INTELLIGENT PEST INSECTS CLASSIFICATION SYSTEM BASED IMAGE PROCESSING AND NEURAL ARBITRATION

**A THESIS SUBMITTED TO THE
GRADUATE SCHOOL OF APPLIED SCIENCES
OF
NEAR EAST UNIVERSITY**

**By
SALAH.S.A. ZARROUK**

**In Partial Fulfillment of the Requirements for
the Degree of Master of Science
in**

Computer Engineering

NICOSIA, 2016

SALAH.S.A.ZARROUK

INTELLIGENT PEST INSECTS CLASSIFICATION SYSTEM BASED
IMAGE PROCESSING AND NEURAL ARBITRATION

NEU
2016

INTELLIGENT PEST INSECTS CLASSIFICATION SYSTEM BASED IMAGE PROCESSING AND NEURAL ARBITRATION

A THESIS SUBMITTED TO THE

GRADUATE SCHOOL OF APPLIED SCIENCES

OF

NEAR EAST UNIVERSITY

By
SALAH.S.A. ZARROUK

In Partial Fulfillment of the Requirements for the Degree of Master of Science in

Computer Engineering

NICOSIA, 2016

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, last name:

Signature:

ACKNOWLEDGMENT

I would like to gratefully and sincerely thank Assist. Prof. Dr. Kamil Dimililer for his guidance, understanding, patience, and most importantly, his supervising during my graduate studies at Near East University. His supervision was paramount in providing a well-rounded experience consistent my long-term career goals. He encouraged me to not only grow as an experimentalist, but also as an instructor and an independent thinker. For everything you've done for me Assist. Prof. Dr. Kamil, I thank you. I would also like to thank Associate Prof. Rahib Abiyev for giving me the opportunity to be a member in such university and such department. Her help and supervision concerning taking courses was unlimited.

I would also like to thank NEU Grand library administration members, since it provided me with the appropriate environment for conducting my research and writing my thesis. Additionally, I am very grateful for my family, in particular my father for his help throughout my life.

ABSTRACT

Detection of insects in the agricultural fields is a major challenge in the field of agriculture. Therefore, effective and intelligent systems should be designed to detect the infestation while minimizing the use of pesticides. This thesis aims to develop an intelligent insect classification system that would be capable of detecting and classifying 8 types of most common insects that the paddy fields suffer from. The developed system comprises of two main stages. In the first stage, the insect's images are processed using different image processing technique in order to detect the geometric shapes of insects that will be used to distinguish different insects. In this phase, the images are smoothed using median filter and then segmented using canny edge detection. Then, the images are rescaled using pattern averaging in order to be ready for the next stage: neural network. The next phase is the classification phase where two backpropagation neural networks with different number neurons in the input layer are trained by different learning schemes and then tested on a large number of processed images till they converge and learn the geometric shape of each insect. Experimentally, the systems were tested on different insect images with two testing ratios the results show a high efficiency and classification rate of 94% for the system that uses more training than testing images and greater image size.

Keywords: Agriculture; backpropagation neural network; canny edge detection; classification; geometric shapes; image processing; insects; intelligent systems; median filter; pattern averaging

ÖZET

Tarım alanında böceklerin saptanması tarım alanında büyük bir sorundur. Bu nedenle, böcek ilacı kullanımını en aza indirirken, istismarın tespiti için etkili ve akıllı sistemler tasarlanmalıdır. Bu tez, çeltiklerin uğradığı en yaygın böceklerin 8 tipini tespit edebilen ve sınıflandırabilen akıllı bir böcek sınıflandırma sistemi geliştirmeyi amaçlamaktadır. Geliştirilen sistem iki ana aşamadan oluşmaktadır. İlk aşamada, böceklerin görüntüleri, farklı böcekleri ayırt etmek için kullanılacak böceklerin geometrik şekillerini tespit etmek için farklı görüntü işleme tekniği kullanılarak işlenir. Bu aşamada, görüntüler medyan filtre kullanılarak pürüzlendirilir ve daha sonra canny kenar algılaması kullanılarak bölünür. Ardından, görüntüler, bir sonraki aşamaya hazır olmak için desen ortalaması kullanarak yeniden ölçeklendirilir: sinir ağı. Bir sonraki aşama, giriş katmanında farklı sayıdaki nöronlara sahip iki geri yayılım nöral ağının farklı öğrenme planları tarafından eğitildiği ve her böceğin geometrik şeklini öğrenene kadar çok sayıda işlenmiş görüntü üzerinde test edildiği sınıflandırma fazıdır. Deneysel olarak, sistemler iki test oranına sahip farklı böcek görüntüleri üzerinde test edildi; sonuçlar, test görüntülerinden ve daha büyük görüntü boyutundan daha fazla eğitim kullanan sistem için% 94'lük yüksek verimlilik ve sınıflandırma oranı verdi.

Anahtar Kelimeler: Tarim; yayilma yayilimi sinir aği; canny kenar algilama; siniflandirma; geometrik şekiller; görüntü işleme; böcekler; akilli sistemler; medyan filtresi; örüntü ortalamalari

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER ONE
INTRODUCTION

1.1 Introduction

Agriculture field is one of the central points that are identified with social steadiness and monetary improvement. Nonetheless, a few hundred distinct types of insects are discovered connected with put away grains and their items, and insects that assault our stores of oat sustenances constitute a standout amongst the most genuine dangers to our development. It is evaluated that insects devastate 5%~10% of the world generation of grains. As a rule, the visual technique, which is broadly utilized the world over, depends basically on visual assessment and examination with standard pictures of insects. This methodology is very subjective and requires impressive preparing and involvement with insects to accomplish predictable results. Taking into account natural chemistry and biophysics systems, the trial strategies are tedious, repetitive, and are not reasonable for routine use (Thiago et al., 2011).

The manual grouping of such creepy crawly bothers in paddy fields can be tedious and requires generous specialized ability. The undertaking turns out to be all the more difficult when bug irritations are to be perceived from still pictures utilizing a mechanized framework. Pictures of one bug vermin might be taken from various perspectives, messed foundation, or may endure change, for example, revolution, commotion, and so forth. So it is likely that two pictures of the same bug nuisance will be distinctive (Mittal et al., 2012).

On a basic level, a goal, simple to-behavior, and routine technique for consequently distinguishing and ordering the put away item insects by PC is exceptionally attractive. To address these difficulties, we have received the geometric-based elements in characterizing pictures of paddy field creepy crawly bothers. The essential point of preference of this methodology is that it is invariant to changes in stance and scale the length of the elements can be dependably identified. Besides, with a fitting decision of classifier, for example, backpropagation neural classifier, not all components should be recognized keeping in mind the end goal to accomplish high characterization exactness. Subsequently, regardless of the possibility that some elements are impeded or neglect to be recognized, the technique can in any case succeed.

The main aim of this approach is to classify different types of insects based on their geometric features extracted using pattern averaging. The classification is handled using a backpropagation neural network due to its simplicity and efficiency in such applications. The images were processed using different image processing techniques in order to adjust the image edges and make them ready for pattern averaging. The images were processed using different image processing techniques in order to adjust the image edges and make them ready for pattern averaging. The images were rescaled to two sizes for simulation and testing purposes. Therefore, two backpropagation neural networks were used. The first one has 4096 input neurons since is fed by images of size 64*64, however the second one has 1024 neurons since it is fed by images of size 32*32. Experimentally, two learning schemes were used for testing the network one uses 80 images for training and 50 for testing. The other uses 80 for training and 112 for testing. The simulation results show difference in classification rate and effectiveness of both networks when tested by both testing ratios.

The experiment demonstrates a classification success rate of 90% for the first network, and 93% for the second one. The method developed in this paper has potential in classification of pest insects, as it is fast, non-destructive, reliable, robust to noise and tolerant to deformity of insects.

1.2 Literature Review

In (Larios et al., 2008) the authors have utilized a pack of elements way to deal with mechanize quick throughput taxonomic recognizable proof of stonefly hatchlings. 263 stonefly hatchlings were gathered of four stonefly taxa from freshwater streams in the mid-Willamette valley and Cascade Range of Oregon. Around ten photographs were gotten of every example, which yields 20 singular pictures. These were then physically inspected, and all pictures that gave a dorsal perspective inside 30 degrees of vertical were chosen for examination. The pictures were then grouped through a procedure that includes: Identification of districts of interest, representation of those areas as SIFT vectors, classification of the SIFT vectors into a histogram of distinguished components, and classification of the histogram by an outfit of logistic model trees. In their work, they have connected three district finders: Hessian-relative indicator and the Kadir entropy identifier, including a recently created primary ebb and flow based locale (PCBR) locator. The development of a codebook was performed by a Gaussian blend model (GMM). The authors guarantee that their PCBR identifier beats the other two locators while demonstrating a classification exactness of 82% for four classes and 95% for three classes.

In (Mundada et al., 2013), the authors have proposed a framework to recognize whiteflies, aphids and thrips on the tainted harvests in nursery. Pictures of the contaminated leaf are caught by a camera and pre-handled utilizing picture preparing procedures, for example, changing over pictures from RGB to dark scales and sifting keeping in mind the end goal to get an improved picture set of irritations. In highlight extraction, a few properties of the picture are considered. An assortment of locale properties and dark covariance lattice properties, for example, entropy, mean, standard deviation, contrast, vitality, relationship and capriciousness are removed from those pictures. The classification was performed by the utilization of bolster vector machines. The creators assert that the model framework demonstrated quick identification of irritations and displays the same execution level as a traditional manual methodology.

1.3 Aims of Thesis

Because of the quick advancement of computerized innovation, there is an open door for picture handling joined with simulated wise innovation to be utilized as a part of the field of horticultural exploration which could help the analyst to take care of a mind boggling issue. Image investigation gives a sensible chance to the robotization of bug location. This study broadens the usage of picture handling procedures notwithstanding a clever classification framework to recognize and arrange diverse sorts of insects by setting up a mechanized identification framework. Through this framework, crop professionals can without much of a stretch number the bugs from the gathered examples, and right vermin's administration can be connected to increment both the amount and nature of organic product or plants generation. Utilizing the computerized keen framework, crop professionals can make the observing procedure less demanding.

1.4 Thesis Overview

The thesis is divided into 6 chapters, which are structured as follows.

Chapter 1 is an introduction about the thesis. In this chapter we define our thesis; we set the aims, the contributions, and motivations. In addition, the structure of the thesis is discussed.

Chapter 2 introduces the insects in several aspects including the types of them, the risk and characteristics of each one of them. In addition, the impacts of insects on the agricultural and economical filed id presented. Moreover, the advantages and importance of insects are also discussed in this chapter.

Chapter 3 is a general introduction about neural network in which the concept of neural network is discussed; in addition to the backpropagation algorithm.

Chapter 4 is a general explanation of image analysis and common processing techniques used in this field.

Chapter 5 is a detailed explanation about the first phase of our proposed system, in which the image processing phase is discussed. In this chapter, we explain the image processing phase using graphs, figures, and flowcharts in order to explain our new developed intelligent system for the classification of pest insects. The image processing techniques used in the designed system are discussed in details in this chapter.

Chapter 5 discusses also the classification phase of the proposed system. This phase is based on an artificial neural network classifier. In this chapter, we define neural network and explain its concepts, in addition to the backpropagation learning algorithm that is used in our system and the reasons we used it. In this chapter, the training results of the system using tables, figures and curves such as the learning curves are provided. The system performance and the experimental results are also discussed in this chapter through tables and figures.

Finally**,** results comparison of the proposed identification system with previously proposed systems for the same aim which is the identification or detection of RA using image processing and neural classifier was explained in **Chapter 6**.

CHAPTER TWO
PLANT PEST INSECTS

2.1 Overview

This chapter presents an overview about the pests. In addition, it discusses the different types of insects that can found on plants. Moreover, this chapter presents the characteristics of each insect used in developing the system.

2.2 Pests

A pest is a living being with qualities that individuals see as harming or undesirable, as it damages farming through nourishing on yields or parasitizing domesticated animals. A creature can likewise be a pest when it causes harm to a wild environment or conveys germs. The term pest is utilized to allude particularly to destructive creatures yet it additionally identifies with all other unsafe living beings, including parasites and infections. It is feasible for a creature to be a pest in one setting however valuable or tamed in another. Numerous weeds (plant pests) are additionally seen as valuable under certain conditions (Martin & Moisan, 2008).

Crops should be shielded from an assortment of various pests, life forms that present a danger to the harvest. While we frequently consider pests insects, a pest can likewise be a weed, an infection or a creature, (for example, a rodent) or even microorganisms.

2.2.1 Targeting specific pests

Bug sprays, fungicides and herbicides are all harvest assurance items. Bug sprays are utilized to control creepy crawly pests, for example, aphids or greenfly. Fungicides manage the organisms or molds that can influence seed germination, crop development and the nature of the collected produce (Martin & Moisan, 2008). Weed-executioners control plant pests, for example, chickweed, knifes and blackgrass that ransack the harvest plant of light, water and nourishment.

