In most cases these modules are allowed to remain for classification to improve its ability to generalise. Once the final number of two-class problems has been obtained, separate modules handle each two-class problem [87].

When module, M12 is trained to classify patterns 1 and 2, it classifies all test patterns as belonging to either class-1 or class-2. Even a class-3 pattern in region 'U' may get a high response as the classification is now based on only two patterns. This is accurate for the local network, but is not accurate on a global scale. This is illustrated below in Figure 4.3.

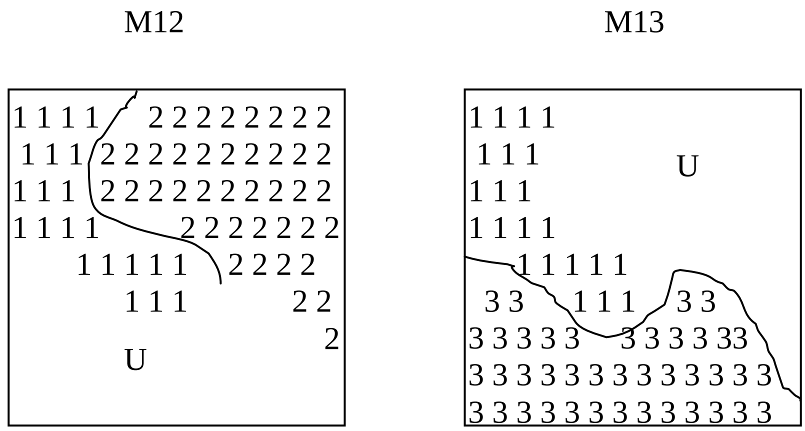


Figure 4.3: Illustration of decomposition of a three-class problem into a two-class problem. Redrawn from [87].

To eliminate this problem, a minimisation (MIN) unit followed by a maximisation (MAX) unit is used as shown in Figure 4.4. The minimisation unit picks out the minimum output from each network and then it is passed on to the maximisation unit, which picks out the maximum of the inputs to the unit [87].

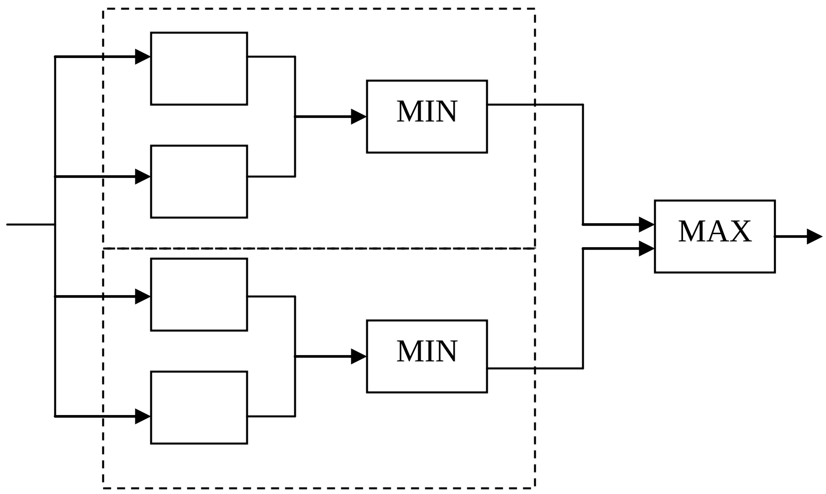


Figure 4.4: Min-Max Modular Network. Redrawn from [87]

The maximisation and minimisation units act as filters and thus draw boundaries between various classes [87].

Such networks with maximisation and minimisation units are sometimes also called M^3 networks. The third module that may or may not be included along with the MIN and MAX units is the inverter (INV). The inverter acts like the NOT gate and inverts its input. The transfer function of the INV unit can be given by,

$q = b-p$, where q is the output,

b is the upper limit of the input

p is the input to the unit

The network has been successfully applied to various problems such as the XOR, DNA and two-spiral problem [20] and Handwriting Recognition [88].

4.4.3 CO-OPERATIVE MODULAR NEURAL NETWORK (CMNN)

The CMNN proposed by Kamel and Auda uses an unsupervised network to decompose the classification task into several homogeneous regions. As in most MNNs, each sub-task is allocated to a module of the CMNN as shown in Figure 4.5 [89].

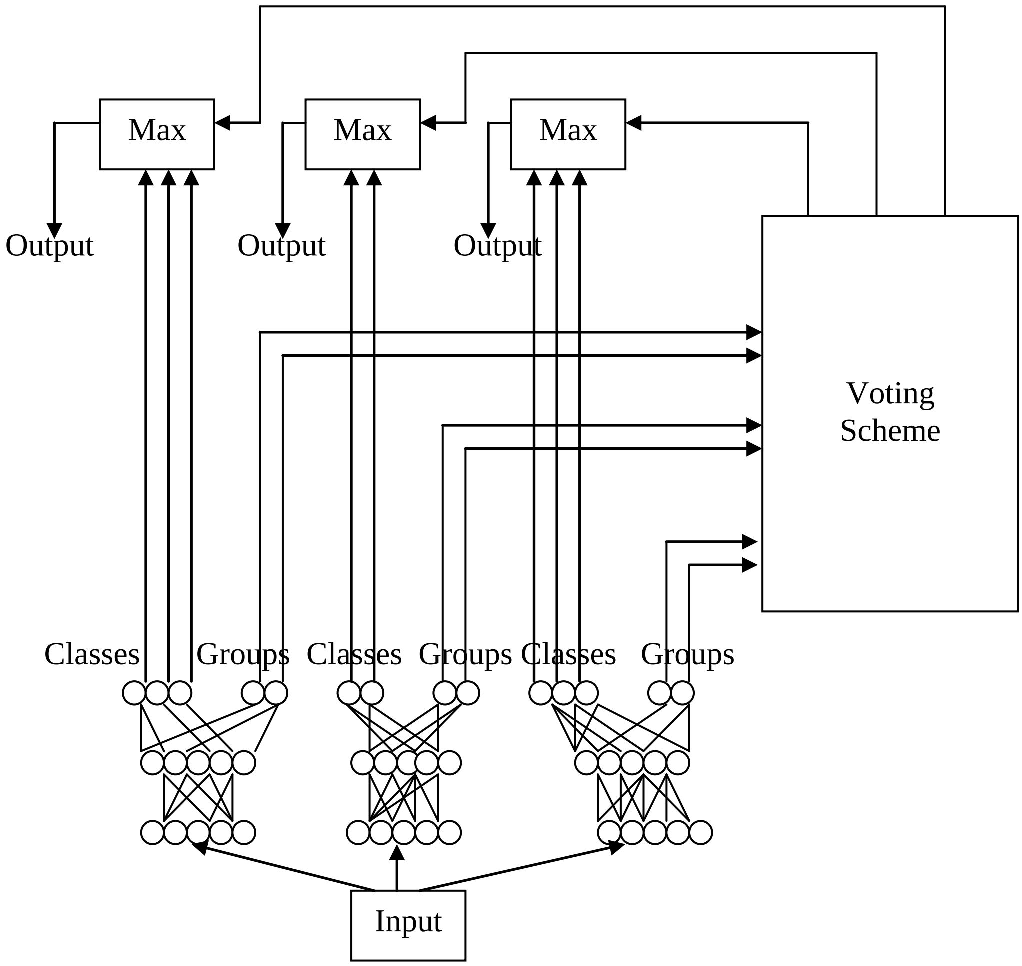


Figure 4.5: Co-operative Modular Neural Network.
Redrawn from [89]

All modules are trained using samples representing that particular module and all other modules, i.e., the modules are trained using the complete training set. The outputs consist of the number of classes in the group and the number of other modules. Modules can be trained in parallel using any supervised training method [89].

The process of classification is based on the highest activation value of various neurons. When a pattern belongs to a certain class, the corresponding output neuron shows an activation value approaching 1 and the rest of the neurons corresponding to the other classes and the neurons representing the other groups have an activation value closer to zero [87].

The group outputs are passed on to a voting layer, where every group tries to inhibit or excite other modules. The group output with the highest votes succeeds in classifying the pattern as belonging to its boundaries. Hence the scope for overlapping regions reduces in this method [89].

The training time for CMNN may be longer at times when compared to other MNNs. However, the co-operation between modules introduced by the voting scheme, improves the efficiency of classification [89].

4.5 SUMMARY

The aim of this chapter was to demonstrate the significance of MNNs in Biological Neural Networks. Although there is much debate on issues related to the detailed architecture and operation of the neuronal clusters in the brain. Very few researchers would deny the broad concept. This, in turn, has inspired the development of artificial networks incorporating modularity, which proved useful in difficult problems.

Current models of Artificial MNNs have also been discussed in this chapter. Some of these methods are more successful than others. The trade-off in many MNNs has been the training time. Artificial MNNs, while being efficient at fine classification, take a long time to train. Moreover, due to lack of biological knowledge, most systems have failed to capture the essence of real intelligence.

Experiments were conducted on traditional neural networks and MNNs, to identify some of the fundamental problems with these methods and provide useful data for comparison with advanced networks. The details of the tests and the results obtained are presented in Chapter 5.

CHAPTER 5: EXPERIMENTS ON TRADITIONAL METHODS

5.1 INTRODUCTION

Artificial Neural Networks are capable of recognising images using simple learning algorithms such as Back Propagation. Although they exhibit some success at generalising and are tolerant to noise to a large extent, there are several problems that remain prevalent in networks using such training methods.

A review of the popular neural network architectures that are used for various applications was given in Chapter 3. The chapter also presented several problems that were identified based on available literature. Recent developments in the field of Artificial Neural Networks include the addition of modularity evident in biological neural nets. The subject of Modular Neural Networks, both biological and artificial, was discussed in Chapter 4.

This chapter presents a series of experiments conducted in the research programme. The tests were conducted for the following reasons:

- To evaluate some of the methods detailed in Chapters 3 and 4.
- To confirm some of the fundamental problems in using Artificial Neural Networks for image analysis.
- To generate benchmarks, which could be used to evaluate the performance of advanced systems.
- To understand the abilities of Artificial Neural Networks in recognising

images based on their features as compared to recognising images as a whole, which is the norm.

- To determine the effect of Modular Neural Nets in recognising such images and their features.

These issues are prevalent in robotic vision systems and hence are problems that need to be addressed to build an efficient ANS.

A representative selection of fully connected and modular networks was chosen for the tests. These networks and experiments conducted were selected to represent a good cross-section of the available methods.

The experiments conducted were:

- Experiment 1- Modular Neural Networks for Whole Images
- Experiment 2 – Artificial Neural Networks for Features of Images
- Experiment 3 – Modular Neural Networks for Features of Images
- Experiment 4 – Hardwired Networks for Features

The details of the experiments conducted and the results obtained are presented in the following sections.

5.2 MODULAR NEURAL NETWORKS FOR WHOLE IMAGES *(EXPERIMENT 1)*

Biological evidence suggests that the Visual System in both vertebrates and invertebrates is highly modular. Many Engineering Systems are easy to maintain and update because of their modular structure.

To test the capabilities and limitations of this kind of system, a Modular Neural Network was implemented based on category specific modularisation. The network implemented comprised of two independent sub-networks trained for two separate tasks.

The first network was trained for the English characters, ‘A’, ‘E’, ‘I’ and ‘O’ as shown in the Figure 5.1.

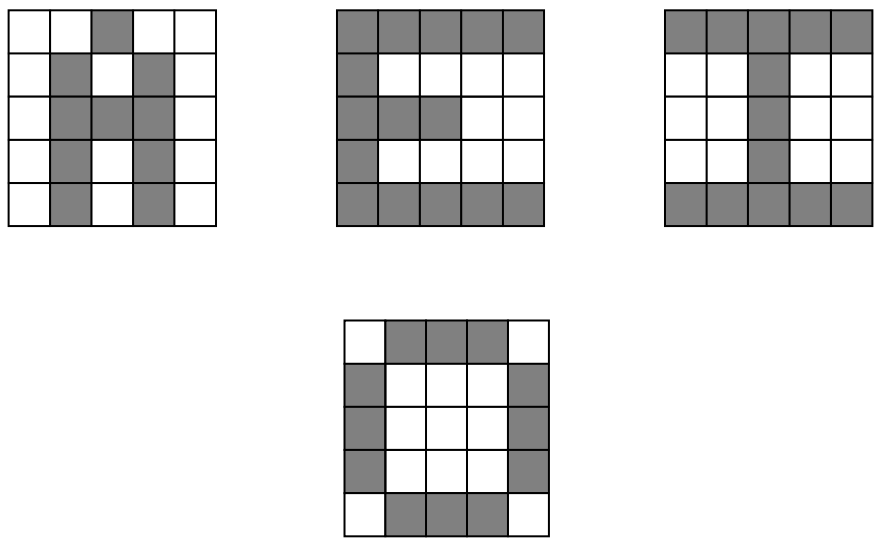


Figure 5.1: Test Patterns – English Letters

The geometrical shapes as shown in Figure 5.2, formed the training data for the second network. The images included simple shapes such as a Square, Triangle, Rectangle and Circle.

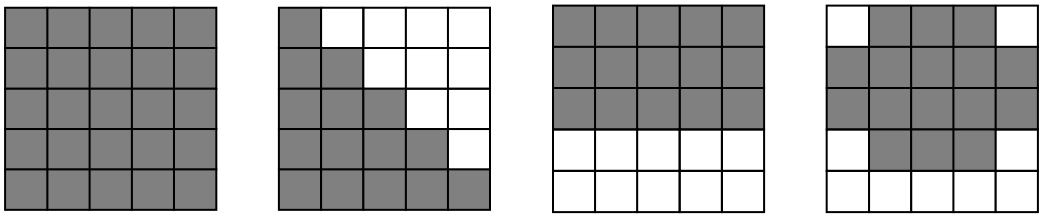


Figure 5.2: Test Patterns – Geometrical Shapes

The characters were line patterns and the shapes were filled (solid) patterns. This step was taken to ensure that the two sets of patterns were distinct from each other. An illustration of the solid and line patterns is given in Figure 5.3.

Figure 5.3: Illustration of a "Solid Pattern" and a "Line Pattern"

As the main aim was to analyse the capabilities of Artificial Neural Networks for image recognition in general, simple images such as English characters and shapes were chosen. These images are generally used for testing Neural Networks in the research community and so will serve the purpose of the experiments as mentioned in Section 5.1. The images were also ideal to test the capabilities of Artificial Neural Networks in recognising features as discussed in Section 5.3. The features used in experiments 2, 3 and 4 are the features that make up the letters E, I, O, U.

The images that formed the data set allowed ease of manipulation and extraction of features. Also, the time taken for training simple images is relatively less than training complex images. Simplicity was based on the size of the image, the type of the image and the number of important features that formed the image.

5.2.1 LETTER RECOGNITION

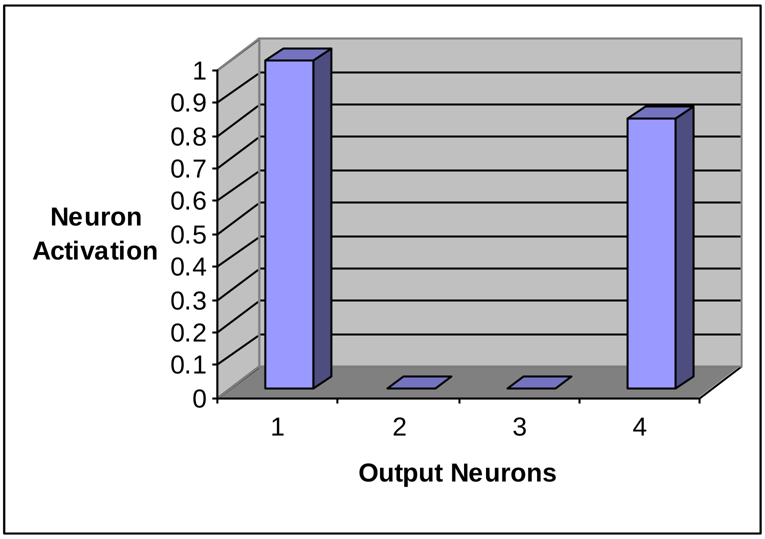
A fully connected ANN was trained to recognise the letters ‘A’, ‘E’, ‘I’ and ‘O’. This section gives details of the training and results obtained.

- *Network Type:* Fully connected Feed Forward Network
- *Network Architecture:* 25 Input Neurons, 9 Hidden Neurons and 4 Output Neurons
- *Neuron Model:* Perceptron with Sigmoid Activation Function
- *Training Method:* Back Propagation
- *Training Data:* Four Letters ‘A’, ‘E’, ‘I’, ‘O’
- *Image Size:* 5x5 Pixels
- *Implementation:* Software using C++
- *Average Mean Squared Error Achieved:* 0.1
- *Number of Iterations:* 16775

- *Training Time:* 1 hr and 05 minutes on a PC with a 933 MHz Pentium III processor
- *Recognition Criteria:* The output neuron (allocated for a certain letter) with the highest activation

The network was successfully trained and tested. The results obtained are illustrated in the following Graphs 5.1-5.4.

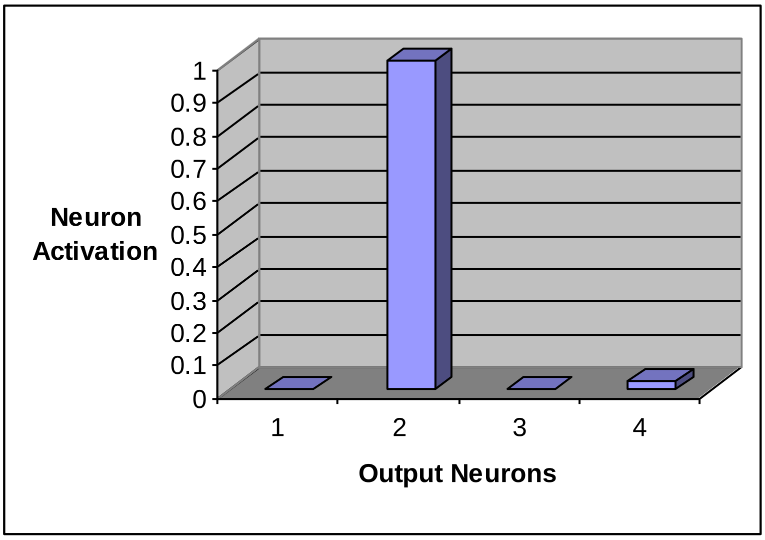
Test for ‘A’



Graph 5.1: Results obtained when tested for letter ‘A’ Output Neuron 1 is allocated for ‘A’, Neuron 2 for ‘E’ and Neuron 3 for ‘I’ and Neuron 4 for ‘O’

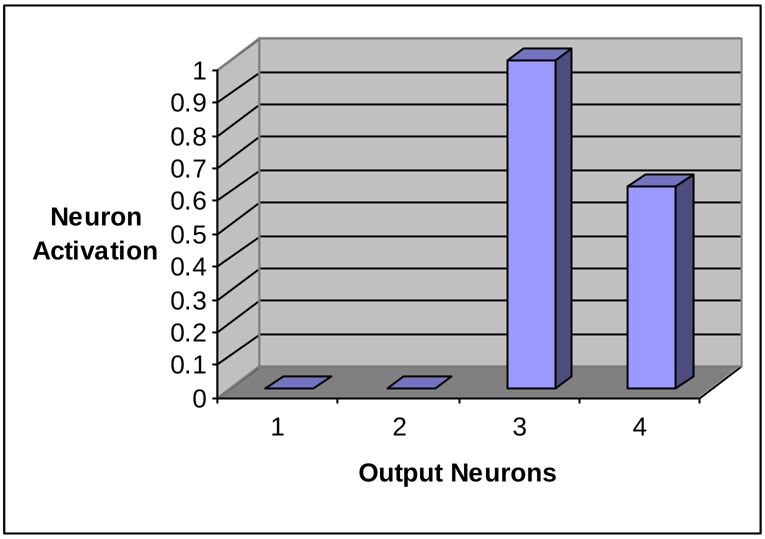
In the Graph 5.1, Neuron 1 allocated for ‘A’ shows the highest activation, although Neuron 4 has an activation >0.4.

Test for 'E'



**Graph 5.2: Results obtained when tested for letter 'E'
Output Neuron 1 is allocated for 'A', Neuron 2 for 'E' and
Neuron 3 for 'I' and Neuron 4 for 'O'**

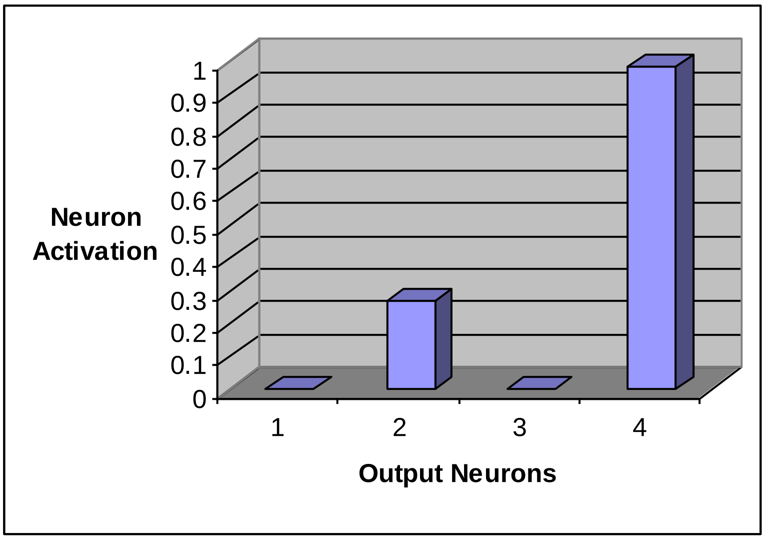
Test for 'I'



**Graph 5.3: Results obtained when tested for letter 'I'
Output Neuron 1 is allocated for 'A', Neuron 2 for 'E' and
Neuron 3 for 'I' and Neuron 4 for 'O'**

Graphs 5.2 and 5.3 clearly show the successful recognition of 'E' and 'I' by the highest activation of the corresponding allocated neurons 2 and 3 respectively.

Test for ‘O’



Graph 5.4: Results obtained when tested for letter ‘O’
Output Neuron 1 is allocated for ‘A’, Neuron 2 for ‘E’ and Neuron 3 for ‘I’ and Neuron 4 for ‘O’

Graph 5.4 shows that the Letter ‘O’ is recognised as the allocated neuron 4 has the highest activation. Hence the ANN trained for all 4 letters ‘A’, ‘E’, ‘I’ and ‘O’ has successfully recognised all patterns.

5.2.2 SHAPE RECOGNITION

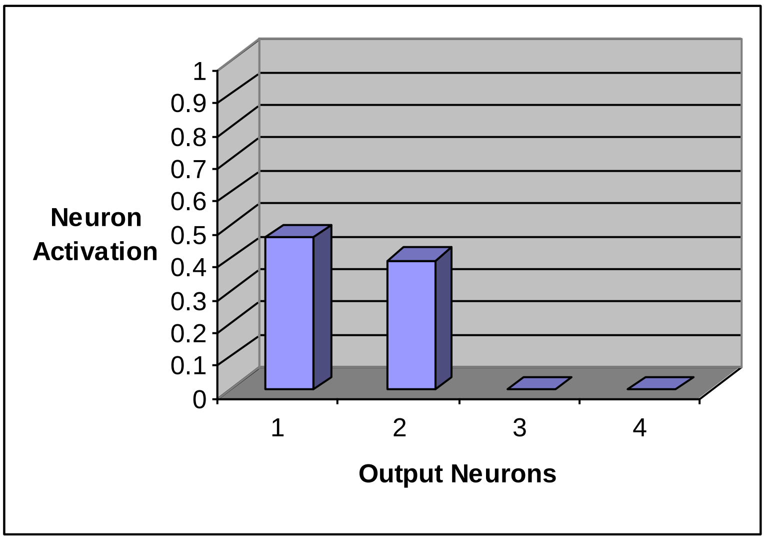
A second fully connected ANN was trained to recognise four shapes that resembled a Triangle, a Square, a Rectangle and a Circle. The details of the training are given below.

- *Network Type:* Fully connected Feed Forward Network
- *Network Architecture:* 25 Input Neurons, 9 Hidden Neurons and 4 Output Neurons

- *Neuron Model:* Perceptron with Sigmoid Activation Function
- *Training Method:* Back Propagation
- *Training Data:* Four Shapes 'Triangle', 'Square', 'Rectangle', 'Circle'
- *Image Size:* 5x5 Pixels
- *Implementation:* Software using C++
- *Mean Squared Error Achieved:* 0.03
- *Number of Iterations:* 14073
- *Training Time:* 1 hour and 10 minutes on a PC with a 933 MHz Pentium III processor
- *Recognition Criteria:* The output neuron (allocated for a certain shape) with the highest activation

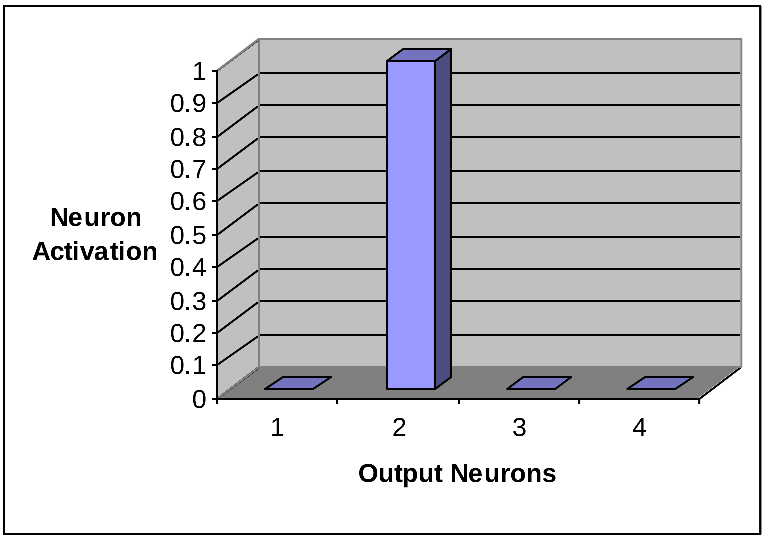
The Graphs 5.5-5.8 show the results obtained when tested for the four shapes.

Test for Triangle



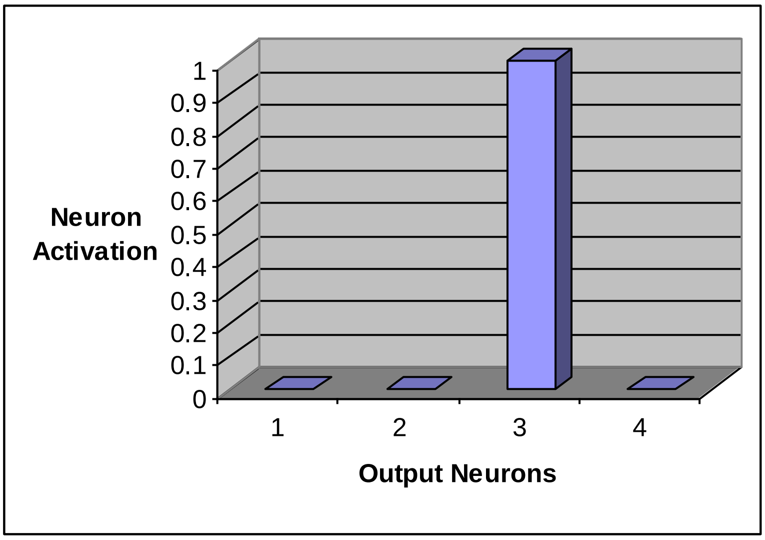
Graph 5.5: Results obtained when tested for 'Triangle'
Output Neuron 1 is allocated for 'Triangle', Neuron 2 for 'Square' and Neuron 3 for 'Circle' and Neuron 4 for 'Rectangle'

Test for Square



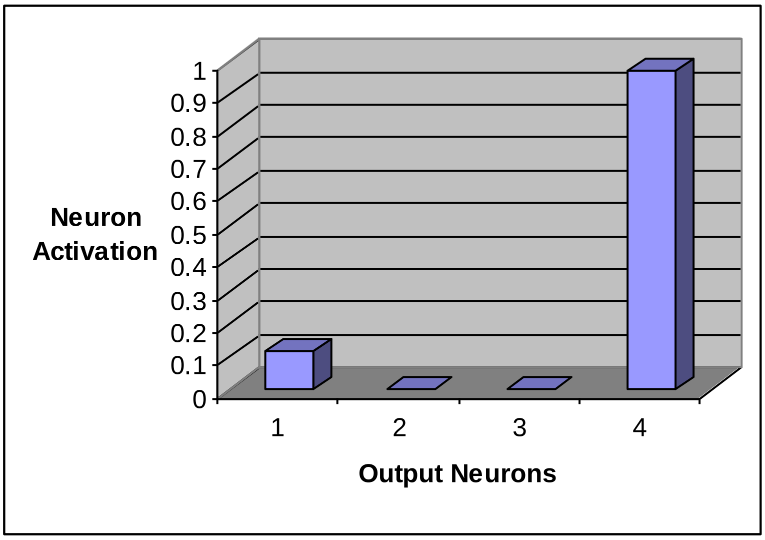
Graph 5.6: Results obtained when tested for 'Square'
Output Neuron 1 is allocated for 'Triangle', Neuron 2 for 'Square' and Neuron 3 for 'Circle' and Neuron 4 for 'Rectangle'

Test for Circle



Graph 5.7: Results obtained when tested for 'Circle'
Output Neuron 1 is allocated for 'Triangle', Neuron 2 for 'Square' and Neuron 3 for 'Circle' and Neuron 4 for 'Rectangle'

Test for Rectangle



**Graph 5.8: Results obtained when tested for 'Rectangle'
Output Neuron 1 is allocated for 'Triangle', Neuron 2 for 'Square'
and Neuron 3 for 'Circle' and Neuron 4 for 'Rectangle'**

Graphs 5.5-5.8 show that the network has recognised all 4 shapes that it was trained for by the highest activation of the corresponding allocated neurons. In Graph 5.5, the activation of Neuron 1 is <0.5; nevertheless, it is the most highly activated neuron among the output neurons, which is the criteria for recognition. Hence, all shapes were successfully recognised by the specialist ANN trained to recognise shapes.

5.2.3 MODULAR NEURAL NETWORK

Two networks were trained individually to recognise Alphabets and Shapes as discussed in the Sections 5.2.1 and 5.2.3. The individual networks learnt to recognise the patterns and there was a 100% success rate in case of both the networks.

The two networks trained for recognising alphabets and shapes were used to implement the MNN as shown in Figure 5.4. There were no sideways connections between networks. The model here was based on popular theories that one neuron or a class of neurons is responsible for recognising one pattern.

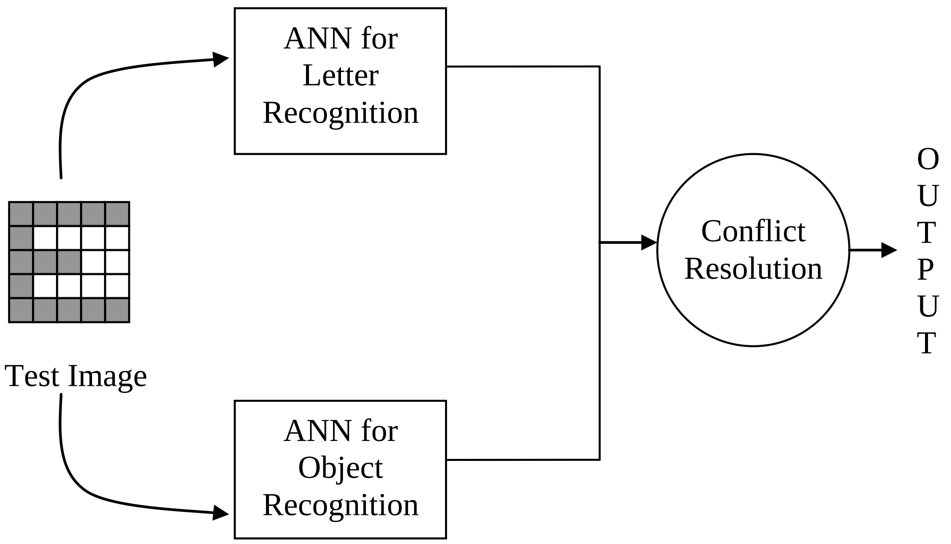


Figure 5.4: A Simple Modular Neural Network

The conflict resolution is used in the MNN for differentiation between Letters and Shapes. It is a decision made according to the most highly activated neuron.

The algorithm of the procedure used to test the MNN with images is as given below.

Input the test image to both of the networks that were each trained individually for recognising Alphabets and Shapes

↓

Find the activation of the output neurons of each network

↓

Pass the information of the output neuron activation to the Conflict Resolution module

↓

At the Conflict Resolution module check the neuron with the highest activation

↓

Declare the neuron with the highest activation as the winner

The results, as tabulated in Table 5.1, showed only 62.5% success rate in recognising the patterns when the modular network was used.

A network trained for a certain task failed to respond and give the appropriate output in 37.5% of the cases. Also, among the patterns that have been recognised, 50% of the cases showed that the Modular Neural Network almost

Test For	Activation of Neurons Corresponding to							
	A	E	I	O	Triangle	Square	Circle	Rectangle
A	0.997	0.000	0.000	0.820	0.000	0.000	0.981	0.000
E	0.000	0.999	0.000	0.024	0.000	0.997	0.000	0.000
I	0.000	0.000	0.998	0.617	0.000	0.999	0.000	0.015
O	0.000	0.266	0.000	0.983	0.000	0.066	0.003	0.000
Triangle	0.911	0.000	0.002	0.599	0.465	0.393	0.000	0.000
Square	0.000	0.001	0.003	0.979	0.000	0.998	0.001	0.002
Circle	0.000	0.000	0.000	0.999	0.000	0.000	0.998	0.000
Rectangle	0.000	0.745	0.896	0.186	0.115	0.001	0.000	0.968

Table 5.1: Activation Value of Output Neurons of MNN Corresponding to Alphabets and Shapes

categorised them as the rival or opposite class (the difference between the output activation values being very small).

It can be noted that a network that was not trained for a particular pattern had the highest activation in many cases. Hence, the results tend to disprove the

Grandmother Cell Theory, which states that one neuron or a specific group of neurons in the human brain are activated for any specific set of features [90].

The conclusions that can be drawn from the results are that a more robust conflict resolution procedure and parallel information flow between the two modules is necessary to solve the problems with classification using Modular Neural Nets, although adding such a conflict resolution system may complicate the network considerably.

5.3 ANNs FOR FEATURE DETECTION *(EXPERIMENT 2)*

Humans find that the features, which make up a pattern, are important in recognising it as a particular object. Certain features put together make up an object or a pattern. For example, the basic features that make up a face are two eyes, one nose, one mouth and two ears. The features that make up a table are a flat surface supported by one, three or four legs. A cube is made up of 6 square surfaces joined at the edges to make up a solid block. However, incomplete images can nevertheless be recognised as a certain image based on the few existing features.

This section includes a discussion of the training and tests carried out to identify the feature detecting abilities of fully connected ANNs trained using Back Propagation. The features selected for training the network are given in Figure 5.5.

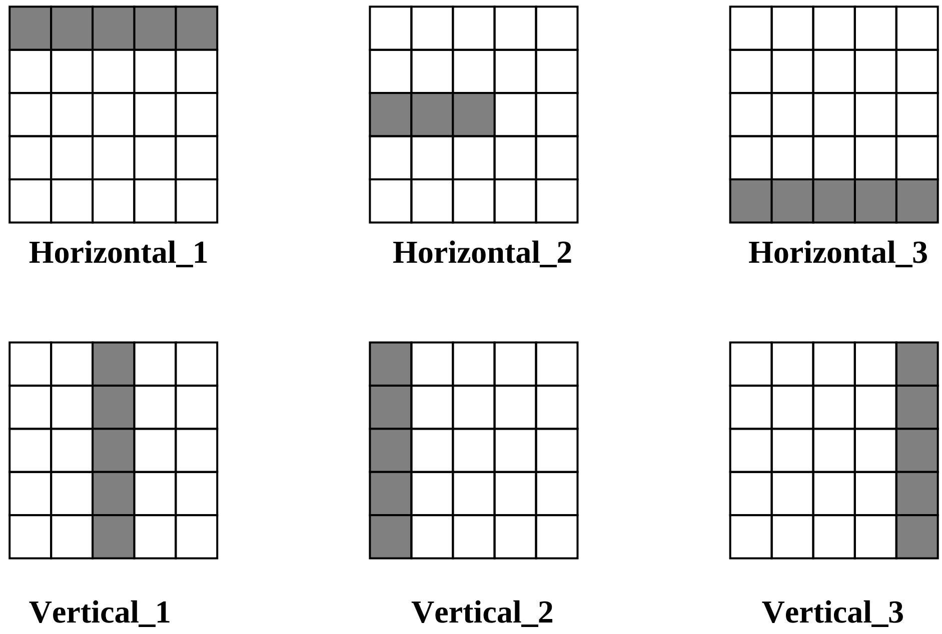


Figure 5.5: Features comprising the Training Data

5.3.1 TRAINING STATISTICS

The main aspects of the network training are:

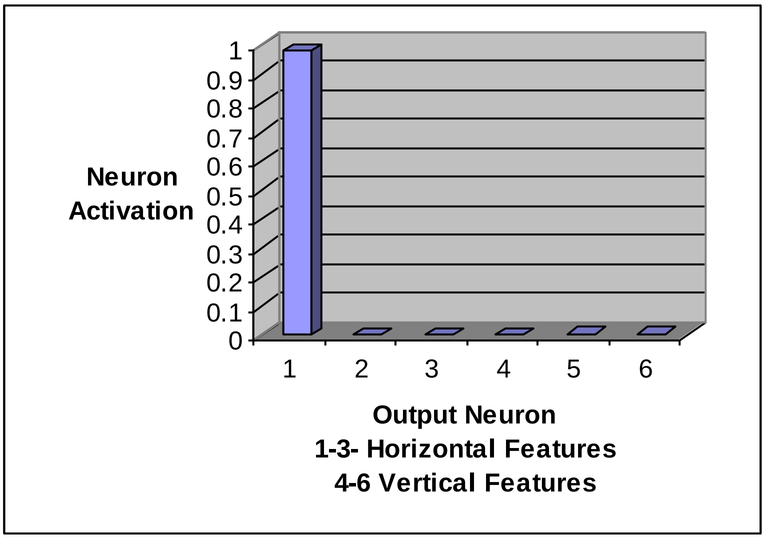
- *Network Type:* Fully-connected Feed Forward Network
- *Network Architecture:* 25 Input Neurons, 9 Hidden Neurons and 6 Output Neurons
- *Neuron Model:* Perceptron with Sigmoid Activation Function
- *Training Method:* Back Propagation
- *Training Data:* Six Features were selected. Three horizontal features H_1, H_2 and H_3 and three vertical features V_1, V_2 and V_3.
- *Image Size:* 5x5 Pixels
- *Implementation:* Software using C++

- *Mean Squared Error Achieved:* 0.02
- *Number of Iterations:* 18125
- *Training Time:* 1 hour and 15 minutes on a PC with a 933 MHz Pentium III processor
- *Recognition Criteria:* The output neuron (allocated for a certain feature) with the highest activation

5.3.2 TEST WITH TRAINING DATA

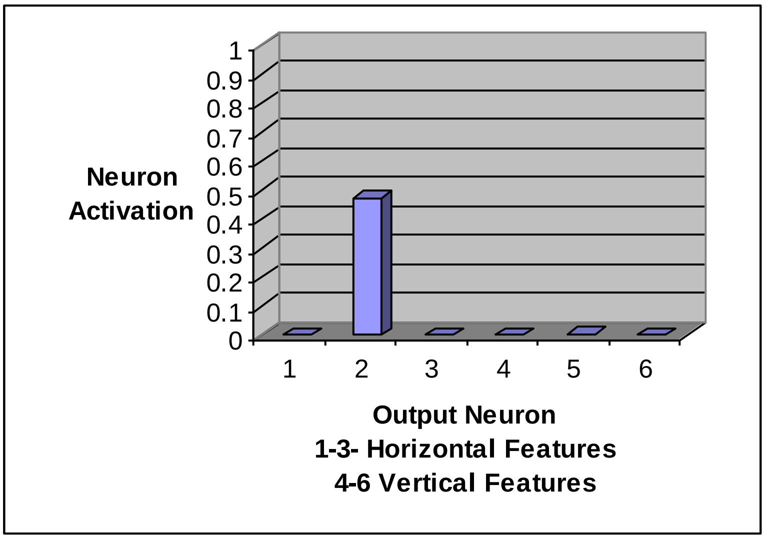
The network was successfully trained and tested for all the features as shown in the Graphs 5.9-5.14. This can be noted by the high activation of the corresponding allocated neurons for the six features.

Test for Horizontal_1 (H_1) Feature



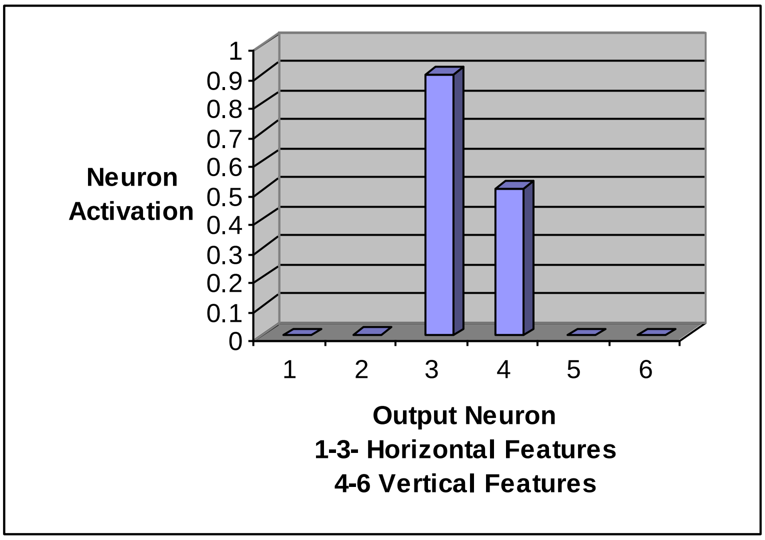
Graph 5.9: Test for H_1. Illustration of Output Neuron Activation.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for Horizontal_2 (H_2) Feature



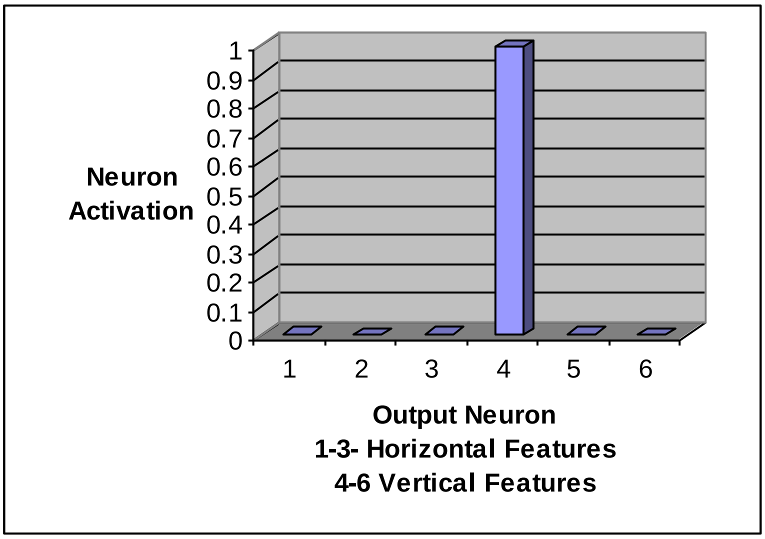
Graph 5.10: Test for H_2. Illustration of Output Neuron Activation. Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3. Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for Horizontal_3 (H_3) Feature



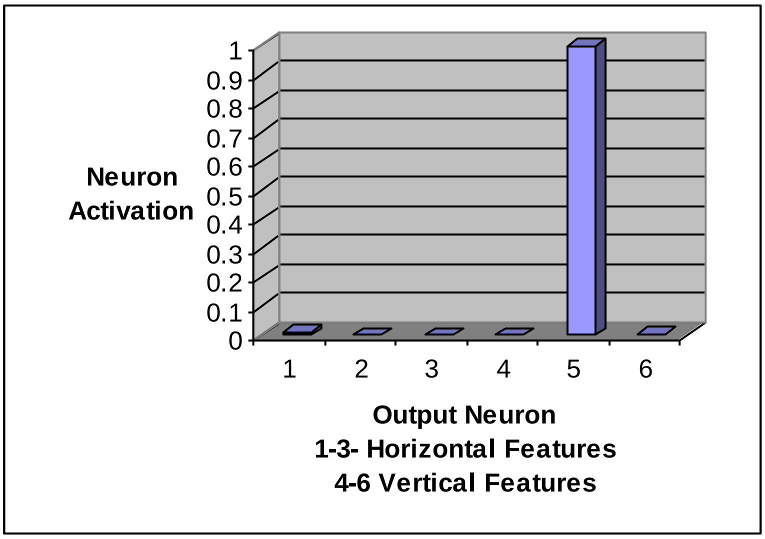
Graph 5.11: Test for H_3. Illustration of Output Neuron Activation. Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3. Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for Vertical_1 (V_1) Feature



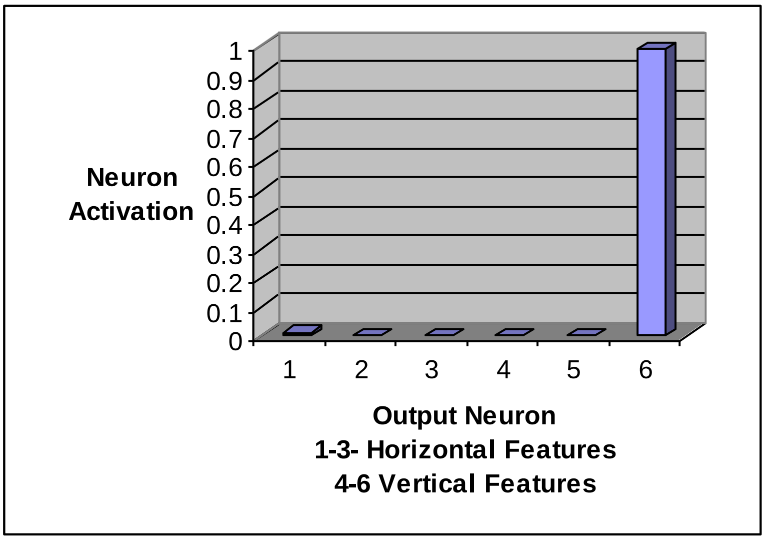
Graph 5.12: Test for V_1. Illustration of Output Neuron Activation. Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3. Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for Vertical_2 (V_2) Feature



Graph 5.13: Test for V_2. Illustration of Output Neuron Activation. Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3. Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for Vertical_3 (V_3) Feature



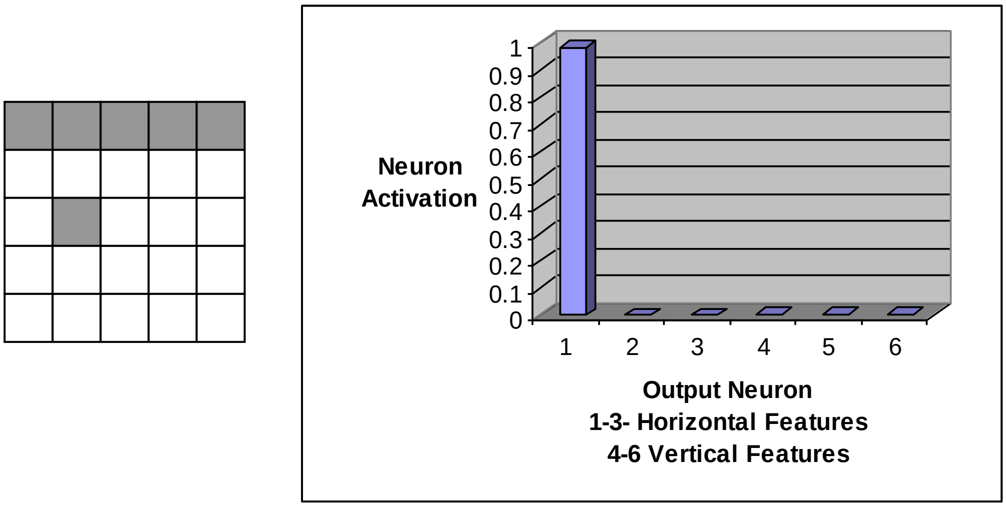
Graph 5.14: Test for V_3. Illustration of Output Neuron Activation. Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3. Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

5.3.3 TEST WITH NOISY_1 FEATURES

The network was tested with noisy images. Addition of noise in ANNs usually involves adding a small percentage of random noise to every pixel of the image. Here noise is added in terms of speckles. A pixel is changed from a 1 to 0.

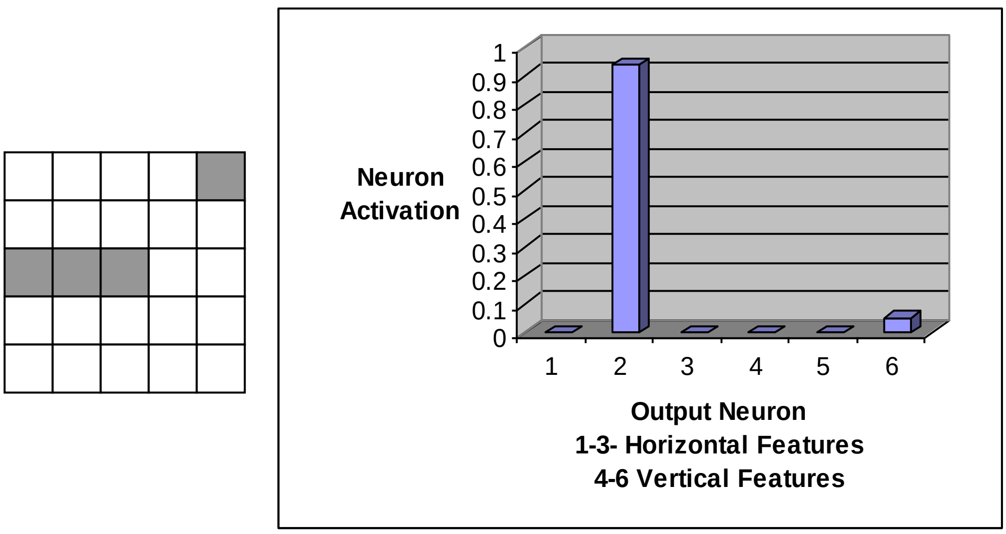
The Graphs 5.15-5.26 illustrates the results obtained for the corresponding noisy features. The criterion for recognition is the neuron with the highest activation. The ANN was successful at recognizing all noisy features except H_3 and V_3 as can be seen in Graphs 5.17 and 5.20 respectively. Hence the success rate of the network in recognizing 1st stage of noisy (Noisy_1) features is 66.67%.

Test for Noisy_1 H_1 Feature



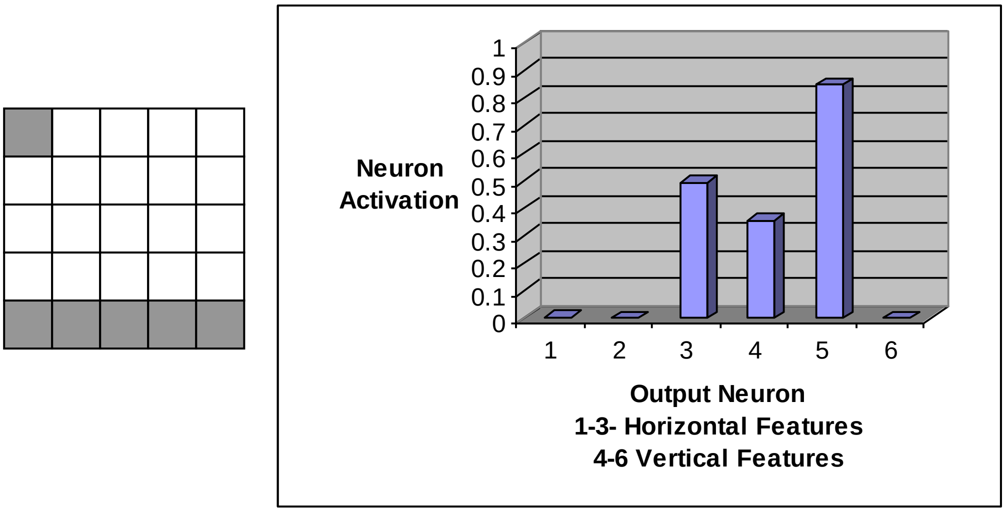
Graph 5.15: Illustration of the Output Neuron Activation when tested for Noisy_1 H_1 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for Noisy_1 H_2 Feature



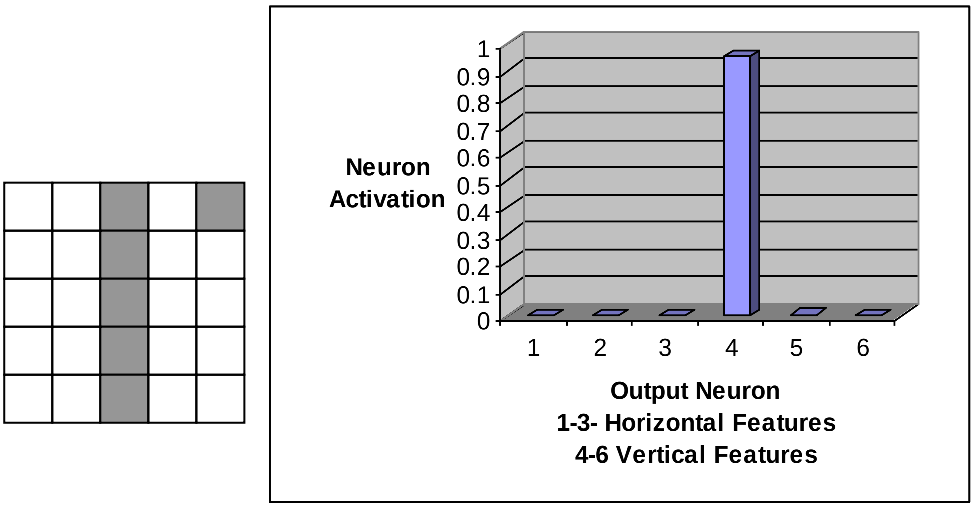
Graph 5.16: Illustration of the Output Neuron Activation when tested for Noisy_1 H_2 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for Noisy_1 H_3 Feature



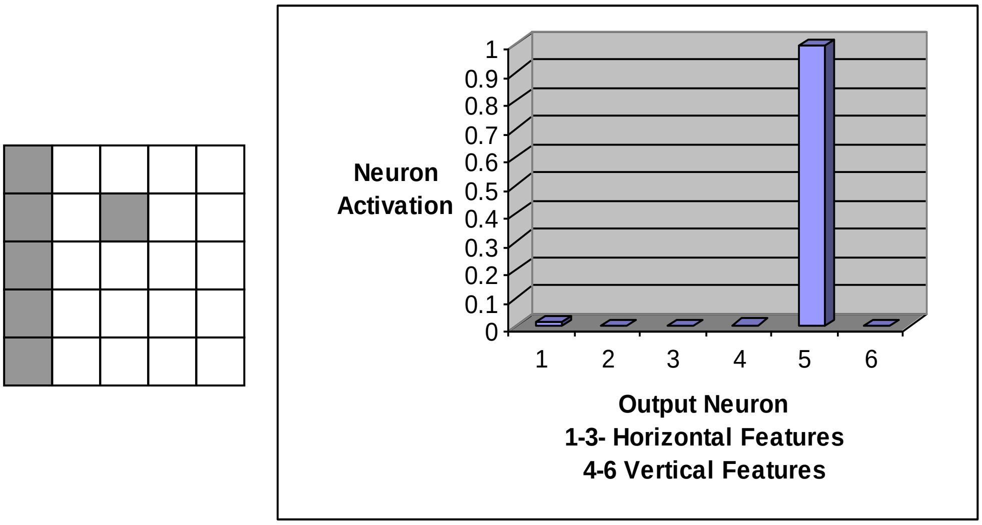
Graph 5.17: Illustration of the Output Neuron Activation when tested for Noisy_1 H_3 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for Noisy_1 V_1 Feature



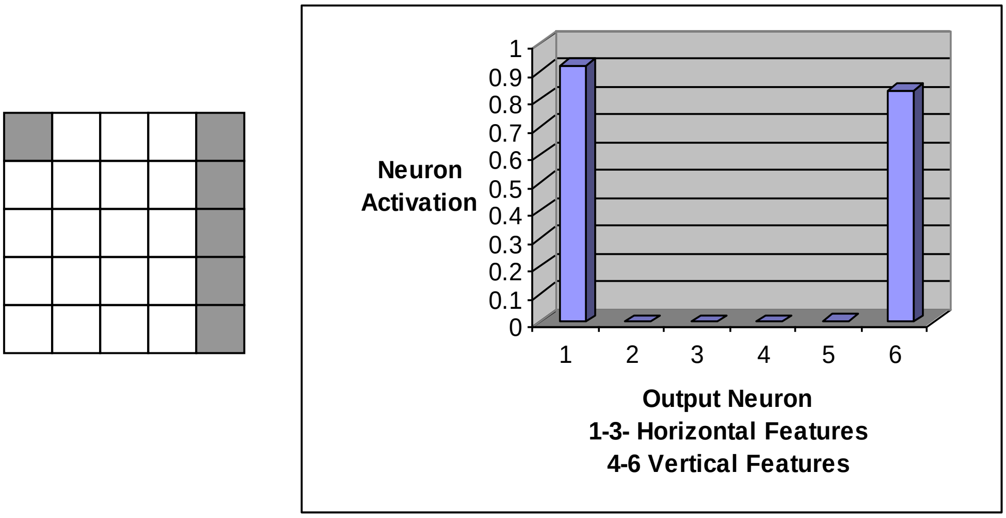
Graph 5.18: Illustration of the Output Neuron Activation when tested for Noisy_1 V_1 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for Noisy_1 V_2 Feature



Graph 5.19: Illustration of the Output Neuron Activation when tested for Noisy_1 V_2 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for Noisy_1 V_3 Feature

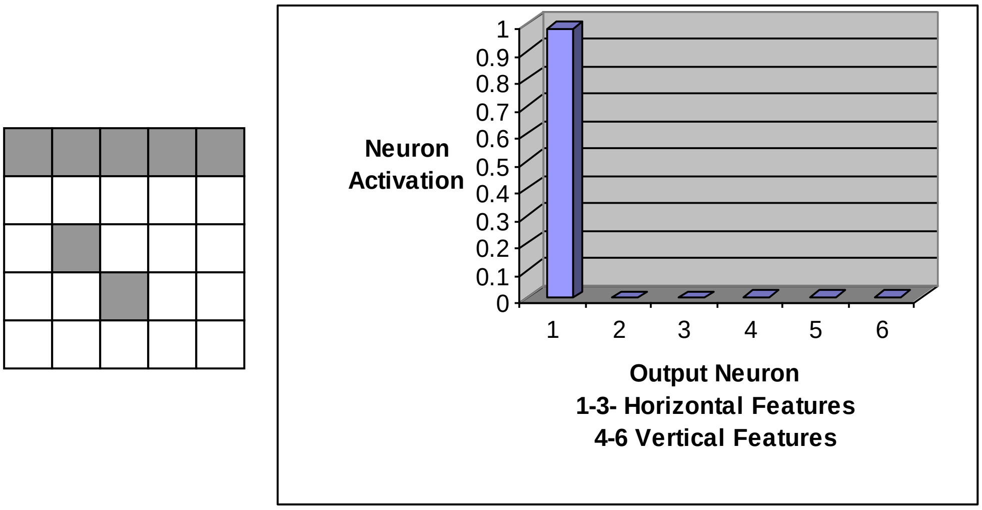


Graph 5.20: Illustartion of the Output Neuron Activation when tested for Noisy_1 V_3 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

5.3.4 TEST WITH NOISY_2 DATA

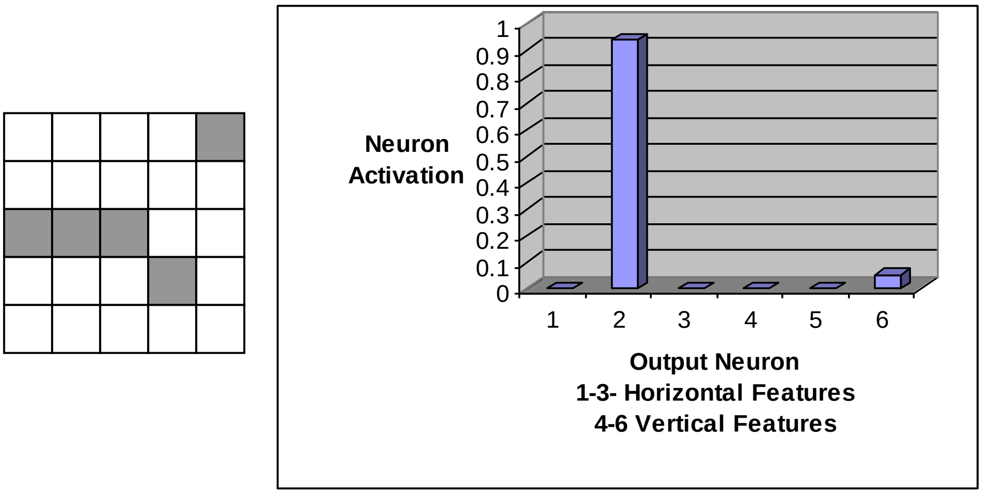
Adding two speckles in the original images of features increased the level of noise. The Graphs 5.21-5.26 illustrates the results obtained.

Test for Noisy_2 H_1 Feature



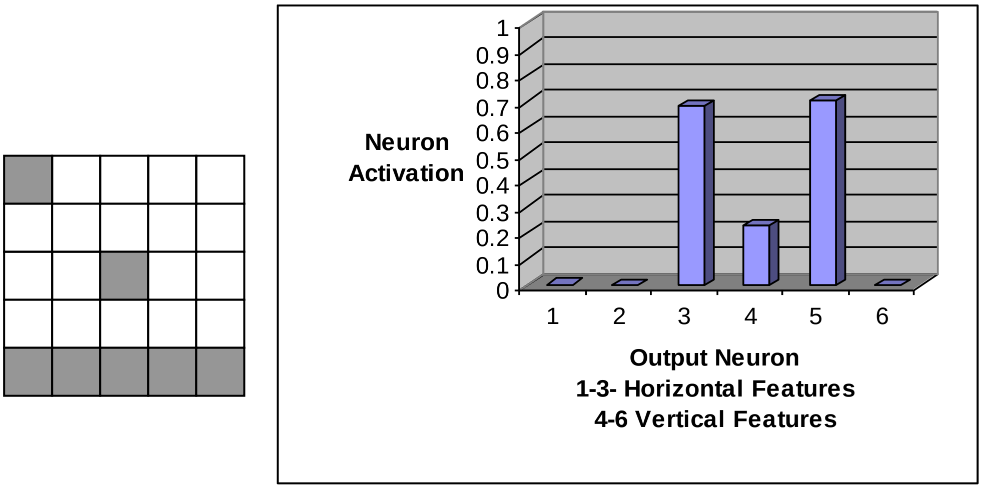
Graph 5.21: Illustration of the Output Neuron Activation when tested for Noisy_2 H_1 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for Noisy_2 H_2 Feature



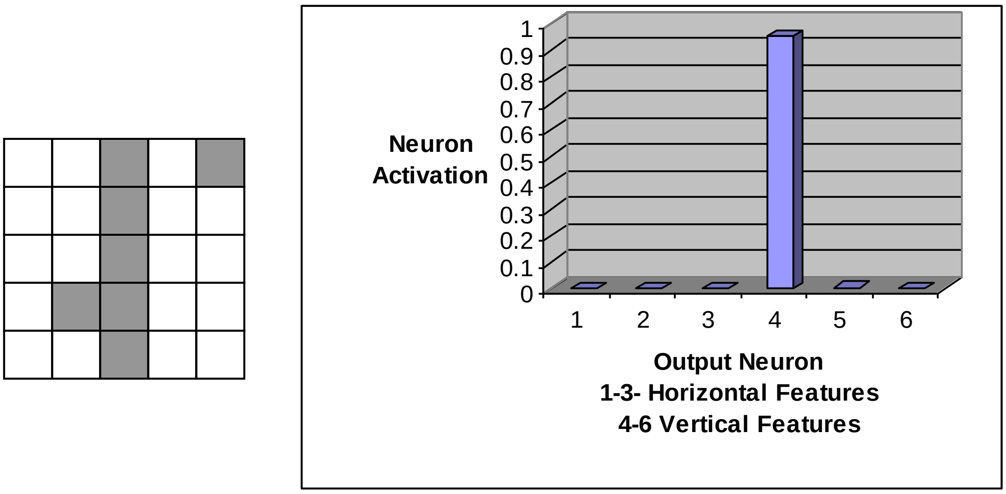
Graph 5.22: Illustration of the Output Neuron Activation when tested for Noisy_2 H_2 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for Noisy_2 H_3 Feature



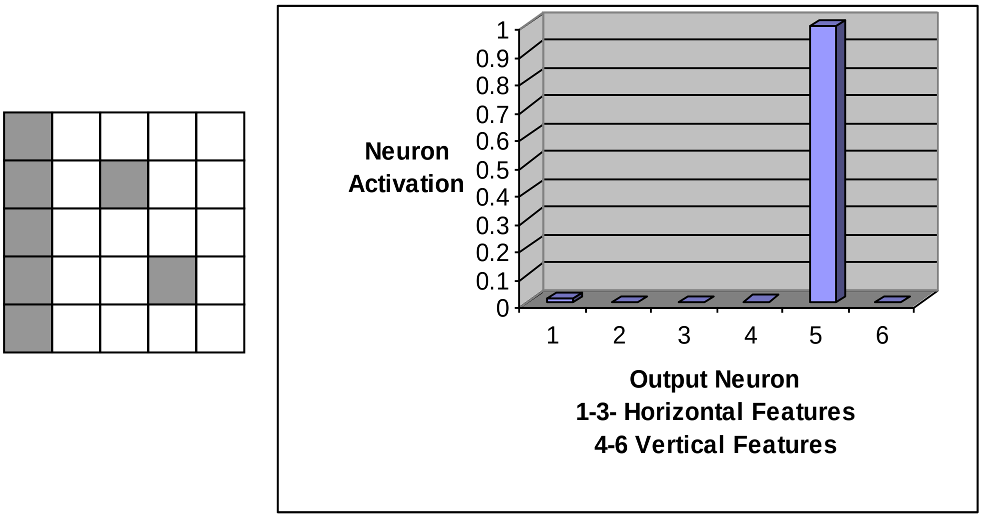
Graph 5.23: Illustration of the Output Neuron Activation when tested for Noisy_2 H_3 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for Noisy_2 V_1 Feature



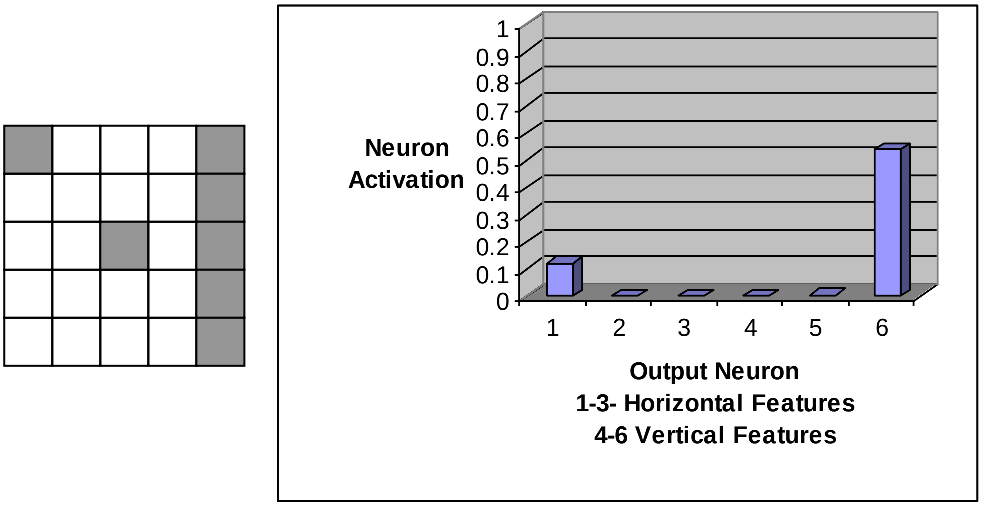
Graph 5.24: Illustration of the Output Neuron Activation when tested for Noisy_2 V_1 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for Noisy_2 V_2 Feature



Graph 5.25: Illustration of the Output Neuron Activation when tested for Noisy_2 V_2 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for Noisy_2 V_3 Feature



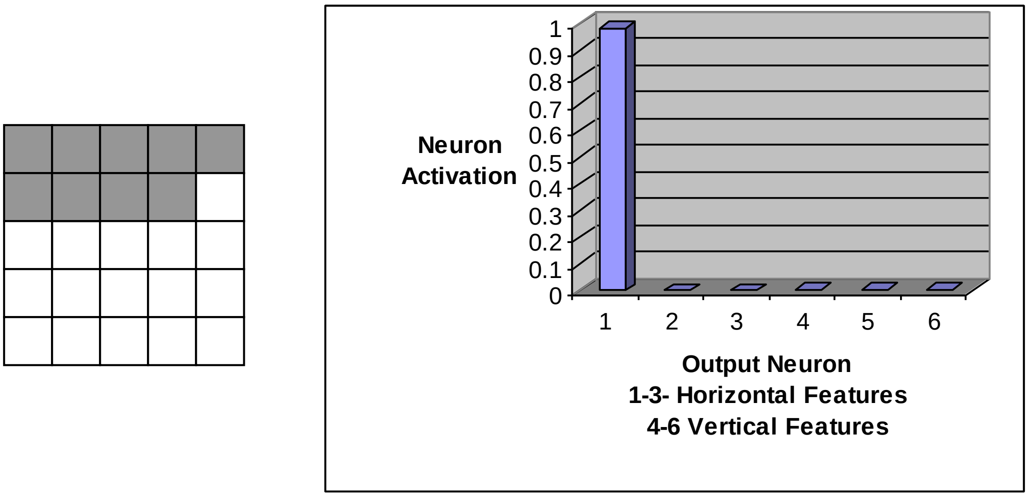
Graph 5.26: Illustration of the Output Neuron Activation when tested for Noisy_2 V_3 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

All features were successfully recognized except H_3 (Graph 5.23). Hence the success rate of the ANN is 83.33%.

5.3.5 TESTS WITH NEW FEATURES

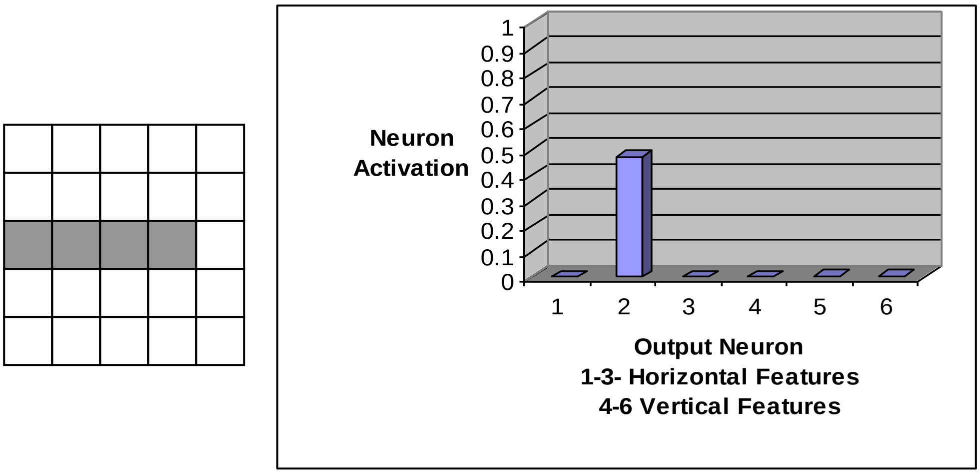
The Network was tested for new features that resembled the original features in the training set. The following Graphs 5.27-5.32 illustrate the results obtained.

Test with New Feature H_1



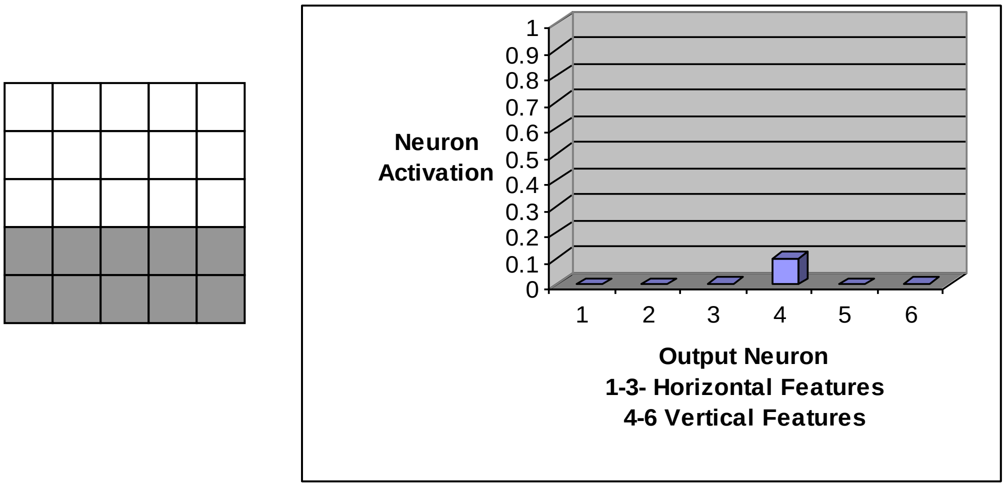
Graph 5.27: Illustration of the Output Neuron Activation when tested for New H_1 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test with New Feature H_2



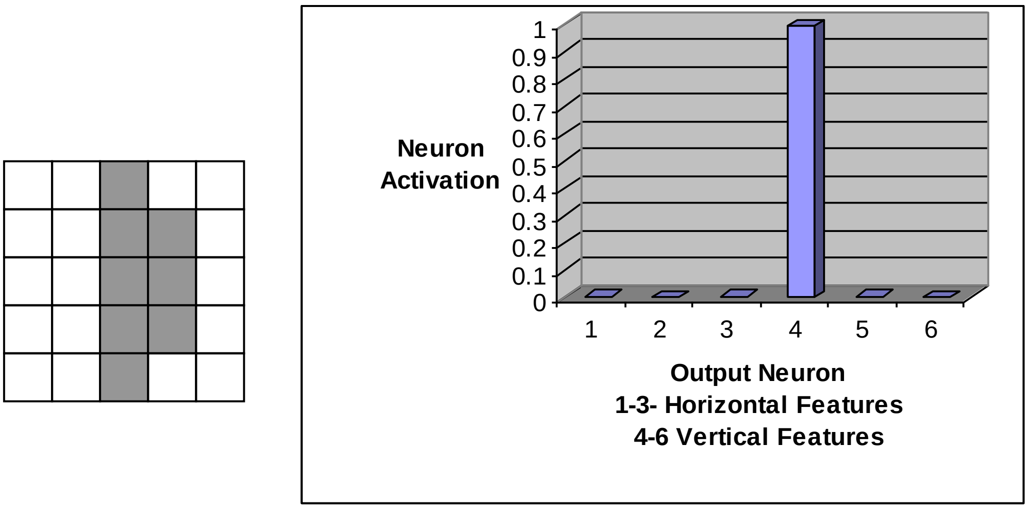
Graph 5.28: Illustration of the Output Neuron Activation when tested for New H_2 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test with New Feature H_3



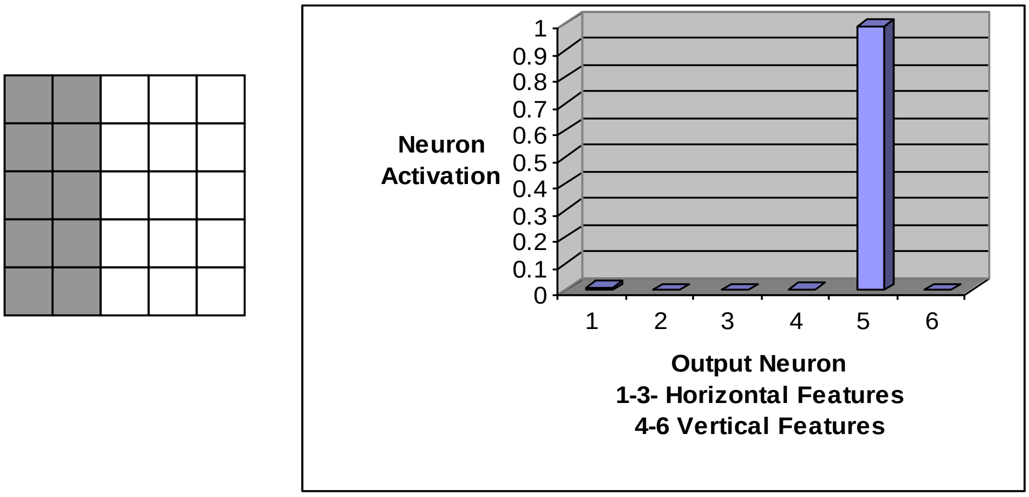
Graph 5.29: Illustration of the Output Neuron Activation when tested for New H_3 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test with New Feature V_1



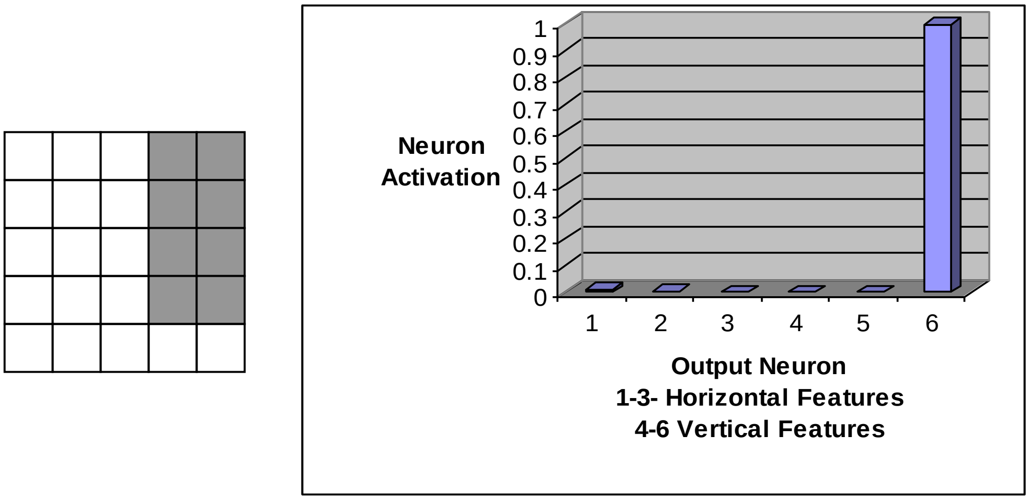
Graph 5.30: Illustration of the Output Neuron Activation when tested for New V_1 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test with New Feature V_2



Graph 5.31: Illustration of the Output Neuron Activation when tested for New V_2 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test with New Feature V_3



Graph 5.32: Illustration of the Output Neuron Activation when tested for New V_3 Feature.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

The network had a success rate of 83.33%. The only unsuccessful case was the test with H_3 feature as shown in Graph 5.29.

5.3.6 TEST WITH COMBINATION OF FEATURES

Six basic features are required to create characters, ‘E’, ‘I’, ‘O’ and ‘U’ as shown in Figure 5.1. The network was tested for these letters, which are combinations of the features that the network was trained for.

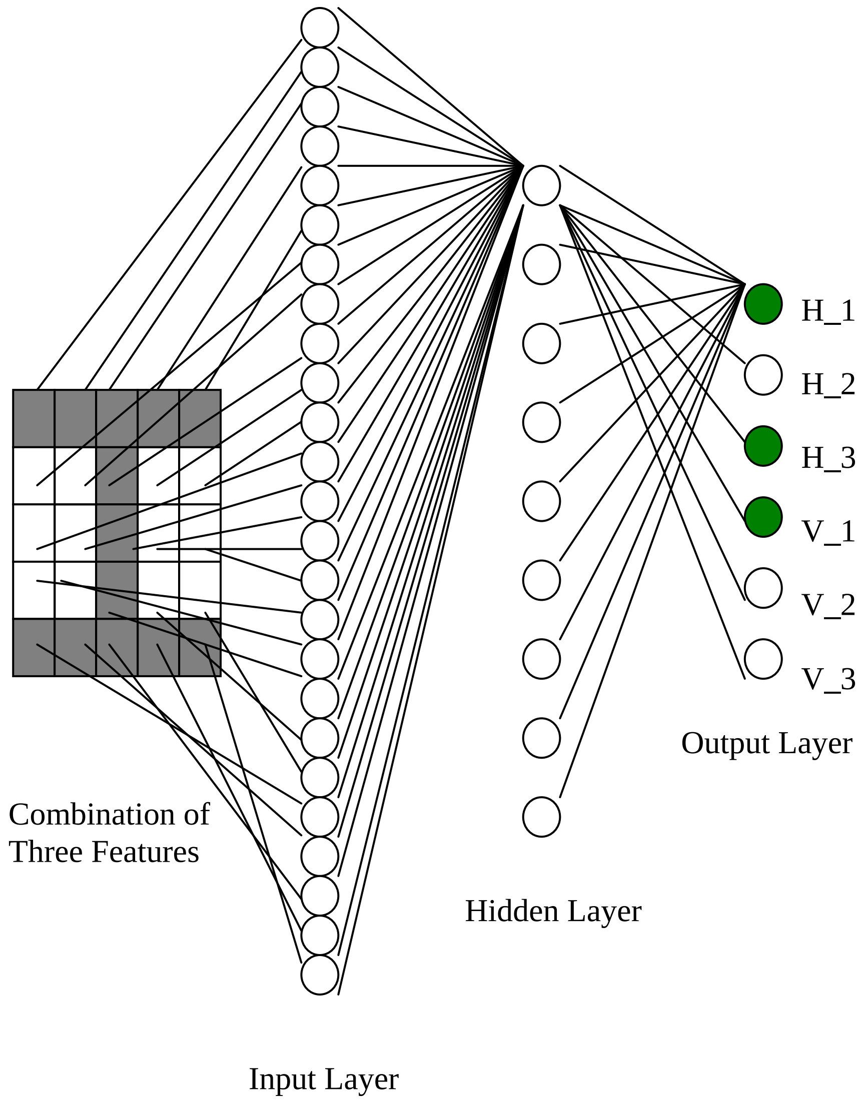


Figure 5.6: Test for Feature Detection using a Back Propagation Network

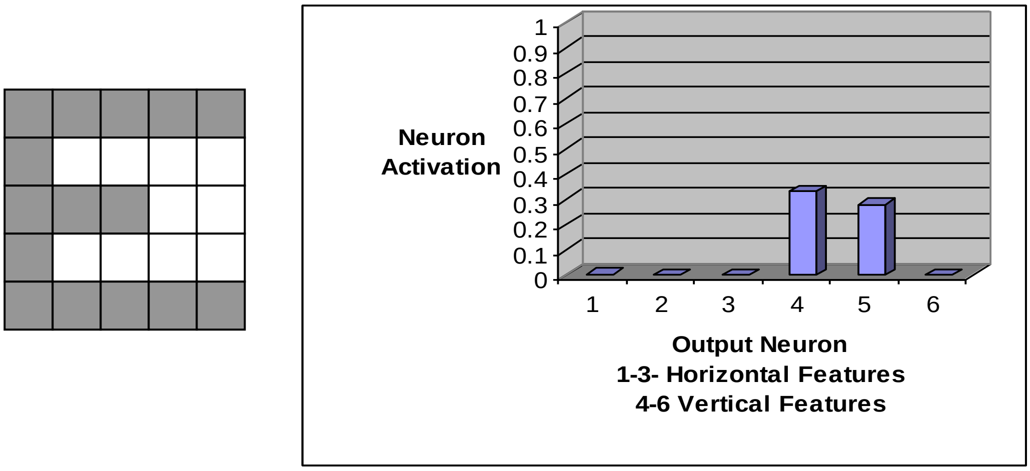
The Figure 5.6 illustrates a test image 'I' being tested for the presence of features, H_1, H_3 and V_1. It shows the connections between a limited numbers of neurons, although the network is fully connected. The six output neurons correspond to the 6 features that the network was trained for.

Hence, when the letter 'I' is tested, the output neurons corresponding to Horizontal_1, Horizontal_2 and Vertical_1 are expected to be fired, i.e. have a very high activation value. However, the network completely failed to recognise any of the features as shown in Table 5.2 and Graphs 5.33-5.36. The reason for its failure is that any combination of, albeit, known features to the network are treated as new images by the network when presented with a combination of known features. Hence, Back Propagation Networks are only good at recognising whole images, but fail very badly at feature extraction, i.e. recognising combination of trained or known features that make up an image.

Test For	**Activation of Output Neurons Corresponding to**					
	H_1	H_2	H_3	V_1	V_2	V_3
E	0.002	0.000	0.000	0.331	0.277	0.000
I	0.002	0.000	0.000	0.345	0.319	0.000
O	0.409	0.000	0.000	0.002	0.004	0.001
U	0.000	0.000	0.000	0.003	0.035	0.336

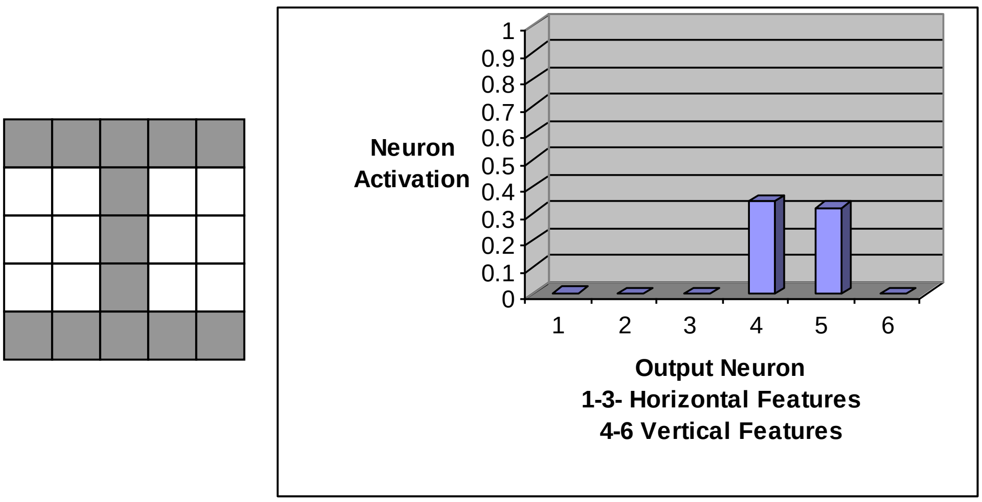
Table 5.2: Activation Values of Specialised Neurons Trained for Horizontal and Vertical Features and Tested for Alphabets

Test for ‘E’



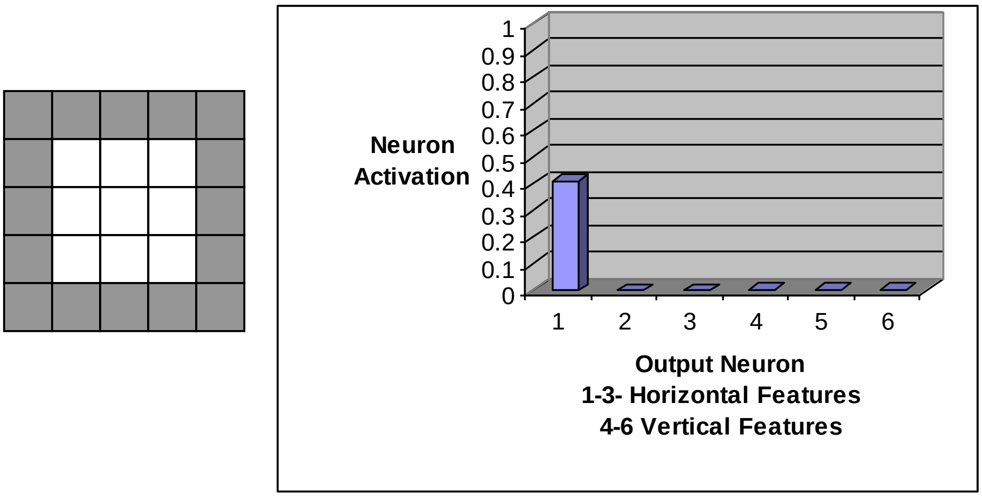
Graph 5.33: Illustration of the Output Neuron Activation when tested for ‘E’.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for ‘I’



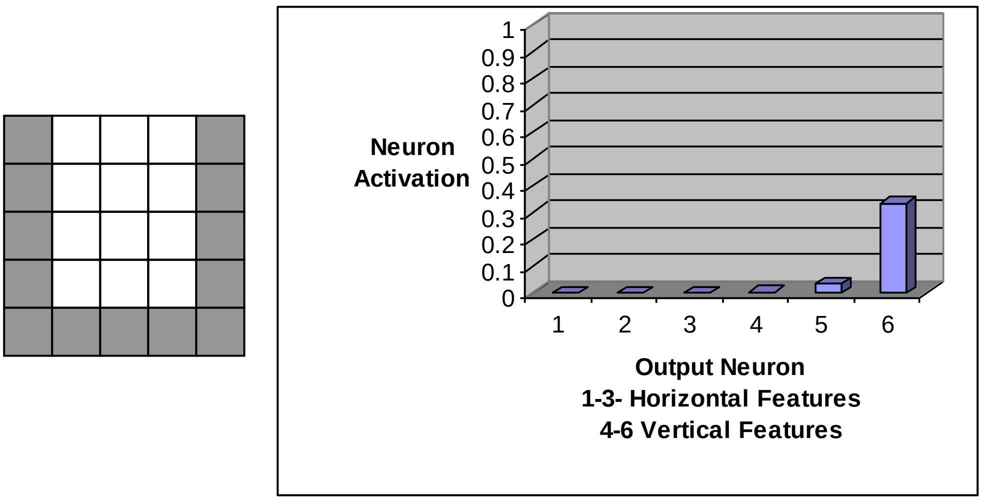
Graph 5.34: Illustration of the Output Neuron Activation when tested for ‘I’.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for ‘O’



Graph 5.35: Illustration of the Output Neuron Activation when tested for ‘O’.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for ‘U’



Graph 5.36: Illustration of the Output Neuron Activation when tested for ‘U’.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

The idea behind these experiments was that when a pattern such as, 'H' is tested, the network that was trained to recognise the horizontal and vertical features should be able to recognize these features. On the contrary, the network was in a state of confusion. This is evident from the results shown above in Table 5.2 and the Graphs 5.33-5.36.

5.4 MODULAR FEATURE DETECTOR *(EXPERIMENT 3)*

The second method involved exploring the suitability of Modular Neural Networks in recognising features.

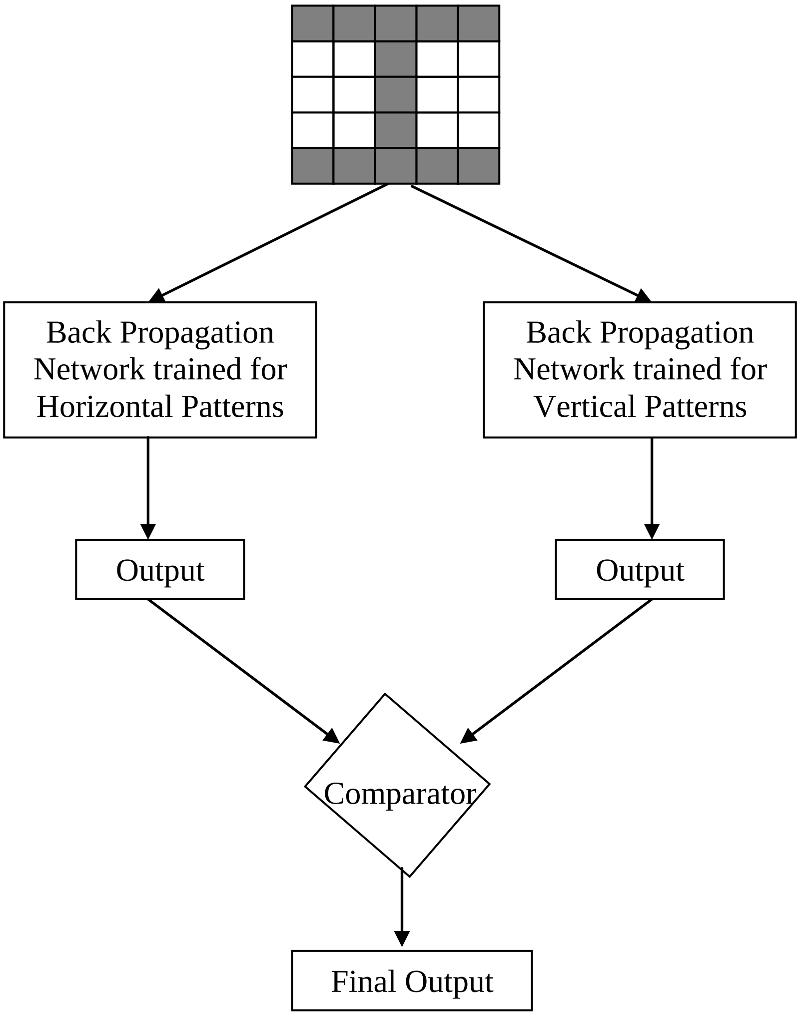


Figure 5.7: Modular Neural Networks for Recognition of Features

Here, two modules of fully connected feed forward networks were trained separately for horizontal and vertical patterns using the Back Propagation algorithm as shown in Figure 5.7. The two modules were similar in architecture. They consisted of 25 input neurons, 3 output neurons and 9 hidden neurons.

Each module was successfully trained for 3 horizontal patterns and 3 vertical patterns respectively, as shown in Figure 5.5. The recognition criterion was based on the highest activation of a specialized neuron, which is checked at the Comparator as shown in Figure 5.7. The results are tabulated in Tables 5.4 and 5.5.