While these three are the most widely recognized yield insurance items, different sorts are utilized against particular pests.

Pests can build up imperviousness to the yield insurance items. Resistance might be characterized as a heritable change in the affectability of a pest populace that is reflected in the rehashed disappointment of an item to accomplish the normal level of control when utilized by name proposal. Cross-resistance happens when imperviousness to one item presents imperviousness to another item, even where the pest has not been presented to the last item. Resistance may happen because of a rehashed utilization of the same item, or utilization of items with a solitary method of activity in the focused on pest. The best way to deal with resistance administration is Integrated Pest Management (IPM) which is the utilization of all accessible control techniques in a monetary and maintainable way (INGO, 2005).

2.3 Types of Plant Insects

It would be unimaginable for me to list each conceivable creepy crawly that assaults plants. The insects experienced the frequently are white or red creepy crawly vermin, aphids, mealybugs, scale, whitefly, thrips and numerous dirt staying insects. On the off chance that the issue is discovered sufficiently early, it can regularly be taken consideration without the utilization of insectcides, and that is, by a wide margin, the most ideal way (PRSSP, 2004).

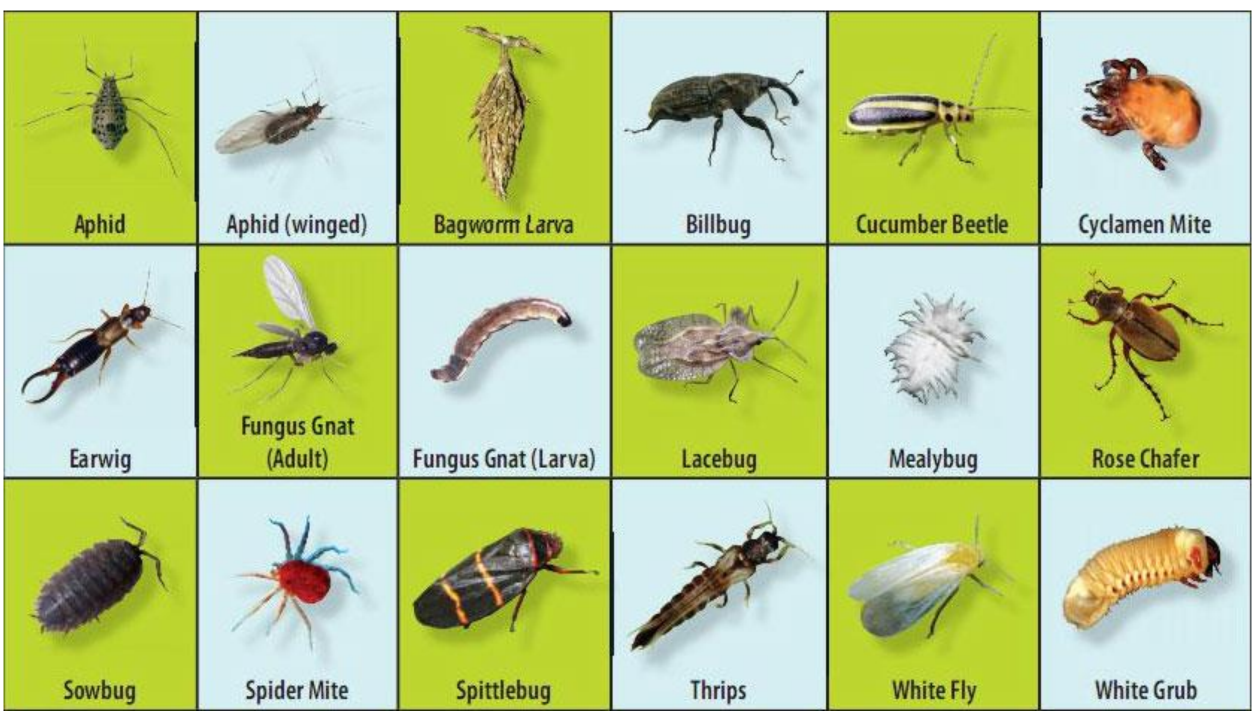


Figure 2.1: Plant pest insects (PRSSP, 2004)

2.3.1 Common plant pests

i. Aphids

Aphids are common on house plant insects but fortunately, are easily controlled. Aphids suck sap from the plant and can cause new growth to be stunted and distorted. Aphids may be just about any color and are found on new growth and the undersides of the leaves, usually clustered together in a group. Heavy infestations cover the plants with sticky honeydew. If you can handle living with female bugs in the house, they will take care of any aphid problem for you. Aphids can also be controlled with malathion, diazinon, and systemic insecticides.

Figure 2.2: Aphids (Naturepic, 2013)

ii. Spider mites

Spider mites are verging on difficult to see with the bare eye. They are to a great degree little and an amplifying glass is typically expected to see them. They typically assault new leaves and buds. Plants invaded with bugs lose their green shading and seem bronzed or washed out. In serious cases, the bugs will shape fine webbing covering the underside of clears out. Once a plant is swarmed with parasites, control will be troublesome, if not unimaginable. Disconnect your plant promptly, and plunge it or splash it week by week with insecticidal cleanser. Systemic bug sprays are now and again compelling if utilized soon enough.

Besides utilizing concoction showers which indicate that they will control arachnid bugs, the main thing I could propose would be great old cleanser and water.... Disposing of the insect parasites will take perserverance or you will simply put off the following infestation. I would attempt a natural insecticidal cleanser once more, and the following day start a procedure of showering every plant, every day with as solid of a surge of water as you think your plants will withstand, making sure to splash the bottoms of the leaves (Carino et al., 1979).

Parasites can imitate every 3-7 days, so it is important that you shower once a day until the issue is under control. Dry air urges bug bugs to breed so anything you can do to expand the encompassing dampness will help you in your 'parasite battle (Martin & Kumar, 2011).

Figure 2.3: Spider mites (Naturepic, 2013)

iii. Mealybugs

Mealybugs look like minimal white tufts of cotton so are regularly mixed up for a sickness. They are regularly found on the undersides of leaves or on stems at the zenith of leaf joins. The white, waxy covering shields the insects from showers, making control troublesome. Touching every creepy crawly with a little brush or Q-tip dunked in liquor will murder them, however the children are little and regularly neglected, so a repeat is conceivable. Be vigilante! Systemic bug sprays may work. Diazinon is viable the length of the mealys is completely wetted (Martin & Moisan, 2008).

Figure 2.4: Mealybugs (Naturepic, 2013)

iv. White flies

The grown-up whitefly is a little "white" fly. Their control is made more troublesome by the way that they will leave the plant when you attempt to splash them. The youthful phase of white fly is scale-like and doesn't move, so it is in this phase you should crush them by week by week showering or plunging with insecticidal cleanser. Malathion and diazinon splashes are powerful.

Figure 2.5: White flies (Naturepic, 2013)

v. Scale insects

Scale insects regularly develop to expansive numbers since they go undetected. These insects are oval, around 3mm long and look like little chestnut limpets. Their shell shields them from pesticides which makes their control more troublesome. Scales are typically found on stems and the undersides of leaves however can be on top of the takes off. Little infestations can be expelled by touching every creepy crawly with a cotton swab plunged in liquor. Scales suck your plants juices, hindering the plants development and they emit sticky honeydew which draws in ants and different pests. In the event that it gets to be fundamental, splash or plunge utilizing malathion as coordinated.

Figure 2.6: Scale insects (Naturepic, 2013)

vi. Soil insects

Soil insects are seen when conveyed to the surface amid watering. Grown-ups and hatchlings of a few insects may fly or slither around on the dirt surface. By and large they do no genuine mischief to the plant. Extensive populaces can bring about shrinking and poor plant development because of minor root pruning. Dirt splashing with insecticidal cleanser will more often than not take care of the issue. Systemic bug sprays are generally powerful.

Figure 2.7: Soil insects (Naturepic, 2013)

vii. Ants

The harm done by ants is generally backhanded and really brought about by the aphids, mealybugs or different insects which they "ranch" for the honeydew. They can however make harm the root arrangement of the plant as they tunnel to manufacture their home. A dirt dousing with insecticidal cleanser will as a rule take care of the issue. Systemic bug sprays are typically successful as are malathion and diazinon (PRSSP, 2004).

Figure 2.8: Ants (Myrmecos, 2013)

viii. Thrips

Thrips are little and difficult to see. They are light chestnut, thin insects, who while in the grown-up stage will travel to different plants when exasperates. They "scratch" into the leaves to get the plants juices, leaving the leaf mutilated, with detectable scars. Splashing with insecticidal cleanser will more often than not take care of the issue. Showering with malathion or diazinon is compelling. Systemic bug sprays are infrequently viable (Martin & Moisan, 2008).

Figure 2.9: Thrips (Naturepic, 2013)

ix. Earwigs

Earwigs are a genuinely surely understood bug, from fables if not from real experience. The earwig is the bug rumored in superstition to deliberately slither into the ears of dozing persons with the end goal of tunneling into the cerebrum to lay eggs. Obviously, there is no truth to these stories, however earwigs, similar to moths, insects, cockroaches, ants and flies may meander into our ear channels unintentionally.

Earwigs are genuinely basic, however not frequently bounteous in Iowa. They are once in a while saw aside from after delayed times of a year or more with wet climate. Earwigs are generally simple to recognize by the conspicuous pliers or forceps on the end of the belly. On females the pliers are genuinely straight, while male pliers are more bended and caliper-like. These pliers are utilized as both hostile and guarded weapons. Despite the fact that they may attempt to squeeze if caught and took care of, they don't hurt individuals. The normal earwig is around 5/8 creep long and dull cocoa with a rosy head and light yellow-chestnut legs (Catino et al., 1979).

Earwigs are outside insects generally found in sodden ranges, for example, under mulch, dead leaves, logs, and heaps of kindling, loads up, stones and different garbage or in spoiled wood where they eat soggy, rotting plant material. In spite of the fact that earwigs periodically assault living plants, including vegetables, blossoms and elaborate plants, they are viewed as just minor pests of plants in Iowa (PRSSP, 2004).

Figure 2.10: Earwigs (Naturepic, 2013)

x. Caterpillars

Caterpillar is the normal name for the hatchlings of individuals from the request Lepidoptera (the bug request involving butterflies and moths). Likewise with most regular names, the utilization of the word is subjective and the hatchlings of sawflies usually are called caterpillars too.

Caterpillars of most species are herbivorous, however not all; some are insectivorous, even savage. Some feast upon other creature items; for instance garments moths eat fleece, and horn moths eat the hooves and horns of dead ungulates.

Caterpillars generally speaking are insatiable feeders and a large number of them are among the most genuine of rural pests. Actually numerous moth species are best known in their caterpillar stages as a result of the harm they cause to foods grown from the ground agrarian produce, while the moths are dark and do no immediate damage. Then again, different types of caterpillar are esteemed as wellsprings of silk, as human or creature nourishment, or for organic control of pest plants.

Figure 2.11: Caterpillars (Naturepic, 2013)

xi. Grasshoppers

Grasshoppers are medium to expansive insects. Grown-up length is 1 to 7 cm, contingent upon the species. Like their relatives the katydids and crickets, they have biting mouthparts, two sets of wings, one thin and extreme, and the other wide and adaptable, and long rear legs for hopping. They are not quite the same as these gatherings in having short radio wires that don't achieve exceptionally far back on their bodies (Patil & Kumar, 2012).

Grasshoppers for the most part have expansive eyes, and are shaded to mix into their surroundings, more often than not a mix of cocoa, dim or green. In a few animal groups the guys have splendid hues on their wings that they use to pull in females. A couple of animal types eat lethal plants, and keep the poisons in their bodies for security. They are brilliantly shaded to caution predators that they taste terrible.

Female grasshoppers are bigger than the guys, and have sharp focuses toward the end of their guts that they to help lay eggs underground. Male grasshoppers now and then have exceptional structures on their wings that they can rub their rear legs on or rub together to make sounds.

Figure 1.12: Grasshoppers (Naturepic, 2013)

2.4 Impacts of Insects

Since they overwhelm every single physical environment that bolster human life, insects are typically our most imperative rivals for nourishment, fiber, and other common assets. They directly affect agrarian nourishment generation by biting the leaves of harvest plants, sucking out plant juices, exhausting inside the roots, stems or leaves, and spreading plant pathogens. They eat characteristic strands, demolish wooden building materials, ruin put away grain, and quicken the procedure of rot. They additionally profoundly affect the strength of people and residential creatures by bringing about inconvenience, perpetrating chomps and stings, and transmitting ailment.

The monetary effect of insects is measured not just by the business sector estimation of items they obliterate and the expense of harm they perpetrate additionally by the cash and assets consumed on anticipation and control of pest episodes. Despite the fact that dollar values for these misfortunes are about difficult to compute, particularly when they influence human wellbeing and welfare, business analysts for the most part concur that insects devour or crush around 10% of gross national item in huge, industrialized countries and up to 25% of gross national item in some creating nations (Thiago et al., 2011).

2.5 Importance and Advantages of Insects

2.5.1 Insects as Food

Insects can represent a critical sustenance hotspot for a wide assortment of other creature species. Freshwater amusement fish, for example, trout, bass, and bream encourage broadly on sea-going insects like mayflies, stoneflies, or hellgrammites. Fake "flies" utilized by fishermen are regularly made to take after a fish's common prey. Numerous amphibians, frogs, turtles, snakes, and reptiles additionally expend insects as a noteworthy piece of their eating regimen. Insectivory is regular among area staying flying creatures. Purple martins, stable swallows, vireos, songbirds, glimmers, whippoorwills, and swifts, for instance, survive only on insects. Different fowls, (for example, egrets, quail, geese, plovers, kills, and bluebirds) have a more fluctuated eating routine, yet regardless they determine a vast rate of their aggregate sustenance from insects. There are even some insectivorous warm blooded creatures: vixens, moles, bats, armadillos, and insect eating animals, for instance. At the point when other nourishment is rare, even foxes, racoons, skunks, and bears will swing to insects as a wellspring of sustenance (Thiago et al., 2011).

Insects were without a doubt a critical wellspring of nourishment for our initial human precursors. Indeed, even today, they are still gathered and eaten by individuals of numerous societies. In Mexico, dried grasshoppers are sold in town markets. High in protein and low in fat, they might be browned or ground into supper and blended with flour to make tortillas. Sago grubs, the hatchlings of a wood-exhausting scarab, are viewed as a delicacy in Papua New Guinea. The islanders heat up the hatchlings or meal them over an open flame. Ants, honey bees, termites, caterpillars, water bugs, bug hatchlings, flies, crickets, katydids, cicadas, and dragonfly fairies are among an extensive rundown of consumable insects that give nourishment to the general population of Australia, Africa, South America, the Middle East, and the Far East. Without a doubt, Americans and different descendants of western European society give off an impression of being exceptional among people groups of the world in having such a solid social forbidden against the utilization of insects as nourishment (PRSSP, 2004).

2.5.2 Ecological impact

As buyers, foragers, and decomposers, insects assume an indispensable part in the biogeochemical cycling of supplements. Insects circulate air through the dirt, enhance its maintenance of water, and upgrade its tilt. They turn more soil than night crawlers and redistribute supplements inside the root zone as they tunnel and home in the ground. Flies and compost scarabs keep the development of excrement from extensive creatures and rate up its deterioration by growths and microscopic organisms. Without such foragers, the progressive amassing of waste items from extensive crowds of dairy cattle and different ungulates (warm blooded animals with hooves) would soon render a great part of the scene inadmissible for horticultural purposes (INGO, 2005).

As parasites and predators of different living beings, insects are a piece of a characteristic arrangement of governing rules that fortifies group solidness and keeps dangerous populace development from overwhelming regular assets. In this way, more than 6000 creepy crawly species have been tried and discharged as organic control operators to battle insects and weeds that we see as pests. In any case, there are likewise innumerable different species that work for us as populace controllers, frequently unnoticed until they are accidently annihilated by a characteristic calamity or human intercession. Undoubtedly, human disturbance of regular environments is a typical reason for pest episodes. Over portion of every single farming pest in the United States have been accidently transported in from abroad: e. g. fire ants from South America, Japanese insects from the Orient, and tramp moths from Europe. Large portions of these species are not viewed as genuine pests in their countries since populace development is stifled by local parasites, predators, and maladies (INGO, 2005).

2.5.3 Products of beneficial insects

Since old times, bumble bees (Apis mellifera) have been esteemed for the nectar and beeswax they create. In numerous societies, these items, and the honey bees themselves, are seen with magical or religious criticalness (Thiago et al., 2011). To antiquated Greeks and Romans, nectar was the "nectar of the divine beings" - invested with recuperating properties and super-characteristic forces. Individuals of Asia utilized it as an additive for foods grown from the ground, Egyptians utilized it as a preserving liquid, and specialists in Europe and frontier America connected it as a germ-free to treat smolders and slashes. In Europe, nectar was frequently blended with wine or lager and could be matured to create a mainstream mixed drink known as mead. For some a huge number of years, nectar was the main sweetener a great many people utilized. The crystalline type of sugar sold today was not promptly accessible until the mid 1800's the point at which the primary business refineries were worked to concentrate sucrose from stick or beets (PRSSP, 2004).

Figure 2.12: Honey and Beeswax (Naturepic, 2013)

Nectar is one of the colony's standard sustenance assets. It is created from beads of blossom nectar accumulated by working drones. The nectar is incidentally held in the honey bee's foregut where enzymatic activity starts to change over sucrose into dextrose (glucose) and levulose (fructose). In the hive, this nectar-protein blend is exchanged to waxen cells, decreased in volume by vanishing of water, and permitted to mature into nectar. The honey bees seal every cell with a wax top when the procedure is finished. Working drones make upwards of 50,000 excursions to and from the hive and visit up to 4 million blooms keeping in mind the end goal to create a solitary kilogram of nectar (2.2 lbs). Substantial, solid hives may normal more than 25 kg (55 lbs) of nectar every year. In spite of the fact that the business sector for nectar is not as huge or as productive as it once seemed to be, yearly U.S. generation is still more than 115 million kg. A large portion of this nectar is utilized as an essential sweetener or as a substitute for refined sugar in heated merchandise. It is likewise a fixing in a couple hack drugs and intestinal medicines (INGO, 2005).

CHAPTER THREE
IMAGE PROCESSING

3.1 Image Analysis

Image analysis is the way of extract the meaningful information from image especially from digital image, this means of digital image processing techniques (Gonzalez & Woods, 2002).

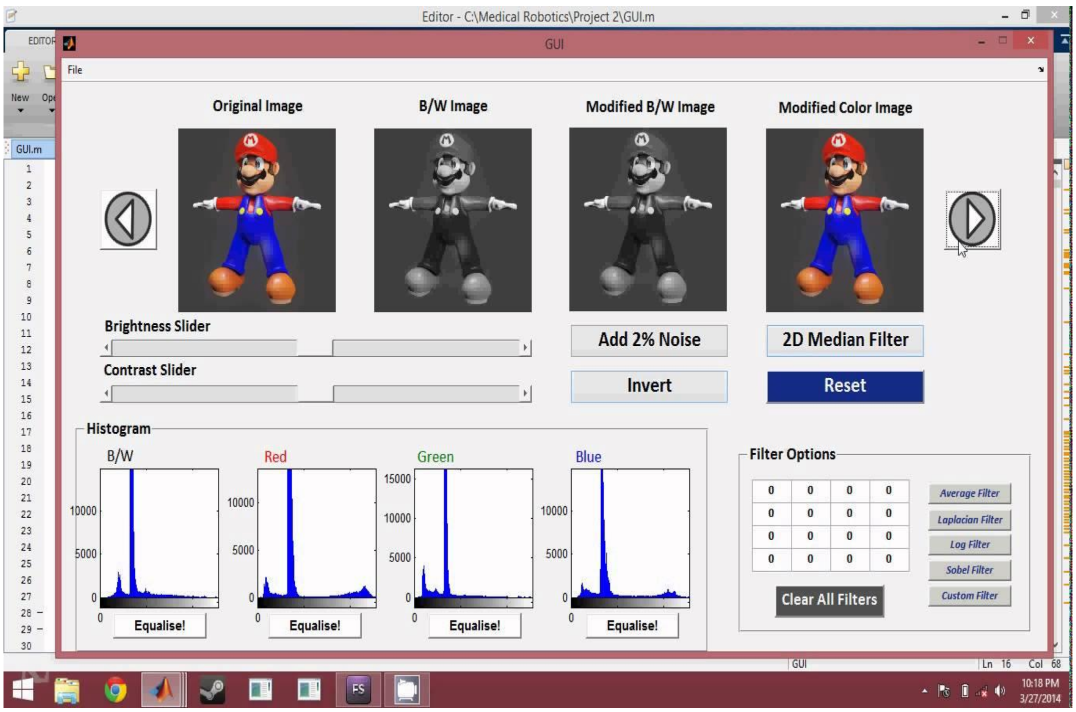


Figure 3.1: Image analysis (Gonzalez & woods, 2002)

3.2 History

Right on time of 1920s link picture transmission framework was found by Bartlane, it was utilized to transmit daily paper pictures over the Atlantic. The pictures were coded and sent by broadcast then printed by an uncommon transmit printer. It took around three hours to send a picture. The primary frameworks bolstered 5 dark levels.

In 1964 – NASA's Jet Propulsion Laboratory started dealing with PC calculations to enhance pictures of the moon. Picture was transmitted by Ranger 7 test.

In 1960s advanced picture handling was created at the Jet Propulsion Laboratory, Massachusetts Institute of Technology, Bell Laboratories, University of Maryland (Kim, 1997).

3.3 Image Processing Applications

The field of digital image has expansion in recent years. The usefulness of this technology is clear in many different disciplines (Gonzalez & Woods, 2002).

The fields of image processing are:-

1- Cinema

2- Medical industry

3- Machine vision

4- Digital camera images

3.3.1 Medical imaging

Medical imaging is a process and art that used to create visual representations of the body for medical intervention and clinical analysis. Medical imaging seeks to the internal structures hidden by the skin and bones, so it can diagnose and treat disease (Aggarwal et al., 2013).

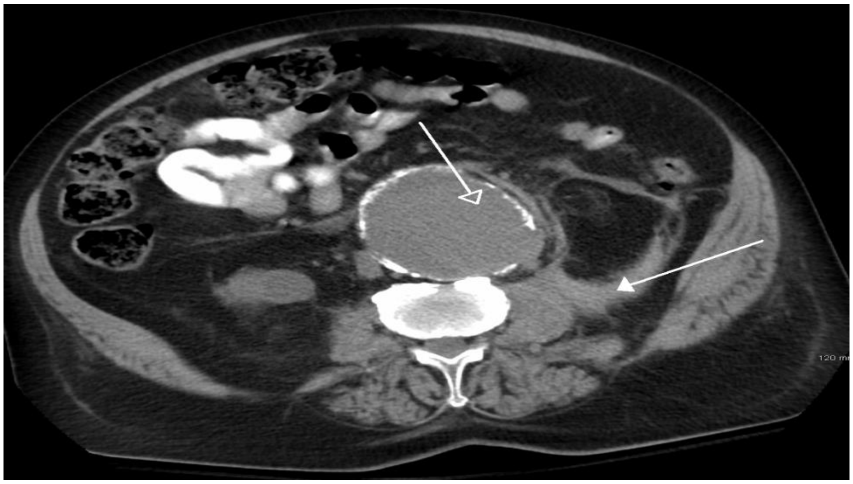


Figure 3.2: A CT scan image showing a ruptured abdominal aortic aneurysm (James et al., 2008)

3.3.2 Machine vision

Machine vision (MV) is the innovation or a strategy gives imaging-based programmed investigation and examination for applications like control of process, robot direction and programmed review in industry (Jamil et al., 2012).

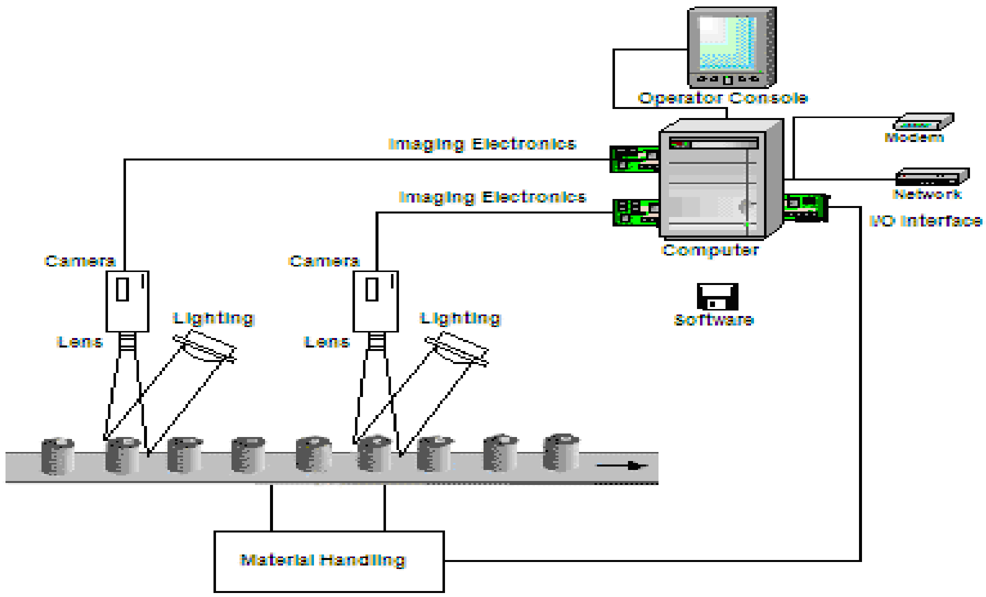


Figure 3.3: Machine vision system (Jamil et al., 2012)

3.3.3 Digital camera images

An advanced camera is a camera that encodes digitally computerized pictures and recordings and after that stores them for later multiplication. Today, most cameras are advanced, and computerized cameras are embedded into numerous gadgets extending cell phones (called camera telephones) to vehicles (Beham et al., 2012).