The network was then tested for combinations of features. The features were those that make up the letters ‘E’, ‘I’, ‘O’ and ‘U’ as used in the previous sections. The Table 5.6 and the Graphs 5.37-5.40 illustrate the results obtained. The network was unsuccessful at recognizing the combinations of features.

Tested For	**Activation of Output Neuron corresponding to**		
	H_1	H_2	H_3
H_1	0.997684	0.002932	0.000081
H_2	0.006722	0.559855	0.416522
H_3	0.000855	0.000181	0.059791

Table 5.4: Results obtained by training a MNN for Horizontal Features

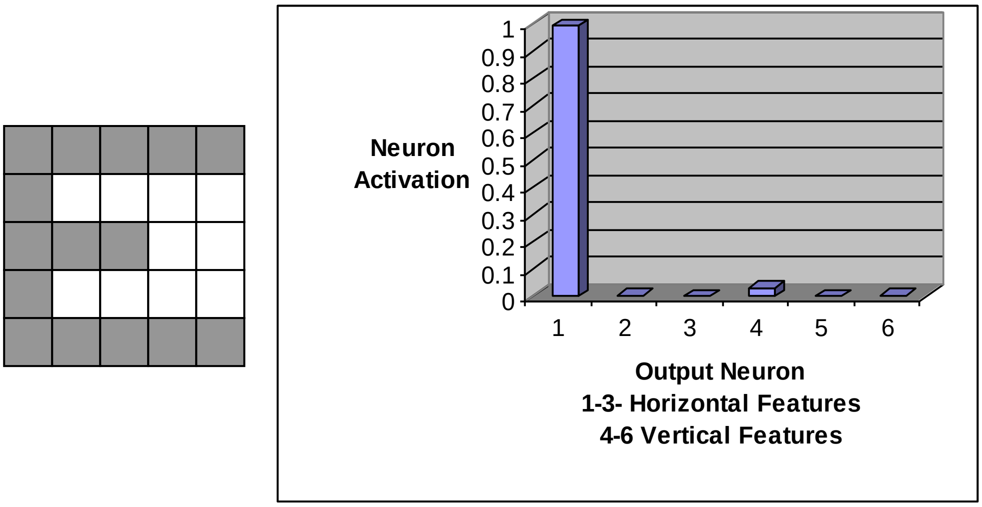
Tested For	**Activation of Output Neuron corresponding to**		
	V_1	V_2	V_3
V_1	0.993011	0.000085	0.000012
V_2	0.000017	0.990635	0.002761
V_3	0.000776	0.000002	0.299124

Table 5.5: Results obtained by training a MNN for Vertical Features

Test For	**Activation of Output Neurons Corresponding to**					
	H_1	H_2	H_3	V_1	V_2	V_3
E	0.993321	0.002339	0.000251	0.029460	0.000338	0.000874
I	0.973454	0.000061	0.004878	0.107542	0.000014	0.000130
O	0.973339	0.006741	0.000080	0.022960	0.000713	0.053173
U	0.601179	0.091937	0.000051	0.003286	0.031972	0.576141

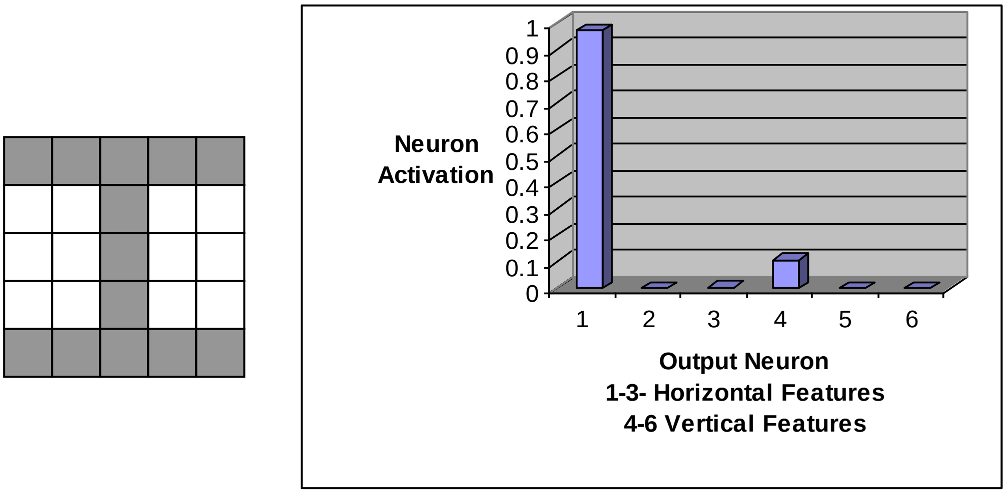
Table 5.6: Results obtained when the MNN was tested for Combinations of Patterns

Test for ‘E’



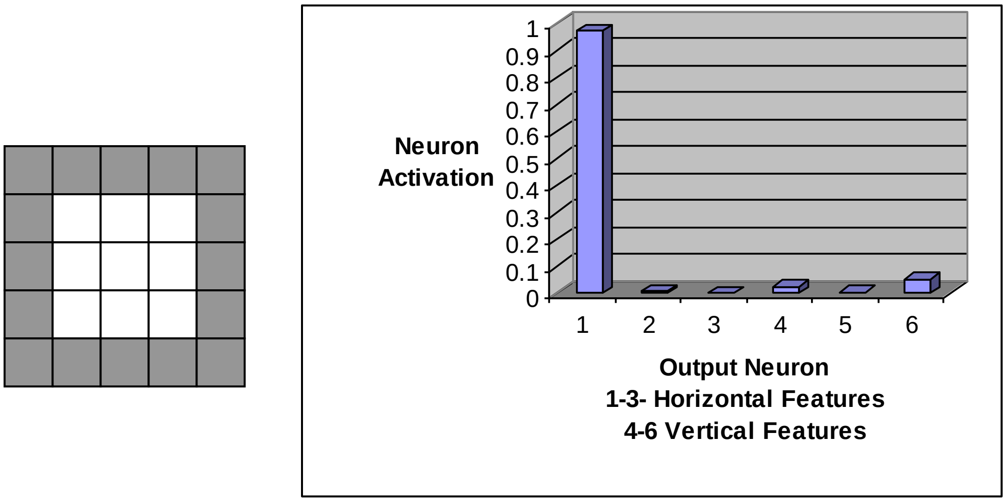
Graph 5.37: Illustration of the Output Neuron Activation when tested for ‘E’ with the MNN Feature Detector.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for ‘I’



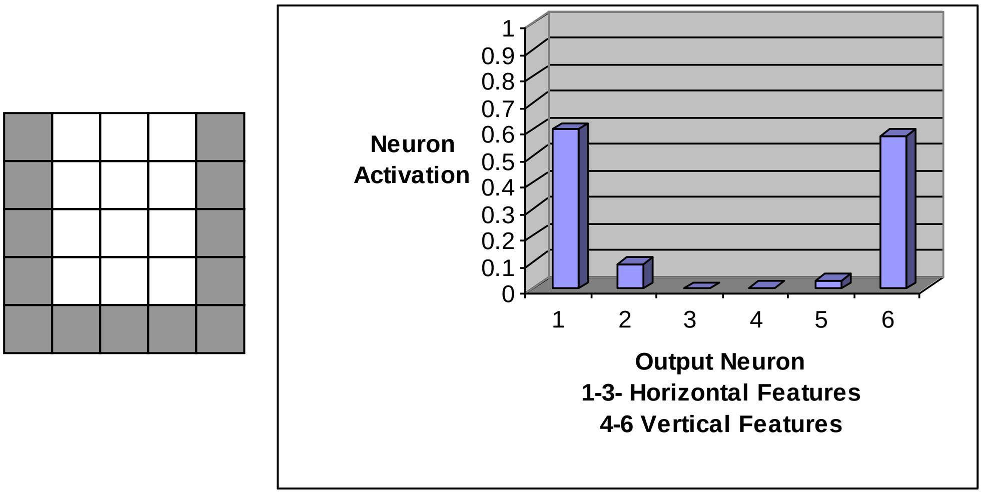
Graph 5.38: Illustration of the Output Neuron Activation when tested for ‘I’ with the MNN Feature Detector.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for 'O'



Graph 5.39: Illustration of the Output Neuron Activation when tested for 'O' with the MNN Feature Detector.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

Test for 'U'



Graph 5.40: Illustration of the Output Neuron Activation when tested for 'U' with the MNN Feature Detector.
Output Neurons 1, 2 and 3 are allocated for H_1, H_2 and H_3.
Output Neurons 4, 5 and 6 are allocated for V_1, V_2 and V_3.

5.5 HARDWIRED NETWORK *(EXPERIMENT 4)*

Tests show that the networks mentioned in sections 5.3 and 5.4 are clearly incapable of recognising combinations of features that the network was trained for, although they are very good at generalising (Test with new features – Section 5.3.5) and dealing with noisy features (Test with Noisy_1 and Noisy_2 features – Section 5.3.3 and 5.3.4). This problem can be solved, by designing a Hardwired Network.

The Hardwired network is inspired by the idea of 'Grandmother Cells', which suggests that there is a unique neuron at the apex of a hierarchical collection of cells that are involved in processing one's Grandmother's face that responds only to the Grandmother's face and nothing else, i.e. one dedicated neuron fires on recognising a particular object. If this theory were true, then the same could be applied to individual features of an image. The feature- detecting network has one neuron for each feature, as shown in Figure 5.8.

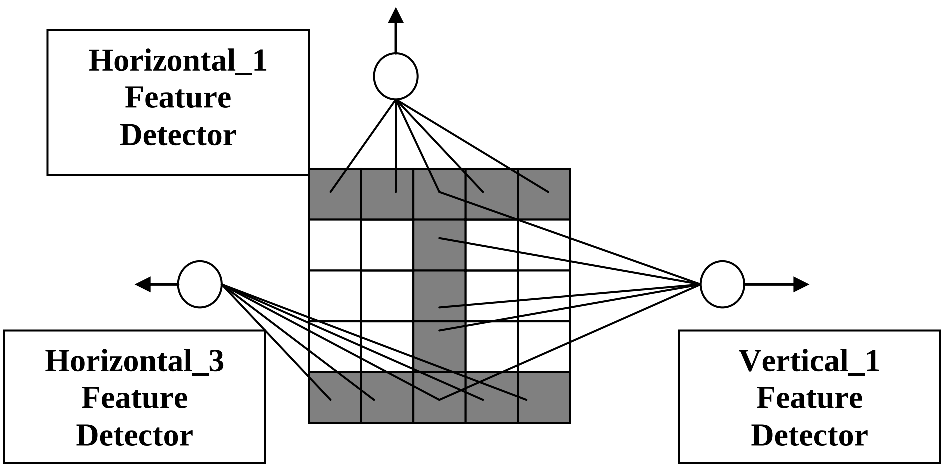
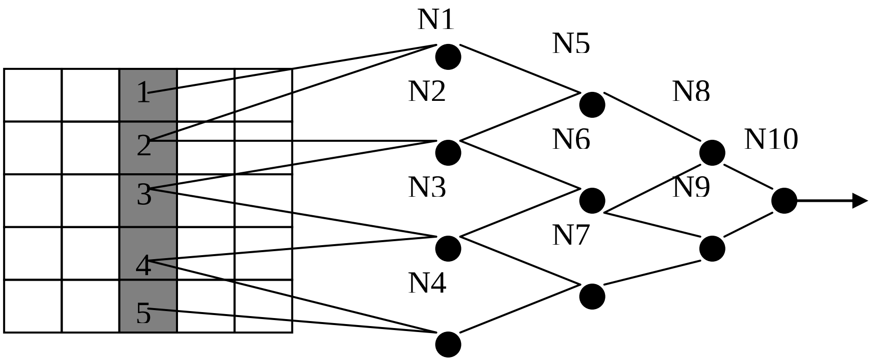
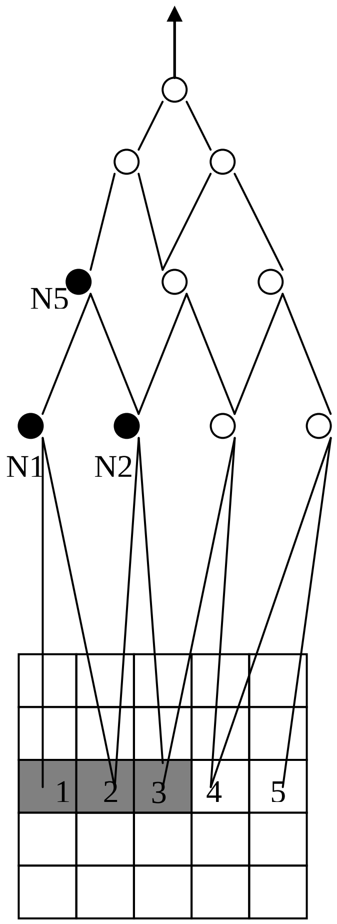


Figure 5.8: Feature Detecting Neurons

The hard-wired network for two of the six features (Figure 5.6) is illustrated in Figure 5.9



(i) Hardwired Network for Vertical Feature



(ii) Hardwired Network for Horizontal Features

Figure 5.9: Diagram illustrating the Hardwired Network for (i) Vertical and (ii) Horizontal Features

Each connection weight in the hardwired network has a value of 1. The threshold of each neuron is set to 1.5. The neuron fires whenever the activation is greater than or equal to 1.5. Two active cells connected to a neuron should cause the neuron to fire. A module of 10 neurons is required to recognise a vertical or horizontal feature made up of 5 cells.

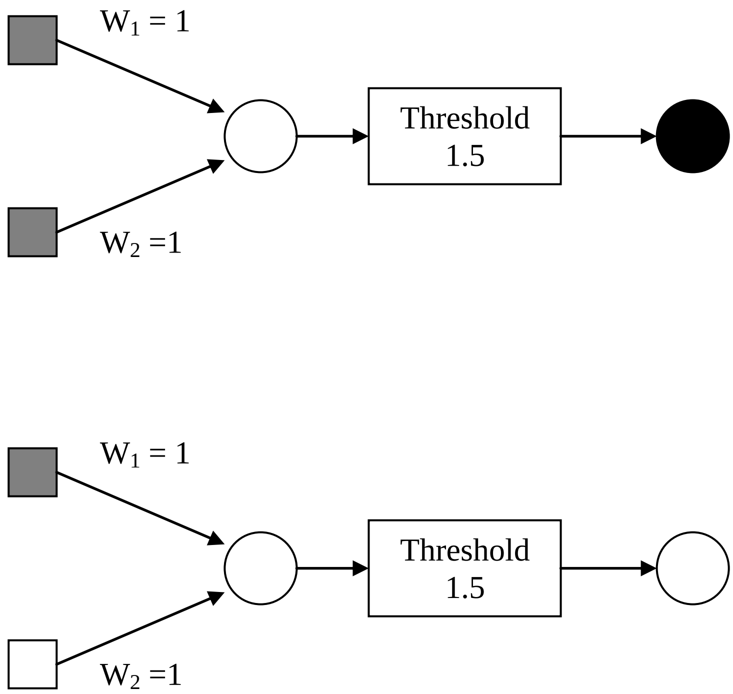


Figure 5.10: Rules of Neuron Activation (Filled Circles) and De-activation (Unfilled Circles)

In the Figure 5.9 (i), a vertical feature along the cells 1, 2, 3, 4 and 5 is shown. The network that recognises such a feature is also illustrated in the diagram. A feature along two consecutive cells aligned vertically is recognised by a dedicated neuron. Hence, the feature along cells 1 and 2 in the 5x5 grid is recognised by neuron N1. The feature along cells 2 and 3 is recognised by neuron N2. When a feature along cells 1, 2 and 3 is present, neurons N1 and N2 fire because of the presence of the vertical features along cells 1, 2 and 2, 3 respectively. As the neurons N1 and N2 are active, according to the rules of activation depicted in Figure 5.10, neuron N3 fires. Thus, the network shown in the figure comprising of 10 neurons recognises 10 features as shown in Table 5.7.

Vertical Feature comprising of cells	1,2	2,3	3,4	4,5	1,2,3	2,3,4	3,4,5	1,2,3,4	2,3,4,5	1,2,3,5
Active Neurons	**N1**	**N2**	**N3**	**N4**	N1 N2 **N5**	N2, N3 **N6**	N3 N4 **N7**	N1 N2 N3 N5 N6 **N8**	N2 N3 N4 N6 N7 **N9**	N1 N2 N3 N4 N5 N6 N7 N8 N9 **N10**

Table 5.7: Response of Neurons for Vertical Features

The table illustrates the neural activity when the network encounters vertical features along one single column of a 5x5 grid. As a network of 10 neurons are required for the detection of all possible vertical features in a single column, 5 networks made of 50 neurons are required to recognise all possible vertical features in a 5x5 grid. Similarly, 50 neurons are required for all possible horizontal features. In addition to this, 30 neurons are required for diagonal features.

Also noticeable from the Table 5.7 is that there are unique neurons responsible for the final identification of particular features in the hardwired networks. This is in accordance with the Grandmother Cell Theory.

The hardwired network is capable of recognising features and their combinations, but it is a very inefficient Engineering System.

Three main reasons for its inefficiency are that

1. A large number of neurons are required for simple features
2. The same neurons cannot be used for a new feature
3. There is no real learning involved in such systems, which is the essence of Artificial Neural Networks.

5.6 SUMMARY

The experiments conducted on Back Propagation networks and Modular Neural Networks have been discussed in this chapter. The main aims of the examination were to determine the difficulty of recognising combinations of learnt features in an image, to investigate the impact that the use of Modular Neural Networks has on classification accuracy and to identify the problems related to using traditional Neural Network architectures for Pattern Recognition.

The results obtained raised several intriguing questions including:

(a) *The Grandmother Cell Theory:* Are dedicated neurons among a group of cells that respond to specific set of features essential in Artificial Neural Networks?

(b) Feature detecting capabilities of Artificial Neural Networks: Why are Artificial Neural Networks capable of only recognising sets of features as whole images rather than recognising individual features that make up the image?

(c) The Architecture of Modular Neural Networks: What type of simple and elegant but robust interaction and conflict resolution between the modules of a Modular Neural Network can improve the accuracy of these networks?

(d) The relevance of Hardwired Networks in Artificial Vision Systems: Are hardwired networks completely incompatible with Artificial Vision Systems because of their large size?

To answer these questions and be able to solve the problems encountered with currently available network architectures, an investigation of biological vision systems was conducted to understand the gradual development of the visual system in the animal kingdom. The findings are discussed in Chapter 6.

CHAPTER 6: BIOLOGICAL VISION SYSTEMS

6.1 INTRODUCTION

Investigations and experiments conducted using traditional AI methods, as explained in Chapter 5, showed that such systems have several limitations. Therefore, attention was directed at biological systems in an attempt to find clues to overcome these problems. This chapter reviews some of the biological avenues, which were explored.

Humans are endowed with an extremely complex vision system. This highly advanced system evolved over the course of countless aeons, starting from a few light sensitive cells making up a simple eyespot (with sensitivity that could only distinguish between the presence or absence of light) to an eyespot with optics, to the aggregation of optic cells, to tissues surrounding the optic cells, to organs capable of photoreception and finally to a complete vision system [91]. Figure 6.1 shows the gradual evolution of the eye in modern molluscs [92].

The animal kingdom, even today, contains many versions of the eye, from simple eyespots in invertebrates to the complete vertebrate systems capable of avoiding obstacles and recognising prey, predator and mate [91]. Hence, a study of the biological vision systems was undertaken to understand the way various lower animals recognise objects.

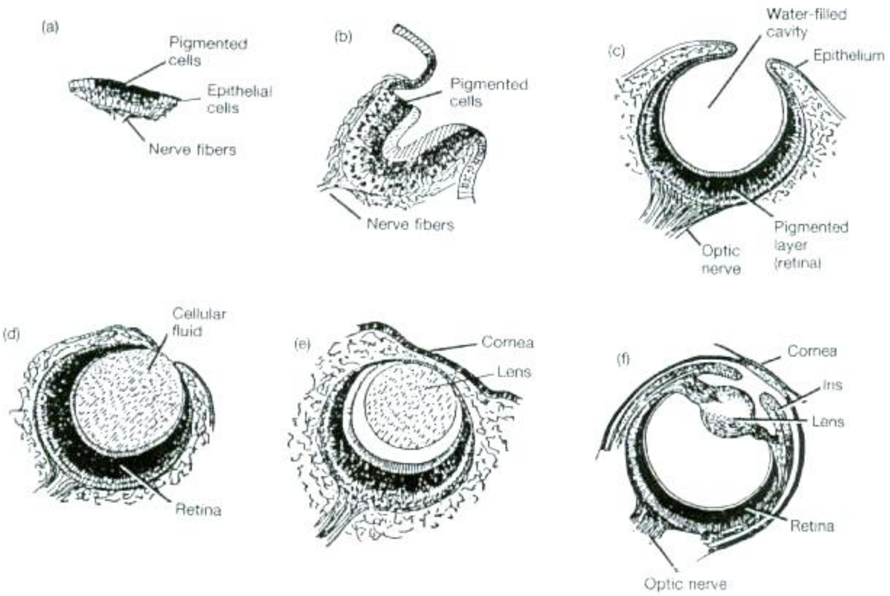


Figure 6.1: Illustration of Evolution of the Eye in Various Molluscs. Taken with author's permission from [92]

(a) Spot of Pigmented Cells (b) Folded region of Pigmented Cells (c) Pin-hole Camera (d) Eye Cavity filled with Cellular Fluid (e) Eye Protection by transparent cover of skin (f) Full Complex Eye in Octopus and Squid

This chapter discusses the development of vision in the animal kingdom and some of the attempts made to exploit this in Artificial Intelligence. Also a detailed discussion of the insect eye is given in Section 6.3, as it is the most studied visual system in the animal kingdom. A description of the trials to reproduce its operation is given in Section 6.3.5. Aspects of the human visual system are explored in Section 6.4. Finally, the toad's visual system is discussed in Section 6.5

6.2 VISION IN THE ANIMAL KINGDOM

The evolution of vision in the animal kingdom was dependent on the physiological characteristics of the animal and the environment in which it lived. A significant amount of information available on biological vision systems relates to the operation of the eye itself and not on the neurological processes that go on after the image is captured by the photoreceptors. Unfortunately, this means that studies are restricted to those animals whose vision system is better understood. Hence, a detailed study can be conducted only in a few animal vision systems.

Although extensive research of several animal vision systems was undertaken in this research programme, only a brief review of these systems is given in this section to give an overview of the vision system in the animal kingdom and the type of biological information available to design a biologically inspired Artificial Vision System.

There are nine main phyla into which animals can be subdivided. Each phylum consists of several sub-phyla under which various animals are classified. A general overview of the vision systems under the main phyla whose vision systems were better studied is given below. The information given here is simplified.

1. **Porifera:** Sponges belong to this phylum. These simplest of aquatic animals are an aggregation of cells with no differentiated tissue or

organs. Pores and canals on the body allow the passage of water that carries organisms (food) that stick onto the walls of the cells, which is directly absorbed. There is no visual perception in these animals [93].

2. **Cnidaria:** Hydras, Jellyfish, Sea Anemones are classified under this category. They live under water and several cells group together to form layered tissues. They detect their prey or predators by using stinging cells called nematocysts on the body surface, which are sensitive to touch [94]. Certain of these animals are sensitive to light.
3. **Platyhelminthes:** Flatworms fall under the phyla Platyhelminthes. These animals have eyespots that are sensitive to light and darkness. The eyespots have different types of photoreceptors in the various sub-phyla, displaying varied photosensitivity based on displacement and orientation [95].
4. **Nematoda:** Roundworms belong to the phylum Nematoda. Most Nematodes are photosensitive, despite the lack of photo spots. They do not have true eyes. Some Nematodes have two anterior cup-shaped photo spots of dense pigment [96].
5. **Mollusca:** Snails and Squids are some of the animals that fall under the phylum Molluscs. These creatures are an "evolutionary dead-end" that didn't lead to more advanced creatures, but were notable for their successful adaptation to various marine environments. The molluscs have a wide variety of eye types. Limpets have eyecups, Scallops use mirrors to form an image and Squid and Octopuses have advanced eyes. Photoreception in Molluscs is not just confined to eyes, but is also present in several internal neurons as observed in the sea slug Aplysia.

Most aquatic animals do not have a cornea. Hence, the lens carries out the refraction of the light. Lenses are generally spherical as this gives a shorter focal length. These lenses are an evolutionary invention, the properties of which have not yet been replicated by Optical Engineers [97].

6. **Annelida:** Leeches are an example of Annelids. In Annelids, photoreceptors are present at various parts of the body including the head. In leeches the pigmented cells form a cup around the photoreceptor cells. An increase in the conductance of the microvillar membrane is observed during absorption of light. This results in a flow of current in the Medicinal Leech, Hirudo Medicinalis, which flows inward through the microvillar membrane and outwards through the external membrane. These animals respond to changes in illumination caused by shadows of other animals and adapt very quickly to any changes in the environment [98]. Therefore, they can sense the direction of incoming light.
7. **Arthropoda:** Crustaceans, Insects and Horseshoe Crab are some of the animals classified as Arthropods. Groups of photosensitive receptors called ommatidia form compound eyes in these animals. This represents a more complex type of optical detector and is discussed in detail in section 6.3 [99] [100] [101].
8. **Echinodermata:** A Starfish is an example of an Echinoderm. In some Echinoderms, an optic cushion is present on the surface of the tip of each arm. The optic cushion is composed of several ocelli. These are sometimes called compound ocelli, as they are an aggregation of pigmented cells. A layer of corneal cells covers each ocellus. The

sensory cells send axons to the nerve fibres situated below the ocellar cup [102].

9. **Chordata:** All animals with backbones fall under this phylum. Some of the animals are Fish, Amphibians, Birds and Mammals [103]. These animals are highly intelligent and have very advanced vision systems. A general theory on the human vision system is given in section 6.4. The vision system of the amphibians is discussed in section 6.5.

As mentioned earlier, among the animals mentioned under various phyla, the insect eye is the most understood and researched system. A detailed literature search on the insect eye mechanisms was undertaken.

6.3 THE INSECT EYE

The visual system of the various insects differs by varying degrees. However, the main structure of the visual system and the components, which help the insects with perception of various objects, is basically the same [104].

The main organs that insects use to perceive light are compound eyes, one on either side of the head. Compound eyes are made up of cells called ommatidia. The size, shape and number of ommatidia vary among different types of insects. In the eyes of the worker ant, Ponera Punctatisssima, there is only one ommatidium, whereas there are about ten thousand in those of the dragonfly [105].

6.3.1 STRUCTURE OF THE OMMATIDIA

The ommatidium gathers light with the help of the biconvex corneal lens, which forms the outer transparent surface of the compound eye. The hard crystalline cone, which is produced by the four sempler cells beneath the corneal lens, also helps in capturing light. The crystalline cone is surrounded by the primary pigment cells. In some insects the crystalline cone is not present and the number of sempler cells may vary [105] [106]. The structure of a typical single ommatidium is as shown in Figure 6.2.

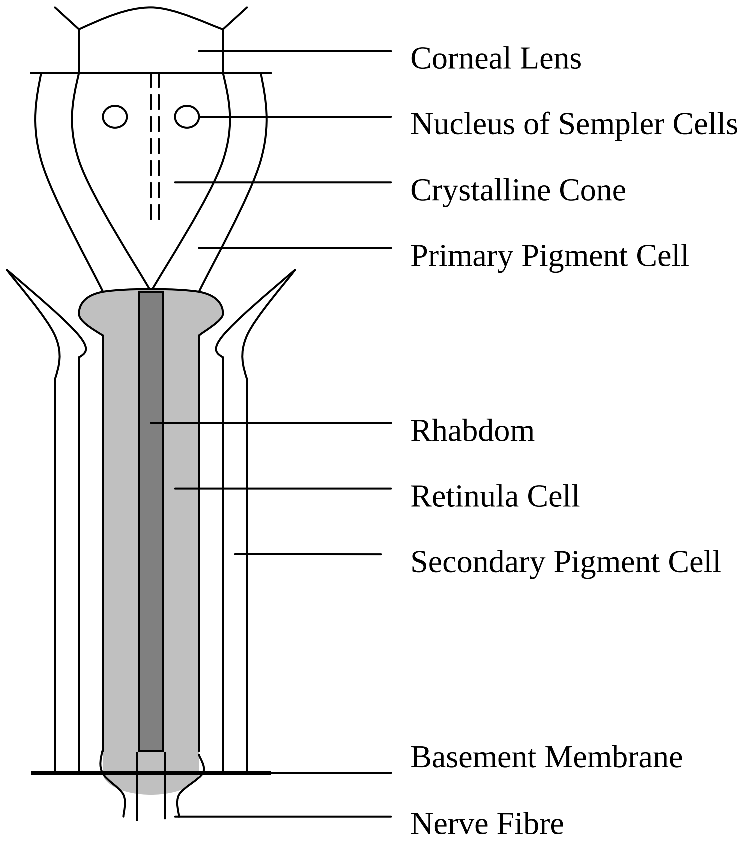


Figure 6.2: Structure of the Ommatidium [105]

Below the crystalline cone, long sensory nerve cells called retinula cells form a rhabdomere at the axis of the ommatidium. A nerve axon from each cell passes through the basement membrane and is connected to the optic lobe of the brain. The secondary pigment cells surround the retinula cells and isolate the various ommatidia in an insect eye [105].

In insects with few ommatidia, they are usually round, spread apart and separated by cuticle, whereas in insects with a larger number of ommatidia, they are hexagonal in shape and are more closely packed as shown in the Figure 6.3 [105].

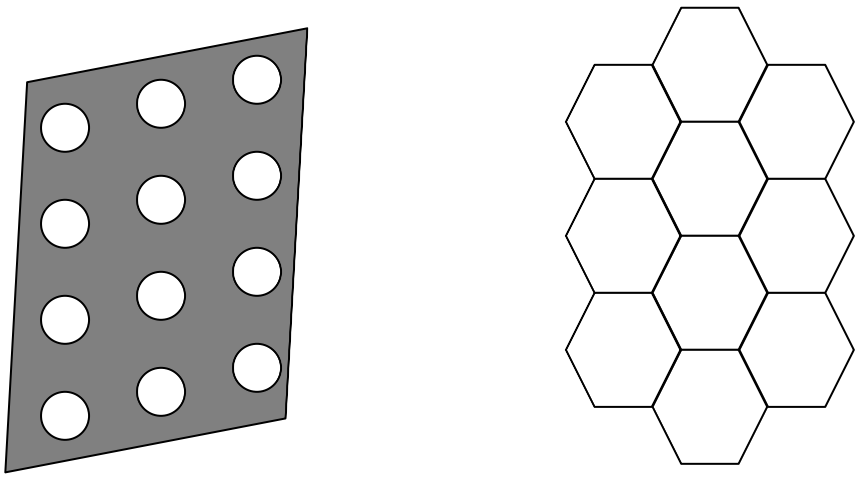


Figure 6.3: The arrangement of ommatidia of fewer numbers (left) and larger numbers (right) [105]

6.3.2 NEURAL NETWORKS IN THE OPTIC LOBE OF THE INSECT

The optic lobe, to which the insect eye is connected through its retinula cell axons, consists of three ganglion layers. They are the Lamina Ganglionaris, the

Medulla Externa and the Medulla Interna. All these layers are interconnected by chiasmata. Short axons from the retinula cells run to the Lamina Ganglionaris and longer axons sometimes run directly to the Medulla Externa. There is convergence of the nerve fibres from the eye to the Medulla Externa as illustrated in the Figure 6.4.

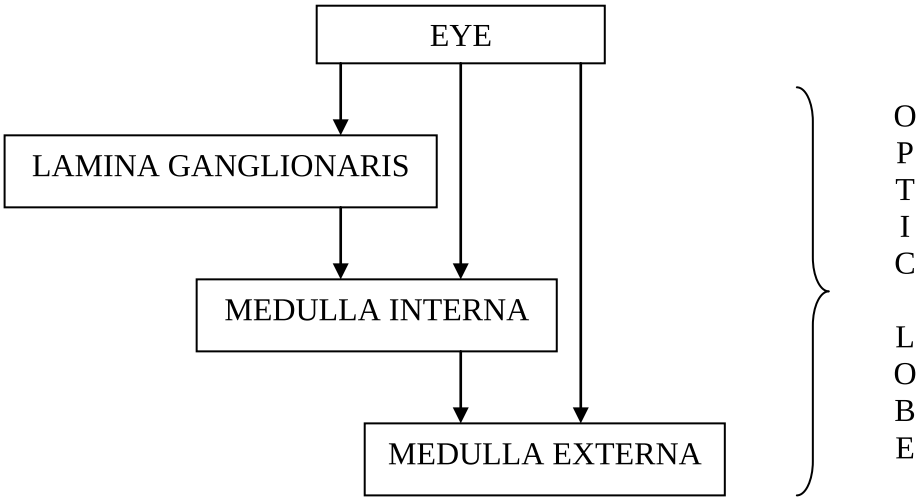


Figure 6.4: Connections from the eye to the Ganglion Layers in the Optic Lobe [105]

6.3.3 THE OPERATION OF THE INSECT EYE

The ommatidia receive light from a limited part of the visual field, which in some insects overlap with the visual field of other adjacent ommatidia. It is the lens system that allows the light to enter the rhabdomere in a narrow beam. The ommatidia then receive illumination from part of the visual field, based on the amount of light reflected from the object. The result is that the ommatidia of the insect eye produce several spots of light intensity, which are integrated together to form an image of the object being perceived [105].

Visual contrast is enhanced by the phenomenon of lateral inhibition that occurs between ommatidia. Lateral inhibition may be particularly important where there is a large overlap of the visual fields of the ommatidia. The strongly illuminated ommatidia inhibit the surrounding ones, which are weakly illuminated [105][106].

6.3.4 PERCEPTION

Visual perception varies between different insects based on the number of ommatidia and their construction.

The Locust responds mainly to vertical stripes on a white background. It also reacts positively to larger complex objects in the absence of vertical stripes [105].

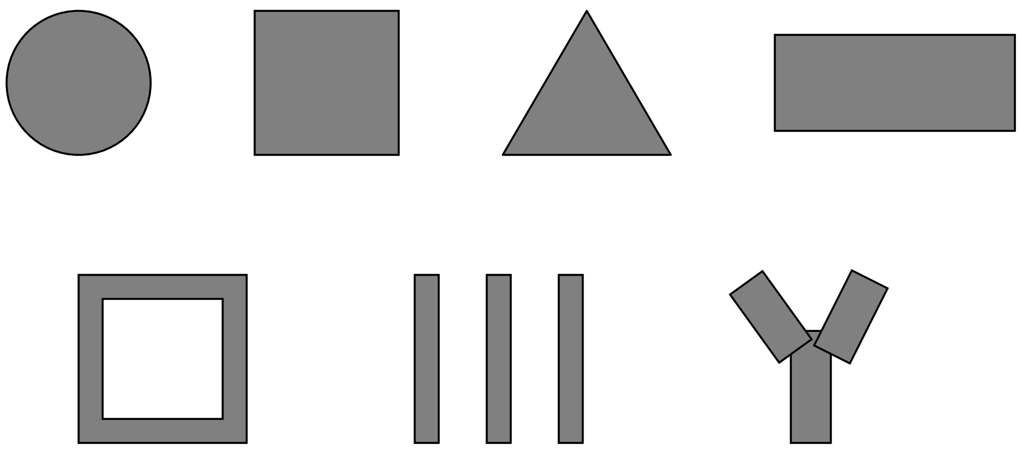


Figure 6.5: Solid Objects (top) and Broken Objects (bottom) [14]

The Apis differentiates between solid objects and broken objects, responding positively to broken objects. However, they cannot differentiate between several solid objects or several broken objects as shown in the Figure 6.5 [105].

Philanthus recognises landmarks such as pinecones and it gets confused in finding its nest in the absence of such landmarks. Bees are reported to respond to the flicker effect caused by the movement of flowers, which causes changes in stimulation of their ommatidia [105].

Most insects are also able to perceive depth that enables them to avoid obstacles (other insects) in air and to land with precision. Binocular vision plays a very important role in the perception of depth when the ommatidia in both the eyes are illuminated simultaneously [105].

6.3.5 HARDWARE IMPLEMENTATION OF THE INSECT EYE

Inspired by the fly vision system, Franceschini developed an obstacle avoidance system for a robot. The system was based on the principle of motion detection [107][108].

The artificial vision system used an array of electro-optical Elementary Motion Detectors (EMDs) that computed the angular speed of a point P that crosses the visual field. A photosensor layer consisting of the sensors and related circuits is connected to the EMDs. The outputs of the EMDs (analogue signals) are integrated at an anti-collision layer, which steer the robot and avoid obstacles successfully [107].

6.4 HUMAN VISUAL SYSTEM

The other vision system studied in detail is the Human Visual System [109][110]. The main aim of the research conducted in the human visual system was to understand the processes that occur at various stages in analysing a visual scene.

The ganglion cells in the retina of the eye perceive various objects and this information is passed to the Lateral Geniculate Nucleus (LGN) and through other inter-neurons before they arrive at the Primary Visual Cortex, V1. The pathway that facilitates the information flow to area V1 of the brain is called the Ventral Visual Pathway, which passes on information to higher visual centres such as V2, V3, V4 and V5 [111].

Visual information also flows through the Tectal Pathway to the Superior Colliculus (SC), which passes information only to some visual areas of the brain. The pathways that carry information to V1 and back to the LGN and SC consist of neurons that can filter sensory information and recognise objects [111].

Although the interconnections between various modules of neurons and their operation are not fully understood, a general structure of the functions of neurons in the Ventral Visual Pathway was proposed by many researchers.

In Humans, the layers closer to the retina respond to light in varying degrees such as ‘On-centre and Off-surround’ or ‘Off-centre and On-surround’. Some animals may carry out basic responses based on the level of illumination of these cells.

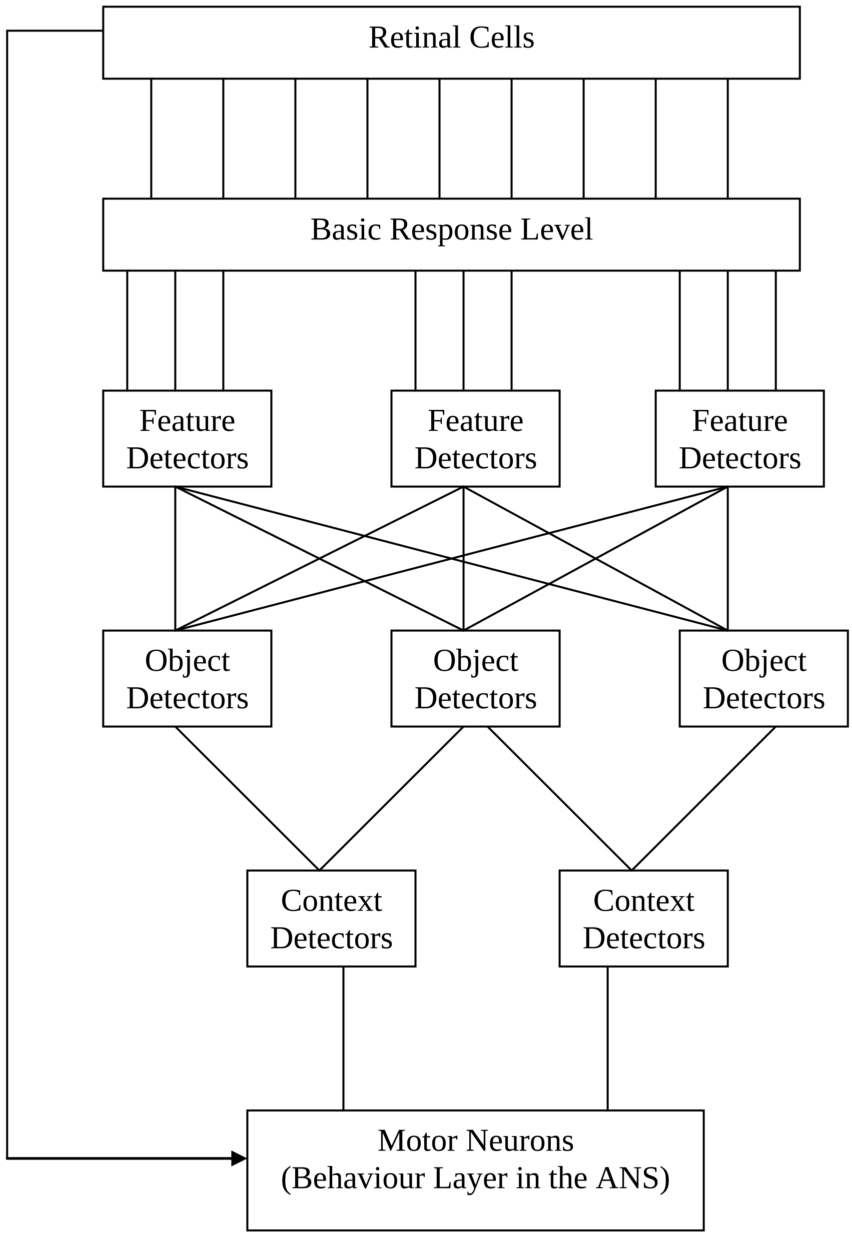


Figure 6.6: Hierarchical Structure of the Visual System [112]

It is thought that the next level of neurons respond to basic features of an image. These features are then integrated together by neurons in deeper layers, which act as object detectors. The next level of neurons pools together the objects in a scene to understand the context. Connections from the context detectors run to the motor layers to produce appropriate behaviours [112]. This layered structure is illustrated in Figure 6.6.

The research undertaken on the human visual system showed that analysis of a scene starts from a very low level and gradually builds up a more complex picture. Hence, any robust Artificial Vision System based on the human visual system will need to be built on the basic aspect of feature analysis rather than processing the image as a whole. Also, the human visual system seems highly modular (see Chapter 4). During this investigation, a clear idea of the importance of a modular vision system that recognises simple features was formed.

6.5 TOAD'S VISUAL SYSTEM

Since the 1960s, extensive research has been carried out on frog and toad visual systems. Ewert, Arbib, Lettvin, Ingle and many other neuroscientists have been involved in analysing the frog and toad vision systems. Famous papers such as "What the frog's eye tells the frog brain" by Lettvin, et al published in 1959 [113] have made an important impact in the field of neuroscience.

Neuroethologist, Jorg-Peter Ewert and his co-workers have investigated the mechanisms involved in prey-catching in toads [114][115]. This section considers the biological vision system of the common toad, *bufo bufo* and its engineering implementation in an artificial context.

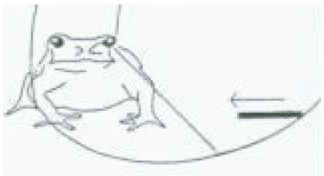
6.5.1 SIGN STIMULI AND BEHAVIOURS

Ethological studies have shown that only certain features of the input stimulus in an environment are necessary to cause behaviours, such as 'orient', 'snap', 'approach' and 'escape'. These features are called Sign Stimuli [116].

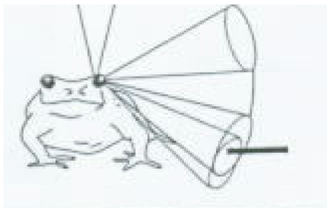
A series of classic experiments were conducted to analyse the sign stimuli related to prey-catching [116]. This was done by confining a hungry common toad, *bufo bufo,* in a glass vessel and circling a cardboard model round it. When it recognises the model as prey, it tries to orient itself to bring the object into its frontal visual field.

Elongation of the model in the horizontal direction (along the direction of movement) elicits prey-catching behaviour and elongation of the model in the vertical direction (perpendicular to the direction of movement) releases predator avoidance behaviour.

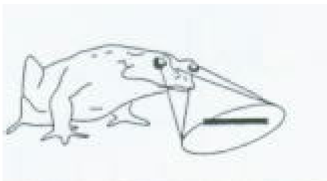
This differentiation between prey and predator is independent of the model's colour or velocity of movement. Therefore, it was found that the sign stimuli in the toad are only based on the elongation of the object with respect to its direction of movement [116].



(a) Orient



(b) Approach



(c) Fixate

(d) Snap

Figure 6.7: Action Patterns associated with Prey-catching Toads. Reproduced with permission from author [115].

Prey-catching in toads involves four action patterns – orientating (o), approaching (a), fixating (f) and snapping (s), which is illustrated in Figure 6.7.

The action patterns are released by monitoring the location of prey in space. The action patterns are not always released in the order o, a, f, s. Depending on an object's location in the toad's visual field, it could trigger any sequence of action patterns such as o, s or o, f, s, etc, [115].

6.5.2 THE NEURAL BASIS OF BEHAVIOURS

The interesting aspect of the toad's visual system was that neuro-ethologists reported that different classes of neurons were activated for specific stimuli. They could comment and speculate on the neuronal processes that occur in the toad's brain due to the presence of fewer number and classes of neurons compared to the millions in the human brain.

Research has shown that there are at least four classes of ganglion cells in the toad's retina (R1-R4). These ganglion cells are connected to the Optic Tectal (OT) neurons and the Pretectal (PT) neuropil. Modulatory inputs from the Thalamic (AT – Anterior Thalamus) and Pretectal structures arrive from the Telecephalon (TC) as shown in Figure 6.8 [115].

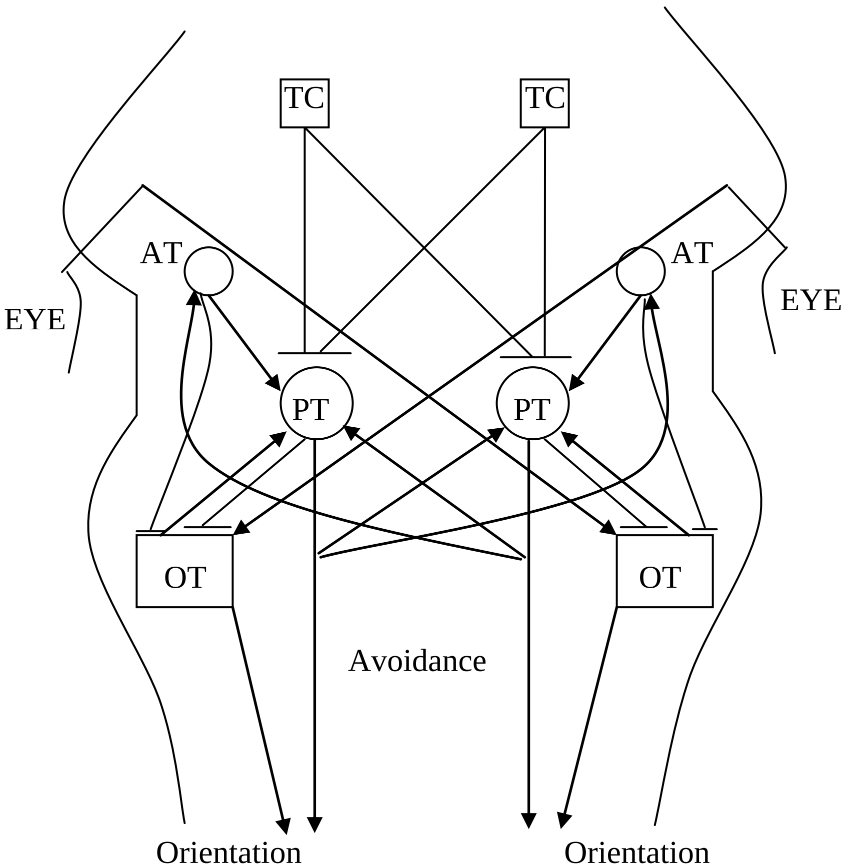


Figure 6.8: Prey-Catching/Predator Avoidance [115].

Ewert [115] proposed that:

1. Prey-catching is controlled by the optic tectum depending on inhibitory influences from the thalamic - pretectal nuclei.
2. Predator avoidance is controlled by the thalamic-pretectal nuclei with excitatory influences from the optic tectum.

Several classes of neurons in the thalamic, tectal and pretectal regions (e.g. TH3, T5-1, T5-1′, T5-2, T5-3, R1-R4) have been identified in the brain, as shown in Figures 6.9, 6.10 and 6.11. Their responses to various types of objects have been studied. Ewert proposed possible neural circuits of the sensori-motor connections in the toad as shown in Figures 6.9, 6.10 and 6.11.

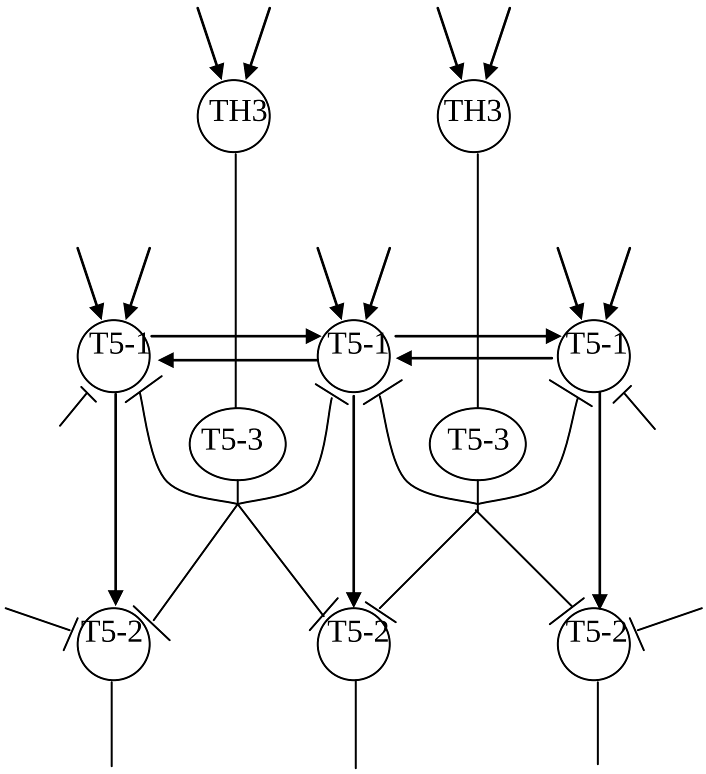


Figure 6.9: Toad's Sensori-motor Connections. Ewert's Proposed Circuit-1 [115]

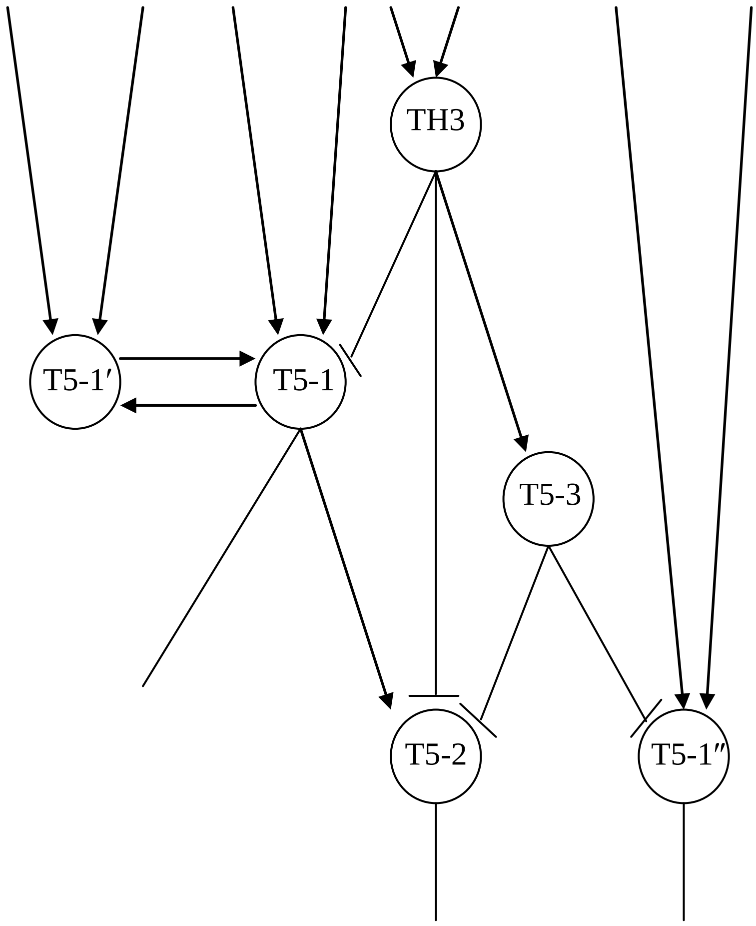


Figure 6.10: Toad's Sensori-motor Connections. Ewert's Proposed Circuit-2 [115]

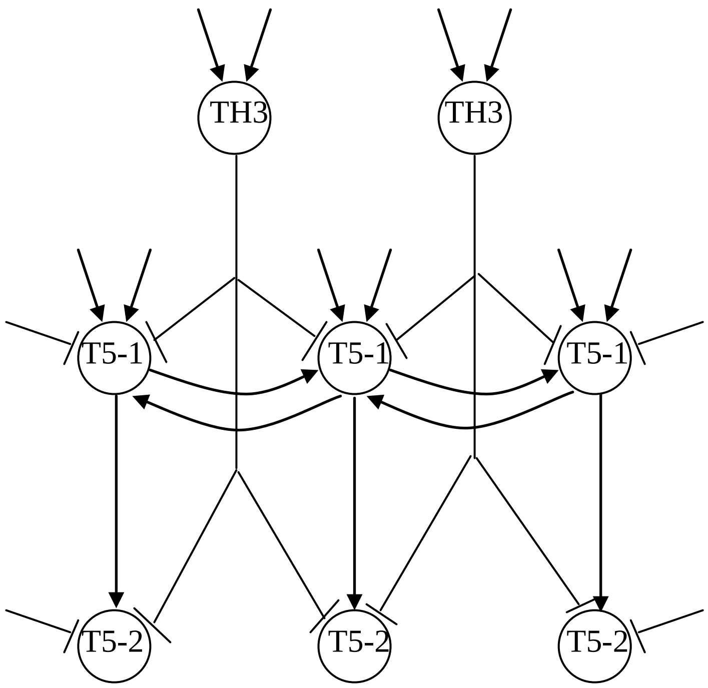


Figure 6.11: Toad's Sensori-motor Connections. Ewert's Proposed Circuit-3 [115]

Studies have been conducted on the ganglion cells such as R2, R3 and R4, which have different activation values corresponding to various stimuli [117]. Other research has been involved in assessing the behaviour of the toad/frog and the respective neuronal responses when presented with a prey and predator at the same time [118]. Learning and motivation also affect the response of the toad to prey and predator [119].

6.5.3 ENGINEERING APPLICATIONS

The most important engineering applications of the toad visual system were in robotics and machine vision. The various systems developed were primarily based on the model of the neural circuit (shown in the Figure 6.12) proposed by Ewert and Von Seelan following several neuroethological experiments on the toad, as mentioned in previous sections [120].

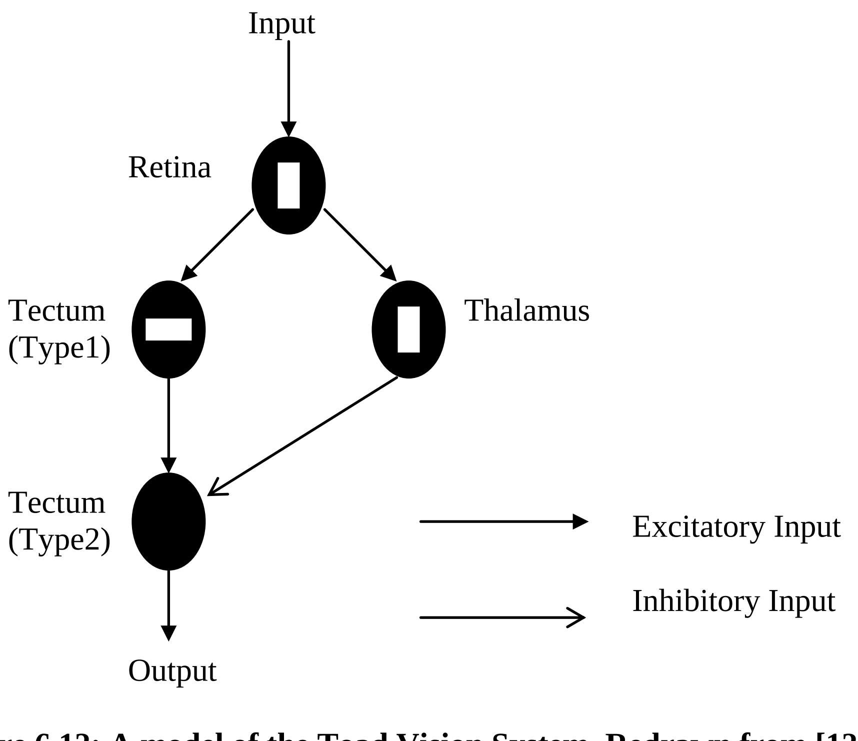


Figure 6.12: A model of the Toad Vision System. Redrawn from [120]

The input via the retinal cells is passed to both the Tectum–Type 1 and the Thalamus. Tectum-Type1 (a lateral window) and the Thalamus (a vertical window) are viewed as two filters working in parallel. When the network processes a rectangular shape, the output is very high as the Thalamus being a vertical filter hardly contributes anything as an inhibitory input to the Tectum–Type 2.

Similarly, when the retina captures a vertical stimulus, the output from the Tectum-Type 1 neurons is very low as it is a horizontal filter. Although, the output from the Thalamus neurons is high, as it provides a negative input to the summing junction or Tectum- Type 2 output neuron, the total output is very small.

A robot that was capable of sorting objects on a conveyor belt was successfully developed by Michael Arbib using the principles of Artificial Neural Networks [121].

The main tasks were to detect a moving object, measure its dimensions and analyse its orientation relative to the direction of motion. An Artificial Neural Network called MODNet was simulated using the programming language C++ to achieve the objectives of the sorting problem [121].

The image of an object of dimensions, l_1 and l_2 moving at a velocity, v is captured by a CCD camera, C. The image is then digitised (D) and a bit-map transformation (B) is obtained. The image is further processed using the operations of filtering and clustering (F). The dislocation of the stripe shown in the difference image, d is calculated using an XOR procedure that correlates to the ON and OFF events corresponding to the leading and trailing edges of the moving object [121]. A robot was successfully trained to classify the objects, grasp them and sort them into separate bins [121].

Functional units called Schemas were first used by Arbib to explain the behaviours in animals [122]. As explained, toads snap at small moving objects, elongated horizontally relative to the direction of movement, as they recognise them as prey. Large moving objects elongated vertically with respect to the direction of motion are recognised as predator. Thus, there are two simple systems involving pattern recognition, which according to the schema theory are called *Perceptual Schemas*. These perceptual schemas (based on their level of activation) bring the Motor Schemas into action and trigger the motor controls to elicit appropriate behaviours [123] [124][125].

Nishio, et al proposed an analogue circuit for shape recognition based on the visual system of the frog proposed by Ewert and Von Seelan [126][127].

6.6 SUMMARY

Although research in the field of biological vision throughout the animal kingdom has revealed many fascinating facts, there are huge gaps in our knowledge of the neurobiology and functionality of these systems. More information is available on the structure of the eye and its evolution. After the visual information is captured, the various modules of neurons process the information to add meaning to the visual stimuli. The architecture of the neurons in the visual pathway and its operation, which facilitates the visual analysis, is still unknown.

The knowledge acquired from the biological models studied, which were presented in this chapter was insufficient for the purpose of the research project of developing an intelligent vision system using Artificial Neural Networks, as very little was known of the neuronal processes that take place. However, during this study of biological vision systems, it was discovered that the neuroethological research undertaken on the toad’s vision system seemed to have more promise in developing the unique vision system.

Improvements in biologically inspired vision systems are very dependent on the research done by biologists. Although, there are some successful technological implementations of the toad’s vision system, development of an intelligent general-purpose artificial vision system has not yet been achieved.

The Artificial Vision System called Distributed Neural Network (DNN), developed in this research programme to fulfil the purpose of this research is proposed and explained in Chapter 7. This system tackles some of the fundamental problems of neural networks used for Pattern Recognition and is proposed as a step forward in the right direction, contributing to the state of the art to develop an intelligent artificial vision systems.

CHAPTER 7: DEVELOPMENT OF THE DISTRIBUTED NEURAL NETWORK (DNN)

7.1 INTRODUCTION

The research to develop a biologically inspired Artificial Vision System led to the study of several biological vision systems, including the toad's visual system, which was explained in Chapter 6.

The success of implementing a biologically inspired vision system was to a large extent based on the current literature available in the biological research community. As the neuronal processes in the toad's brain that are involved in the analysis of visual stimuli and subsequent behavioural response were known to a greater extent compared to others, the toad's visual system was chosen over the other visual systems to develop a novel Artificial Neural Network called the Distributed Neural Network (DNN).

In this chapter, the development of the unique network architecture, DNN, developed in the research programme is discussed. The implementation of the system using Artificial Neural Networks (ANNs) and training using an Evolutionary Algorithm for Reinforcement Learning (EARL) are presented in Chapter 8.

7.2 ARTIFICIAL MODEL OF TOAD'S VISUAL SYSTEM

The development of any Artificial Vision System using ANNs involves four important factors. They are:

1. The Design of the Network Architecture
2. The Interpretation of the Input Data
3. The Interpretation of Output Data and
4. The Training Algorithm

The sections that follow explain the gradual development of the network architecture that draws inspiration from Biology and AI methods.

7.2.1 ARCHITECTURE OF THE DNN

To design the network structure, DNN, several ANN architectures and training methods available in the literature were explored (as explained in Chapters 3 and 4). Alongside ANNs, a literature search was also conducted into Biological Vision Systems (Chapter 6).

The DNN was the result of inspiration from

1. Biology – The Toad's Visual System and
2. Artificial Intelligence – Feed Forward Networks

7.2.1.1 TOAD: INSPIRATION FROM BIOLOGY

The artificial model of the toad's vision system is based on the neuronal interconnections proposed by Jorg Peter Ewert [128][129][130][131][132]. The neuroethological findings shed some light on the neural networks in the toad's brain involved in the classification and interpretation of visual stimuli as described in Chapter 6.

The main factors that were considered in the development of the DNN were,

1. The specific routines for prey and predator recognition
2. The various classes of neurons with individual functionality
3. The interactions between the neural clusters TH3, T5-1, T5-2, T5-3 (AT, TC, PT and OT)
4. The type of connections (inhibitory or excitatory) between neurons
5. The connections between Optic Tectum (Prey) and Pre-tectal (Predator) neurons

The proposed Artificial Vision System, DNN, is not biologically accurate, although biologically inspired. Several deviations from the biological model were made (mentioned appropriately throughout the chapter) and the network was pruned to simplify the model as much as possible.

Many biological systems are capable of complex activities using very simple mechanisms. Hence, simplicity was one of the important factors considered during the design stage for ease of training.

7.2.1.2 FEED FORWARD NETWORKS: INSPIRATION FROM ANNs

As mentioned in section 3.9, around 70% of real world applications are solved using layered, fully connected, feed forward networks trained by the Back Propagation Algorithm.

Neurons of one layer are only connected to the next layer and connections are never formed within the same layer. Such networks, simply called Feed Forward Networks (FFNs) are one of the earliest and simplest models of ANNs. Although they are not biologically plausible, they have proved to be very efficient in adapting to engineering problems.

The features of the FFNs that were considered important for the design of DNN are:

1. All the neurons in these networks are uniform in form and function.
2. Also, none of the neurons are specialised for any specific purpose.
3. The neurons in the network perform the summation of the products of inputs and connection weights of the network.
4. A sigmoid activation function is then used to generate the output of the neuron as illustrated in sections 3.2 and 3.4.

FFNs were an inspiration in constructing the artificial vision system based on the toad's vision system because of their simple architecture and operation. Hence the following fundamental characteristics were established early during the design process.

1. Neuron Functionality

The specialised neurons in the toad (such as TH3, T5-1, T5-2 and T5-3) were all considered to function in the same way as the basic neuron model explained in section 3.2. Also, very little of the biological properties of the specialised neuronal clusters was known. This drawback did not hinder the architectural design process of the network because the functionality of individual neurons was not considered to be significant, as the FFN, despite its similar simple neurons is very successful at classification. The few cases of failure in BPNs occur when very fine classification of computationally complex datasets is required, where higher order neurons prove useful [133].

2. Input (Retinal) Cells

Although several types of retinal cells, R2, R3 and R4 were identified, all retinal cells in the artificial model were considered to be the same in their operation, which facilitates the input of visual stimuli to the network. The way external stimuli propagate through the network is similar to the process in FFNs.

3. Layered Structure

Another major design consideration was the layered structure of the network similar to FFNs. The layered structure enables the ease of computation at each level. Also, clear demarcation between the input, output and hidden layers could be achieved.

7.2.1.3 NETWORK DESIGN

The network architecture was designed considering all the aspects mentioned above in previous sections. The design process of the DNN involved several stages.

Design Stage 1- Neurons and Layers

The proposed 5-layered network architecture comprised of layers of neurons related to the specialised neuron types found in the toad's brain. The first layer consisted of the retinal cells, which are the input neurons of the network. The second layer of neurons are the thalamic neurons (TH3). The third layer comprises of Pre-tectal T5-1 type neurons. The fourth layer is made up of T5-3 type Pre-tectal neurons. The fifth and final layer encompasses the T5-2 type optic tectum neurons, which form the output layer as shown in Figure 7.1.

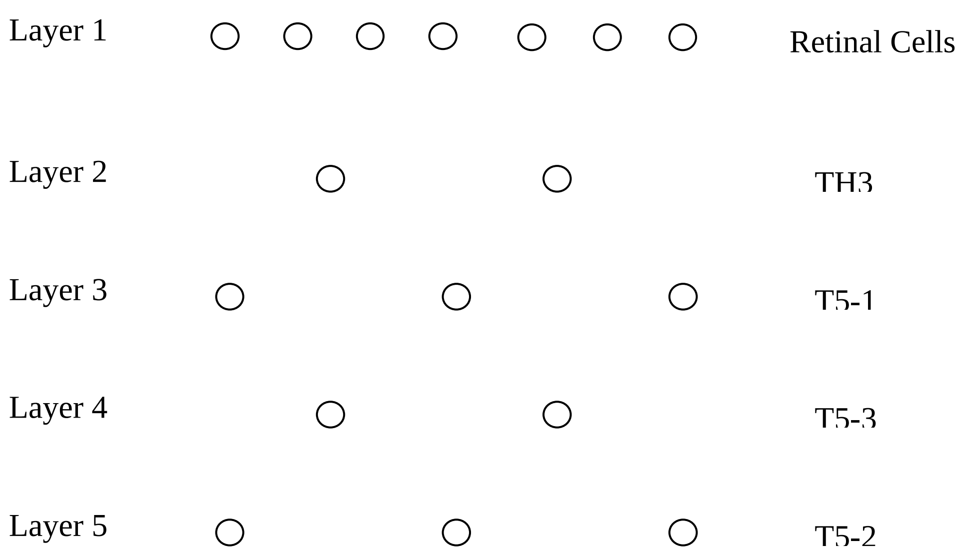


Figure 7.1: DNN Design Stage 1 - Neurons and Layers

Design Stage 2 – Neural Connections

All the neurons in the DNN had the same functionality, but the inter-connections between neurons were not all the same type in the proposed DNN. It was evident from the investigations performed on the toad's visual system that the types of connections between various neurons were very important.

Some of the features of the connections between specialised neurons in the DNN are given below and shown in Figure 7.2.

1. In the toad's brain, every neuron was not connected to every other neuron. The number and type (excitatory or inhibitory) of connections made were specific to certain neurons. This was maintained in the design of the DNN.

2. Connections from TH3 neurons were only made with pre-tectal neurons T5-1 and T5-3. Hence, the final connections allow input to the optic tectum neurons (T5-2), only via the pre-tectal neurons.

3. According to Ewert's proposition, the pre-tectal neurons (T5-3) receive excitatory influences from the optic tectum (T5-2) and are specialised for eliciting the predator avoidance behaviour (as discussed in Chapter 6) .

   Also, the optic tectum neurons (T5-2), responsible for prey-catching behaviour in toads are inhibited by the pre-tectal neurons, T5-3. Hence, this was incorporated in the DNN.

4. Lateral inhibitory connections between TH3 and T5-1 were also included as there is evidence of such connections in the insect eye as well as the toad's visual system.

5. The only backward connections that became part of the network were the connections running from T5-2, optic tectal (OT) neurons to the pre-tectal (PT) neurons, T5-3. These connections were in line with the proposed neuronal interactions to allow prey-catching and predator avoidance in toads.

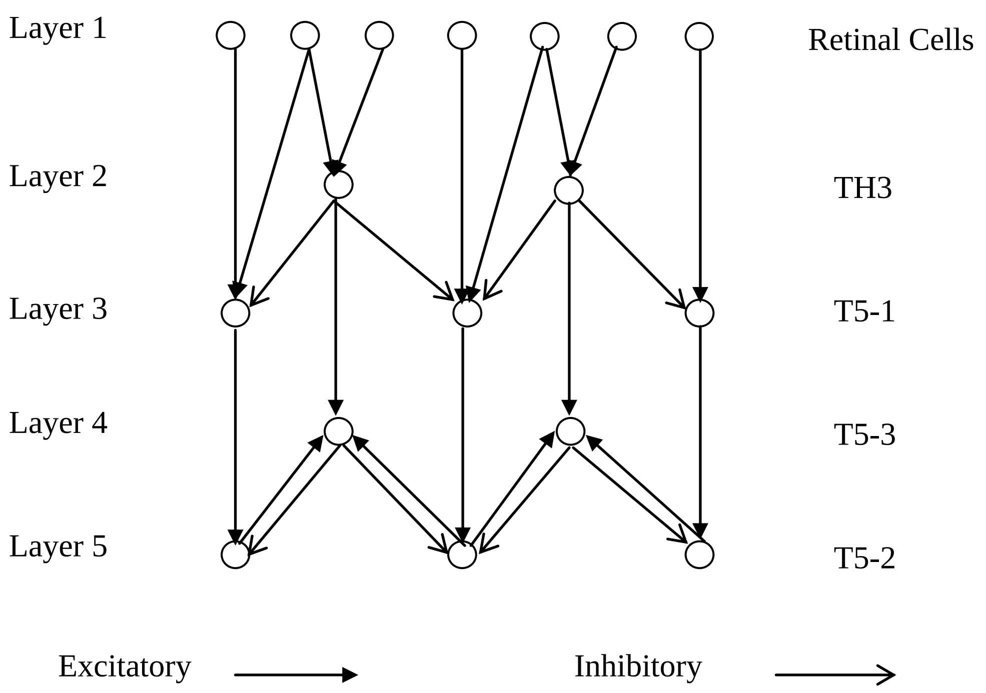


Figure 7.2: DNN Design Stage 2 - Neural Connections

As mentioned earlier, the DNN is not biologically accurate. Some of the connections present in the visual system of toads were discarded in the DNN. Some new connections were also included. For example,

1. The connections between T5-2 neurons within the same layer and the backward connections between T5-3 (layer 4) and T5-1 (layer 3) neurons were omitted from the network architecture. This was done as T5-3 and T5-1 neurons were pre-tectal neurons and the connections between similar types of neurons were considered to be less significant as it does not serve the purpose of sensory filtering and analysis.

2. T5-1′ neurons were not represented in the DNN and hence all connections from T5-1′ neurons to the other specialised neurons that may be present in the biological model are absent in the DNN. This action was taken to reduce the number of pre-tectal neurons and keep the DNN simple to allow for modularity in the network structure.

3. Also, in the Ewert's neural circuit models, connections between the thalamic neurons, TH3, and optic tectal neurons, T5-2 were made directly, unlike the proposed DNN. This was done to allow all connections from TH3 to the Optic tectum to run through Pre-tectal neurons. This was done to improve the filtering process, especially because T5-1′ neurons were absent in the DNN.

4. Excitatory connections between T5-2 to T5-3 neurons, which were absent in the biological model are present in the DNN. They were included to improve feature analysis.

5. In the DNN, there are no connections running from the Optic Tectum to the Anterior Thalamus regions. This is another deviation from the biological model, as it was thought that the presence of such connections might disrupt the top-down sensory filtering process as they form backward connections.

6. Although the retinal cells in the toad were functionally specialised and facilitated different levels of light to be absorbed, the neurons in the first layer of the DNN simply allow the input to be fed to the network and do not perform any function. This was done to reduce the computational complexity and adhere to using neurons of the same form and function as in FFNs.

The divergence from the biological model, defined either by the inclusion of new connections or deletion of known connections and altering the functionality of specific neurons and retinal cells, was crucial for the following reasons.

1. Reduce the number of backward connections to allow top-down sensory filtering
2. Maintain simplicity
3. Accommodate ease of training

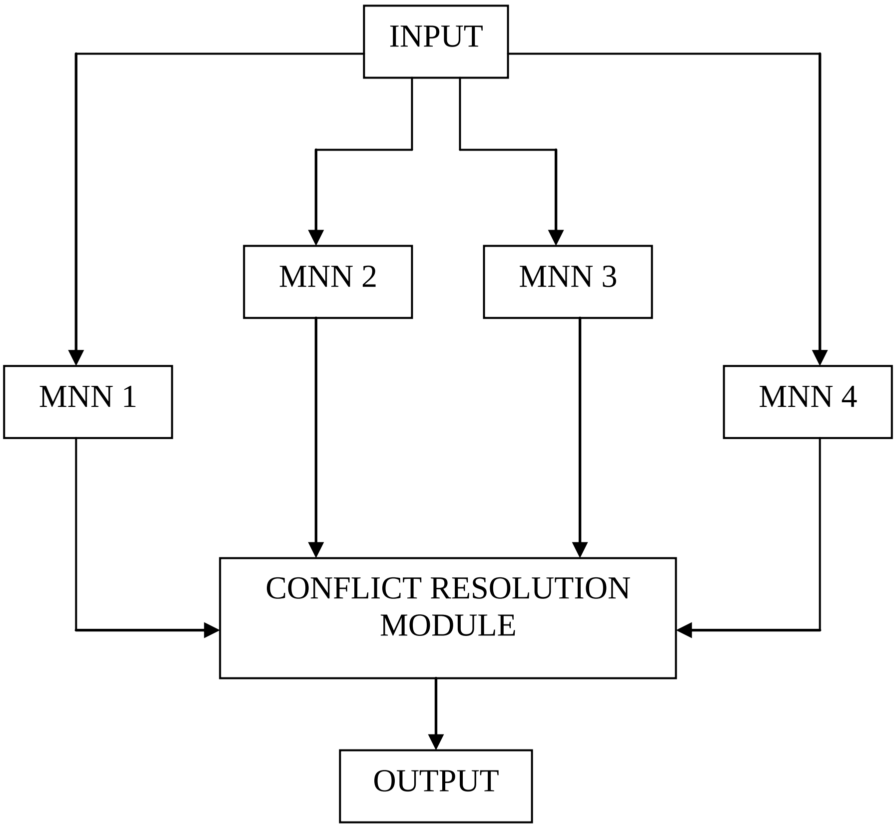
4. Improve sensory filtering
5. Enhance feature analysis
6. Alleviate problems that may thwart modularity
7. Include standards of ANNs (architecture and training) in DNNs to improve its universality

Design Stage 3 – Modularity

Modularity is one of the fundamental characteristics observed in Biological Systems. Most Engineering Systems also incorporate modularity in their design. The investigations and experiments conducted on Modular Neural Networks (Chapter 4) have shown that modularity is vital and to a large extent helpful in ANNs despite the drawbacks in terms of training as mentioned in Chapter 5. Hence effort was made to embrace modularity in the DNN.

Many ways of achieving the modularity were considered. The most common approach used to accomplish modularity in ANNs is by including a conflict resolution network, which receives information from several modules of specialised neural networks as shown in the Figure 7.3.

Experiments conducted on traditional ANNs as part of the research programme showed that Conflict Resolution Modules in conventional methods have limited success. It can be observed from Figure 7.3 that there is no interaction between the specialised modules of ANNs - MNN 1, MNN 2, MNN 3 and MNN 4. Biological substantiation suggests that interaction between specialised modules occurs before information is passed to other specialised units adapted for conflict resolution.



.

Figure 7.3: Schematic of a Modular Neural Network

If communication were to be established between specialised MNNs,

1. Through what method would the output of one MNN be fed to another MNN?
2. How should the information be handled by the receiving MNN?
3. What type of training method needs to be engaged by the MNN when dealing with unfamiliar information from other MNNs?
4. How does introduction of new data affect original specialisation of the MNN?

5. What technical/ mathematical purpose would the transfer of data between MNNs achieve?
6. What would the final output of dedicated MNNs be after all dealings with other MNNs?
7. Is a Conflict Resolution Component necessary in such systems?

Implementation of interaction between functionally specialised units is impractical in traditional ANNs due to the limitations inherent in long-established ANN architectures and training methods (fully connected networks and back propagation of error) as mentioned in previous chapters. Hence, it was thought that rather than establish exchanges between separate specialised DNNs, it would be easier and feasible to include communication links between neurons in the outermost layers 4 and 5 as shown in Figure 7.2, which can be redrawn as shown in Figure 7.4.

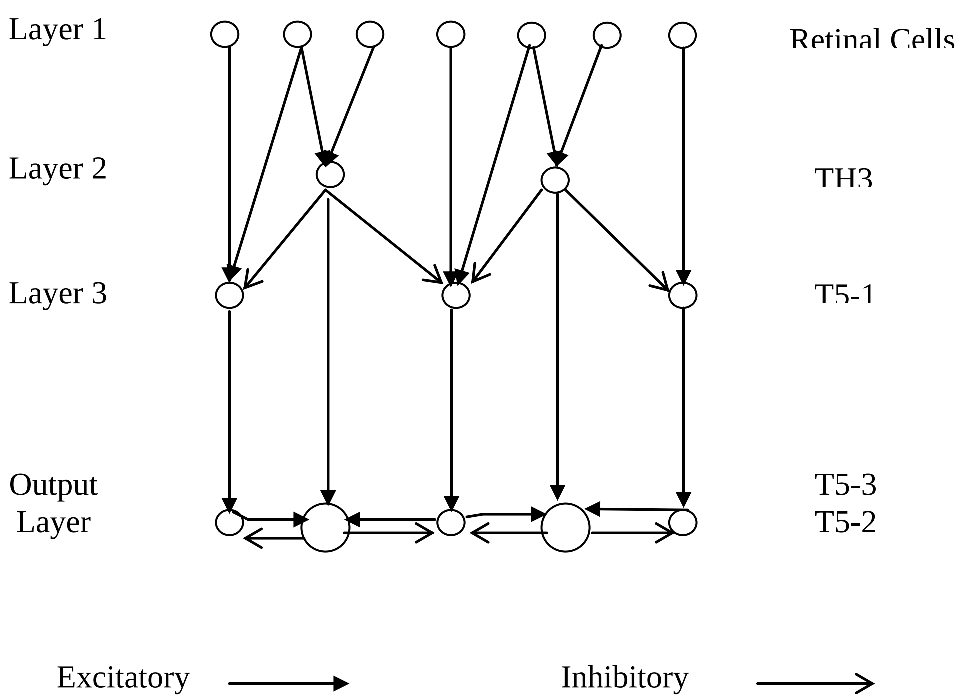


Figure 7.4: DNN Design Stage 3 - Neural Connections Reinterpreted

The neural connections in the DNN are reinterpreted in Figure 7.4 to illustrate that the bilateral connections are among the outermost neurons that usually constitute the output layer of any structure of ANNs. Mutual contacts between output neurons are not customary in established ANNs.

Another aspect of modularity that was intentionally maintained by the systematic selection of the neurons and their connections during the development of the DNN is illustrated in Figure 7.5.

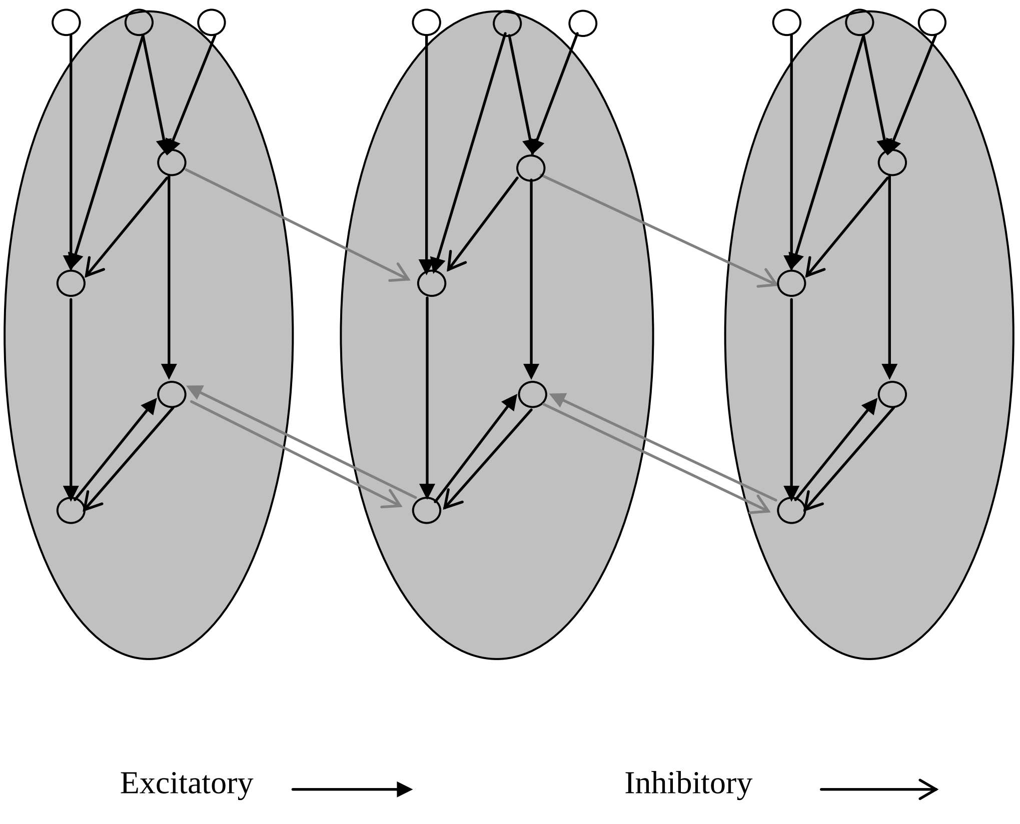


Figure 7.5: DNN Design Stage 3 – Illustration of Modularity

Figure 7.5 shows those specific groups of neurons forming a definite structure by means of their connections making up a module. These modules of neurons are interconnected through lateral connections. Several such modules can be concatenated through sideways connections to accommodate any number of inputs through its first layer. Figure 7.6 demonstrates a DNN with 25 input neurons.

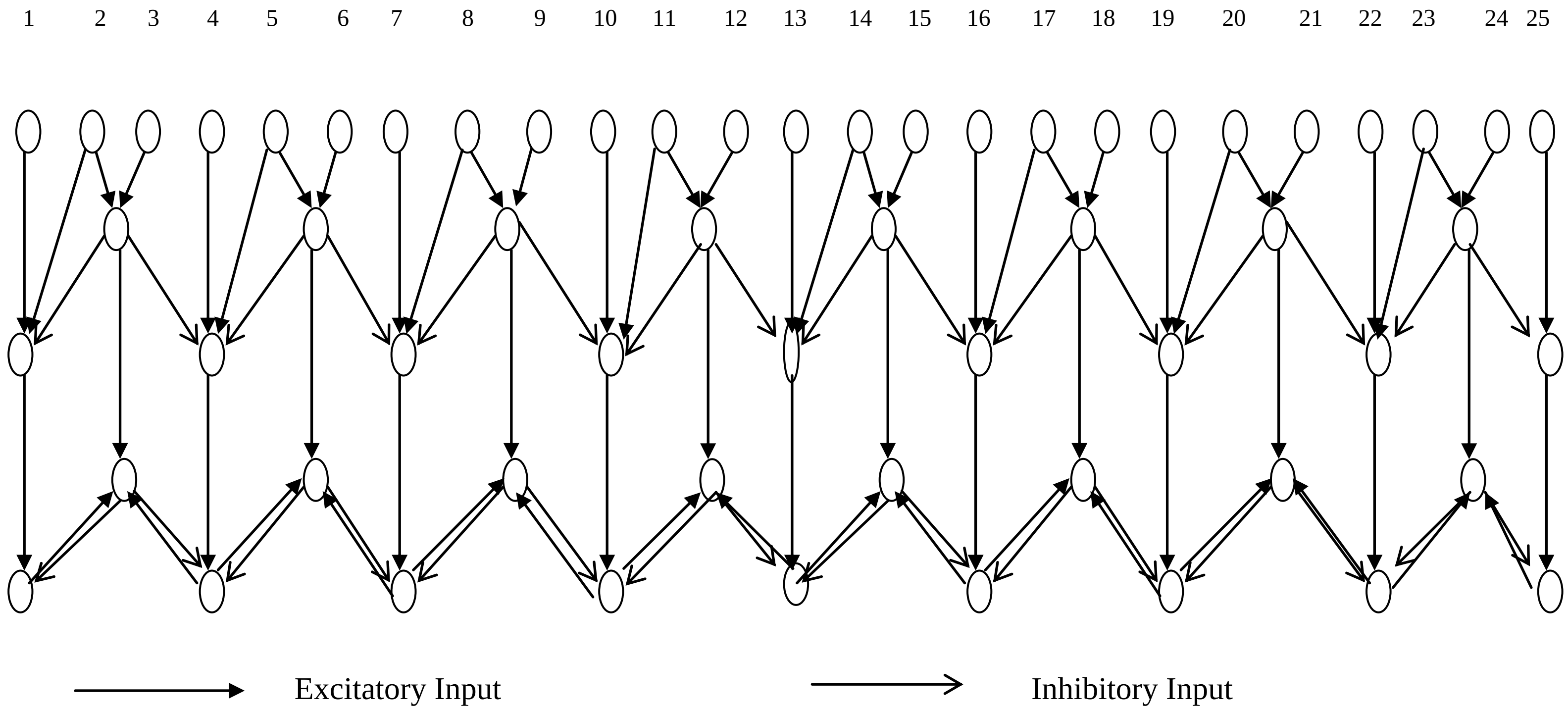


Figure 7.6: The Distributed Neural Network (DNN)

7.3 SUMMARY

In this research programme, extensive investigation conducted in Biological Vision Systems and ANNs has paved the way to the invention of a unique network called the Distributed Neural Network (DNN).

The features of the architecture of DNN are:

1. The DNN embodies the qualities of traditional ANNs and the Toad's Visual System.
2. The DNN included aspects of MNNs that were intended to promote modularity and generalisation in ANNs.
3. The DNN is modular in structure.
4. The DNN incorporates bilateral interaction between neurons in the output layer.
5. The DNN while being unique does not deviate from the standard principles of ANNs.
6. The design of DNN was laid on the foundations of adaptability, standardisation and universality.

Chapter 8 presents the Interpretation of the Input/ Output Data and the Implementation of the DNN using Evolutionary Algorithms for Reinforcement Learning.

CHAPTER 8: IMPLEMENTATION OF THE DISTRIBUTED NEURAL NETWORK (DNN)

8.1 INTRODUCTION

The creation of a unique architecture of the biologically inspired Distributed Neural Network (DNN) developed in this research programme was discussed in Chapter 7.

As the configuration of the DNN is different from traditional ANNs, the most favourable methods to implement the DNN successfully had to be examined. The objective was to develop a system that could be employed for obstacle avoidance in robotics.

The main aspects that were considered to bring the DNN to fruition are:

- The Representation of Input Data
- The Representation of Output Data
- The Learning Algorithm to train the DNN

Effort was made to ensure that all phases of training the DNN were biologically inspired while maintaining standard ANN practices. The design of DNN’s structure and training endeavoured to develop a system that is adaptable for other application areas other that Robotics. This chapter explains the methodical development of DNN’s training algorithm.

8.2 REPRESENTATION OF INPUT

Inspiration was once again drawn from the toad's visual system in the interpretation of input data and presentation to the DNN. Horizontal and vertical features were chosen as features for the training set. The features selected are also a good representation of obstacles in a Robot's path.

The training data consisted of three classes of features based on the type and location in a 5x5 grid. The spatial position of the individual classes of features was considered to be of importance in a recognition problem such as obstacle avoidance.

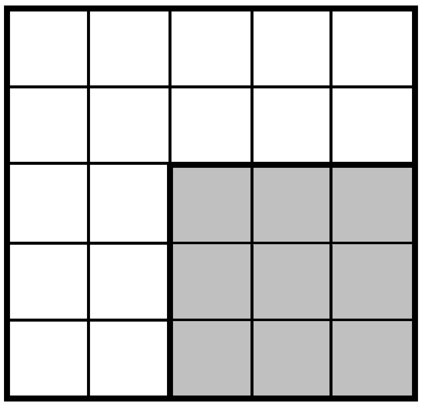
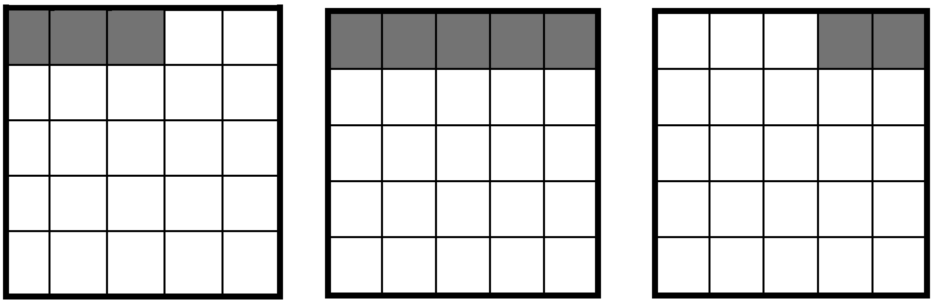


Figure 8.1: Partition of Input Space

The 5x5 grid was partitioned as shown in Figure 8.1 into two main regions. The horizontal and vertical features appear in these regions. The definition of various classes of features is based on the type and spatial position of the features in the 5x5 grid.

Based on the criteria mentioned above, the features for each class were selected for training as shown in Figures 8.2, 8.3 and 8.4 below. They are,

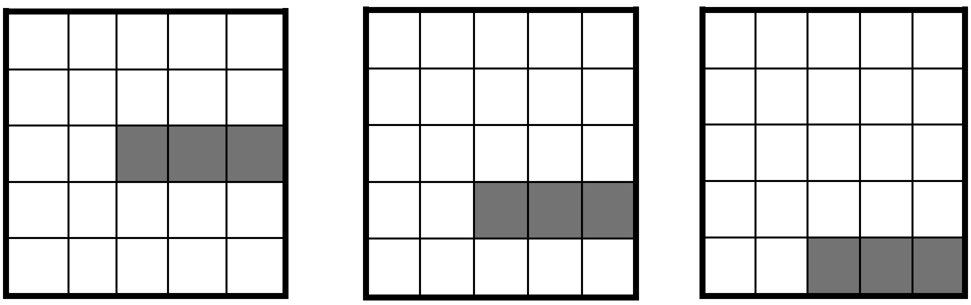
1. Class 1 – Horizontal features representing overhangs



Feature 1 **Feature 2** **Feature 3**

Figure 8.2: Class-1 Features

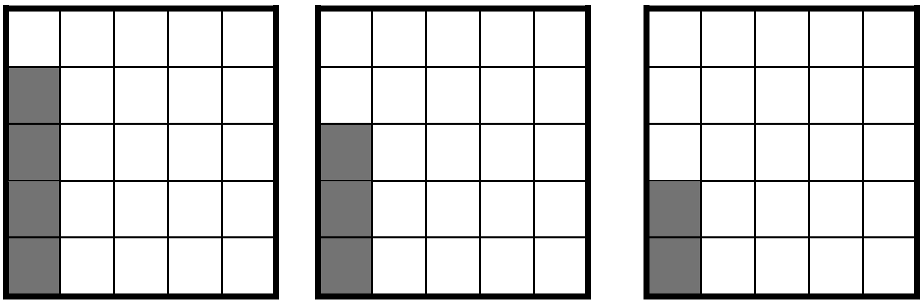
2. Class 2 – Horizontal features representing protruding obstacles in the right hand corner covering a certain area



Feature 1 **Feature 2** **Feature 3**

Figure 8.3: Class-2 Features

3. Class 3 – Vertical Features representing obstacles alongside the left hand wall



Feature 1 **Feature 2** **Feature 3**

Figure 8.4: Class-3 Features

All through the research programme, effort was made to adopt current methods in ANNs. Thus, inspiration from ANNs was central in the presentation of input data to the DNN. As in most ANNs, a binary version of the input data was presented to the DNN as shown in Figure 8.5, 8.6 and 8.7.

1. Class 1 Features

1	1	1	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

Feature 1

1	1	1	1	1
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

Feature 2

0	0	0	1	1
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

Feature 3

Figure 8.5: Binary Representation of Class-1 Features

2. Class 2 Features

0	0	0	0	0
0	0	0	0	0
0	0	1	1	1
0	0	0	0	0
0	0	0	0	0

Feature 1

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	1	1	1
0	0	0	0	0

Feature 2

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	1	1	1

Feature 3

Figure 8.6: Binary Representation of Class-2 Features

3. Class 3 Features

0	0	0	0	0
1	0	0	0	0
1	0	0	0	0
1	0	0	0	0
1	0	0	0	0

Feature 1

0	0	0	0	0
0	0	0	0	0
1	0	0	0	0
1	0	0	0	0
1	0	0	0	0

Feature 2

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
1	0	0	0	0
1	0	0	0	0

Feature 3

Figure 8.7: Binary Representation of Class-3 Features

8.3 REPRESENTATION OF OUTPUT

The DNN was used to train three classes of features as mentioned above. Each class consisted of three features in the training set. The output representations for the three classes of features are as shown in the Table 8.1.

CLASS (All features)	OUTPUT PATTERN
1	1 0 0
2	0 1 0
3	0 0 1

Table 8.1: Representation of Output Patterns

This varies from FFNs trained using Back Propagation algorithms. The two main deviations are:

1. Output patterns in the DNN correspond to each class and not every feature within the class.
2. More than one committed neuron is assigned for each class in the DNN.

Dedicated output neurons in the DNN selected for the three classes of obstacles and the presentation of input to the DNN are shown in Figure 8.5.