Figure 3.4: Digital camera (Beham et al., 2012)

3.4 Image Storage

We require particular consideration to ensure that the computerized photographs won't harm or lose. The earth of PC that computerized photographs are put away gives extraordinary open doors and in the meantime awesome perils. If not legitimately moved down, PC disappointment can scratch out your advanced photograph accumulation. A little error in altering can overwrite the photograph with another record. To ensure that the advanced photographs are legitimately put away we required a work process, a standard procedure of taking, putting away, altering and documenting your computerized photographs (Gonzalez & Woods, 2002).

Shading profundity, it is known as bit profundity, is various bits used to show the shade of a pixel, in a bitmapped picture or video or the quantity of bits that utilized for a solitary pixel. For High Efficiency Video Coding (H.265), the bit profundity indicates the quantity of bits utilized for every shading (Beham et al., 2012).

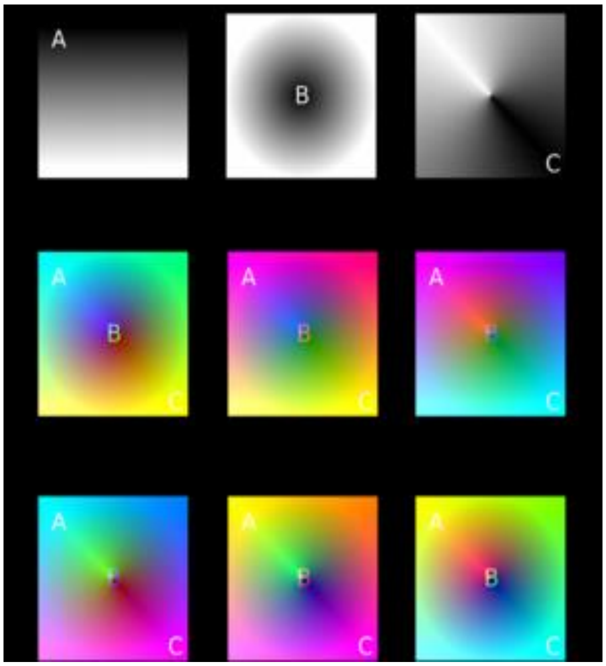


Figure 3.5: Color images composed from 3 grayscale images (Gonzalez & Woods, 2002)

3.5 Different Image Processing Techniques

Image processing techniques has a various and broad techniques that can be used in many fields. Here, some of these main techniques are discussed.

3.5.1 Image segmentation

In computer image segmentation is the process of partitioning an image into multiple segments (sets of pixels). Segmentation is used to simplify and/or change the representation of an image into more easier to analyse. Image segmentation is used to locate (lines, curves, etc.) in images (Shapiro & Stackman, 2000)

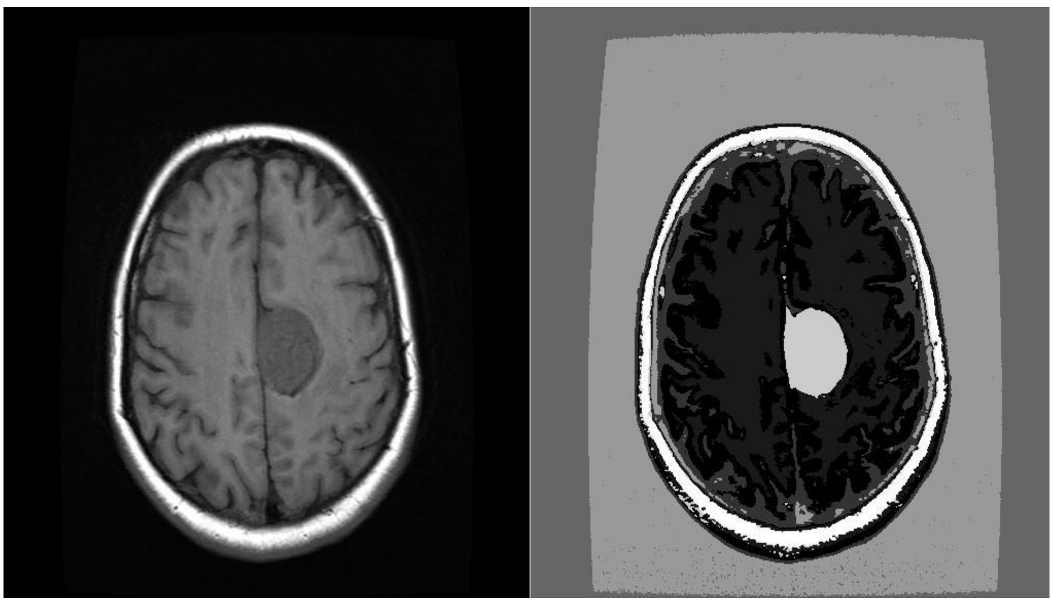


Figure 3.6: Images segmentation (Shapiro & Stackman, 2000)

3.6 Edge Detection

Edge identification is an arrangement of numerical strategies which used to recognize focuses in a computerized picture at which the picture shine changes forcefully. The focuses which changes pointedly are normally sorted out into a bended line sections called edges.

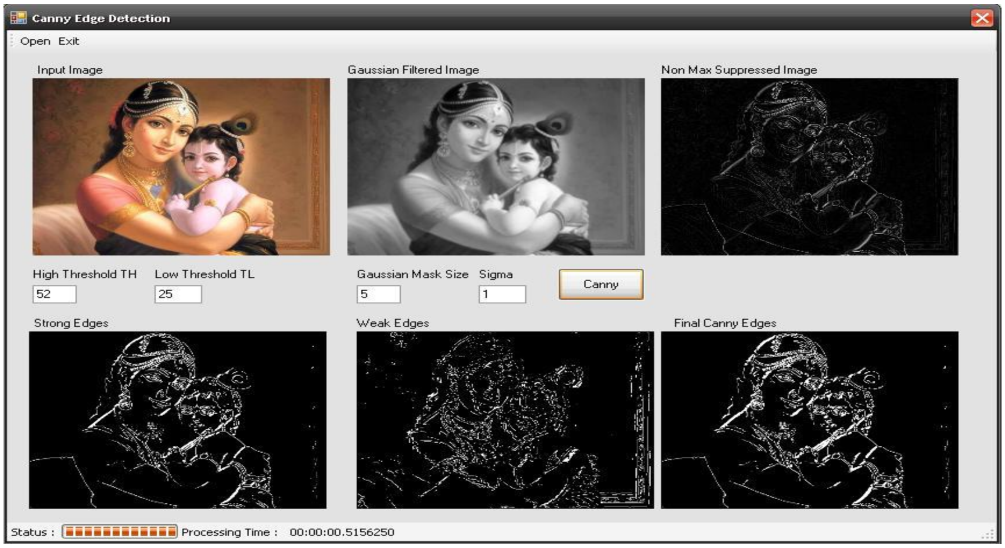


Figure 3.9: Sample of edge detection (Shapiro & Stackman, 2000)

3.7 Canny Operators

The Canny edge detector is generally utilized as a part of PC vision to find sharp force changes and to discover object limits in a picture. Pixel edges are connected with some force changes or discontinuities; hence, edge location is the procedure of distinguishing such sharp power contrasts (i.e., discontinuities) in a picture. Traditional edge identification administrators Sobel and Prewitt utilizes 3×3 bits which are convolved with the first picture to figure approximations of the subsidiaries - one for flat changes, and one for vertical. In this proposed framework, we distinguished edges utilizing shrewd administrators. This system is the most widely recognized utilized strategy for distinguishing edges and portioning the picture. The Canny edge indicator is considered as one of the best as of now utilized edge finders since it gives great commotion insusceptibility and identifies the genuine edges or force discontinuities while protecting a base mistake (Gonzalez & Woods, 2002).Vigilant administrator has been utilized for such calculation as to the accompanying criteria (Saif et al., 2012):

1. To maximize the signal-to-noise ratio of the gradient.

2. To ensure that the detected edge is localized as accurately as possible.

3. To minimize multiple responses to a single edge.

The steps of canny algorithm in order to segment an image into many regions are as follows:

1. Smoothing: it means blurring an image in order to remove noise and it is done by convolving the image with the Gaussian filter.

2. Finding gradients: Since the edges must be marked where the gradients of the image has large magnitudes, we have to find the gradient of the image by feeding the smoothed image through a convolution operation with the derivative of the Gaussian filtering both the vertical and horizontal directions.

3.7.1 Image enhancement

In computer representation, Image Enhancement is the procedure of enhancing the nature of a digitally put away picture by treating the picture with programming; for instance, to make a picture lighter or darker. Picture upgrade programming additionally bolsters numerous channels for pictures (Gonzalez & Woods, 2002).

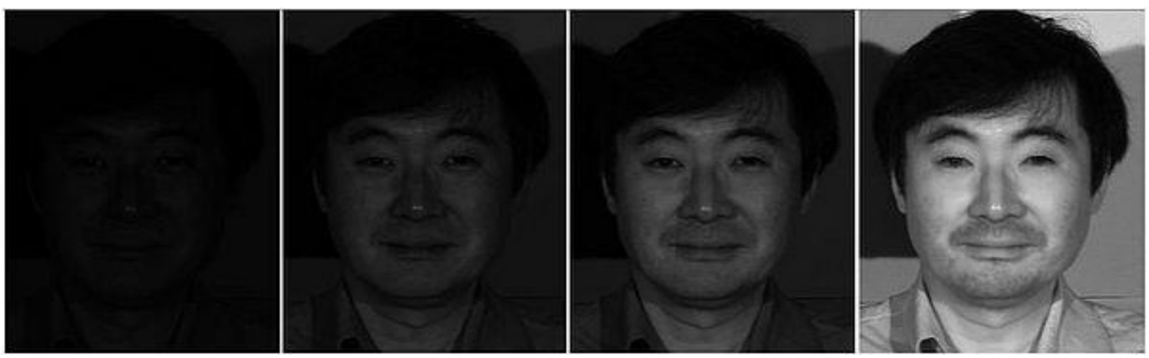


Figure 3.10: Sample of image enhancement (Kim, 1996)

3.7.2 Sobel operator

The Sobel operator utilized as a part of PC, and picture handling, particularly inside edge recognition calculations, which makes a picture with underscores edges and moves.

It depends on convolving the picture inside a little, divisible, and number esteemed channel in vertical and level bearing and is in this way generally as far as calculations (Wan et al., 1999).

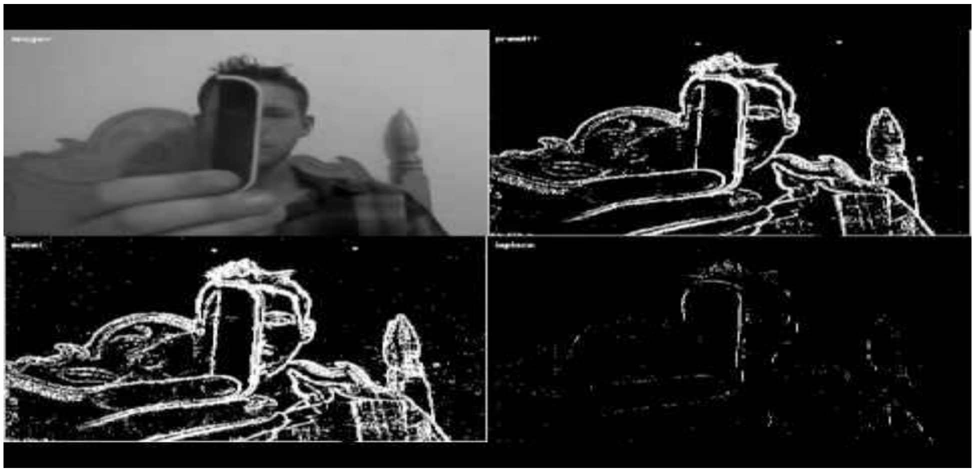


Figure 3.11: A sobel operator (Wan et al., 1999)

The Sobel operator performs a 2-D spatial inclination estimation on a picture. At that point, the estimated total inclination extent (edge quality) at every point can be found. The Sobel operator utilizes a couple of 3x3 convolution covers, one evaluating the angle in the x-course (sections) and the other assessing the slope in the y-axis (rows)

$$|G| = |G_x| + |G_y| \tag{3.3}$$

Where G_x is the gradient in the x-direction, while G_y is the gradient in the y-direction.

3.7.3 Top-hat transform

It is an operation which removes little points of interest and components from pictures. There are two sorts of top-cap change:

1-white top-cap change is the distinction between the information picture and its opening.

2-The dark top-cap change is the distinction between the information and the end picture. It is utilized for different picture handling undertakings, for example, highlight extraction, picture upgrade, and others (Wan et al., 1999).