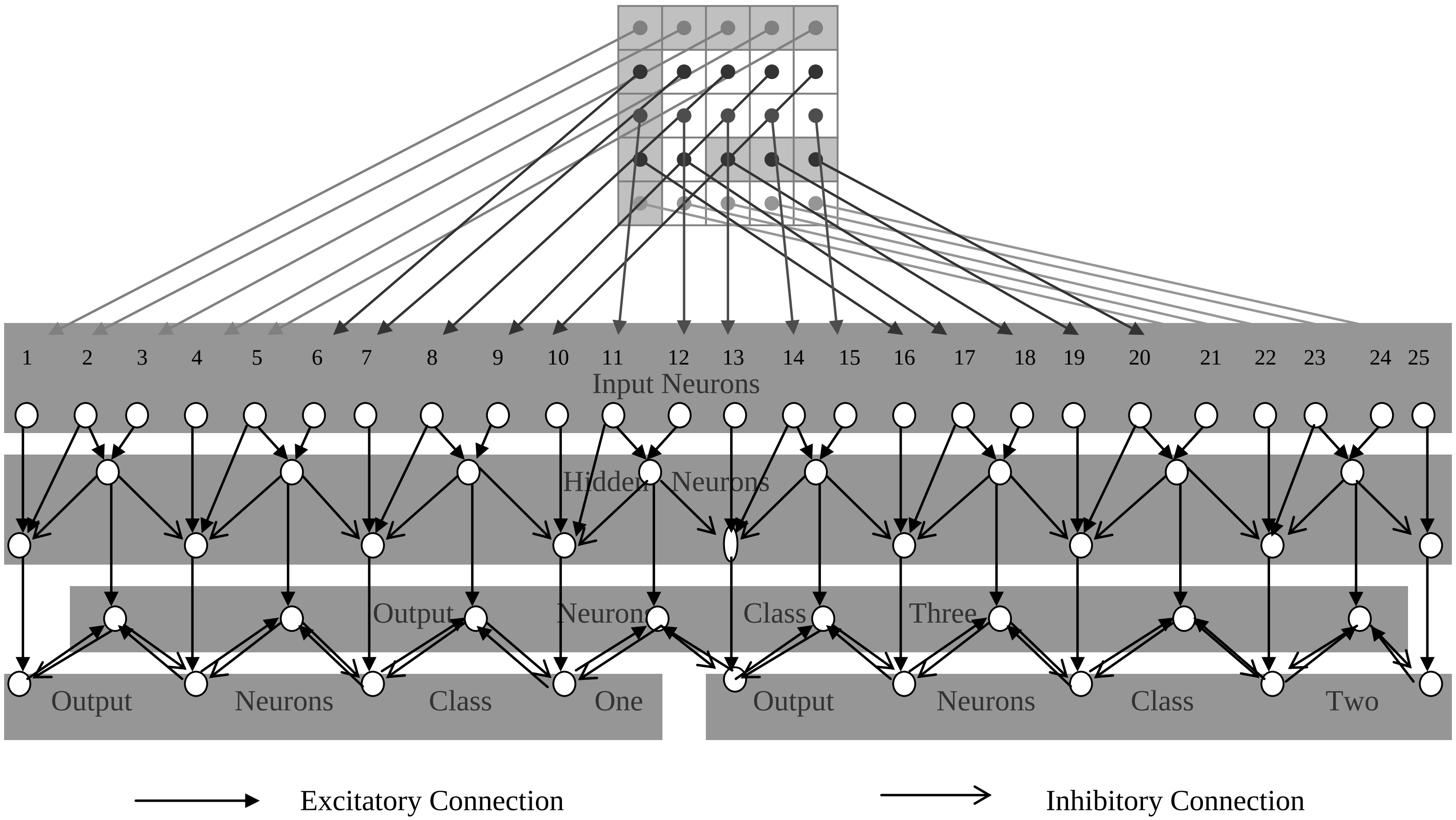


Figure 8.5: Input Data to The Distributed Neural Network (DNN)

Table 8.2 shows the allocation of more than one dedicated neuron for each class. The choice behind the dedicated neurons in this research programme was based on the representation of the features related to the Toad's Visual System, where Horizontal Features represented Prey and Vertical Features represented Predator. Hence, the vertical features have a higher number of dedicated neurons allocated to it when compared to the Horizontal Features. The relevant output pattern is allocated based on relative activation of the dedicated neurons. Hence an Output Pattern '1 0 0', depicts that Class-1 dedicated neurons are highly activated compared to Class-2 and Class-3 dedicated neurons.

CLASS	FEATURE	NUMBER OF DEDICATED NEURONS	OUTPUT PATTERN
1	1	4	1 0 0
	2		
	3		
2	1	5	0 1 0
	2		
	3		
3	1	8	0 0 1
	2		
	3		

Table 8.2: Illustration of the Association between Features within a Class, Dedicated Neurons and the Output Pattern

As there are several dedicated neurons for various classes of features; it was imperative to use a training method where the activation of the dedicated neurons for various classes of features and their relationship with each other is significant.

8.4 LEARNING ALGORITHM

The following sections give an explanation of the main aspects of the learning method developed called FUSION, which is a combination of several learning methods including Evolutionary Algorithms for Reinforcement Learning (EARL).

8.4.1 REINFORCEMENT LEARNING

After considering several training methods, Reinforcement Learning proposed by Barto and Sutton was chosen for the purpose of training the network [134].

The reasons for the choice of this training method are,

1. Reinforcement Learning is biologically plausible.
2. Also, it is a good compromise between supervised and unsupervised learning as it involves feedback, which is qualitative and not quantitative. The fitness used in Reinforcement learning does not directly quantify the deviation of the obtained results from those required. The fitness provides feedback regarding the performance of a solution compared to other solutions.

3. The feedback is not directly used to change any parameters of the system (unlike, Back Propagation, where the error is used to directly change the connection weights of the network).
4. The feedback is a fitness or goodness value usually represented by a real number. Any number of factors could be used to substantiate an outcome, which increases the flexibility of the training method for a novel architecture such as the DNN.
5. Reinforcement can be implemented using Evolutionary Algorithms, which are compatible with ANNs.

8.4.2 EVOLUTIONARY ALGORITHMS

Evolutionary Algorithms (EAs) are optimisation methods based on the Theory of Evolution in Nature as proposed by Charles Darwin. Optimal solutions (called chromosomes) represented by a string of relevant factors (called genes) are drawn from a search space (population of chromosomes) based on the performance of the solution for a specific problem. Operators (called mutation and crossover) are used to generate new solutions.

Genetic Algorithms discovered by Holland [135], Evolutionary Strategies developed by Rechenberg and Schwefel [136] and Genetic Programming proposed by Koza [137] can be collectively grouped under the umbrella, Evolutionary Algorithms. Although, these methods differ by the representations (real valued, integer or binary) of solutions and the choice of operators (mutation, crossover or both) used to evolve new solutions to a problem, their basic operation is similar. EAs are often used to train Artificial Neural Networks. They are mainly used for the evolution of best possible network architectures and the strength of connection weights between various neurons.

Evolutionary Algorithms (EAs) are probably the best method by which evaluative feedback using Reinforcement Learning (RL) could be provided to an Artificial Neural Network. Such a method that uses a combination of EAs and RL is called Evolutionary Algorithm for Reinforcement Learning (EARL) [138].

8.4.3 TRAINING METHOD 'FUSION'

One of the main characteristics of FUSION that is unique to this research programme is the management of rewards or fitness.

In conventional methods where EAs are used to train ANNs, the error across all the output neurons is calculated based on the 'Actual Output' and the 'Desired Output' (similar to Back Propagation used in FFNs). The fitness of the chromosome (which usually comprises the connection weights of the FFN) is dictated by the error, which varies inversely to it.

The distinctiveness of FUSION used in this research programme is the exploitation of numerous factors and their combinations in determining the rewards. The reward mechanism was based on three factors, which are listed below.

1 The classification of the feature

One of the most important factors is the classification of the pattern itself. This is based on any dedicated neuron with the highest activation.

For example, consider the features 'Class-1', 'Class-2' and 'Class-3'. The recognition of a pattern representing 'Class-3' as 'Class-2' is undesirable, as Class-3 represent vertical features whereas Class-2 represent horizontal features. Hence, a very low fitness value (zero) is associated with that particular chromosome. However, if the 'Class-2' pattern is recognised as 'Class-1', it is equally bad, as possible damage caused by the impact of collision with obstacles of type Class-2 is higher than the effect of damage caused by an overhang of type Class-1.

2 The highest activation value among the rival neurons

The highest activation value among the specialised neurons was considered important because the more dominant a dedicated neuron is in recognition of its class of features, the more tolerant the DNN becomes to noise. In other words, the fitness criteria being adapted encourages unparalleled dominance of dedicated neurons to its particular class of features. The highest activation of specialised neurons was considered to be relative to each other. A method by which the dominance of winning neurons was encouraged was by rewarding states where the rival neuron activation was low.

3 *The number of rival neurons with activation ≤ 0.5*

Another factor considered was the number and activation of all dedicated neurons of other classes of features when the DNN was being trained for a specific class. The fewer active opposition neurons (dedicated neurons of other classes other than the class that the DNN is being trained for) with very low activation values would give the corresponding chromosome a higher fitness value.

The reward systems used for the three classes of features are given in Tables 8.3, 8.4 and 8.5. The rewards are a function of factors that are based on the activation of the number of dedicated output neurons as the classification is based on the activation of these neurons.

The Re*ward*, *R is given by*

$$R = f\left(\omega_c,\ \gamma_c^a,\ \gamma_c^n\right)$$

$$\begin{aligned} \textit{where}\quad \omega_c &= \textit{Winning Class} \\ \gamma_c^a &= \textit{Range of Highest Rival Class Activation} \\ \gamma_c^n &= \textit{Number of Rival Class Neurons with Activation} \leq 0.5 \end{aligned}$$

It can be noted from the equation above that the activation value and the number of neurons with certain activation are important factors in the reward scheme chosen. Hence, the class to which the highest number of dedicated neurons is allocated is most likely to have a higher reward allocated to it.

Neurons with Highest Activation ω_c	Range of Highest Class-3 Neuron Activation, γ_c^a	Number of Class-2 Neurons with Activation $\gamma_c^n <= 0.5$	Reward R
Class-1	< 0.5	≥4	42
		=3	40
		=2	38
		≤1	36
	(0.5, 0.6)	≥4	34
		= 3	32
		=2	30
		≤1	28
	(0.6, 0.7)	≥4	26
		= 3	24
		= 2	22
		≤1	20
	(0.7, 0.8)	≥4	18
		= 3	16
		= 2	14
		≤1	12
	≥ 0.8	≥4	10
		= 3	8
		= 2	6
		≤1	4
Class-2	All Conditions	All Conditions	2
Class-3	All Conditions	All Conditions	0

Table 8.3: Reward Scheme for Class-1 Features Recognition

Neurons with Highest Activation ω_c	Range of Highest Class-3 Neuron Activation, γ_c^a	Number of Class-1 Neurons with Activation γ_c^n<=0.5	Reward R
Class-2	< 0.5	= 5	42
		=4	40
		=3	38
		≤2	36
	(0.5, 0.6)	=5	34
		=4	32
		=3	30
		≤2	28
	(0.6, 0.7)	=5	26
		=4	24
		=3	22
		≤2	20
	(0.7, 0.8)	=5	18
		= 4	16
		= 3	14
		≤2	12
	≥ 0.8	=5	10
		= 4	8
		= 3	6
		≤2	4
Class-1	All Conditions	All Conditions	2
Class-3	All Conditions	All Conditions	0

Table 8.4: Reward Scheme for Class-2 Features Recognition

Neurons with Highest Activation ω_c	**Range of Highest Class-1 & 2 Activation,** γ_c^a	**Number of Class-1 & 2 Neurons with Activation** γ_c^n **<=0.5**	**Reward R**
Class-3	**< 0.5**	**≥8**	**70**
		=7	**68**
		=6	**66**
		=5	**64**
		=4	**62**
		=3	**60**
		<3	**58**
	(0.5, 0.6)	**≥8**	**56**
		=7	**54**
		=6	**52**
		=5	**50**
		=4	**48**
		=3	**46**
		<3	**44**
	(0.6, 0.7)	**≥8**	**42**
		=7	**40**
		=6	**38**
		=5	**36**
		=4	**34**
		=3	**32**
		<3	**30**

Neurons with Highest Activation ω_c	Range of Highest Class-1 & 2 Activation, γ_c^a	Number of Class- 1 & 2 Neurons with Activation γ_c^n <=0.5	Reward R
Class-3	**(0.7, 0.8)**	**≥8**	**28**
		=7	**26**
		=6	**24**
		=5	**22**
		=4	**20**
		=3	**18**
		<3	**16**
	≥ 0.8	**≥8**	**14**
		=7	**12**
		=6	**10**
		=5	**8**
		=4	**6**
		=3	**4**
		<3	**2**
All Other Conditions	**All Conditions**	**All Conditions**	**0**

Table 8.5: Reward Scheme for Class-3 Features Recognition

Although, rewards were allocated for almost all possible states, some anomalies were included in the reward proposal. For example,

1. Rewards were not assigned to all possible states. For example in Table 8.4, rewards are assigned to conditions such as highest activation of rival neurons, γ_c^a in the range, (0.5 to 0.6) and (0.6 to 0.7). But no indication is

provided for γ_c^a = 0.6 or 0.7. However, there is a reward for, 'All Other Conditions', which is zero.

2. In Table 8.4, there is a reward allocation for the condition with the number of neurons that belong to rival Class-1 neurons γ_c^n = 5. There are only 4 dedicated neurons for Class-1 features. This condition might not have been used before, but it might be encouraged to drive the learning algorithm towards a state where it might align other neighbouring neurons to work towards Class-1 recognition.
3. In Table 8.4, when the condition γ_c^a < 0.5 is given for opposition Class-3 neurons during recognition of Class-2, there is no evidence to suggest that it is desirable to have γ_c^a as low as possible. The best-case scenario would be γ_c^a = 0, but this is not explicitly stated.

Such fuzziness in the reward scheme was introduced in order to help:

1. FUSION to explore new possibilities that may be better and more promising.
2. FUSION achieve the true meaning of 'Learning' that involves development of new reward schemes to train the DNN.
3. Improve the DNN's tolerance to noise as FUSION explores and learns to exploit to achieve better rewards.
4. To ultimately enhance DNN's ability to generalise.

The characteristics of FUSION that were used to train the DNN in the research programme are:

1. Several criteria that included individual neurons and their activation were used to draw rewards.
2. Rewards for all possible states of neurons were not explicitly known to the DNN.
3. FUSION explores other avenues that are not given in the reward system. Anomalies are included in the reward scheme to enhance exploration.
4. FUSION has limited knowledge of the input-output relationship, as it does not involve calculation of errors.
5. FUSION has enough freedom to explore because of fewer constraints in the reward scheme.
6. Another option left for FUSION to probe was the selection of optimal number of specialised neurons that would achieve the best classification results. The intention of not including the factor regarding the number of highly excited neurons within a certain class was to examine the result that FUSION might choose to understand more about the role of dedicated neurons in ANNs.
7. FUSION embodies aspects of
    - Supervised Learning (allocation of dedicated neurons for each class)
    - Unsupervised Learning (more than one dedicated neuron in each class)
    - Reinforcement Learning (reward mechanism for various states)
    - Fuzzy Logic (vague, ambiguous and imprecise anomalies)

FUSION was used to train the network to recognise various classes of features. The following process was implemented to evolve the optimal DNN.

Chromosomes

Sixteen chromosomes, each consisting of 98 genes representing strengths of all the network connections was randomly generated as shown in Table 8.6. Every chromosome represented a unique neural network with distinctive architecture and connection strengths. The structure of the DNN could be altered if some of the connection weights are set to zero.

Chromoseome c	GENE (g)							
	1	2	3	4	5	....	97	98
1	$w_{1,1}$	$w_{1,2}$	$w_{1,2}$	$w_{1,4}$	$w_{1,5}$	....	$w_{1,97}$	$w_{1,98}$
2	$w_{2,1}$	$w_{2,2}$	$w_{2,2}$	$w_{3,4}$	$W_{2,5}$	....	$w_{2,97}$	$w_{2,98}$
3	$w_{3,1}$	$w_{3,2}$	$w_{3,2}$	$w_{3,4}$	$w_{3,5}$	....	$w_{3,97}$	$w_{3,98}$
⋮	⋮	⋮	⋮	⋮	⋮	⋮ ⋮ ⋮ ⋮	⋮	⋮
15	$w_{15,1}$	$w_{15,2}$	$w_{15,2}$	$w_{15,4}$	$w_{15,5}$	....	$w_{15,97}$	$w_{15,98}$
16	$w_{16,1}$	$w_{16,2}$	$w_{16,2}$	$w_{16,4}$	$w_{16,5}$	....	$w_{16,97}$	$w_{16,98}$

Table 8.6: Table showing the Chromosomes, weights and genes

Each weight $w_{c,g}$ is a real valued number randomly generated as given below using the random number generator function rand ( ) in C++.

$$w_{c,g} = \left(\left(\left(\frac{(float)\ rand\ ()}{RAND\_MAX} \right) - 0.07 \right) * 2 \right)$$

rand ( ) generates values in the range [0, RAND_MAX]. RAND_MAX is 32767. Hence the weights generated can have a gene, which is 0 or negative as shown in Table 8.6. The genetic operators also have a significant role in the final outcome of the weights.

16	$w_{16,1}$	$w_{16,2}$	$w_{16,2}$	$w_{16,4}$	$w_{16,5}$	....	$w_{16,97}$	$w_{16,98}$
16	0	-0.2	1	0.89		....	0	0

Table 8.6: Table showing the weights including negative and zero

Each network (chromosome) is tested for its recognition capabilities. A cumulative fitness score (accumulated across all features in the three classes) was derived based on its performance. Three separate reward mechanisms were used for the three classes of features. The reward scheme is given in Tables 8.3, 8.4 and 8.5.

Crossover

Tournament Selection and single point crossover were used in FUSION. Crossover was applied every generation. The following procedure chooses 4 random pairs of chromosomes from the best half of the population,

```
//Choosing the mates and verifying that they are not the same
 mate1[chrnum] = random ( CHR / 2 );
 do
 {
      mate2[chrnum] = random ( CHR/2 );
 }while ( mate1[chrnum] == mate2[chrnum] );


//choosing a random crossover point
cp =random (GENE);
```

where

chrnum	Number of pairs selected for reproduction
mate1[4]	Parent 1
mate2 [4]	Parent 2
cp	random crossover point
GENE	number of genes
CHR	total number of chromosomes

The choice of random points of crossover every generation is advantageous for better exploration in the search space.

Mutation

An unusual mutation operator was used in FUSION, where all weights were randomly changed by a small random amount once in every 100 generations. This was done to improve its ability to generalise. Also, the noise tolerance was expected to improve. This is not standard practice in ANNs. The following procedure was used to randomly change the weights, $w_{c.g}$.

```
k = random (180);
```

$$w_{c.g} = w_{c,g} - (\cos(k) * 39.5);$$

Noise

Random noise was added to the input every 5 generations.

```
If (generation % 5 == 0)
{
       x = random(100);
       y = ((float) random (100) / 1000.0);

       if ( (x<50) && (input[p] ==0)
       {
            input[p] = input[p] + y;
       }
       else
       {
            input[p] = input[p] - y;
       }
}
```

The newly evolved networks (chromosomes) obtained after application of crossover and mutation operators are again tested for their performance and the cycle continued until an optimum solution was reached, which was based on a desired fitness.

8.5 IMPLEMENTATION IN C++

Implementing the DNN involved the following phases.

1. Presentation of classes of features as input to the DNN.
2. Generation of random chromosomes.
3. Calculation of activation of neurons, which was the weighted sum of inputs passed through the sigmoid function to produce an output that lies between 1 and 0.
4. Development of a method to derive the corresponding output features based on the activation of the output neurons.
5. Application of FUSION and the reward scheme to train the DNN.
6. Application of Mutation and Crossover operators to evolve new chromosomes until an optimal solution (DNN) is reached that successfully recognises all classes of features.

All stages involved in the training the DNN were implemented in software using C++. The pseudo code is given in Appendix VI.

8.6 SUMMARY

This chapter discussed the implementation of DNN using FUSION and the methodology of fitness evaluation. The allocation of output neurons for the three main classes of features was also given in this chapter. Explanation of the procedure, ‘FUSION’ and its implementation to fulfil the purpose of the research programme are discussed in this chapter.

Chapter 9 presents the first phase of results obtained when the DNN was tested for the classes of features that formed the training set. New features and noisy features were also used to test the DNN.

CHAPTER 9: TRAINING AND TESTING THE DNN

9.1 INTRODUCTION

The DNN was an amalgamation of ideas drawn from Biology and ANNs. The rationale behind the choice of various decisions made was also explained in Chapters 7 and 8.

The fundamental aspects derived, established and implemented from extensive study of the Toad's Visual System and ANNs are:

1. The architecture of the DNN.
2. The input-output representation of training data.
3. The allocation of input neurons and dedicated output neurons in the DNN.
4. The training method 'FUSION'.
5. The fitness criteria and reward mechanism.

The next step was to train the Distributed Neural Network (DNN) for various patterns and test its capabilities.

The chapter discusses training of the DNN, tests conducted and the results obtained. It gives an insight into the nature of the DNN and its intrinsic properties.

9.2 TRAINING THE DNN

The DNN was implemented using C++ as detailed in Chapter 8. This section gives details of the training procedure.

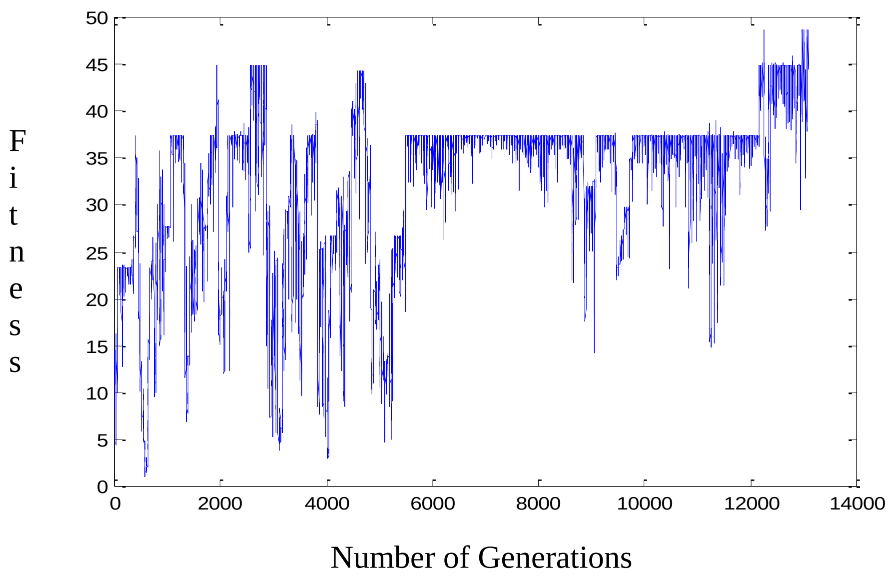
- *Network Architecture:* Distributed Neural Network, which had a novel architecture and distribution of output neurons
- *Neuron Model:* Perceptron with sigmoid activation function
- *Training Method:* FUSION, which employed a unique reward scheme
- *Training Data Set:* 3 Classes of Features
- *Number of Features:* 9 – 3 in each class
- *Classification Criteria:* Activation of anyone of the specified dedicated neurons being greater than or equal to 0.4
- *Maximum Fitness Achievable:* 51.33 (Table 9.1)

Class	Max Possible Reward	Max Possible Average Reward for all Classes
1	42	(42 + 42 + 70)/ 3 = 51.33
2	42	
3	70	

Table 9.1: Illustration of Maximum Possible Reward

- *Highest Fitness Achieved:* 48.6667
- *Number of Training Generations to achieve a fitness of 48.67:* 13083
- *Crossover:* Applied every generation
- *Mutation:* Unique operator based on population randomisation
- *Noise:* Random Noise was added to each pixel in an image every 5 generations to improve the network's ability to generalise
- *Training Time:* The training time on a PC with a 933 MHz Intel Pentium III processor was 74 minutes.

FUSION was used to train the network with the aim of achieving a maximum fitness score of 51.33 consecutively for 20 generations. The persistence of high fitness scores in succession was to introduce consistency of pattern recognition during the training process. Otherwise there was a danger of high fitness scores being achieved early on in the training process without the network reaching its full capabilities of recognising features from the data set.



Graph 9.1: Illustration of the Fitness Achieved

The DNN was trained over 13083 generations to achieve a total fitness of 48.667 as shown in the Graph 9.1. The total fitness is the average fitness across all features.

It is clear from the graph that the fitness of the network for the first 6800 generations varies between many peaks and troughs. Although, a peak fitness of approximately 45 is reached in the initial stages of training, the network goes through other trials due to which the fitness drops.

Several plateaus feature throughout the training process, which may be because of the insistence of more reliable fitness scores over 20 consecutive generations. It is noticeable that the plateaus are more significant in the second half of the training cycle, which can be interpreted that FUSION is learning to produce more consistent outputs.

9.3 TESTING THE DNN

Several tests were conducted following the training of the DNN. The objectives of the test conducted were to:

1. Check the DNN for basic pattern recognition potential
2. Verify the DNN's capabilities for generalisation
3. Test the DNN's tolerance to noise
4. Evaluate advanced image processing features of the DNN

5. Understand the significance of DNN’s architecture and the various neuronal connections
6. Assess the impact of the reward mechanism of FUSION on the DNN
7. Analyse the relevance of DNN’s specific inhibitory and excitatory connections

The network was tested in several stages. They included images that could test DNN’s ability to recognise:

1. Horizontal and Vertical Features from the training set
2. New Horizontal and Vertical Features *(not included in the training phase)*
3. Horizontal and Vertical Features with various levels of noise added *(not included in the training phase)*
4. Combinations of Horizontal and Vertical Features *(not included in the training phase)*
5. Well Defined Digital Images *(not included in the training phase)*
6. Fuzzy Digital Images Representing the Sea Bed *(not included in the training phase)*

Chapter 9 presents the testing stages 1, 2 and 3. Chapter 10 deals with the results achieved when combinations of features were used as test data and Chapter 11 presents the tests conducted with digital Images. The network was never trained with the images or patterns listed against 2, 3, 4, 5 and 6 above. These tests demonstrate the unique aspects of the novel architecture of the DNN.

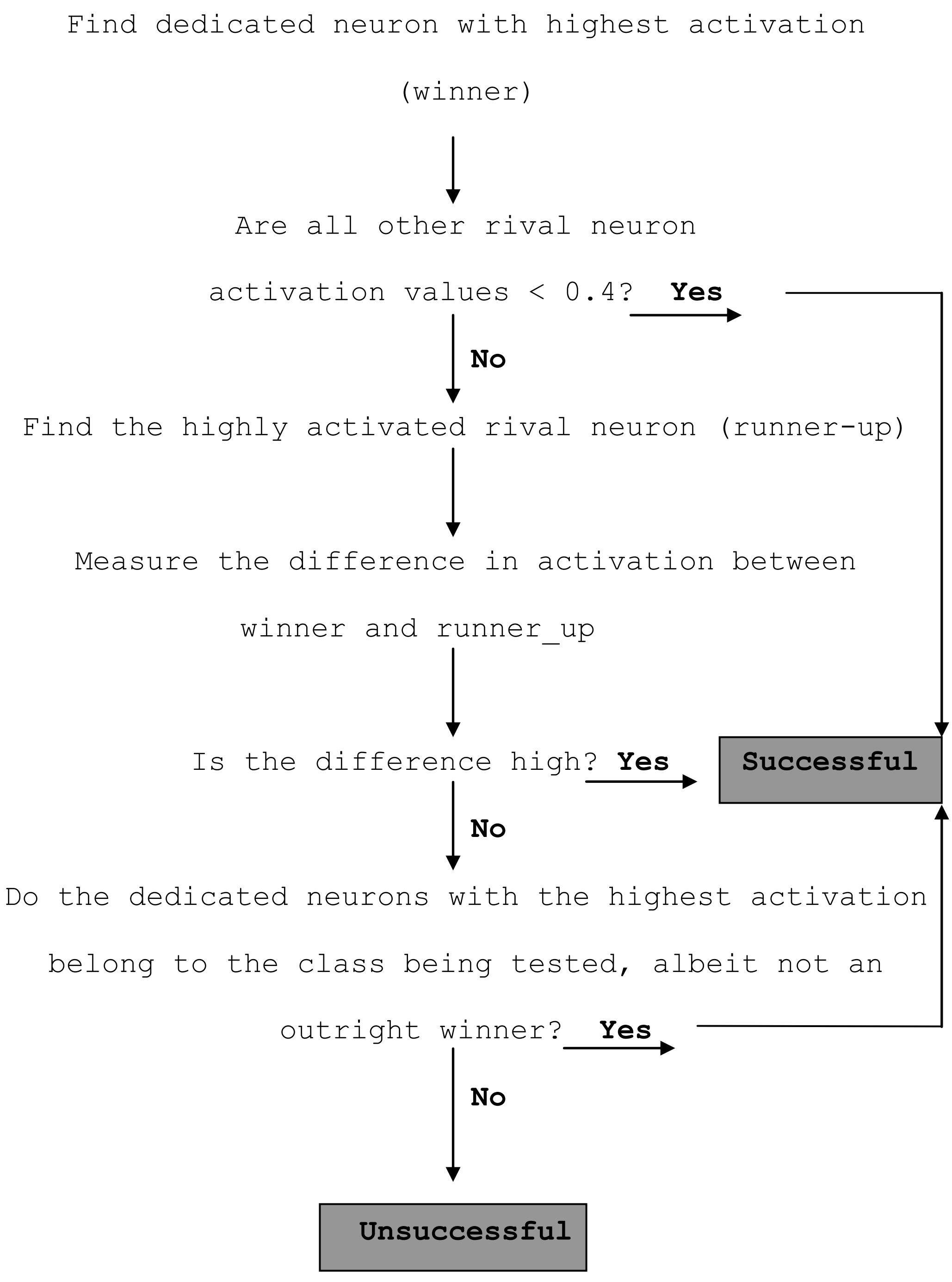
9.3.1 TESTS WITH OBSTACLES FROM TRAINING DATA

The initial tests were conducted using 3 Classes of features from the training set as given in Section 8.2. Any other tests would be futile if the DNN fails to recognise the basic features of the training data.

The following criteria were used to assess the recognition of various classes of features.

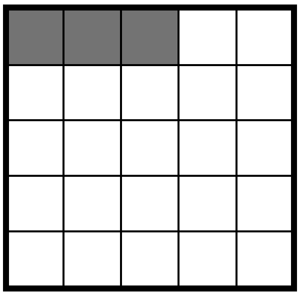
- The DNN is said to have successfully recognised a feature belonging to a particular class, if the corresponding dedicated neurons (winning class) allocated to that particular class show an activation that is greater than or equal to 0.4.
- In cases, where some of the rival neurons (runner-up class) may also show activation ≥ 0.4, then the difference between the activations of the winning and runner-up classes is measured and if it is > 0.3, then the winning class neurons are declared outright winners.
- In cases where there is no outright winner, the first or most important obstacle is taken as the class that has the highest activation of dedicated neurons.
- Outright winners are highly desirable and considered to be most successful.

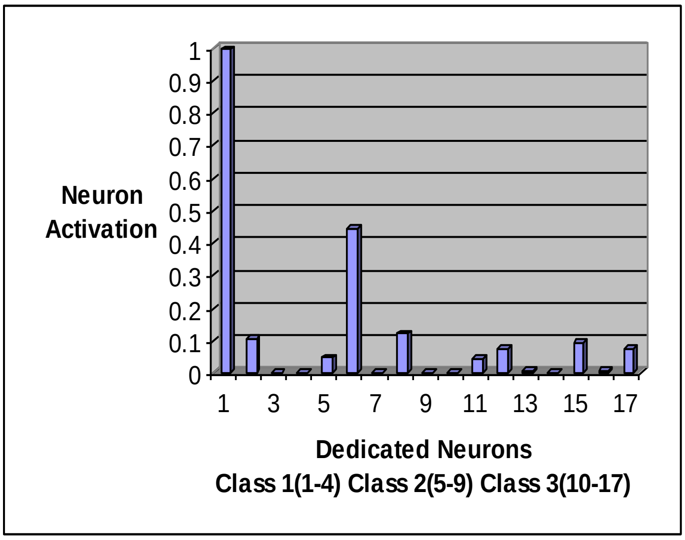
Hence the algorithm for assessing the recognition is:



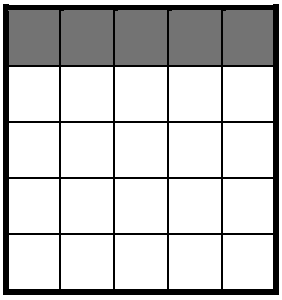
The graphs 9.2 - 9.10 show the activation values of dedicated neurons for all three classes when tested for 9 features from the training set.

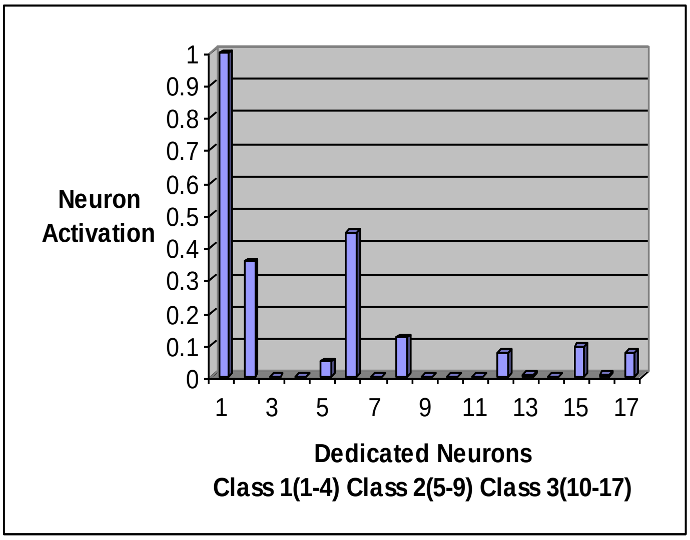
Test with Class 1 Features from Training Set



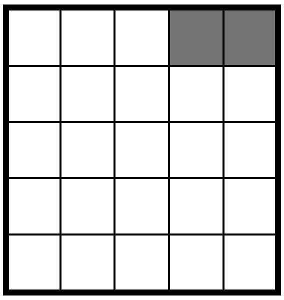


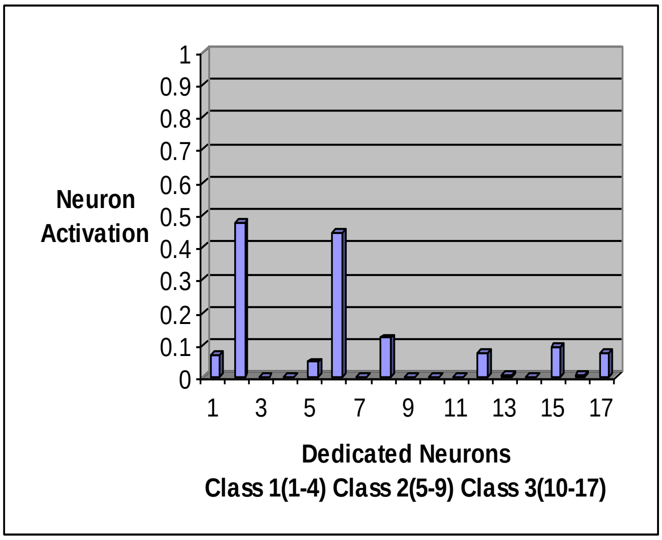
Graph 9.2: Test for Class-1-Feature-1
Illustration of Activation of Various Classes of Neurons





Graph 9.3: Test for Class-1-Feature-2
Illustration of Activation of Various Classes of Neurons





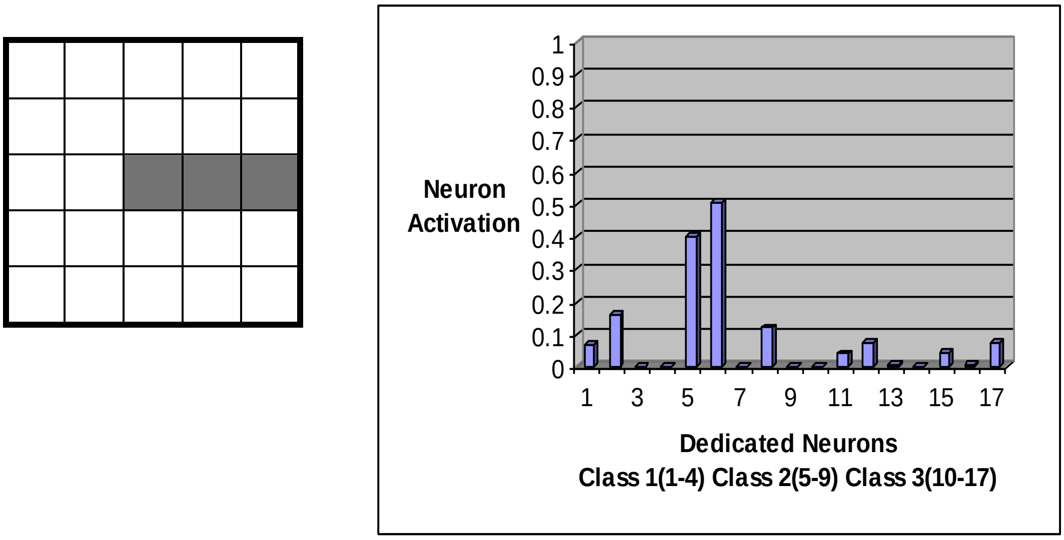
Graph 9.4: Test for Class-1-Feature-3
Illustration of Activation of Various Classes of Neurons

Graphs 9.2 and 9.3 show the activation of output neurons when tested for Class 1 features. The most highly activated neurons belong to class 1 and hence recognise the features.

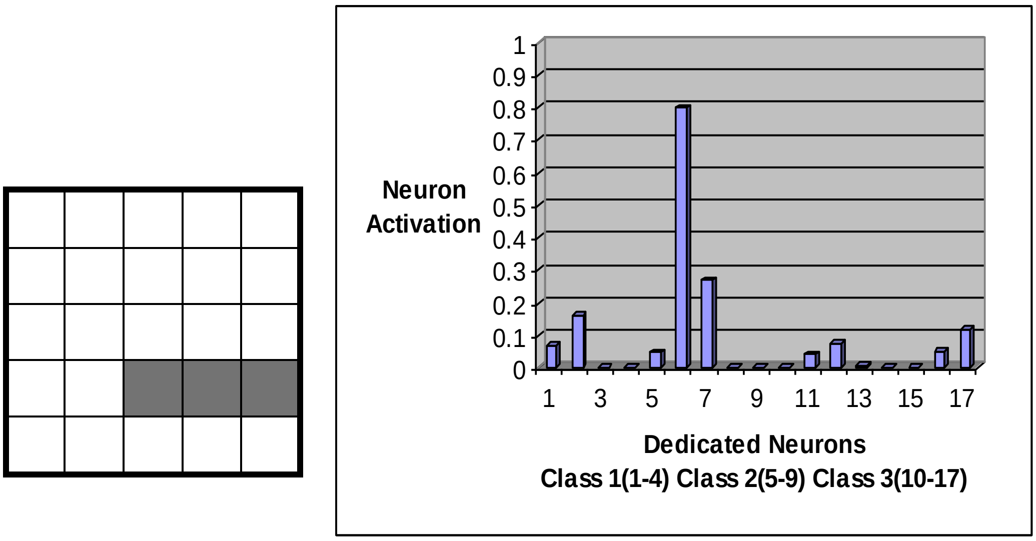
Graph 9.4 shows weak activation for a Class 1 feature, although it is the highest activation among the dedicated neurons. Class 2 neurons are also highly activated, but as the activation of Class 1 dedicated neurons is the highest the feature is classified as Class 1 type.

Test with Class 2 Features from the Training Set

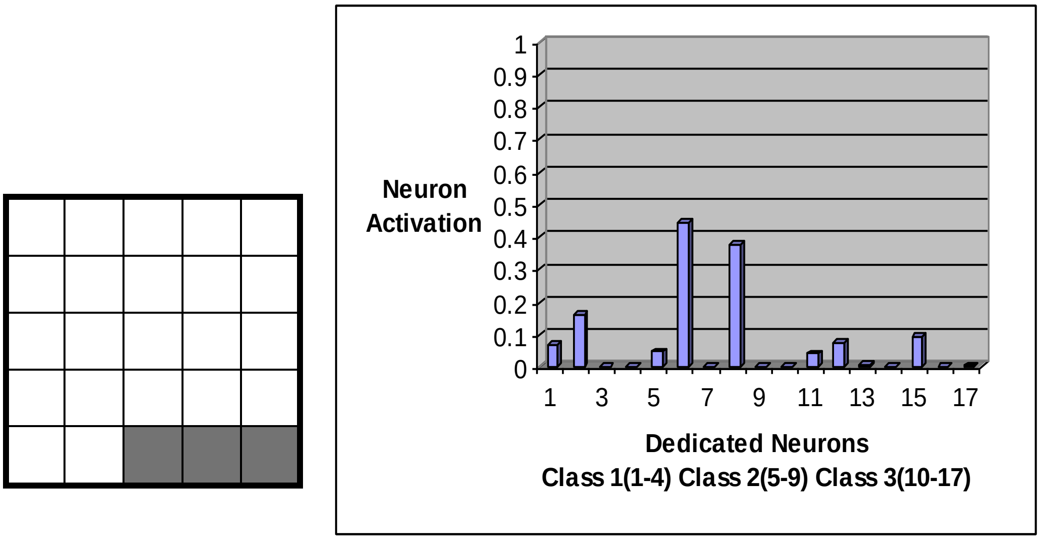
Graphs 9.5- 9.7 show the results obtained when the DNN was tested for Class 2 features from the training set.



Graph 9.5: Test for Class-2-Feature-1
Illustration of Activation of Various Classes of Neurons



Graph 9.6: Test for Class-2-Feature-2
Illustration of Activation of Various Classes of Neurons



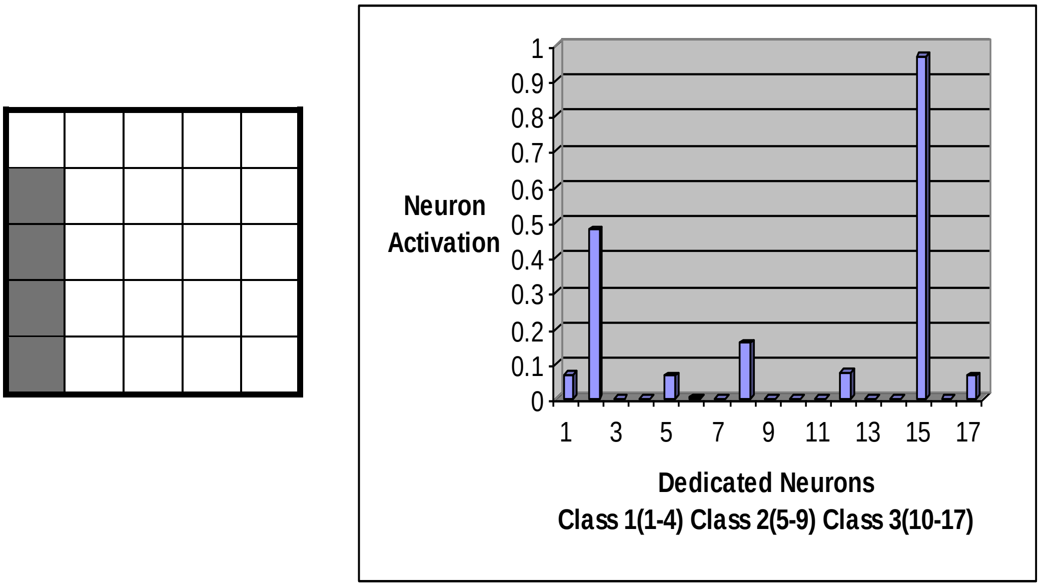
Graph 9.7: Test for Class-2-Feature-3
Illustration of Activation of Various Classes of Neurons

Graph 9.5 shows high activation of Class 2 neurons for a Class 2 feature and hence is successful. Graphs 9.6 and 9.7 show high activation among the Class 2 dedicated neurons when tested with two corresponding Class 2 features.

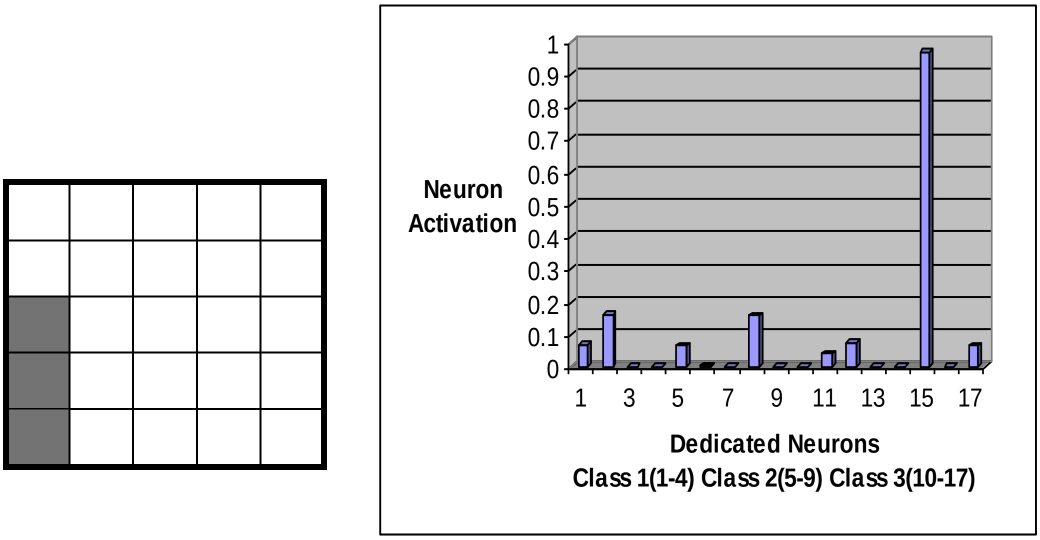
It can also be noted from the Graphs 9.5 and 9.7 that the output Class 2 neurons have very weak activation. It can be noted that this weak excitation among Class 2 features will have an affect on the results reported in all the results presented in that the Class 2 feature neuron activation will continue to be quite low compared to Class 1 and Class 2 features.

Test with Class 3 Features from Training Set

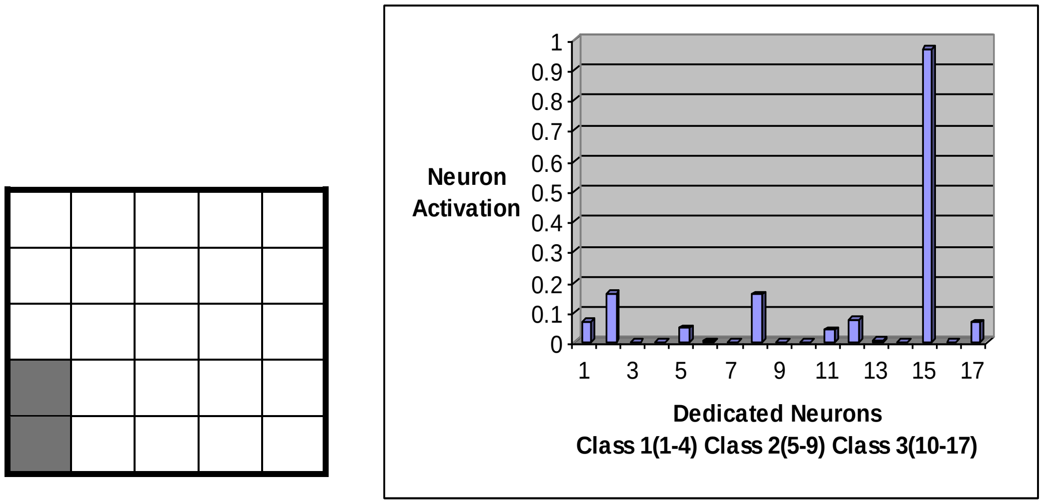
Graphs 9.8-9.10 present the results obtained when Class 3 features were used to test the DNN.



Graph 9.8: Test for Class-3-Feature-1
Illustration of Activation of Various Classes of Neurons



Graph 9.9: Test for Class-3-Feature-2
Illustration of Activation of Various Classes of Neurons



Graph 9.10: Test for Class-3-Feature-3
Illustration of Activation of Various Classes of Neurons

Graphs 9.8, 9.9 and 9.10 show high activation among the Class 3 neurons when tested with Class 3 features as shown above.

The results obtained when the DNN was tested with features from the training set show that the DNN successfully recognised all the patterns. It not only showed high activation among the dedicated neurons corresponding to the specific class of features being tested, but also snubbed the activation of the opposition neurons as evident in Graph 9.2 – Graph 9.10.

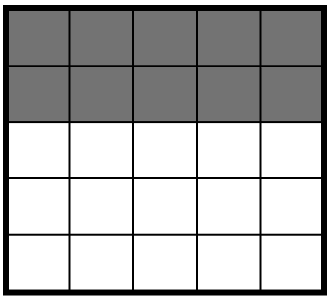
In Graph 9.4, the activation of Class-1 neurons, although over 0.4 is not significantly higher than a neighbouring Class-2 neuron. This is because of the type of feature being recognised, which is defined by only two active pixels. In Graph 9.8, Class-1 neurons show a relatively high activation value when a Class-3 feature is tested as the feature is encroaching onto the Class-1 zone.

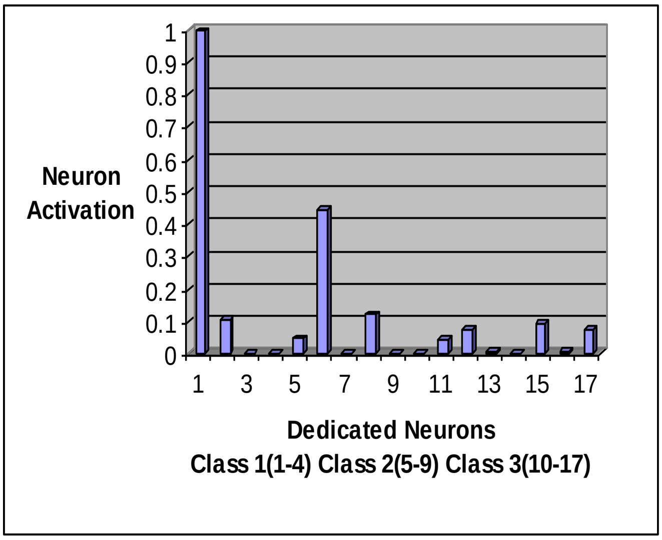
9.3.2 TESTS WITH NEW DATA

The DNN was tested with new features to analyse its ability to generalise. Three new features in each class were chosen. Graphs 9.11 – Graph 9.18 show the results obtained when the DNN was tested with new patterns that were not included in the training set.

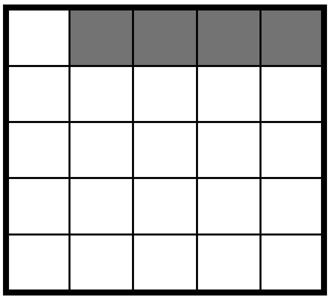
Test with New Class 1 Features

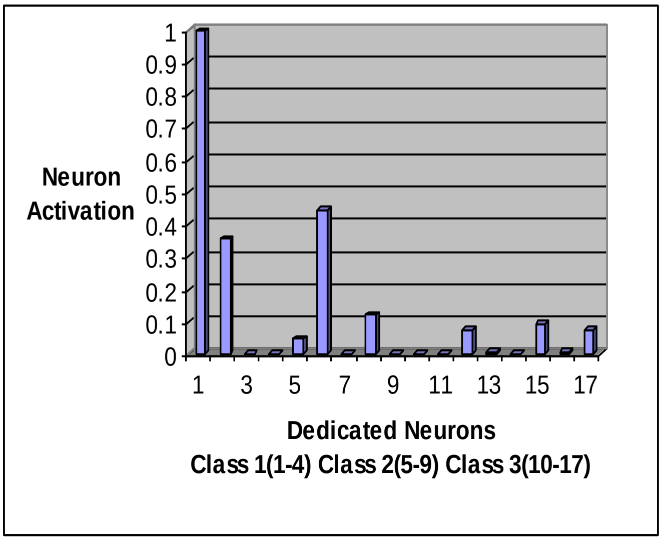
Graphs 9.11 – 9.13 show that the new Class 1 features were recognised successfully by the high activation of corresponding dedicated neurons.



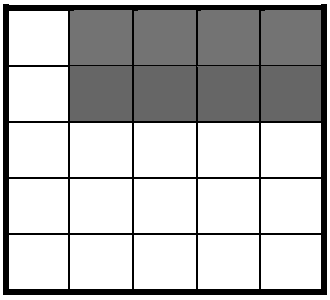


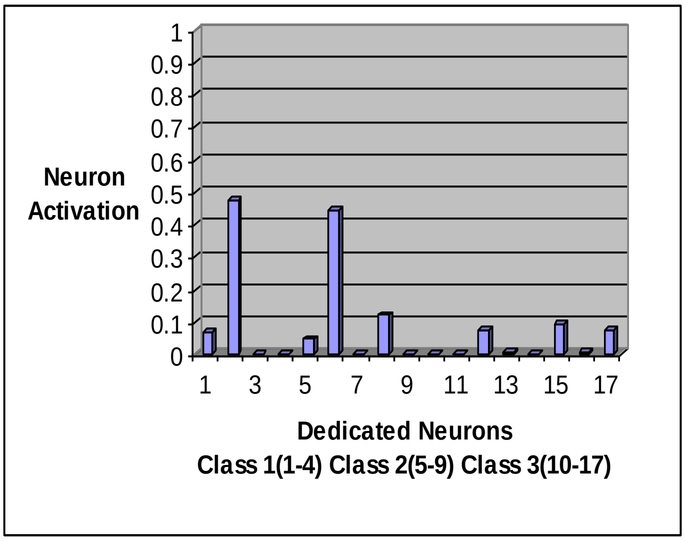
Graph 9.11: Test for Class-1-New Feature-1
Illustration of Activation of Various Classes of Neurons





Graph 9.12: Test for Class-1-New Feature-2
Illustration of Activation of Various Classes of Neurons

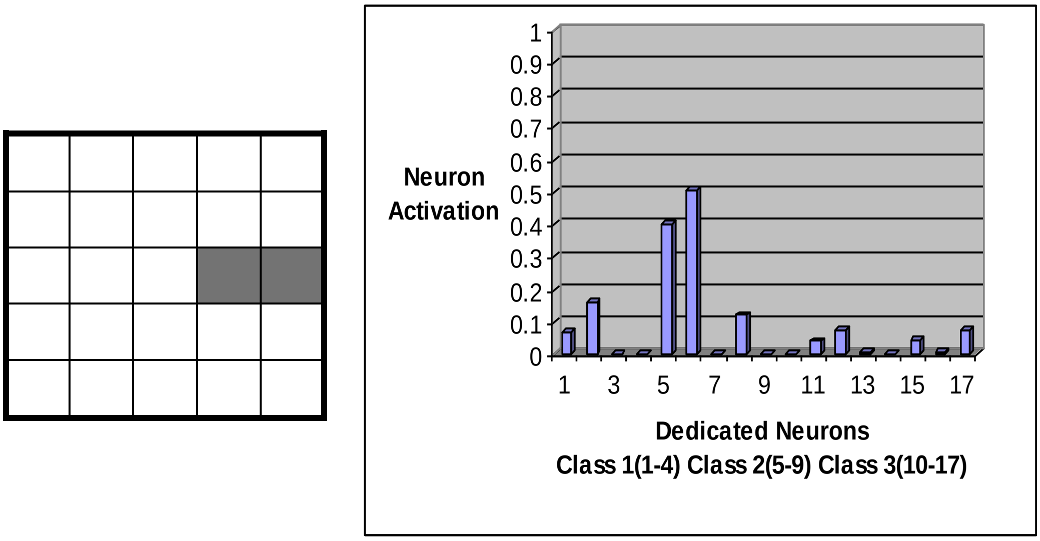




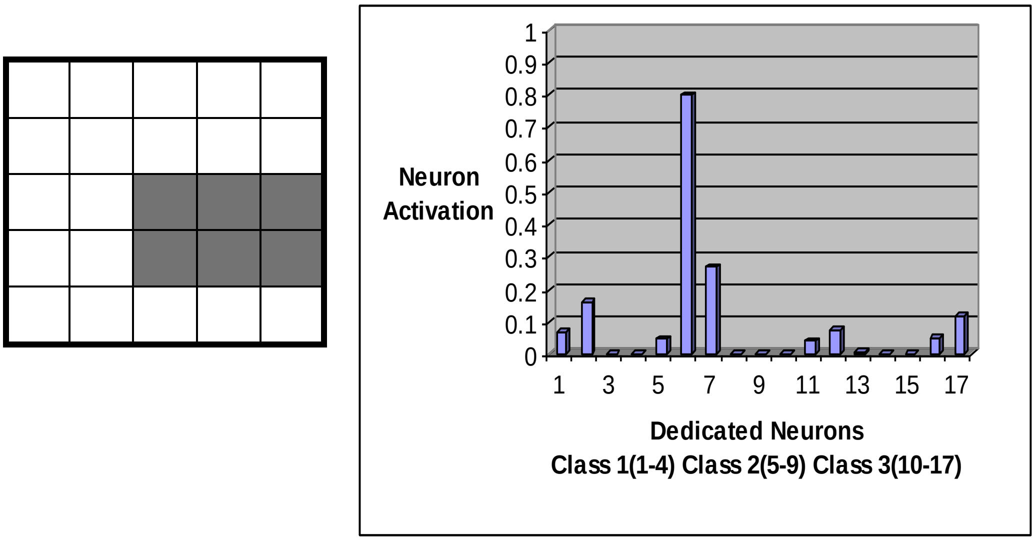
Graph 9.13: Test for Class-1-New Feature-3
Illustration of Activation of Various Classes of Neurons

Test with New Class 2 Features

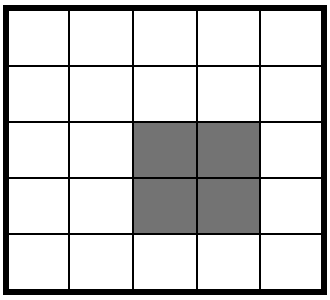
Graphs 9.15 – 9.17 represent the results obtained when tested for new Class 2 features.

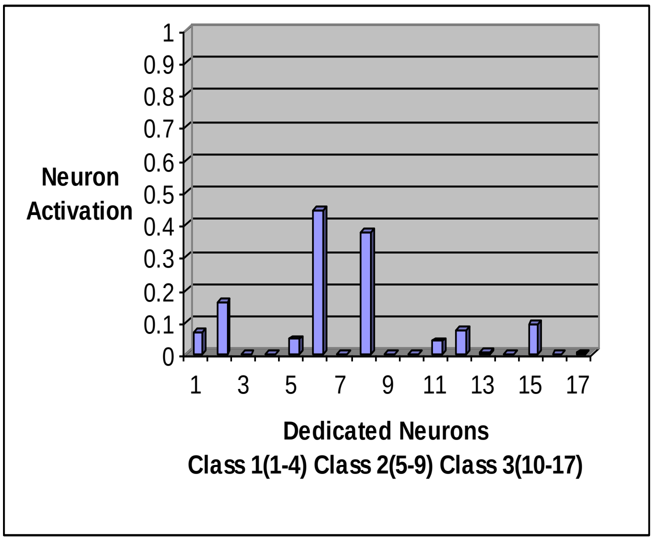


Graph 9.14: Test for Class-2-New Feature-1
Illustration of Activation of Various Classes of Neurons



Graph 9.15: Test for Class-2-New Feature-2
Illustration of Activation of Various Classes of Neurons



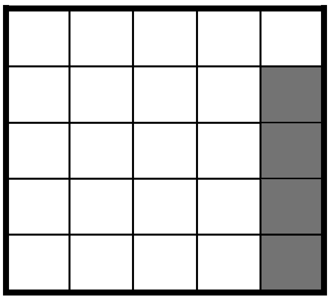


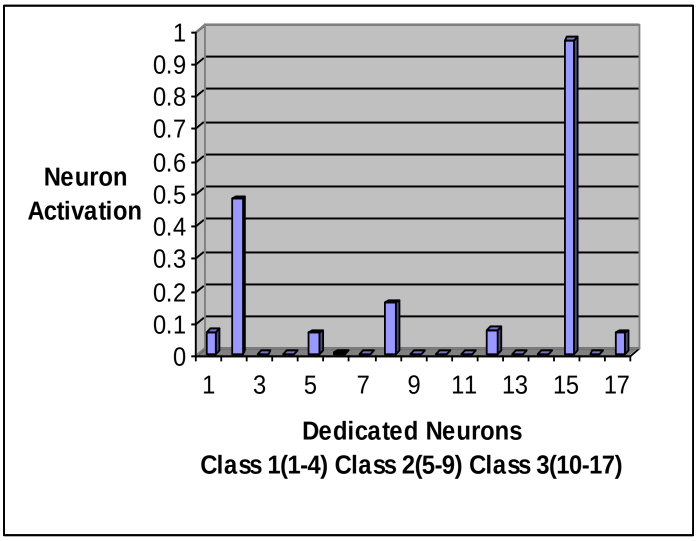
Graph 9.16: Test for Class-2-New Feature-3
Illustration of Activation of Various Classes of Neurons

High activation among Class 2 features is observed, when the corresponding new Class 2 features are used to test the DNN, which is evident from the Graphs 9.14 - 9.16. However, class 2 features do not show activation values >0.6. This is a trend that continues in other results as mentioned earlier.

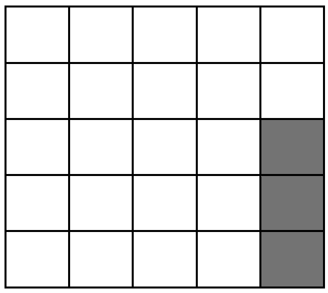
Test with New Class 3 Features

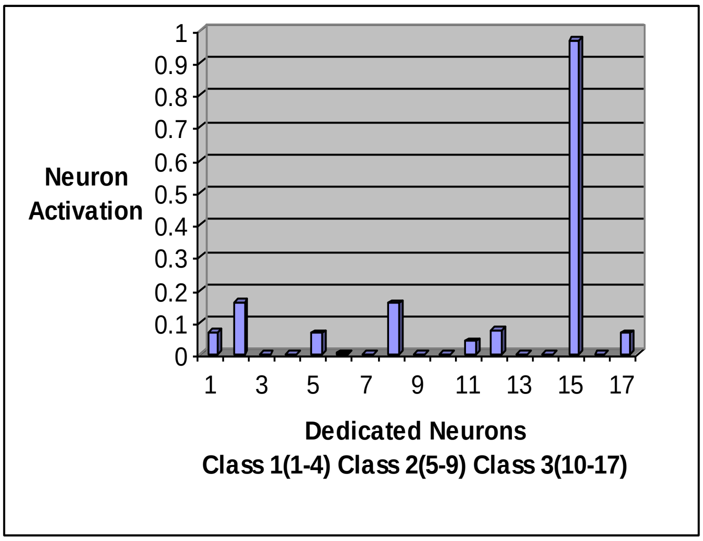
Graphs 9.17-9.19 show the results obtained when the new Class 3 features were used to test the DNN. The new features also vary by their spatial position in the grid. However, this did not affect the DNN's recognition capabilities.



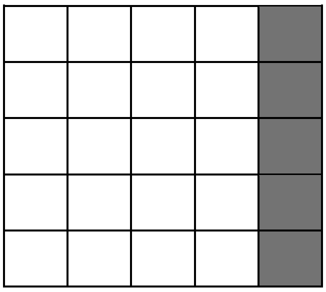


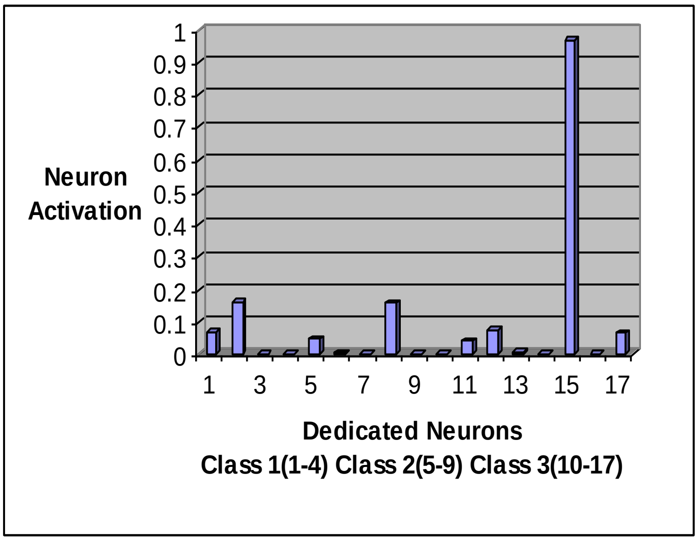
Graph 9.17: Test for Class-3-New Feature-1
Illustration of Activation of Various Classes of Neurons





Graph 9.18: Test for Class-3-New Feature-2
Illustration of Activation of Various Classes of Neurons





Graph 9.18: Test for Class-3-New Feature-3
Illustration of Activation of Various Classes of Neurons

All features were recognised by the DNN, which is apparent from the high activation value of the corresponding dedicated neurons. Graph 9.13 is the only one that does not show distinct recognition of Class-1 features as the Class-2 neurons are also fired up. However, the Class-1 neuron was relatively higher than the Class-2 neuron showing that it classifies the feature as Class-1.

The results prove that the DNN is good at generalising. The conclusions can be drawn as the new features belonging to each class vary distinctly from the features in the training set. The common characteristics of the two sets of data are the type of features and its spatial location in a 5x5grid.

9.3.3 TESTS WITH NOISY DATA (STAGE-1)

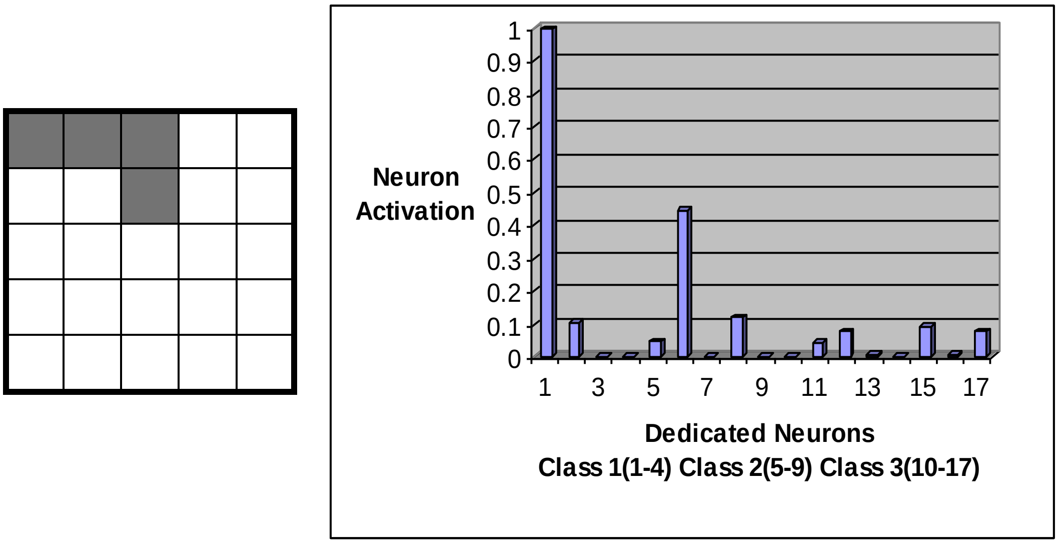
The DNN was tested for its tolerance to noise. Noise was introduced in the form of speckles- random pixels being changed from 0 to 1. Speckles were intentionally introduced in areas of the rival classes to understand the sensitivity of the DNN.

This was done in two stages. The first stage involved using features from the training set and changing a pixel from a 0 to 1. The second stage involved introducing two speckles.

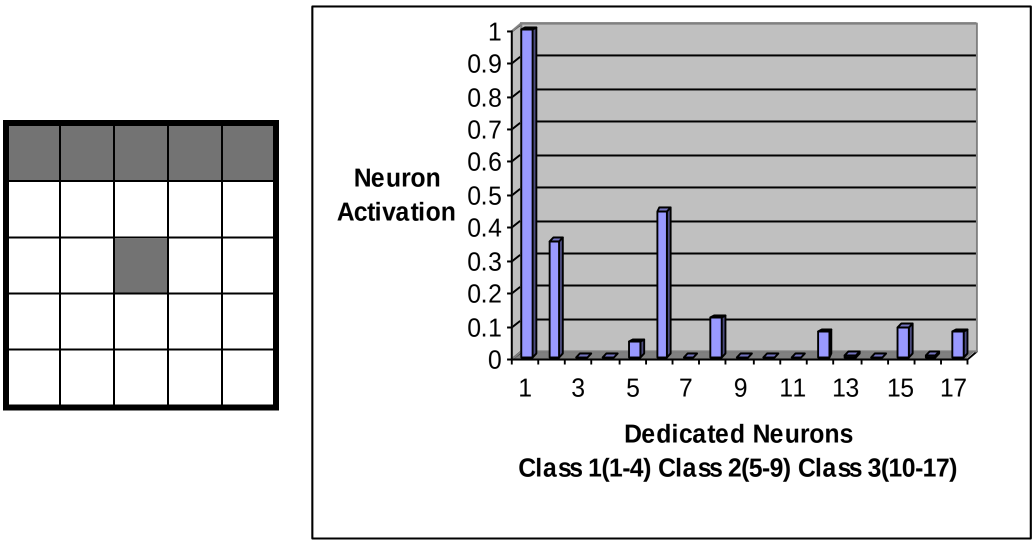
The reasons for the introduction of speckles are:

- To bring in the concept of corrupted images and test DNN's ability to recognise the main features in the image.
- The speckles could also represent insignificant, but prevalent features. Hence, in case of robotic obstacle avoidance, it would be beneficial for the DNN to hint regarding the presence of an obstacle, although very small, in the relevant region through the activation of relevant dedicated neurons. However, such a scenario may occur only when there is a densely speckled region in the image. For, example 2 speckles occurring in neighbouring pixels.

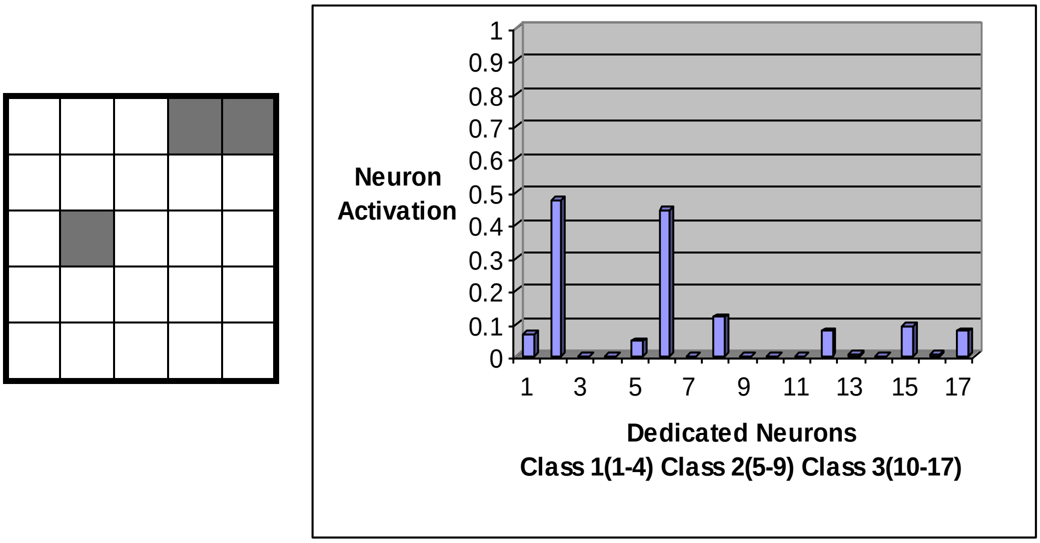
Graphs 9.19 – Graph 9.27 show the results obtained when the DNN was tested with images that included stage-1 noise-one speckled pixel.



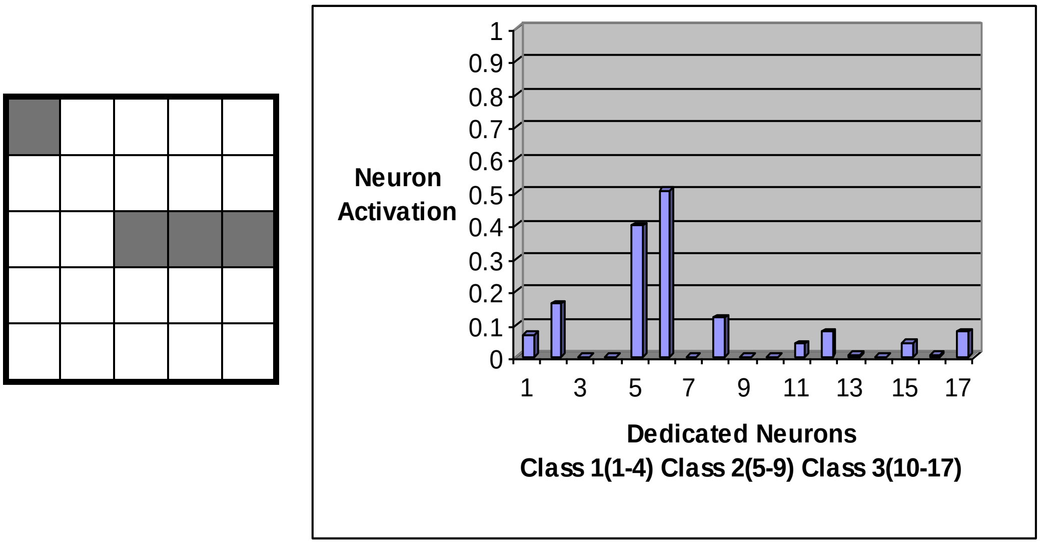
Graph 9.19: Test for Noisy-1 Class-1-Feature-1
Illustration of Activation of Various Classes of Neurons



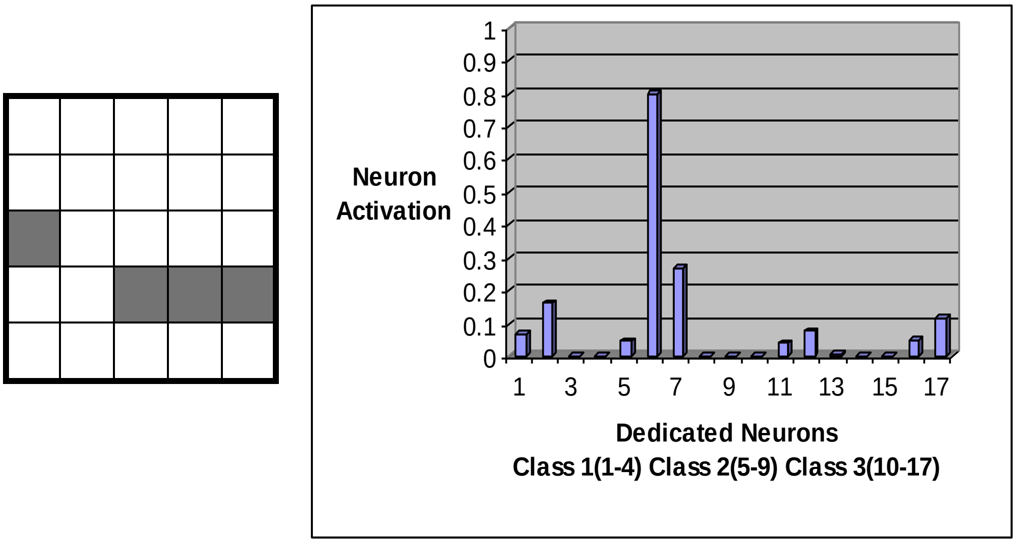
Graph 9.20: Test for Noisy-1 Class-1-Feature-2
Illustration of Activation of Various Classes of Neurons



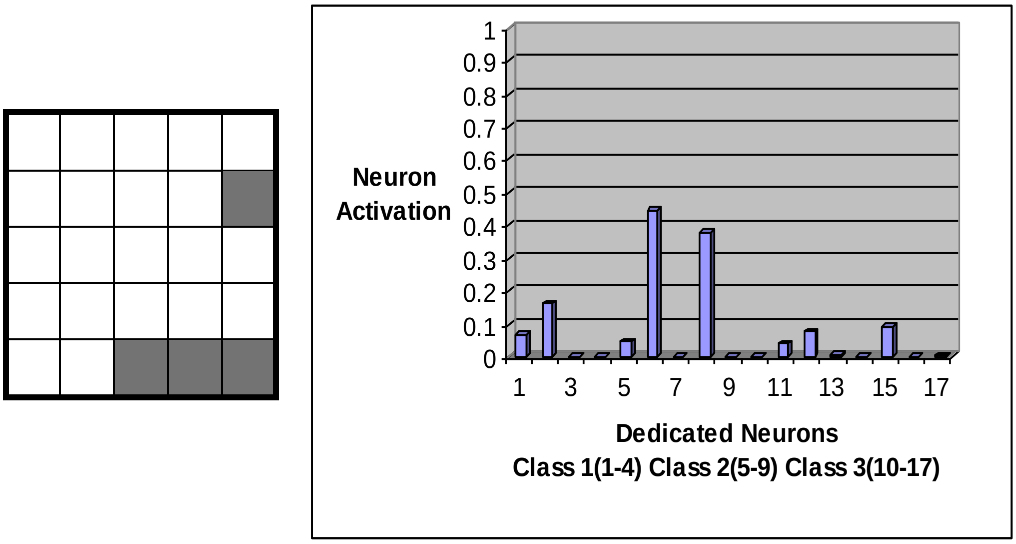
Graph 9.21: Test for Noisy-1 Class-1-Feature-3
Illustration of Activation of Various Classes of Neurons



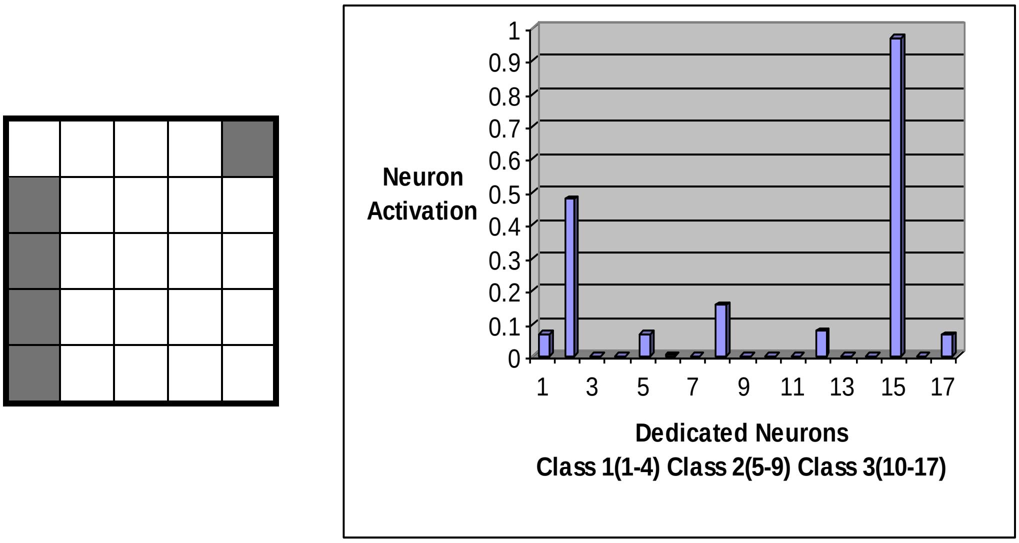
Graph 9.22: Test for Noisy-1Class-2-Feature-1
Illustration of Activation of Various Classes of Neurons



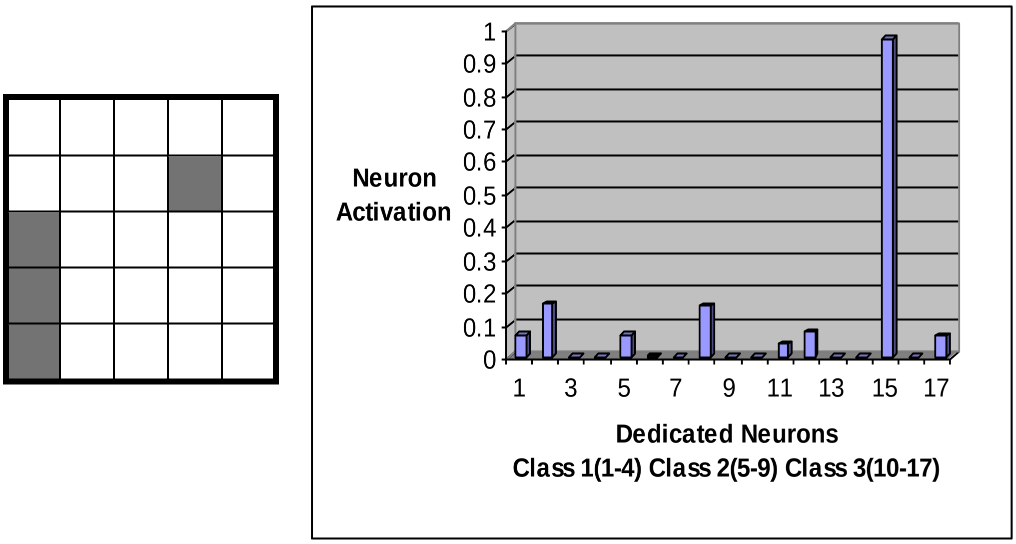
Graph 9.23: Test for Noisy-1 Class-2-Feature-2
Illustration of Activation of Various Classes of Neurons



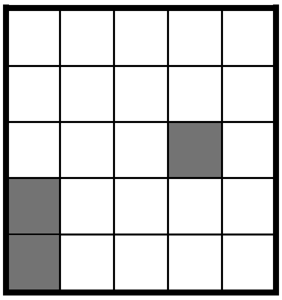
Graph 9.24: Test for Noisy-1Class-2-Feature-3
Illustration of Activation of Various Classes of Neurons

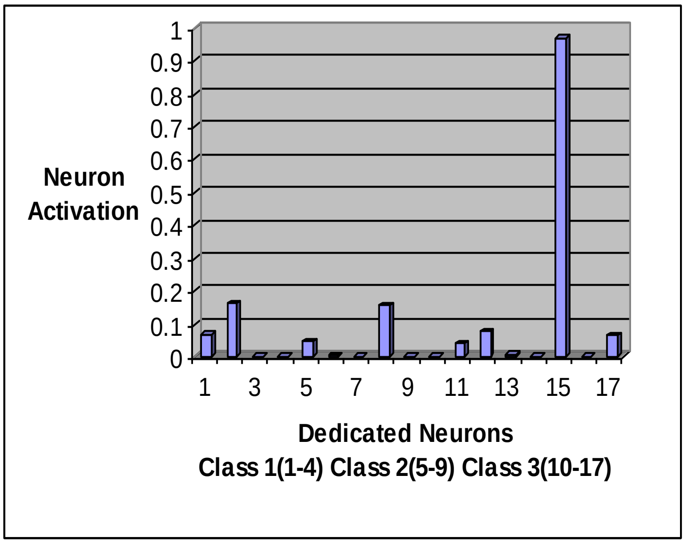


Graph 9.25: Test for Noisy-1 Class-3-Feature-1
Illustration of Activation of Various Classes of Neurons



Graph 9.26: Test for Noisy-1 Class-3-Feature-2
Illustration of Activation of Various Classes of Neurons





Graph 9.27: Test for Noisy-1 Class-3-Feature-3
Illustration of Activation of Various Classes of Neurons

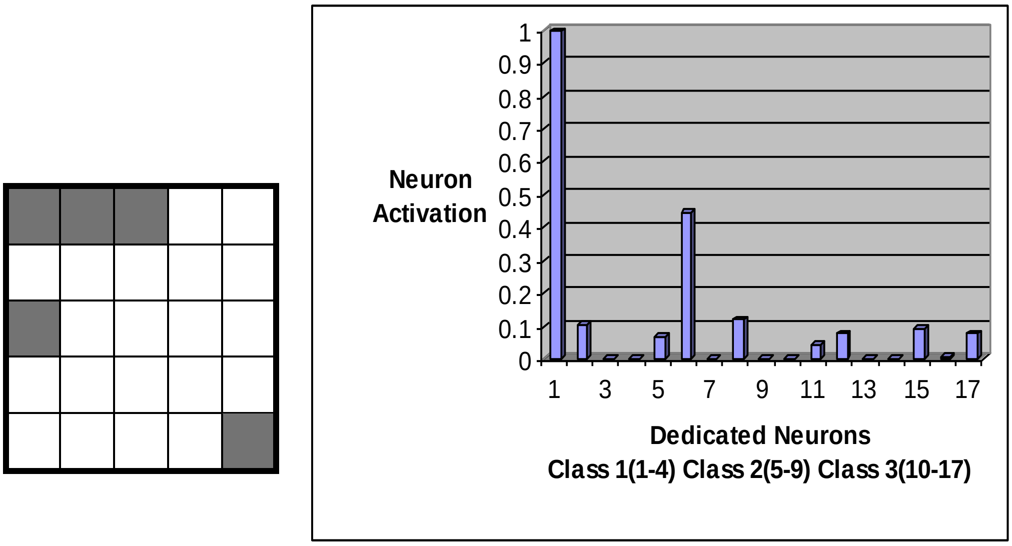
All features have been successfully classified as seen from the high activation of dedicated neurons belonging to a certain class in the Graphs 9.19 - 9.27.

Graph 9.21 shows weak recognition, but nevertheless has recognised the feature as belonging to Class-1. This again is due to the nature of the feature itself that has little scope for definition. Such features were included to test DNN's ability to recognise weakly defined features.

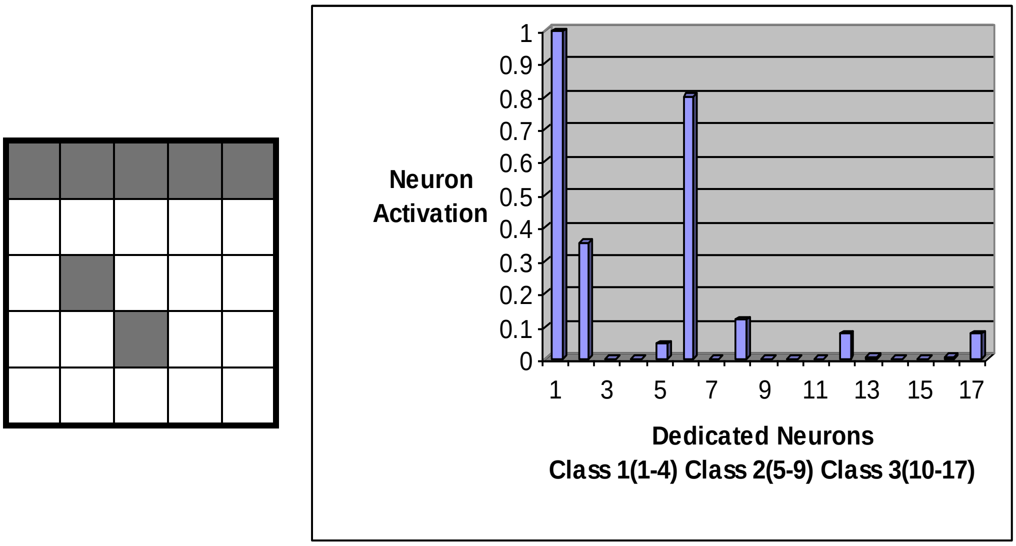
It can be concluded that the DNN is tolerant to noise, but, before this conclusion can be drawn, it was thought that to substantiate these results, the DNN must be tested with increased noise- two speckled pixels. The results are listed in Section 9.3.4.

9.3.4 TESTS WITH INCREASED NOISE (STAGE-II)

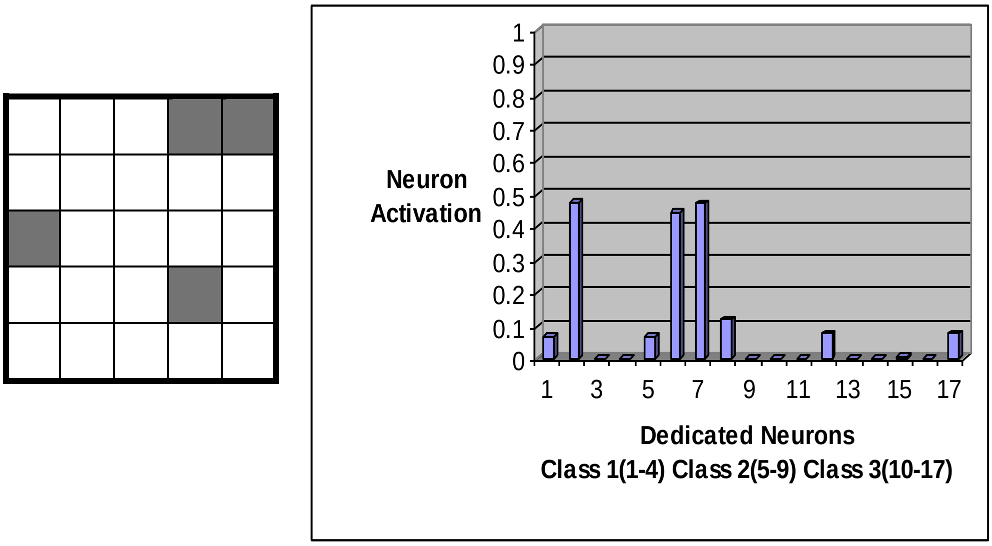
Graph 9.28 – Graph 9.36 show the results obtained when the DNN was tested with images that contained increased levels of noise. Increased noise was defined by random changes in two pixels of the image from a value of 0 to 1.



Graph 9.28: Test for Noisy-2 Class-1-Feature-1
Illustration of Activation of Various Classes of Neurons

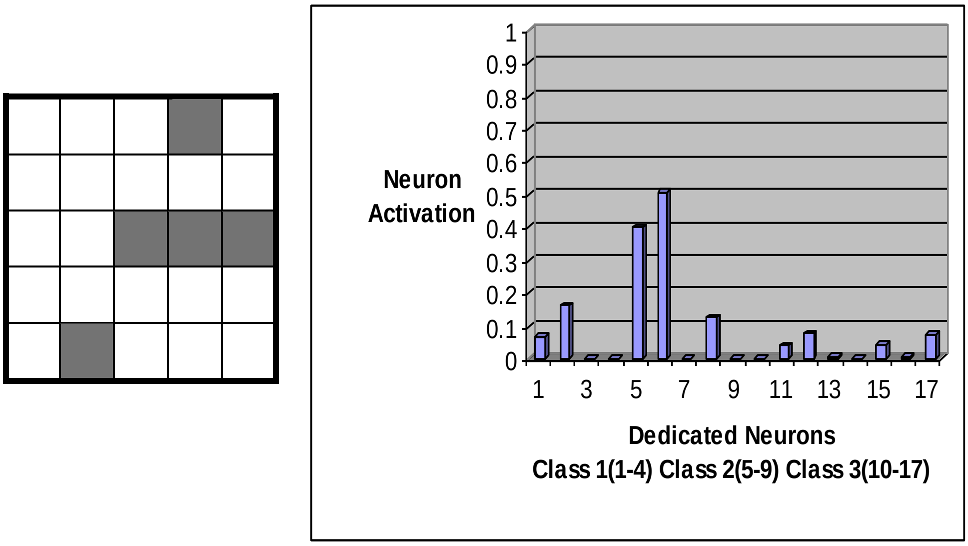


Graph 9.29: Test for Noisy-2 Class-1-Feature-2
Illustration of Activation of Various Classes of Neurons

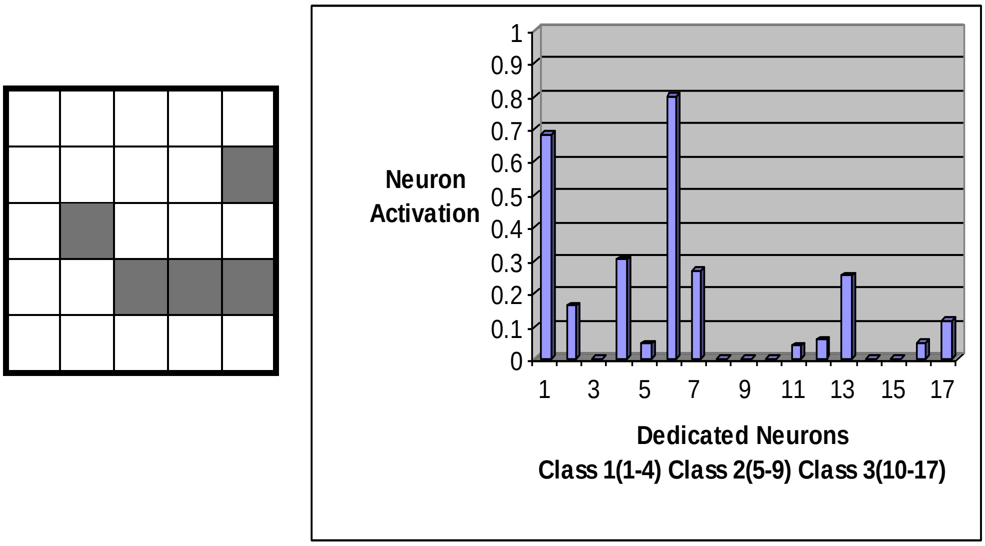


Graph 9.30: Test for Noisy-2 Class-1-Feature-3
Illustration of Activation of Various Classes of Neurons

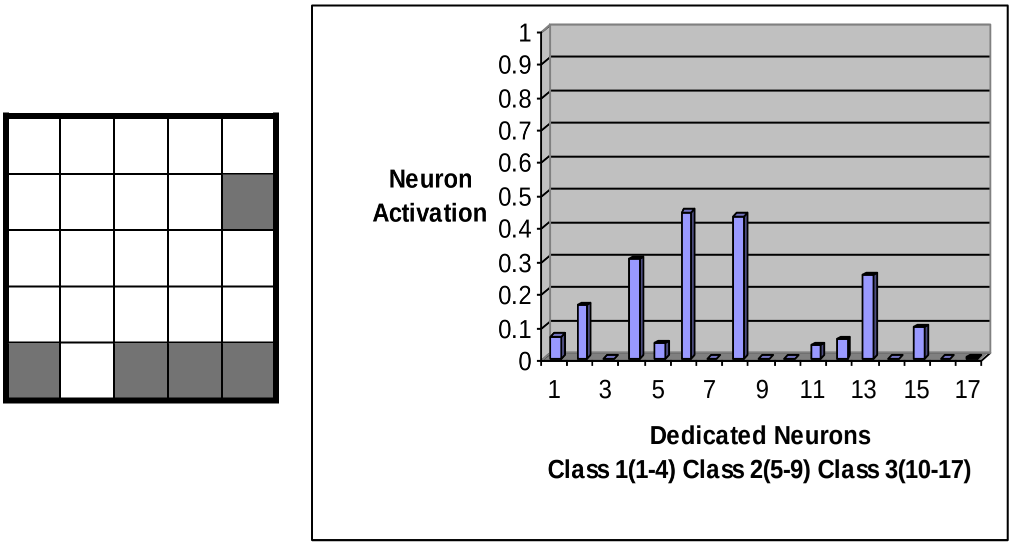
The Graph 9.29 and 9.30 show high activation among Class 2 neurons, although the test features belong to Class 1. This is due to the presence of speckles in the Class 2 region. However, the highest activation is among the Class 1 neurons.



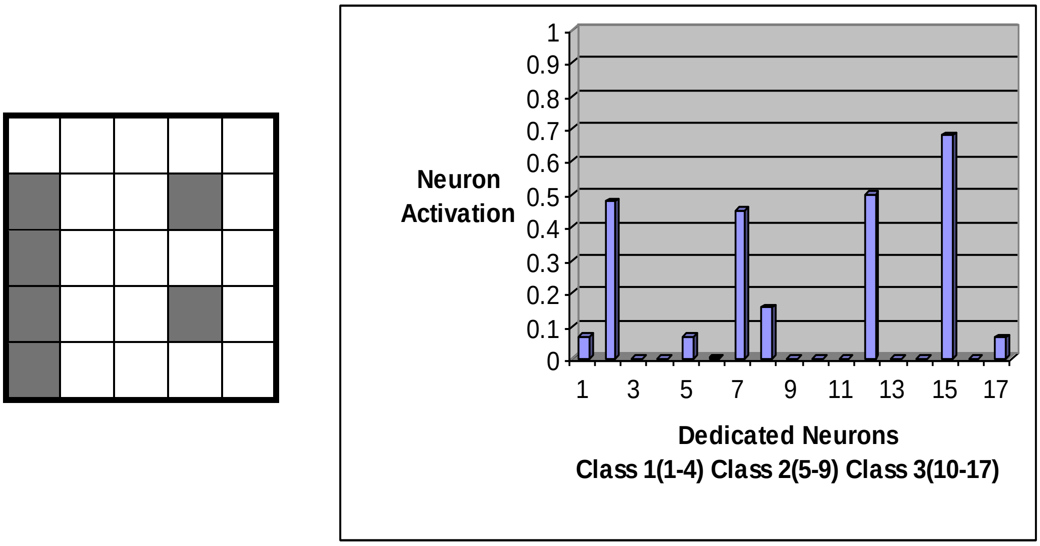
Graph 9.31: Test for Noisy-2 Class-2-Feature-1
Illustration of Activation of Various Classes of Neurons



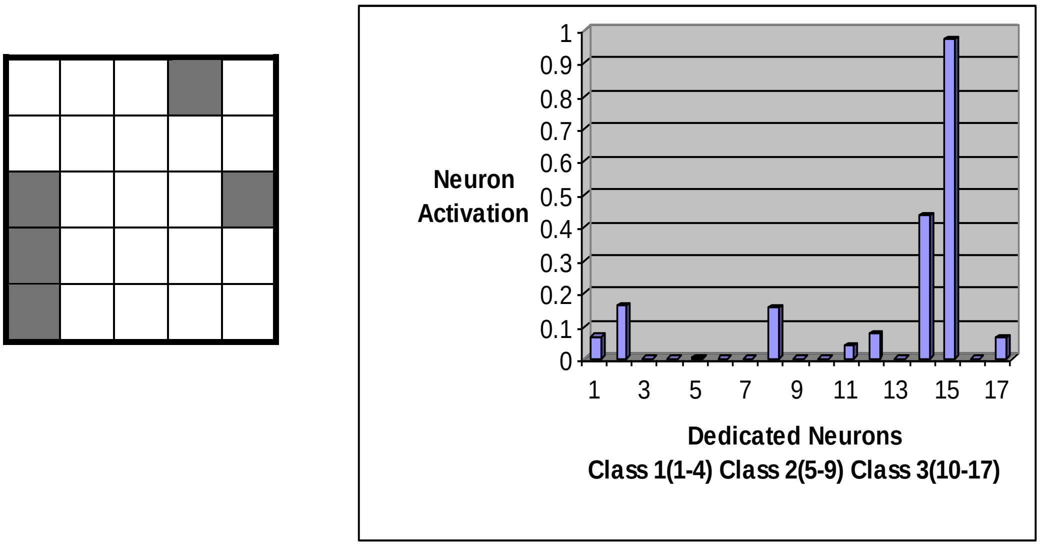
Graph 9.32: Test for Noisy-2 Class-2-Feature-2
Illustration of Activation of Various Classes of Neurons



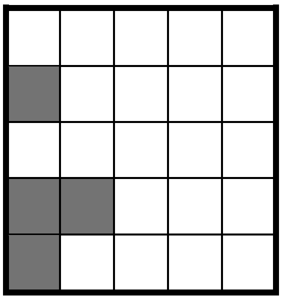
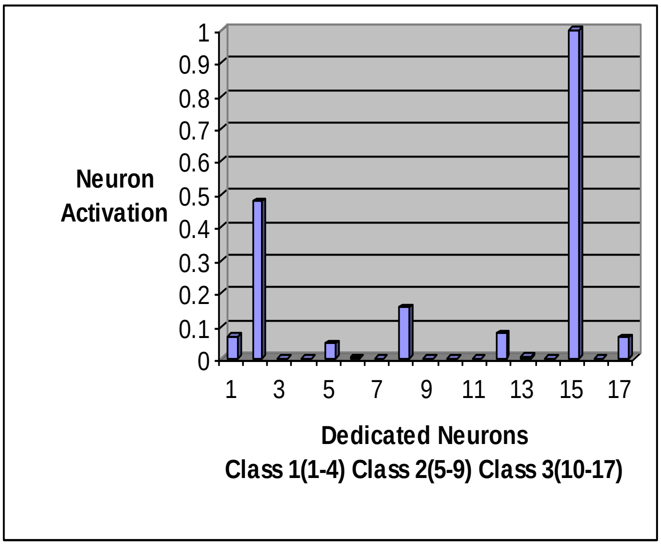
Graph 9.33: Test for Noisy-2 Class-2-Feature-3
Illustration of Activation of Various Classes of Neurons



Graph 9.34: Test for Noisy-2 Class-3-Feature-1
Illustration of Activation of Various Classes of Neurons



Graph 9.35: Test for Noisy-2 Class-3-Feature-2
Illustration of Activation of Various Classes of Neurons

Graph 9.36: Test for Noisy-2 Class-3-Feature-3
Illustration of Activation of Various Classes of Neurons

It has to be noted that in both stages of noise inclusion, the noise added to all features were in the regions of the corresponding rival classes of features. This was intentionally done to check the resilience of the DNN to noise.

When the new noisy images were used to test the DNN, although the activation of dedicated neurons allocated for a certain class of features is high compared to other neurons, the distinction between various classes of features is at times coarse due to a relatively high activation among the rival dedicated neurons.

However, high activation is consistently observed among dedicated neurons that belonged to a certain class when tested for that particular class. It can be concluded that all features were successfully classified based on the activation of dedicated neurons belonging to various classes.

9.4 SUMMARY

This chapter presented the details of the training process of the DNN. It also gives details of features that were used as test data. Some of the test data comprised of new features and features which consist various levels of noise. The features in the test data were not included in the training set.

The DNN was tested with features from the test set to evaluate the characteristics of noise tolerance, generalisation and robustness. The results obtained show that the DNN is very successful at recognising features from the training and test set. Hence, it can be concluded that the DNN is robust in terms of its ability to recognise and classify features in an image.

The next phase of testing involved increasing the complexity of classification by presenting the DNN with combinations of features from the training set. Chapter 10 presents the results obtained when the DNN was tested with combinations of features.

CHAPTER 10: TEST WITH COMBINATIONS OF FEATURES

10.1 INTRODUCTION

The DNN was trained and tested using features from the training set, new features that bear resemblance in terms of type of feature and its position in input space and two sets of varying degrees of noisy data. The results were given in Chapter 9.

The next step was to test the DNN for more advanced image processing capabilities. One of the problems that is apparent in traditional ANNs is its lack of comprehension of any new data that is not in line with the training set. This breakdown of proficiency extends to understanding combinations of known features. Results obtained in the research programme as discussed in Chapter 5 support this conclusion.

There are several ways in which this problem is tackled, one of them being introduction of Modularity. MNNs using traditional ANNs and training methods have not been very successful, as demonstrated by the results presented in Chapter 5. Despite the discouraging application of modularity in ANNs, effort was made during the design of the DNN to incorporate modularity. This chapter presents results from tests conducted using combinations of known features. Algorithms for testing combinations of two and three features are given in the following sections. A universal algorithm for an unknown number of features is not included in this thesis.

10.2 COMBINATION OF TWO PATTERNS

The next stage in verifying advanced recognition properties of the DNN was to test the network with combinations of two features. The results obtained are presented in this section.

There are 27 possible combinations of two features, each belonging to a different class. The DNN was tested with images that consisted of a combination of features. The expected result is the high activation of the corresponding dedicated neurons of the class that the features in the image belonged to. For example consider the image in Figure 10.1.

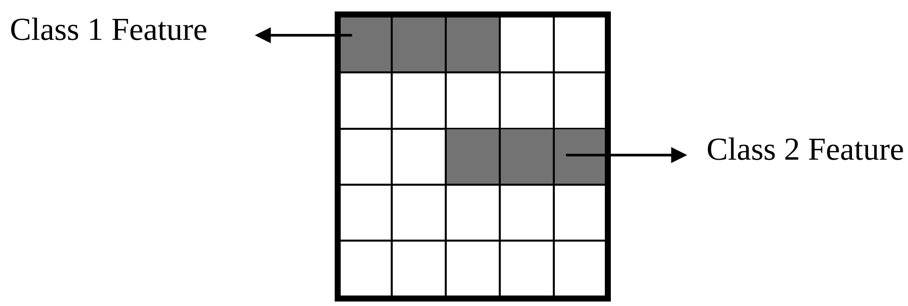


Figure 10.1: Illustration of Combinations of Two Features

As the features in the image are of type Class 1 and Class 2, when the DNN is tested with the image in Figure 10.1, then the dedicated neurons for Class 1 and Class 2 are expected to have a high activation.

The recognition criteria are based on the following algorithm.

Find the highest activation of each class of dedicated neurons

Select the two classes of neurons with the highest activation

Is the activation of each class > 0.4? **No**

Yes

Classify the features as belonging to the top two classes

If top two classes are the required classes **No**

(both features recognised- 100% success)

Is there a neuron with activation > 0.4? **No**

Yes

Recognise the class with an activation >0.4

If recognised class is required class **No**

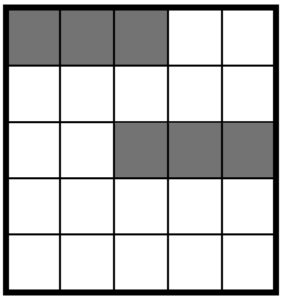
(one feature recognised – 50% success)

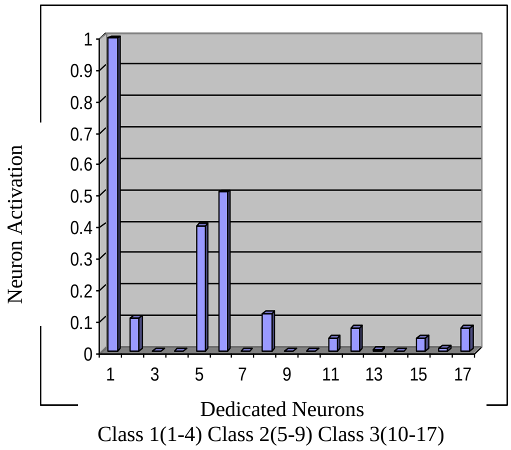
None of the features are recognised

(0% success)

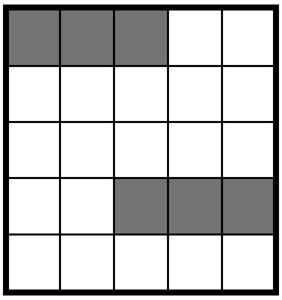
Combinations of Class 1 and Class 2 Features

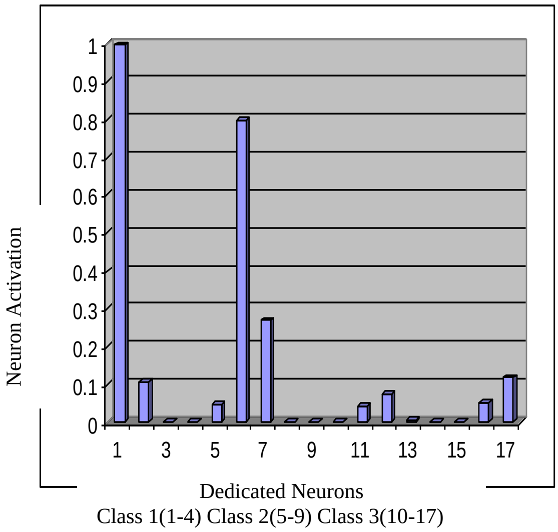
A series of images consisting of two features belonging to Class 1 and Class 2 were used to test the DNN for its capabilities in recognising the features in the image. The results obtained showing the activation of the dedicated neurons of each class is illustrated in Graphs 10.1-10.9.



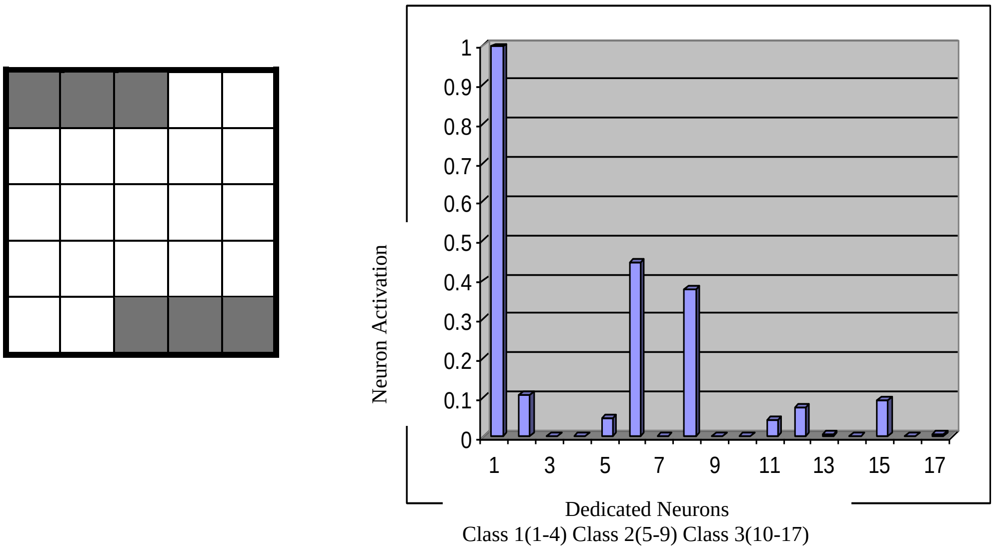


Graph 10.1: Results from tests on DNN using Combination of Class-1-Feature-1 and Class-2-Feature-1

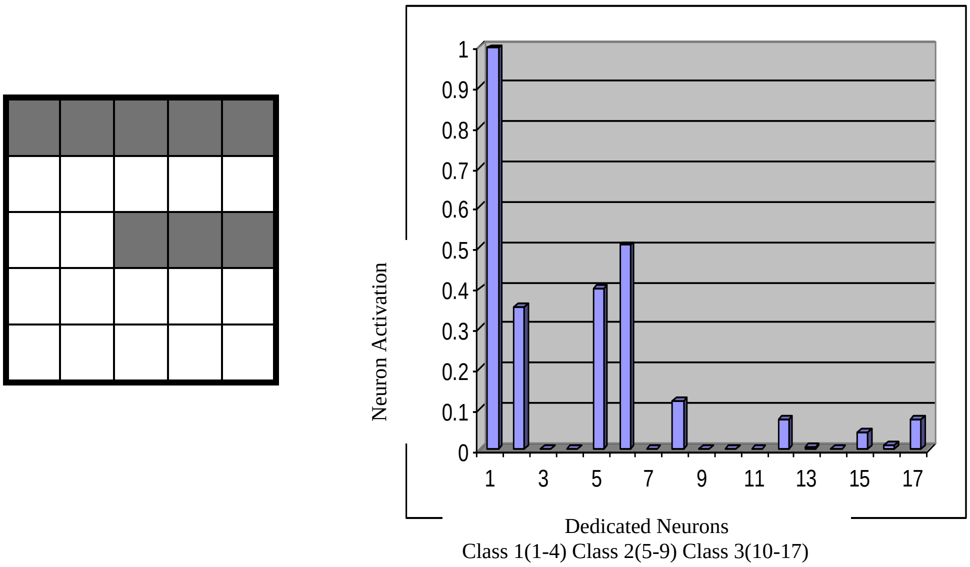




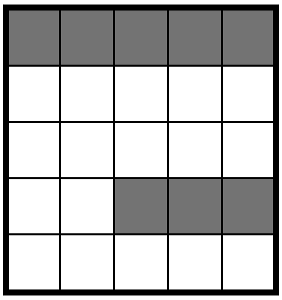
Graph 10.2: Results from tests on DNN using Combination of Class-1-Feature-1 and Class-2-Feature-2

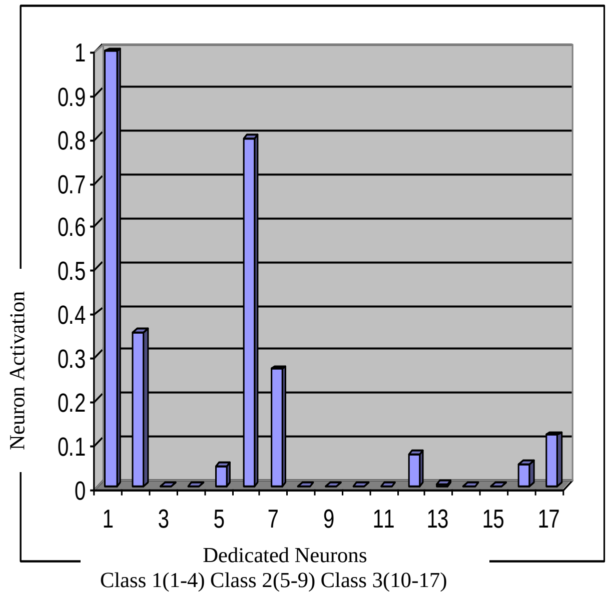


Graph 10.3: Results from tests on DNN using Combination of Class-1-Feature-1 and Class-2-Feature-3

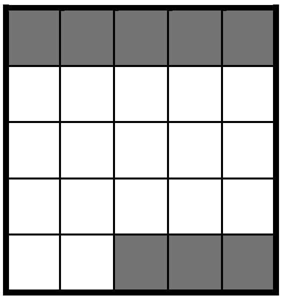


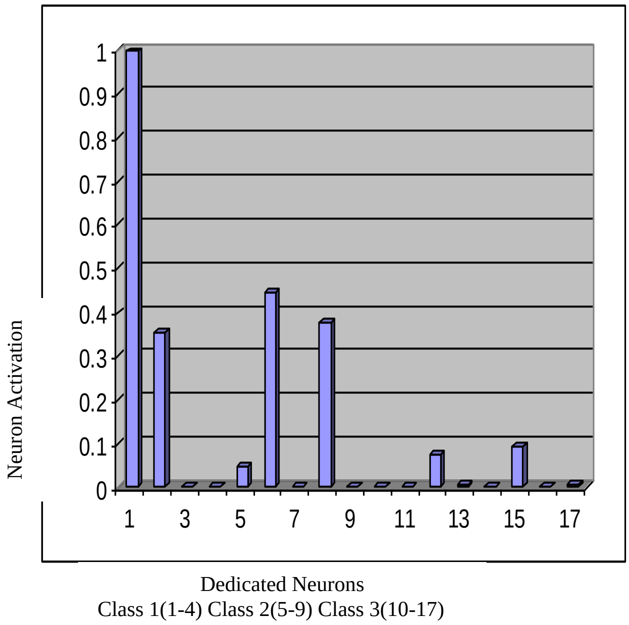
Graph 10.4: Results from tests on DNN using Combination of Class-1-Feature-2 and Class-2-Feature-1



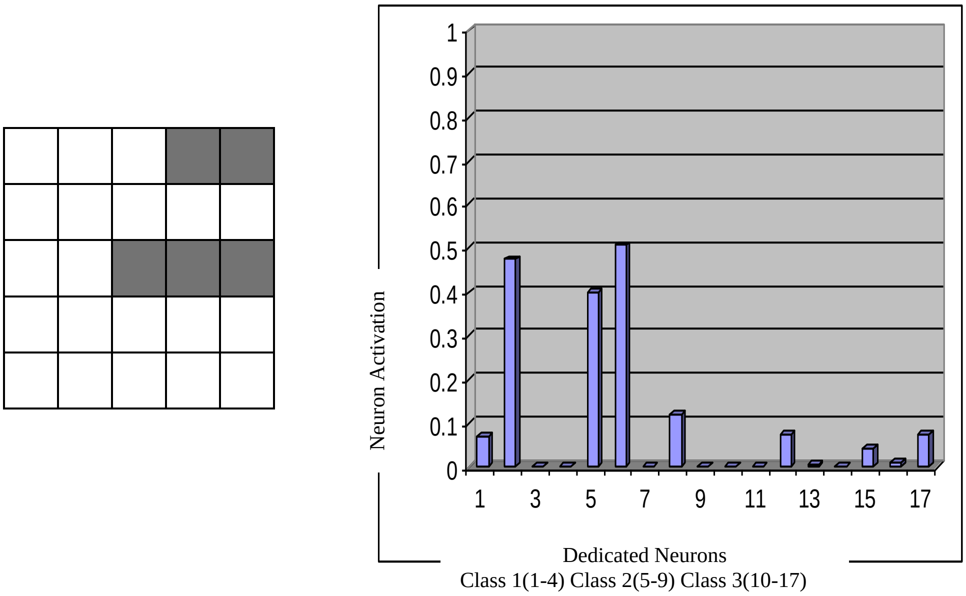


Graph 10.5: Results from tests on DNN using Combination of Class-1-Feature-2 and Class-2-Feature-2

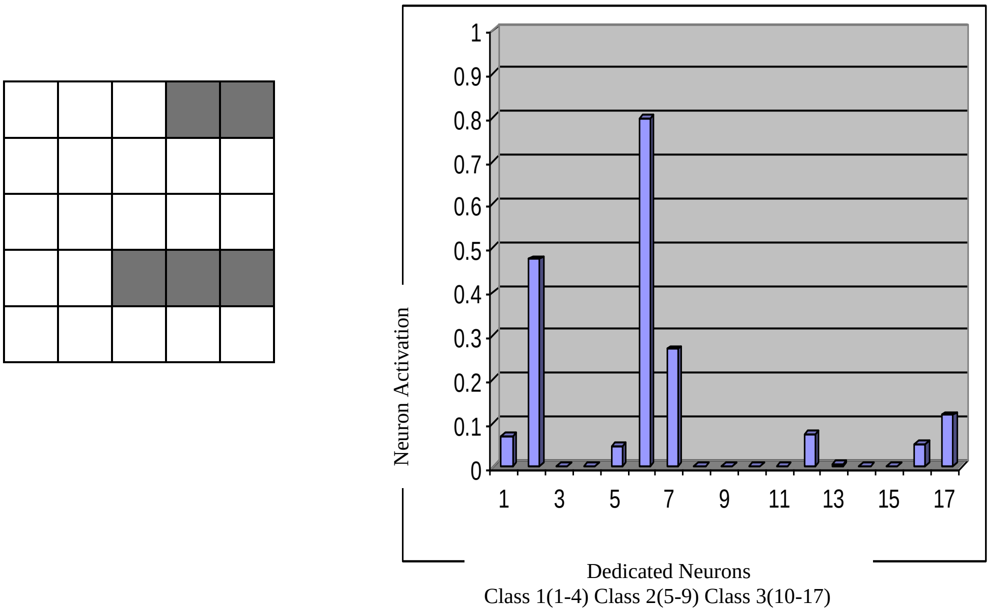




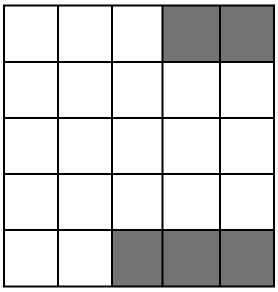
Graph 10.6: Results from tests on DNN using Combination of Class-1-Feature-2 and Class-2-Feature-3

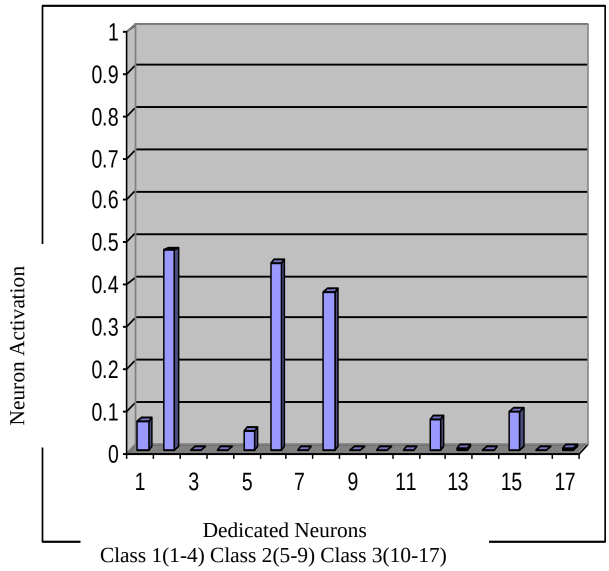


Graph 10.7: Results from tests on DNN using Combination of Class-1-Feature-3 and Class-2-Feature-1



Graph 10.8: Results from tests on DNN using Combination of Class-1-Feature-3 and Class-2-Feature-2





Graph 10.9: Results from tests on DNN using Combination of Class-1-Feature-3 and Class-2-Feature-3

According to the recognition criteria for two features mentioned earlier and as evident from the graphs, the dedicated neurons of Class 1 and Class 2 show an activation, which is >0.4. Also the activation of the Class 3 neurons is < 0.4.

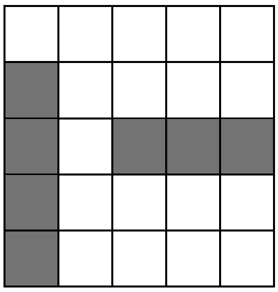
Hence the DNN has successfully recognised the two types of features in the image when presented with a combination of known features. It has to be noted that the DNN was trained only with the individual features and had never encountered a combination of these features during the training phase.

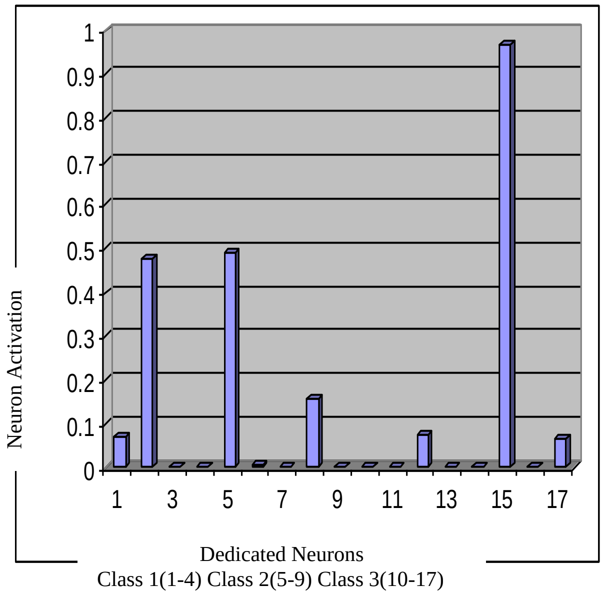
The results also show that the success rate of recognition was 100% in all 9 cases because both the features were recognised with an activation > 0.4.

Combinations of Class 2 and Class 3 Features

The next set of tests involved using images made of features of Class 2 and Class 3. The results obtained are presented in the graphs 10.10 – 10.16.

The recognition criteria are as mentioned earlier and based on the activation, the type and number of classes recognised. The success rate of the DNN in recognising various images are either 100%, where both the features are recognised or 50%, where only one feature is recognised and 0%, where none of the features are recognised.

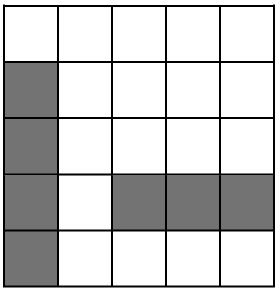


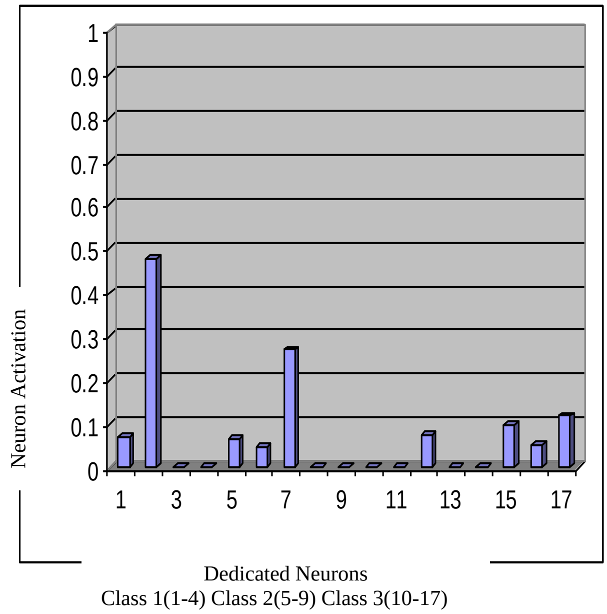


Graph 10.10: Results from tests on DNN using Combination of Class-2-Feature-1 and Class-3-Feature-1

Graph 10.10 shows that the DNN was 100% successful. This can be concluded by the activations of the Class 2 and Class 3 neurons, which are very high. Although, Class 1 neurons show activation > 0.4, it is not the 1st or 2nd highest activation values and hence ignored.

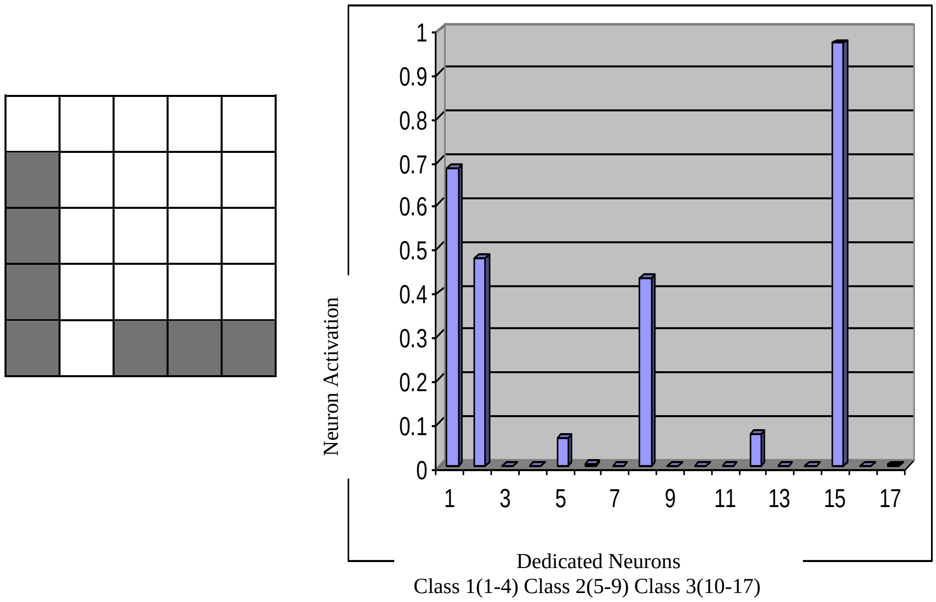
In the Graph 10.11, the highest neuron activation belongs to Class 1, which is a feature that is not present in the image. The next highest activation is among Class 2 neurons, but it is < 0.4 and the Class 3 neuron activation is also < 0.4. Hence the DNN's success rate in this case is 0%, as none of the features are recognised.



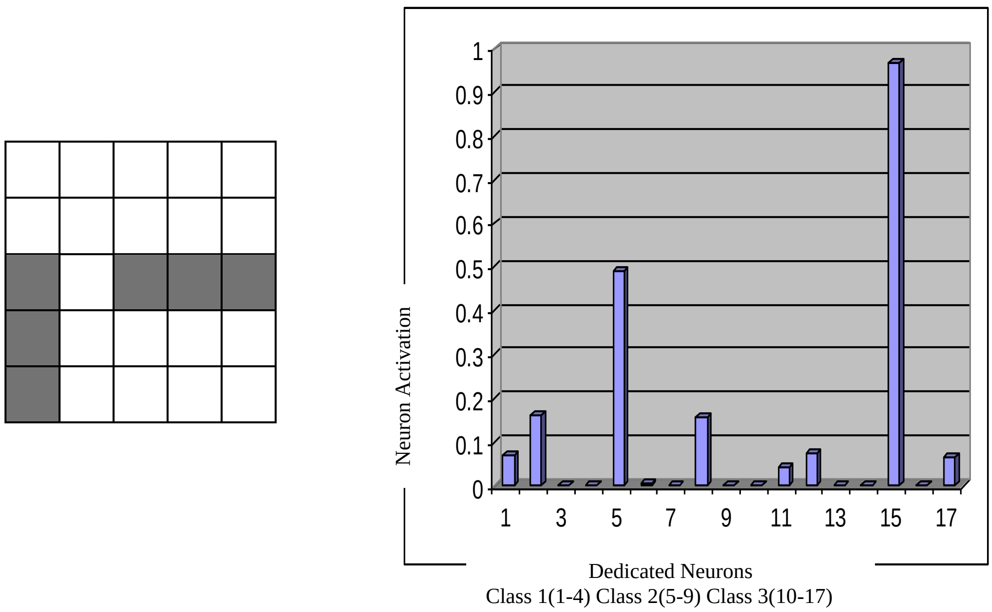


Graph 10.11: Results from tests on DNN using Combination of Class-2-Feature-2 and Class-3-Feature-1

In the Graph 10.12, the 1st highest activation is among the Class 3 neurons, which recognises the Class 3 feature in the image. The second highest activation belongs to Class 1 neurons, which is not present in the image. Although, the Class 2 features have activation > 0.4, this is not considered as a success as this is not among DNN's top two class preferences. Hence the success rate is 50%.



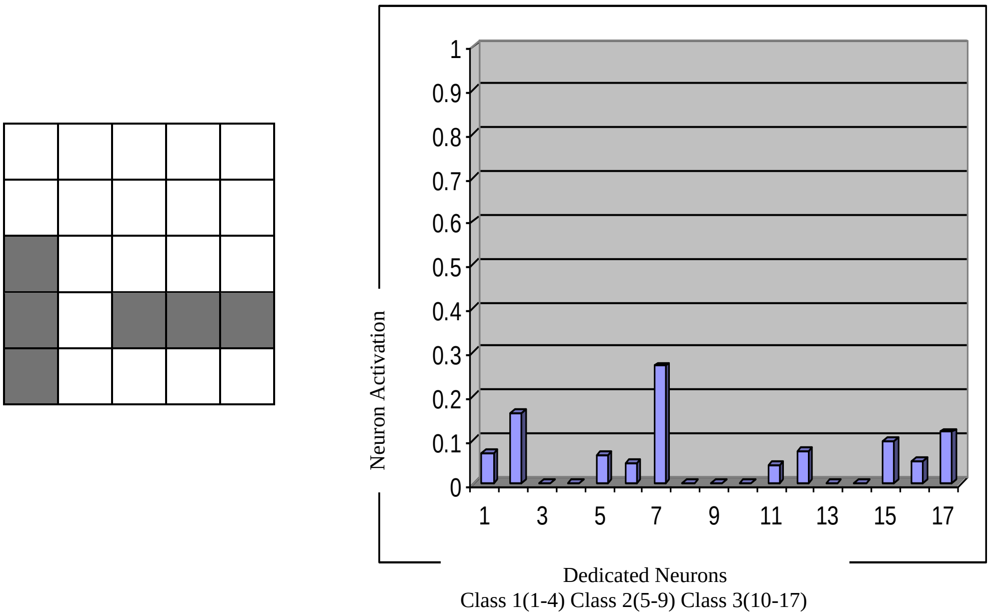
Graph 10.12: Results from tests on DNN using Combination of Class-2-Feature-3 and Class-3-Feature-1



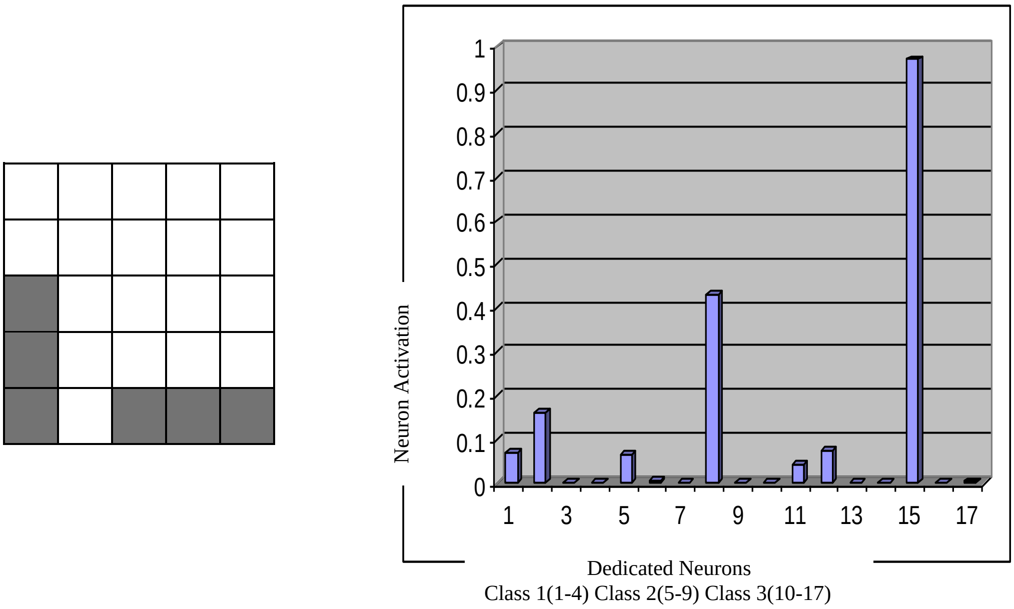
Graph 10.13: Results from tests on DNN using Combination of Class-2-Feature-1 and Class-3-Feature-2

In the Graph, 10.14, none of the features are recognised, as all features are <0.4.

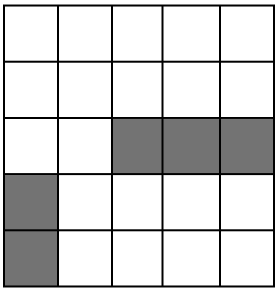
Hence, the success rate of DNN is 0%.

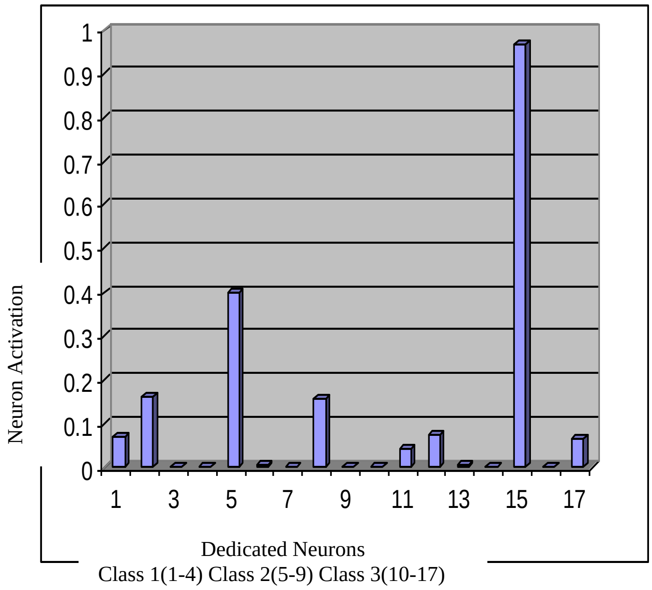


Graph 10.14: Results from tests on DNN using Combination of Class-2-Feature-2 and Class-3-Feature-2

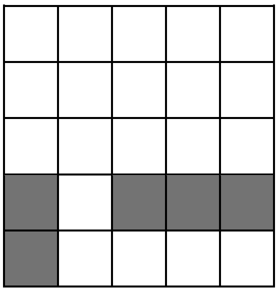


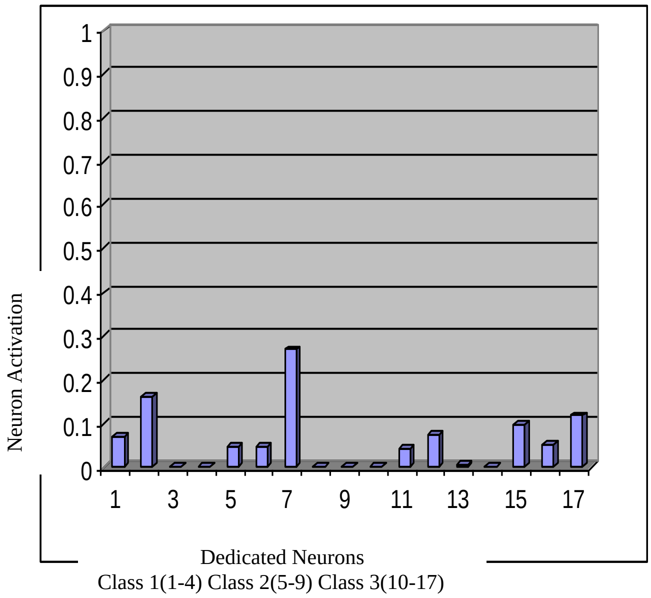
Graph 10.15: Results from tests on DNN using Combination of Class-2-Feature-3 and Class-3-Feature-2





Graph 10.16: Results from tests on DNN using Combination of Class-2-Feature-1 and Class-3-Feature-3

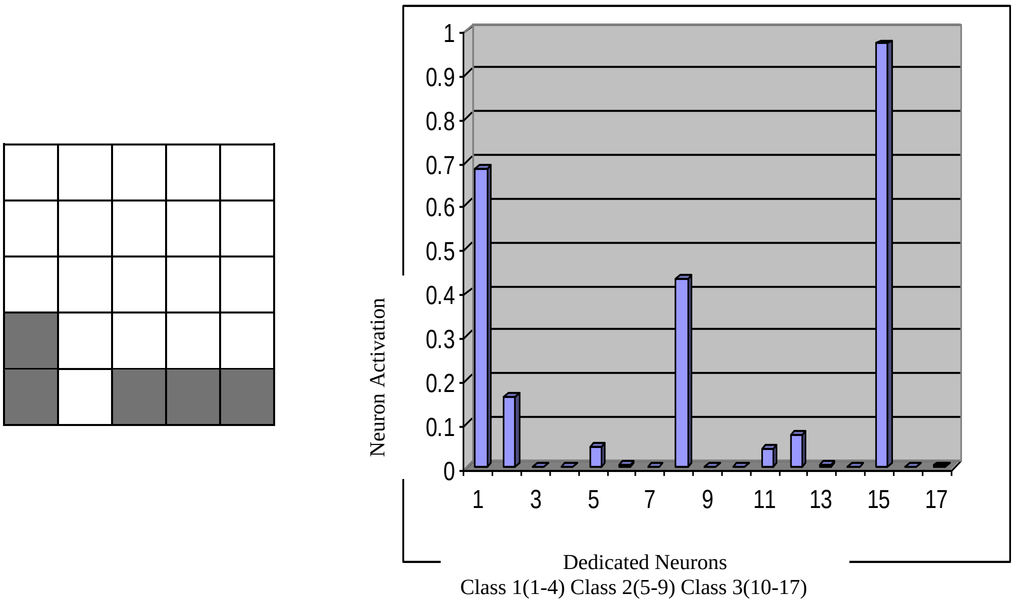




Graph 10.17: Results from tests on DNN using Combination of Class-2-Feature-2 and Class-3-Feature-3

In Graph 10.17, it can be seen that none of the features were recognised.

Graph 10.18 shows that the Class 3 feature is recognised and hence the DNN has 50% success rate due to the recognition of at least one of the features.

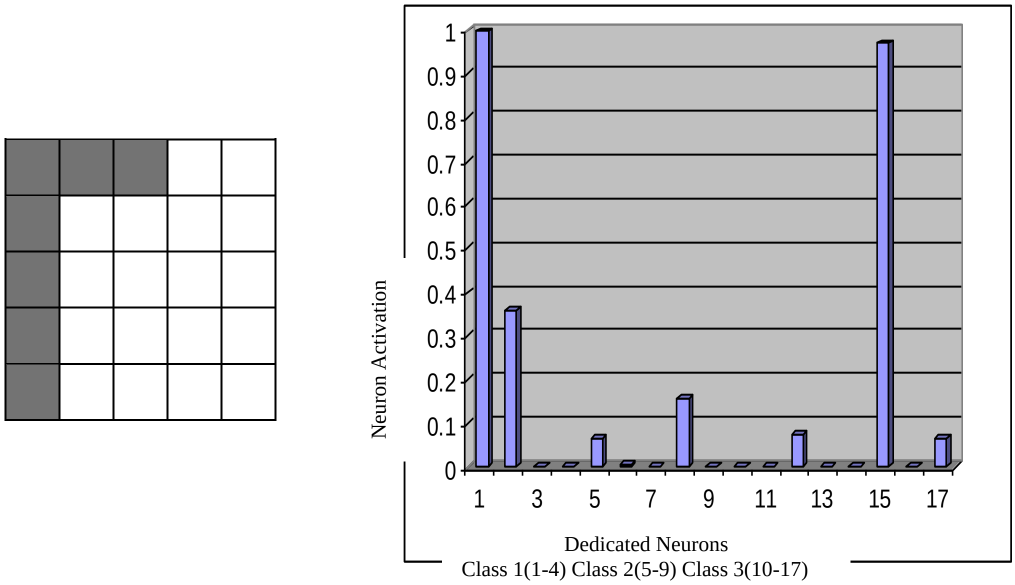


Graph 10.18: Results from tests on DNN using Combination of Class-2-Feature-3 and Class-3-Feature-3

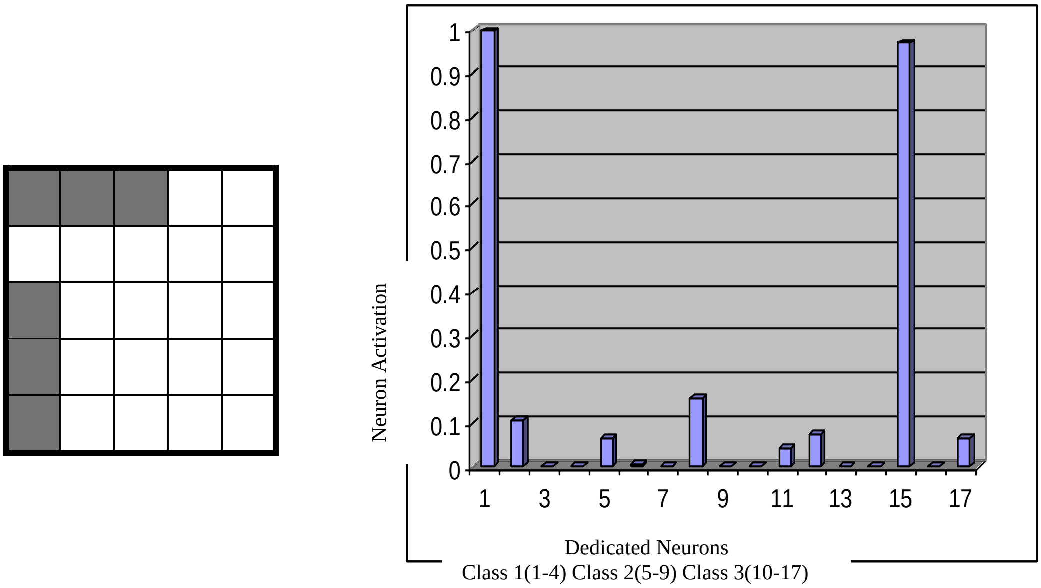
Combinations of Class 1 and Class 3 Features

The final set of combination of two features comprise of images that were made up of Class 1and Class 2 features. There are 9 possible combinations of these features.

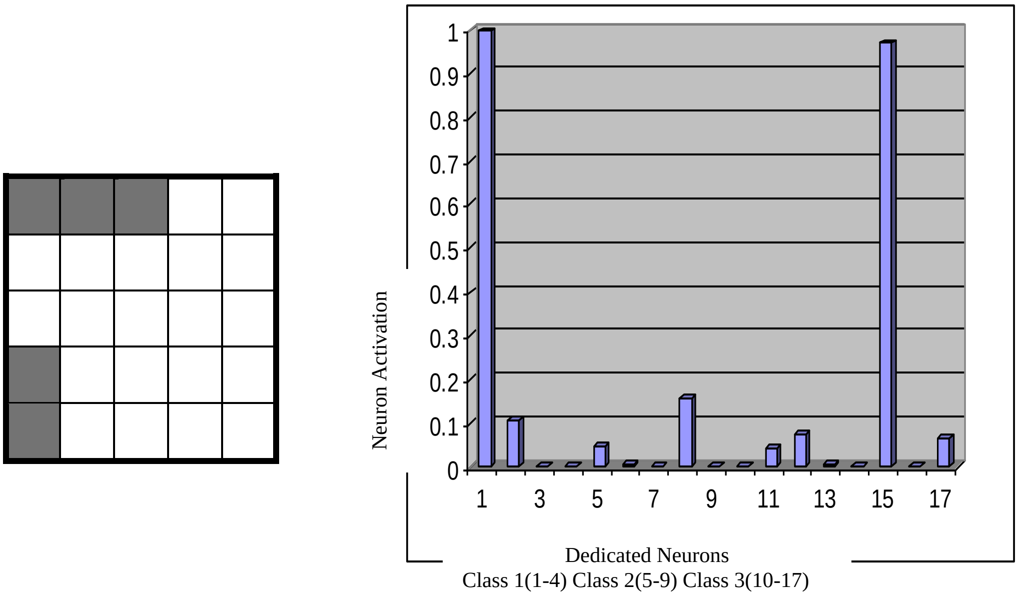
The results are presented in Graphs 10.19-10.27. From the activation values of various classes of neurons and the criteria laid out for recognition of features, it can be concluded that all the features were recognised and the DNN had a 100% success rate in classification.



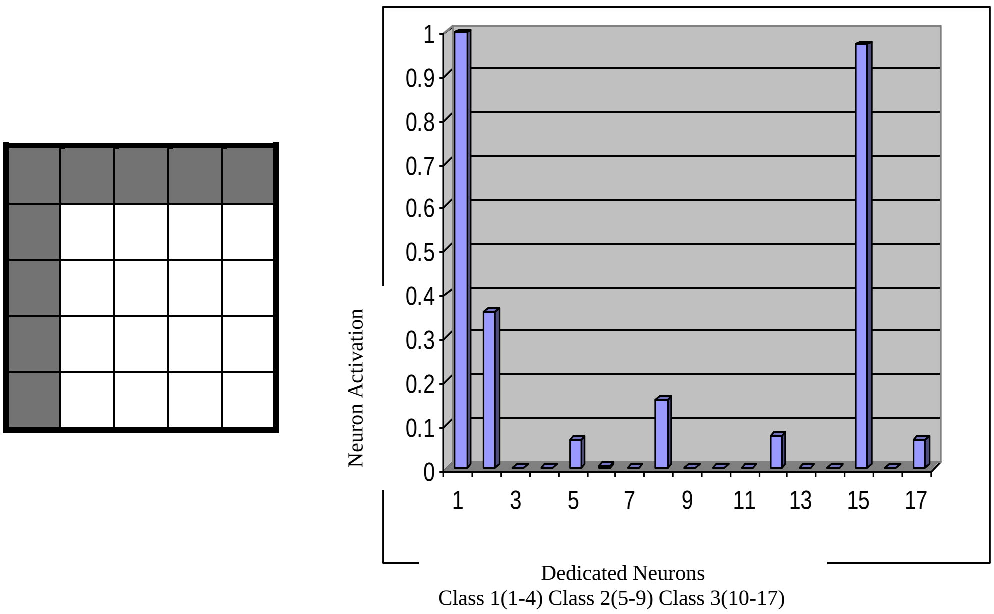
Graph 10.19: Results from tests on DNN using Combination of Class-1-Feature-1 and Class-3-Feature-1



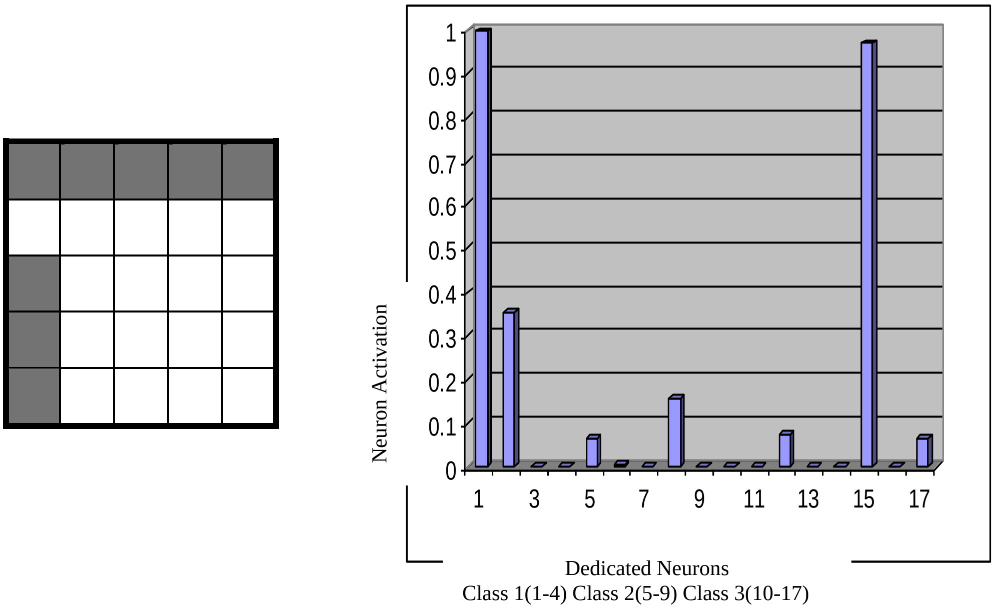
Graph 10.20: Results from tests on DNN using Combination of Class-1-Feature-1 and Class-3-Feature-2



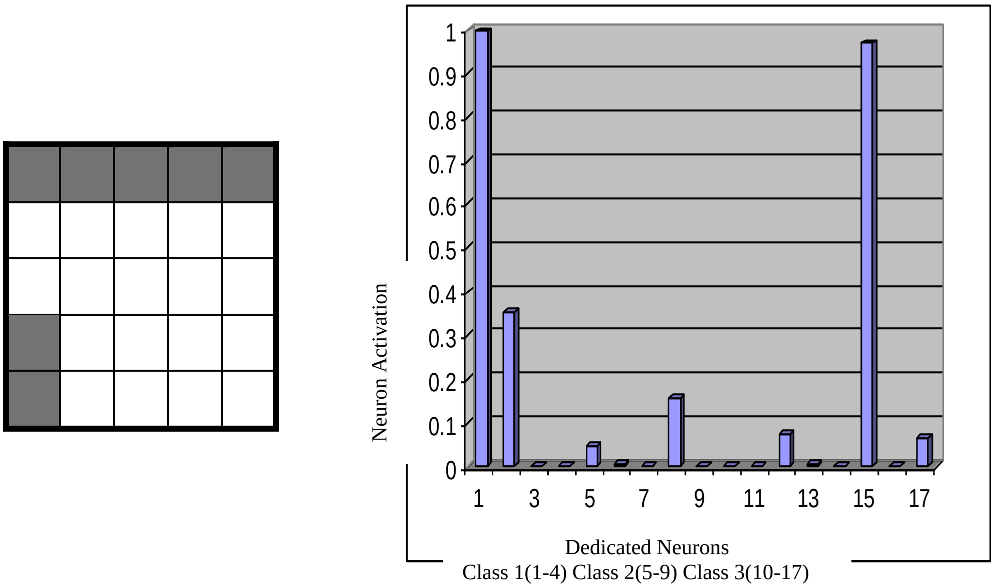
Graph 10.21: Results from tests on DNN using Combination of Class-1-Feature-1 and Class-3-Feature-3



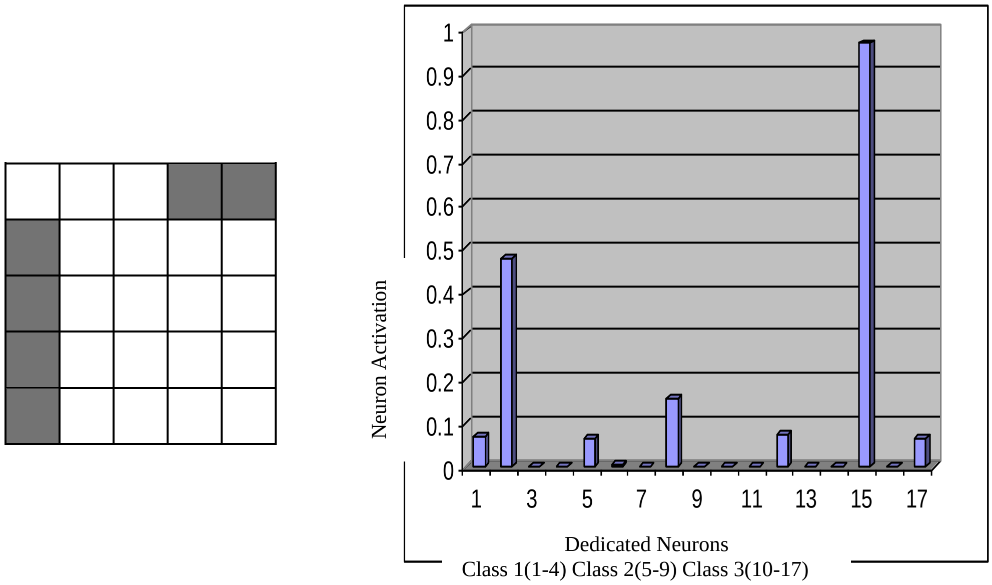
Graph 10.22: Results from tests on DNN using Combination of Class-1-Feature-2 and Class-3-Feature-1



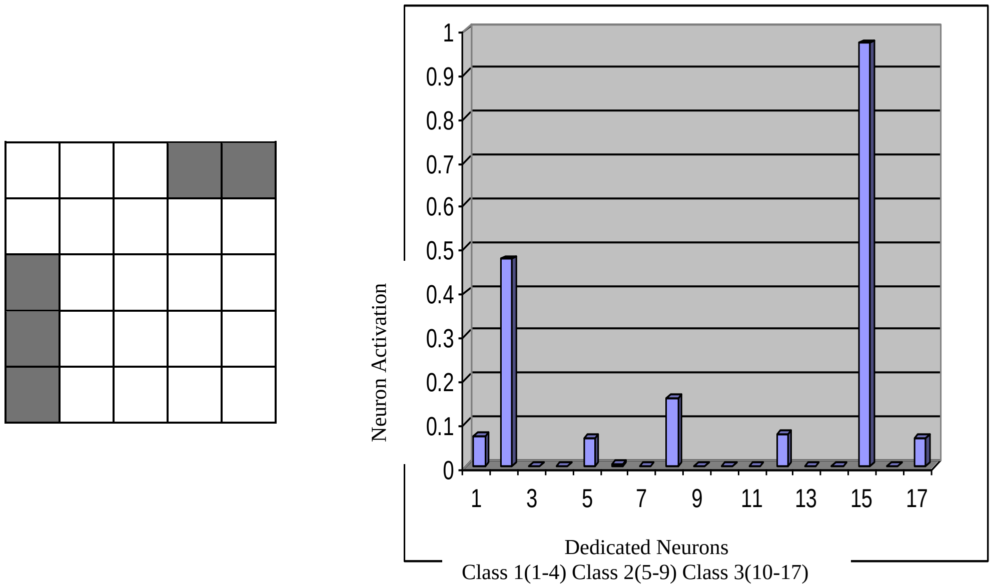
Graph 10.23: Results from tests on DNN using Combination of Class-1-Feature-2 and Class-3-Feature-2



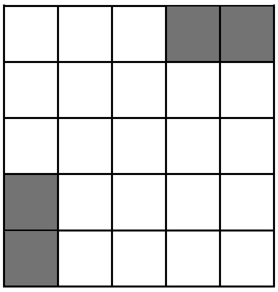
Graph 10.24: Results from tests on DNN using Combination of Class-1-Feature-2 and Class-3-Feature-3

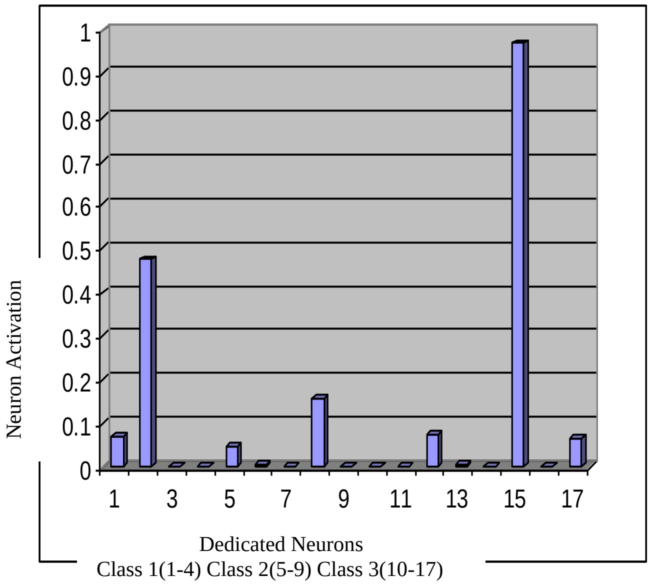


Graph 10.25: Results from tests on DNN using Combination of Class-1-Feature-3 and Class-3-Feature-1



Graph 10.26: Results from tests on DNN using Combination of Class-1-Feature-3 and Class-3-Feature-2





Graph 10.27: Results from tests on DNN using Combination of Class-1-Feature-3 and Class-3-Feature-3

Graph 10.1 – Graph 10.27 show the results obtained when combinations of two known features were used to test the DNN. The combinations of features were never included in the training set. The individual features in the combination were features that the DNN was trained to classify.

The results show that unlike traditional ANNs, the DNN is capable of recognising known individual features from an unfamiliar combination of features. An evaluation of results is presented in Table 10.1.

The results suggest that the DNN utilises its inherent modularity in recognising classes of features in whatever combination they might appear.

Combination	Features									Success Rate In Recognition of Combinations of Two Features		
	Class-1			Class-2			Class-3					
	1	2	3	1	2	3	1	2	3	100%	50%	0%
1	√			√						√		
2	√				√					√		
3	√					√				√		
4		√		√						√		
5		√			√					√		
6		√				√				√		
7			√	√						√		
8			√		√					√		
9			√			√				√		
10				√			√			√		
11				√				√		√		
12				√					√	√		
13					√		√					√
14					√			√				√
15					√				√			√
16						√	√				√	
17						√		√		√		
18						√			√		√	
19	√						√			√		
20	√							√		√		
21	√								√	√		
22		√					√			√		

23		√						√		√		
24		√							√	√		
25			√				√			√		
26			√					√		√		
27			√						√	√		

Table 10.1: Evaluation of Success Rate in Recognising Combinations of Two Features

With 9 features in the training set, a total of 27 possible combinations of two features were used to test the DNN. Table 10.1 collates the results obtained when the DNN was tested for combinations of two features.

There are 3 possible outcomes when an image with a combination of two features is tested with the DNN; the first being both features are recognised, the second being one of the features being recognised and finally the third case where neither of the features are recognised. Each of these cases is represented by a success rate of 100%, 50% and 0% respectively.

In case of combinations of Class-2-Feature-2 with Class 3 features, there was a low activation (<0.4) among the Class 2 dedicated neurons. Class-1 neurons were highly excited despite absence in the image and the Class-3 neurons showed very poor activation. The success rate was considered as 0 % as the Class-2 neurons were very weakly active compared to other neurons.

Also, the test with combinations of Class-2-Feature-3 & Class-3-Feature-1 and combination of Class-2-Feature–3 & Class-3-Feature-3, the Class-1 neurons showed a high activation. The success rate was considered as 50% as only one feature was recognised.

It can be concluded from the results obtained that the DNN was highly successful in recognising various combinations of two features. The next phase was to test its ability to recognise combinations of three features.

10.3 COMBINATION OF THREE FEATURES

This section presents the results obtained when 27 possible combinations of 3 features were used to test the DNN. For example the Figure 10.2 illustrates an image consisting of three distinct features that belong to three different classes.

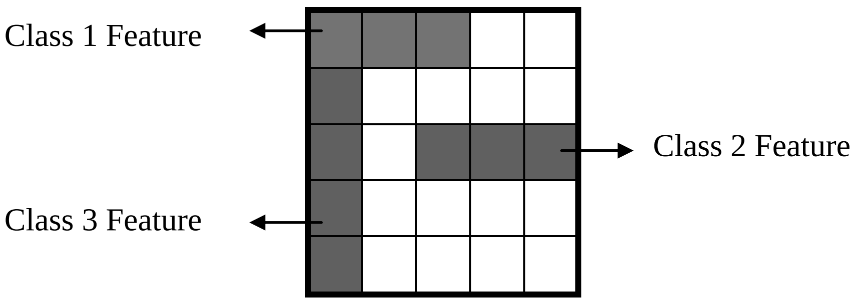


Figure 10.2: Illustration of Combination of Three Features

The recognition criteria are based on the following algorithm.

Find Neurons with highest activation

Is activation for each of the neurons ≥0.4?

Do they belong to all three classes? **No**

Yes

100% success rate

All three features recognised

Find Number of Neurons with Activation ≥0.4

Is it 2 Classes? **No**

Yes

66% Success Rate

Is it 1 Class? **No**

Yes

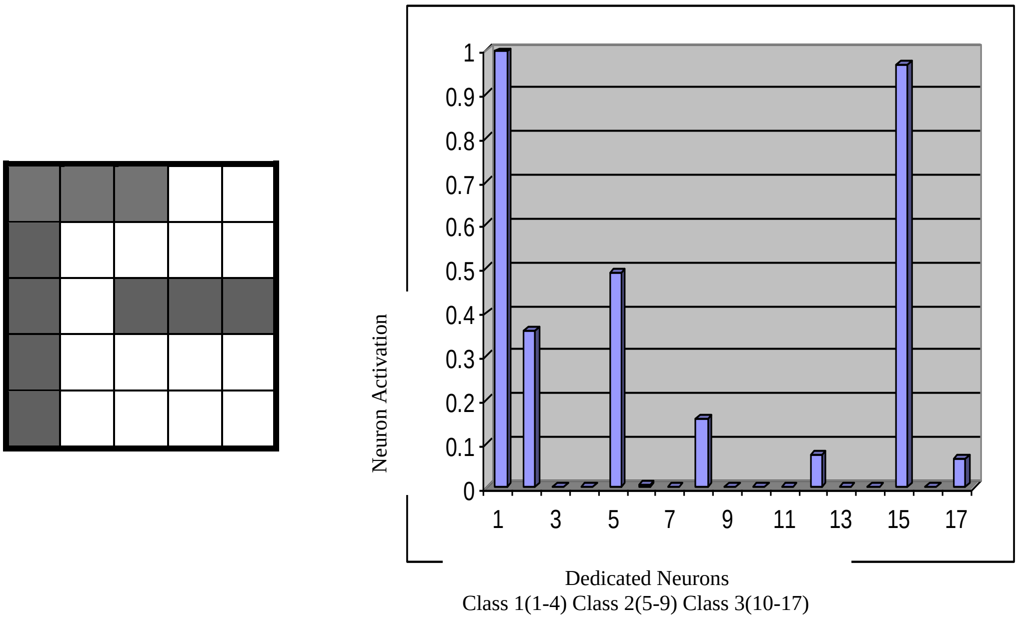
33%% Success Rate

Is it 0 Classes?

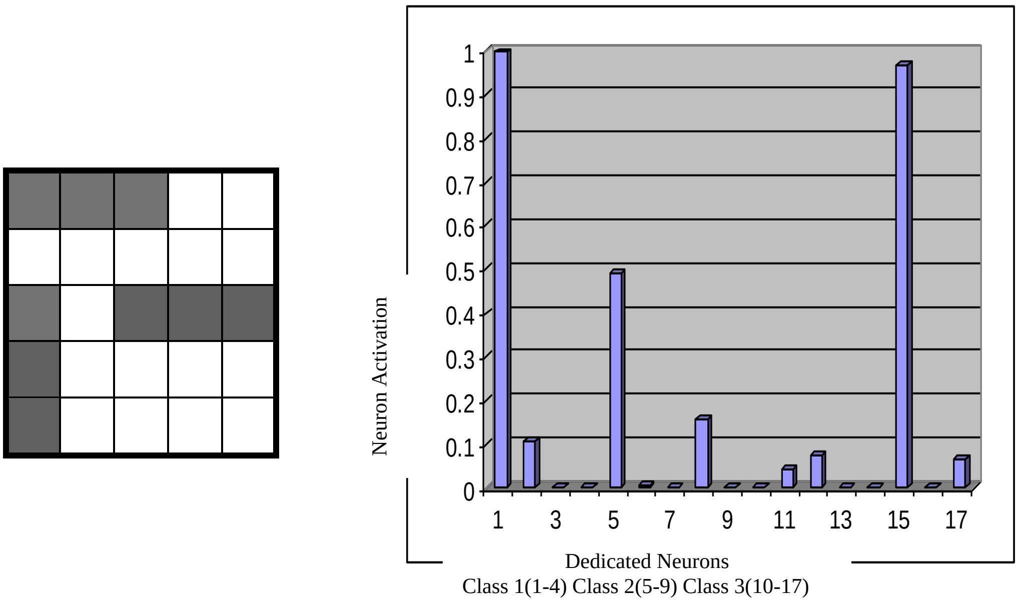
Yes

0% Success Rate

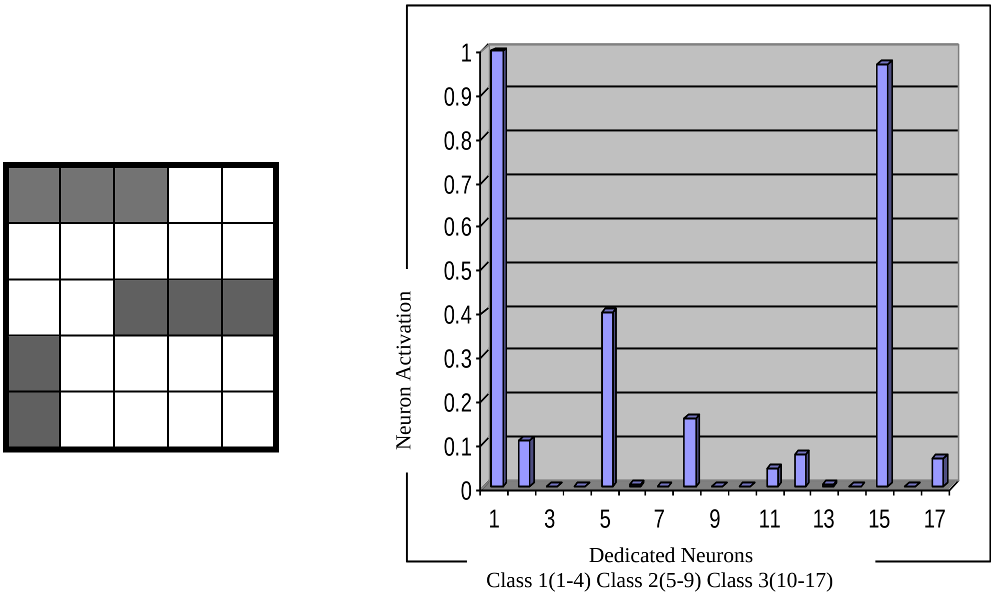
Graphs 10.28 and 10.29 illustrate examples of 100% success rate as the 3 classes of features have dedicated neurons, which have activation ≥ 0.4.



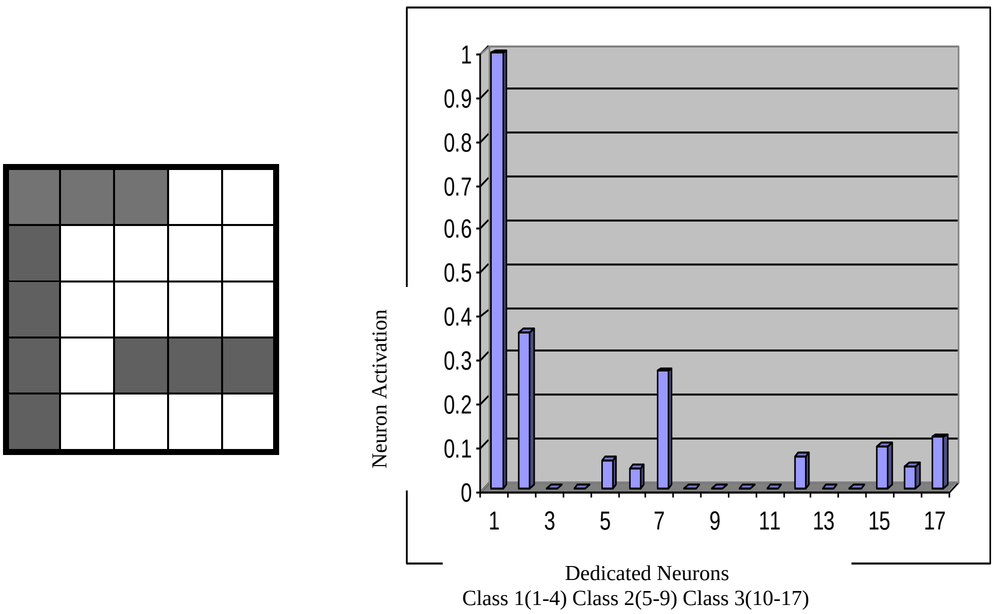
Graph 10.28: Results from tests on DNN using Combination of Class-1-Feature-1, Class-2-Feature-1 and Class-3-Feature-1



Graph 10.29: Results from tests on DNN using Combination of Class-1-Feature-1, Class-2-Feature-1 and Class-3- Feature-2

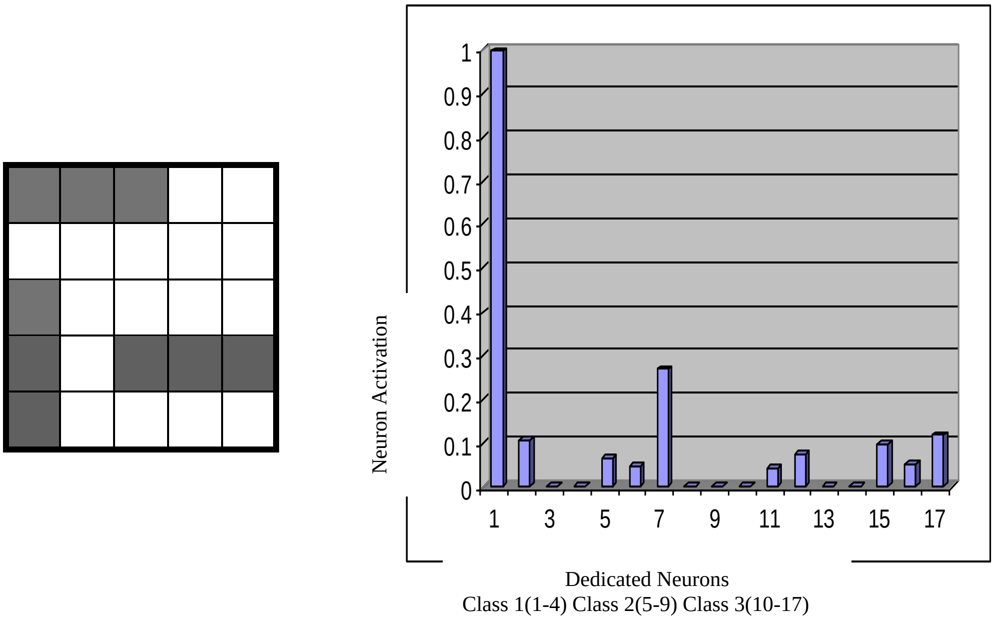


Graph 10.30: Results from tests on DNN using Combination of Class-1-Feature-1, Class-2-Feature-1 and Class-3-Feature-3

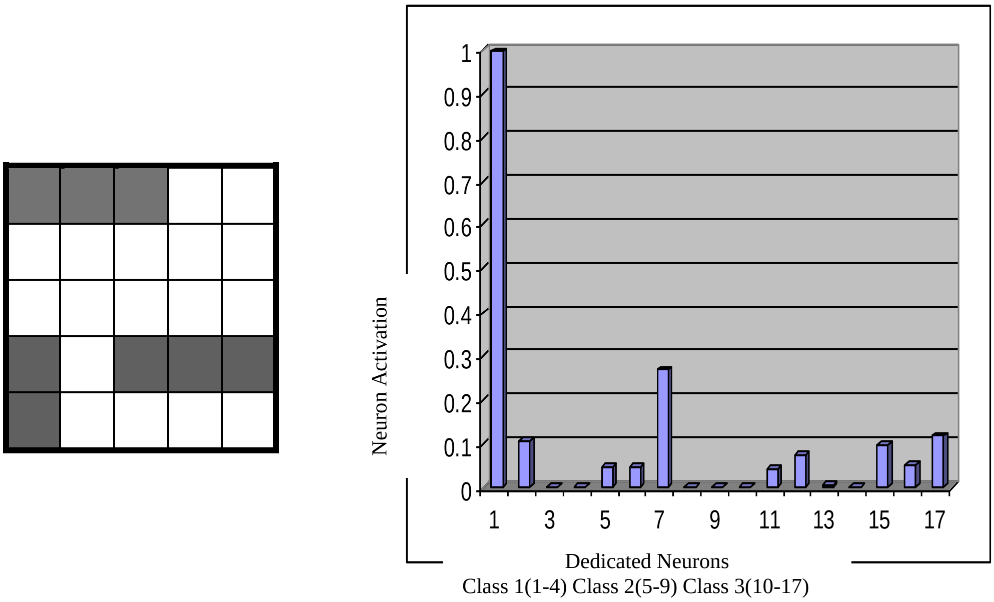


Graph 10.31: Results from tests on DNN using Combination of Class-1-Feature-1, Class-2-Feature-2 and Class-3-Feature-1

In Graph 10.31, only one class has been recognised as the corresponding dedicated neurons show activation ≥ 0.4. Hence the success rate is 33%.

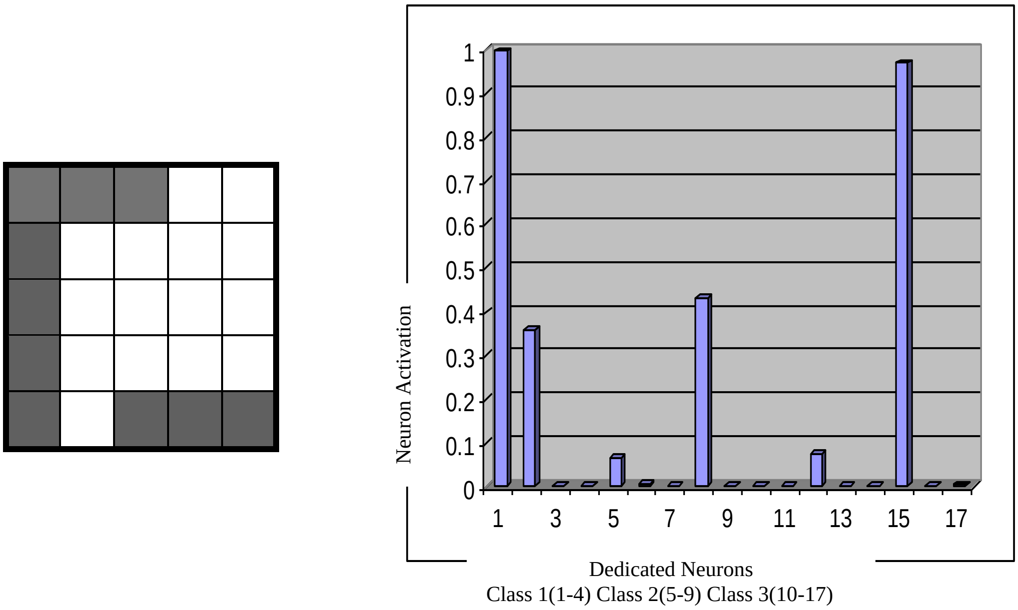


Graph 10.32: Results from tests on DNN using Combination of Class-1-Feature-1, Class-2-Feature-2 and Class-3-Feature-2

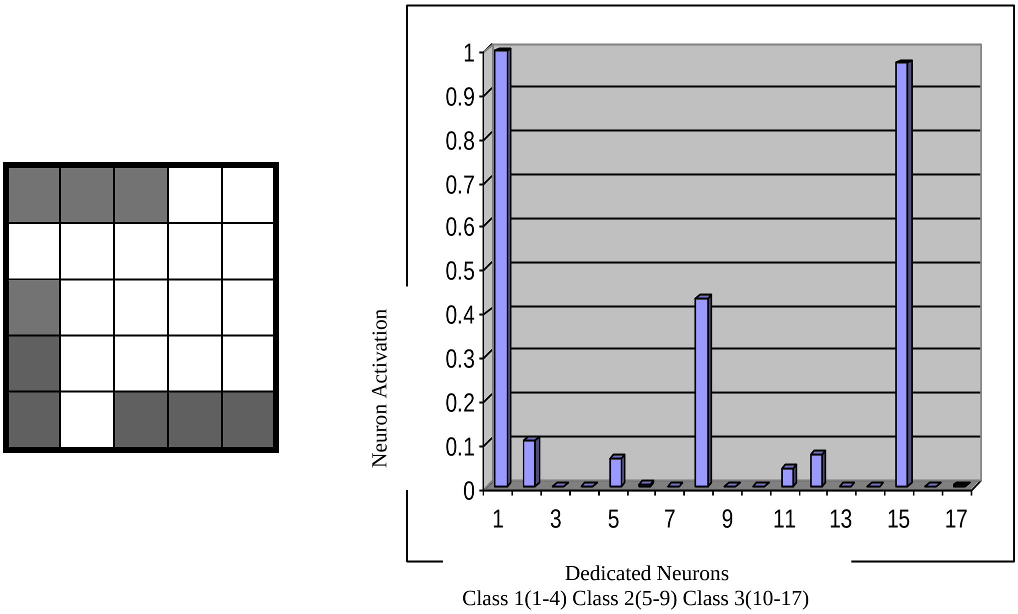


Graph 10.33: Results from tests on DNN using Combination of Class-1-Feature-1, Class-2-Feature-2 and Class-3-Feature-3

In Graph 10.32 and 10.33, only one class has corresponding dedicated neurons with activation ≥ 0.4. Hence the success rate is 33%.

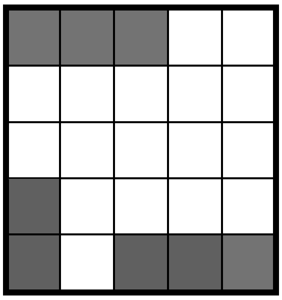


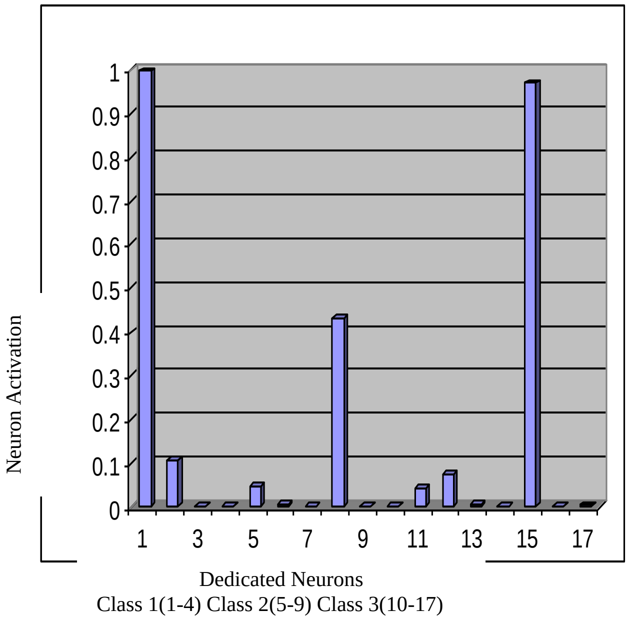
Graph 10.34: Results from tests on DNN using Combination of Class-1-Feature-1, Class-2-Feature-3 and Class-3-Feature-1



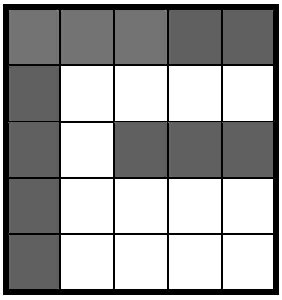
Graph 10.35: Results from tests on DNN using Combination of Class-1-Feature-1, Class-2-Feature-3 and Class-3-Feature-2

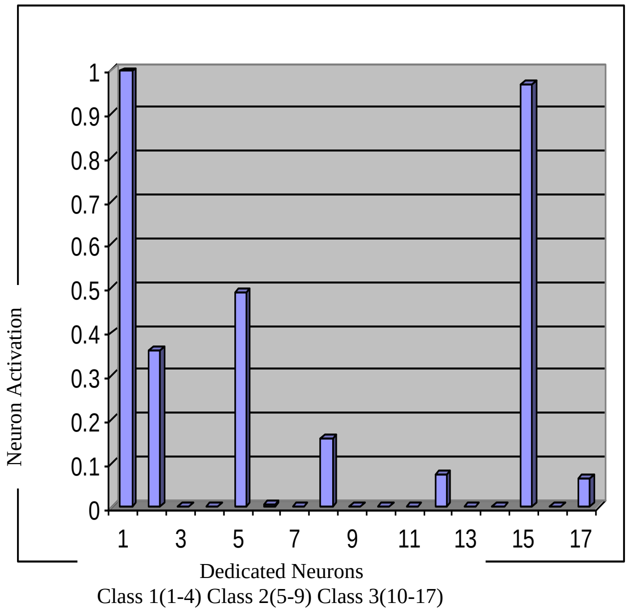
In Graph 10.34 and 10.35, all classes have corresponding dedicated neurons with activation ≥ 0.4. Hence the success rate is 100%.





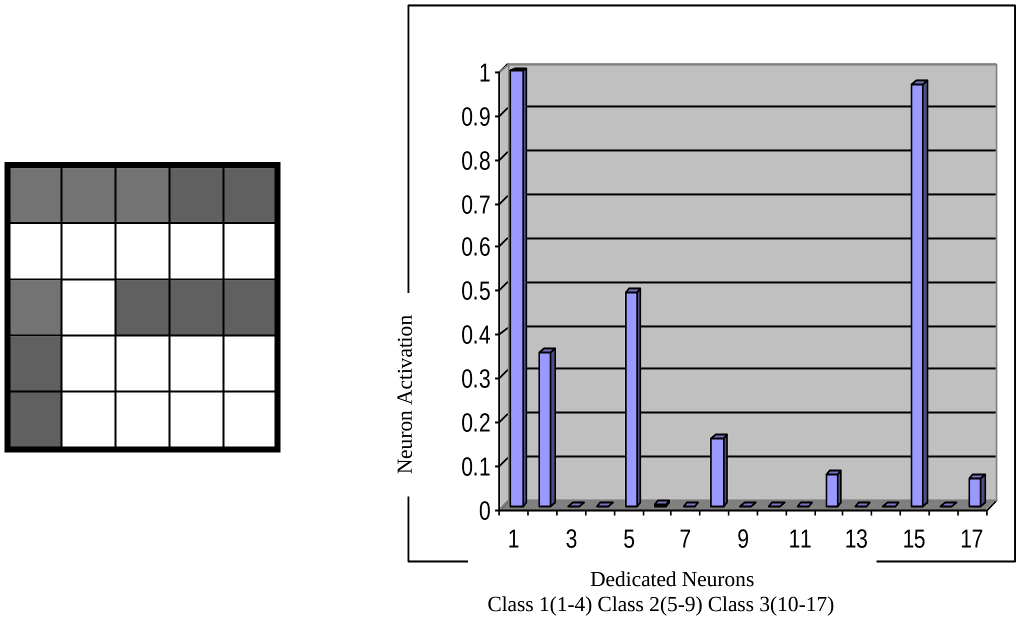
Graph 10.36: Results from tests on DNN using Combination of Class-1-Feature-1, Class-2-Feature-3 and Class-3-Feature-3



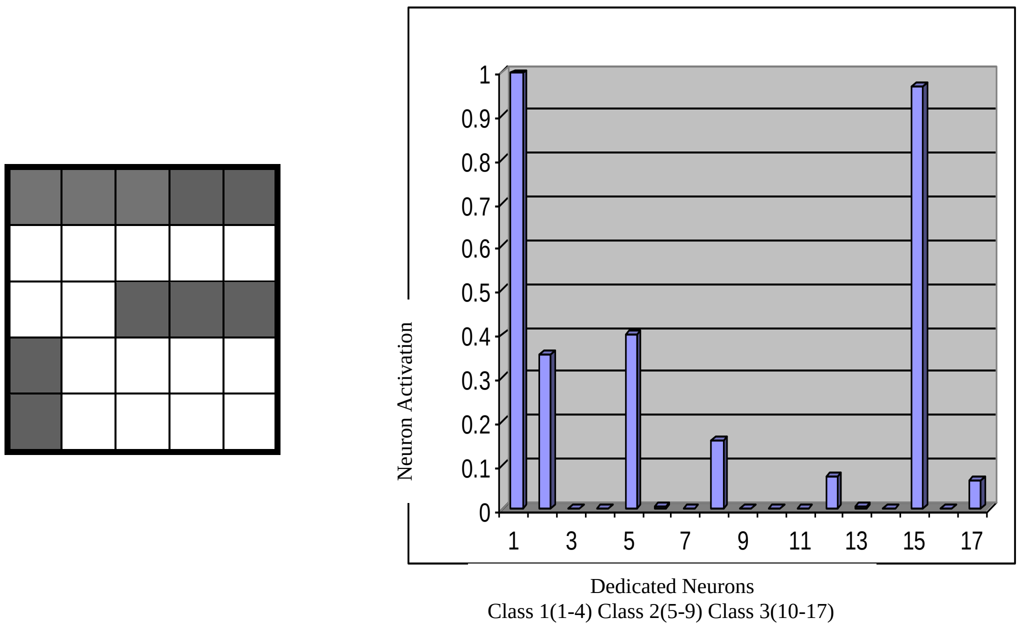


Graph 10.37: Results from tests on DNN using Combination of Class-1-Feature-2, Class-2-Feature-1 and Class-3-Feature-1

In Graph 10.36 and 10.37, all classes have corresponding dedicated neurons with activation ≥ 0.4. Hence the success rate is 100%.

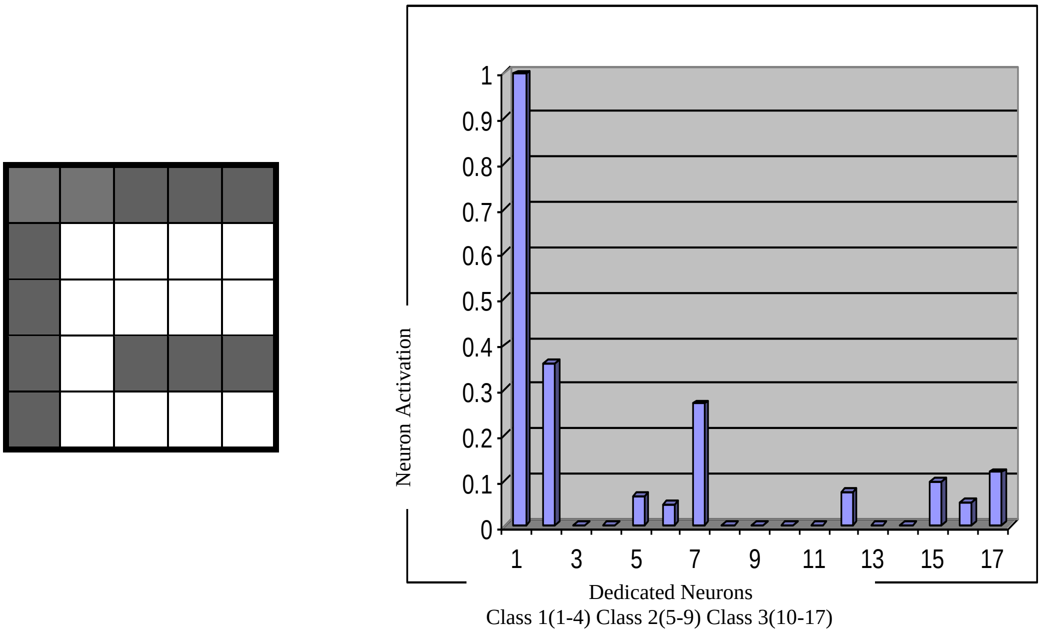


Graph 10.38: Results from tests on DNN using Combination of Class-1-Feature-2, Class-2-Feature-1 and Class-3-Feature-2

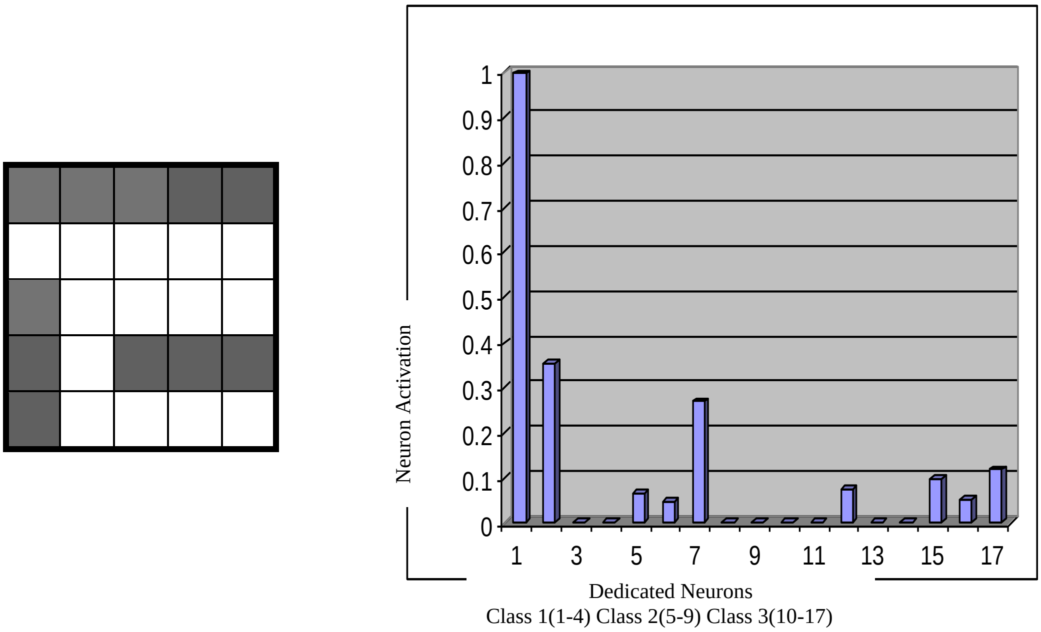


Graph 10.39: Results from tests on DNN using Combination of Class-1-Feature-2, Class-2-Feature-1 and Class-3-Feature-3

In Graph 10.38 and 10.39, all classes have corresponding dedicated neurons with activation ≥ 0.4. Hence the success rate is 100%.