3.8 Summary

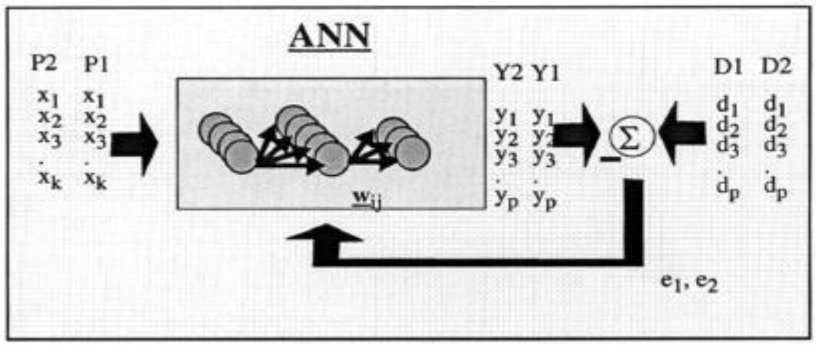
In this chapter the image processing and its basic techniques were presented, where the image processing techniques are used to improve the quality of the image. The applications of image processing that are used in our life are very useful, especially for health, industry and security.

CHAPTER FOUR
ARTIFICIAL NEURAL NETWORKS

4.1 Artificial Neural Network (ANN)

Artificial neural networks (ANNs) are the simple simulation of the structure and the function of the biological brain. The complex and accurate structure of the brain makes it able to do hard different simultaneous tasks using a very huge number of biological neurons connected together in grids. A first wave of interest in neural networks emerged after the introduction of simplified neurons by McCulloch and Pitts in 1943. These neurons were presented as models of biological neurons and as conceptual components for circuits that could perform computational tasks (Krose &Smagt, 1996). At that time, Von Neumann and Turing discussed interesting aspects of statistical and robust nature of brain-like information processing. But it was only in 1950s that actual hardware implementations of such networks began to be produced (Fyfe, 1996). ANNs are used widely nowadays in different branches of science. It is used for medical purposes like in (Khashman, 1999) and (Khashman, 2000). Used for image processing for different purposes like (Khashman & Demililer, 2007). It is also invested in power and power quality applications and active power filters (Valiviita, 1998) and (Sallam & Khafaga, 2002). In (Yuhong & Weihua, 2010) a survey on the application of the ANNs in forecasting financial market prices, financial crises, and stock prediction was presented.

The different mentioned applications of neural networks imply firstly the learning of the ANNs to do defined tasks. One of the most common methods of teaching ANNs to perform given tasks is the back propagation algorithm. It is based on a multi-stage dynamic system optimization method proposed by Arthur E. Bryson and Yu-Chi Ho in 1969. In 1974, it was applied in the context of ANNs through the works of Paul Werbos, David E. Rumelhart, Geoffrey E. Hinton and Ronald J. Williams, and it became famous and led to a renaissance in the field of artificial neural networks.



The style of neural computation.

Figure 4.1: The Artificial Neural Network (Negnevitsky, 2005)

4.1.1 The Biological model

The artificial neural network is an imitation of the function of the human biological brain. It's using the structure and the function of brain. The human brain is composed of billions of interconnected neurons. Each one of these neurons is said to be connected to more than 10000 neighbour neurons. Figure 4.1 shows a small snip portion of the human brain where the yellow blotches are the body of the neural cells (soma). The connecting lines are the dendrites and axons that connect between the cells. The dendrites receive the electrochemical signals from the other cells and transmit it to the body of the cell. If the signals received are powerful enough to fire the neuron; the neuron will transmit another signal through the axon to the neighbor neurons in the same way. The signal is going also to be received by the connected dendrites and can fire next neoruns (Al-Milli, 2013).

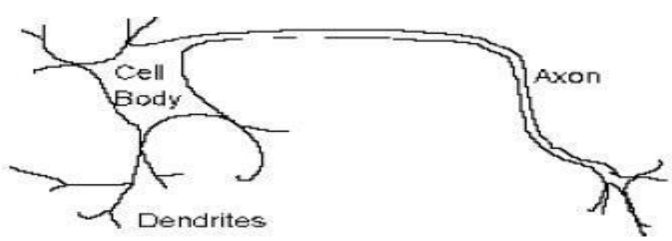


Figure 4.2: The Neuron's Structure (Al-Milli, 2013)

Artificial neural networks are based on the last described model of the biological neural networks. The artificial neural networks still not close to modeling the complex structure of the brain, but they have proved to be efficient in problems that are done easily by humans but difficult for classical computers. An example of these applications is image recognition and prediction based on existing data. Figure 4.1 present the relation and analogy between the human brain and the neural networks.

4.1.2 The Mathematical model

Artificial neural networks are a structure that has inspired its origins from the human thinking centre or the brain. This structure has been inspired and developed to build a mechanism that can solve difficult problems in the science. Most of the structures of neural networks are similar to the biological brain in the need for training before being able to do a required task (Kaki, 2009). Similar to the principle of the human neuron, neural network computes the sum of all its inputs. If that sum is more than a determined level, the correspondent output can then be activated. Otherwise, the output is not passed to the activation function. Figure 4.3 presents the main structure of the artificial neural network where we can see the inputs and weights in addition to the summation function and the activation function.

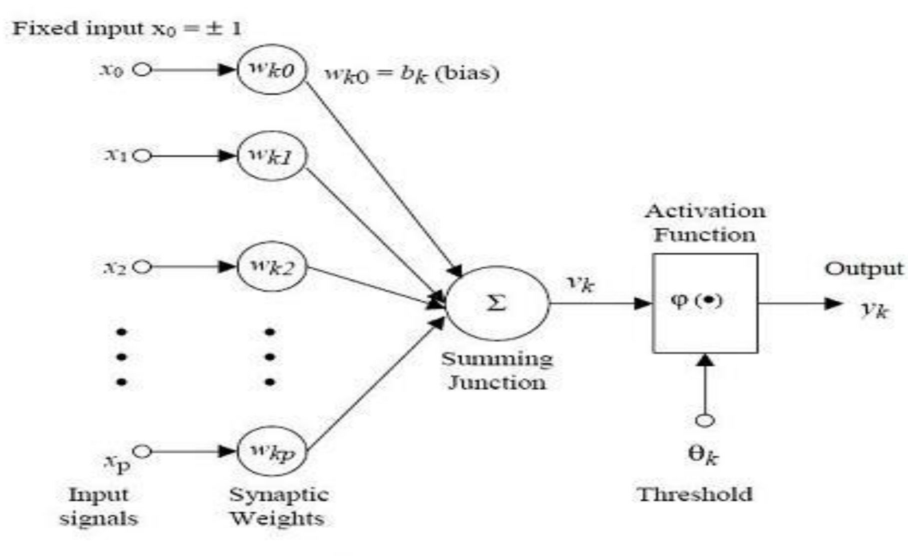


Figure 4.3: The Mathematical model (Kriger, 1996)

4.1.3 The Activation function

When the inputs are fed to the layers through the associated weights and finding the sum of them, an activation or transfer function is used to determine whether the output is to be activated or not. Or in some activation functions, the function is used to determine how much the processed input will share in constructing the total output of the network. Activation functions are very important in neural networks because they can decide whether the input to the neuron is enough to be passed to the next layer or not. There are many types of activation functions in artificial neural networks:

$$\varphi(v) = \begin{cases} 1 & \text{if } v \geq 0 \\ 0 & \text{if } v < 0 \end{cases} \tag{4.1}$$

In the threshold function the output is zero if the summed input is less than certain value of threshold, and 1 if the summed input is greater than threshold. This way the output is oscillating between two values. It can be either activated or deactivated. The function of the hard function is defined by:

$$\varphi(v) = \begin{cases} 1 & v \geq \frac{1}{2} \\ v & -\frac{1}{2} > v > \frac{1}{2} \\ 0 & v \leq -\frac{1}{2} \end{cases} \tag{4.2}$$

This function can range between 0 and 1, but in some cases it can be useful to range it between -1 and 1. The logarithmic sigmoid and hyperbolic tangent is of the most common sigmoid functions. These two functions are the most used in the back propagation because they are differentiable. The formulas of these two functions in addition to the curves are presented below. The slope of the curves can be varied based on the application for which it is used (Kaki, 2009).

$$\varphi(v) = \tanh\left(\frac{v}{2}\right) = \frac{1-\exp(-v)}{1+\exp(-v)} \tag{4.3}$$

4.1.4 Neural network topologies

The network focuses on the pattern of connections between the units and propagation of data. We can make distinction between:

i. **Feed-Forward Neural Networks**, where the data is strictly feed forward from input to output units (Palamiapgan, 2008).

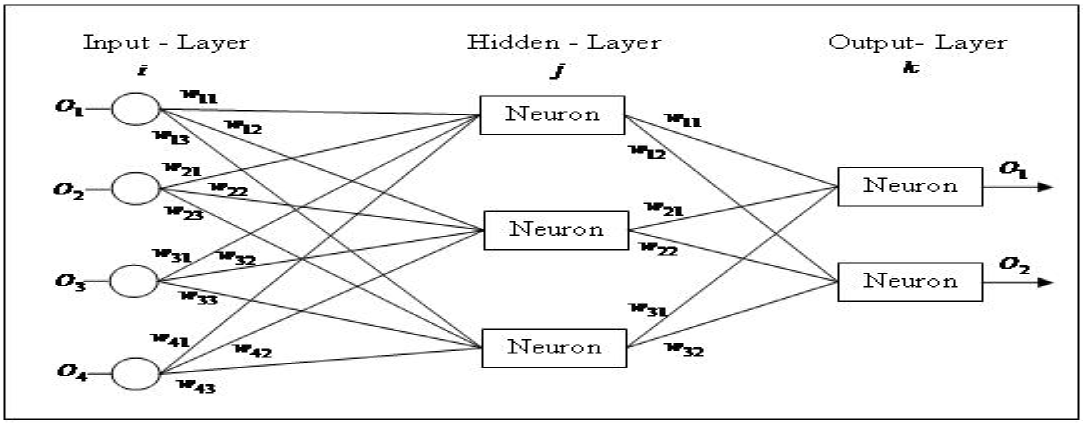


Figure 4.4: The Feed-forward neural networks (Palamiappan, 2008)

ii. **Recurrent Neural Networks** which contain feedback connections. Contrary to feed-forward networks (Haykin, 2000).

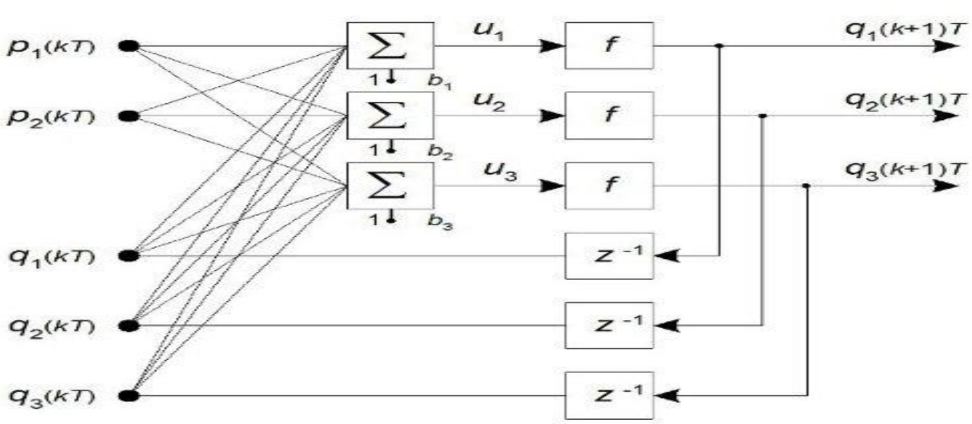


Figure 4.5: The Recurrent neural networks (Haykin, 2000)

Thus, we have three separate matrices of weights:

- Input-to-hidden weights $W_{hx}W_{hx}$
- Hidden-to-hidden weights $W_{hh}W_{hh}$
- Hidden-to-output weights $W_{yh}W_{yh}$

As you may easily guess, the forward propagation equations for this network are quite simple:

$$\begin{aligned} h_i &= \sigma(W_{hh}h_{i-1} + W_{hx}x_i + b_h) \\ \hat{y}_i &= W_{yh}h_i \end{aligned} \tag{4.4}$$

4.1.5 Training of the neural network

Generally, the training of a network is an attempt to lead the network to converge toward desired output or outputs. Two main learning methods are used in teaching the networks. These are the supervised and the unsupervised learning method.

- Supervised learning

The ANN is provided by input data and desired target for this data. The network then updates its weights according to a defined algorithm rule until it converges to a minimum error or reaches a maximum number of iterations. A very important example of the supervised learning method is the error back propagation method.

- Unsupervised learning

In this method, the input data is provided to the network which in turn modifies its weights according to defined conditions.

a) Supervised learning: the network is trained by feeding it with input and matching output patterns.