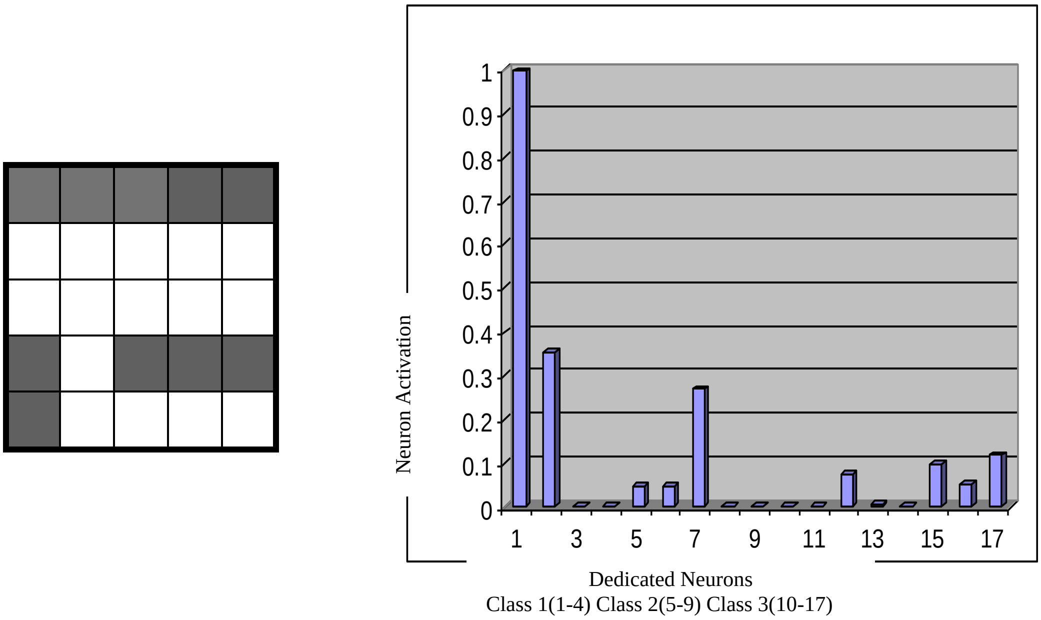
Graph 10.40: Results from tests on DNN using Combination of Class-1-Feature-2, Class-2-Feature-2 and Class-3-Feature-1



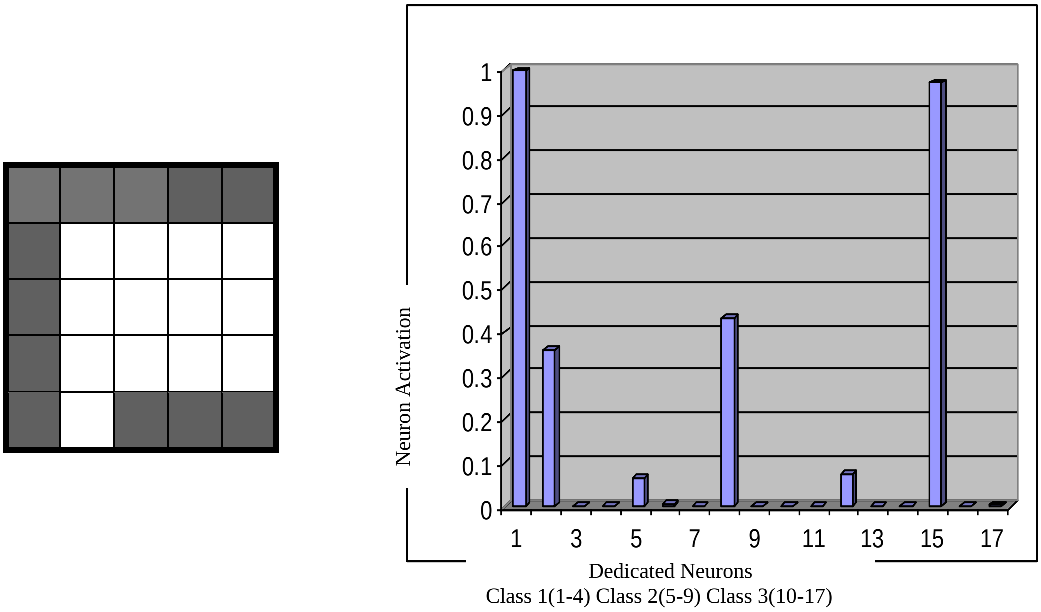
Graph 10.41: Results from tests on DNN using Combination of Class-1-Feature-2, Class-2-Feature-2 and Class-3-Feature-2

In Graph 10.40 and 10.41, only one class has corresponding dedicated neurons with activation ≥ 0.4. Hence the success rate is 33%.

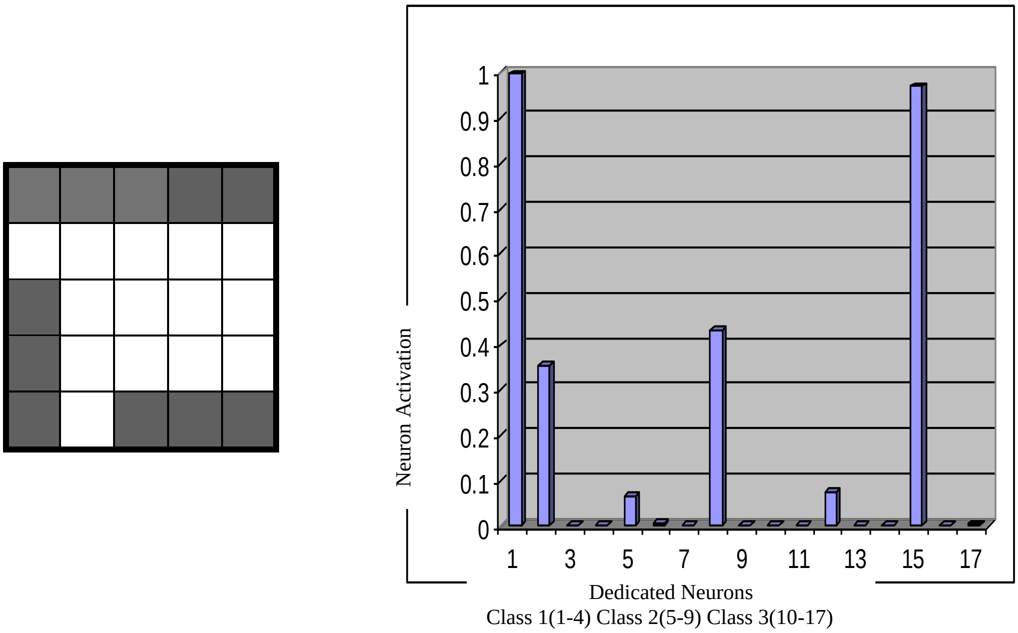
In Graph 10.42, the only neuron belonging to Class 1 has activation ≥ 0.4. Hence the success rate is 33%.



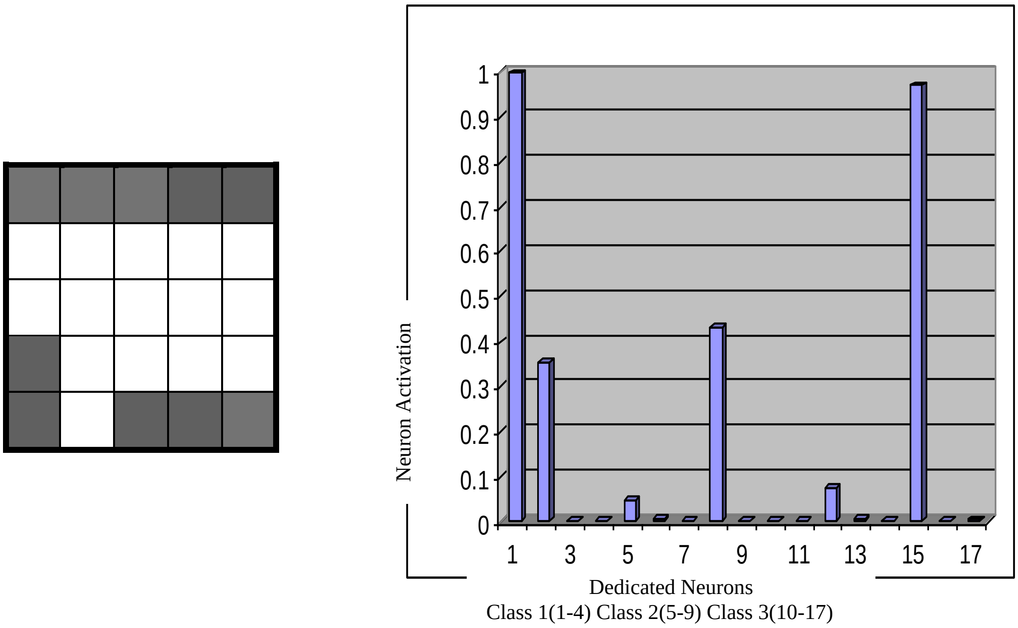
Graph 10.42: Results from tests on DNN using Combination of Class-1-Feature-2, Class-2-Feature-2 and Class-3-Feature-3



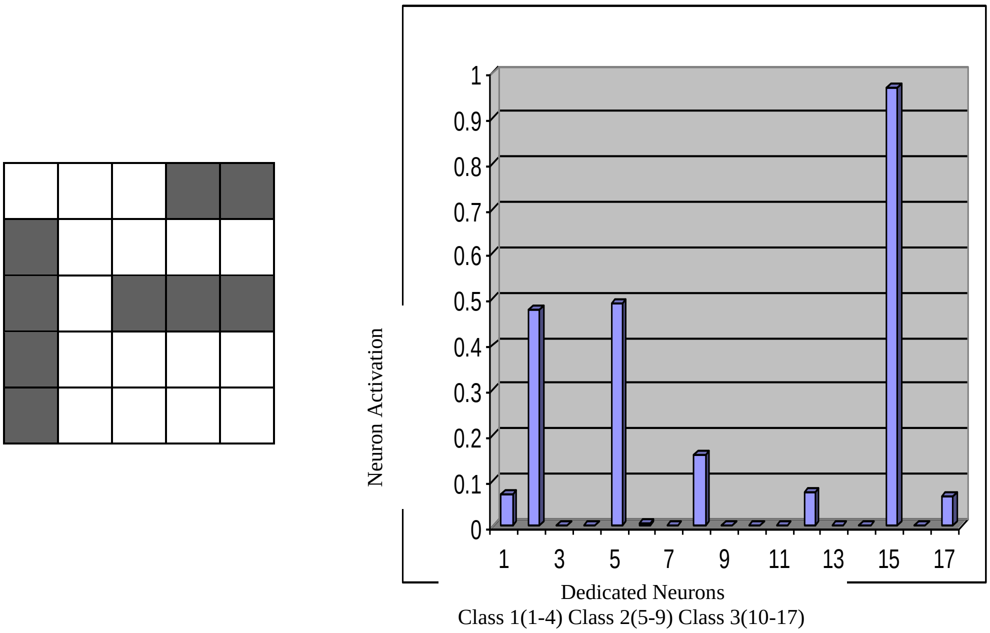
Graph 10.43: Results from tests on DNN using Combination of Class-1-Feature-2, Class-2-Feature-3 and Class-3-Feature-1



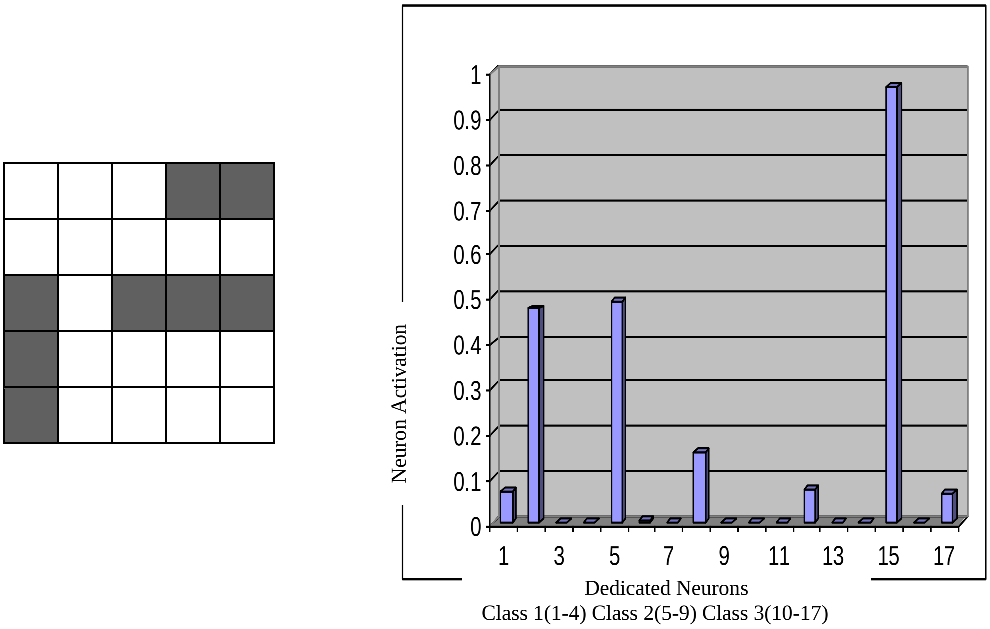
Graph 10.44: Results from tests on DNN using Combination of Class-1-Feature-2, Class-2-Feature-3 and Class-3-Feature-2



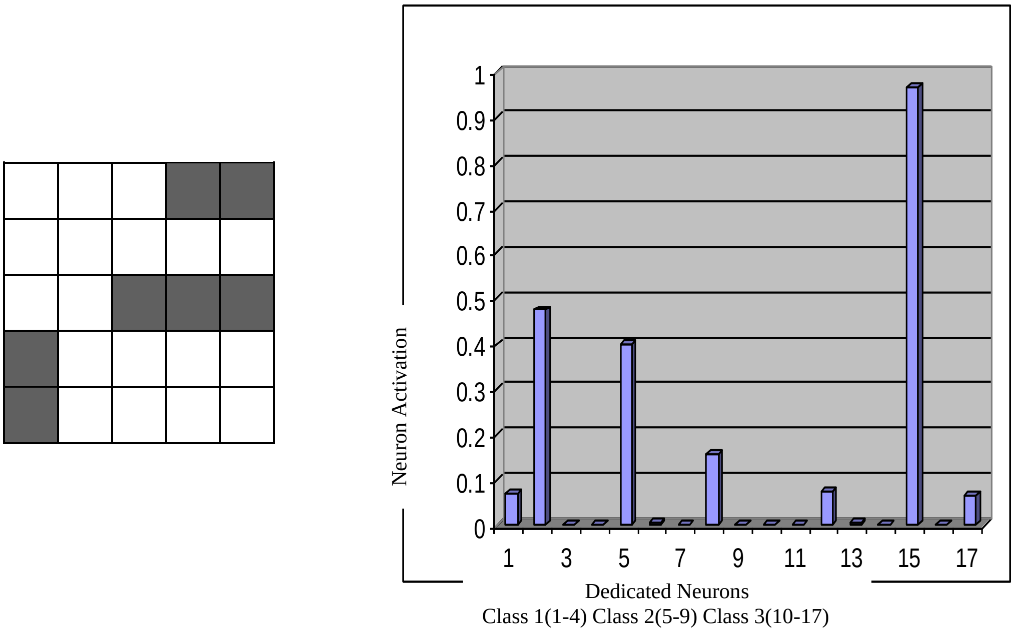
Graph 10.45: Results from tests on DNN using Combination of Class-1-Feature-2, Class-2-Feature-3 and Class-3-Feature-3



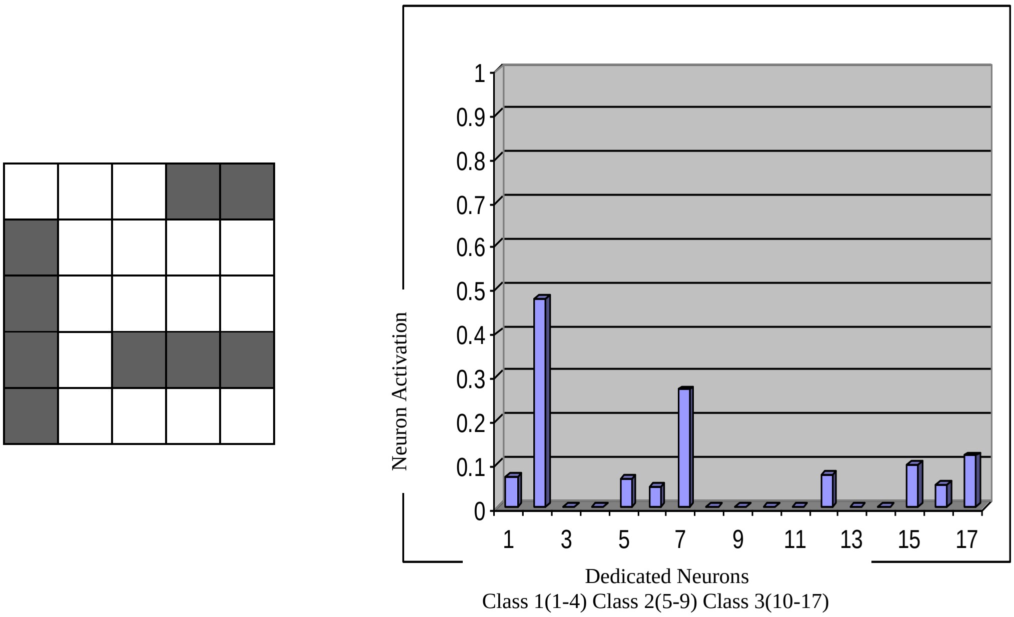
Graph 10.46: Results from tests on DNN using Combination of Class-1-Feature-3, Class-2-Feature-1 and Class-3-Feature-1



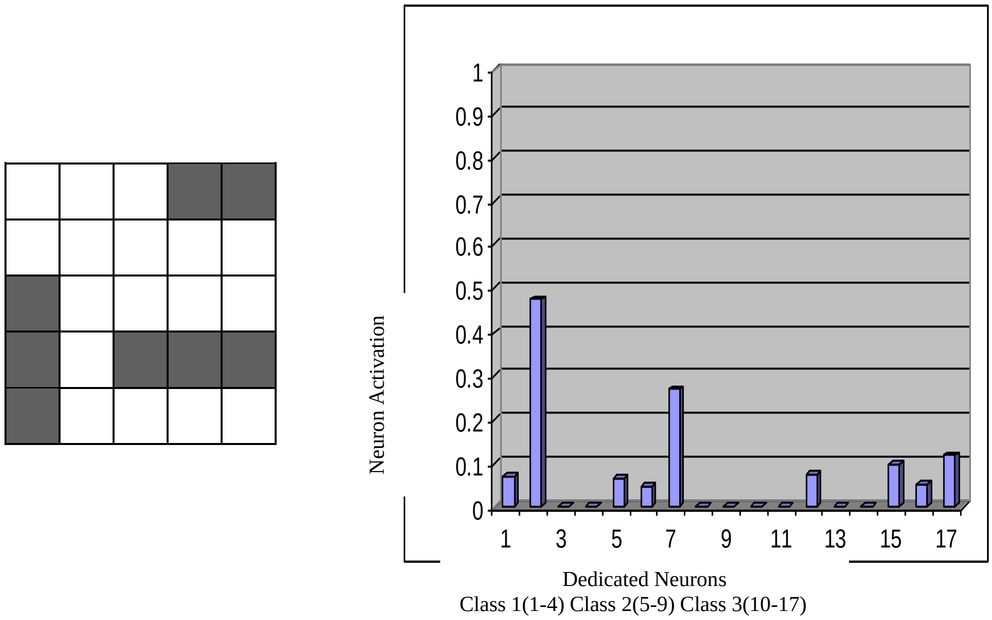
Graph 10.47: Results from tests on DNN using Combination of Class-1-Feature-3, Class-2-Feature-1 and Class-3-Feature-2



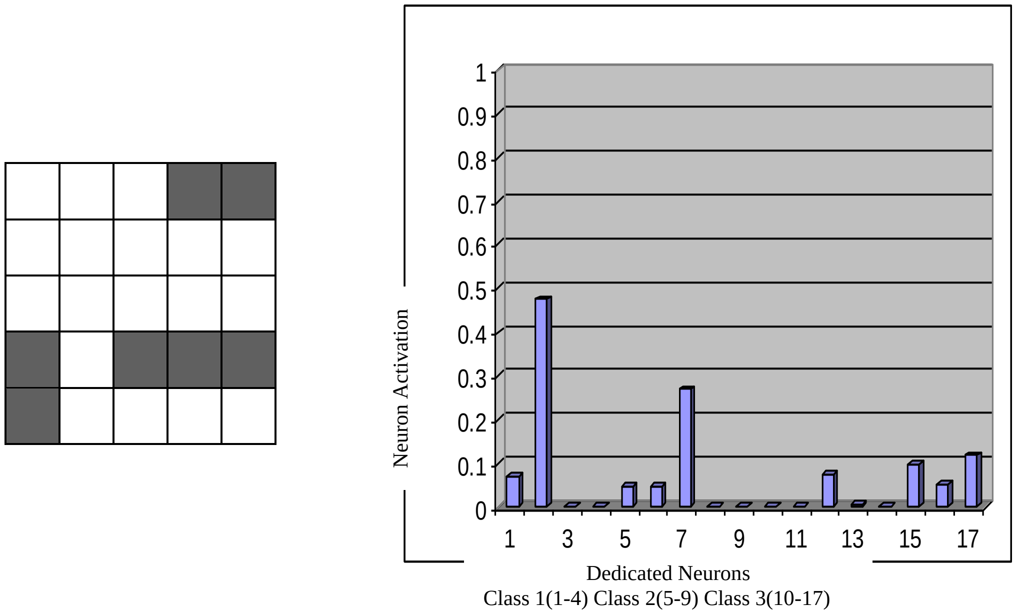
Graph 10.48: Results from tests on DNN using Combination of Class-1-Feature-3, Class-2-Feature-1 and Class-3-Feature-3



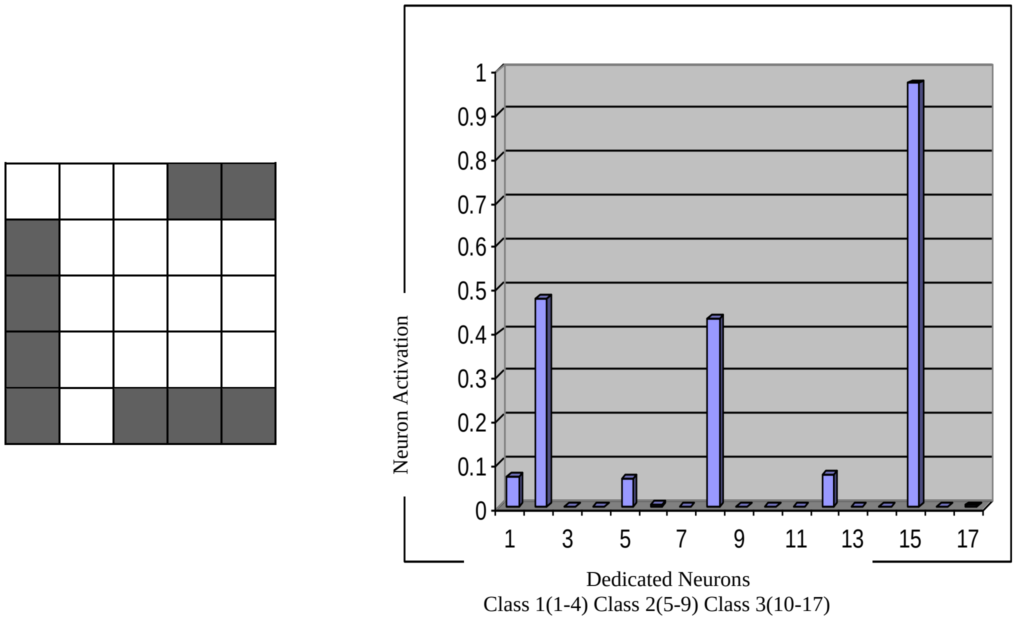
Graph 10.49: Results from tests on DNN using Combination of Class-1-Feature-3, Class-2-Feature-2 and Class-3-Feature-1



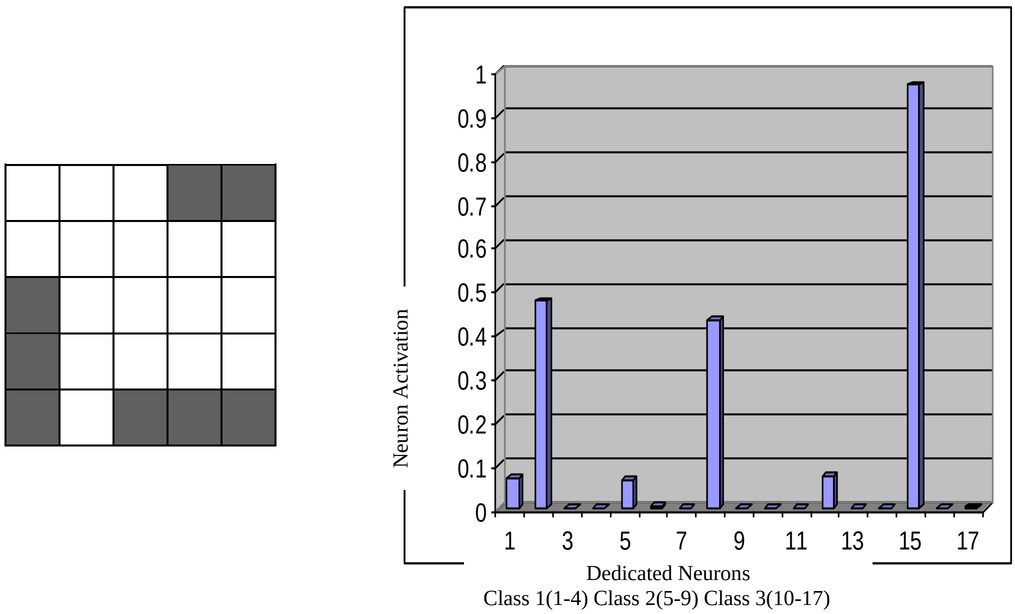
Graph 10.50: Results from tests on DNN using Combination of Class-1-Feature-3, Class-2-Feature-2 and Class-3-Feature-2



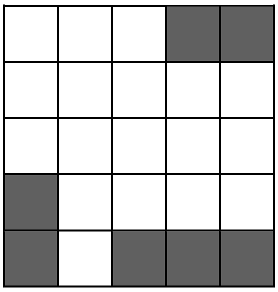
Graph 10.51: Results from tests on DNN using Combination of Class-1-Feature-3, Class-2-Feature-2 and Class-3-Feature-3

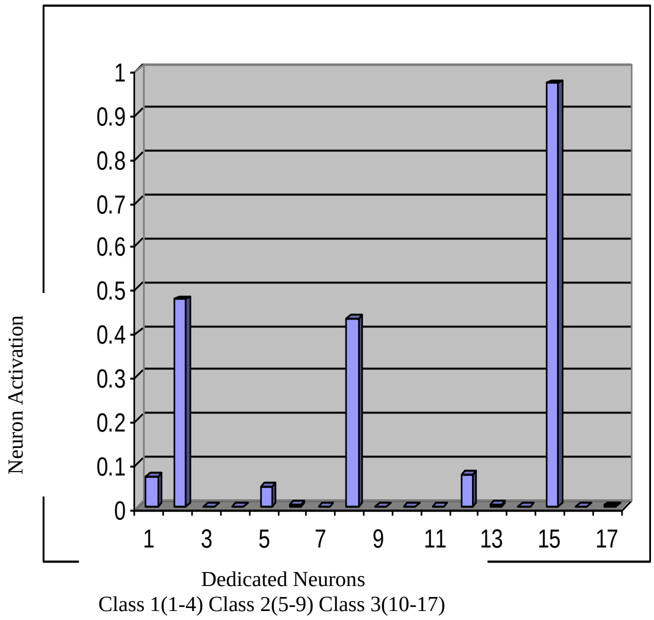


Graph 10.52: Results from tests on DNN using Combination of Class-1-Feature-3, Class-2-Feature-3 and Class-3-Feature-1



Graph 10.53: Results from tests on DNN using Combination of Class-1-Feature-3, Class-2-Feature-3 and Class-3-Feature-2





Graph 10.54: Results from tests on DNN using Combination of Class-1-Feature-3, Class-2-Feature-3 and Class-3-Feature-3

Graphs 10.28-10.54 are illustrations of the results obtained when tested with images that contained combinations of three features. The recognition criteria was based on the activation being ≥ 0.4.

It can be observed from the results that the DNN has one very active neuron among the dedicated neurons that belong to a class, although there is more than one dedicated neuron for each class. This makes it easier to categorise the 3 features in an image. In a scenario, where all dedicated neurons of all 3 features are very high, it would be very tricky to classify the features.

The results of the success rate of various combinations of features being recognised accurately is presented in Table 10.2.

Combination	Features									Success Rate In Recognition of Combinations of 3 features		
	Class-1			Class-2			Class-3					
	1	2	3	1	2	3	1	2	3	100 %	66 %	33 %
1	√			√			√			√		
2	√			√				√		√		
3	√			√					√	√		
4	√				√		√					√
5	√				√			√				√
6	√				√				√			√
7	√					√	√			√		
8	√					√		√		√		
9	√					√			√	√		
10		√		√			√			√		
11		√		√				√		√		
12		√		√					√	√		
13		√			√		√					√
14		√			√			√				√
15		√			√				√			√
16		√				√	√			√		
17		√				√		√		√		
18		√				√			√	√		
19			√	√			√			√		
20			√	√				√		√		
21			√	√					√	√		

22			√		√		√					√
23			√		√			√				√
24			√		√				√			√
25			√			√	√			√		
26			√			√		√		√		
27			√			√			√	√		

Table 10.2: Evaluation of Success Rate in Recognising Combinations of Three Features

In case of using combinations of three features, the success rate was decided based on whether all 3, 2, 1 or none of the features were recognised as 100%, 66%, 33% or 0%.

66.67% of the population of test data that comprised combination of three features had a success rate of 100% in recognising the features. 33.33% of the population of the test set achieved a success rate of 33% as shown in Table 10.2.

Hence from the results obtained, conclusions can be drawn that the DNN is successful at recognising known features in an image.

10.4 SUMMARY

All possible combinations of features where the features in the image distinctively fall under different classes were considered as test data to test the DNN for advanced image recognition capabilities.

Any combinations where there is more than one feature that belongs to the same class were not considered. For example, an image with three Class-3 features defeats the purpose of obstacle avoidance in robotics, as the overload of details of an obstacle in the same region does not facilitate acquisition of additional knowledge of the environment.

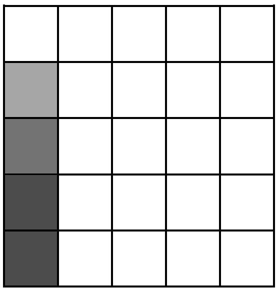


Figure 10. 3: Combinations of Features of the same Class 3

Hence 54 possible distinct combinations of features, which comprised combinations of 2 and 3 features, were included in the test set. The results obtained demonstrate that the DNN is capable of recognising combinations of known features.

The next phase of testing the DNN involved using digital images in the test data. The details of the tests conducted using digital images are given in Chapter 11.

CHAPTER 11: TEST WITH DIGITAL IMAGES

11.1 INTRODUCTION

The DNN was tested for its abilities of pattern recognition, generalisation, noise tolerance and recognition of features. The results obtained from tests conducted using various test data were presented in previous chapters.

The next phase was to test the DNN using digital images. The digital images in the test set were not included in the training data. The digital images were of two distinct categories.

1. Digital Images of well-defined objects and
2. Digital images with fuzzy objects that resemble the seabed

The choice of these types of images was to test the DNN with images of increasing levels of complexity and noise. The images also comprise a selection of combinations of features.

All images were pre-processed before being handled by the DNN. Paintshop Pro, Microsoft Photo Editor and Visual Basic were used to pre-process the digital images. The methods used for pre-processing the images and the results obtained when tested with the DNN are presented in this chapter.

11.2 TESTS FOR WELL-DEFINED DIGITAL IMAGES

The pre-processing of images involved extracting data from the image in a form that could be presented to the DNN keeping in line with standard ANN practices. The main stages of the pre-processing phase involved:

- Compressing the images to a size of 5x5 pixels through pixel averaging.
- Converting the image to an uncompressed bit map.
- Normalising the grey scale values between (0 and 255) to fall between 0 and 1.

  This is done to avoid dealing with large numbers and reduce the computational complexity of the DNN. A visual basic program was coded and used to implement normalisation of the grey scale values of the image

Each pattern is listed below with the different stages of processing involved.

- Stage 1-The properties of the image being used as test data.
- Stage 2-The reduction of the image to size 5x5 is shown in Paintshop Pro.
- Stage 3-The Hexadecimal and decimal values of the grey scale of pixels is given.
- Stage 4-The Normalised data of the pixels is given.
- Stage 5-Finally, the results from the DNN when tested with the image is illustrated graphically.

IMAGE 1

Figures 11.1 and 11.2 show the properties of Image-1 and its re-sizing to size 5x5.

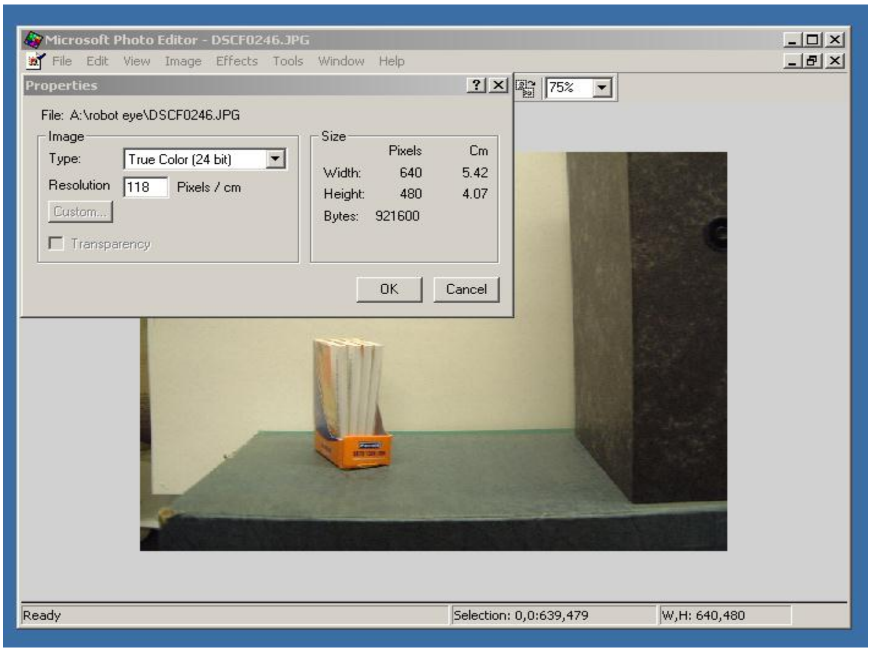


Figure 11.1: Illustration of the Properties of Image-1

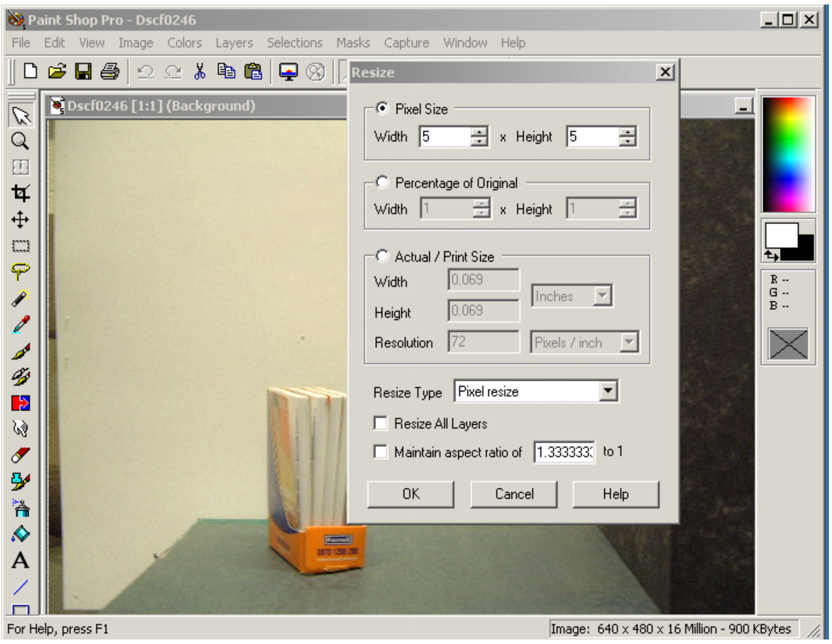


Figure 11.2: Illustration of Re-sizing Image-1

Tables 11.1, 11.2 and 11.3 give the pixel values in hexadecimal, decimal and normalised forms.

D0	D2	D0	C6	86
CE	D0	CD	BD	7F
C6	C8	C6	B6	7C
B8	B0	C0	B1	78
91	8B	9A	8C	62

Table 11.1: Grid showing the Image-1 in Hexadecimal Format

208	210	208	198	134
206	208	205	189	127
198	200	198	182	124
184	176	192	177	120
145	139	154	140	98

Table 11.2: Grid showing Image-1 in Decimal Form

0.01785713	0	0.01785713	0.1071429	0.6785715
0.03571427	0.0178571	0.04464287	0.1875	0.7410715
0.1071429	0.089285	0.1071429	0.25	0.7678571
0.2321429	0.3035714	0.1607143	0.2946429	0.8035715
0.5803571	0.6339285	0.5	0.625	1

Table 11.3: Grid showing the Normalised Values of the Greyscale of Pixels in Image-1

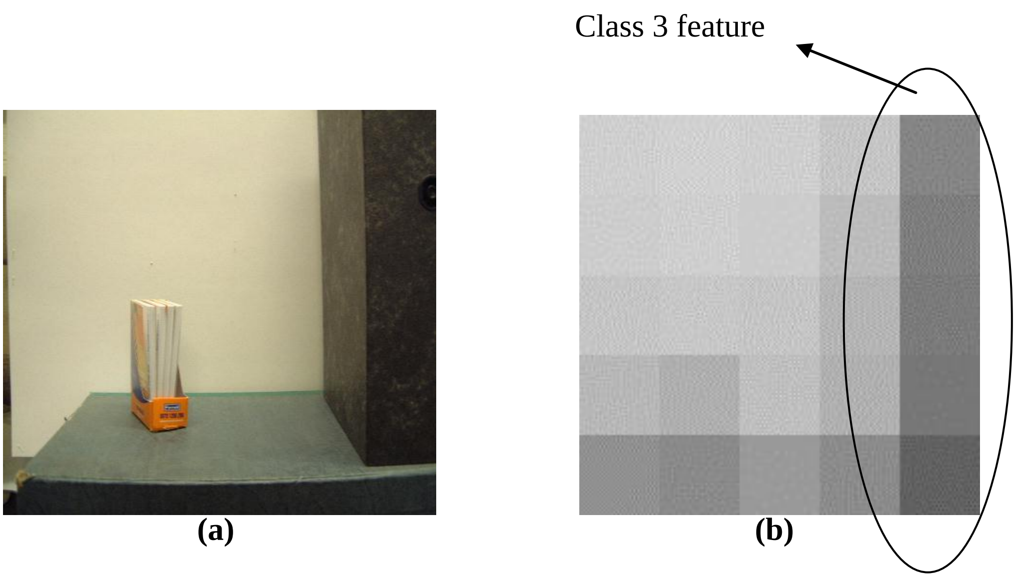
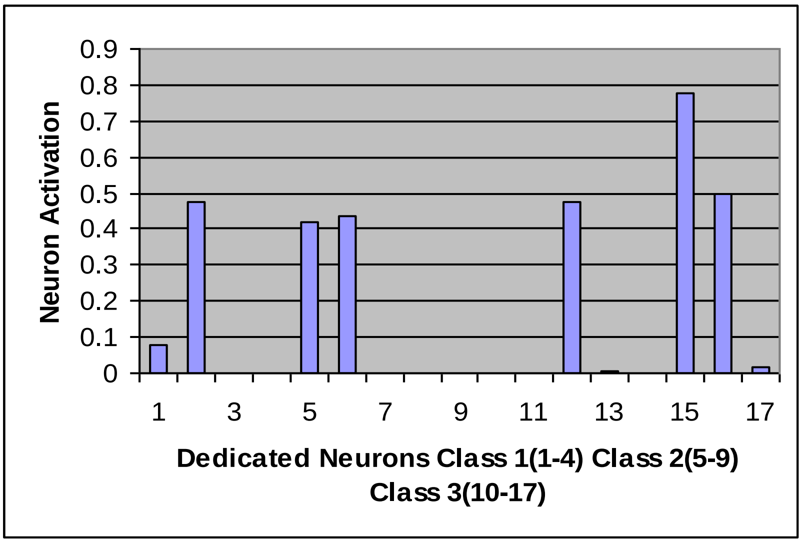


Figure 11.3: Image-1 before (a) and after (b) pre-processing

Figure 11.3 shows the image before and after pre-processing. Similarities between the digital Image-1 and features in the training set of the DNN can be observed, although the processed digital image is very fuzzy. Also, the vertical feature appears in an unfamiliar spatial position in the 5x5 grid, as it appears on the left wall of the 5x5 grid in the training set.



Graph 11.1: Testing DNN with Image-1. Illustration of the Activation of Dedicated Neurons

The Graph 11.1 shows a very high activation among the Class-3 neurons due to the presence of a vertical feature along the right hand side in the image, which has been correctly recognised. The medium activation of a dedicated Class-1 neuron is because the vertical feature extends into the region of Class-1 features. The horizontal feature at the bottom excites the Class-2 neurons.

The level of neuron activation suggests the importance of the features in the image.

IMAGE 2

Other images were selected, processed and tested with the DNN. Figure 11.4 shows the image before and after pre-processing.

As seen in the Figure 11.4 (b), the processed image consists of two prominent features. The expected outcome when tested with the DNN would be high activation among the Class 1 and Class 3 features.

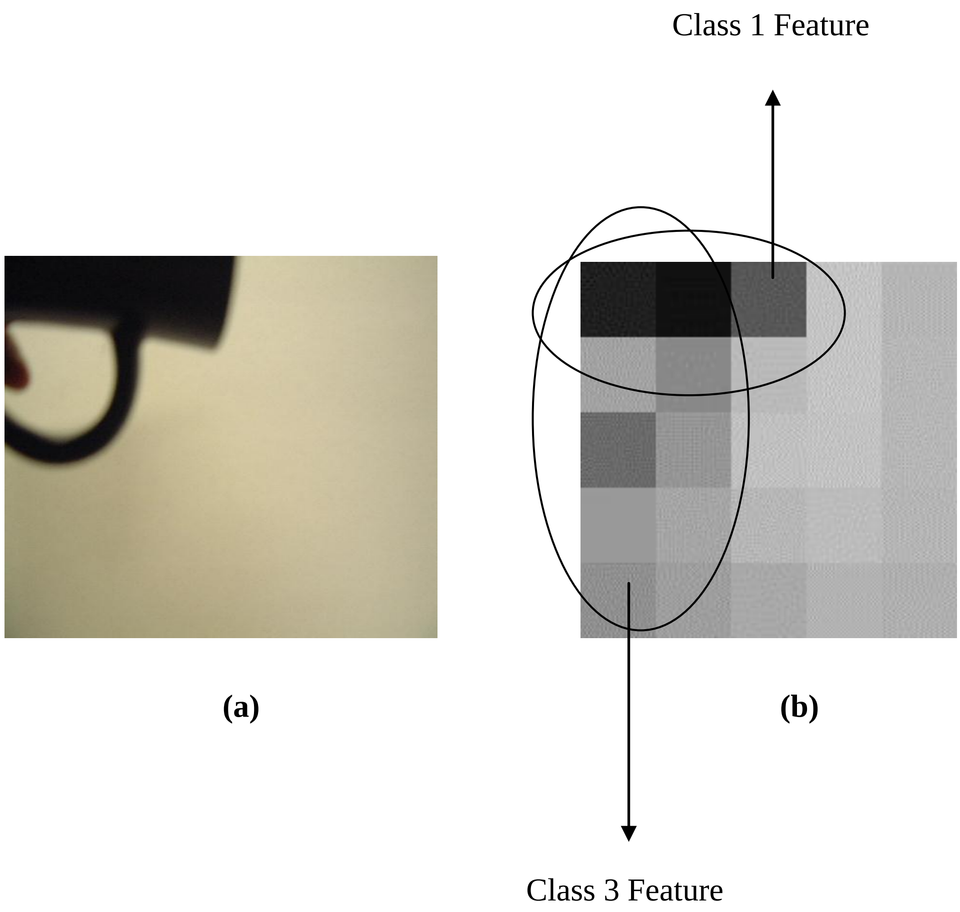
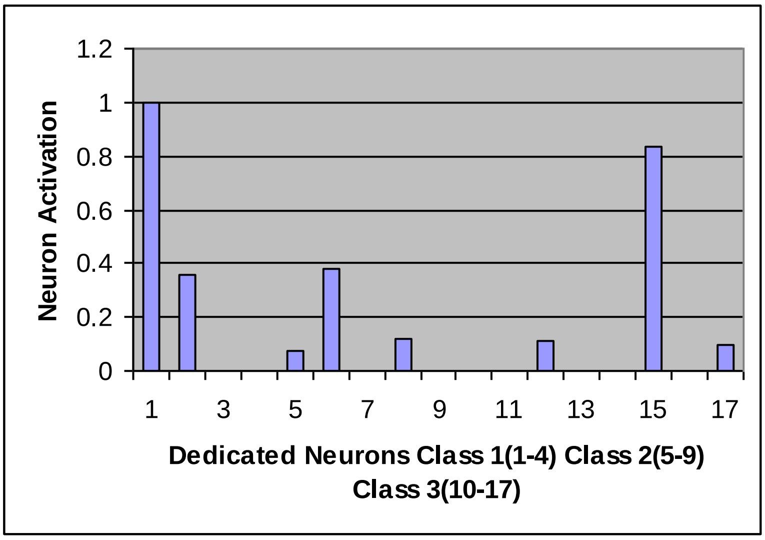


Figure 11.4: Image-2 before (a) and after (b) pre-processing



Graph 11.2: Testing DNN with Image-2. Illustration of the Activation of Dedicated Neurons

Graph 11.2 shows the activation of the dedicated neurons. The highest activation is observed among Class 1 and Class 3 features. This is in accordance with the image, which features an overhang with an extension down along the left wall. The DNN was successful in recognising the main features in the image.

In case of an obstacle avoidance system, the features with the highest corresponding dedicated neuron activation can be regarded as the most important obstacles in the path; the class with the highest activation being given the greatest importance followed by others with lower neuron activation levels.

IMAGE 3

Image 3 consists of a prominent feature along the left wall as illustrated in Figure 11.5. The figure also shows the image after the pre-processing stage.

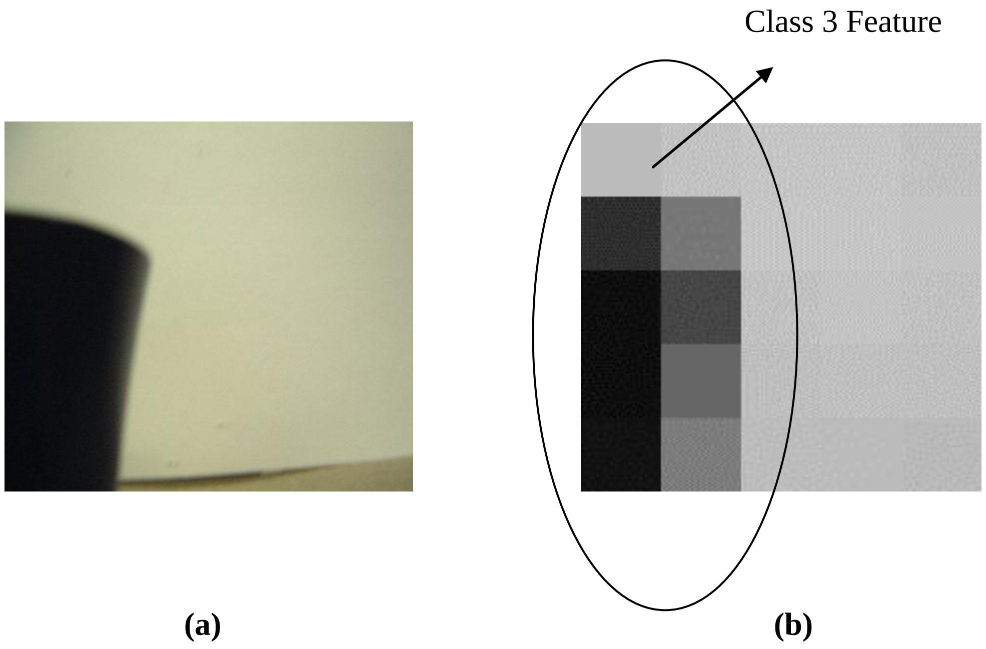
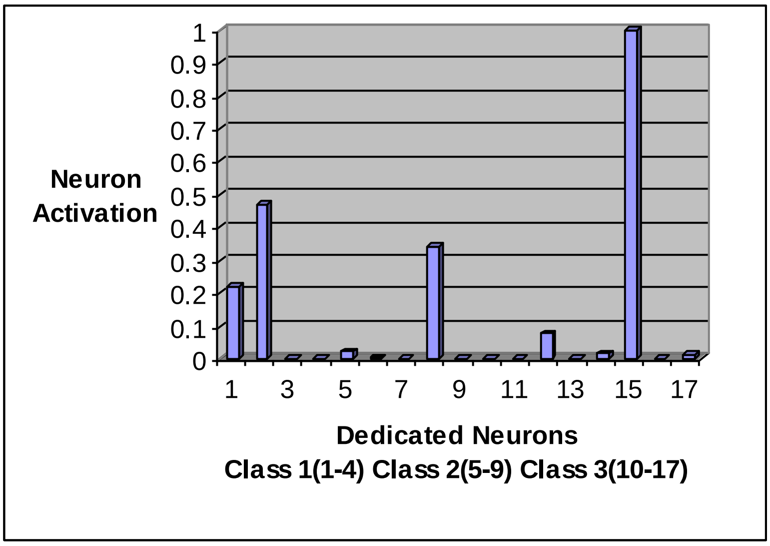


Figure 11.5: Image-3 before (a) and after (b) pre-processing



Graph 11.3: Testing DNN with Image-3. Illustration of the Activation of Dedicated Neurons

Graph 11.3 shows a very high activation among Class-3 neurons, which are dedicated to recognise vertical features. Low-level neuron activity is observed among Class-1 and Class-2 neurons. This might be due to the presence of a horizontal feature along the bottom of the grid.

IMAGE 4

The Image 4 in Figure 11.6 shows prevailing features of type Class 1, 2 and 3 as shown.

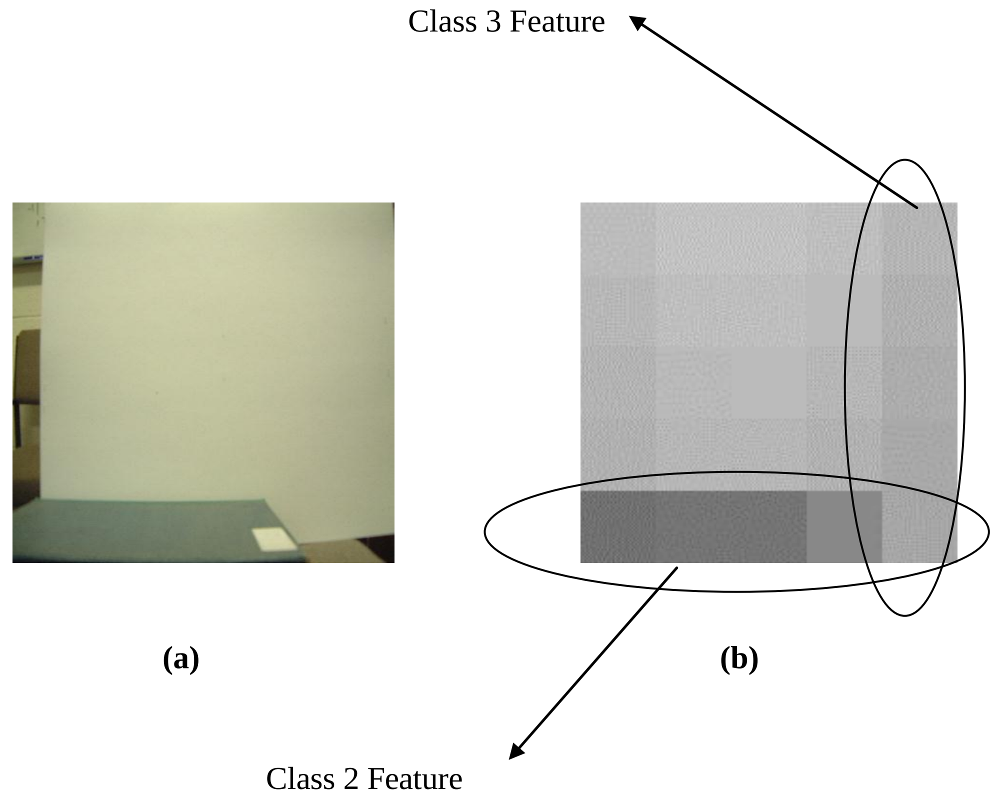
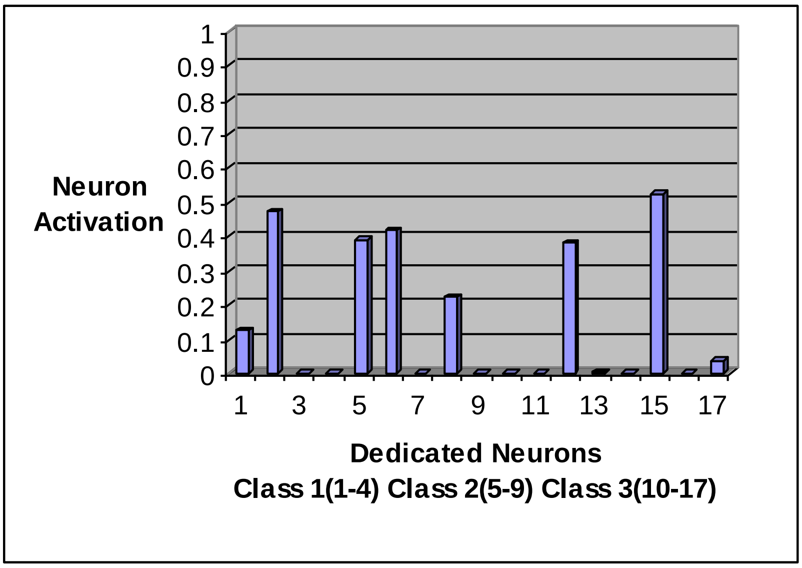


Figure 11.6: Image-4 before (a) and after (b) pre-processing



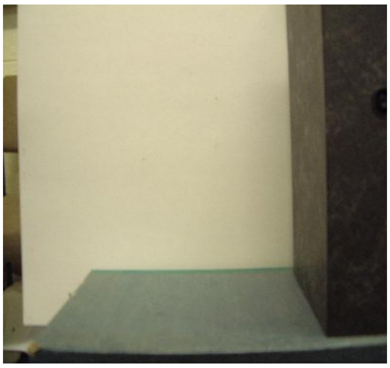
Graph 11.4: Testing DNN with Image-4. Illustration of the Activation of Dedicated Neurons

Graph 11.4 shows the activation of neurons that belong to all three classes, with the Class 3 neurons showing the highest activation.

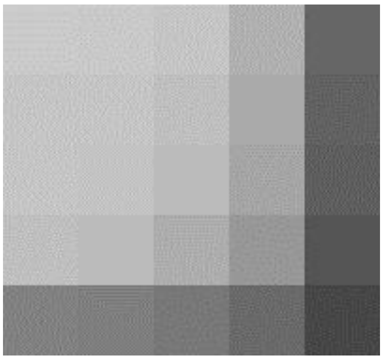
The activation of the Class 1 neurons is due to the horizontal feature at the bottom of the image. The activation of the Class 2 neurons is due to the presence of the feature at the bottom right, although the intensity of the feature fades towards the corner; hence the weak activation.

IMAGE 5

Figure 11.7 shows Image 5 before and after pre-processing. The results obtained when the image was used to test the DNN's recognition capabilities are illustrated in the Graph 11.5.

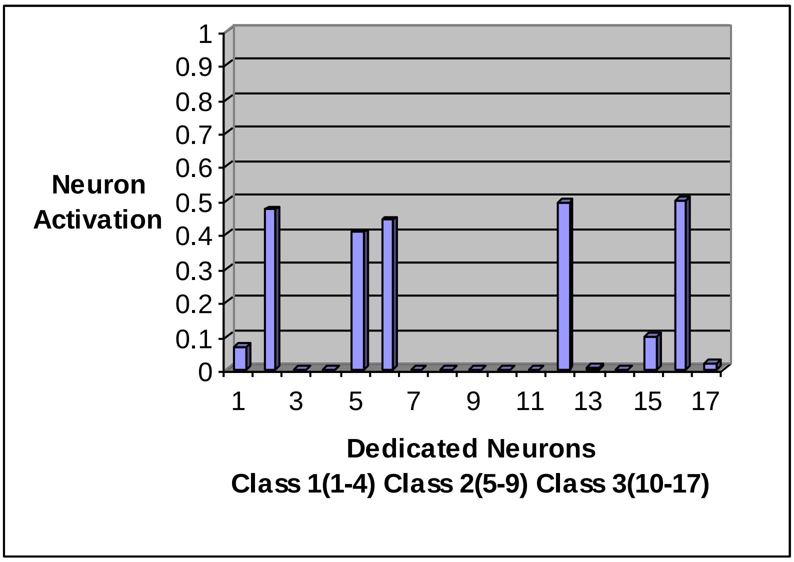


(a)



(b)

Figure 11.7: Image-4 before (a) and after (b) pre-processing



Graph 11.5: Testing DNN with Image-5. Illustration of the Activation of Dedicated Neurons

The activation of neurons in the Graph 11.5 shows the presence of three classes of features in the image due to the presence of a vertical feature along the right wall of the grid and horizontal feature at the bottom of the grid.

The Class-3 neurons are activated although the vertical object is not along the left wall (as present in the training set) but the right wall of the grid. The DNN seems to be capable of generalising and recognising the type of feature despite changes in the spatial location of the feature.

11.3 TEST FOR SEA BED IMAGES

The third phase involved further testing the DNN for its generalisation abilities, its noise tolerance and its overall robustness. To carry out these tests, simple images that represented the seabed were used. The features in the images were less defined and noisier when compared to the real images used in Section 11.2.

The results obtained showed that the DNN was capable of recognising the features in the image, despite none of the images being included in the training set. No other ANN is capable of such analysis of an image.

SEA BED IMAGE 6

Figure 11.8 shows the properties of seabed image 6 that is used to test the DNN. Figure 11.9 illustrates the image before and after the pre-processing stage.

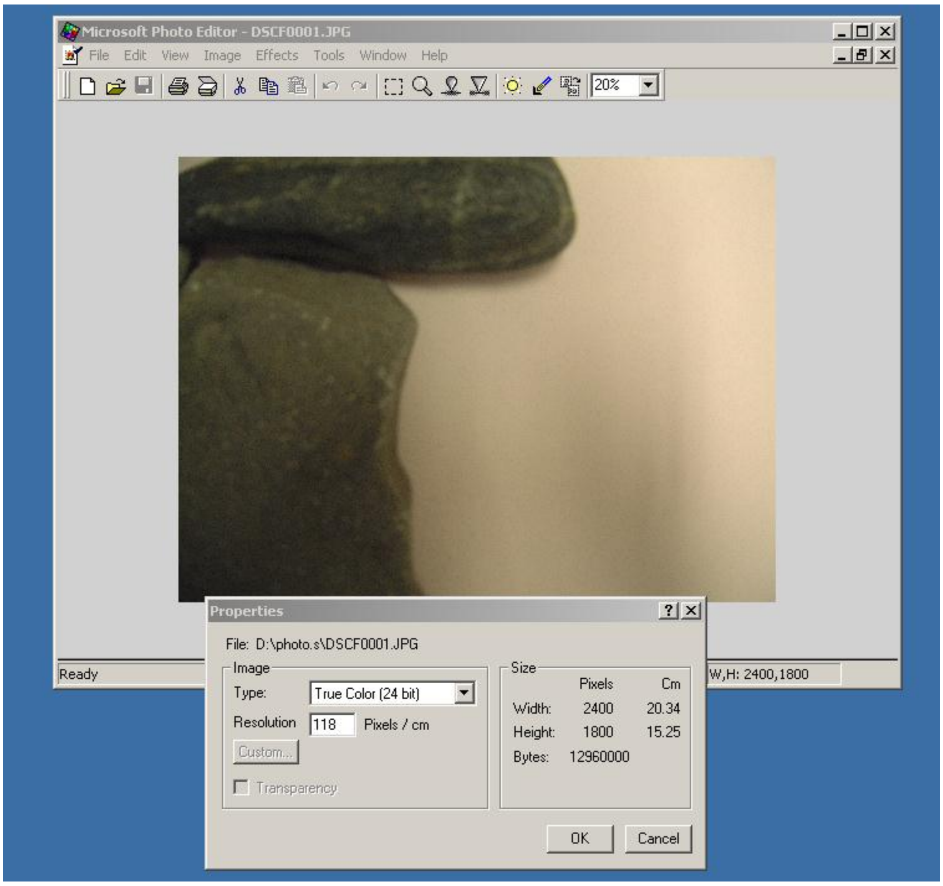


Figure 11.8: Illustration of the Properties of the Image-6

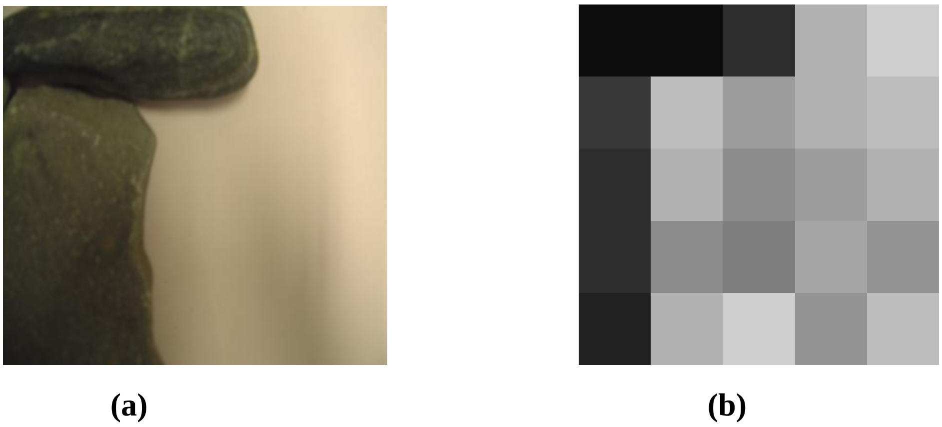
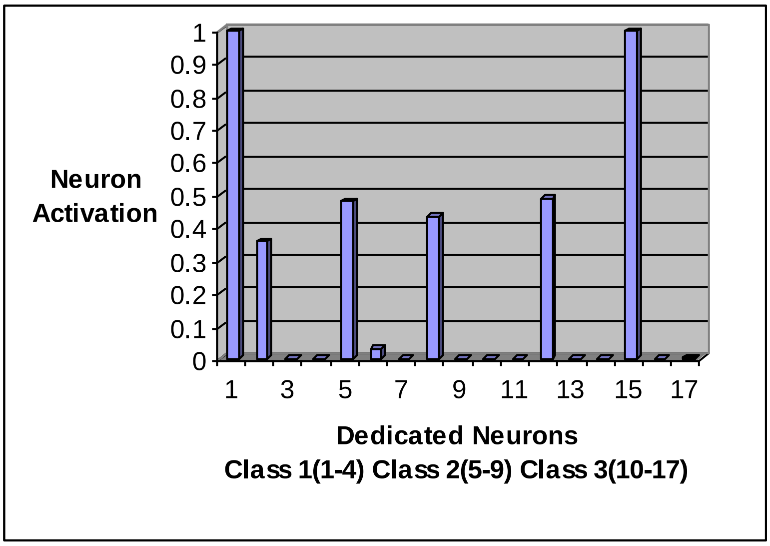


Figure 11.9: Illustration of the Image-6 before (a) and after (b) pre-processing

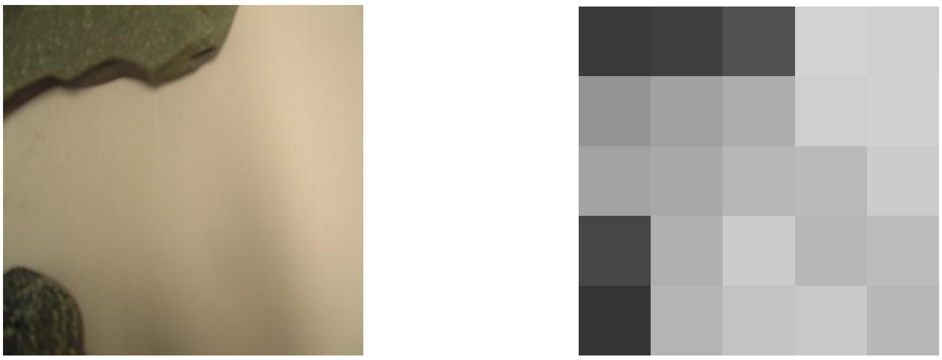


Graph 11.6: Activation of Output Neurons when tested with Image-6

The results obtained as illustrated in Graph 11.6 show a very high activation of neurons allocated for Class 1 and Class 3 features. The weak activation of class 2 neurons is due to the shadow in the bottom right region of the Class 2 features.

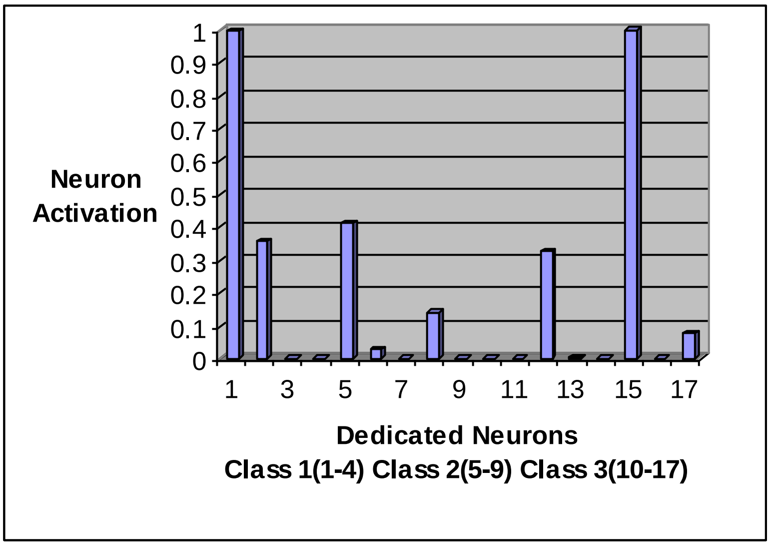
SEA BED IMAGE 7

Figure 11.10 illustrates Image 7 before and after pre-processing.



(a) (b)

Figure 11.10: Illustration of the Image-7 before (a) and after (b) pre-processing

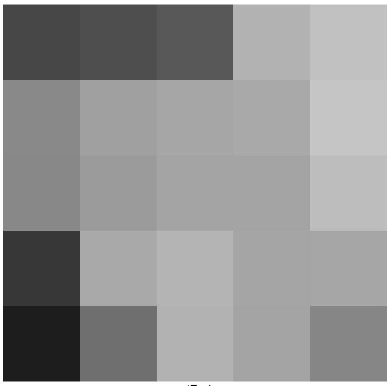


Graph 11.7: Activation of Output Neurons when tested with Image-7

Graph 11.7 illustrates the activation of the dedicated output neurons for the three classes of features. The highest activation is observed among the Class 1 and Class 3 features. These features are clearly identifiable in the pre-processed image.

SEA BED IMAGE 8

(a)

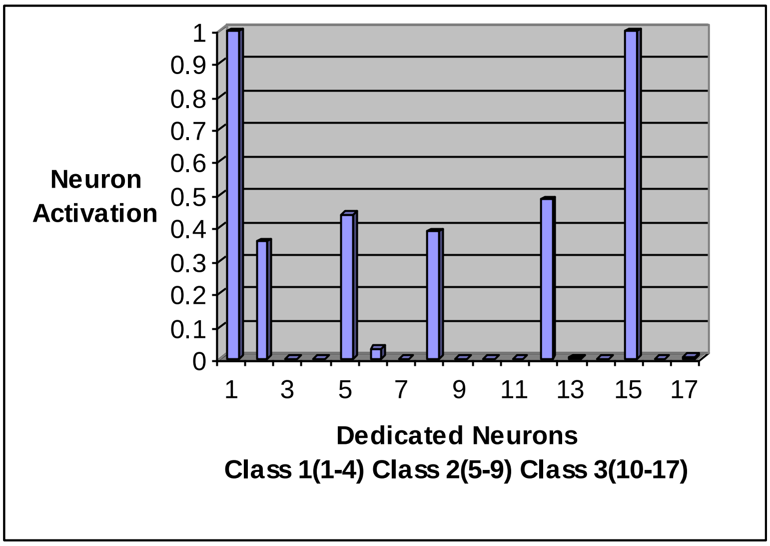


(b)

Figure 11.11: Illustration of the Image-8 before (a) and after (b) pre-processing

Figure 11.11 illustrates Image 8, which was used to test the DNN. The features in the image follow the theme of Class 1 and Class 3 features. However it can be noticed that the variation among these features can be quite broad.

The results as depicted in Graph 11.8 show that the activation of various classes of neurons are quite consistent despite the differences in various types of Class 1 and Class 3 features in the images.

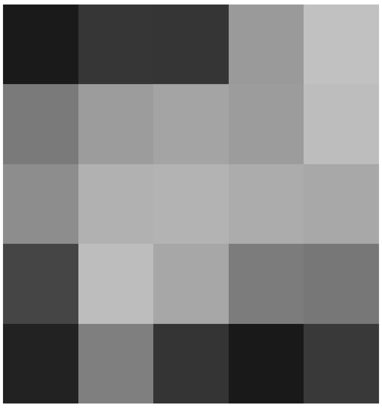


Graph 11.8: Activation of Output Neurons when tested with Image-8

Graph 11.8 shows very high activation of Class 1 and Class 3 features. Hence, the DNN has successfully recognised the main features in the image.

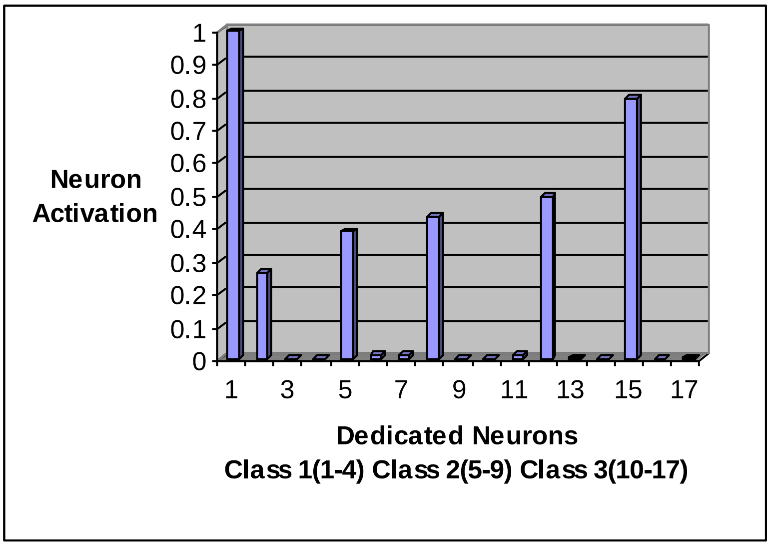
SEA BED IMAGE 9

Figure 11.8 shows an image with three features that belong to Class 1, Class 2 and Class 3 respectively. The figure shows the pre-processed image, where all pixels are normalised to a value that lies between 0 and 1.



(a) **(b)**

Figure 11.12: Illustration of the Image-9 before (a) and after (b) pre-processing



Graph 11.9: Activation of Output Neurons when tested with Image-9

The DNN was tested with Image 9 and the results obtained are presented in the Graph 11.9 as shown.

The Graph 11.9 clearly shows a very high activation of Class 1and Class 3 neurons. Class 2 neurons show an activation ≥ 0.4. This is due to the feature in the bottom right region of the image.

From the other graphs showing results of tests conducted using Images 6, 7 and 8, it can be gathered that the Class 2 neurons show some activation when tested with these images due to the presence of shadows. Throughout all the tests carried out during this research project Class 2 neurons showed weaker activation compared to Class 1 and Class 3 neurons.

All seabed images include a feature of type Class-1. The feature appears in the images with varying degrees of noise. The images also have a Class-3 type feature on the left wall, which appear in different shapes and sizes in the image.

Graphs 11.6, 11.7 and 11.8 clearly identify Class-1 and Class-3 features. Class-2 features have a very small activation, which is due to the shadows in the image in the zone of Class-2 features.

11.4 SUMMARY

The DNN was tested using digital images with varying degrees of combination of fuzzy features. The DNN was tested successfully with all the images presented. The images, which were not part of the training set, introduced a greater level of complexity to the test data.

The tests conducted on the DNN using Digital Images affirm the unique capabilities of the network that were revealed in previous chapters.

The structure of the DNN that manages and promotes feature recognition in an image is evaluated in Chapter 12.

CHAPTER 12: ANALYSIS OF THE ARCHITECTURE OF DNN

12.1 INTRODUCTION

The unique DNN developed and implemented in this research programme was tested for its Pattern Recognition abilities. Systematic test data was devised and used to evaluate the novel characteristics of the DNN. The results presented in Chapters 9, 10 and 11 shows the remarkable capabilities of the DNN.

One of the noteworthy qualities of the DNN was its capacity to recognise features in an image. Also, the DNN was trained for 9 features, but could recognise 90 images, each a large variance from the training data. Although, images in the test data were ambiguous and purposely misleading in many cases, the DNN was successful at recognising the features in the image.

During the various stages of the development of the structure and training method of the DNN, effort was made to incorporate features such as modularity, groups of dedicated neurons, lateral inhibition and interaction between dedicated neurons in the DNN. These novel aspects of the DNN were included in the design to overcome some of the problems in ANNs.

Results obtained from an analysis of the significance of some of the design features considered during the development of the architecture of the DNN are presented in this chapter.

Two main aspects about the structure of DNN that were tested in this research programme are:

1. The significance of the specific type of connections, namely inhibitory and excitatory
2. The significance of the order of information flow between various groups of dedicated neurons

12.2 SIGNIFICANCE OF TYPE OF NEURAL CONNECTIONS

The study of biological vision systems conducted in this research programme has been very helpful in designing the DNN.

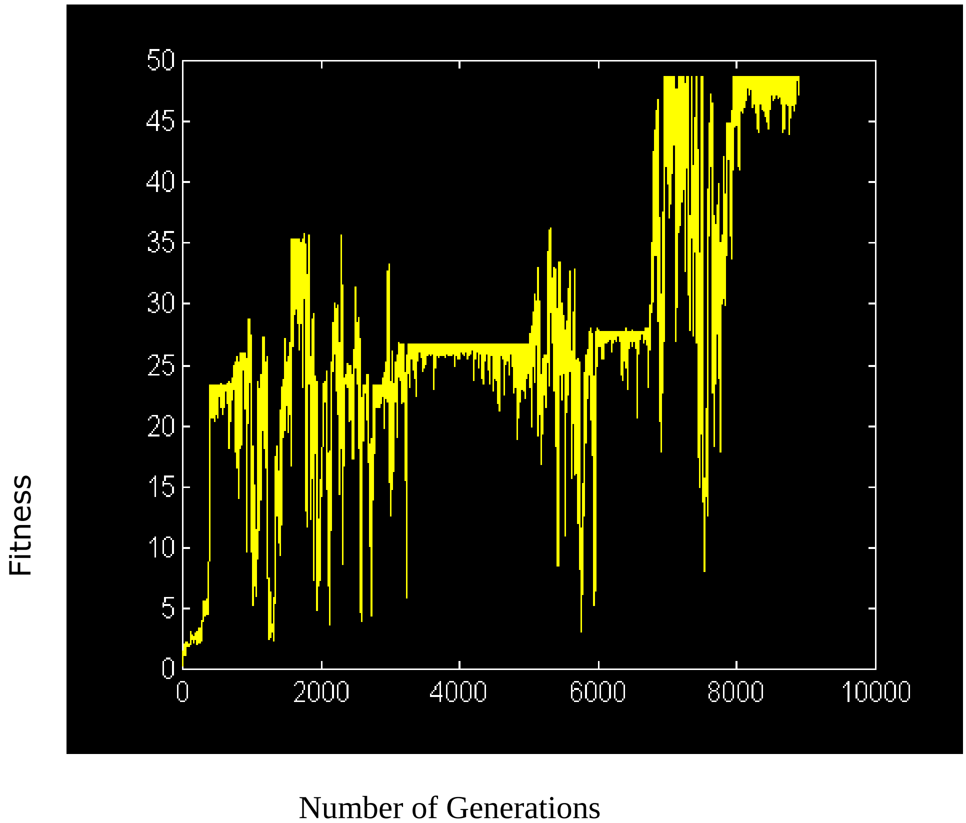
One of the main features of many biological vision systems (insect and toad) is the presence of inhibitory connections between various neurons. The elimination of such inhibitory connections had an immediate detrimental effect on the recognition process. The prominence of these connections in biological systems (as mentioned in chapters 4, 5 and 6) was the reason behind inclusion of inhibitory connections in the DNN.

The relevance of inhibitory connections in systems that employ traditional ANNs and various learning algorithms such as Back Propagation and Evolutionary Algorithms are not completely known. One of the contributions of this research to the state of art is the investigation conducted to analyse the significance of the type of connections in the DNN.

One of the straightforward ways in which the analysis on the type of connections could be conducted was to make all connections in the DNN positive or excitatory. Where all connections are similar, the DNN can be called a homogenous network.

The DNN (with all connections of an excitatory type) was trained using FUSION. All other aspects of the training were unaltered. For example, the distribution of the dedicated neurons for various features and the reward mechanism were unchanged.

The DNN reached a maximum fitness of 48.667 in 8897 generations. The Graph 12.1 shows the fitness profile of the DNN. A discussion regarding the fitness is given in the Section 12.4.



Graph 12.1: Illustration of Fitness achieved by a Homogenous DNN

The homogenous DNN was trained and tested using features from the training and test data sets used for the DNN originally, which are given in Chapter 9.

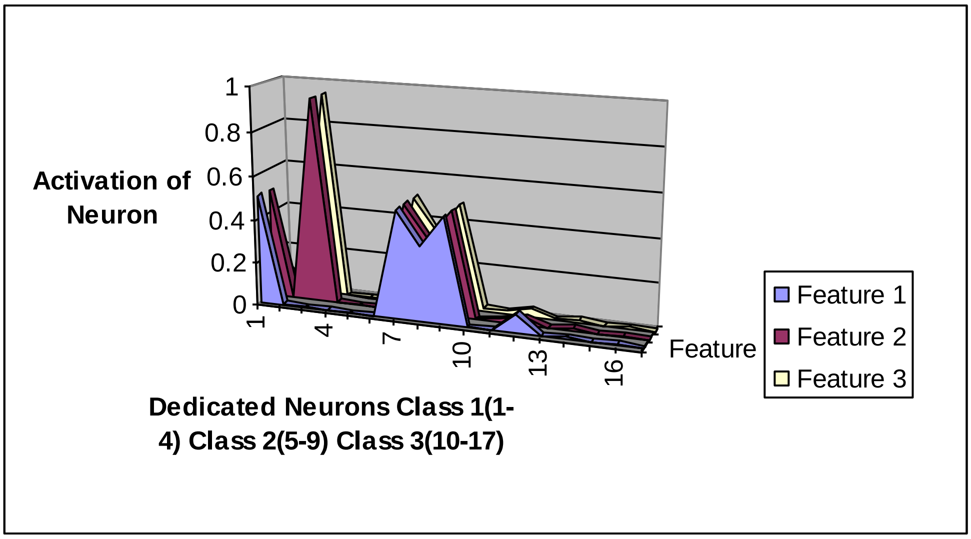
12.2.1 TESTING THE HOMOGENOUS DNN WITH THE TRAINING SET

The results presented in Graph 12.2 – Graph 12.4 illustrate the activation of various dedicated neurons when tested for the 3 features in 3 classes of features used in the training set; Class-1, Class-2 and Class-3 respectively.

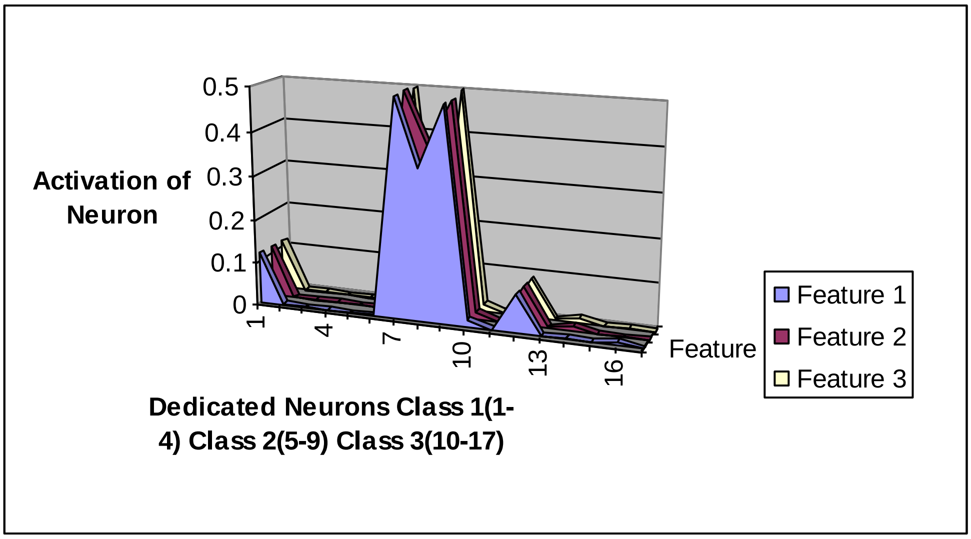
The results obtained for the set of features that belong to a certain class are presented together in one graph. Instead of using Bar charts, a continuous surface chart was used to alleviate problems with visualisation of the results. In case of the bar graphs a minimum of 51 bars will have to be presented, which appear very cluttered and confusing. Hence, whenever groups of data were presented in one graph, the continuous method was preferred.

In Graph 12.2 it can be observed that the activation values of the Class 1 dedicated neurons for test with Feature 1 is not significantly higher than the other neurons.

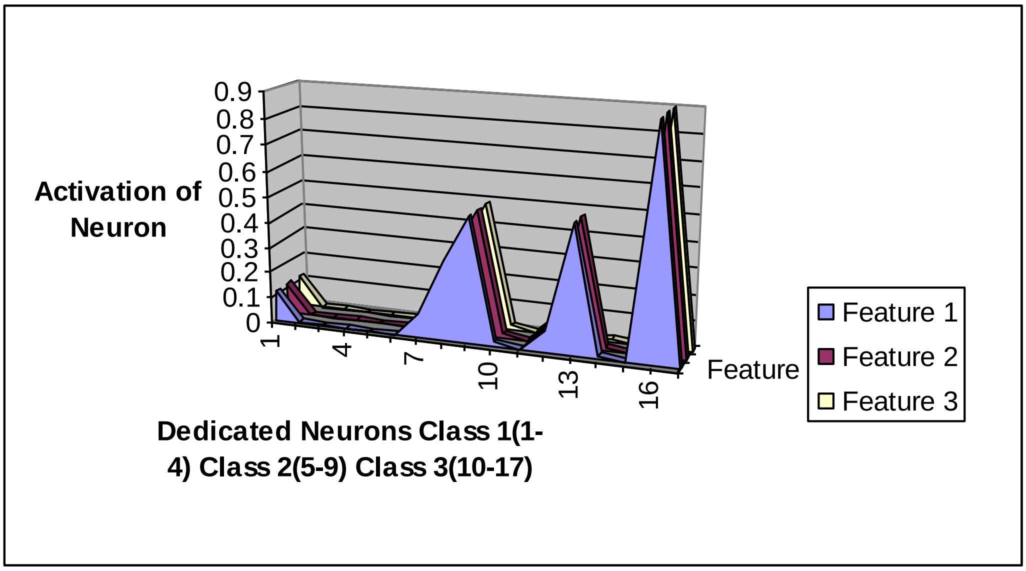
In all other cases, the results show that the DNN has successfully trained to recognise the 9 features classified under various classes. The activation value of neurons in a dedicated class was significantly higher than the other dedicated neurons when tested for that particular class of features respectively.



Graph 12.2: Homogenous DNN - Classification of Three Class-1 Features from the Training Set



Graph 12.3: Homogenous DNN - Classification of Three Class-2 Features from the Training Set



Graph 12.4: Homogenous DNN - Classification of Three Class-3 Features from the Training Set

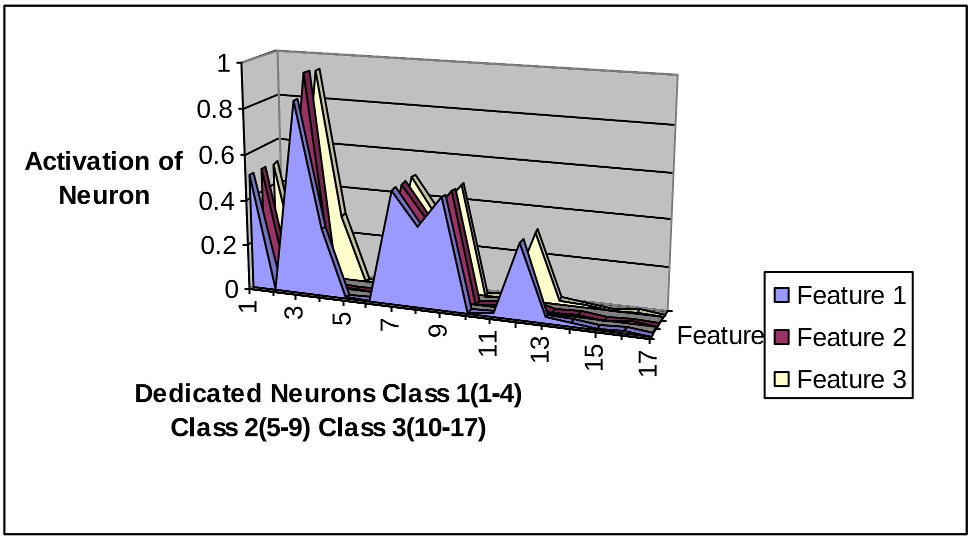
Hence, the initial results suggest that the type of connection may not be very significant. The learning method FUSION is powerful enough to override existing type of connections. In order to confirm the results, the homogenous DNN was tested using New and Noisy features given in Chapter 9.

12.2.2 TESTING THE DNN WITH NEW FEATURES FROM THE TEST SET

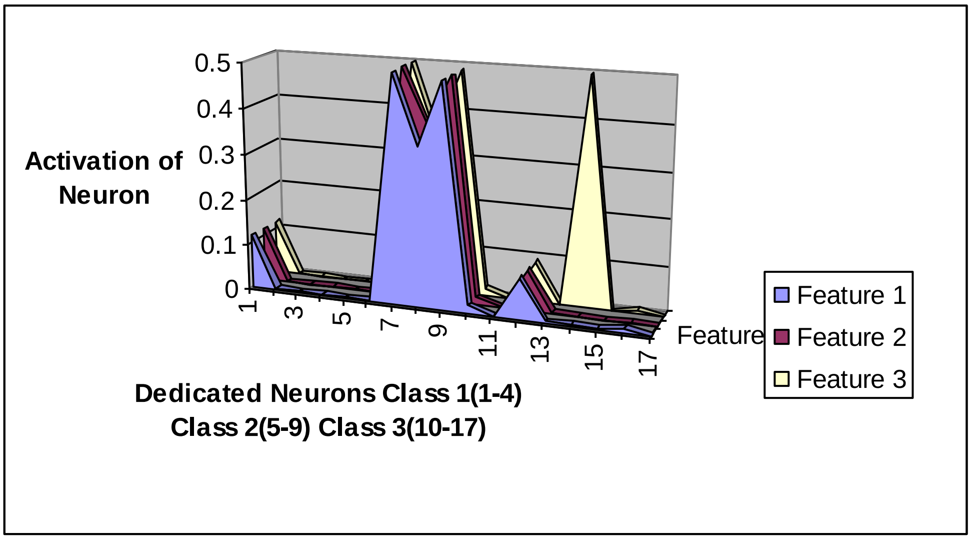
The homogenous DNN was successful at recognising various features from the test set. The ability for Noise Tolerance and Generalisation are essential features for a network to be employed in various applications.

The Homogenous DNN's Generalising abilities are tested using New Features that were presented in Chapter 9.

Graphs 12.5-12.7 illustrate the activation of various dedicated neurons when tested with Class 1, Class 2 and Class 3 features respectively.

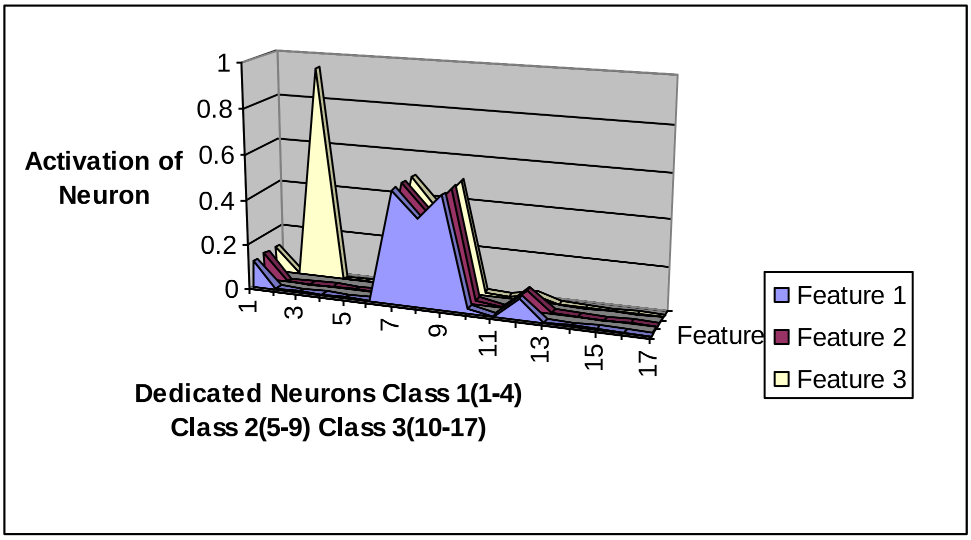


Graph 12.5: Homogenous DNN - Classification of Three New Class-1 Features from the Test Set



Graph 12.6: Homogenous DNN - Classification of Three New Class-2 Features from the Test Set

The Graphs 12.5 and 12.6 show that the Homogenous DNN classified both Class 1 and Class 2 features successfully as shown by the highly activated dedicated neurons of the corresponding class.



Graph 12.7: Homogenous DNN - Classification of Three New Class-3 Features from the Test Set

Graph 12.7 shows that the homogenous DNN did not recognise any of the Class 3 features. The activation values of all three Class 3 features were < 0.2.

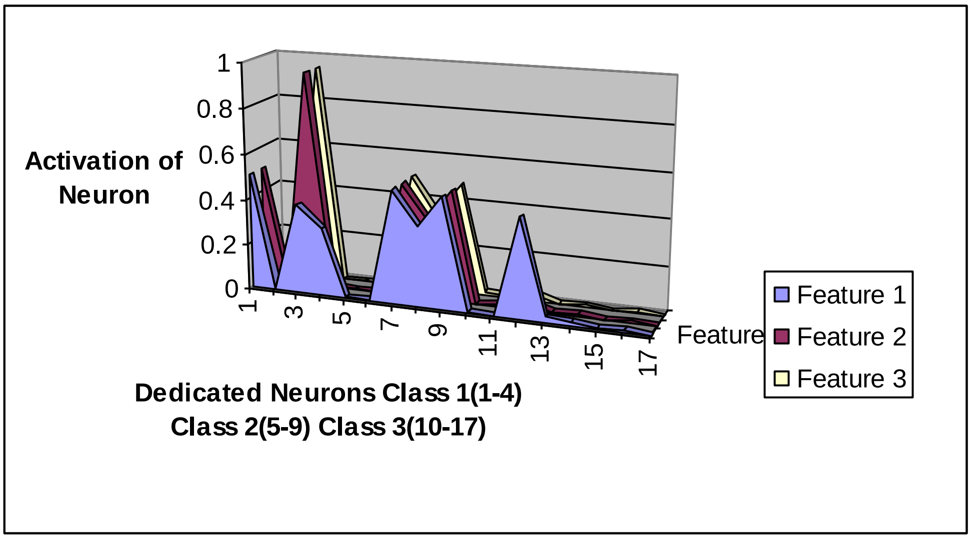
Hence, the homogenous DNN was not as successful as the original DNN in recognising new features.

12.2.3 TESTING THE DNN WITH NOISY FEATURES FROM THE TEST SET

The homogenous DNN was further examined to check its ability to recognise noisy images. The noisy features were speckled as given in Chapter 9. For ease of comparison, noise was introduced in 2 stages. In the first stage 9 noisy features were used – 3 in each class.

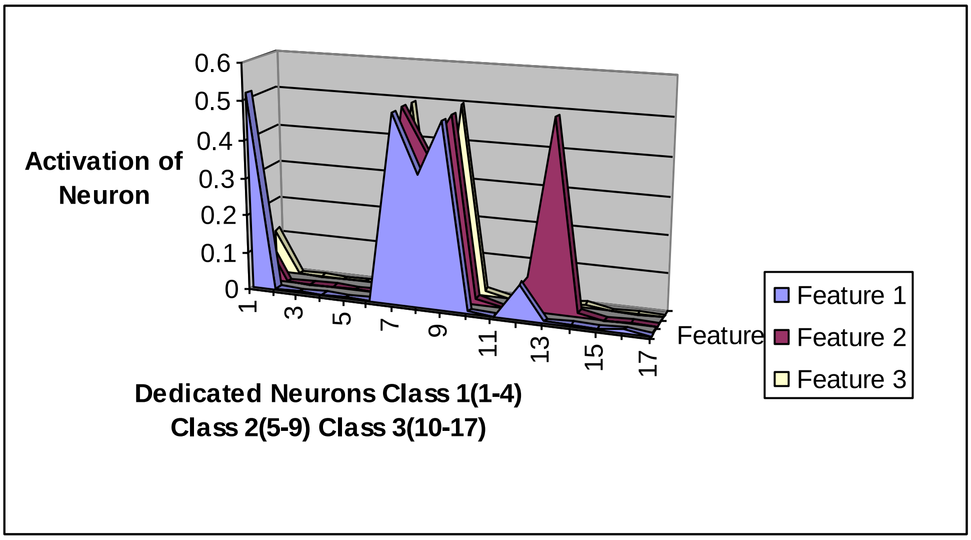
Graphs 12.8-12.10 illustrate the results obtained when the homogenous DNN was tested with 1st level of noisy images.

Graph 12.8 shows that the homogenous DNN clearly recognises Class 1 features.



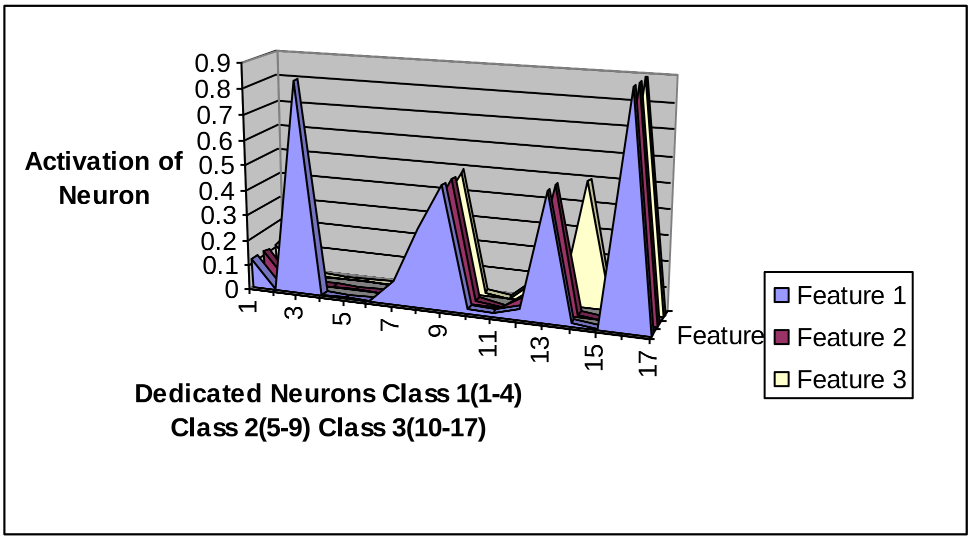
Graph 12.8: Homogenous DNN - Classification of Three Noisy Class-1 Features from the Test Set

From the results presented in Graph 12.9, it can be concluded that the homogenous DNN recognised one (Feature 3) Class 2 feature due to high activation of the corresponding dedicated neurons. The dedicated neurons of Class 1 showed the highest activation when tested with Feature 1 of Class 1. Also Class 3 neurons had the highest activation when tested with Feature 2. Hence, the Homogeneous DNN recognised only two of the three features as shown in Graph 12.9.



Graph 12.9: Homogenous DNN - Classification of Three Noisy Class-2 Features from the Test Set

The network also successfully recognised all Class 3 features as evident in Graph 12.10. This is an unexpected result, as the Class 3 features from the original test data were not recognised as shown in Graph 12.7.



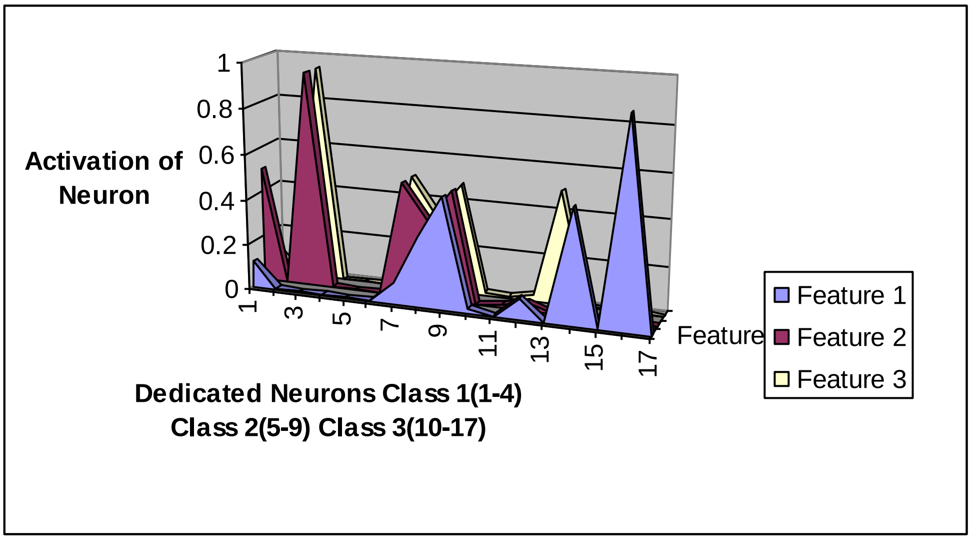
Graph 12.10: Homogenous DNN - Classification of Three Noisy Class-3 Features from the Test Set

As the homogenous DNN recognised 7 out of 9 features, the success rate is 77.78%.

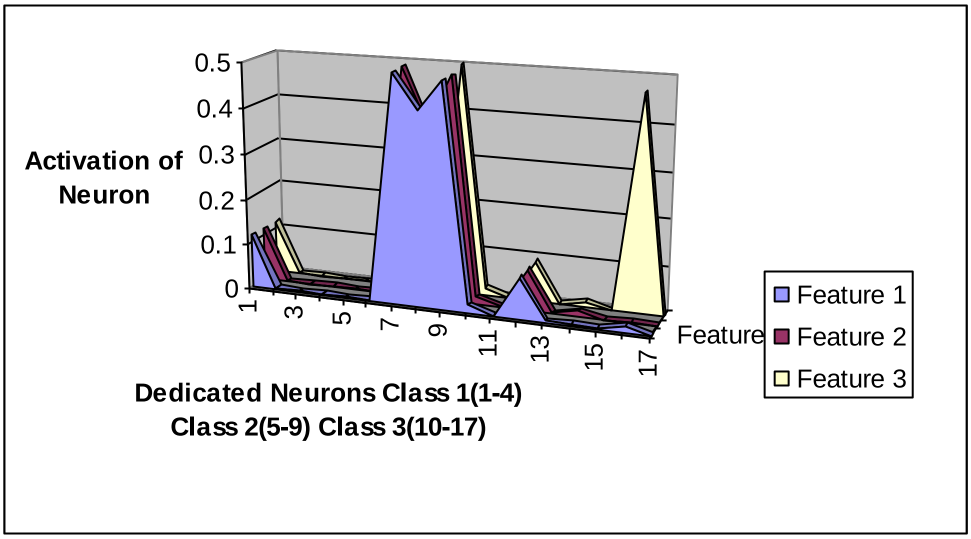
The next stage involved testing the homogenous DNN with increased levels of noisy images.

12.2.4 TESTING THE HOMOGENOUS DNN WITH FEATURES FROM THE TEST SET WITH INCREASED NOISE

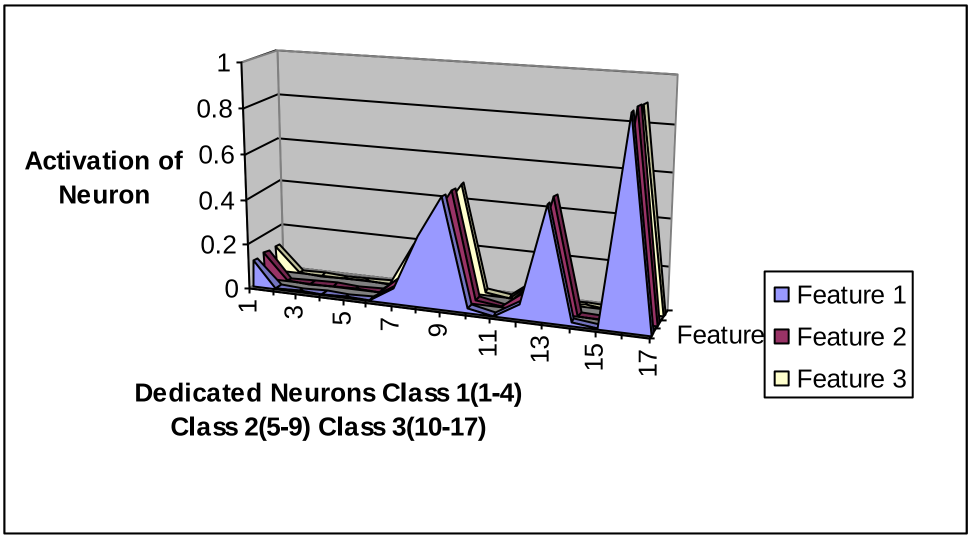
The images with increased levels of noise are given in Chapter 9. Graphs 12.11 –12.13 present the results obtained when the 9 noisy images with increased levels of noise were used to test the network.



Graph 12.11: Homogenous DNN - Classification of Three Class-1 Features with Increased Noise from the Test Set



Graph 12.12: Homogenous DNN - Classification of Three Class-2 Features with Increased Noise from the Test Set



Graph 12.13: Homogenous DNN - Classification of Three Class-3 Features with Increased Noise from the Test Set

Graph 12.11 – Graph 12.13 illustrate the results obtained when the homogenous DNN was tested with Noisy (2nd stage) features that belong to Class 1, Class 2 and Class 3 features respectively. The results show that in majority of the cases the DNN was successful at recognising the features.

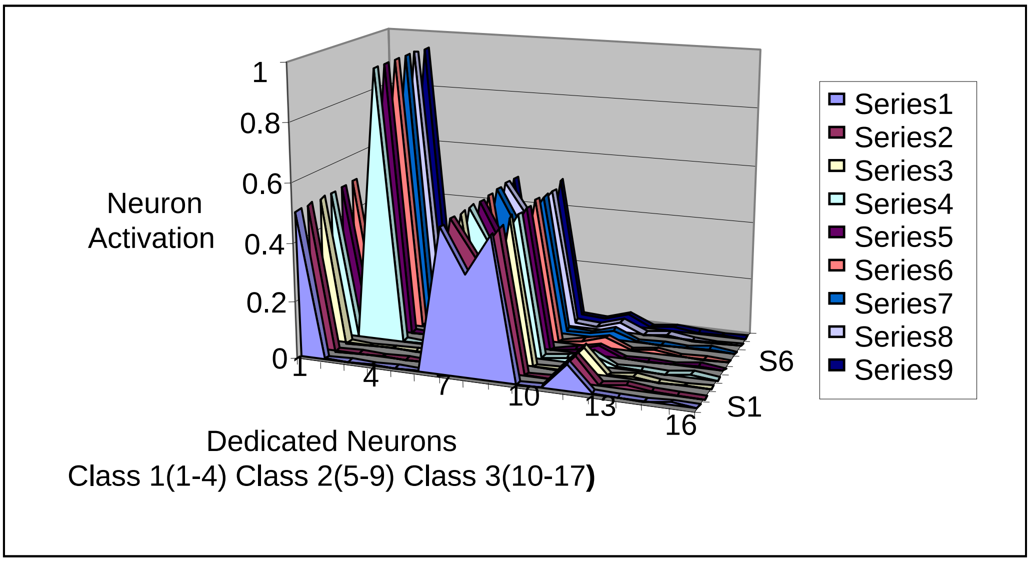
There are some cases (Feature 1 in Class 1 and Feature 3 in Class 2) where the Homogenous DNN has failed when tested with features of increased noise.

The results also suggest that the homogenous DNN may not be very tolerant to noise. In many cases the activation of dedicated neurons does not reach very high values and usually reach around 0.5. In such cases, if the activation of other dedicated neurons is significantly less, then it can be assumed that the DNN was clear with its classification. However, this is not the case with some of the failures with the homogenous DNN.

The DNN was tested with combinations of features to verify the results.

12.2.5 TEST WITH COMBINATION OF TWO FEATURES

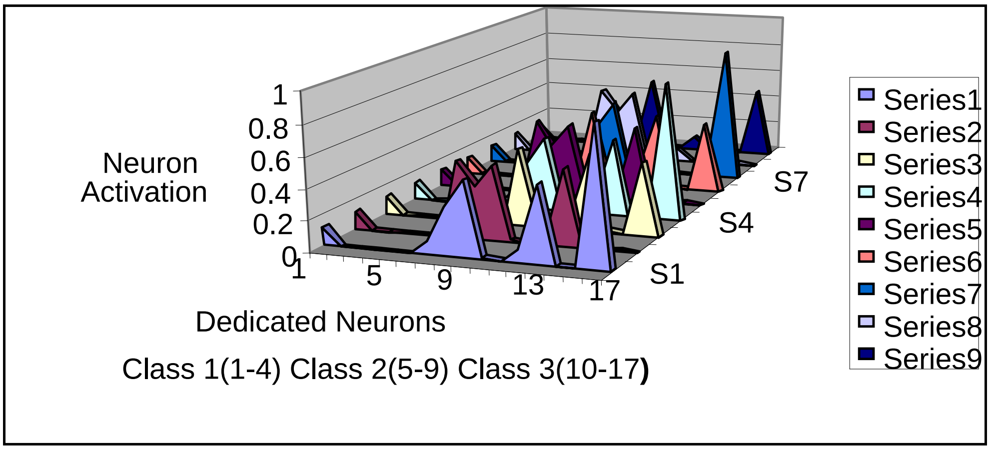
Graph 12.14 – 12.16 present results obtained when the Homogenous DNN was tested with combinations of two features.



Graph 12.14: Homogenous DNN - Test with 9 possible Combinations of Class-1 and Class-2 Features

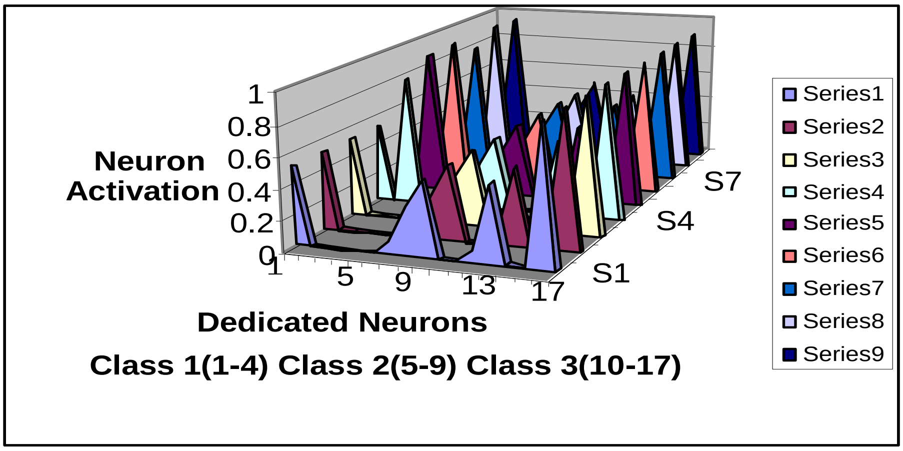
Graph 12.14 illustrates the results obtained when tested with combinations of Class 1 and Class 2 features. All the 9 combinations were successfully recognised as the corresponding dedicated neurons had an activation ≥ 0.4. The activation of Class 3 neurons in each case was < 0.4.

Graph 12.15 reveals the results obtained when tested with patterns that were combinations of Class 2 and Class 3 features. Most of the dedicated neurons show activations ≤ 0.4. In cases where the activation is ≥ 0.4, the activation is not high in both the features being tested in a pattern. Hence 100% recognition was not achieved as per the recognition criteria mentioned in Section 10.2.



Graph 12.15: Homogenous DNN - Test with 9 possible Combinations of Class-2 and Class-3 Features

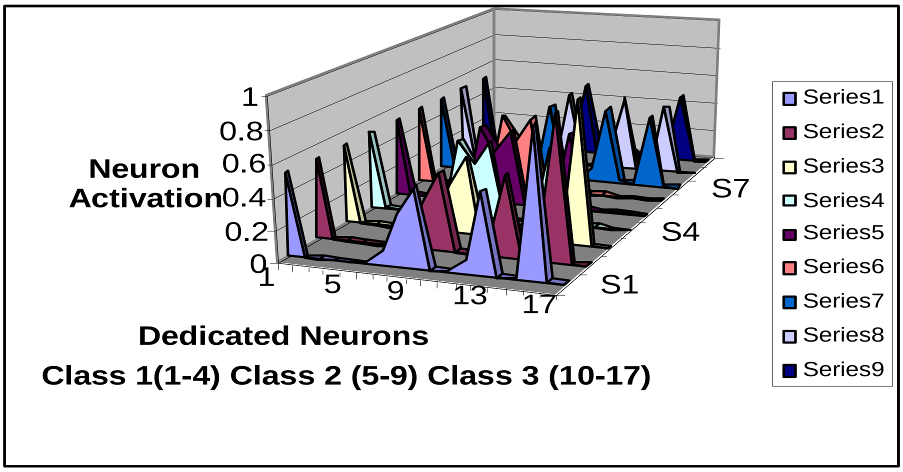
Graph 12.16 presents the results obtained when Class 3 and Class 1 features were tested with the Homogenous DNN. All 9 combinations were successfully recognised as the activation value of the corresponding dedicated neurons have an activation ≥ 0.4.



Graph 12.16: Homogenous DNN - Test with 9 possible Combinations of Class-1 and Class-3 Features

12.2.6 TEST WITH COMBINATION OF THREE FEATURES

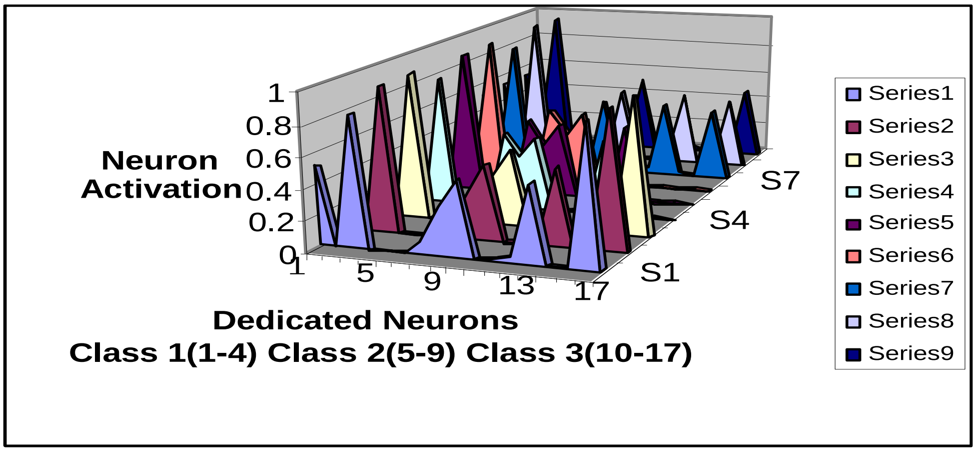
The Homogenous DNN was then tested with combinations of three features. Graphs 12.17 – 12.19 show the 27 combinations of 3 features. As 27 series in one graph reduce the visual clarity of the results, 9 series in each graph are presented.



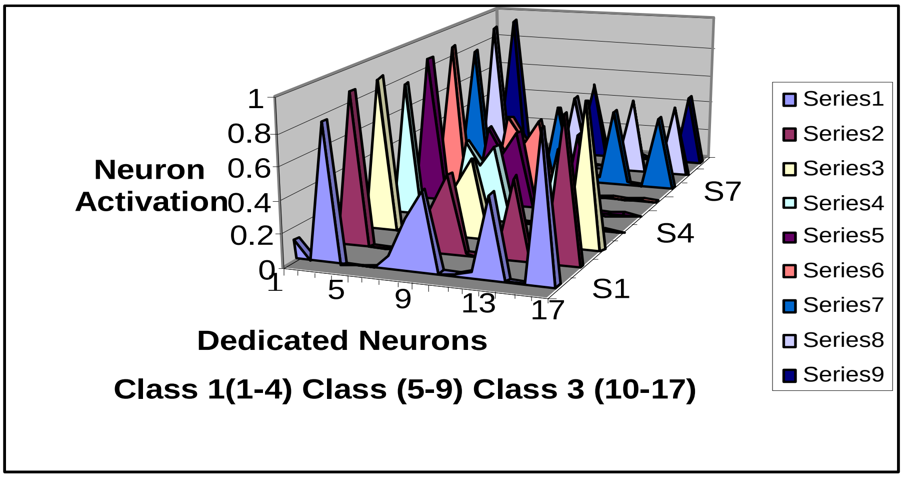
Graph 12.17: Homogenous DNN - Test with 9 Combinations of Class-1, Class-2 and Class-3 Features

The Graph 12.17 shows all of Series 9 being correctly identified i.e., all 3 features in the pattern have been recognised as the corresponding dedicated neurons have an activation ≥ 0.4.

In some cases, at least two features are recognised.



Graph 12.18: Homogenous DNN - Test with 9 Combinations of Class-1, Class-2 and Class-3



Graph 12.19: Homogenous DNN - Test with 9 Combinations of Class-1, Class-2 and Class-3

Also, in Graphs 12.18 and Graph 12.19, Series 9 is the only feature that displayed high activation (≥ 0.4) of corresponding dedicated neurons

It can be concluded from the results obtained that the homogenous DNN was not as successful as the original DNN with lateral inhibitory connections. The results are compared in Table 12.1.

12.3 SIGNIFICANCE OF ORDER OF INFORMATION OF FLOW

One of the features incorporated in the design of the architecture of DNN was the lateral interaction between dedicated neurons in Layers 4 and 5 as shown in Figure 12.1.

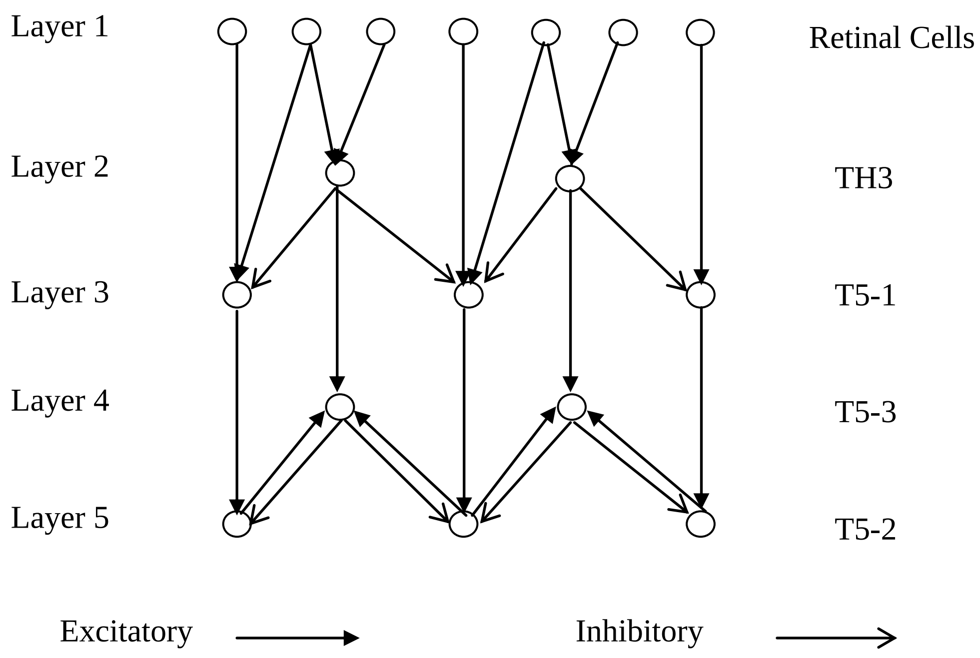


Figure 12.1: DNN Design Stage 2 - Neural Connections Revisited

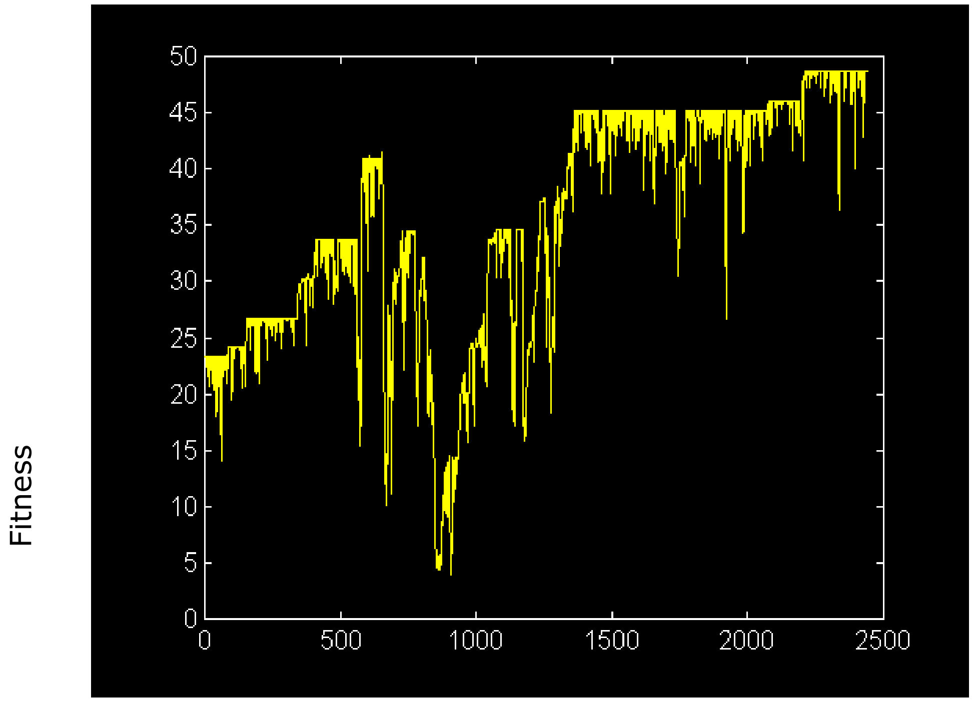
During the computation process, decisions had to be made regarding the interaction between output neurons. The steps involved in the interaction between output neurons in the original training phase, discussed in Chapter 9 were:

1. The temporary output of Neurons in Layer 4 is calculated, based on information received from other neurons in the top layers.
2. This temporary output of Layer 4 neurons is fed via inhibitory connections to Layer 5 neurons.
3. The output of Layer 5 neurons is calculated.
4. This output of Layer 5 neurons is passed via excitatory connections to Layer 4 neurons where it is combined with the temporary output mentioned in Point 1, above.
5. The final output of Layer 4 neurons is calculated.

The next stage in the analysis of the architecture was reversing the information flow between output neurons. The DNN where the information flow is reversed compared to the original DNN is called DNN_R. This was done to check if the order of information flow between dedicated output neurons was important.

Evaluating the significance of the order of information flow in ANNs with lateral inhibitory connections is essential to understand the relevance of FUSION with regards to its impact on the learning process in the DNN and DNN_R.

The DNN_R was trained using the training set mentioned in Chapter 9. A fitness of 48.667 was reached in 2432 generations as shown in Graph 12.20. The high fitness score is due to the reward mechanism. A discussion regarding the fitness is given in the Section 12.4. The DNN_R was tested for all features in the training and test sets as given in Chapter 9.

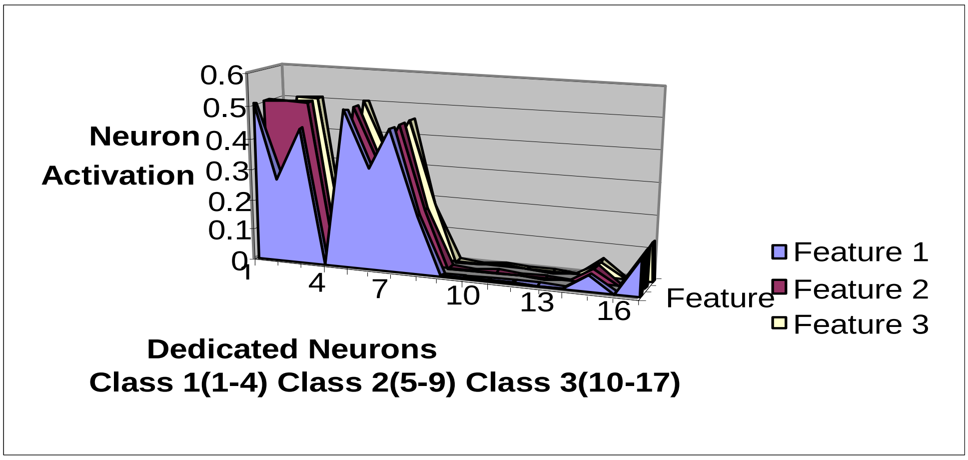


Graph 12.20: Training the DNN with the information flow between dedicated neurons reversed

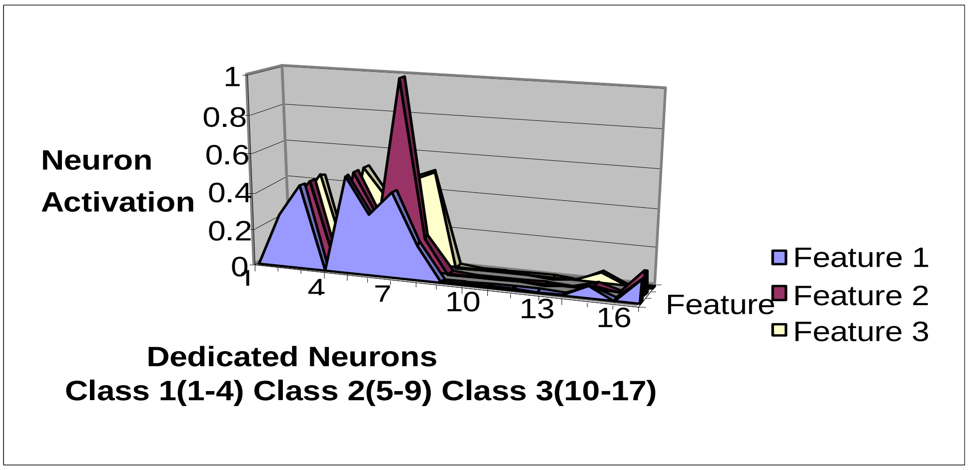
12.3.1 TEST WITH IMAGES FROM THE TRAINING SET

As has been the pattern of testing during this research programme, the DNN_R was tested with features from the training set followed by new and noisy features.

The purpose of the tests was to study the importance of order of information flow between the specific types of connections between the dedicated neurons.

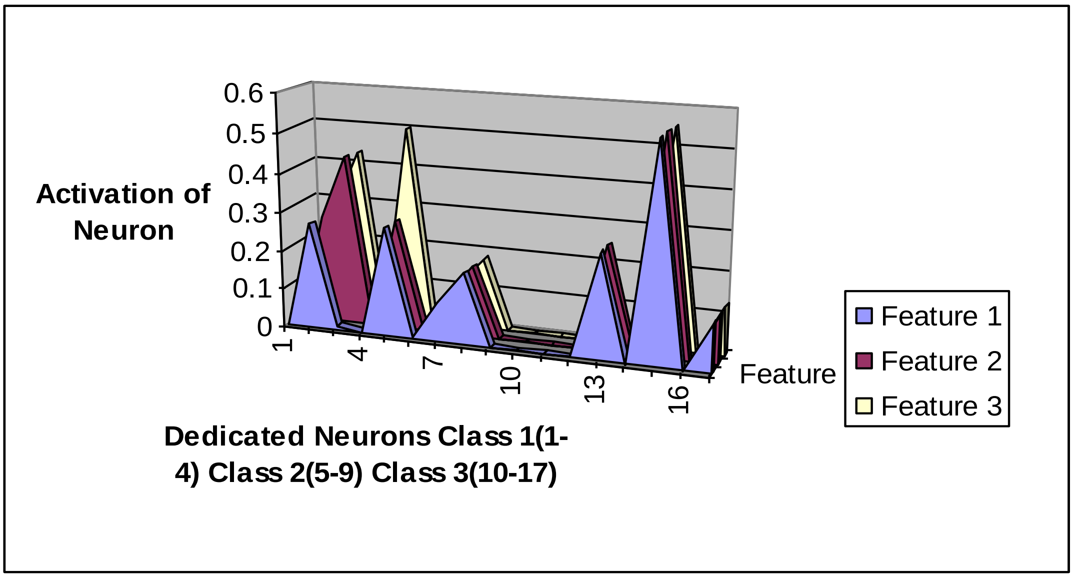


Graph 12.21: Reversed Flow DNN - Classification of Three Class-1 Features from the Training Set



Graph 12.22: Reversed Flow DNN - Classification of Three Class-2 Features from the Training Set

As seen in Graph 12.21, 2 of the Class 1 features were recognised, although the difference between the activation levels of Class 1and Class 2 features is indistinguishable. Graph 12.22 shows that all 3 features were recognised. The conclusion can be drawn as the criteria for recognition as specified earlier was based on the highest activation of dedicated neurons.



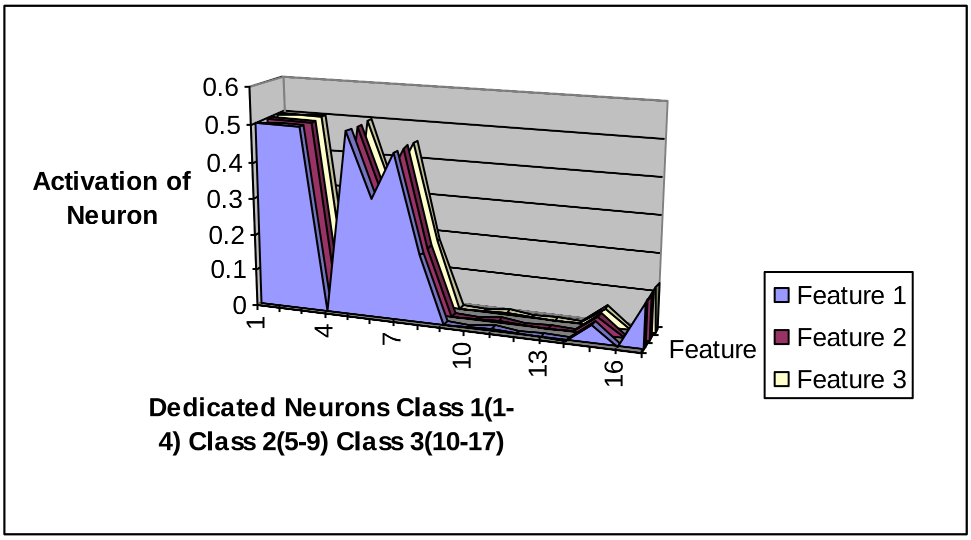
Graph 12.23: Reversed Flow DNN - Classification of Three Class-3 Features from the Training Set

In Graph 12.23, it can be seen that all three features were recognised. This can be concluded because the Class 3 feature dedicated neurons show the highest activation.

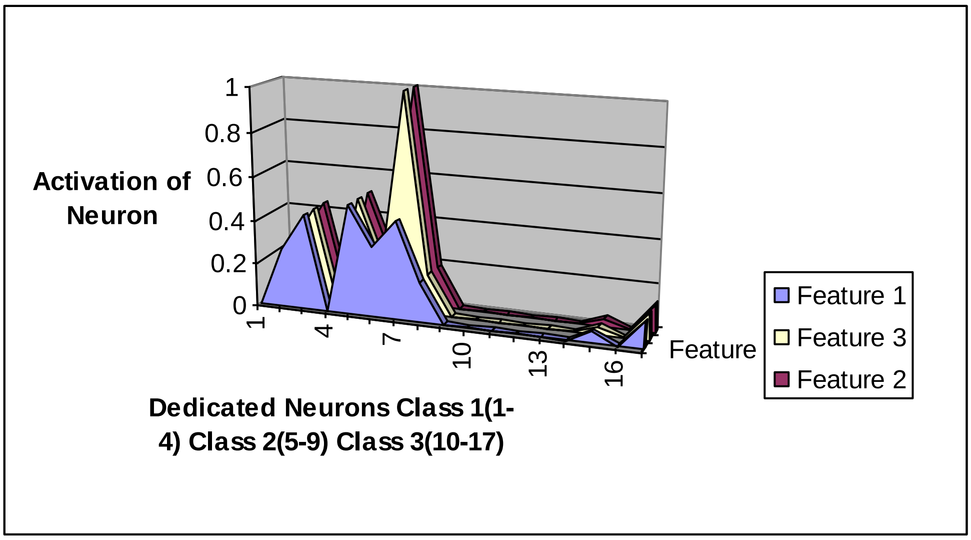
It can be noted that the distinction between various Classes is very coarse.

12.3.2 TEST WITH NEW FEATURES IN THE TEST SET

The second stage of testing involved testing the DNN_R with new features. The new features are given in Chapter 9. Graphs 12.24 – 12.26 illustrate the results obtained when the new features belonging to Class 1, Class 2 and Class 3 were respectively tested.

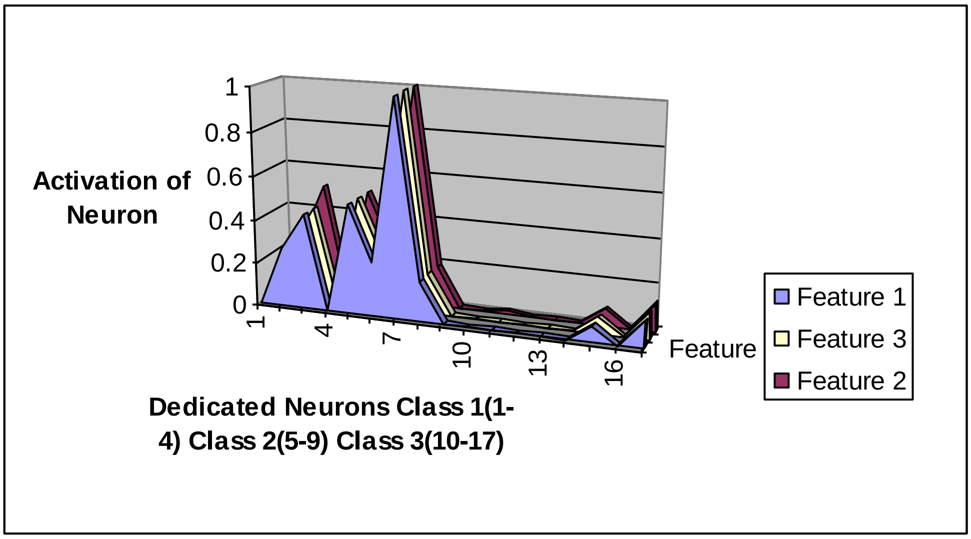


Graph 12.24: Reversed Flow DNN - Classification of Three New Class-1 Features from the Test Set



Graph 12.25: Reversed Flow DNN - Classification of Three New Class-2 Features from the Test Set

Graph 12.24 shows that 3 of the new Class 1 features were recognised by the DNN_R. Also, Graph 12.25 shows 3 Class 2 features being recognised by the DNN_R.



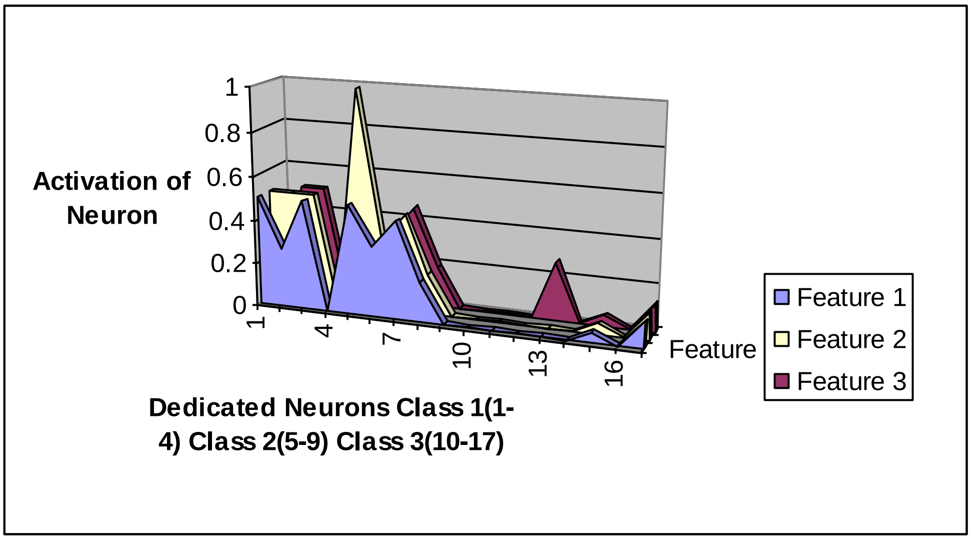
Graph 12.26: Reversed Flow DNN - Classification of Three New Class-3 Features from the Test Set

In Graph 12.26, none of the dedicated Class 3 features show any activation when the new Class 3 features were used to test the DNN_R.

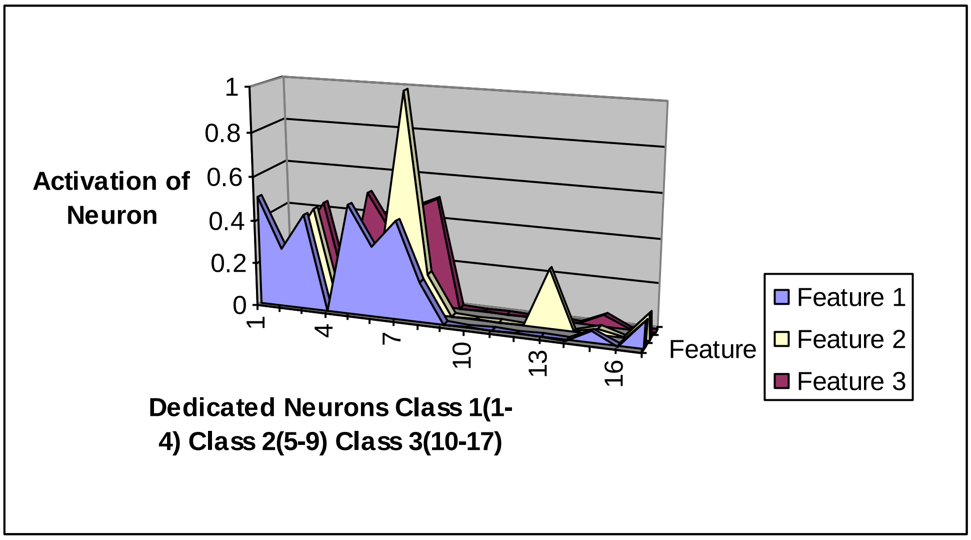
12.3.3 TEST WITH NOISY FEATURES IN THE TEST SET

Tests were conducted on the DNN_R with reversed information flow using two sets on noisy features. The features are presented in Chapter 9. This section presents results obtained when tested with the first set of noisy features with 1 speckle introduced in various pixels of the image.

Graphs 12.27 – 12.29 illustrate the results obtained when tested with Class 1, Class 2 and Class 3 features.

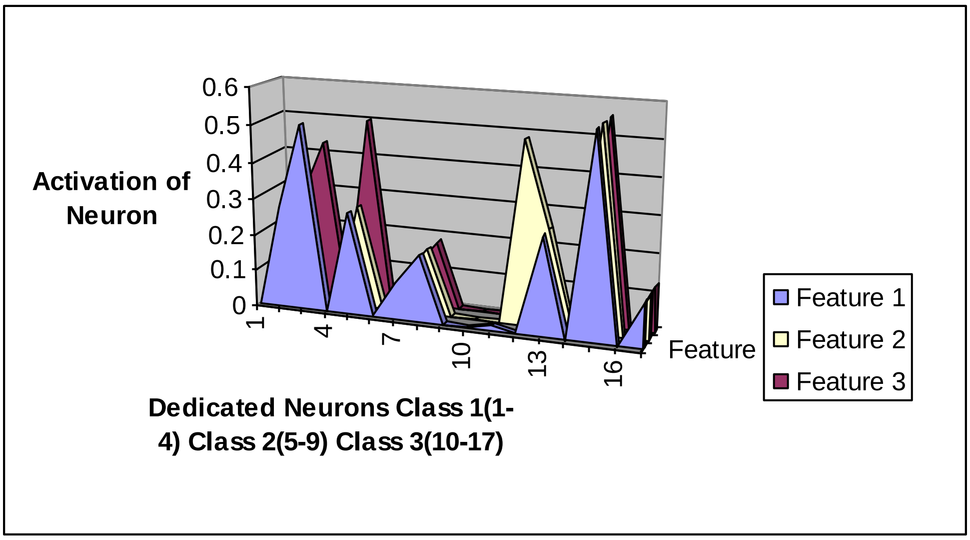


Graph 12.27: Reversed Flow DNN - Classification of Three Noisy Class-1 Features from the Test Set



Graph 12.28: Reversed Flow DNN - Classification of Three Noisy Class-2 Features from the Test Set

In Graph 12.27, it can be seen that dedicated neurons belonging to two Class 1 features show highest activation. Hence, two of the Class 1 features are recognised. In Graph 12.28, two of the class 2 features are recognised.

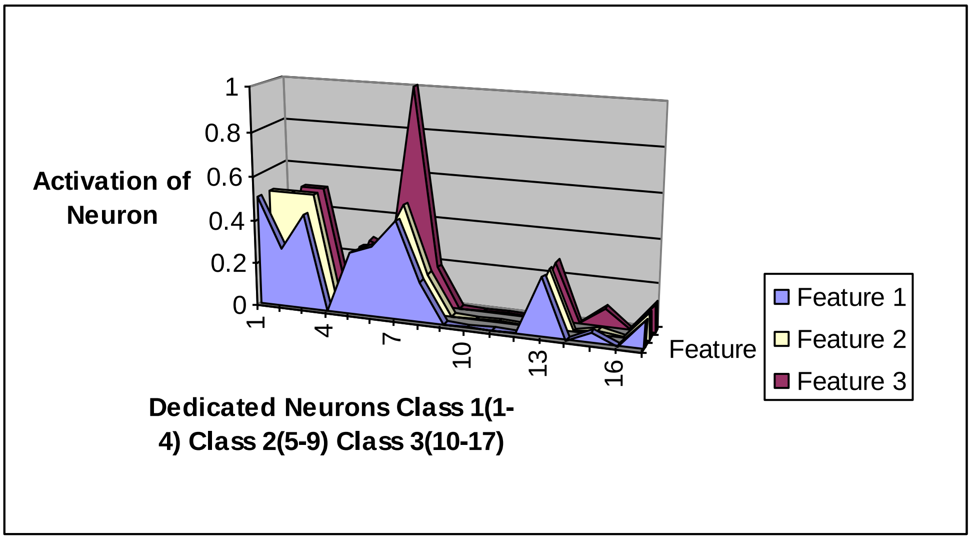


Graph 12.29: Reversed Flow DNN - Classification of Three Noisy Class-3 Features from the Test Set

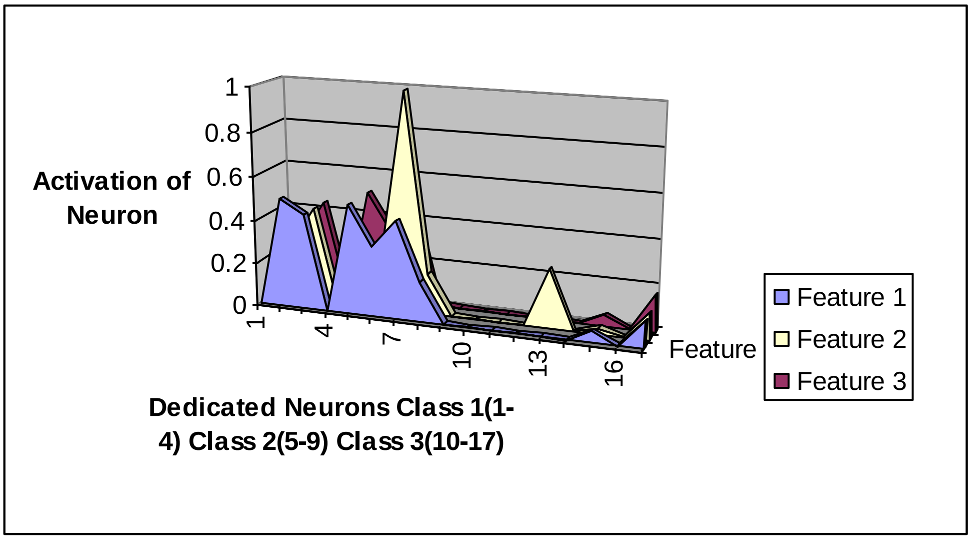
In Graph 12.29, the highest activation values are observed among Class 3 dedicated neurons. Hence, all Class 3 features are recognised.

12.3.4 TEST WITH FEATURES WITH INCREASED LEVELS OF NOISE

The next step involved testing the DNN_R with images containing increased levels of noise as shown in Chapter 9. Graphs 12.30 – 12.32 illustrate the results obtained when the DNN_R is tested with Class 1, Class 2 and Class 3 features respectively.

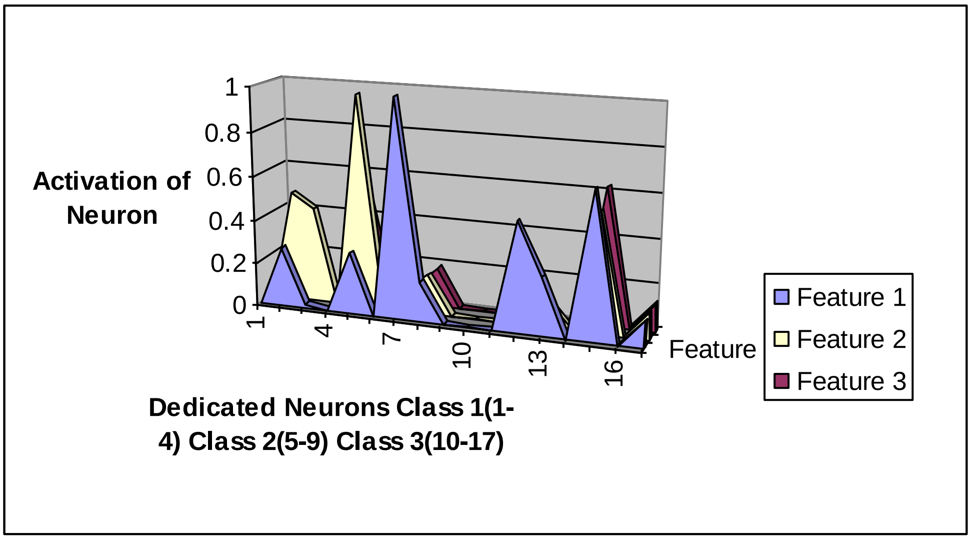


Graph 12.30: Reversed Flow DNN - Classification of Three Class-1 Features from the Test Set with Increased Noise



Graph 12.31: Reversed Flow DNN - Classification of Three Class-2 Features from the Test Set with Increased Noise

Graphs 12.30 and 12.31 show that two features of Class 1 and Class 2 are recognised. This can be seen by the high activation value of the corresponding dedicated neurons.



Graph 12.32: Reversed Flow DNN - Classification of Three Class-3 Features from the Test Set with Increased Noise

As shown in Graphs 12.30, 12.31 and 12.32, two features of Class 1, two features of Class 2 and 1 feature of Class 3 are respectively recognised.

Overall, Graphs 12.21 – Graph 12.32 graphically represent the results obtained when the DNN was tested with a reversal in the information flow between dedicated output neurons to its original design. This network was called DNN_R.

The reward mechanism, training data and test set were the same as those given in Chapter 9. The results obtained show that the classification of features is ambiguous in most cases. The results also show that the order of information flow is significant when there is interaction between dedicated neurons.

12.4 SUMMARY

In this chapter the significance of two main aspects of the architecture of DNN were examined; the type of neural connections and the order of information flow between dedicated neurons.

The positive results obtained while training the homogenous DNN suggest that the training method FUSION might be well endowed to evolve an array of connections between the neurons without any necessity for some foundational genetic make-up with regards to the connections. Results obtained when the homogenous DNN was tested for new patterns, noisy patterns and combinations of patterns hint that the initial genetic make-up of the connections might be more significant than first thought and suggest that the homogenous DNN is not as robust as the original DNN tested in Chapter 9.

Also, the order of information flow between the dedicated neurons was reversed in the DNN. The network was called DNN_R. The results obtained show that the order of information flow has significant bearing on the recognition of various classes of features although the same training method (FUSION) was used.

When the DNN_R was tested, the distinction between the activation of various classes of neurons was coarse, which increased with noise. As the activation of various dedicated neurons for the tests conducted initially were blurred, the DNN_R was not tested for combination of features. Table 12.1 gives a comparison of results obtained.

S. No	TEST DATA	SUCCESS RATE (%)		
		ORIGINAL DNN	HOMOGENOUS DNN	DNN WITH REVERSED INFORMATION FLOW (DNN_R)
1	Original Training Features	100%	100%	88.89%
2	New Features	100%	55.56%	66.67%
3	Noisy Features (1 speckle)	100%	77.78%	77.78%
4	Increased Noise Features (2 speckles)	100%	77.78%	55.56%
5	Combination of Two Features	81.48%	66.67%	-
6	Combination of Three Features	66.67%	11.11%	-

Table 12.1: Summary and Comparison of Results obtained for the Tests Conducted on the Architecture of DNN

Hence, it can be concluded that although some promising results were obtained, the performance of the DNN was affected by the change in the type of connections and the order of information flow between output neurons.

Another aspect that needs attention is the training and fitness achieved by the three architectural variations of the DNN. The table 12.2 given below illustrates the fitness achieved in each case.

NETWORK	FITNESS ACHIEVED	NUMBER OF GENERATIONS
Original DNN	48.667	13083
Homogenous DNN	48.667	8897
DNN_R	48.667	2432

Table 12.2: Illustration of Fitness Achieved in three Architectural Variations of the DNN

All the variations of ADD were trained to achieve a fitness of 48.667. This was achieved much faster by the DNN_R and slower in the homogenous DNN. In both these cases, the various Classes of features were not classified to the same degree of perfection as in the Original DNN, which took much longer to achieve the same fitness.

These results show that although learning can play a major role in training ANNs, the initial structure of the DNN is also important in enhancing the overall performance of the DNN.

Analyses of both aspects of architectural variance and the learning algorithm's relationship with the architecture of the network are major contributions to the state of the art.

CHAPTER 13: ANALYSIS OF TRAINING AND REWARDS FOR THE DNN

13.1 INTRODUCTION

The novel characteristics of the DNN were established through methodical analysis of various aspects of the DNN. In Chapter 12, some aspects of the architecture of the DNN were examined.

Other issues that may play a prominent role in training the DNN are the reward mechanisms. Prior to devising a successful reward mechanism, several other reward systems were tried and tested. Some of the stages of improvement of the reward mechanism are presented in this chapter. The training data used in all the proposals is the same set used throughout the research programme.

The investigation of these methods is necessary to appreciate the DNN's response to various modes of reward or training.

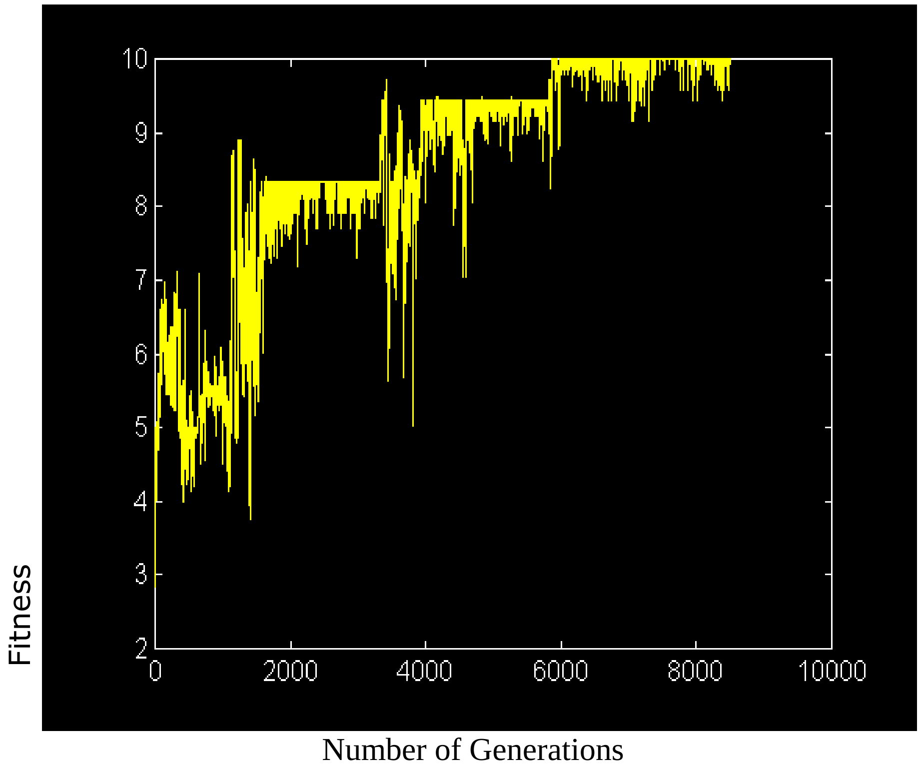
13.2 PROPOSAL-1 FOR TRAINING THE DNN

The very first reward method proposed to train the DNN was the simplest method as well. It involved a reward method that was based only on the recognition of a feature that belonged to a particular class.

The reward proposals involved the following steps:

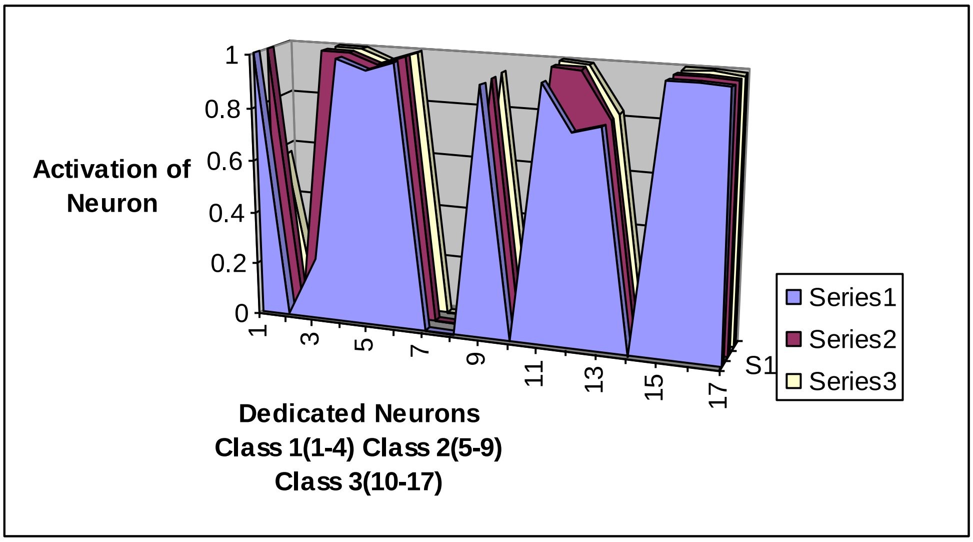
1. Check the neurons with the highest activation.
2. Classify the feature based on the neuron activation.
3. Verify if the classification is similar to the desired categorisation.
4. If the actual and desired classification of a feature is the same, then allocate a reward of 10 points.
5. If the classification is dissimilar, but the feature was recognised as one that closely matches (e.g. Class-1 and Class-2 features are horizontal whereas Class-3 features are vertical), then allocate a reward of 5 points.
6. In all other cases allocate a reward of zero points.

The DNN was trained using the method above. A fitness of 9.998 was reached in 8879 generations as shown in Graph 13.1.

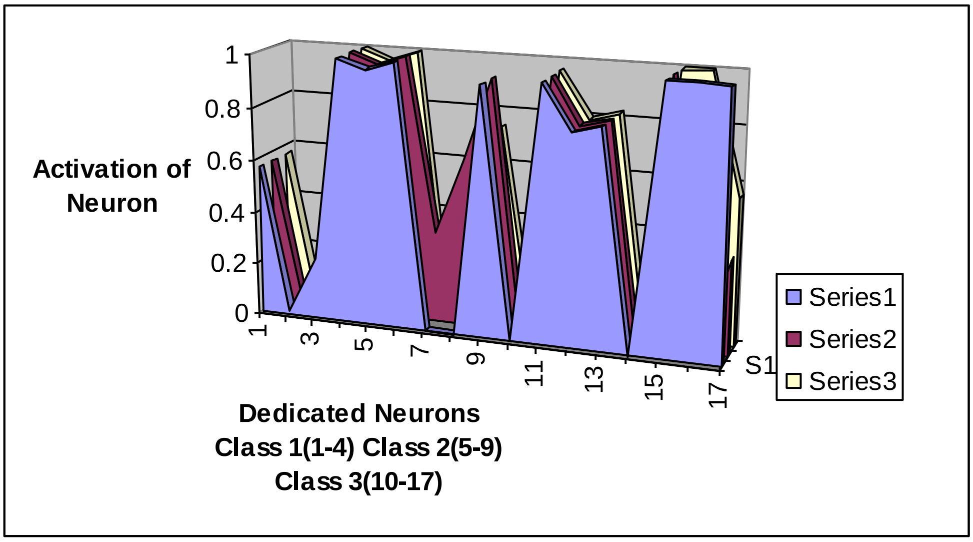


Graph 13.1: DNN Trained using a Fitness Criterion according to Proposal-1

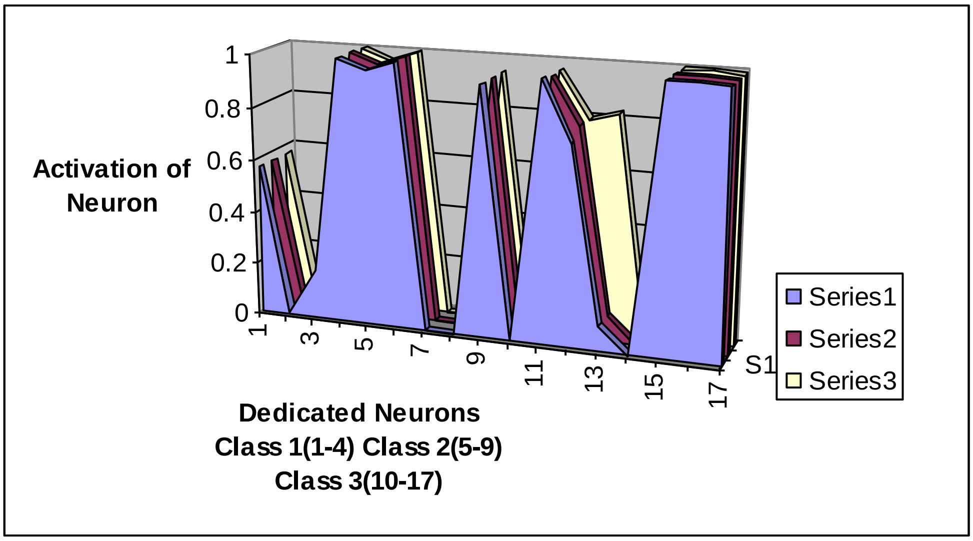
The results obtained from training the DNN using Proposal-1 are shown in Graphs 13.2, 13.3 and 13.4 respectively.



Graph 13.2: DNN Trained for Class-1 Features according to Training Proposal-1



Graph 13.3: DNN Trained for Class-2 Features according to Training Proposal-1



Graph 13.4: DNN Trained for Class-3 Features according to Training Proposal-1

The results show that the DNN has not been successfully trained for recognising the features from the training set.

Reward Proposal-1 is most similar to Evolutionary Algorithms used to train ANNs. However, a direct calculation of error to check the deviation from desired output is not done in Proposal-1. Usually in EAs used for training ANNs the error is calculated in order to establish the fitness of chromosomes. It can be said that EAs used for ANNs are the simplest forms of FUSION.

To improve the results, the reward mechanism had to be improved where factors such as the activation levels of neurons are considered. The Training Proposal-2 is given in Section 13.3.

13.3 TRAINING THE DNN USING TRAINING PROPOSAL-2

The second proposal was similar to the Proposal-1, but included three new factors. Hence, the changes in the reward system are incorporated various steps in Proposal-1, which are given in the list given below.

- Check the neurons with the highest activation.
- Classify the feature based on the neuron activation.
- Count the number of opposition neurons that have activation, which is less than 0.5.
- Verify if the classification is similar to the desired categorisation.
- If the actual and desired classification of a feature is the same, then

Reward = 10 points provided Class-1-opposition neurons = 5 or
Class-2-opposition neurons = 4 or
Class-3-opposition neurons = 9

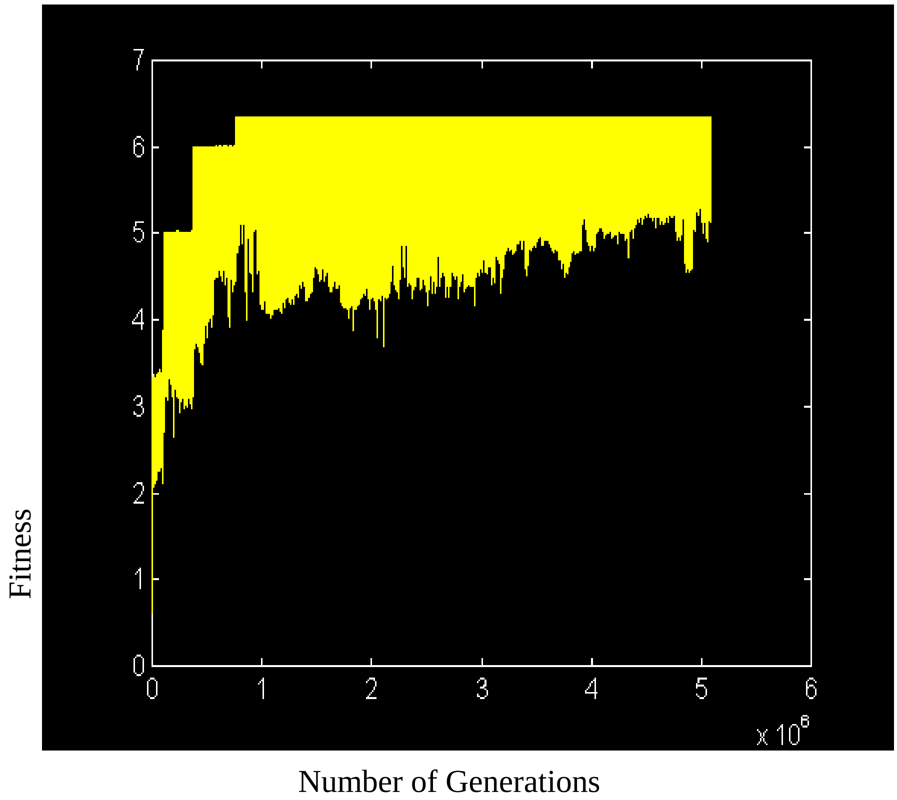
Reward = 9 points provided Class-1-opposition neurons = 4 or
Class-2-opposition neurons = 3 or
Class-3-opposition neurons = 8

Reward = 8 points provided Class-1-opposition neurons = 3 or
Class-2-opposition neurons = 2 or
Class-3-opposition neurons = 7

Reward = 7 points provided Class-1-opposition neurons = 2 or
Class-2-opposition neurons = 1 or
Class-3-opposition neurons = 6

• If the classification is dissimilar, but the feature was recognised as one that closely matches (e.g. Class-1 and Class-2 features are horizontal whereas Class-3 features are vertical), then allocate a reward of 5 points.

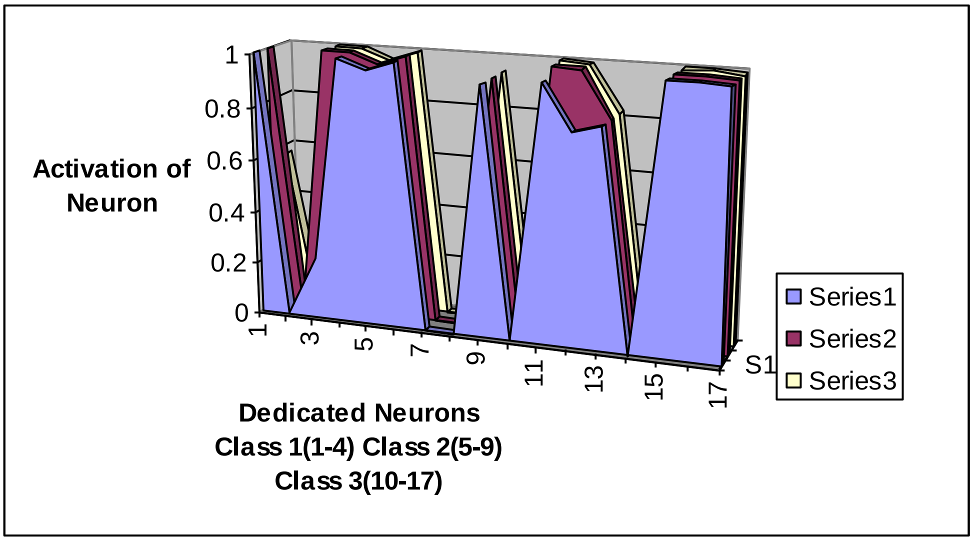
• In all other cases allocate a reward of zero points.



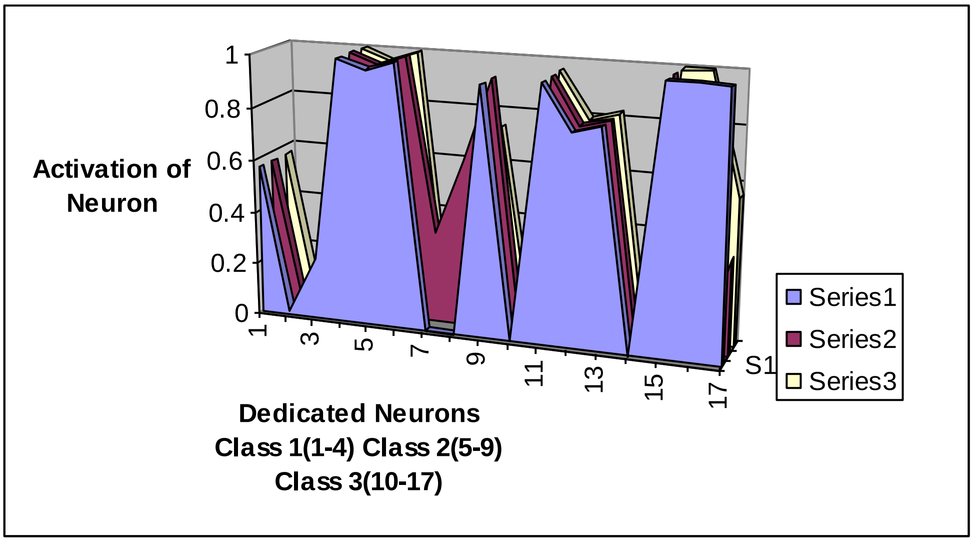
Graph 13.5: DNN Trained using a Fitness Criterion according to Proposal-2

The DNN was trained with the training Proposal-2 mentioned above. A fitness of 6.32 was achieved in 5335079 generations as shown in the Graph 13.5.

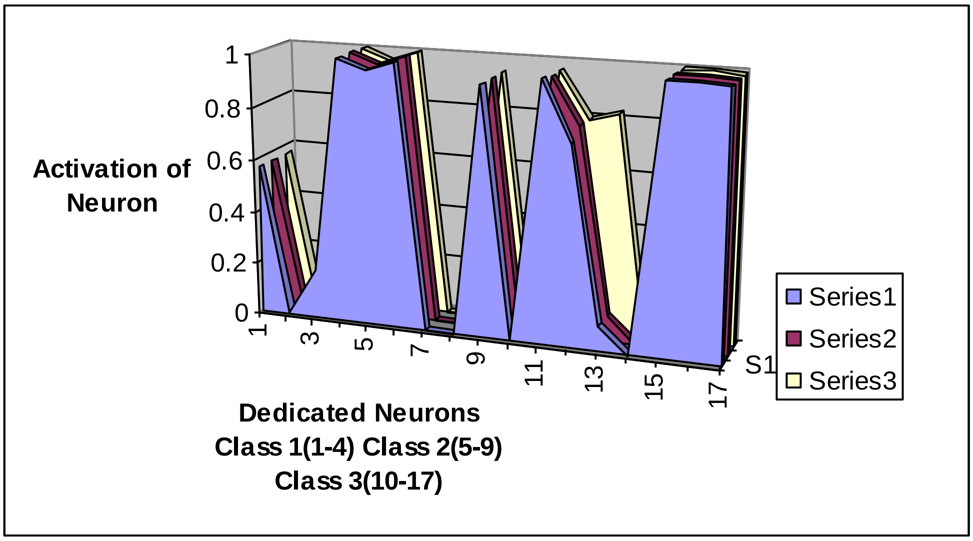
The results obtained from training the DNN using Proposal-2 is shown in Graphs 13.6, 13.7 and 13.8 respectively.



Graph 13.6: DNN Trained for Class-1 Features according to Training Proposal-2



Graph 13.7: DNN Trained for Class-2 Features according to Training Proposal-2



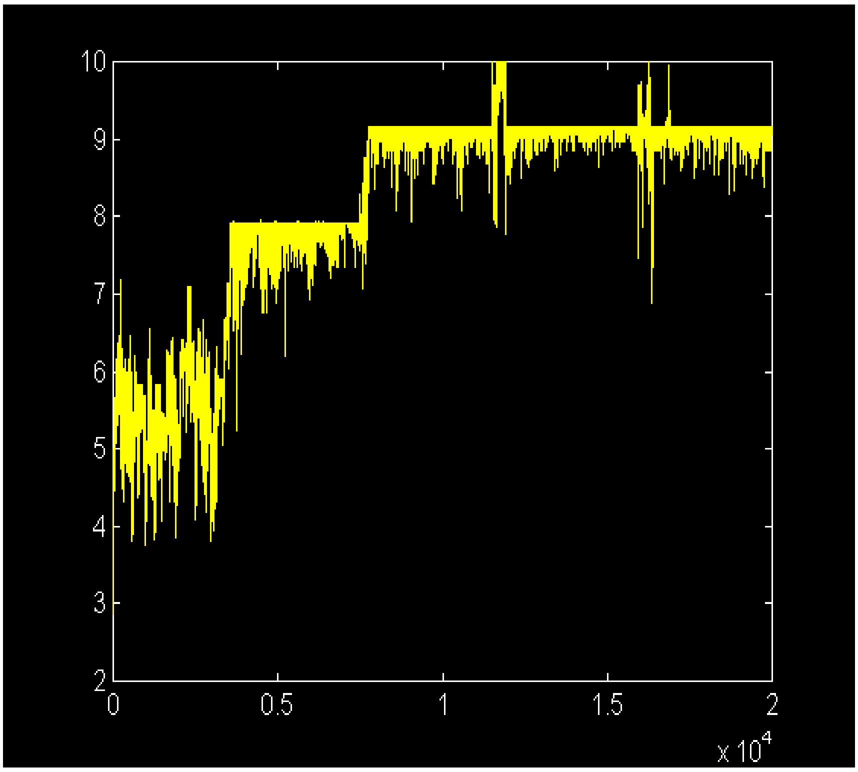
Graph 13.8: DNN Trained for Class-3 Features according to Training Proposal-2

The results presented in Graphs 13.6, 13.7 and 13.8 show that none of the features were recognised. There was no distinction between various Classes of features.

13.4 TRAINING THE DNN USING TRAINING PROPOSAL-3

Training Proposal-2 mentioned in Section 13.3 had a combined reward scheme for recognition of all patterns and it was implemented in C++ using only one function.

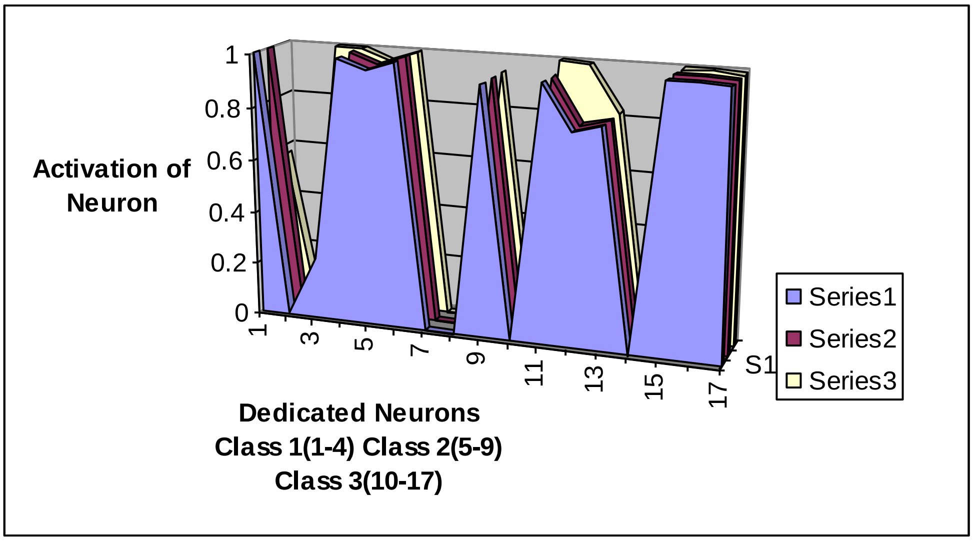
Training Proposal-3 involved enhancement of the Proposal-2, in one important manner. Separate reward schemes were created for Horizontal (Class-1 and Class-2) features and Vertical features (Class-3). The implementation in C++ involved operation of these reward schemes using two separate functions.



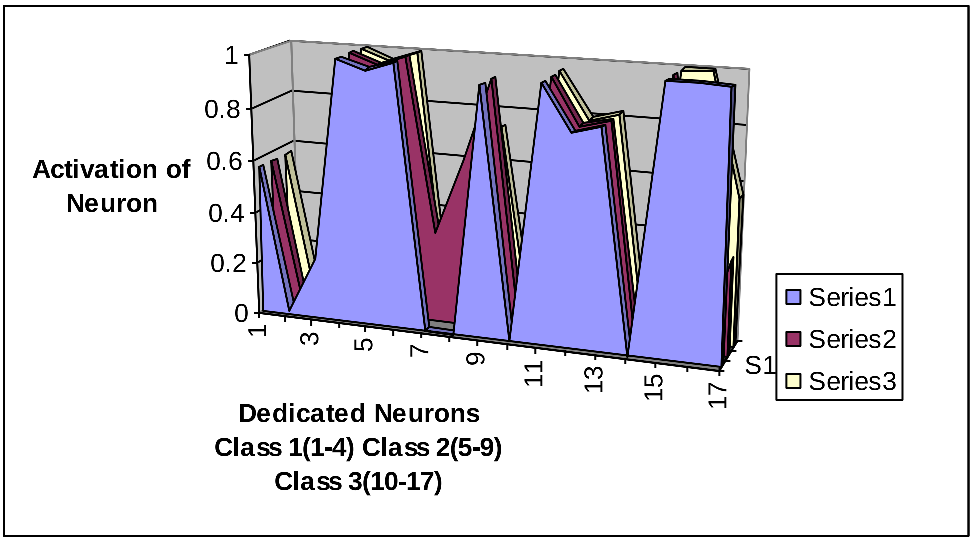
Graph 13.9: DNN Trained using a Fitness Criterion according to Proposal-3

The factors and steps involved in the allocation of rewards were similar to Proposal-2, but were now operated under separate headings reducing the ambiguity and increasing the prospects of using other combinations of factors to decide upon the reward. The maximum reward that could be allocated was 10 points for each Class.

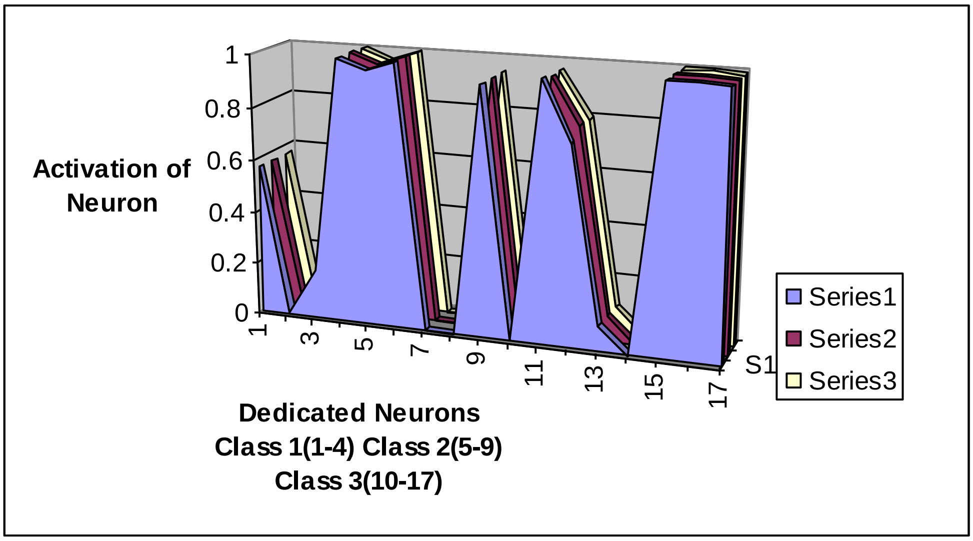
The DNN was trained using Training Proposal-3. A fitness of 9.235 was achieved in 20000 generations as shown in Graph 13.9. The results obtained from training the DNN using Proposal-1 is shown in Graphs 13.10, 13.11 and 13.12 respectively.



Graph 13.10: DNN Trained for Class-1 Features according to Training Proposal-3



Graph 13.11: DNN Trained for Class-2 Features according to Training Proposal-3



Graph 13.12: DNN Trained for Class-3 Features according to Training Proposal-3

Proposal-3 did not improve the results as seen in Graphs 13.10, 13.11 and 13.12. Other methods were used and tested to improve the performance.

13.5 ASPECTS OF NETWORK TRAINING USING FUSION

The final proposal used for training the DNN was FUSION, which was explained in Chapter 8. One aspect of the training method had a considerable impact on the performance of the DNN. The number of consecutive times a maximum reward was achievable had an important role to play in driving the training method. In other words, rather than the training being stopped when a maximum fitness was achieved, insistence on achieving the maximum fitness over a number of consecutive generations improved the reliability of the DNN.

Two sets of results are presented in the following sections. The first was when repeated achievement of fitness was not required and the second one was one in which the reward was achieved 4 consecutive times.

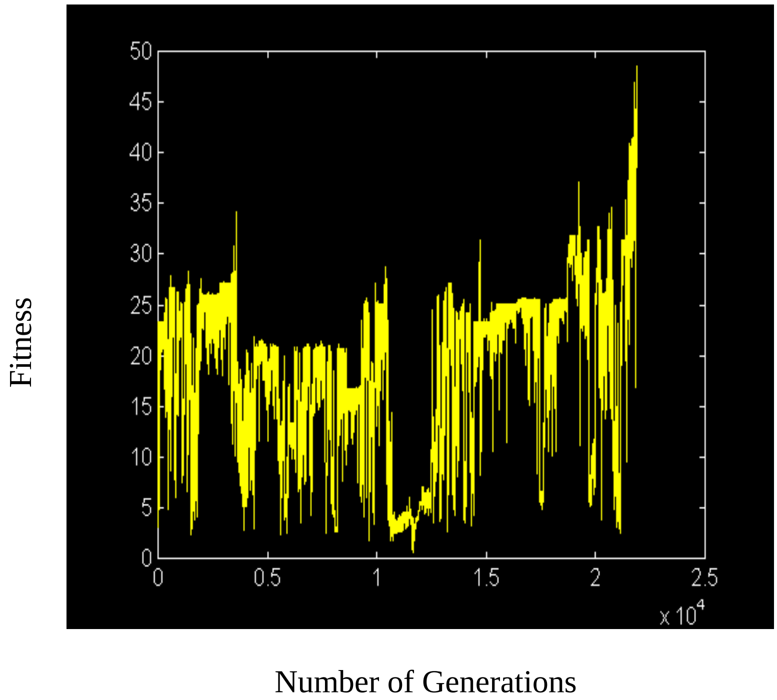
The DNN was tested for images from the test data as given in Chapter 9. The images include:

1. Features from the training set
2. New Features
3. Noisy Features (Stage-I)
4. Noisy Features (Stage-II)
5. Combinations of Two Features
6. Combinations of Three Features

13.5.1 PHASE-1 DNN TRAINING USING FUSION METHOD-1

The DNN was trained using FUSION to achieve the maximum fitness. Repeated achievement of fitness was not considered in the training. A fitness of 48.3333was achieved in 21800 generations as shown in Graph 13.13.

From the graph, it can be observed that the fitness drops to very low ranges even after 20000 generations. The efficiency of FUSION is questionable if the fitness has not reached a state where it is unwavering. It can be concluded that FUSION has not established any reliable network parameter that would help in better network performance. The DNN was tested to verify its abilities.



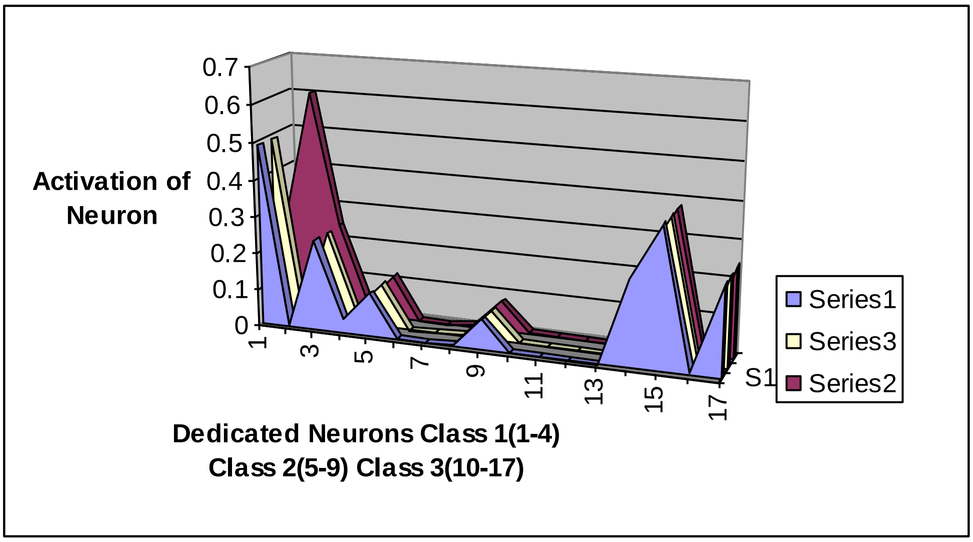
Graph 13.13: Training the DNN using FUSION Method-1

13.5.1.1 TESTING DNN WITH FEATURES FROM TRAINING SET

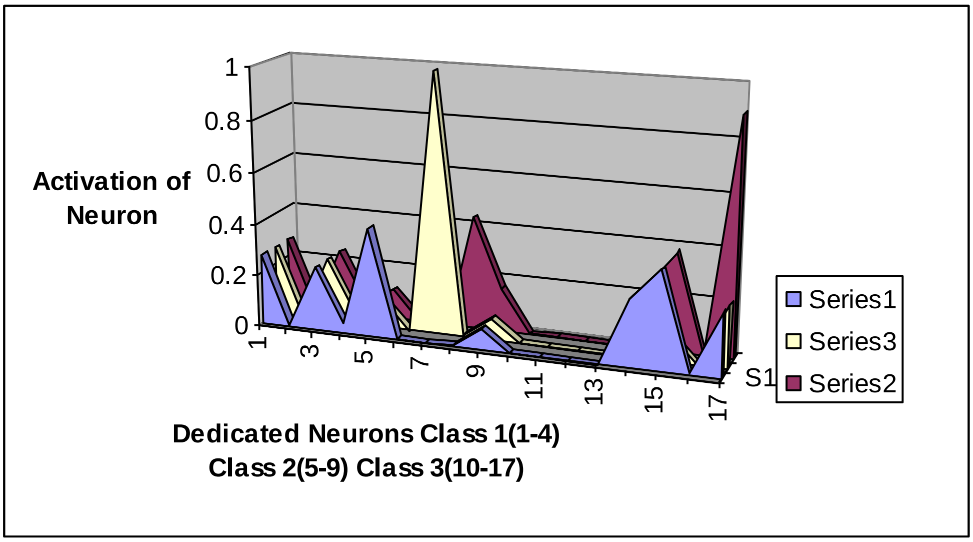
The DNN trained using FUSION Method-1 was tested using features from the training and test sets as given in Chapter 9.

The ideal case of results aimed for in this research programme was one in which the dedicated neurons of a certain class show the highest activation when tested for that particular class of features, while the rival neurons show the least activation, ideally zero.

Graphs 13.14, 13.15 and 13.16 show the results obtained when the DNN trained using FUSION-1 was tested using Class 1, Class 2 and Class 3 features from the training set.

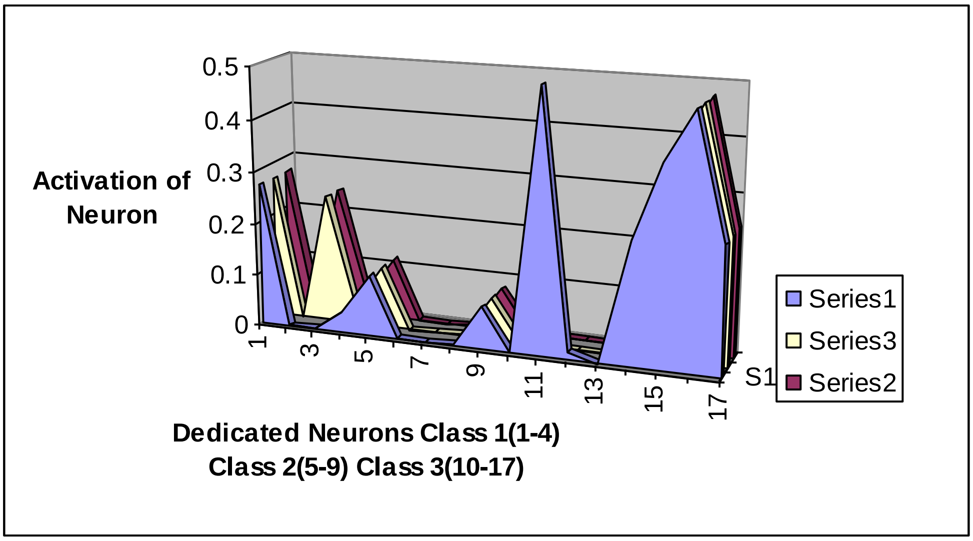


Graph 13.14: Testing the DNN trained using FUSION Method-1 for Class-1 Features from the Training Set



Graph 13.15: Testing the DNN trained using FUSION Method-1 for Class-2 Features from the Training Set

Graph 13.14 shows that all three Class 1 features are recognised. Graph 13.15 shows two Class 2 features being recognised by the high activation of the corresponding dedicated neurons.



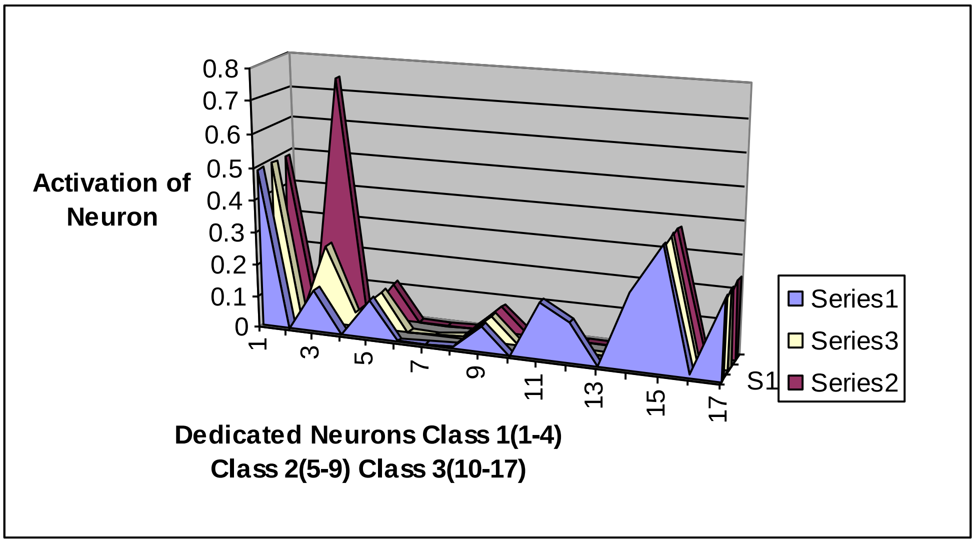
Graph 13.16: Testing the DNN trained using FUSION Method-1 for Class-3 Features from the Training Set

Graph 13.16 illustrates the recognition of all three Class 3 features. Although dedicated neurons belonging to a certain class that is being tested show high activation, the other rival neurons also have a high activation that reduces the distinction between various classes.