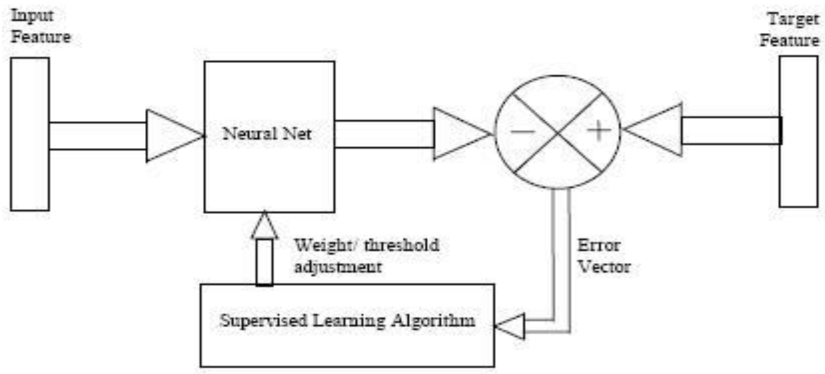


Figure 4.6: The supervised learning neural networks (Oskoei, 2006)

b) Unsupervised learning: In this method, the input data is provided to the network which in turn modifies its weights according to defined conditions (Oskoei, 2006).

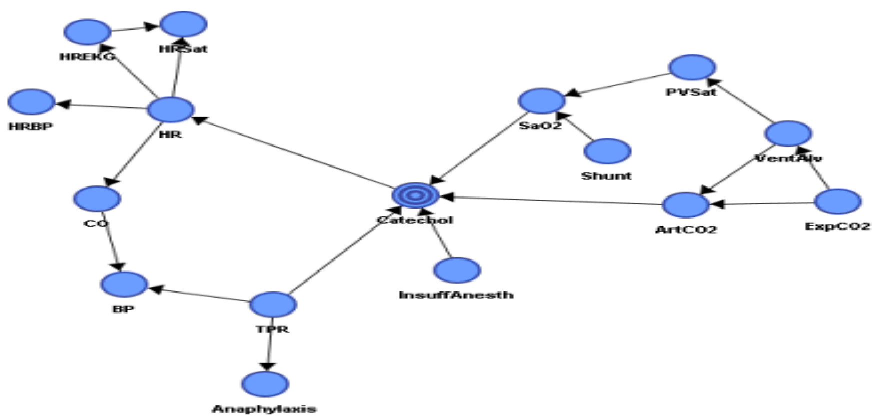


Figure 4.7: The unsupervised learning neural networks (Oskoei, 2006)

c) Reinforcement learning: it is the middle of intermediate kind of the two types of learning; supervised and unsupervised learning (Palamiappan, 2008).

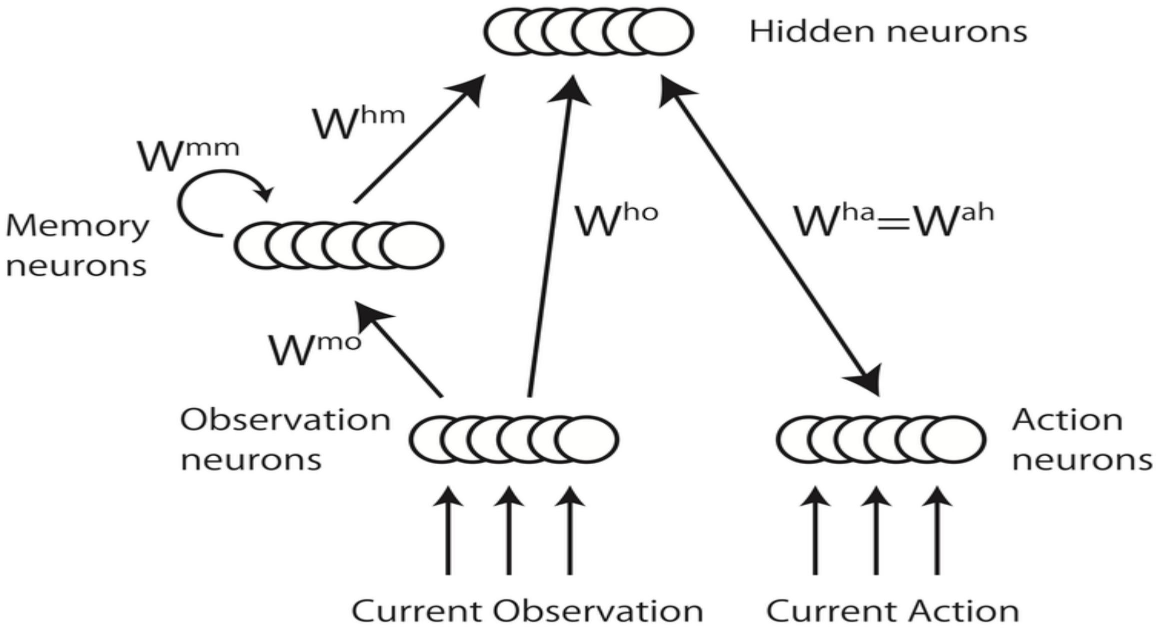


Figure 4.8: The reinforcement learning neural networks (Palamiappan, 2008)

4.1.6 The Simplified neural network model

It consists of two layers of binary neurons (with values 1 and 0).

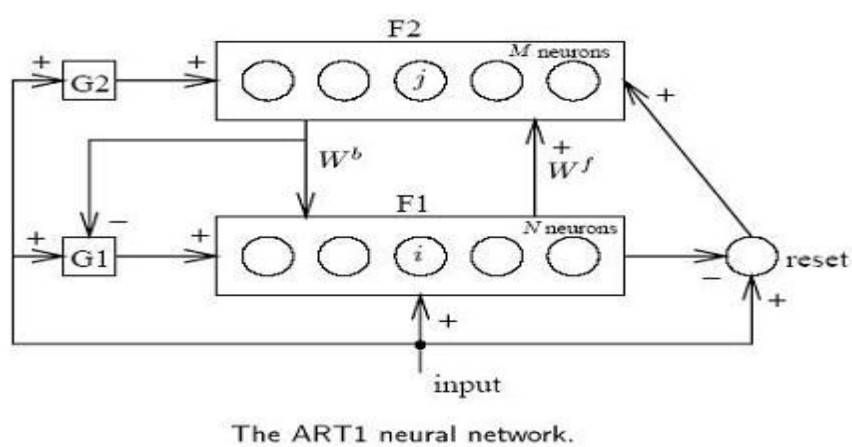


Figure 4.9: The simplified neural networks (Oskoei, 2006)

4.1.7 Applications of artificial neural network

a. Character Recognition using Neural Networks

The character recognition has recently become too significant for our life, which now uses in a lot of fields (Villegas et al., 2006).

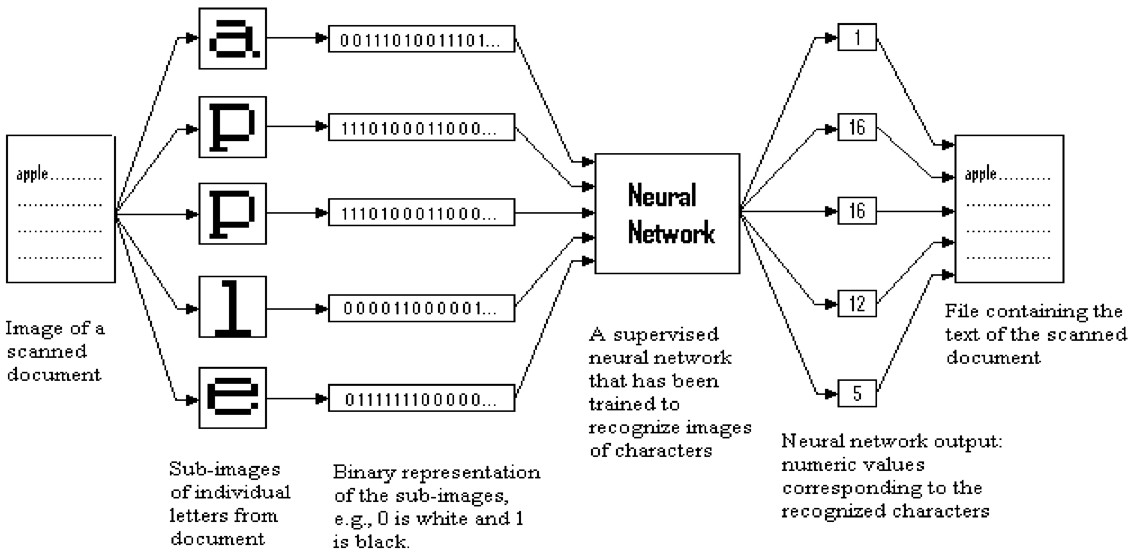


Figure 4.10: The Character Recognition System (Villegas et al., 2006)

b. Image Compression

Neural networks can do both: information receiving and processing to make them meaningful (Kashman & Dimililler, 2005).

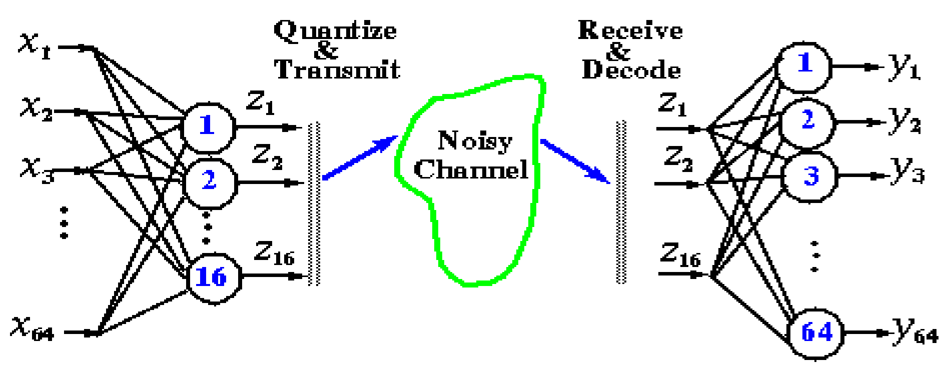


Figure 4.11: The image compression system

c. Stock Market Prediction

Day after day the stock market becomes more complex. various factors weigh will go up or down on any day (Neuro, 2007).

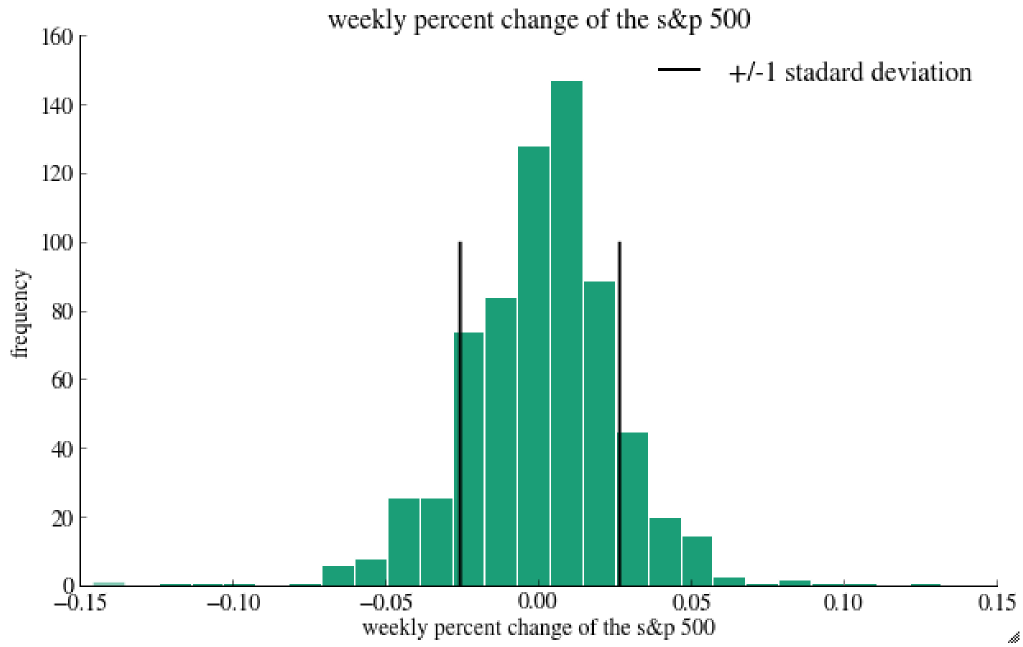


Figure 4.12: The weekly change of the s&p (Neuro, 2007)

d. Travelling Salesman's Problem

Neural networks can solve the travelling salesman problems, only to a certain degree of approximation (Jamil et al., 2012).

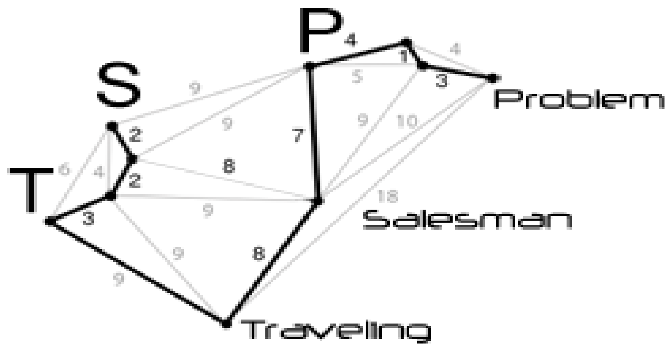


Figure 4.13: The Travelling Salesman's Problem (Neuro, 2007)

e. Medicine, Security, Electronic Nose, and Loan Applications

Which are the some applications that are in their proof-of-concept stage (Neuro, 2007).