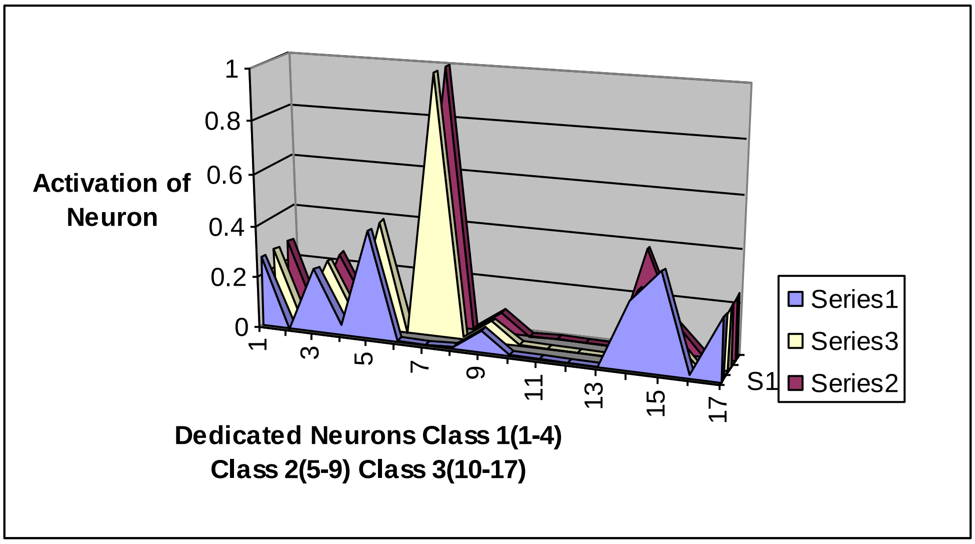
13.5.1.2 TESTING THE DNN WITH NEW FEATURES

The DNN trained using FUSION Method-1 was then tested with new features from the test set given in Chapter 9.

The results obtained when the network was tested with new features belonging to Class 1, Class 2 and Class 3 features are presented in Graphs 13.17, 13.18 and 13.19 respectively.

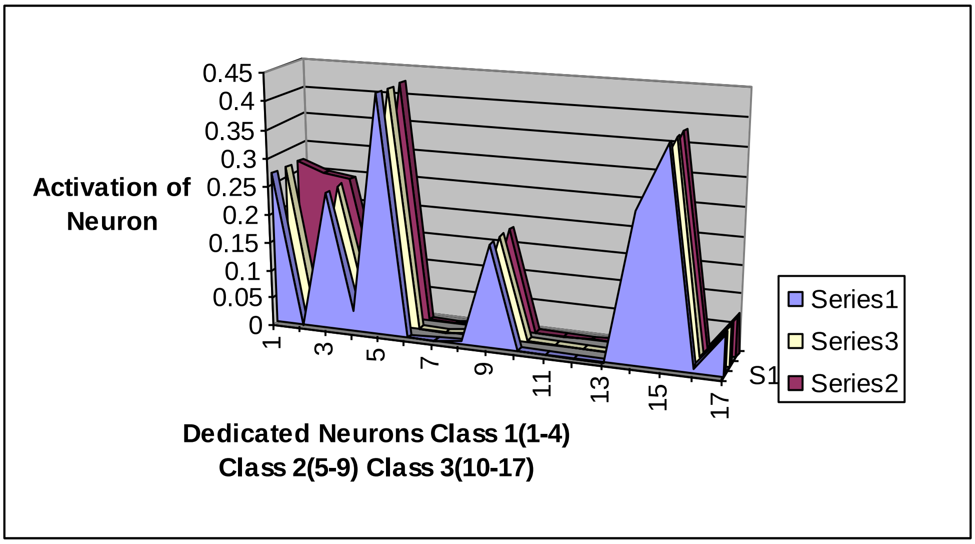


Graph 13.17: Testing the DNN trained using FUSION Method-1 for New Class-1 Features from the Test Set



Graph 13.18: Testing the DNN trained using FUSION Method-1 for New Class-2 Features from the Test Set

Graphs 13.17 and 13.18 show that all three features of Classes 1 and 2 were recognised successfully. But, the distinction between various classes is coarse.



Graph 13.19: Testing the DNN trained using FUSION Method-1 for New Class-3 Features from the Test Set

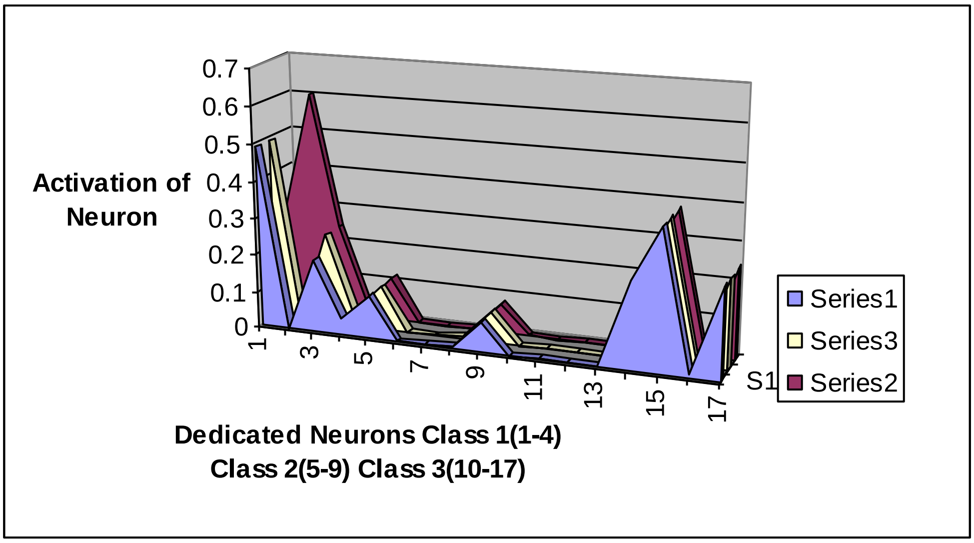
None of the new Class 3 features were recognised as shown in Graph 13.19.

13.5.1.3 TESTING THE DNN USING NOISY IMAGES

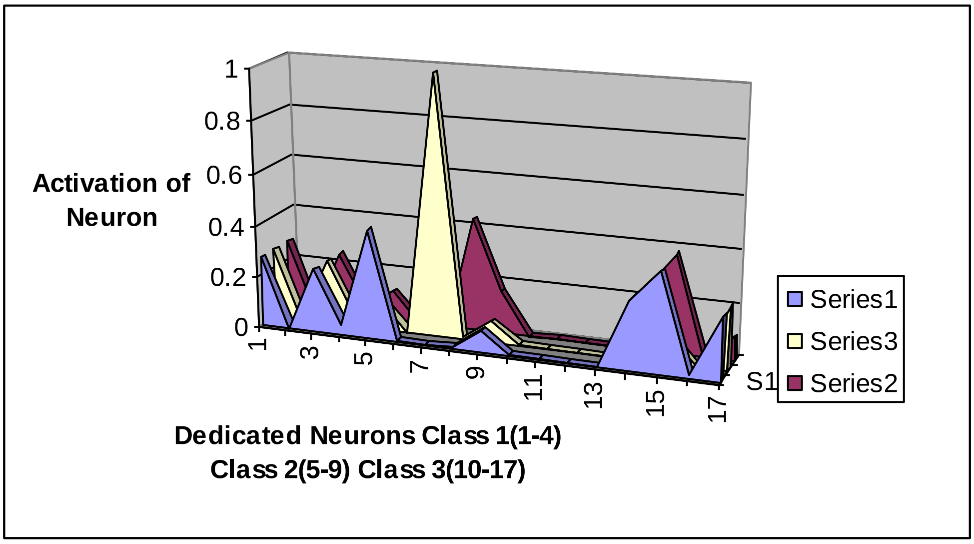
The next phase of testing involved testing the network with noisy images. The noisy images contained speckles as given in Chapter 9.

Tolerance of networks for noise was important as images are rarely in perfect conditions and hence the network will have to deal with noisy and corrupted images.

Graphs 13.20, 13.21 and 13.22 are illustrations of the results obtained when tested with Class 1, Class 2 and Class 3 noisy features respectively.

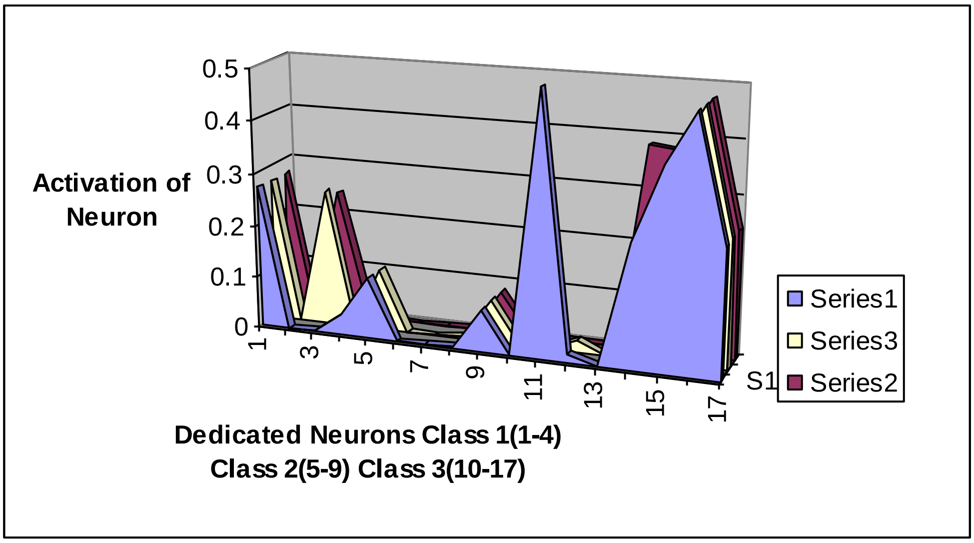


Graph 13.20: Testing the DNN trained using FUSION Method-1 for Noisy Class-1 Features from the Test Set



Graph 13.21: Testing the DNN trained using FUSION Method-1 for Noisy Class-2 Features from the Test Set

Graphs 13.20 and 13.21 show that all noisy Class 1 and Class 2 features are recognised successfully, as seen by the high activation values of the corresponding dedicated neurons.



Graph 13.22: Testing the DNN trained using FUSION Method-1 for Noisy Class-3 Features from the Test Set

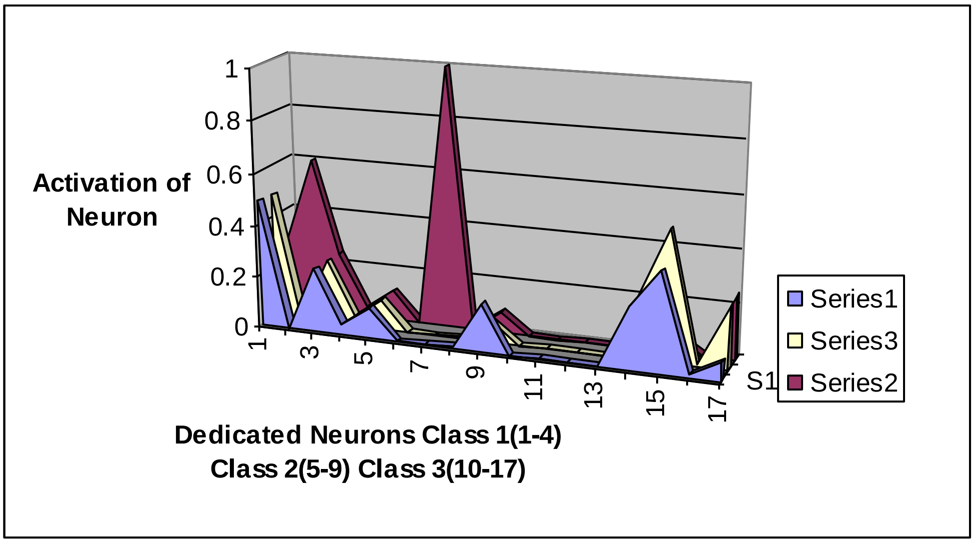
Graph 13.22 shows that two Class 3 features are recognised. It is evident from all the graphs that the distinction between various classes of neurons is not high.

13.5.1.4 TESTING THE DNN WITH FEATURES WITH INCREASED LEVELS OF NOISE

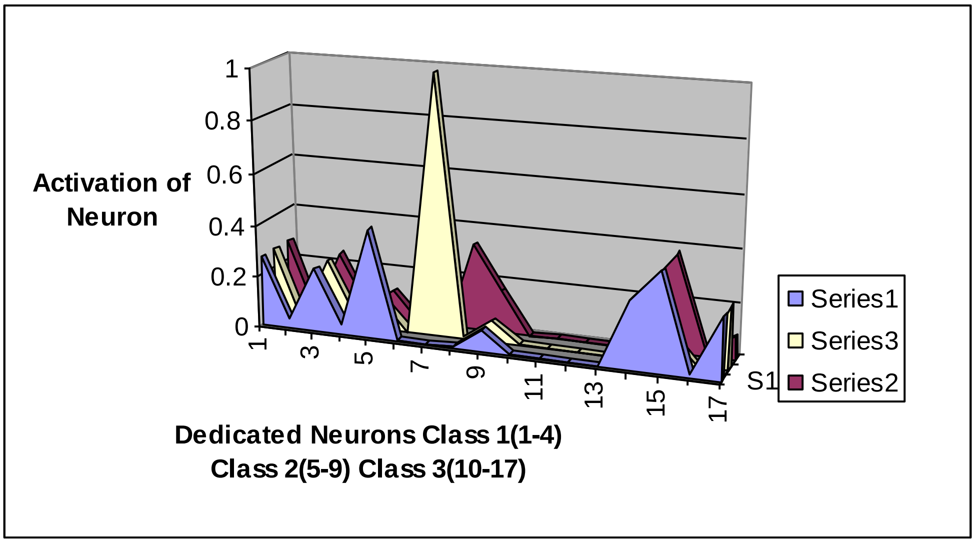
The next level of testing involved testing the network with increased levels of noise that was introduced in the form of two speckles. The test images with increased levels of noise are given in Chapter 9.

Three features each with two speckles of noise that belong to the three classes of features are used to test the DNN trained using FUSION Method-1.

Graphs 13.23, 13.24 and 13.25 illustrate the results obtained when tested with Class 1, Class 2and Class 3 features.

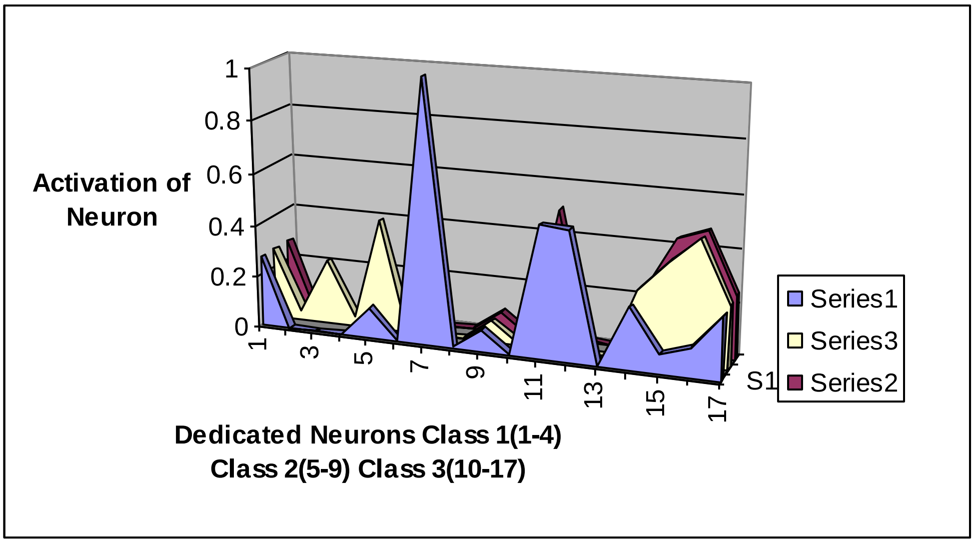


Graph 13.23: Testing the DNN trained using FUSION Method-1 for Class-1 Features with Increased Noise from the Test Set



Graph 13.24: Testing the DNN trained using FUSION Method-1 for Class-2 Features with Increased Noise from the Test Set

As shown in Graphs 13.23 and 13.24, only one feature of Class 1 and Class 2 have respectively been recognised. The corresponding rival dedicated neurons show high activation.



Graph 13.25: Testing the DNN trained using FUSION Method-1 for Class-3 Features with Increased Noise from the Test Set

Graph 13.254 shows that two Class 3 features with increased noise were recognised by the DNN.

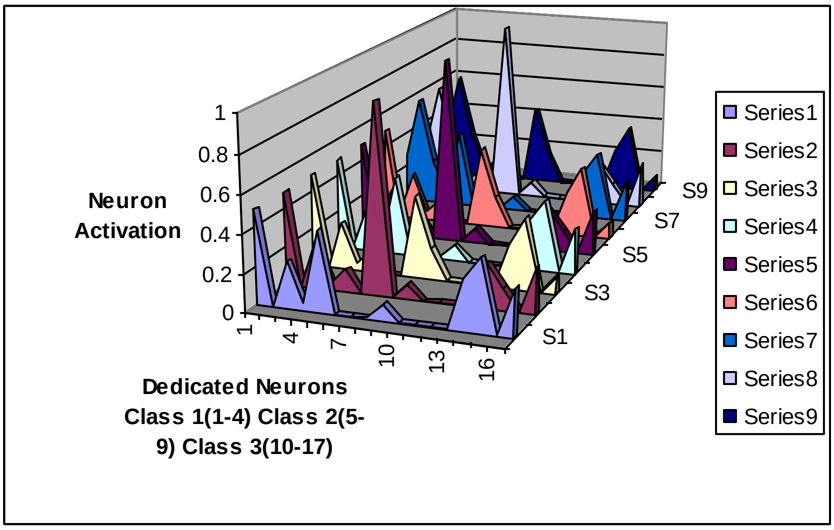
13.5.1.5 TESTING THE DNN WITH COMBINATIONS OF TWO FEATURES

The DNN trained using FUSION Method-1 was further tested for its ability to recognise combinations of features that it was trained for. Combinations of two and three features were used to test the DNN.

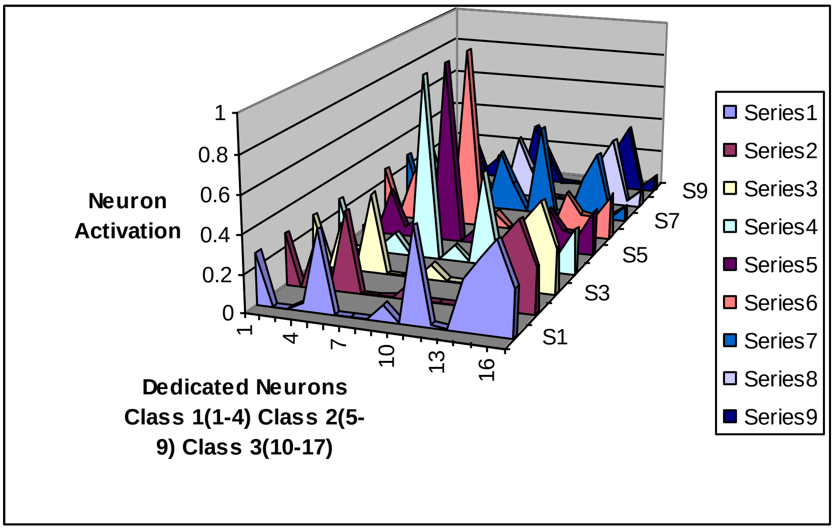
This section presents results obtained when the DNN was tested with combinations of two features.

Graphs 13.26, 13.27 and 13.28 illustrate the results obtained when tested with

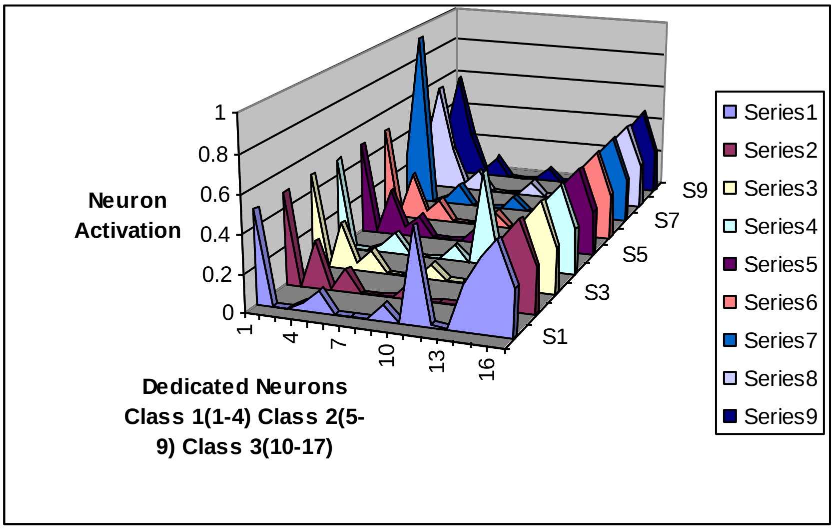
Classes 1 and 2, Classes 2 and 3 and Classes 3 and 1 respectively.



Graph 13.26: Testing the DNN trained using FUSION Method-1 for 9 combinations of Class-1 and Class-2 Features



Graph 13.27: Testing the DNN trained using FUSION Method-1 for 9 combinations of Class-2 and Class-3 Features



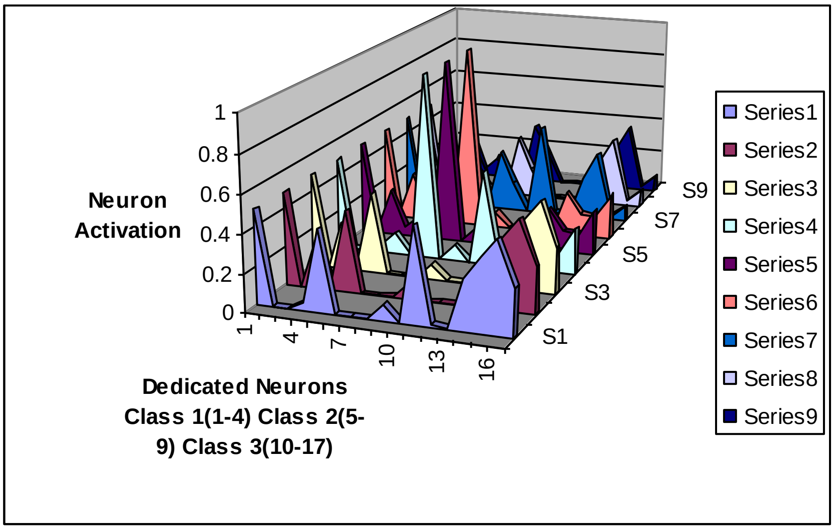
Graph 13.28: Testing the DNN trained using FUSION Method-1 for 9 combinations of Class-1 and Class-3 Features

The Graphs show that the DNN recognises combinations of features as seen by the high activation values of the corresponding dedicated neurons. But, the output values of some of the dedicated neurons of the features being tested were < 0.4. Distinction between various features, especially when combinations of features are presented is difficult when the activation values are < 0.4.

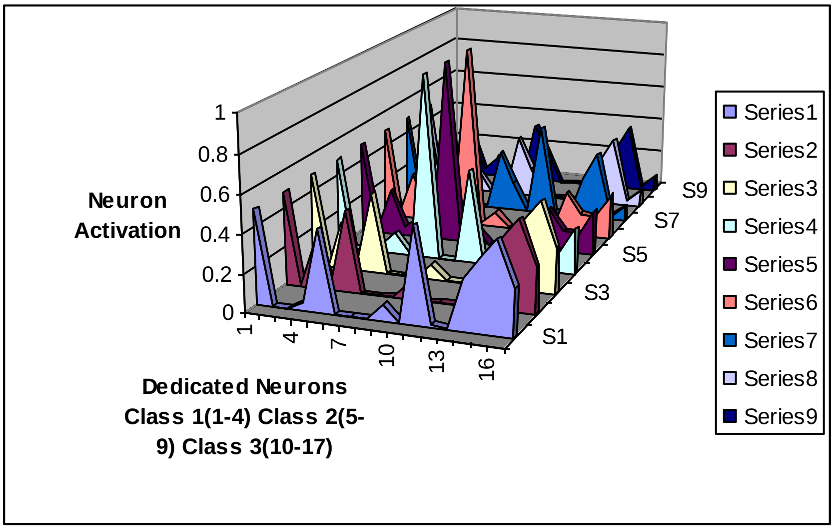
13.5.1.6 TESTING THE DNN FOR COMBINATIONS OF THREE FEATURES

Further testing involved testing the DNN trained using FUSION Method-1 with combinations of 3 features. The combinations of features are given in Chapter 9.

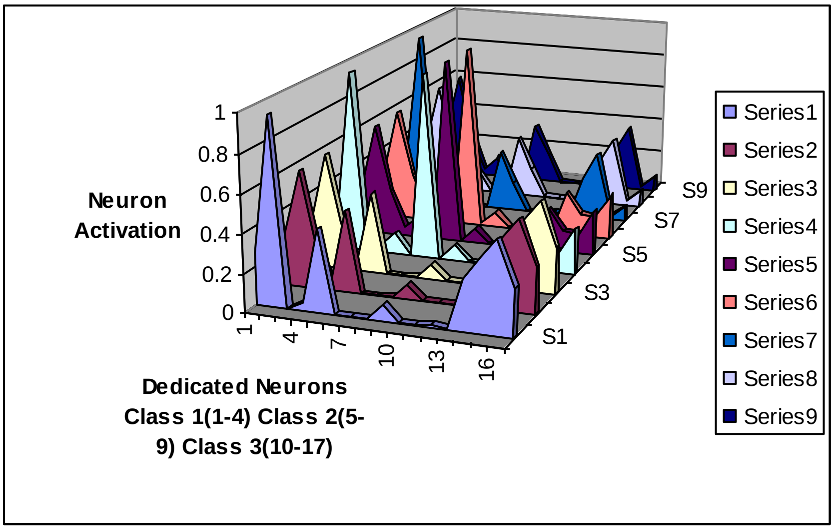
The results obtained are presented in Graphs 13.29, 13.30 and 13.31.



Graph 13.29: Testing the DNN trained using FUSION Method-1 for 9 combinations of Class-1, Class-2 and Class-3 Features



Graph 13.30: Testing the DNN trained using FUSION Method-1 for 9 combinations of Class-1, Class-2 and Class-3 Features



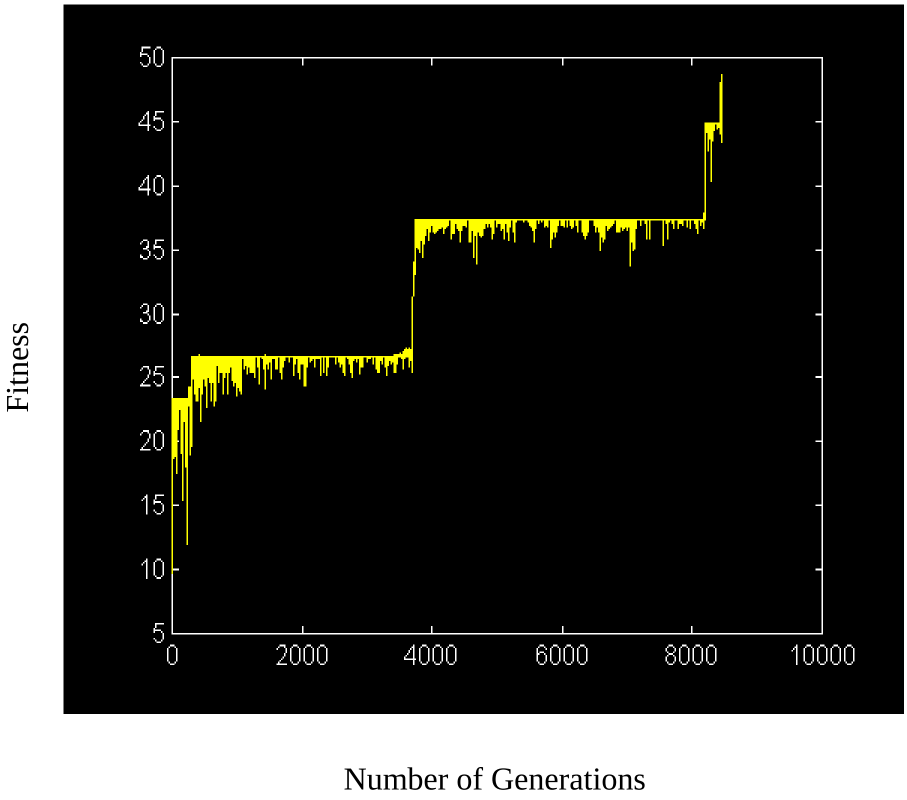
Graph 13.31: Testing the DNN trained using FUSION Method-1 for 9 combinations of Class-1, Class-2 and Class-3 Features

All the graphs shown above show that the activation values of the Class 3 dedicated neurons are < 0.4. Hence, the DNN could not achieve 100% recognition of combinations of all classes.

13.5.2 PHASE-2 DNN TRAINING USING FUSION METHOD-2

In order to improve the distinction between various classes and improving performance of recognition of combinations of features, another method was conceived.

The second method of implementing FUSION involved a condition of achieving exceptional fitness at least 4 consecutive times. The DNN was trained for the features in the training set mentioned in Chapter 9. A fitness of 48.667 was achieved in 8459 generations as shown in the Graph 13.32.

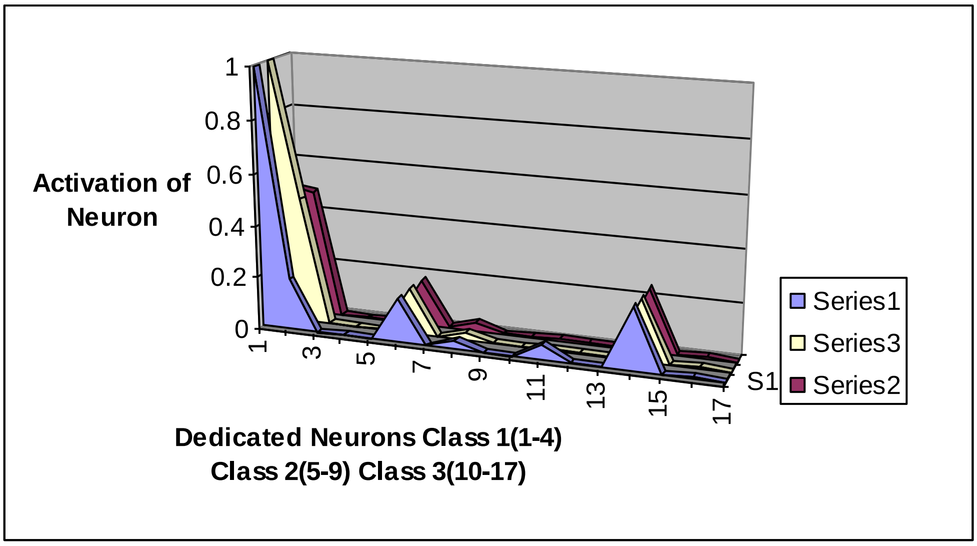


Graph 13.32: Illustration of the Fitness Profile when the DNN was trained using FUSION Method -2

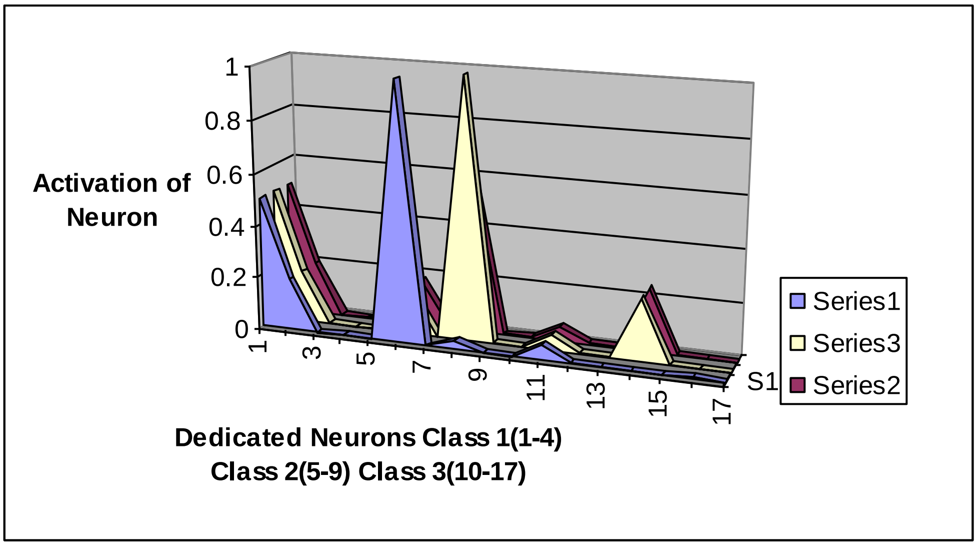
It is evident from the fitness profile that consistent fitness was being achieved due to the repeated high performance required by FUSION. However, such a profile may not be very desirable as the DNN fails to explore varied solutions and hence may be unsuccessful in analysing images consistently.

13.5.2.1 TEST WITH FEATURES FROM THE TRAINING SET

The DNN trained using FUSION Method-2 was tested with features from the training set. Class 1, Class 2 and Class 3 features were tested and the results obtained are illustrated in Graphs 13.33, 13.34 and 13.35.

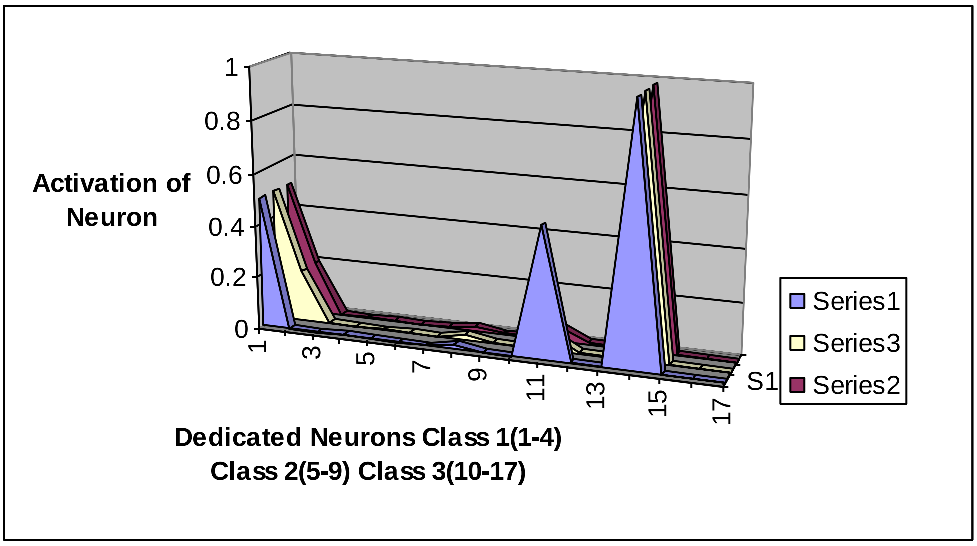


Graph 13.33: Testing the DNN trained using FUSION Method-2 for Class-1 Features from the Training Set



Graph 13.34: Testing the DNN trained using FUSION Method-2 for Class-2 Features from the Training Set

The graphs above show that all Class 1 and Class 2 features were recognised. More important than the recognition of the features is the way the rival neurons show weak activation, which leads to clearer distinction between the various classes.



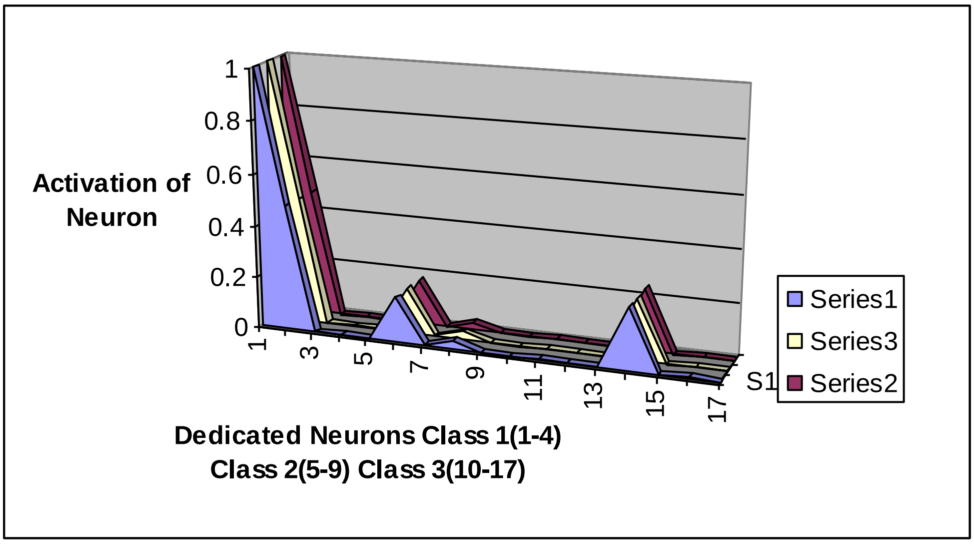
Graph 13.35: Testing the DNN trained using FUSION Method-2 for Class-1 Features from the Training Set

All Class 3 features are successfully recognised with the corresponding dedicated neurons having very high activation values as shown in the Graph 13.35.

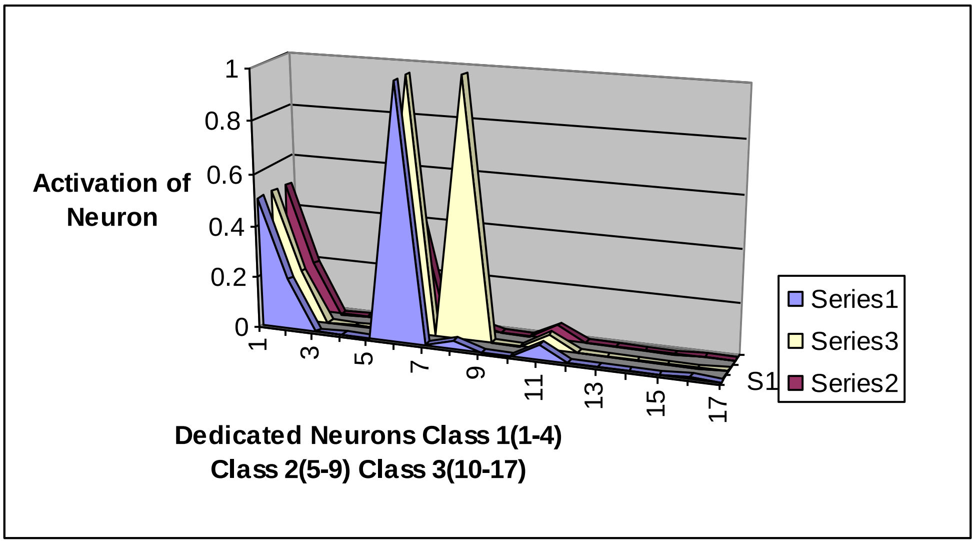
13.5.2.2 TESTING THE DNN USING NEW FEATURES

The DNN trained using FUSION Method-2 was tested with new features. The new features are from the test set given in Chapter 9. All the various variations of the DNN and FUSION are tested using new images to test DNN's ability to generalise.

Graphs 13.36, 13.37, 13.38 show the activation of various dedicated neurons when tested for new features from Class 1, Class 2 and Class 3 features.

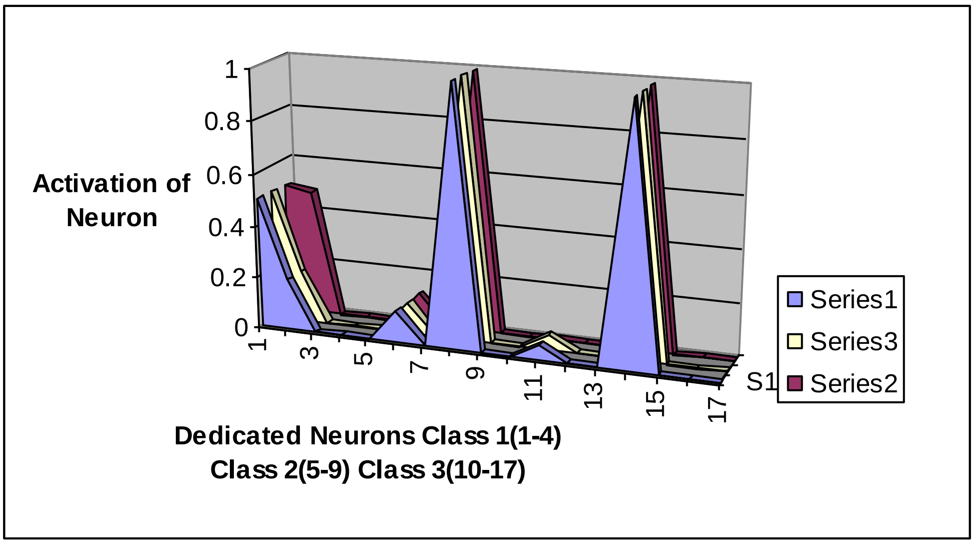


Graph 13.36: Testing the DNN trained using FUSION Method-2 for New Class-1 Features from the Test Set



Graph 13.37: Testing the DNN trained using FUSION Method-2 for New Class-2 Features from the Test Set

All Class 3 features of Class 1 and Class 2 are successfully recognised with the corresponding dedicated neurons having very high activation values as shown in Graphs 13.36 and 13.37.



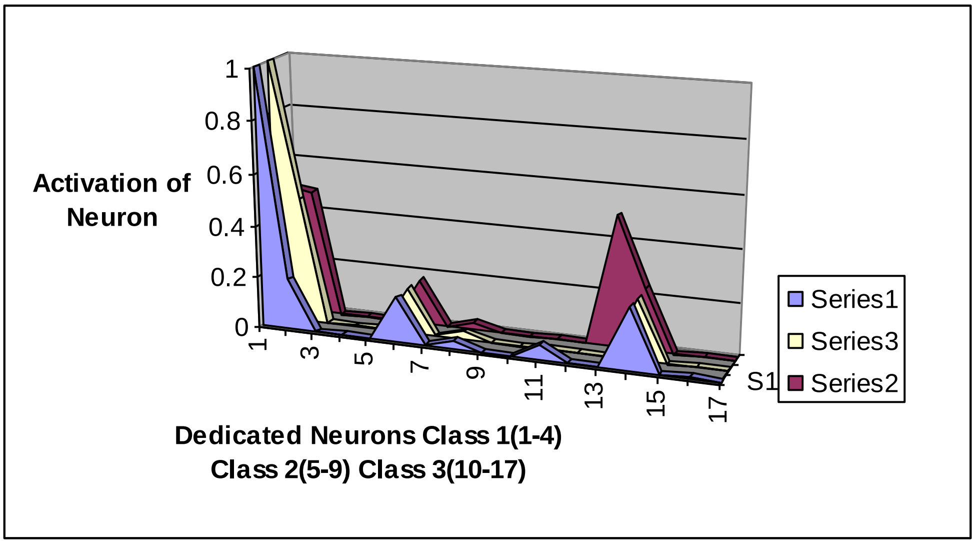
Graph 13.38: Testing the DNN trained using FUSION Method-2 for New Class-3 Features from the Test Set

All Class 3 features are successfully recognised with the corresponding dedicated neurons having very high activation values as shown in the Graph 13.38. However, the dedicated neurons for Class 2 also show very high activation.

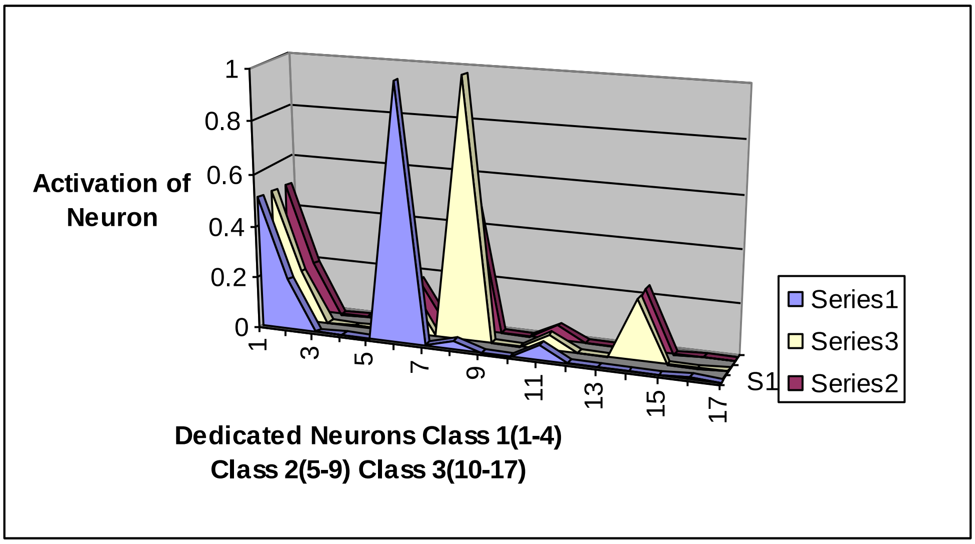
13.5.2.3 TESTING THE DNN USING NOISY FEATURES

The DNN trained using FUSION Method-2 was tested with noisy features from the test set given in Chapter 9.

The results obtained when the network was tested with noisy features belonging to Class 1, Class 2 and Class 3 features are presented in Graphs 13.39, 13.40 and 13.41 respectively.

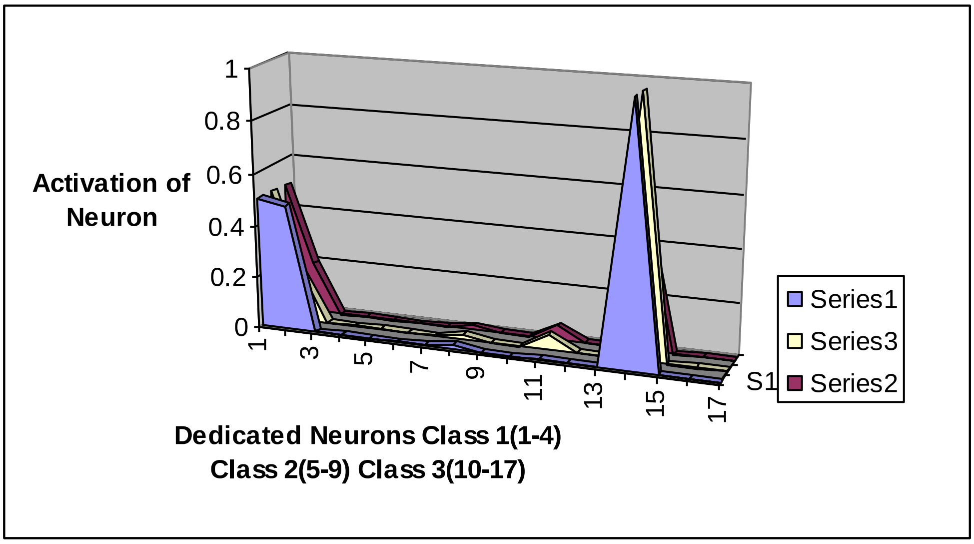


Graph 13.39: Testing the DNN trained using FUSION Method-2 for Noisy Class-1 Features



Graph 13.40: Testing the DNN trained using FUSION Method-2 for Noisy Class-2 Features

All Classes 1 and 2 features are successfully recognised with the corresponding dedicated neurons having very high activation values as shown in the Graph 13.39 and 13.40.



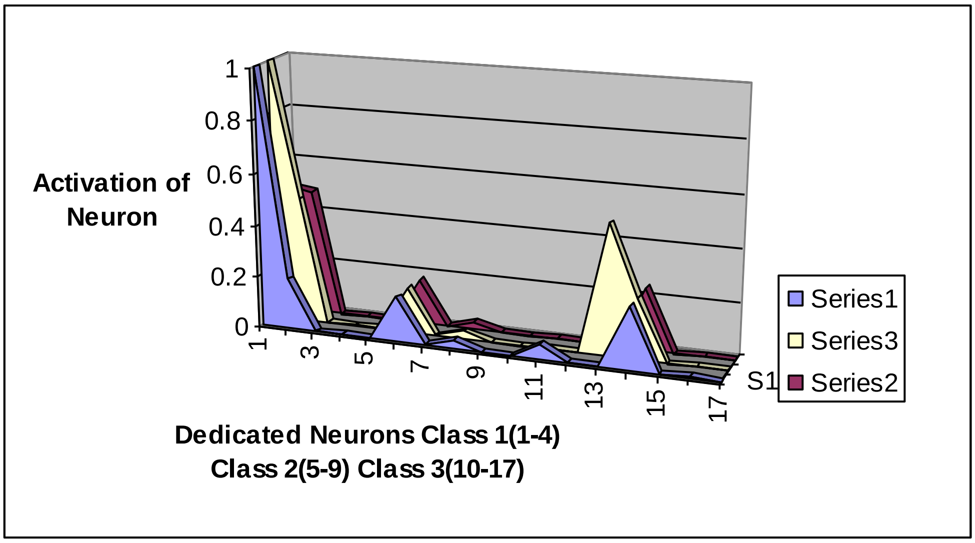
Graph 13.41: Testing the DNN trained using FUSION Method-2 for Noisy Class-3 Features

All Class 3 features are successfully recognised with the corresponding dedicated neurons having very high activation values as shown in the Graph 13.41.

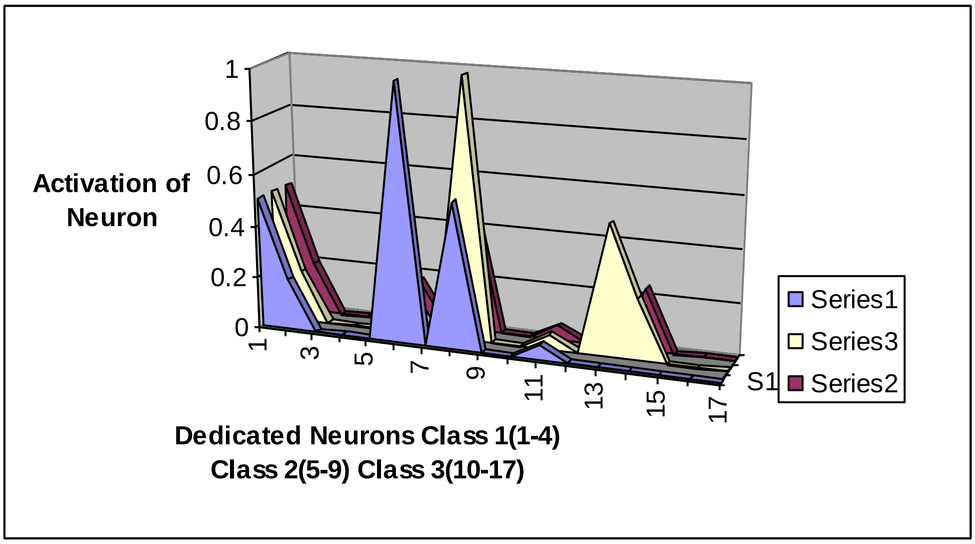
13.5.2.4 TESTING THE DNN WITH FEATURES WITH INCREASED LEVELS OF NOISE

The DNN trained using FUSION Method-2 was then tested with noisy features with increased levels of noise from the test set given in Chapter 9.

The results obtained when the network was tested with noisy features belonging to Class 1, Class 2 and Class 3 features are presented in Graphs 13.42, 13.43 and 13.44 respectively.

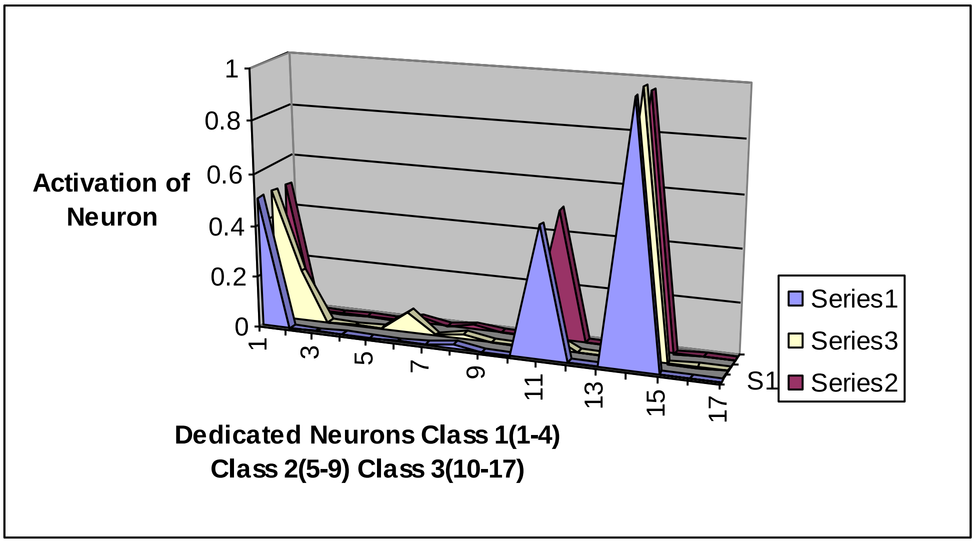


Graph 13.42: Testing the DNN trained using FUSION Method-2 for Class-1 Features with Increased Noise from the Test Set



Graph 13.43: Testing the DNN trained using FUSION Method-2 for Class-2 Features with Increased Noise from the Test Set

All Classes 1 and 2 features are successfully recognised with the corresponding dedicated neurons having very high activation values as shown in the Graph 13.42 and 13.43.



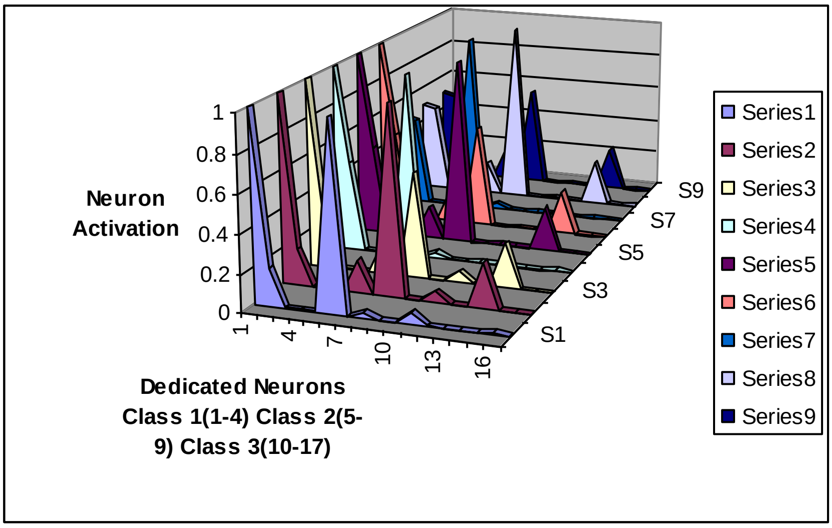
Graph 13.44: Testing the DNN trained using FUSION Method-2 for Class-3 Features with Increased Noise from the Test Set

All Class 3 features are successfully recognised with the corresponding dedicated neurons having very high activation values as shown in the Graph 13.44.

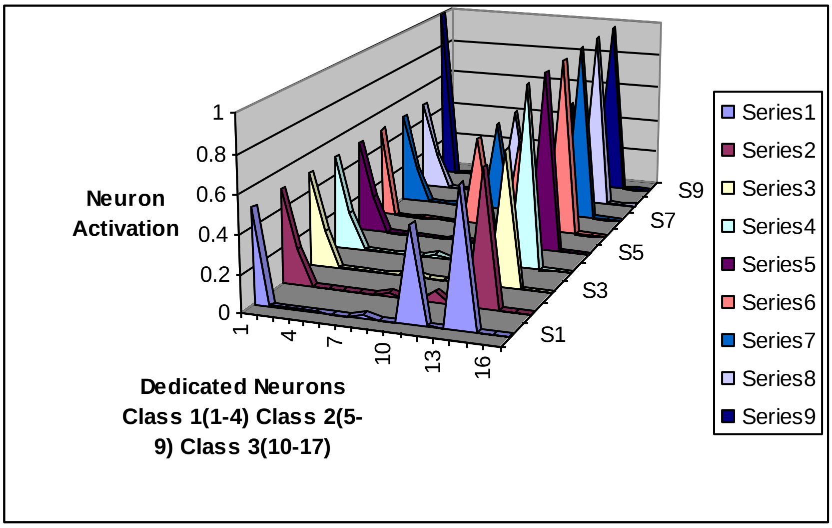
13.5.2.5 TESTING THE DNN WITH COMBINATIONS OF TWO FEATURES

The DNN trained using FUSION Method-2 was tested with combinations of two and three features. This section presents results obtained when combinations of two features were used to test the network.

Graphs 13.45, 13.46 and 13.47 illustrate results obtained when combinations of Classes 1 and 2, Classes 2 and 3 and Classes 3 and 1 were used to test the DNN trained using FUSION Method-2.

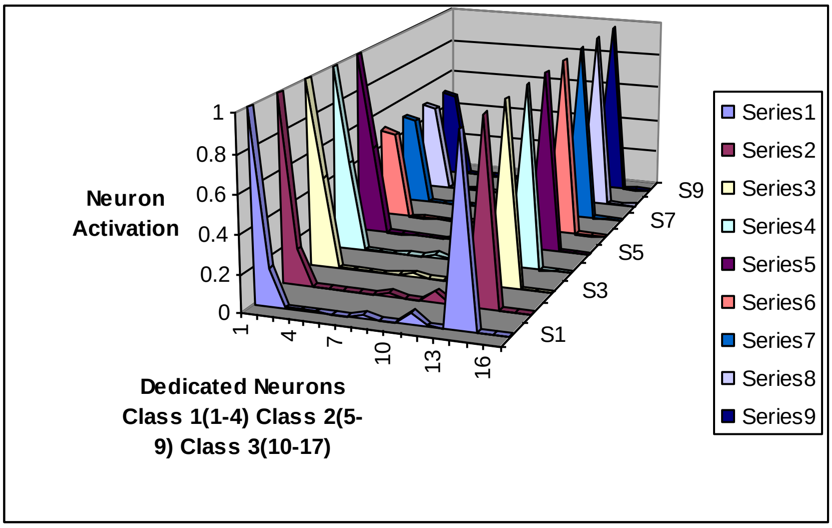


Graph 13.45: Testing the DNN trained using FUSION Method-2 for 9 combinations of Class-1 and Class-2 Features



Graph 13.46: Testing the DNN trained using FUSION Method-2 for 9 combinations of Class-2 and Class-3 Features

Graphs 13.45 and 13.46 shows that the corresponding rival neurons when the DNN trained using Method-2 were tested were very weak. This enhances the networks classification ability.



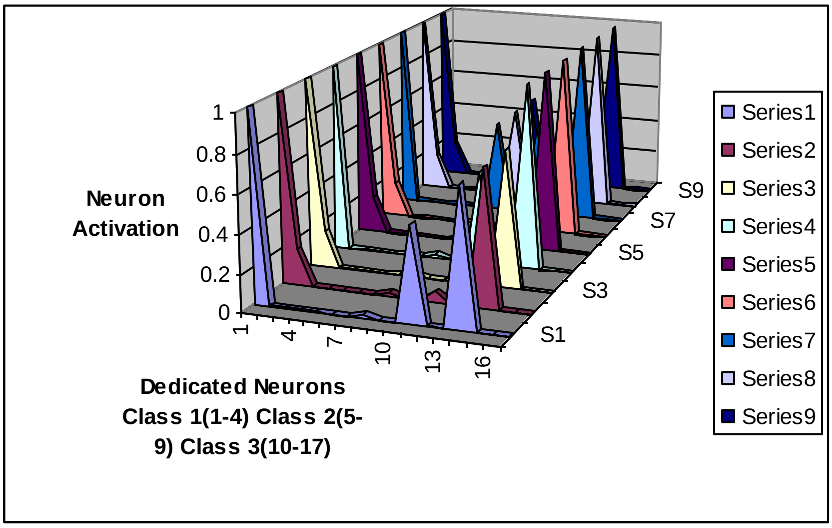
Graph 13.47: Testing the DNN trained using FUSION Method-2 for 9 combinations of Class-1 and Class-3 Features

Graph 13.47 show the recognition of all combinations of Class 1 and Class 3 features. Also, the activation value of all the winning neurons are >0.4.

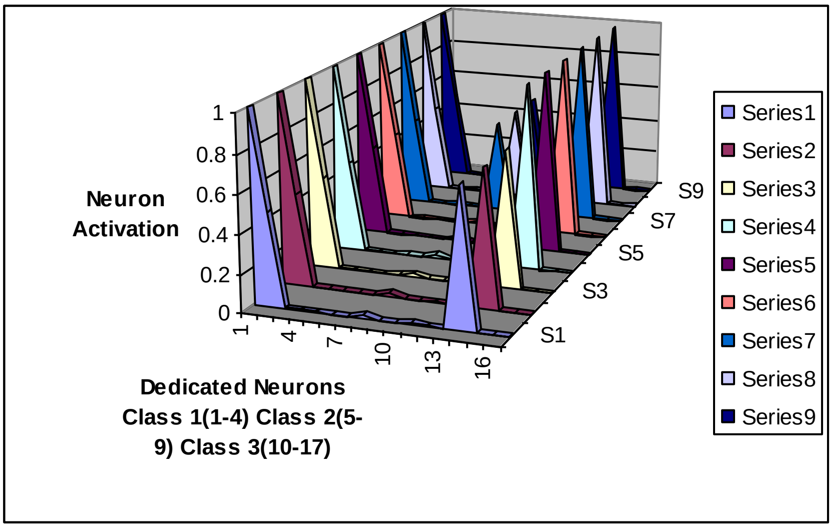
13.5.2.6 TESTING THE DNN WITH COMBINATIONS OF THREE FEATURES

The DNN trained using FUSION Method-2 was tested with combinations of three features as given in Chapter 9.

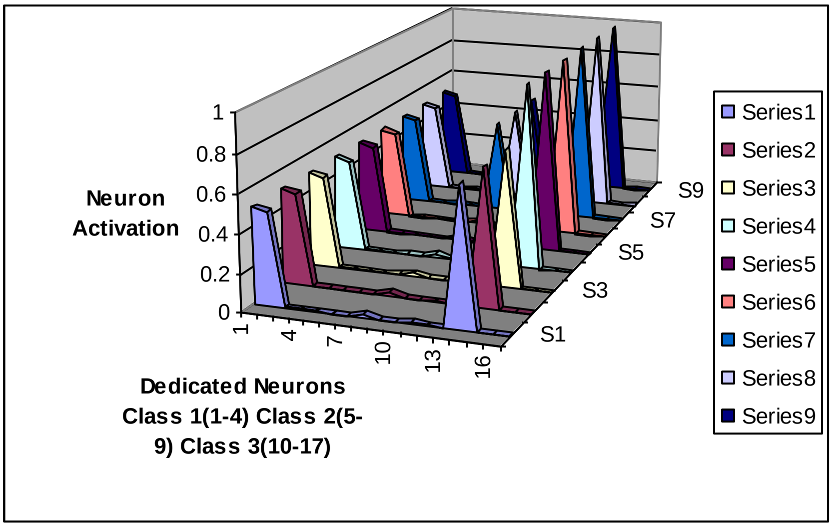
The results obtained are illustrated graphically in Graphs 12.48, 12.48 and 12.50. It can be concluded that the DNN was poor in recognising Class 2 features among the combinations of three features.



Graph 13.48: Testing the DNN trained using FUSION Method-2 for 9 combinations of Class-1, Class-2 and Class-3 Features



Graph 13.49: Testing the DNN trained using FUSION Method-2 for 9 combinations of Class-1, Class-2 and Class-3 Features



Graph 13.50: Testing the DNN trained using FUSION Method-2 for 9 combinations of Class-1, Class-2 and Class-3 Features

13.6 COMPARISON OF FUSION METHODS 1 AND 2

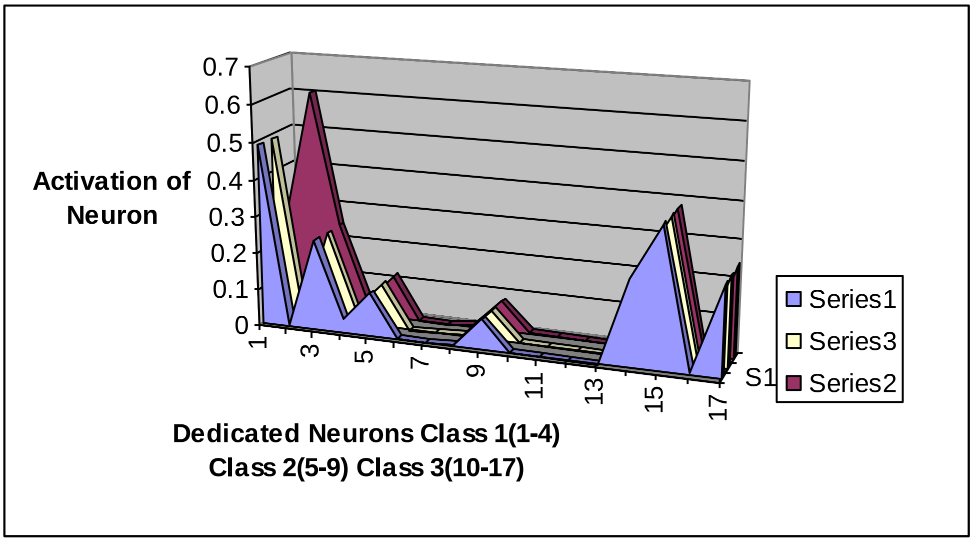
This sections presents a few examples of results obtained and generally compares the achievements of FUSION Methods 1 and 2. All the results are not compared in this section.

A selection of results, which clearly show FUSION Method-2 in action in terms of achieving a clear distinction between various classes of neurons are presented in this section.

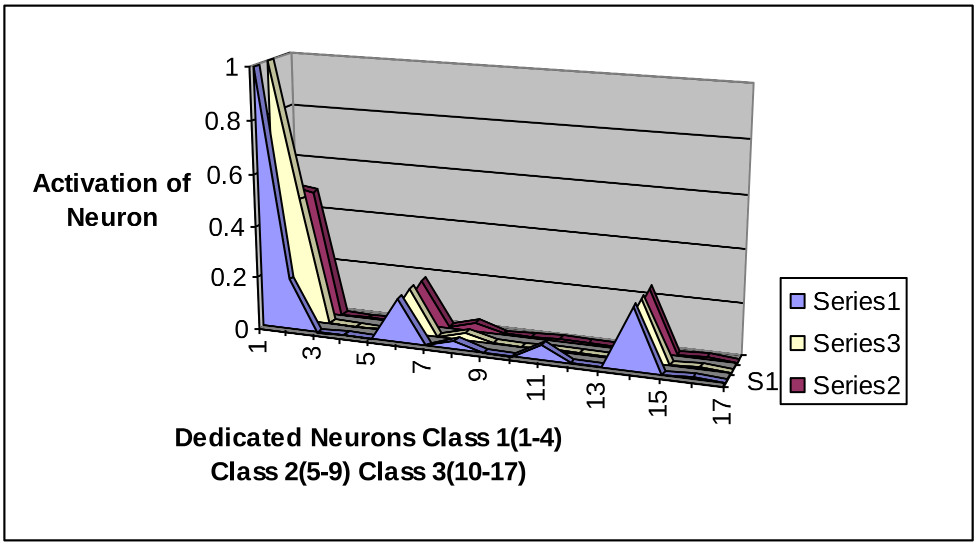
Although, some improvements were achieved, too fine a differentiation can lead to problems with noise tolerance and generalisation. The training method was further improved culminating in the final reward based training algorithm FUSION.

Comparisons from Training Set

The Graphs 13.14 and 13.33 show that FUSION Method-2 has achieved a clear distinction between various classes of dedicated neurons when tested with Class 1 features when the DNN was trained using FUSION Method-2.



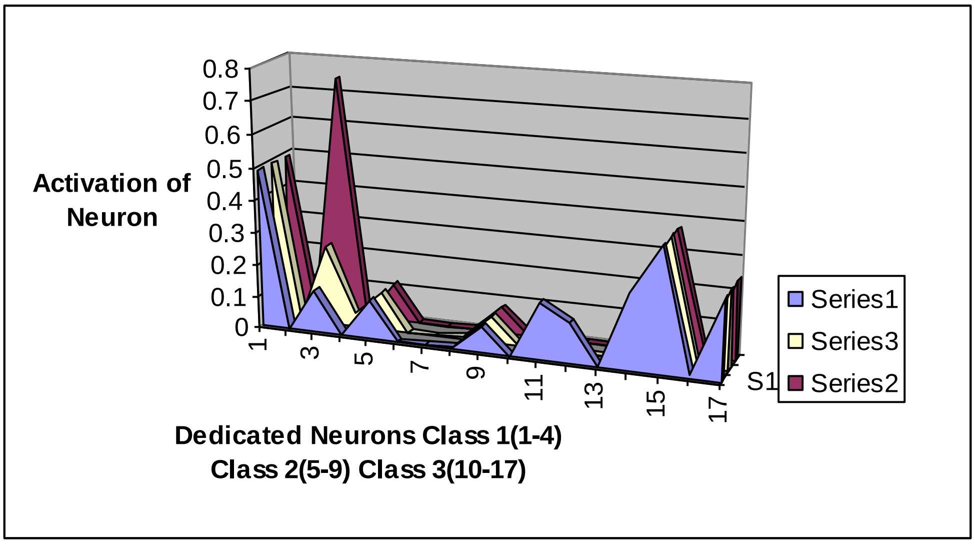
Graph 13.14: Testing the DNN trained using FUSION Method-1 for Class-1 Features from the Training Set



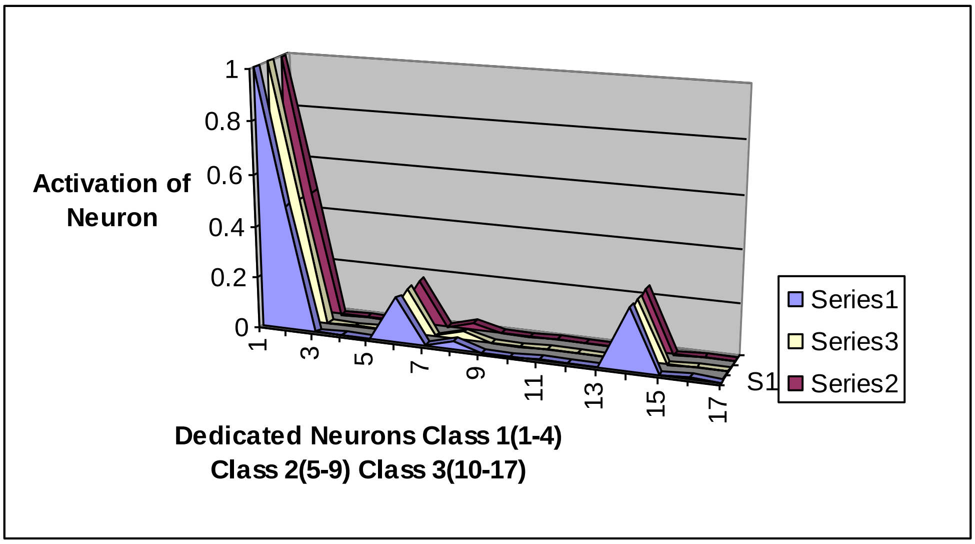
Graph 13.33: Testing the DNN trained using FUSION Method-2 for Class-1 Features from the Training Set

Comparisons from Test Set – New Patterns

The Graphs 13.17 and 13.36 show the consistency in achievement of the distinction between the activation values of various dedicated neurons when tested with new Class 1 features when the DNN was trained using FUSION Method-2.



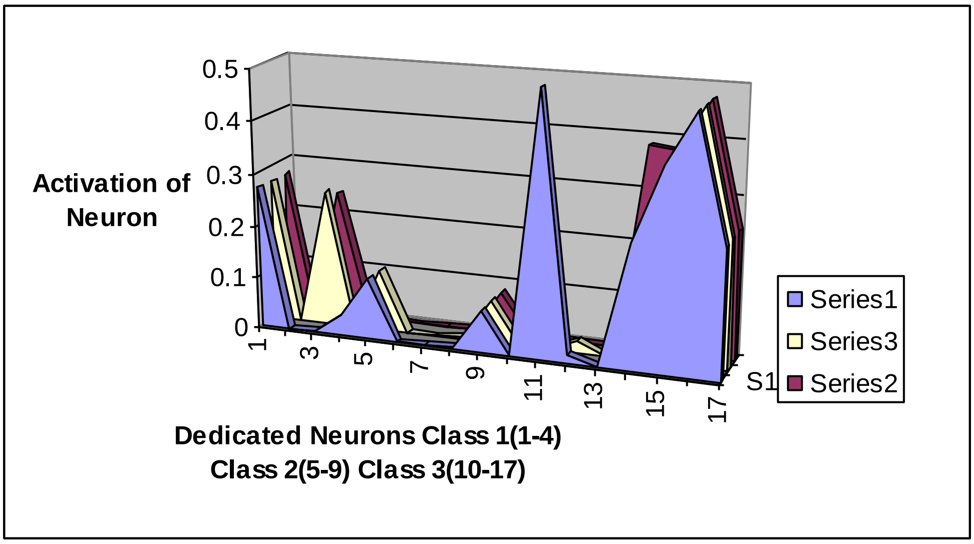
Graph 13.17: Testing the DNN trained using FUSION Method-1 for New Class-1 Features from the Test Set



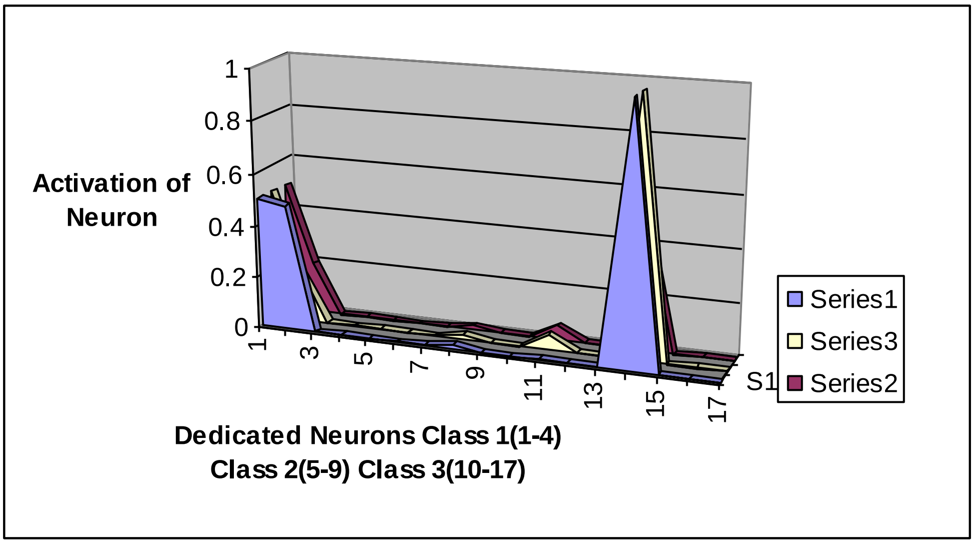
Graph 13.36: Testing the DNN trained using FUSION Method-2 for New Class-1 Features from the Test Set

Comparisons from Test Set-Noisy Data

The Graphs 13.22 and 13.41 continue to show the consistency in achievement of the distinction between the activation values of various dedicated neurons when tested with noisy Class 3 features when the DNN was trained using FUSION Method-2.



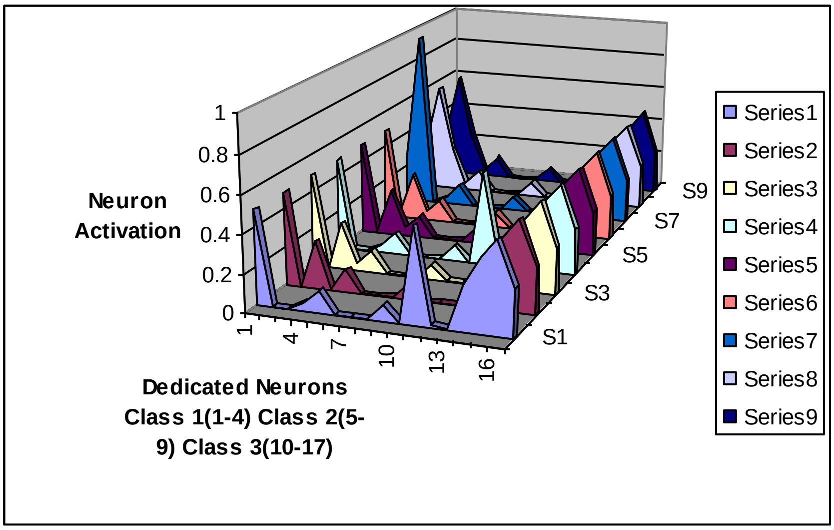
Graph 13.22: Testing the DNN trained using FUSION Method-1 for Noisy Class-3 Features from the Test Set



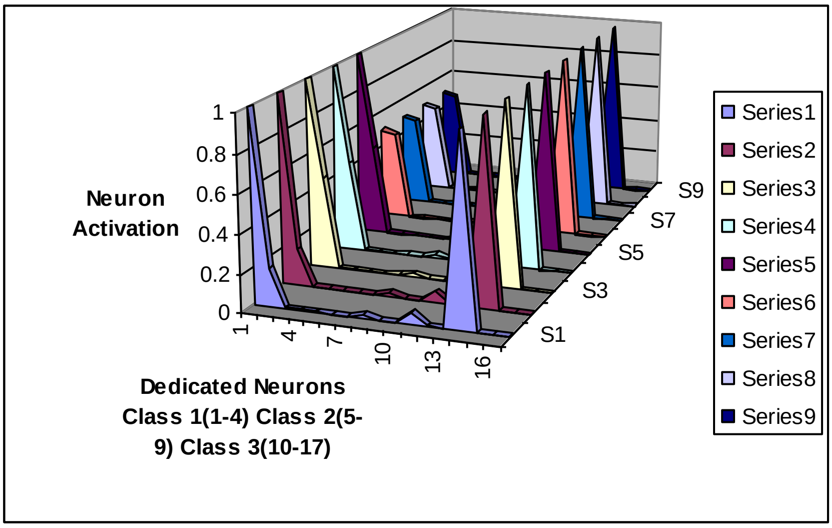
Graph 13.41: Testing the DNN trained using FUSION Method-2 for Noisy Class-3 Features

Comparisons from Test Set-Combination of Features

The Graphs 13.28 and 13.47 show the consistency in achievement of the distinction between the activation values of various dedicated neurons when tested with combinations of Class 1 and Class 3 features when the DNN was trained using FUSION Method-2.



Graph 13.28: Testing the DNN trained using FUSION Method-1 for 9 combinations of Class-1 and Class-3 Features



Graph 13.47: Testing the DNN trained using FUSION Method-2 for 9 Combinations of Class-1 and Class-3 Features

13.7 DISCUSSION AND SUMMARY

Graphs 13.14 -13.31 show the results obtained by testing the DNN trained using FUSION Method-1. Graphs 13.33 – 13.50 illustrate the results achieved when the DNN was trained using FUSION Method-2.

It is evident from the graphs that the insistence of consistent fitness successively over several generations enhanced the Reinforcement Learning of FUSION. There are clear winners among the dedicated neurons for specific Classes of features. Also the activation of the winning neurons is very high.

The DNN's ability to classify new and noisy features improves when trained with FUSION Method-2. Also, the DNN when trained with FUSION Method-2 performs better at recognising combinations of features. The Graphs 13.13 and 13.32 showing the fitness achieved using FUSION Method-1 and FUSION Method-2 show that FUSION Method-2 is able to ascertain more reliable solutions to improve the performance. A selection of graphs from previous sections was presented here to illustrate the comparative performance of FUSION Methods 1& 2.

The design of the novel biologically inspired DNN involved the process of making informed choices based on Biology and traditional ANNs. The tests conducted and results obtained show that the DNN is unique and proficient in recognising features in an image. A discussion of results and ideas for further work are given in Chapter 14.

CHAPTER 14: DISCUSSION OF RESULTS

AND

FURTHER WORK

14.1 INTRODUCTION

The aim of the research was to develop a biologically inspired Artificial Vision System. After thorough study of Biological and Artificial systems, the Distributed Neural Network (DNN) was successfully implemented and tested for its classification abilities. The network architecture, training method, training data set and test data were detailed in the previous chapters. The unique features of the DNN are clear from the results obtained.

The DNN with its simple, elegant architecture along with an efficient and computationally less-taxing training algorithm could classify images like no other traditional pattern recognition or classification technique.

The unique aspects of the DNN were established from the experiments conducted and results obtained. They are:

- The unique architecture of the DNN with its lateral inhibitory connections between repeated modules of neurons.
- The novel training method FUSION that includes the benefits of various aspects of learning methods such as Unsupervised, Supervised and Reinforcement Learning.

- Its ability to recognise combinations of features that the network was trained for, although these combinations are completely new to the network.
- Its ability to process images based on its features rather than process an image as a whole, which is the usual method of image processing.

This chapter discusses the results obtained and gives suggestions for further work.

14.2 DISCUSSION OF RESULTS

The DNN's capabilities are evident from the results presented in previous chapters.

Sparsely Connected DNN

In a FFN, every neuron is connected to every other neuron in the next layer. Hence each pixel's information or every neuron's output information is passed on to the other neurons that it is connected to. This means that in a FFN there is a danger of:

1. Information overload at each neuron
2. Noisy data being amalgamated with other defining information of features in the image

The FFN’s structure does not facilitate filtration or elimination of noise in an image as it acts as a ‘Broadcasting System’, distributing noise globally throughout the system.

One of the structural features of the DNN is its sparsely connected network. Every neuron is not connected to every other neuron in the neighbouring layer.

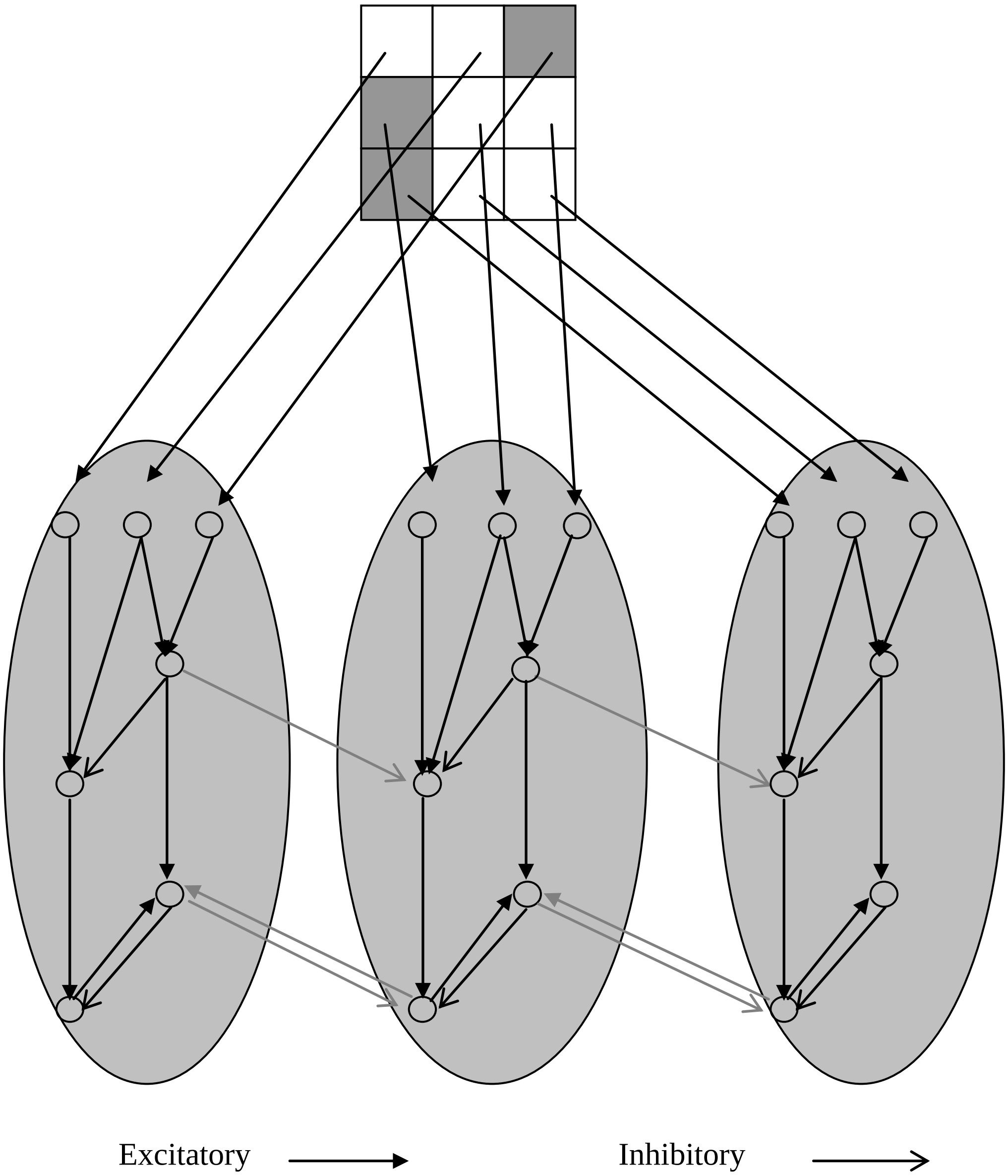


Figure 14.1: Illustration of Input to a Sparsely Connected DNN

The sparsely connected DNN helps eliminate noise and hence is more tolerant to noise than other traditional ANNs.

Inherent modularity

The inherent modularity of the DNN is also illustrated in Figure 14.1. Modules of networks are intertwined together via inhibitory connections. The sideways inhibitory connections facilitate 'handshaking' and enhance communication between various modules of neurons in the DNN.

The interaction between various specialised units of a MNN that is absent in Artificial Systems is very conveniently implemented in the DNN.

The inhibitory connections are very important communication channels that control the information flow.

Partition of Input Space

The distributed structure of the DNN allows the partition of input space. Hence in Figure 14.1 it can be noticed that features in a region tend to get distributed to neighbouring neurons. However, this cannot be claimed for vertical features.

If the input scheme illustrated in Figure 14.1 is adapted, the pixel information related to the vertical feature gets distributed across the DNN. The inhibitory connections again must play a very important role in passing the information of the vertical feature across the DNN.

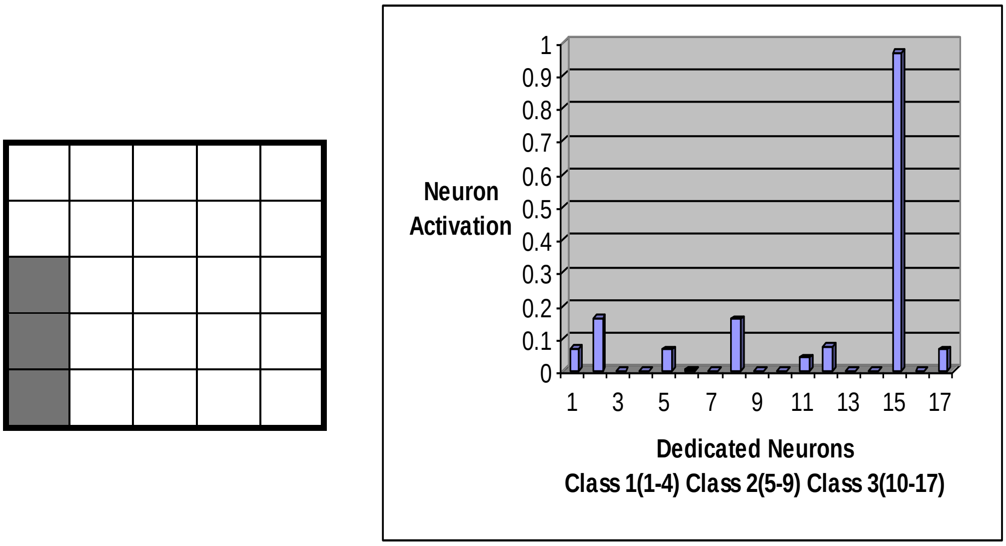


Figure 14.2: Illustration of Successful Passage of Information through the DNN to produce a high activation at Neuron 15

All the information seems to be accumulated at one dedicated neuron.

Usage of Only One Dedicated Neuron

Although 8 dedicated neurons are allocated for classifying Class-3 features, only one dedicated neuron is highly activated as evident in the Figure 14.2.

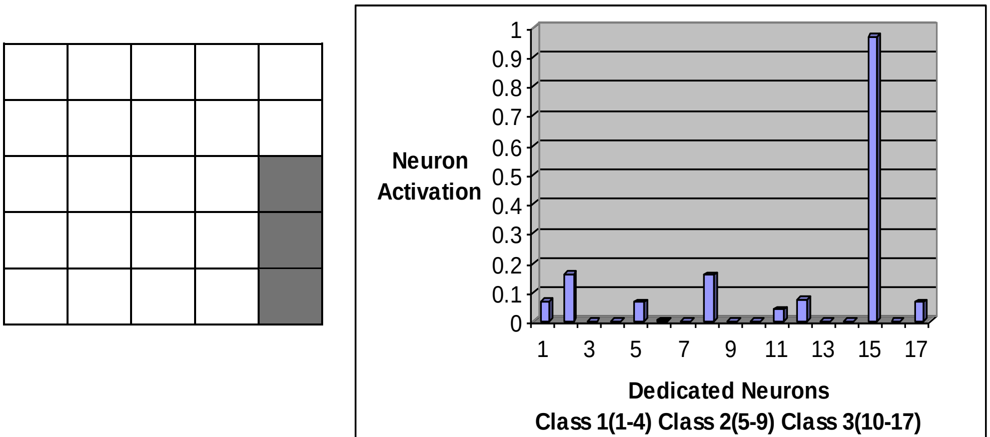


Figure 14.3: Illustration of the Activation of One Dedicated Neuron, although 8 Output Neurons are dedicated for Class-3 Features

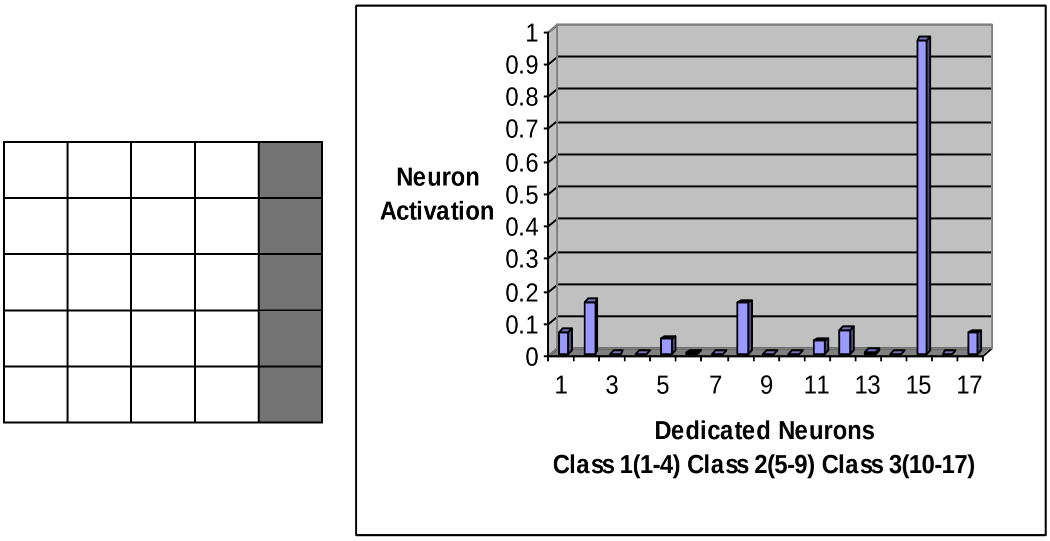


Figure 14.4: Illustration of the Activation of One Dedicated Neuron, although 8 Output Neurons are dedicated for Class-3 Features

Figures 14.2**,** 14.3 and 14.4 show that there is a preference of one dedicated neuron. This phenomenon is observed across other features and other dedicated neurons in the DNN.

The inclination to choose only one dedicated neuron can be viewed as quite strong because it is apparent from the tests conducted in previous chapters. The initial reason to allocate more than one dedicated neuron was to encourage clusters of neurons to be activated for specific features as seen in biology.

This was completely overthrown by the DNN's preference for one dedicated neuron. In fact, when the DNN is unsure of classification there is an irregular eruption of activation among a few neurons rather than having one outright winner.

FUSION and Rewards

Results presented in previous chapters show that the reward scheme in FUSION used to train ANNs is very important.

Ambiguity within the reward scheme and insistence of best performance (fitness) in succession over a few specified generations forces FUSION to explore.

Some results showed that the DNN reached a fitness of 48.667 (where maximum possible fitness is 51.333), but failed to show clear classification. This is due to the uneven distribution of rewards; Class-1 features are allocated up to 70 points whereas, Class-2 and Class-3 features receive a maximum reward of 42 points.

Significance of Genetic Dispositions and Information of Flow

Results show that the initial structure of the DNN is important in the recognition of various features. When all connections in the DNN were made positive, the network was less tolerant to noise. Also, when the order of information flow between dedicated neurons was changed, the DNN was not very efficient in recognising various classes of features.

It can be suggested that the genetic dispositions and order of information flow play a significant role, but they are by no means the only reason for the poor performance of the DNN_R and Homogenous DNN. The other reasons may include the reward mechanism.

Relationship between the Architecture of the DNN and FUSION

The results presented in Chapters 12 and 13 show that the architecture of the DNN was very important. When several other architectural variations of the DNN were trained using the same training method FUSION, the performance of the network did not improve. On the contrary, its performance reduced. This shows that FUSION alone cannot override initial genetic connections between various neurons to achieve the best performance.

Combination of Learning Methods

The architecture of DNN and the training method FUSION allow the incorporation of various learning methods. A combination of aspects of Unsupervised Learning, Supervised Learning, Reinforcement Learning and Fuzzy Logic are used in training the DNN, which has proven to be very useful in overcoming some of the problems in ANNs.

14.3 FURTHER WORK

During this research project, the DNN was developed, designed, trained and implemented. It also helped in identifying the capabilities of the network. Some additional tests may prove very useful in understanding the properties of the DNN that give it its unique classification features.

Suggestions and ideas that could form the basis of further work are listed below.

1. *Training the DNN with Back Propagation:*

   Back Propagation algorithm has been very successful with applications in Pattern Recognition and Classification. The DNN has many common features with Feed Forward Networks and hence BP could be used to train the network. The results would be useful in analysing the network further.

2. *Understanding the Importance of the Genetic Dispositions:*

From the results obtained when the direction of flow of information between dedicated output neurons was reversed, it was clear that some innate hard-wired structures or paths are important to the DNN's operation.

The importance of the genetic features could be understood to some extent, if the DNN was a fully connected network and trained in the same manner with FUSION. One of the questions that needs to be addressed, to find out about the hard-wired nature of the network, is whether the network will end up with the designed structure of the DNN, when a fully connected DNN is trained using FUSION, by setting some connections to zero and thus discarding them and making some others inhibitory connections by evolving a connection strength with a negative value.

3. Biologically Accurate Model:

Several assumptions were made, while developing the DNN. If the discarded biological information is incorporated, the efficiency of the network may improve. It is worth testing whether this is the case.

4. *Biologically Inspired Hybrid Model:*

Other biological vision systems have features that provide the system with its unique ability to process visual stimuli. A hybrid model, developed from a combination of features from different biological systems, could be beneficial. It would be interesting to study such systems.

5. *Test for Various Applications:*

   The DNN could be used for various applications such as Face Recognition and Image Processing. Any change in the image is very important in such applications. The DNN's unique ability to process images based on features may give it an edge over other techniques.

6. *Test for Whole Images:*

   Further work could be involved in testing the DNN for its ability to recognise whole images. In this research project the DNN was trained for individual features and tested to identify known features from an image.

7. *Modularity and Lateral Inhibition:*

   The structure of the DNN is modular. It consists of modules of neurons that are repeated across the network. These modules are inter-connected by Lateral Inhibitory connections. Lateral inhibition is also observed in insects' visual systems. More experiments that could evaluate the structure of the network with emphasis on the lateral connections could form the basis of further work.

8. *Forecasting/ Prediction Applications:*

The DNN is very good for recognition of visual data or images. The network's ability in other Pattern Recognition techniques such as Forecasting and Prediction are yet to be established.

9. *Scaling of Network:*

Another aspect that could be explored is the scaling of the network. The affect of the reward mechanism and the training on various sizes of networks can give further insight into the architectural features of the DNN.

10. *Universal Algorithm:*

The DNN was not tested with a universal algorithm for unknown number of features, which can be taken up for further study and testing of the DNN. The algorithms presented in this research were specific to a known number of features present in the grid.

Results from further work on the DNN, as suggested above could help in understanding more about Artificial and Biological Neural Networks, which may in turn pave the path to "Real Intelligence".

CHAPTER 15: CONCLUSIONS

15.1 INTRODUCTION

The research was successful in the development of the Distributed Neural Network (DNN), a system inspired by the toad's visual system, based on Artificial Neural Networks, and trained using a novel training method called FUSION.

This chapter revisits the objectives of the research and evaluates the contribution to the state of the art.

15.2 ACHIEVEMENTS OF OBJECTIVES

One way of assessing the success of the project is to consider the original objectives laid out at the beginning of the project, in light of what was achieved. The objectives of this research were explained in Chapter 1. They are:

1. Literature Survey:

A literature search was conducted into both Biological and Artificial Vision Systems. This was very important to understand the limitations of current systems and explore possible avenues to overcome some of the problems. Chapters 3 and 4 give details of the Artificial Systems and Chapters 6 focuses on the biological research.

Also an understanding of the background to the research, the Artificial Nervous System (ANS) was necessary to evaluate the requirements of the system. The details of the ANS are given in Chapter 2.

2. Study of Biological Sensory Systems:

Study of the biological vision systems included the study of human, insect and toad visual systems. Most of the information published in this field was about the development of the eye itself. Although there were some interesting findings and theories of the operation of various visual systems, very little is known about the processes that occur in the brain when analysing visual stimuli.

The toad's visual system was found interesting and useful in view of the aim of the research. Detailed explanation of the toad's visual system is given in Chapter 6, as a foundation towards understanding and designing the Artificial Vision System.

3. Investigation of Neural Network based Pattern Recognition Techniques:

Several Neural Network based Pattern Recognition techniques such as Back Propagation, Kohonen's Self Organising Maps and Competitive Networks were studied. Other techniques reviewed included Statistical Pattern Recognition Techniques. A literature search of such systems is given in Chapter 3.

4. Implementation of a Vision System using Parallel Modular Networks:

Modularity has several operational advantages in both Engineering and Biological Systems. It is one of the essential aspects of the ANS and was a major consideration in developing the Artificial Visual System during this research.

Following a review of Artificial Neural Networks, experiments were conducted using traditional ANN methods, which included Modular Networks. The results obtained and the tests conducted are given in Chapter 5.

5. Comparison of results against published work:

Experiments were conducted to establish the limitations of popular network architectures and training methods. Several simple experiments using Modular Neural Networks and Back Propagation were conducted, which revealed some drawbacks. The details of the experiments conducted, the results obtained and a discussion of the results is given in Chapter 5.

6. Implementation of an Evolutionary Vision System:

A unique Intelligent Vision System was developed based on the toad's visual system. It was implemented using Artificial Neural Networks and trained using FUSION. The novel system is called the Distributed Neural Network (DNN).

The DNN is unique in its operation. Unlike popular architectures, the DNN recognises combinations of features in an image, even though the network is trained only for the features and not for any combinations of these features.

The details of the design of DNN, its operation and tests conducted are given in Chapters 7, 8, 9, 10, 11, 12, 13 and 14.

7. Interface of the Sensory System to the lower levels of the ANS:

The Sensory System or Vision System is modular, implemented using ANNs and trained using Evolutionary Algorithms, which are the foundation blocks of the ANS.

The model developed was based on the sensori-motor connections in the toad that help the animal to differentiate between various objects in its environment and elicit appropriate behaviours. Hence, the outputs from the network are directly associated with the generation of appropriate behaviours, which constitute the lower levels of the ANS model.

8. Testing the network with real data input from a camera:

The DNN was tested with digital images of well-defined objects that formed obstacles and representations of the seabed. The results obtained are given in Chapter 11.

All the objectives mentioned have been met and the aim of the research undertaken has been achieved.

15.3 CONTRIBUTION TO STATE OF THE ART

The important contributions to the state of the art are:

- The development of a novel, biologically inspired Artificial Neural Network called the DNN with many unique properties.
- The formulation and testing of a new and unique training algorithm called the FUSION, which utilises aspects of Unsupervised, Supervised and Reinforcement Learning.
- The study of information flow and parameter setup within the DNN.
- The systematic study of alternative models and their limitations.
- The study of these limitations in the context of a robotic vision system.

The main contribution to the state of the art was the development and implementation of a novel Artificial Neural Network based on the toad's visual system called the Distributed Neural Network.

Several unique aspects of the DNN have been identified, which cannot be found in any other system.

The DNN not only recognises the features that it was trained for, but also recognises combinations of features, which cannot be achieved by any other network or system. This solves one of the major problems associated with so called Adaptive Systems employed in changing environments, as the system will be able to recognise any known features in the visual scene and need not have all the information of the scene at all times. The contribution to the state of the art may be quite important in the field of Biometrics, where current systems fail for the slightest change in the image (for example, Face Recognition)

The network is very robust and efficient in its operation, as it can handle very high compression ratios of the images as illustrated when digital images were used to test the DNN. The network is able to recognise features although a lot of information may be lost due to compression. This may contribute in finding new image processing tools and may be helpful in analysing satellite images and it might prove useful in areas such as Video Conferencing.

The results that prove that genetic dispositions and the direction of flow of information (in other words, the network architecture) are important even in Artificial Systems. This research showed that although learning can cause many changes in the system, some vital features of the system are maintained.

The final contribution is the number of experiments and tests conducted during this research both in traditional ANNs and DNNs, which are unique to this project and have never been published before.

15.4 FINAL WORD ON VISION SYSTEMS

This research has shown that improvements in Artificial Vision Systems can be possible, when engineers draw inspiration from biology. As the ultimate aim is to develop truly intelligent systems, it is obvious to look to biological systems for better understanding of the functionality of the neurons, their neuronal structures and the whole aspect of Intelligence. The research also shows that this may result in the development of simple yet powerful structures.

Only more co-operations between engineers and biologists will help solve the communication gap between these two fields.

There are no foolproof efficient vision systems available in the market that could recognise images in uncontrolled environments. Even in controlled environments, the systems fail if there is variation in the image.

The DNN is a small contribution towards developing better systems that may revolutionise the Artificial Vision Systems currently available.

REFERENCES

[1] C. MacLeod, G. M. Maxwell and D. McMinn, “A Framework for Evolution of an Animat Nervous System,” EUREL-European Advanced Robotics Systems Development: Mobile Robotics, Leiria, Portugal, Sep 1998.

[2] D. McMinn, “Using Evolutionary Networks to design Hierarchical Animat Nervous System,” PhD Thesis, The Robert Gordon University, Aberdeen, 2001.

[3] P. J. Simmons and D. Young, “Nerve Cells and Animal Behaviour,” Cambridge University Press, UK, 1999.

[4] X. Yao and Y Liu, “Towards designing Artificial Neural Networks by Evolution”, Applied Mathematics and Computation, Volume 91, Issue 1, pp. 83-90, April 1998

[5] D. E. Moriarty, A. C. Schultz, J. J.Grefenstette, “Evolutionary Algorithms for Reinforcement Learning,” Journal of Artificial Intelligence Research, Vol. 11, pp. 241-276, 1999.

[6] H. J. Chiel and R. D. Beer, “The Brain has a Body: Adaptive Behaviour Emerges from Interactions of Nervous System, Body and Environment,” Trends in Neurosciences, Vol. 20, Issue 12, pp. 553-557, 1997.

[7] H. Cruse, T. Kindermann, M. Schumm, J. Dean and J. Schmitz, “Walknet - A Biologically Inspired Network to Control Six-legged Walking,” Neural Networks, Vol. 11, Issues 7-8, pp. 1435-1447, 1998.

[8] N. Francheschini, “Vision, Flies, Neurons and Robots,” IEEE International Congress of Robotics and Automation, Nagoya, pp. 21-27, May 1995.

[9] E. J. Gonzalen-Galan, S. R. Cruz-Ramirez, M. J. Seelinger, J. J. Cervantes-Sanchez, “An Efficient Multi-camera Multi-target Scheme for the 3-D Control of Robots using Uncalibrated Vision”, Robotics and Computer-Integrated Manufacturing, Vol. 19, Issue 5, pp. 387-400, 2003.

[10] R C Arkin, “Reactive Robotic Systems” in “The Handbook of Brain Theory and Neural Networks”, edited by M A Arbib, MIT Press, pp. 793-796, 1998.

[11] J Albert, “Computational Modelling of an Early Evolutionary Stage of the Nervous System,” Biosystems, Vol. 54, Issues 1-2, pp. 77-90, 1999.

[12] R. I. Damper, R. L. B. French and T. W. Scutt, “ARBIB: An Autonomous Robot based on Inspirations from Biology” Robotics and Autonomous Systems, Vol. 31, Issue 4, pp. 247-274, 2000.

[13] P. Husbands, A. Philippides and T. Smith, “Volume Signalling in Real and Robot Nervous Systems”, Theory in Biosciences, Vol. 120, Issues 3-4, pp. 253-269, 2001.

[14] C. MacLeod, G. M. Maxwell and D. McMinn, “A Framework for Evolution of an Animat Nervous System,” EUREL-European Advanced Robotics Systems Development: Mobile Robotics, Leiria, Portugal, Sep 1998.

[15] D. McMinn, “Using Evolutionary Networks to design Hierarchical Animat Nervous System,” PhD Thesis, The Robert Gordon University, Aberdeen, 2001.

[16] Gao Daqi and Ji Yan, “Classification Methodologies of Multiplayer Perceptrons with Sigmoid Activation Functions” Pattern Recognition, Volume 38, Issue 10, pp.1469-1482, October 2005.

[17] J. Agirrebeitia, R. Avilés, I. F. de Bustos and G. Ajuria, “Inverse Position Problem in Highly Redundant Multibody Systems in Environments with Obstacles”, Mechanism and Machine Theory, Volume 38, Issue 11, pp. 1215-1235, 2003.

[18] M. J. L. Boada, R. Barber and M. A. Salichs, “Visual Approach Skill for a Mobile Robot using Learning and Fusion of Simple Skills”, Robotics and Autonomous Systems, Volume 38, Issues 3-4, pp. 157-170, March 2002.

[19] M. J. Aitkenhead and A. J. S. McDonald, “A Neural Network based Obstacle-Navigation Animat in a Virtual Environment”, Engineering Applications of Artificial Intelligence, Volume 15, Issues 3-4, pp. 229-239, 2002.

[20] S. Srinivasan, D.P. Mital and S. Haque, “A Novel Solution for Maze Traversal Problems using Artificial Neural Networks”, Computers & Electrical Engineering, Volume 30, Issue 8, pp. 563-572, 2004.

[21] G. N. Marichal, L. Acosta, L. Moreno, J. A. Méndez, J. J. Rodrigo and M. Sigut, “Obstacle Avoidance for a Mobile Robot: A Neuro-Fuzzy Approach”, Fuzzy Sets and Systems, Volume 124, Issue 2, pp. 171-179, December 2001.

[22] David P. M. Northmore, “A Network of Spiking Neurons develops Sensorimotor Mechanisms while Guiding Behavior”, Neurocomputing, Volumes 58-60, pp. 1057-1063, 2004.

[23] Fumiya Iida, “Biologically Inspired Visual Odometer for Navigation of a Flying Robot”, Robotics and Autonomous Systems, Volume 44, Issues 3-4, pp. 201-208, September 2003.

[24] M. V. Srinivasan, J. S. Chahl, K. Weber, S. Venkatesh, M. G. Nagle and S. W. Zhang, “Robot Navigation Inspired by Principles of Insect Vision”, Robotics and Autonomous Systems, Volume 26, Issues 2-3, pp. 203-216, February 1999.

[25] Cornelius Weber, Stefan Wermter and Alexandros Zochios, “Robot Docking with Neural Vision and Reinforcement”, Knowledge-Based Systems, Volume 17, Issues 2-4, pp. 165-172, May 2004.

[26] Iñaki Monasterio, Elena Lazkano, Iñaki Rañó and Basilo Sierra, “Learning to Traverse Doors using Visual Information”, Mathematics and Computers in Simulation, Volume 60, Issues 3-5, pp. 347-356, September 2002.

[27] Kuang-Hsiung Chen and Wen-Hsiang Tsai, “Vision-based Obstacle Detection and Avoidance for Autonomous Land Vehicle Navigation in Outdoor Roads”, Automation in Construction, Volume 10, Issue 1, pp. 1-25,2000.

[28] P. Wasserman, “Neural Computing-Theory and Practice”, Van Nostrand Reinhold, 1989.

[29] M. Bosque, “Understanding 99% of Artificial Neural Networks: Introduction and Tricks”, Writers Club Press, 2002.

[30] D. Skapura, “Building Neural Networks” ACM Press, NY, 1996.

[31] Kidong Lee, David Booth and Pervaiz Alam, “A comparison of Supervised and Unsupervised Neural Networks in Predicting Bankruptcy of Korean firms Expert Systems with Applications”, Volume 29, Issue 1, pp. 1-16 , July 2005.

[32] R. D. Reed, R. J. Marks II, “Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks”, Bradford Books, 1999.

[33] Ana M. Fonseca, José L. Biscaya, João Aires-de-Sousa and Ana M. Lobo, “Geographical Classification of Crude Oils by Kohonen Self-Organizing Maps”, Analytica Chimica Acta, Volume 556, Issue 2, pp. 374-382, 25 January 2006.

[34] A. Carling, “Introducing Neural Networks”, Sigma Press, Wimslow, UK, pp. 155-160,1993.

[35] Kazunori Iwata, Kazushi Ikeda and Hideaki Sakai, “The Asymptotic Equipartition Property in Reinforcement Learning and its Relation to Return Maximization”, Neural Networks, Volume 19, Issue 1, pp. 62-75, January 2006.

[36] Alex v.E. Conradie and Chris Aldrich, “Development of Neurocontrollers with Evolutionary Reinforcement Learning”, Computers & Chemical Engineering, Volume 30, Issue 1, 15, pp. 1-17, 2005.

[37] Ido Erev and Greg Barron, “On Adaptation, Maximization, and Reinforcement Learning Among Cognitive Strategies”, Psychological Review, Volume 112, Issue 4, pp. 912-931 October 2005.

[38] Anke Meyer-Bäse, Peter Gruber, Fabian Theis and Simon Foo, “Blind Source Separation based on Self-Organizing Neural Network”, Engineering Applications of Artificial Intelligence, Volume 19, Issue 3, pp. 305-311, April 2006.

[39] Ana M. Fonseca, José L. Biscaya, João Aires-de-Sousa and Ana M. Lobo , “Geographical Classification of Crude Oils by Kohonen Self-Organizing Maps” Analytica Chimica Acta, Volume 556, pp. 374-382, 2006.

[40] F. Sebelius, L. Eriksson, H. Holmberg, A. Levinsson, G. Lundborg, N. Danielsen, J. Schouenborg, C. Balkenius, T. Laurell and L. Montelius, “Classification of Motor Commands using a Modified Self-Organising Feature Map”, Medical Engineering & Physics, Volume 27, Issue 5, pp. 403-413, 2005.

[41] H. Ritter, “Self-Organising Feature Maps: Kohonen Maps” in “The Handbook of Brain Theory and Neural Networks”, Ed. M. A. Arbib, MIT Press, pp. 846-851, 1998.

[42] K Fukushima, “Neocognitron: A Model of Visual Pattern Recognition”, in “The Handbook of Brain Theory and Neural Networks”, Ed. M. A. Arbib, MIT Press, pp. 613-617, 1998.

[43] Kunihiko Fukushima , “Neocognitron Capable of Incremental Learning” Neural Networks, Volume 17, Issue 1, pp. 37-46, January 2004.

[44] Kunihiko Fukushima, “Neocognitron for Handwritten Digit Recognition”, Neurocomputing, Volume 51, pp. 161-180 April 2003.

[45] A. D. Kulkarni, “Computer Vision and Fuzzy-Neural Systems”, Prentice Hall PTR, pp. 281-290, 2001.

[46] Daoqiang Zhang, Songcan Chen and Zhi-Hua Zhou , “Learning the Kernel Parameters in Kernel Minimum Distance Classifier”, Pattern Recognition, Volume 39, Issue 1, pp. 133-135, January 2006.

[47] Hyungchul Kim and Chanan Singh, “Power System Probabilistic Security Assessment using Bayes Classifier”, Electric Power Systems Research, Volume 74, Issue 1, pp. 157 –165.

[48] Suman K. Mitra, Te-Won Lee and Michael Goldbaum, “Bayesian Network based Sequential Inference for Diagnosis of Diseases from Retinal Images”, Pattern Recognition Letters, Volume 26, Issue 4, pp. 459-470, March 2005.

[49] A. D. Kulkarni, “Computer Vision and Fuzzy-Neural Systems”, Prentice Hall PTR, pp. 351-360, 2001.

[50] P. J. Werbos, “Backpropagation: Basics and New Developments” in “The Handbook of Brain Theory and Neural Networks”, Ed. M. A. Arbib, MIT Press, pp. 134-139, 1998.

[51] P. Frasconi, M. Gori and A.Tesi, “Successes and Failures of Backpropagation: A theoretical Investigation”, in “Progress in Neural Networks”, Ed. O.Omidvar, Ablex Publishing, pp. 205-242, 1997.

[52] N. Intrator, “Competitive Learning”, in “The Handbook of Brain Theory and Neural Networks”, Ed. M. A. Arbib, MIT Press, pp. 220-223, 1998.

[53] A. Carling, “Introducing Neural Networks”, Sigma Press, Wimslow, UK, pp.160, 1993.

[54] P. S. Churchland and T. J. Sejnowski, "The Computational Brain", MIT Press, pp. 104, 1992.

[55] P. S. Churchland and T. J. Sejnowski, "The Computational Brain", MIT Press, pp. 97, 1992.

[56] Enzo Grossi, Giulia Massini, Massimo Buscema, Rita Savarè and Guido Maurelli, "Two Different Alzheimer Diseases in Men and Women: Clues from Advanced Neural Networks and Artificial Intelligence"Gender Medicine, Volume 2, Issue 2, pp. 106-117, June 2005.

[57] M.-C. Belanger, F. Auclair, L. Bertrand and R. Marchand, "The Early Neuronal Organisation Predicts the Path Followed by Some Major Axonal Bundles in the Embryonic Brainstem", Neuroscience, Vol. 78, No. 1, pp. 259-270, 1997.

[58] E. Ronco and P. Gawthrop, "Modular Neural Networks: A State of the Art", http://www.mech.gla.ac.uk/Control/Publications/Rabstracts/../Reports/csc95026.ps, Technical Report: CSC-95026, Centre for System and Control, Glasgow University, UK, May 1995.

[59] M. J . Aitkenhead and A. J. S. McDonald, "Complex Environments, Complex Behaviour**",** Engineering Applications of Artificial Intelligence, Vol. 17, Issue 6, pp. 611-62, 2004.

[60] M. N. Dailey and G. W. Cotrell, "Organisation of Face and Object Recognition in Modular Neural Network Models", Neural Networks, Vol.12, pp. 1053-1073, 1999.

[61] Sonia Dollfus, Annick Razafimandimby, Pascal Delamillieure, Perrine

Brazo, Marc Joliot, Bernard Mazoyer and Nathalie Tzourio-Mazoyer, "A Typical Hemispheric Specialization for Language in Right-handed Schizophrenia Patients", Biological Psychiatry, Volume 57, Issue 9, pp.1020-1028, May 2005.

[62] Timothy J. Andrews and Denis Schluppeck, "Neural Responses to Mooney Images reveal a Modular Representation of Faces in Human Visual Cortex", NeuroImage, Volume 21, Issue 1, pp. 91-98, January 2004.

[63] Eiji Hoshi, "Functional Specialization within the Dorsolateral Prefrontal Cortex: A review of Anatomical and Physiological Studies of Non-Human Primates", Neuroscience Research, Volume 54, Issue 2, pp. 73-84, February 2006.

[64] Goulven Josse, Pierre-Yves Hervé, Fabrice Crivello, Bernard Mazoyer and Nathalie Tzourio-Mazoyer, "Hemispheric Specialization for Language: Brain Volume Matters", Brain Research, Volume 1068, Issue 1, pp. 184-193, 12 January 2006.

[65] Dara S. Manoach, Nathan White, Kristen A. Lindgren, Stephan Heckers, Michael J. Coleman, Stéphanie Dubal, Donald C. Goff and Philip S. Holzman, "Intact Hemispheric Specialization for Spatial and Shape Working Memory in Schizophrenia", Schizophrenia Research, Volume 78, Issue 1, pp. 1-12, October 2005.

[66] W. M. Cowan, "The Development of the Brain", Scientific American, Vol. 241, pp. 112-133, 1979.

[67] E. R. Kandel, "Small Systems of Neurons", Scientific American, Vol. 241, pp. 67-76, September 1979.

[68] C. Wright, P. Sham, R. M. Murray, D. R. Weinberger and E. T. Bullmore, “Genetic Contributions to Regional Variability in Human Brain Structure: Methods and Preliminary Results” NeuroImage, Volume 17, Issue 1, pp. 256-271, September 2002.

[68] Dennis D. M. O'Leary and Yasushi Nakagawa, “Patterning Centers, Regulatory Genes and Extrinsic Mechanisms Controlling Arealization of the Neocortex” Current Opinion in Neurobiology, Volume 12, Issue 1, pp. 14-25, February 2002.

[70] Kimihiro Nakamura, Tatsuhide Oga, Tomohisa Okada, Norihiro Sadato, Yoshihiro Takayama, Taeko Wydell, Yoshiharu Yonekura and Hidenao Fukuyama, “Hemispheric asymmetry emerges at distinct parts of the occipitotemporal cortex for objects, logograms and phonograms: A functional MRI study” NeuroImage, Volume 28, Issue 3, pp. 521-528, November 2005.

[71] M. J. Farah, “Is Face Recognition ‘Special’? Evidence from Neuropsychology”, Behavioural Brain Research, Vol. 76, pp. 181-189, 1996.

[72] T. J. Andrews and D. Schluppeck, “Neural Responses to Mooney Images Reveal a Modular Representation of Faces in Human Visual Cortex” NeuroImage, Vol. 21, Issue 1, pp. 91-98, 2004.

[73] T. A. Polk and M. J. Farah, “The Neural Development and Organisation of Letter Recognition: Evidence from Functional Neuroimaging , Computational Modelling and Behavioural Studies”, Proc. Natl. Acad. Sci, USA, Vol. 95, pp. 847-852, Feb 1998.

[74] G. K. Humphrey, L. A. Symons, A. M. Herbert and M. A. Goodale, "A Neurological Dissociation between Shape from Shading and Shape from Edges", Behavioural Brain Research, Vol. 76, pp. 117-125, 1996.

[75] H. Parthasarathy, "Mind Rhythms", New Scientist, pp. , 30th Oct 1999.

[76] R. Cotterill, "Enchanted Looms: Conscious Networks in Brains and Computers", Cambridge University Press, pp. 183-238, 1998.

[77] A. Ishai, L. G. Ungerleider, A. Martin, J. L. Schouten and J. V. Haxby, "Distributed Representation of Objects in the Human Ventral Visual Pathway", Proc. Natl. Acad. Sci, USA, Vol 96, pp: 9379- 9384, Aug 1999.

[78] A. Sahraie, L. Weiskrantz, J. L. Barbur, A. Simmons, S.C. R. Williams and M. J. Brammer, "Pattern of Neuronal Activity Associated with Conscious and Unconscious Processing of Visual Signals", Proc. Natl. Acad. Sci, USA, Vol. 94, pp. 9406-9411, Aug 1997.

[79] S. K. Sharma, G. W. Irwin, M. O. Tokhi and S. F. McLoone, "Learning Soft Computing Control Strategies in a Modular Neural Network Architecture", Engineering Applications of Artificial Intelligence, Vol. 16, Issues 5-6, pp: 395-405, August-September 2003.

[80] P. Gallinari, "Modular Neural Network systems, Training of", in "The Handbook of Brain theory and Neural Networks", Ed. M.A. Arbib, MIT Press, pp. 582-585, 1998.

[81] Sheng-Uei Guan, Qi Yinan, Syn Kiat Tan and Shanchun Li, "Output Partitioning of Neural Networks", Neurocomputing, Volume 68, pp. 38-53, October 2005.

[82] İnan Güler and Elif Derya Übeyli, "A Modified Mixture of Experts Network Structure for ECG beats Classification with Diverse Features", Engineering Applications of Artificial Intelligence, Volume 18, Issue 7, pp. 845-856, October 2005.

[83] Inan Güler and Elif Derya Übeyli , "A Mixture of Experts Network Structure for Modelling Doppler Ultrasound Blood Flow Signals", Computers in Biology and Medicine, Volume 35, Issue 7,pp. 565-582, October 2005.

[84] S-B. Cho, "Fuzzy Aggregation of Modular Neural Networks with Ordered Weighted Averaging Operators", International Journal of Approximate Reasoning, Vol. 13, pp: 359-375, 1995.

[85] M. I. Jordon and R. A. Jacobs, "Modular and Hierarchical Learning Systems", in "The Handbook of Brain theory and Neural Networks", Ed. M.A. Arbib, MIT Press, pp. 579-582, 1998.

[86] P. A. Estévez, H. Paugam-Moisy, D. Puzenat and M. Ugarte, "A Scalable Parallel Algorithm for Training a Hierarchical Mixture of Neural Experts" Parallel Computing, Vol. 28, Issue 6, pp:861-891, 2002.

[87] B-L. Lu and M. Ito, "Task Decomposition and Module Combination based on Class Relations: A Modular Neural Network for Pattern Classification", IEEE Transactions on Neural Networks, Vol. 10, No. 5, Sep 1999.

[88] I-S. Oh and C. Y. Suen, "A Class-Modular Feedforward Neural Network for Handwriting Recognition", Pattern Recognition, Volume 35, Issue 1, pp. 229-244, 2002.

[89] G. Auda and M. Kamel, “CMNN: Cooperative Modular Neural Networks for Pattern Recognition”, Pattern Recognition Letters, Vol. 18, pp. 1391-1398, 1997.

[90] Katja Gaschler, “One Person, One Neuron?”, Scientific American: Mind, February 2006.

[91] M. A. Ali (Ed), “Photoreception and Vision in Invertebrates”, Plenum Press, 1984, pp. 1-5.

[92] M. W. Stricberger, “Evolution”, Jones and Bartlett Sudbury, Massachusetts, USA, 2nd Edition, pp. 37, 1996.

[93] J. N. A. Hooper and R. W. M. van Soest (Ed), “Systemata Porifera: A Guide to the Classification of Sponges”, Kluwer Academic, Plenum Publishers, 2002.

[94] M. Logue, “Sea Jellies”, Childs World, 2004.

[95] A. Fournier, “Photoreceptors and Photosensitivity in Platyhelminthes”, in M. A. Ali (Ed), “Photoreception and Vision in Invertebrates”, Plenum Press, 1984, pp. 217-239.

[96] A. H. Burr, “Evolution of Eyes and Photoreceptor Organelles in the Lower Phyla”, in M. A. Ali (Ed), “Photoreception and Vision in Invertebrates”, Plenum Press, 1984, pp. 131-178.

[97] M. F. Land, “Molluscs”, in M. A. Ali (Ed), “Photoreception and Vision in Invertebrates”, Plenum Press, 1984, pp. 699-725.

[98] M. Verger-Bocquet, “Photoreception Et Vision Chez Les Annelides”, in M. A. Ali (Ed), “Photoreception and Vision in Invertebrates”, Plenum Press, 1984, pp. 289-333.

[99] D. G. Stavenga and R. C. Hardie (Eds), “Facets of Vision”, Springer-Verlag, 1989.

[100] H. Autrum, R. Jung and W. R. Loewenstein, “Handbook of Sensory Physiology”, Springer-Verlag, 1979.

[101] M. H. Pirenne, “Vision and the Eye”, Chapman and Hall Ltd, 1967.

[102] M. Yoshida. N. Takasu and S. Tamotsu, “Photoreception in Echinoderms”, in M. A. Ali (Ed), “Photoreception and Vision in Invertebrates”, Plenum Press, 1984, pp. 743-772.

[103] N. J. Dominy, J-C. Svenning and W. H. Li, “Historical Contingency in the Evolution of Primate Colour Vision”, Journal of Human Evolution, Vol. 44, Issue 1, pp. 25-45, 2003.

[104] Eric J. Warrant, Almut Kelber, Anna Gislén, Birgit Greiner, Willi Ribi and William T. Wcislo, “Nocturnal Vision and Landmark Orientation in a Tropical Halictid Bee”, Current Biology, Volume 14, Issue 15, pp. 1309-1318, 2004.

[105] R. F. Chapman, “The Insects, Structure and Function”, London: Hodder and Stoughton, pp. 515-572, 1980.

[106] S. R. Shaw, “Early Visual Processing in Insects”, Journal of Experimental Biology, Vol. 112, pp. 225-251, 1954.

[107] N. Francheschini, J. M. Pichon and C. Blanes, “From Insect Vision to Robot Vision”, Phil. Transactions of Royal Society, London, Vol. 337, pp. 283-294, 1992.

[108] N. Francheschini, “Sequence Discriminating Neural Network in the Eye of the fly”, in F. H. Eeckman (Ed), “Analysis and Modelling of Neural Systems”, Kluwer Academic Publishers, 1992.

[109] Fulvio Domini, Corrado Caudek and Hadley Tassinari , “6-2 Stereo and Motion Information are not Independently Processed by the Visual System”, Vision Research, Volume 46, Issue 11, pp. 1707-1723, 2006.

[110] William H.A. Beaudot and Kathy T. Mullen, “Orientation Discrimination in Human Vision: Psychophysics and Modeling”, Vision Research, Volume 46, Issues 1-2, pp. 26-46, 2006.

[111] J. G. Nicholls, A. R. Martin and B. G. Wallace, “From Neuron to Brain: A Cellular and Molecular Approach to the Function of the Nervous System”, Sinauer Associates Inc, pp. 561-650, 1992.

[112] R. Cotterill, “Enchanted Looms: Conscious Networks in Brains and Computers”, Cambridge University Press, pp. 183-238, 1998.

[113] J. Y. Lettvin, H. Maturana, W. S. McCulloch and W. H. Pitts, “What the Frog’s Eye tells the Frog Brain”, Proceedings IRE, Vol 47, 1959, pp. 1940-1951.

[114] J.-P Ewert, “Concepts in Vertebrate Neuroethology” Animal Behaviour, Vol 33, pp. 1-29, 1985.

[115] J.-P Ewert, “Neuroethology of Releasing Mechanisms: Prey-catching in toads,” Behavioural and Brain Sciences Vol 10(3), pp. 337 - 367,1987.

[116] P. J. Simmons and D. Young, “Nerve Cells and Animal Behaviour,” Cambridge University Press, UK, pp. 1999.

[117] D. L. Wang, M. A. Arbib, “How does the Toad’s Visual System discriminate different worm-like stimuli”, Biological Cybernetics, Vol 64, pp. 251-261, 1991.

[118] J. A. Carr, C. L. Brown, R. Mansouri, S. Venkatesan, “Neuropeptides

and Amphibian Prey-catching Behaviour", Comparative Biochemistry and Physiology, Part B: Biochemistry and Molecular Biology, Vol 132, Issue 1, pp. 151-162, 2002.

[119] D. Ingle, "Selective Choice between Double Prey Objects by Frogs", Brain, Behaviour and Evolution, Vol 7, pp. 127-144, 1973.

[120] J.-P. Ewert and V. Seelan, "Neurobiologie und System-theorie eines Visuellen Muster-Erkennungsmechanismus bei Krote", Kybernetik, Vol. 14, pp. 167-183, 1974.

[121] S. Fingerling, J. -P. Ewert, R. Menzel and F. Pfeiffer, "From the Toad to a Robot: Implementation of Neurobiological Principles of Object Discrimination in Neural Engineering", Naturwissenschaften, 1993, Vol 80, pp. 321-324.

[122] Michael Arbib, "Levels of Modelling of Mechanisms of Visually Guided Behaviour", Behavioural and Brain Sciences, Cambridge University Press, Vol 10, pp. 407-465, 1987.

[123] Michael Arbib, Jim-Shih Liaw, "Sensorimotor Transformations in the Worlds of Frogs and Robots", Artificial Intelligence, Vol 72, pp. 53-79, 1995.

[124] Alberto Cobas, Michael Arbib, "Prey-catching and Predator-avoidance in Frog and Toad: Defining the Schemas", Journal of theoretical Biology, Vol 157, pp. 271-304, 1992.

[125] Bruce A. Draper, Robert T. Collins, John Brolio, Allen R. Hanson, Edward M. Riseman, "The Schema System", International Journal of Computer Vision, Vol 2, pp. 209-250, 1989.

[126] Kimihiro Nishio, Hiroo Yonezu, Masahiro Ohtani, Hitoshi Yamada and Yuzo Furukawa, “Analog Network for Detection of Approaching Object with Shape Recognition based on Visual Systems of Lower Animals”, Proceedings of ICONIP, Vol , pp. , 2002.

[127] Kimihiro Nishio, Hiroo Yonezu, Masahiro Ohtani, Hitoshi Yamada and Yuzo Furukawa, “Analog Metal-Oxide-Semiconductor Integrated Circuit Implementation of Approach Detection with Simple Shape Recognition based on Visual Systems of Lower Animals”, Optical Review, Vol 10, No. 2, pp. 96-105, 2003.

[128] J.-P. Ewert, H. Buxbaum-Conradi, M. Glagow, A. Rottgen, E. Schurg-Pfeiffer and W. W. Scwippert, “Forebrain and Midbrain Structures Involved in Prey-Catching Behaviour of Toads: Stimulus-Response Mediating Circuits and their Modulating Loops”, European Journal of Morphology, Vol. 37, No 1, pp. 111- 115, 1999.

[129] J.-P. Ewert, “Neural Correlates of Key Stimulus and Releasing Mechanism: A Case Study and Two Concepts”, Trends in Neurosciences, Vol. 20, No. 8, pp. 332-339, 1997.

[130] M. Glagow, J.-P. Ewert, “Apomorphine-induced Suppression of Prey Oriented Turning in Toads is Correlated with Activity Changes in Pretectum and Tectum: [14C]2DG studies and single cell recordings”, Neuroscience Letters, Vol. 220, pp. 215-218, 1996.

[131] S. Wachowitz and J. -P. Ewert, “A Key by which the Toad’s Visual System gets Access to the Domain of Prey”, Physiology and Behaviour, Vol. 60, No. 3, pp. 877-887, 1996.

[132] J.-P. Ewert, “Neural Mechanisms of Prey-catching and Avoidance Behaviour in the Toad”, Brain, Behaviour and Evolution, Vol. 3, No. 1-4, pp. 36-56, 1970.

[133] Sung-Kwun Oh and W. Pedryz, “Self-organising Polynomial Neural Networks based on Polynomial and Fuzzy Polynomial Neurons: Analysis and Design”, Fuzzy Systems and Systems, Vol. 142, Issue 2, pp. 163-198, 2004.

[134] R. S. Barto and A. G. Sutton, “Reinforcement Learning – An Introduction”, MIT Press, 1998.

[135] M. Mitchell, “An Introduction to Genetic Algorithms”, MIT Press, 1998.

[136] H. –G. Beyer and H. –P. Schwefel, “Evolution Strategies – A Comprehensive Introduction”, Natural Computing, Vol. 1, No. 1, pp: 3-52, 2002.

[137] J. Koza, “Genetic Programming: Routine Human- Competetive Machine Intelligence”, Kluwer Academic Publishers, 2003.

[138] D. E. Moriarty, A. C. Schultz, J. J. Grefenstette, “Evolutionary Algorithms for Reinforcement Learning”, Journal of Artificial Intelligence Research, Vol. 11, pp: 241-276, 1999.

BIBLIOGRAPHY

[1] J. Reifman and E. E. Feldman, “Multilayer Perceptron for Nonlinear Programming”, Computers and Operations Research, Vol. 29, pp: 1237-1250, 2002.

[2] M. Ferraro, G. Boccignone and T. Caelli, “Entropy-based Representation of Image Formation”, Pattern Recognition Letters, Vol. 23, pp: 1391-1398, 2002.

[3] L. P. Cordella, P. Foggia, C. Sansone and M. Vento, “Learning Structural Shape Descriptions from Examples”, Pattern Recognition Letters, Vol. 23, pp: 1427-1423, 2002.

[4] C. De Stefano, A. D. Cioppa and A. Marcelli, “Character Preclassification based on Genetic Programming”, Pattern Recognition Letters, Vol.23, pp: 1439-1448, 2002.

[5] J. Neumann, H. Samet and A. Soffer, “Integration of Local and Global Shape Analysis for Logo Classification”, Pattern Recognition Letters, Vol. 23, pp: 1449-1457, 2002.

[6] T. Hu, L. C. De Silva and K. Sengupta, “A Hybrid Approach of NN and HMM for Facial Emotion Classification”, Pattern Recognition Letters, Vol. 23, pp: 1303-1310, 2002.

[7] A. Verikas and M. Bacauskiene, “Feature Selection with Neural Networks”, Vol. 23, pp: 1323-1335, 2002.

[8] W. -J. Li and T. Lee, “Object Recognition and Articulated Object Learning by Accumulative Hopfield Matching”, Vol. 35, pp: 1933-1948, 2002.

[9] N. S. Philip and K.B. Joseph, “Adaptive Basis Function for Artificial Neural Networks”, Neurocomputing, Vol. 47, pp: 21-34, 2002.

[10] A. -C. Huang and Y. -F. Chiang, “Function Approximation using Serial Input Neuron”, Neurocomputing, Vol. 47, pp: 85-101, 2002.

[11] G. A. Kohring, “Artificial Neurons with Arbitrarily Complex Internal Structures”, Neurocomputing, Vol. 47, pp: 103-118, 2002.

[12] D. Charles, C. Fyfe, D. McDonald and J. Koetsier, “Unsupervised Neural Networks for the Identification of Minimum Over Complete Basis in Visual Data”, Neurocomputing, Vol. 47, pp: 119-143, 2002.

[13] B. Aryulu and B. Barshan, “Neural Networks for Improved Target Differentiation and Localization with Sonar”, Neural Networks, Vol. 14, pp: 355-373, 2001.

[14] G. N. Marichal, L. Acosta, L. Moreno, J. A. Mendez, J. J. Rodrigo and M. Sigut, “Obstacle Avoidance for a Mobile Robot: A Neuro-Fuzzy Approach”, Fuzzy Sets and Systems, Vol. 124, pp: 171-179, 2001.

[15] H. Maaref and C. Barret, “Sensor-based Navigation of a Mobile Robot in an Indoor Environment”, Robotics and Autonomous Systems, Vol. 38, pp: 1-18, 2002.

[16] I. Monasterio, E. Lazkano, I. Rano and B. Sierra, “Learning to Traverse Doors using Visual Information”, Vol. 2088, pp: 1-10, 2002.

[17] U. Bernardet, M. Blanchard and P. F. M. J. Verschure, “IQR: A Distributed System for Real-Time Real-World”, Neurocomputing, Vol. 150, pp: 1-6, 2002.

[18] T. Belker, M. Beetz and A. B. Cremers, “Learning Action Models for Improved Execution of Navigation Plans”, Robotics and Autonomous Systems, Vol. 38, pp: 137-148, 2002.

[19] A. Al-Hasan and G. Vachtsevanos, “Intelligent Route Planning for Fast Autonomous Vehicles Operating in a Large Natural Terrain”, Robotics and Autonomous Systems, Vol. 981, pp: 1-24, 2002.

[20] K. D. Reilly, “Agent Computing Themes in Biologically Inspired Models of Learning and Development”, International Journal in Developmental Neuroscience, Vol. 756, pp: 1-17, 2002.

[21] P. Auger, R. B. De La Perra, S. Morand and E. Sanchez, “A Predator-Prey Model with Predators using Hawk and Dove Tactics”, Mathematical Biosciences, Vol. 177&178, pp: 185-200, 2002.

[22] T. Kaita, S. Tomita and J. Yamanaka, “On Higher-Order Neural Network for Distortion Invariant Pattern Recognition”, Pattern Recognition Letters, Vol. 23, pp: 977-984, 2002.

[23] J. L. Contreras-Vidal and W. Schultz, “A Predictive Reinforcement Model of Dopamine Neurons For Learning Approach Behaviour”, Journal of Computational Neuroscience, Vol. 6, Issue 3, pp: 191-214, 1999.

[24] M. A. Arbib, “Rana Computatrix: Progress Report”, Federation Proceedings, Vol. 44, Issue 15, pp: 2956-2958, 1985.

[25] D. Chen and P. Burrell, “On the Optimal Structure Design of Multilayer Feedforward Neural Networks for Pattern Recognition”, International Journal of Pattern Recognition and Artificial Intelligence, Vol. 16, Issue 4, pp: 375-398, 2002.

[26] M. Mahowald and R. Douglas, "A Silicon Neuron", Nature, Vol. 354, Issue 6354, pp: 515-518, December 19th – 26th 1991.

[27] D. M. Blitz, A. E. Christie, M. J. Coleman, B. J. Norris, E. Marder and M. P. Nusbaum, "Different Proctolin Neurons Elicit Distinct Motor Patterns from a Multifunctional Neuronal Network", The Journal of Neuroscience: The Official Journal of the Society for Neuroscience, Vol. 19, Issue. 13, pp: 5449-5463, 1999.

[28] C. A. Lowry, J. D. Rose and F. L. Moore, "Corticotropin-Releasing Factor Enhances Locomotion and Medullary Neuronal Firing in an Amphibian", Hormones and Behaviour, Vol. 30, Issue. 1, pp: 50-59, 1996.

[29] O. Marin, A. Gonzalez and W. J. A. J. Smeets, "Basal Ganglia Organization in Amphibians: Efferent Connections of the Striatum and the Nucleus Accumbens", The Journal of Comparative Neurology, Vol. 380, Issue 1, pp: 23-50, 1997.

[30] S. K. Kostyk and P. Grobstein, "Neuronal Organization Underlying Visually Elicited Prey Orienting in the Frog-II. Anatomical Studies on the Laterality of Central Projections", Neuroscience, Vol. 21, Issue. 1, pp: 57-82, 1987.

[31] T. Masino and P. Grobstein, "The Organization of Descending Tectofugal Pathways Underlying Orienting in the Frog, Rana Pipiens.II. Evidence for the Involvement of a Tecto-Tegmento-Spinal Pathway", Experimental Brain Research (Experimentelle Hirnforschung, Experimentation Cerebrale), Vol. 75, Issue. 2, pp: 245-264, 1989.

[32] S. H. Chung, T. V. Bliss and M. J. Keating, “The Synaptic Organization of Optic Afferents in the Amphibian Tectum”, Proceedings of the Royal Society of London. Series B. Biological Sciences, Vol. 187, Issue 1089, pp: 421-447, 1974.

[33] E. A. Debski and M. Constantine-Paton, “Evoked Pre- and Post-Synaptic Activity in the Optic Tectum of the Cannulated Tadpole”, Vol. 167, Issue. 3, pp: 377-390, 1990.

[34] D. J. Ingle, “Some Effects of Pretectum Lesions on the Frogs’ Detection of Stationary Objects”, Behavioural Brain Research, Vol. 1, Issue. 12, pp: 139-163, 1980.

[35] U. Dicke and G. Roth, “Similarities and Differences in the Cytoarchitecture of the Tectum of Frogs and Salamanders”, Acta Biologica Hungarica, Vol. 47, Issue 1-4, pp: 41-59, 1996.

[36] V. S. Fenelon, V. Kilman, P. Meyrand and E. Marder, “Sequential Developmental Acquisition of Neuromodulatory Inputs to a Central Pattern-Generating Network”, Journal of Comparitive Neurology, Vol. 408, Issue 3, pp: 335-351, 1999.

[37] A. Gonzalez, W. J. A. J. Smeets and O. Marin, “Evidence for Shared Features in the Organization of the Basal Ganglia in Tetrapods: Studies in Amphibians”, European Journal of Morphology, Vol. 37, Issue 2-3, pp: 151-154, 1999.

[38] S. T. Grafton, L. Fadiga, M. A. Arbib and G. Rizzolatti, “Premotor Cortex Activation during Observation and Naming of Familiar Tools”, NeuroImage, Vol. 6, Issue 4, pp: 231-236, 1997.

[39] J. Brown, D. Bullock and S. Grossberg, “How the basal Ganglia use Parallel Excitatory and Inhibitory Learning Pathways to Selectively Respond to Unexepected Rewarding Cues”, The Journal of Neuroscience: The Official Journal of the Society for Neuroscience, Vol. 19, Issue 23, pp: 10502-10511, 1999.

[40] A. S. Fernandez, C. Pieau, J. Reperant, E. Boncinelli and M. Wassef, “Expression of the Emx-1 and Dlx-1 Homeobox Genes Three Molecularly Distinct Domains in the Telencephalon of Mouse, Chick, Turtle and Frog Embryos: Implications for the Evolution of Telencephalic Subdivisions in Amniotes”, Development (Cambridge, England), Vol. 125, Issue 11, pp: 2099-2111, 1998.

[41] R. F. Chapman, “The Insects, Structure and Function”, London: Hodder and Stoughton, pp: 1611-1630, 1980.

[42] S. R. Shaw, “Early Visual Processing in Insects”, Journal of Experimental Biology, Vol. 1124, pp: 225-251,1984.

[43] J. B. Nardi, “The Construction of the Insect Compound Eye: The Involvement of Cell Displacement and Cell Surface Properties in the Positioning of Cells”, Developmental Biology, Vol. 61, pp: 287-298, 1977.

[44] J. G. Nicholls, A. R. Martin and B. G. Wallace, “From Neuron to Brain: A Cellular and Molecular Approach to the Function of the Nervous System”, Sinauer Associates Inc, 1992.

[45] Y. Kim and J. A. Carr, “Alterations in Brain and Plasma Concentrations of Apha-Melanocyte-Stimulating Hormone during

Habituation of Prey-Catching Behaviour in Toads”, Brain, Behaviour and Evolution, Vol. 49, Issue 3, pp: 149-161, 1997.

[46] C. M. Olsen, A. T. Lovering and J. A. Carr, “α-Melanocyte-Stimulating Hormone and Habituation of Prey-Catching Behaviour in the Texas Toad, Bufo Speciosus”, Hormones and Behaviour, Vol. 36, Issue 1, pp: 62-69, 1999.

[47] A. M. Carpenter and J. A. Carr, “The Effects of Melanocortin Peptides and Corticosterone on Habituation in the Great Toad, Bufo Cognatus”, Hormones and Behaviour, Vol. 30, Issue 3, pp: 236-243, 1996.

[48] J. T. Heinen, “Predator Cues and Prey Responses: A Test using Eastern Garter Snakes (Thamnophis S. Sirtalis) and American Toads (Bufo A. Amercanus)”, Copeia, Issue 3, pp: 738-741, 1995.

[49] P. Patton and P. Grobstein, “The Effects of Telencephalic Lesions on Visually Mediated Prey Orienting Behaviour in the Leopard Frog (Rana Pipens). I. The Effects of Complete Removal of One Telencephalic Lobe, with a Comparison to the Effects of Unilateral Tectal Lobe Lesions”, Brain, Behaviour and Evolution, Vol. 51, Issue 3, pp: 123-143, 1998.

[50] T. Kozicz and G. Lazar, “The Origin of Tectal NPY Immunopositive Fibres in the Frog”, Brain Research, Vol. 635, Issue 1-2, pp: 345-348, 1994.

[51] C. W. Anderson and K. C. Nishikawa, “The Roles of Visual and Proprioceptive Information during Motor Program Choice in Frogs”, Journal of Comparitive Physiology. A, Sensory, Neural and Behavioural Physiology, Vol. 179, Issue 6, pp: 753-762, 1996.

[52] C. W. Anderson and K. C. Nishikawa, “A Prey-Type Dependent Hypoglossal Feedback System in the Frog Rana Pipens”, Brain, Behaviour and Evolution, Vol. 42, Issue 3, pp: 189-196, 1993.

[53] G. Chevalier, S. Vacher and J. M. Deniau, “Inhibitory Nigral Influence on Tectospinal Neurons, a Possible Implication of Basal Ganglia in Orienting Behaviour”, Experimental Brain Research, Vol. 53, Issue 2, pp: 320-326, 1984.

[54] D. L. Wang and M.A. Arbib, “How Does the Toad’s Visual System Discriminate Different Worm-like Stimuli?”, Biological Cybernetics, Vol. 64, Issue 3, pp: 251-261, 1991.

[55] E. R. Gruberg and J. Y. Lettvin, “Anatomy and Physiology of a Binocular System in the Frog Rana Pipens”, Brain Research, Vol. 192, Issue 2, pp: 313-352, 1980.

[56] K. Donner, “Visual Latency and Brightness: An Introduction based on the Responses of Rods and Ganglion Cells in the Frog Retina”, Visual Neuroscience, Vol. 3, pp: 39-51, 1989.

[57] D. R. Copenhagen, S. Hemila and T. Reuter, “Signal Transmission through the Dark-adapted Retina of the Toad (Bufo Marinus)”, Journal of General Physiology, Vol. 95, 1990, pp: 717-732.

[58] K. Donner and Maj–Len Gronholm, “Centre and Surround Excitation in the Receptive Fields of Frog Retinal Ganglion Cells”, Vision Research, Vol. 24, No, 12, pp: 1807-1819, 1984.

[59] Z. Li, K. V. Fite, N. M. Montgomery and S.R. Wang, “Single-Unit Responses to Whole-field Visual Simulation in the Pretectum of Rana Pipiens!, Neuroscience Letters, Vol. 218, pp: 193-197, 1996.

[60] K. Donner, A. Koskelainen, K. Djupsund and S. Hemila, "Changes in Retinal Time Scale Under Background Light: Observations on Rods and Ganglion Cells in the Frog Retina", Vision Research, Vol. 35, No. 16, pp: 2255-2266, 1995.

[61] B. A. Draper, R. T. Collins, J. Brolio, A. R. Hanson and E. M. Riseman, "The Schema System", International Journal of Computer Vision, Vol. 2, pp:209-250, 1989.

[62] A. Cobast and M. Arbib, "Prey-catching and Predator-avoidance in Frog and Toad: Defining the Schemas", Journal of Theoretical Biology, Vol. 157, pp: 271-304, 1992.

[63] E. R. Gruberg, "A Forebrain Visual Projection in the Frog (Rana Pipiens)", Experimental Neurology, Vol. 44, Issue. 2, pp: 187-197, 1974.

[64] M. A. Arbib, "Artificial Intelligence and Brain Theories", Vol. 3, Issue. 3, pp: 238-274, 1975.

[65] R. Chipalkatti and M. A. Arbib, "The Prey Localisation Model: A Stability Analysis", Mathematical and Computer Modelling, Vol. 12, Issue. 3, pp: 381, 1989.

[66] R. Chipalkatti and M. A. Arbib, "The Cue Interaction Model of Depth Perception: A Satbility Analysis", Journal of Mathematical Biology, Vol. 26, Issue. 3, pp: 235-262, 1989.

[67] R. Chipalkatti and M. A. Arbib, "The Prey Localisation Model: A Stability Analysis", Biological Cybernetics, Vol. 57, Issue 4-5, pp: 287-299, 1987.

[68] R. Lara, M. A. Arbib and A. S. Cromarty, “The Role of the Tectal Column in Facilitation of Amphibian Prey-catching Behaviour: A Neural Model”, The Journal of Neuroscience: The Official Journal of the Society for Neuroscience, Vol. 2, Issue 4, pp: 521-530, 1982.

[69] R. Lara, M. A. Arbib, “A Model of the Neural Mechanisms Responsible for Pattern Recognition and Stimulus Specification Habituation in Toads”, Biological Cybernetics, Vol. 51, Issue 4, pp: 223-237, 1985.

[70] P. F. Dominey, M. A. Arbib, “A Cortico-subcortical Model for Generation of Spatially Accurate Sequential Saccades”, Cerebral Cortex, Vol. 2, Issue 2, pp: 153-175, 1992.

[71] D. L. Wang and M. A. Arbib, “How does the Toad’s Visual System Discriminate different Worm-like Stimuli?”, Biological Cybernetics, Vol. 64, Issue 3, pp: 251-261, 1991.

[72] J. H. Hateren, “Spatiotemporal Contrast Sensitivity of Early Vision”, Vision Research, Vol. 33, No. 2, pp: 257-267, 1993.

[73] D. J. Tolhurst, “Separate Channels for the Analysis of the Shape and the Movement of a Moving Visual Stimulus”, Journal of Physiology, Vol. 231, pp: 385-402, 1973.

[74] G. A. Horridge and L. Marcelja, “On the Existence of ‘Fast’ and ‘Slow’ Directionally Sensitive Motion Detector Neurons in Insects”, Proceedings of the Royal Society London B, Vol. 248, pp: 47-54, 1992.

[75] V. Banerjee and S. L. Zipursky, “The Role of Cell-Cell Interaction in the Development of the Drosophila Visual System”, Neuron, Vol. 4, pp: 177-187, 1990.

[76] J. –P. Ewert, “Concepts in Vertebrate Neuroethology”, Animal Behaviour, Vol. 33, pp: 1-29, 1985.

[77] R. D. Hawkins, E. R. Kendel and S. A. Siegelbaum, “Learning to Modulate Transmitter Release: Themes and Variations in Synaptic Plasticity”, Annual Review of Neuroscience, Vol. 16, pp:625-665, 1993.

[78] D. E. Rumelhart and D. Zisper, “Feature Discovery by Competitive Learning”, Cognitive Science, Vol. 9, pp: 75-112, 1985.

[79] G. Bugmann, “Biologically Plausible Neural Computation”, Biosystems, Vol. 40, pp: 11-19, 1997.

[80] T. Koshizen, K. Akatsuka and H. Tsujino, “A Computational Model of Attentive Visual System Induced by Cortical Neural Network”, Neurocomputing, Vol. 128, pp: 1-7, 1998.

[81] M. Burrows, “Synaptic Potentials Affect the Release of Transmitter from Locust Non-spiking Interneurons”, Science, Vol. 204, pp: 81-83, 1979.

[82] H. G. Krapp and R. Henstenberg, “A Fast Stimulus Procedure to Determine Local Receptive Field Properties of Motion-sensitive Visual Interneurons”, Vision Research, Vol. 37, No. 2, pp: 225-234, 1997.

[83] T. Hrycej, “Modular Learning in Neural Networks: A Modularised Approach to Neural Networks”, John Wiley and Sons, 1992.

[84] Z. Li, S. Hu and H. Lu, "Feedforward Neural Networks Design by Evolutionary Programming", Critical Technology: The Third World Congress on Expert Systems, pp: 1247-1254, 1996.

[85] Z. –C. Li and C. Y. Suen, "The Partition-combination for Recognition of Handwritten Characters", Pattern Recognition Letters, Vol. 21, pp: 701-720, 2000.

[86] L. Cornette, P. Dupont, A. Rosier, S. Snaert, P. Van Hecke, J. Michiels, "Human Brain Regions Involved in Direction Discrimination", The American Physiological Society, pp: 2749-2765, 1998.

[87] C. Schiltz, J. M. Bodart, S. Dubois, S. Dejardin, C. Michel, A. Roucoux. M. Crommelinck and G. A. Orban, "Neuronal Mechanisms of Perceptual Learning: Changes in Human Brain Activity with Training in Orientation Discrimination", Neuroimage, Vol. 9, pp: 46-62, 1999.

[88] M. Schurmann, C. Basar-Eroglu, E. Basar, "Evoked EEG Alpha Oscillations in the Cat Brain – A Correlate of Primary Sensory Processing", Neuroscience Letters, Vol. 240, pp: 41-44, 1998.

[89] M. Burrows and M. V. S. Siegler, "Organization of Receptive Fields of Spiking Interneurons in the Locust With Inputs From Hair Afferents", The American Physiological Society, Vol. 53, pp: 1147-1157, 1985.

[90] P. Husbands, I. Harvey, D. Cliff and G. Miller, "Artificial Evolution: A New Path for Artificial Intelligence", Brain and Cognition, Vol. 34, pp: 130-159, 1997.

[91] K. R. Gibson and T. Ingold (Eds), "Tools, Language and Cognition in Human Evolution", Cambridge University Press, 1993.

[92] P. Auger, R. B. De La Parra, S. Morand and E. Sanchez, "A Predator-Prey Model with Predators using Hawk and Dove Tactics", Mathematical Biosciences, Vol. 177-178, pp: 185-200, 2002.

[93] K. D. Reilly, "Agent Computing Themes in Biological Inspired Models of Learning and Development", International Journal of Developmental Neuroscience, Vol. 756, pp: 1-17, 2002.

[94] C. F. Touzet, "Neural Reinforcement Learning for Behaviour Synthesis", Robotics and Autonomous Systems, Vol. 22, pp: 251-281, 1997.

[95] C. Rodriguez, S. Rementeria, J. I. Martin, A. Lafuente, J. Muguerza and J. Perez, "Fault Analysis with Modular Neural Networks", Electrical Power and Energy Systems, Vol. 18, No. 2, pp: 99-110, 1996.

[96] A. Teller and M. Veloso, "Internal Reinforcement in a Connectionist Programming Approach", Artificial Intelligence, Vol. 120, pp: 165-198, 2000.

[97] M. A. Arbib, A. Billard, M. Iacoboni and E. Oztop, "Synthetic Brain Imaging: Grasping, Mirror Neurons and Imitation", Neural Networks, Vol. 13, pp: 975-997, 2002.

[98] R. P. W. Duin, D. De Ridder, D. M. J. Tax, "Experiments with a Featureless Approach to Pattern Recognition", Vol. 18, pp: 1159-1166, 1997.

[99] S. Bandyopadhyay, S. K. Pal, “Pattern Classification with Genetic Algorithms: Incorporation of Chromosome Differentiation”, Pattern Recognition Letters, Vol. 18, pp: 119-131, 1997.

[100] P. Somol, P. Pudil, J. Novovicova and P. Paclik, “Adaptive Floating Search Methods in Feature Selection”, Pattern Recognition Letters, Vol. 20, pp: 1157-1163, 1999.

[101] C. K. Tham, “Reinforcement Learning of Multiple Tasks Using A Hierarchical CMAC Architecture” Robotics and Autonomous Systems, Vol. 15, pp: 247-274, 1995.

[102] C. Macleod, G. Dror and G. Maxwell, “Training Artificial Neural Networks using Taguchi Methods”, Vol. 13, pp: 177-184, 1999.

[103] K. Nieselt-struwe and P. R. Willis, “The Emergence of Genetic Coding in Physical Systems”, Journal of Theoretical Biology, Vol. 187, pp: 1-14, 1997.

[104] M. Tistarelli and E. Grosso, “Active Face Recognition with a Hybrid Approach”, Pattern Recognition Letters, Vol. 18, pp: 933-946, 1997.

[105] N. V. Subba Reddy and P. Nagabushan, “ A Connectionist Expert System Model for Conflict Resolution in Unconstrained Handwritten Numeral Recognition”, Pattern Recognition Letters, Vol. 19, pp: 161-169, 1998.

[106] R. K. Brouwer, “Automatic Growing of a Hopfield Style Network During Training for Classification”, Neural Network, Vol. 10, No. 3, pp: 529-537, 1997.

[107] K. B. Eom, “Shape Recognition using Spectral Features”, Pattern Recognition Letters, Vol. 19, pp: 189- 195, 1998.

[108] J. Soldek and R. Mantiuk, “A Reconfigurable Processor based on FPGAs for Pattern Recognition, Processing, Analysis and Synthesis of Images”, Pattern Recognition Letters, Vol. 20, pp: 667-674, 1999.

[109] T. M. English, “Stacked Generalisation and Simulated Evolution”, Biosystems, Vol. 39, pp: 3-18, 1996.

[110] P. Varona, J. J. Torres, R. Huerta, H. D. I. Abarbanel and M. I. Rabinovich, “Regularization Mechanisms of Spiking-Bursting Neurons”, Neural Networks, Vol. 14, pp: 865-875, 2001.

[111] F. T. Sommer and T. Wennekers, “Associative Memory in Networks of Spiking Neurons”, Neural Networks, Vol. 14, pp: 825-834, 2001.

[112] K. Y. Kupeev and H. J. Wolfson, “A New Method of Estimating Shape Similarity”, Pattern Recognition Letters, Vol. 17, pp: 873-887, 1996.

[113] T. Caelli and G. West, M. Robey and E. Osman, “A Relational Method for Pattern Recognition and Object Recognition”, Image and Vision Computing, Vol. 17, pp: 391-401, 1999.

[114] F. R. Prete, “Complex Worlds from Simpler Nervous Systems”, MIT Press, 2004.

[115] T. H. Bullock (Ed), “Brain Behaviour and Evolution – Contributions to Neuroethology”, International Symposium, La Jolla, California, January 9-10, 1995.

[116] J. –P. Ewert, H. Buxbaum-Conradi, M.Glagow, A. Rottgen, E. Schurg-Pfeiffer and W. W. Schwippert, “Forebrain and Midbrain Structures Involved In Prey-catching Behaviour of Toads: Stimulus-Response Mediating Circuits and their Modulating Loops”, European Journal of Morphology, Vol. 37, No.1, pp: 111-115, 1999.

[117] J. -P. Ewert, “Neural Correlates of Key Stimulus and Releasing Mechanisms: A Case Study and Two Concepts”, Trends in Neurosciences, Vol. 20, No. 8, pp: 332-339, 1997.

[118] M. Glagow, J. -P. Ewert, “Amorphine-induced Suppression of Prey Oriented Turning in Toads is Correlated with Activity Changes in Pretectum and Tectum: [^{14}C] 2DG Studies and Single Cell Recordings”, Neuroscience Letters, Vol. 220, pp: 215-218, 1996.

[119] S. Wachowitz and J. -P.Ewert, “A Key by Which the Toad’s Visual System Gets Access to the Domain of Prey”, Physiology and Behaviour, Vol. 60, No. 3, pp: 877-887, 1996.

[120] M. A. Arbib and J. -P. Ewert, “Visual Structures and Integrated Functions”, Springer-Verlag, 1991.

[121] J. -P.Ewert, “Tectal Mechanisms that Underlie Prey-catching and Avoidance Behaviours in Toads”, from “Comparitive Neurology of the Optic Tectum”, Horacio Vanegas (Ed), Plenum Publishing Corporation, 1984.

[122] J. –P. Ewert, “Neural Mechanisms of Prey-catching and Avoidance Behaviour in the Toad”, Brain, Behaviour and Evolution, Vol. 3, No. 1-4, pp: 36-56, 1970.

[123] W. W. Schwippert and J. –P. Ewert, “Effect of Neuropeptide-Y on Tectal Field Potentials in the Toad”, Brain Research, Vol. 669, pp: 150-152, 1995.

[124] C. Spreckelsen, E. Schurg-Pfeiffer, J. -P. Ewert, “Responses of Retinal and Tectal Neurons in Non-paralyzed Toads Bufo Bufo and B. Marinus to the Real Size Versus Angular Size of Objects Moved at

Variable Distance", Neuroscience Letters, Vol. 184, pp: 105-108, 1995.

[125] M. Glagow and J. -P. Ewert, "Dopaminergic Modulation of Visual Responses in Toads. II Influences of Apomorphine on Retinal Ganglion Cells and Tectal Cells", Journal of Computational Physiology, Vol. 180, pp: 11-18, 1997.

[126] H. Burghagen and J. P. Ewert, "Question of "Head Preference " in Response to Worm-like Dummies during Prey-capture of Toads, Bufo Bufo", Behavioural Processes, Vol. 7, pp: 295-306, 1982.

[127] J. –P. Ewert, H. Burghagen, L. Albrecht and J. Kepper, "Effects of Background Structure on the Discrimination of Configurational Moving Prey Dummies by Toads Bufo Bufo (L.)", Journal of Computational Physiology, Vol. 147, pp: 179-187, 1982.

[128] E. Schurg-Pfeiffer and J. –P. Ewert, "Investigation of Neurons Involved in the Analysis of Gestalt Prey Features in the Frog Rana Temporaria", Journal of Computational Physiology, Vol. 141, pp: 139-152, 1981.

[129] A. Weerasuriya and J. –P. Ewert, "Prey-Selective Neurons in the Toad's Optic Tectum and Sensorimotor Interfacing: HRP Studies and Recording Experiments", Journal of Computational Physiology, pp: 429-434, 1981.

[130] Th. Finkenstadt, "Disinhibition of Prey-catching in the Salamander Following Thalamic-pretectal Lesions", Naturwissenschaften, Vol. 67, pp; 471, 1980.

[131] J. -P. Ewert, B. Arend, V. Becker and H. –W. Borchers, “Invariants in Configurational Prey Selection by Bufo Bufo (L.)”, Brain, Behaviour and Evolution, Vol. 16, No. 1, pp: 38-51, 1979.

[132] J. –P. Ewert and H. Burghagen, “Ontogenetic Aspects on Visual ‘Size-constancy’ Phenomena in the Midwife Toad Alytes Obstetricans (Laur.)”, Brain, Behaviour and Evolution, Vol. 16, No. 2, pp: 99-112, 1979.

[133] J. -P.Ewert, H. -W. Borchers and A. V. Wietersheim, “Directional Sensitivity, Invariance and Variability of Tectal T5 Neurons in Response to Moving Configurational Stimuli in the Toad Bufo Bufo (L.)”, Journal of Computational Physiology, Vol. 132, pp: 191-201, 1979.

[134] J. -P. Ewert, H. Krug and G. Schonitz, “Activity of Retinal Class R3 Ganglion Cells in the Toad Bufo Bufo (L.) in Response to Moving Configurational Stimuli: Influence of the Movement Direction”, Journal of Computational Physiology, Vol. 129, pp: 211-215, 1979.

[135] A. Beck and J. -P. Ewert, “Prey Selection by Toads (Bufo Bufo) in Response to Configurational Stimuli Moved in the Visual Field z, y-Co-ordinates”, Journal of Computational Physiology, Vol. 129, pp: 207-209, 1979.

[136] H. -W. Borchers and J. -P. Ewert, “Correlation Between Behavioural and Neuronal Activities of Toads Bufo Bufo (L.) in Response to Moving Configurational Prey Stimuli”, Behavioural Processes, Vol. 4, pp: 99-106, 1979.

[137] H. –W. Borchers, H. Burghanen and J. -P. Ewert, “Key Stimuli of Prey for Toads (Bufo Bufo L.): Configuration and Movement Patterns”, Journal of Computational Physiology, Vol. 128, pp: 189-192, 1978.

[138] J. –P. Ewert and W. Kehl, “Configurational Prey-selection by Individual Experience in the Toad Bufo Bufo”, Journal of Computational Physiology, Vol. 126, pp: 105-114, 1978.

[139] A. V. Wietersheim and J. -P. Ewert, “Neurons of the Toad’s (Bufo Bufo L.) Visual System Sensitive to Moving Configurational Stimuli: A Statistical Analysis”, Journal of Computational Physiology, Vol. 126, pp: 35-42, 1978.

[140] J. -P. Ewert, E. Schurg-Pfeiffer and A. Weerasuriya, “Neurophysiological Data Regarding Motor Pattern Generation in the Medulla Oblongata of Toads”, Naturwissenschaften, Vol. 71, pp: 590, 1984.

[141] P. R. Laming, J. –P. Ewert and H. -W. Borchers, “Effects of Telencephalic Ablation on Visual Unit, Sustained Potential Shift and EEGs Recorded from the Toad Tectum in Response to a Visual Stimulus”, Behavioural Neuroscience, Vol. 98, No. 1, pp: 118-124, 1984.

[142] P. R. Laming and J. -P. Ewert, “The Effects of Pretectal Lesions on Neuronal Sustained Potential Shift and Electroencephaloraphic Responses of the Toad Tectum to Presentation of a Visual Stimulus”, Vol. 76A, No. 2, pp; 247-252, 1982.

[143] Th. Finkenstadt, S.O.E. Ebbeson and J. -P. Ewert, “Projections to the Midbrain Tectum in Salamandra Salamandra L.”, Cell Tissue Research, Vol. 234, pp: 39-55, 1983.

[144] H. Burhagen and J. -P. Ewert, “Influence of the Background for Discriminating Object Motion from Self Induced Motion in Toads Bufo Bufo (L.)”, Journal of Computational Physiology, Vol. 152, pp: 241-249, 1983.

[145] T. Finkenstadt and J. -P. Ewert, “Processing of Area Dimensions of Visual Key Stimuli by Tectal Neurons in Salamandra Salamandra”, Journal of Computational Physiology, Vol. 153, pp: 85-98, 1983.

[146] T. Finkenstadt and J. -P. Ewert, “Visual Pattern Discrimination through Interactions of Neural Networks: A Combined Electrical Brain Stimulation, Brain Lesion and Extracellular Recording Study in Salamandra Salamandra”, Journal of Computational Physiology, Vol. 153, pp: 99-110, 1983.

[147] J. –P. Ewert, H. Buxbaum-conradi, S.Fingerling, E. Schurg-Pfeiffer, T. W. Beneke, A. W. Dinges, M. Glagow and W. W. Schwippert, “Adapted and Adaptive Properties in Neural Networks for Visual Pattern Discrimination: A Neurobiological Analysis Toward Neural Engineering”, Adaptive Behaviour, Vol. 1, No. 2, pp: 123-154, 1992.

[148] J. -P. Ewert and R. Traud, “Releasing Stimuli for Antipredator Behaviour in the Common Toad Bufo Bufo (L.)”, Behaviour, pp: 170-180, 1979.

[149] J. –P. Ewert and G. Siefert, “Seasonal Change of Contrast Detection in the Toad’s Bufo bufo (L.) Visual System”, Journal of Computational Physiology, Vol. 94, pp: 177-186, 1974.

[150] J. –P. Ewert and D. Hock, “Movement –sensitive Neurones in the Toad’s Retina”, Experimental Brain Research, Vol. 16, pp: 41-59, 1972.

[151] J. –P. Ewert and D. Ingle, “Excitatory Effects following Habituation of Prey-catching Activity in Frogs and Toads”, Journal of Comparative and Physiological Psychology, Vol. 77, No. 3, pp: 369-374, 1971.

[152] J. –P. Ewert and H. Burghagen, “Configurational Prey Selection by Bufo, Alytes, Bombina and Hyla”, Brain, Behaviour and Evolution, Vol. 16, No. 3, pp: 157-175, 1979.

[153] H. Buxbaum-Conradi and J. –P. Ewert, “Responses of Single Neurons in the Toad’s Caudal Ventral Striatum to Moving Visual Stimuli and Test their Efferent Projection by Extracellular Antodromic Stimulation/ Recording Techniques”. Brain, Behaviour and Evolution, Vol. 54, pp: 338-354, 1999.

[154] J. -P. Ewert, T. W. Beneke, E. Schurg-Pfeiffer, W. W. Schwippert and A. Weerasuriya, “Sensorimotor Processes that Underlie Feeding Bahaviour in Tetrapods”, Advances in Comparative and Environmental Physiology, Vol. 18, pp: 119-162, 1994.

[155] E. Schurgh-Pfeiffer, C. Speckelsen and J. –P. Ewert, “Temporal Discharge Patterns of Tectal and Medullary Neurons Chronically Recorded during Snapping toward Prey in Toads Bufo Bufo Spinosus”, Journal of Computational Physiology A, Vol. 173, pp: 363-376, 1993.

[156] J. –P. Ewert, H. Buxbaum-conradi, F. Dreisvogt, M. Glagow, C. Merkel-Harff, A. Rottgen, E. Schurg-Pfeiffer and W.W. Schwippert, “Neural Modulation of Visuomotor Functions Underlying Prey-catching Behaviour in Anurans: Perception, Attention, Motor Performance, Learning”, Comparative Biochemistry and Physiology, Vol. 128A, No. 3, pp: 417-461, 2001.

[157] W. W. Schwippert, T. W. Beneke and J. -P. Ewert, “Responses of Medullary Neurons to Moving Visual Stimuli in the Common Toad II. An Intracellular Recording and Cobalt-Lysine Labelling study”, Journal of Computational Physiology A, Vol. 167, pp: 509-520, 1990.

[158] F. Gonzalez-Lima, T. Finkenstadt and J. –P. Ewert, “Neural Subtrates for Long Term Habituation of the Acoustic Startle Reflex in Rats: A 2-deoxyglucose Study”, Neuroscience Letters, Vol. 96, pp: 151-156, 1989.

[159] T. Finkenstadt and J. –P. Ewert, “Stimulus Specific Long term Habituation of Visually Guided Orienting Behaviour toward Prey in Toads: a 14 C –2DG Study”, Journal of Computational Physiology A, Vol. 163, pp: 1-11, 1988.

[160] H. J. Tsai and J. -P. Ewert, “Influence of Stationary and Moving Textured Backgrounds on the Response of Visual Neurons in Toads (Bufo Bufo L.)”, Brain, Behaviour and Evolution, Vol. 32, pp: 27-38, 1988.

[161] N. Matsumoto, W. W. Schwippert and J. -P. Ewert, “Intracellular Activity of Morphologically Identified Neurons of the Grass Frog’s

Optic Tectum in Response to Moving Configurational Visual Stimuli", Journal of Computational Physiology A, Vol. 159, pp: 721-739, 1986.

[162] M. Satou and J. -P. Ewert, "The Antidromic Activation of Tectal Neurons by Electrical Stimuli Applied to the Caudal Medulla Oblongata in the Toad, Bufo Bufo L." Journal of Computational Physiology A, Vol. 157, pp: 739-748, 1985.

[163] "Toward An Artificial eye", IEEE Spectrum, May 1996.

[164] K. Chen, "A Connectionist Method for Pattern Classification with Diverse Features", Pattern Recognition Letters, Vol. 19, pp: 545-558, 1998.

[165] G. G. Mascetti, "Interaction Between the Hemispheres in Split-Brain Cats", Neuropsychologia, Vol. 35, No. 6, pp: 913-918, 1997.

[166] F. Gaillard, M. A. Arbib, F. J. corbacho and H. B. Lee, "Modelling the Physiological Responses of Anuran R3 Ganglion Cells", Vision Research, Vol. 38, pp: 2551-2568, 1998.

[167] L. J. Rogers and C. Deng, "Light Experience and Lateralization of the Two Visual Pathways in the Chick", Behavioural Brain Research, Vol. 98, pp: 277-287, 1999.

[168] H. Luksch and G. Roth, "Pretecto-tectal Interactions: Effects of Lesioning and Stimulating the Pretectum on Field Potentials in the Optic Tectum of Salamanders in vitro", Neuroscience Letters, Vol. 217, pp: 137-140, 1996.

[169] R. Glenn Northcutt and J. Kaas, "The Emergence and Evolution of Mamallian Neocortex", Trends in Neuroscience, Vol. 18, No. 9, pp: 373-379, 1995.

[170] S. B. Selleck and H. Stellar, “The Influence of Retinal Innervation on Neurogenesis in the First Optic Ganglion of Drosophila”, Neuron, Vol. 6, pp: 83-99, 1991.

[171] M. E. Power. “The Effect of Reduction in Numbers of Ommatidia upon the Brain of Drosophila Melanogasta”, Journal of Experimental Biology, Vol. 94, pp: 33-72, 1943.

[172] U. Banerjee and S. L. Zipursky, “The Role of Cell-Cell Interaction in the Development of the Drosophila Visual System”, Neuron, Vol. 4, pp: 177-187, 1990.

[173] R. Ernst and M. Heisenberg, “The Memory Template in Drosophila Pattern Vision at the Flight Simulator”, Vision Research, Vol. 39, pp: 3920-3933, 1999.

[174] P. Abshire and A. G. Andreou, “Relating Information Capacity to a Biophysical Model for Blowfly Photoreceptors”, Neurocomputing, Vol. 32-33, pp: 9-16, 2000.

[175] S. C. Chamberlain, “Circadian Rythms in the Horseshoe Crab Lateral Eye: Signal Transduction and Photostasis”, Biochemistry and Bioenergetics, Vol. 45, pp: 111-121, 1998.

[176] P. H. Schiller, “On the Specificity of Neurons and Visual Areas”, Behavioural Brain Research, Vol. 76, pp: 21-35, 1996.

[177] D. M. Wolpert and M. Kawato, “Multiple Paired Forward and Inverse Models for Motor Control”, Neural Networks, Vol. 11, pp: 1317-1329, 1998.

[178] J. Dean, “Animats and What they Can Tell Us”, Trends in Cognitive Sciences, Vol. 2, No. 2, pp: 60-67, 1998.

[179] J-Archady Meyer, "From Natural to Artificial Life: Biomimetic Mechanisms in Animat Designs", Robotics and Autonomous Systems, Vol. 22, pp: 3-21, 1997.

[180] O. Trullier, S. I. Wiener, A. Berthoz and J-archady Meyer, "Biologically Based Artificial Navigation Systems: Review and Prospects", Progress in Neurobiology, Vol. 51, pp: 483-544, 1997.

[181] J. Shin, "Towards Computational and Robotic Modelling of Animal Cognition and Behaviour", Neurocomputing, Vol. 44-46, pp: 985-993, 2002.

[182] M-Oliver Gewaltig, A. Richter and R. Kupper, "BLISS: Towards the Simulation of Brain-like Systems", Neurocomputing, Vol. 44-46, pp: 805- 810, 2002.

[183] J. Spoelstra, M. A. Arbib and N. Schweighofer, "Cerebellar Adaptive Control of a Biomimetic Manipulator", Neurocomputing, Vol. 26-27, pp: 881-889, 1999.

[184] F. Krebs and H. Bossel, "Emergent Value Orientation in Self – Organization of an Animat", Ecological Modelling, Vol. 96, pp: 143-164, 1997.

[185] J. Kodjabachian and J-Arcady Meyer, "Evolution and Development of Control Architectures in Animats", Robotics and Autonomous Systems, Vol. 16, pp; 161-182, 1995.

[186] T. Elliott and N. R. Shabolt, "Growth and Repair: Instantiating a Biologically Inspired Model of Neuronal Development on the Khepera Robot" Robotics and Autonomous Systems, Vol. 36, pp: 149-169, 2001.

[187] H. De Garis and M. Korkin, “The CAM-Brain Machine (CBM): an FPGA-based Hardware Tool that Evolves a 1000 Neuron-net Circuit Module in Seconds and Updates a 75 Million Neuron Artificial Brain for Real-time Robot Control”, Neurocomputing, Vol. 42, pp: 35-68, 2002.

[188] H. Frezza-Buet and F. Alexandre, “From Biological to a Computational Model for the Autonomous Behaviour of an Animat”, Information Sciences, Vol. 144, pp: 1-43, 2002.

[189] F. Hoffman, “Soft Computing Techniques for the Design of Mobile Robot Behaviours”, Information Sciences, Vol. 122, pp: 241-258, 2000.

[190] W. -Po Lee, “Evolving Complex Robot Behaviours”, Information Sciences, Vol. 121, pp: 1-25, 1999.

[191] J. Soldek, R. Mantiuk, “A Reconfigurable Processor based on FPGAs for Pattern Recognition, Processing, Analysis and Synthesis of Images”, Pattern Recognition Letters, Vol. 20, pp: 667-674, 1999.

[192] S. Nolfi, “Evolving Non-trivial Behaviours on Real Robots: A Garbage Collecting Robot”, Robotics and Autonomous Systems, Vol. 22, pp: 187-198, 1997.

[193] F. Mondaada and D. Floreano, “Evolution of Neural Control Structures: Some Experiments on Mobile Robots”, Robotics and Autonomous Systems, Vol. 16, pp; 183-195, 1995.

[194] M. A.Hussain, “Review of the Applications of Neural Networks in Chemical Process Control-simulation and Online Implementation”, Artificial Intelligence in Engineering. Vol. 13, pp: 55-68, 1999.

APPENDIX 1: ALGORITHMS

The DNN was trained using a combination of methods that were implemented in C++ as mentioned in Chapter 8. The algorithm used to train the DNN is given in this section.

```
Procedure for int main()


        Print start time of the program run

        Open File to write connection weights
        Open File to write average fitness of
        chromosomes

        Call function randomize()
        Call function to generate initial population of
        weights
        Call function to read input patterns from file
        Call function to read output patterns from file

        Initiate generation = 1

        Perform the following stepswhile
        Average Fitness < 48.667 && Count < 20

              Set total fitness = 0.0
              Call function to read weights


              For consecutive chromosomes (set of 98
              weights)
```

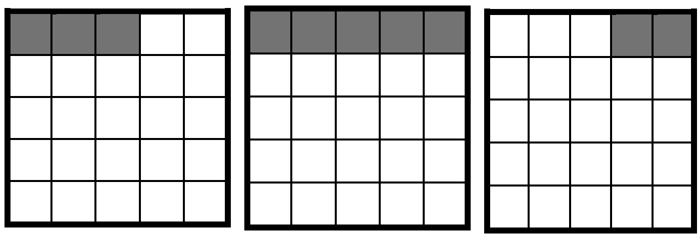
```
        Read input and corresponding output
        patterns
        Add random noise to input every 5
        generations

        Call function to calculate output of
        layer 2 neurons
        Call function to calculate output of
        layer 3 neurons
        Call function to calculate temporary
        output of layer 4 neurons
        Call function to calculate output of
        layer 5 neurons
        Call function to calculate final output
        of layer 4 neurons

        Call function to calculate rewards for
        Class 1 features
        Call function to calculate rewards for
        Class 2 features
        Call function to calculate rewards for
        Class 3 features

Call function to evaluate fitness of each
chromosome across all patterns
Call function to sort chromosomes based on
fitness
Calculate total fitness across all chromosomes

Calculate average fitness of the generation of
weights
```

```
        Write average fitness in file
        Call function to perform crossover operation
        Call function to perform Mutation operation

        Increment number of generations

        If average fitness >= 48.667, increment
        counter count

        Record finish time
        Close all files

End Program
```

APPENDIX 2: DNN CONFIGURATION

This section includes the final connection weights of the DNN that has been trained for patterns shown below in Figures A2.1, A2.2 and A2.3.

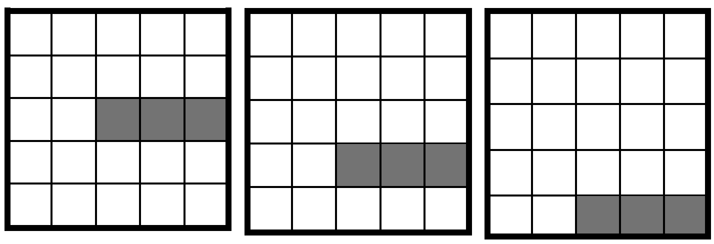
Class 1 – Prey and Orient Features



Feature 1 **Feature 2** **Feature 3**

Figure A2.1: Prey and Orient Features

Class 2 – Prey and Snap Features



Feature 1 **Feature 2** **Feature 3**

Figure A2.2: Prey and Snap Features

Class 3 – Predator Features

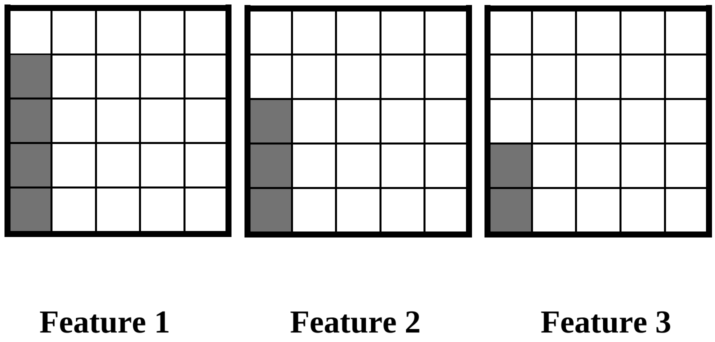


Figure A2.3: Predator Features

The DNN architecture is as given below in the Figure A2.4.

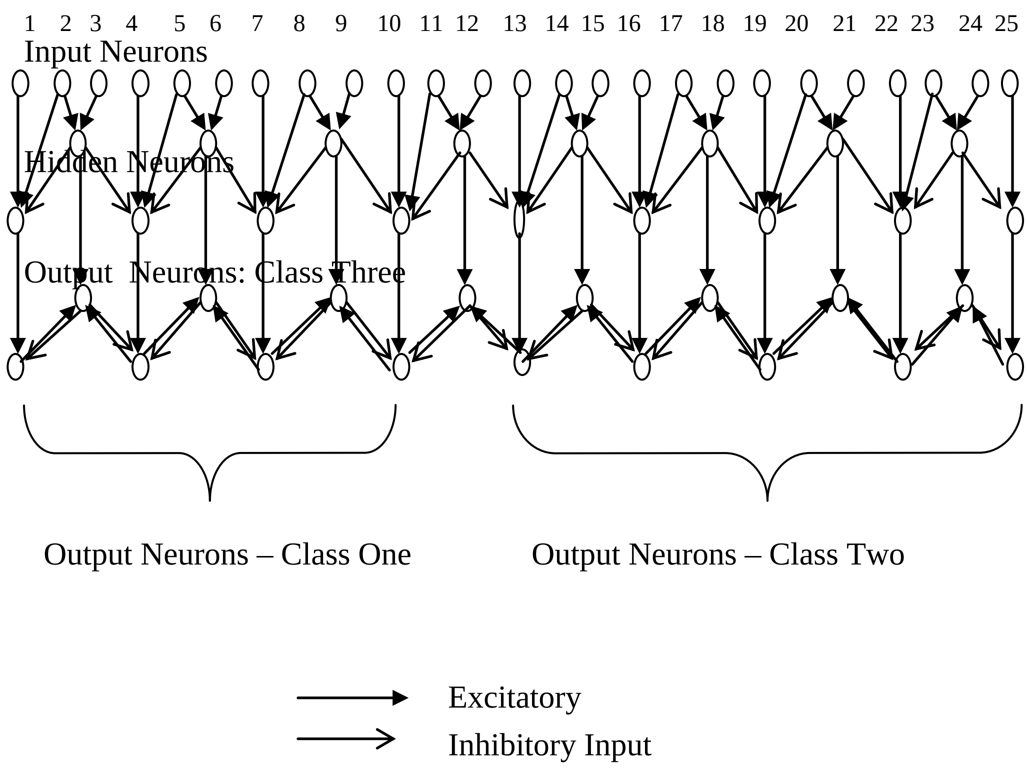


Figure A2.4: The Distributed Neural Network (DNN)

The Figure A2.5 below focuses on a part of the network for better clarity on the distribution of weights.

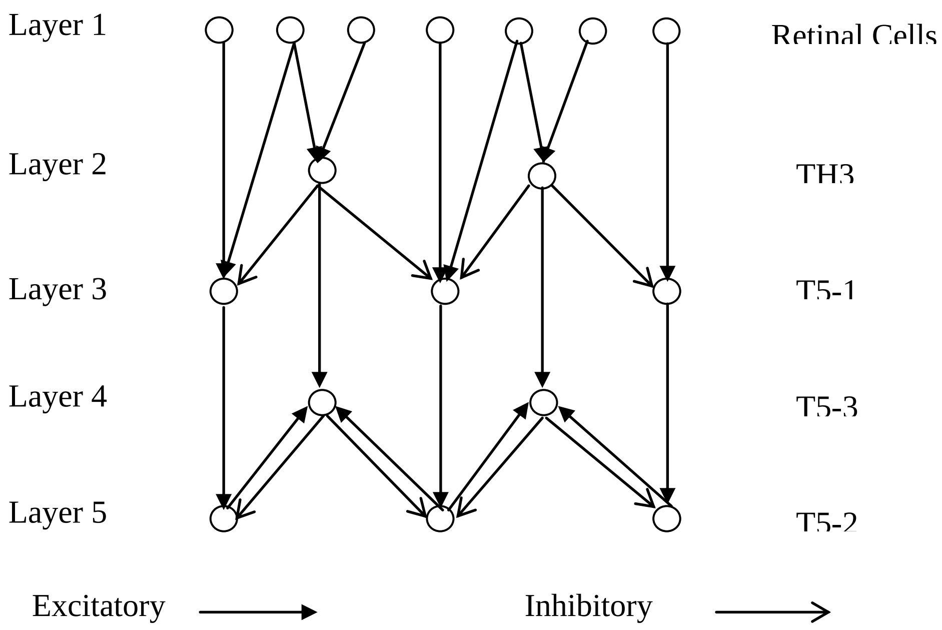


Figure A2.5: Network Design Stage 2 - Neuronal Connections

The weights are listed under the following labels showing the corresponding connection strengths. The connection weights are listed left to right (with regards to the network architecture) within a certain label or category. There are 98 weights corresponding to the 98 connections in the architecture shown in figure A2.4.

Layer 1 to Layer 2 *(Retinal to TH3)*

Layer 1 to Layer 3 *(Retinal to T5_1)*

Layer 2 to Layer 3 *(TH3 to T5_1)*

Layer 2 to Layer 4 *(TH3 to T5_3)*

Layer 3 to Layer 5 *(T5_1 to T5_2)*

Layer 4 to Layer 5 *(T5_3 to T5_2)*

Layer 5 to Layer 4 *(T5_2 to T5_3)*

Layer 1 to Layer 2 *(Retinal to TH3)*

There are 16 weights in total in this category. They are listed from left to right in the architecture and are as follows,

-31.000000

2.000000

3.783383

19.000000

-6.000000

-20.000000

1.000000

0.000000

16.000000

-4.000000

20.000000

-11.000000

-33.000000

17.000000

-11.000000

15.000000

Layer 1 to Layer 3 ***(Retinal to T5_1)***

There are 17 weights. They are,

-6.000000

-5.000000

8.000000

-1.000000

-4.000000

11.000000

-6.000000

41.000000

3.000000

-21.000000

-6.000000

-1.000000

-14.000000

0.000000

-1.000000

-9.000000

18.000000

Layer 2 to Layer 3 ***(TH3 to T5_1)***

There are 16 weights that belong to these types of connections.

1.000000

31.000000

0.000000

-1.000000

-8.277107

-1.000000

-5.000000

0.000000

1.000000

-2.000000

-3.368493

-16.000000

-2.000000

5.798741

3.000000

-5.718401

Layer 2 to Layer 4 *(TH3 to T5_3)*

There are 8 weights that belong to this category.

-12.000000

-6.000000

-5.000000

-10.000000

-13.000000

7.000000

-8.342758

-4.000000

Layer 3 to Layer 5 ***(T5_1 to T5_2)***

There are 9 weights that fall under this label.

-7.000000

0.000000

-6.000000

-16.156643

-6.856081

6.000000

-11.000000

-7.000000

-8.049016

Layer 4 to Layer 5 ***(T5_3 to T5_2)***

There are 16 weights that belong to these connections.

-11.490750

1.000000

34.499863

9.000000

15.000000

2.000000

-12.000000

60.000000

5.000000

11.000000

6.000000

0.000000

2.000000

15.000000

14.000000

-7.000000

Layer 5 to Layer 4 *(T5_2 to T5_3)*

There are 16 weights that belong to the connections between Layer 5 and Layer 4.

-11.000000

-27.000000

-1.000000

7.000000

2.000000

-1.000000

13.000000

0.000000

-6.000000

0.000000

-13.000000

-6.000000

-11.000000

-5.000000

-4.000000

-16.000000