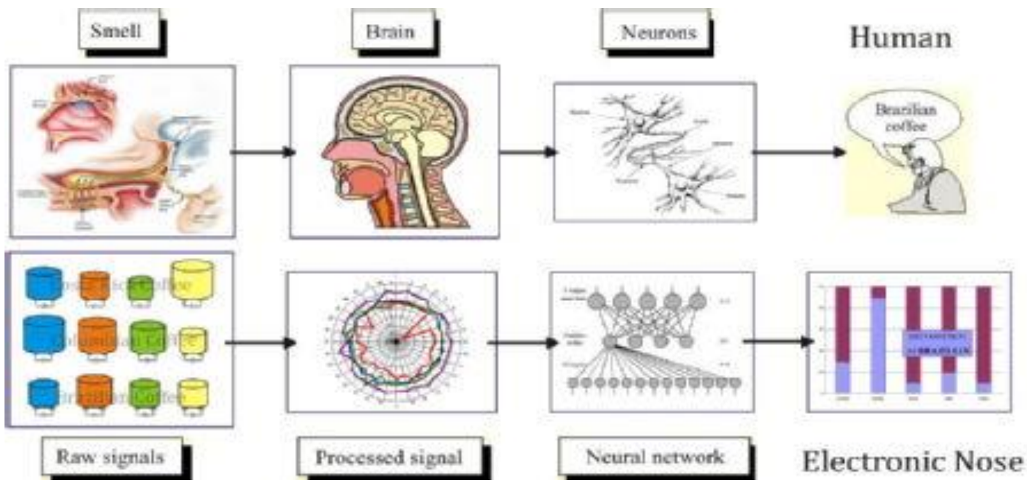


Figure 4.14: The Electronic Nose (Haykin, 2000)

f. Miscellaneous Applications

It is very interesting (albeit at times a little absurd) applications of neural networks (Oskoei, 2006).

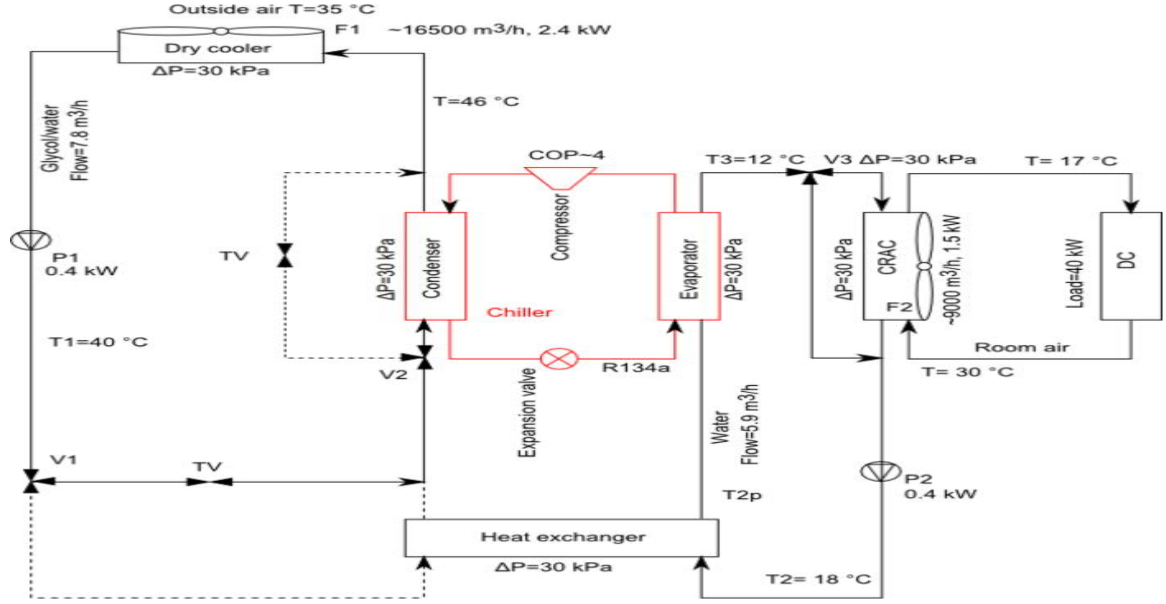


Figure 4.15: The Miscellaneous Applications (Oskoei, 2006)

4.1.8 Types of artificial neural network

Various types of artificial neural networks were found:

a. Feed Forward Neural Network

The feed forward neural system is the most straightforward kind of simulated neural system. The data in this system moves in one and only heading — advances: From the information hubs, the information experiences the shrouded hubs and after that to the yield hubs (Kriger, 1996).

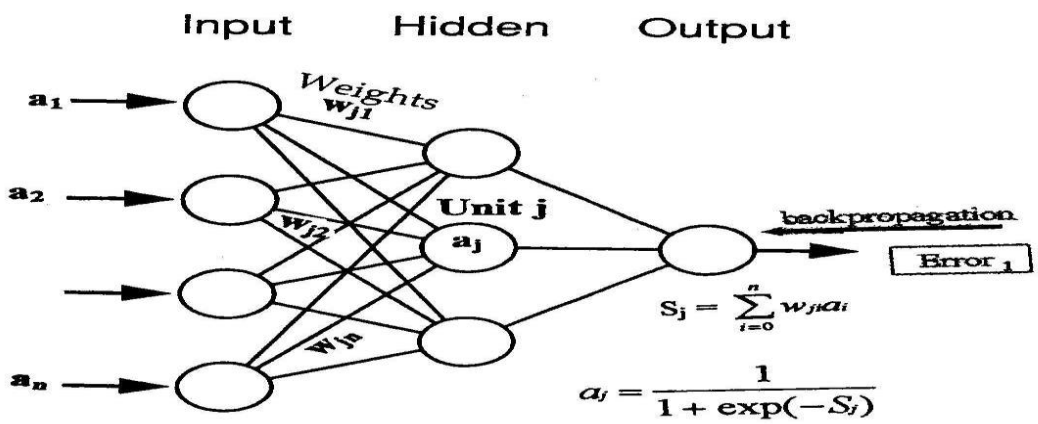


Figure 4.16: The Feed Forward Neural Network (Kriger, 1996)

b. Radial Basis Function (RBF) Network

Radial basis has intense systems for insertion in multidimensional space. A RBF which has incorporated with a separation rule regards to middle (Neuro, 2007).

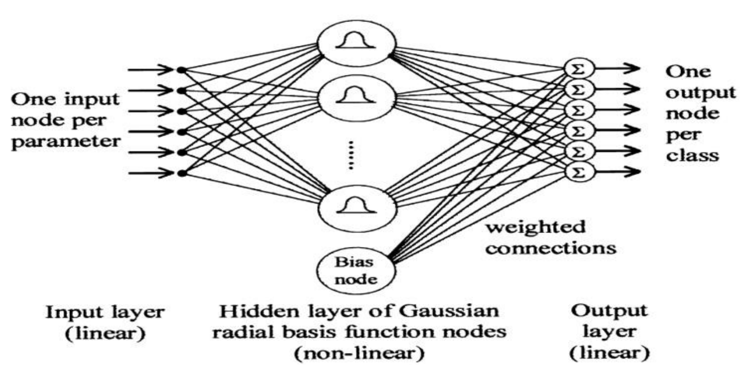


Figure 4.17: The Radial basis function (RBF) network (Haykin, 2000)

c. Kohonen Self-Organizing Network

It is called (The self-sorting out guide), it was developed by Teuvo Kohonen plays out a type of unsupervised learning (Russell et al., 2011).

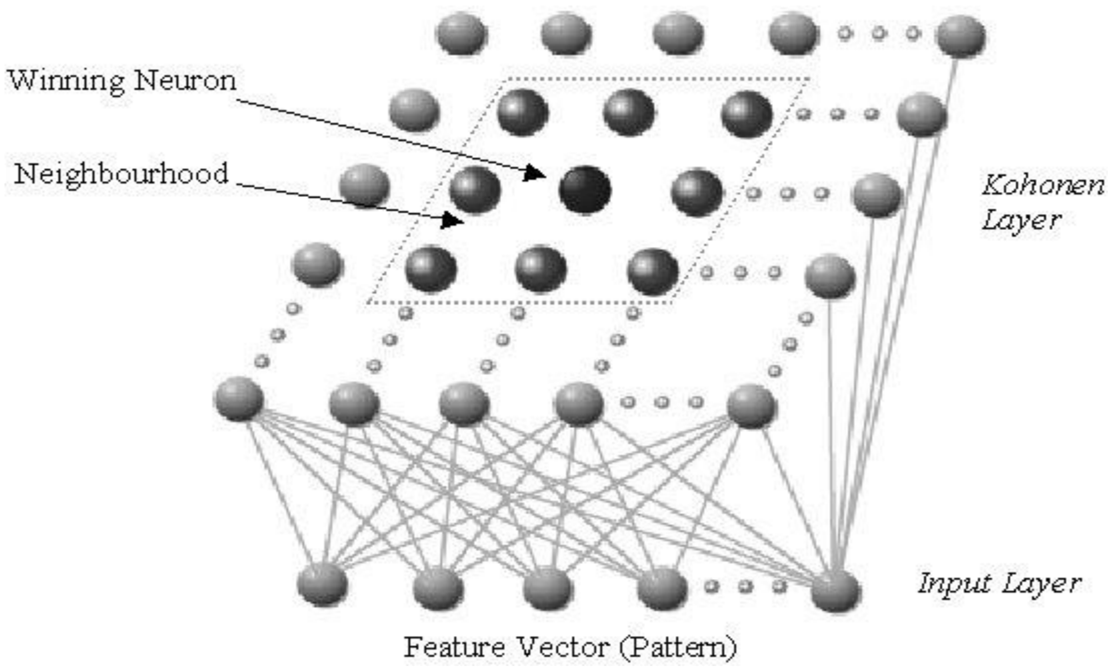


Figure 4.18: The Kohonen self-organizing network (Russell et al., 2011)

d. Learning Vector Quantization

Also it was suggested by Teuvo Kohonen, . In Learning Vector Quantization prototypical representatives of the classes parameterize, together with a distance-based classification scheme. (LVQ) is a neural net that combines both competitive with supervision learning (Russell et al., 2011).

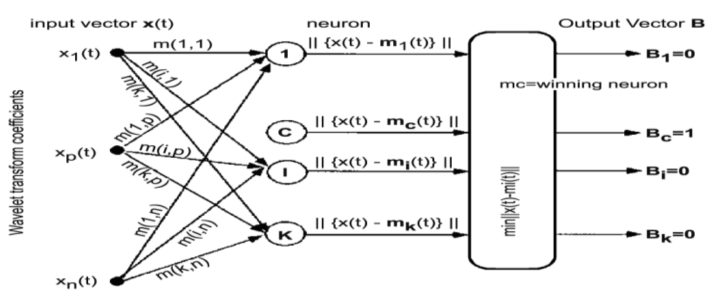


Figure 4.19: The Learning Vector Quantization (Russell et al., 2011)

e. LVQ Algorithm

The essential calculation of the LVQ is a direct technique for moving the Voronoi cell limits to bring about better arrangement. It begins from the information vectors {x} and weights/vectors $\{w_j\}$, and utilizes the order marks of the contributions to locate the best arrangement name for each w_j (Negneviysky, 2005).

f. Recurrent Neural Network

Recurrent neural network (RNN) can be modelled with bi-directional data flow.

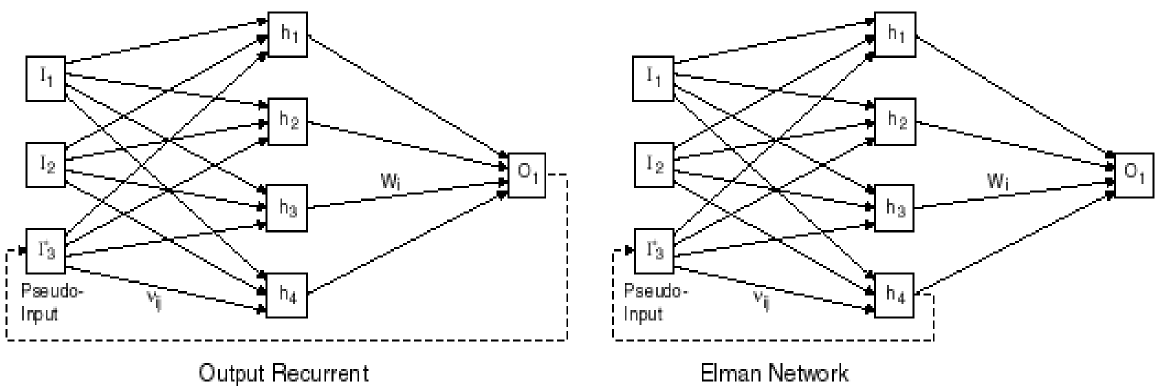


Figure 4.20: The Recurrent neural network (Negneviysky, 2005)

4.2 Summary

In this chapter the Aartificial Neural Network was defined. The ANN become very important for developing new applications, the ANN is acting like natural neural network in human brain, there are a lot of types of ANN, and everyone is different from the other.

CHAPTER 5
INSECT IMAGES ANALYSIS AND PROCESSING

5.1 The Proposed Methodology

The proposed system consists of two main phases which are the processing phase and the classification phase in which the image is classified into one of the 8 types of insects (ant, aphids, spider, caterpillars, earwigs, grasshoppers, mealybugs, or whiteflies). In the image processing phase the images are first pre-processed by some image processing tools such as conversion to grayscale, image smoothing using median filter, and canny edge detection for segmentation. These techniques are done in order to enhance the quality of images and to extract the important features such as geometric shapes of each insects: the wings, legs etc... At the end of the processing stage, the images will be sustained to be fed into the new phase in which they are used as inputs for the backpropagation neural network to be classified as one of the eight different insect types.

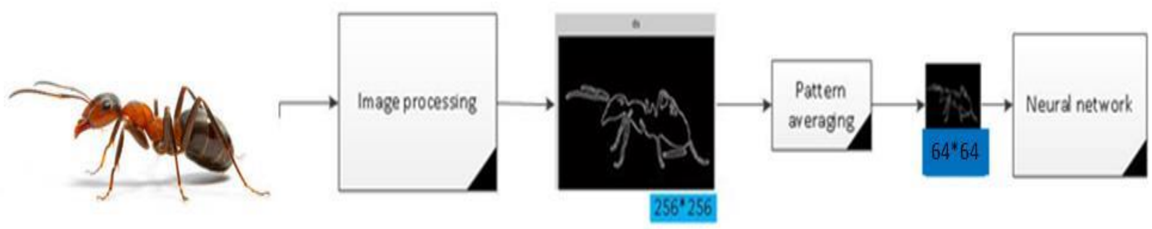


Figure 5.1: System process

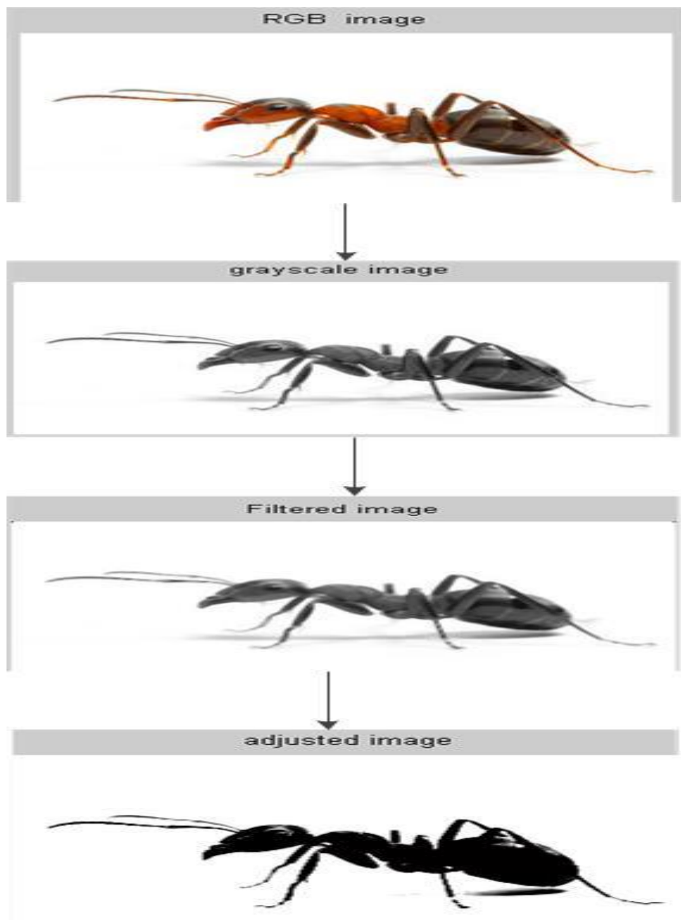
Image processing techniques used:

1. Read RGB images
2. Convert to grayscale
3. Smooth images using median filters
4. Segment images using a canny edge detection technique

6. Extract important features using pattern averaging which is explained in details in the next chapter

7. Classify images using neural network

Fig.5.2 represents a flowchart that illustrates our proposed system for the identification of insects' type. It shows a normal an ant image that undergoes all the system processes in order finally to be processed and finally



'**Figure** **5.2** continued'

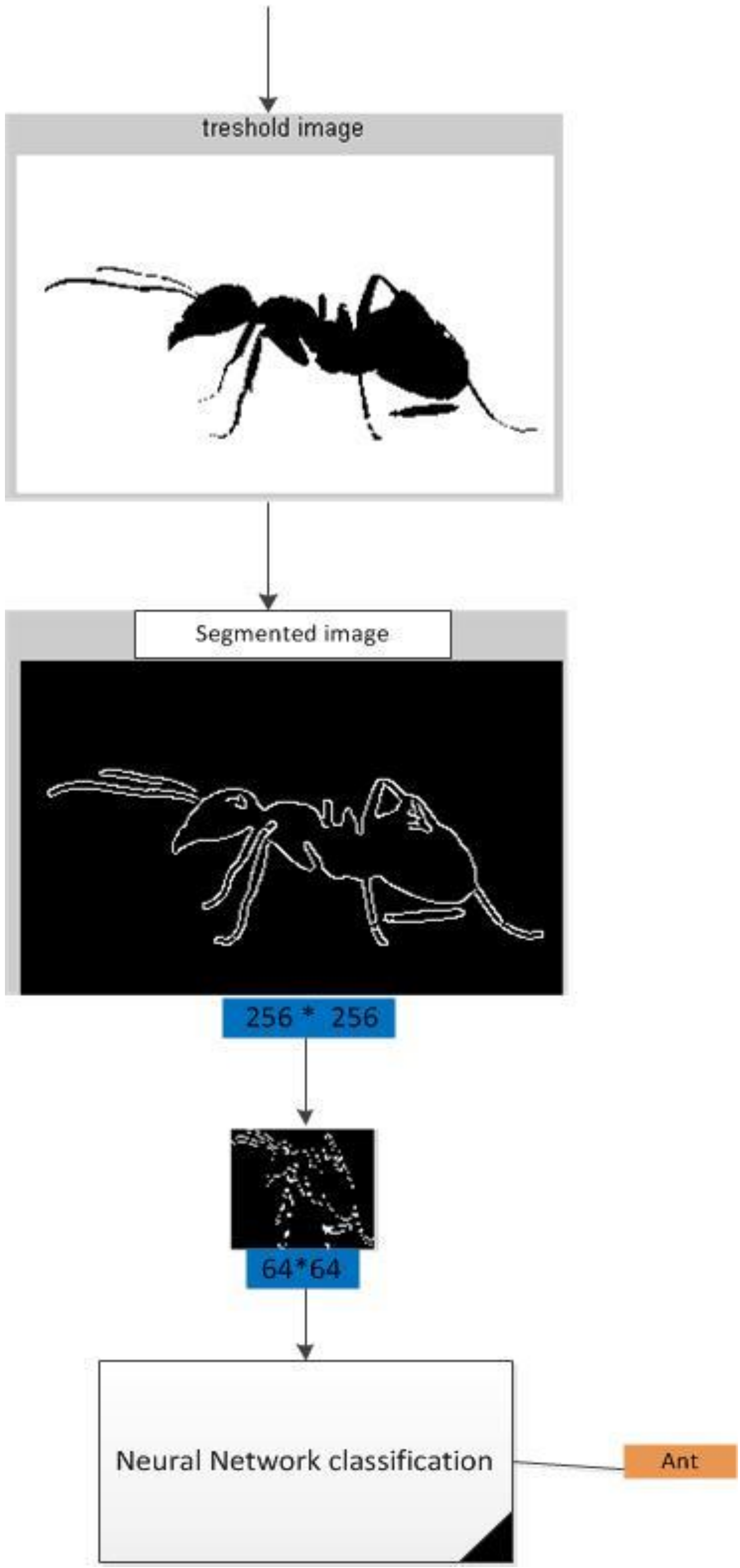


Figure 5.2: Flowchart of the developed intelligent insects' recognition system

5.2 Dataset

Due to the unavailability of one source images of insects, the insect's images were collected from different website for insect's classification. The images were obtained all in different size owing to the fact that some of these images were better than others in resolution and quality of images. One online database was used in this work. The total number of images in the database is 232 images; 29 for each class of insect. The images are divided as following: 29 are ants, 29 are aphids, 29 are caterpillars, 29 are earwigs, 29 are grasshoppers, 29 are mealybugs, 29 are spiders, and 29 are whiteflies. The two networks were first trained using 10 images of each insect type. Hence, 80 images were used for training the networks; 10 images for each class type. Different learning scheme was also used to test the same networks where 80 images were used for training the network and the rest were used for testing purposes. For the testing phase two schemes were used: 80:5 and 80:14. This means that the first scheme is to use 80 images for training and 5 for testing for each insect type. The second scheme is to use also 80 images for training but 14 for testing. Among them, 10 are ants, 10 are aphids, 10 are caterpillars, 10 are earwigs, 10 are grasshoppers, 10, are mealybugs, 10 are spiders, and 10 are whiteflies for training. For testing scheme one: 5 are ants, 5 are aphids, 5 are caterpillars, 5 are earwigs, 5 are grasshoppers, 5 are mealybugs, 5 are spiders, and 5 are whiteflies. For testing scheme two: 14 are ants, 14 are aphids, 14 are caterpillars, 14 are earwigs, 14 are grasshoppers, 14 are mealybugs,14 are spiders, and 14 are whiteflies. Table 5.1 shows the number of total images of the two databases used for training and testing of the designed system.

Table 5.1: Dataset

Type of insects	Database	Training ratio	Testing ratio 1	Testing ratio 2
Ants	29	10	5	14
Aphids	29	10	5	14
Caterpillars	29	10	5	14
Earwigs	29	10	5	14
Grosshoppers	29	10	5	14
Mealybugs	29	10	5	14
Spiders	29	10	5	14
Whiteflies	29	10	5	14
Total	232	80	40	112

The Figure below shows a sample of the different types of insects used in this thesis

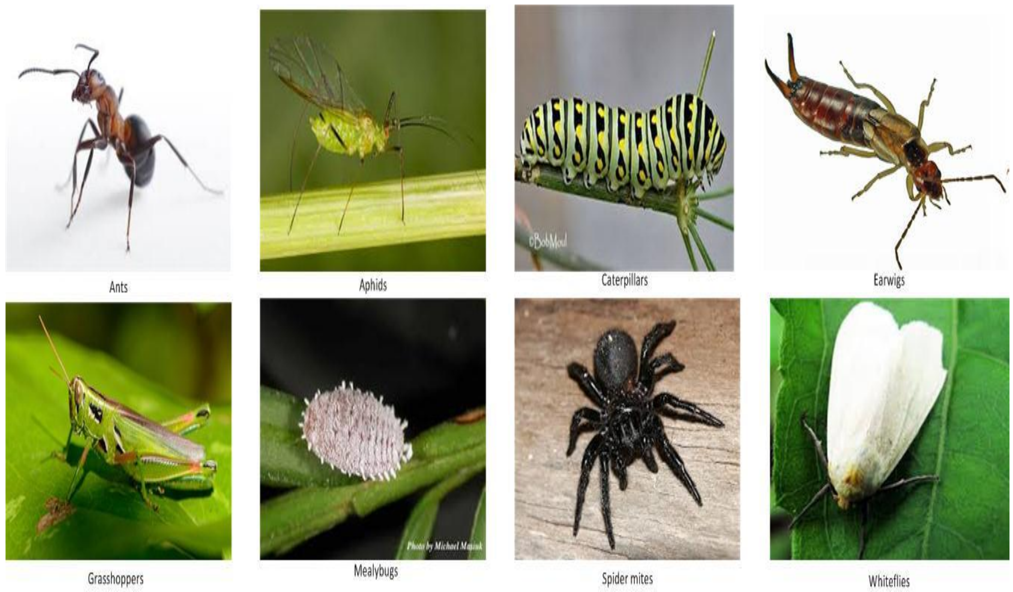


Figure 5.3: Sample of database images

5.3 Image analysis and processing

The images were enhanced for adequate identification through a series of image processing methods. These methods constitute the adequacy of the system in the processing phase.

5.3.1 RGB to grayscale conversion

This technique is used to convert the images from RGB to grayscale by using the luminosity method. This method is a more sophisticated version of the average method. It also averages the values of the image matrix, but it forms a weighted average to account for human perception since humans are more sensitive to green than other colors, therefore; green is weighted most heavily. The formula for luminosity is

$$0.21\ R + 0.72\ G + 0.07\ B \tag{5.1}$$

which relies on the contribution of each color of the three RGB colors. Using this method, the grayscale image is brighter since the colors are weighted according to their contribution in the RGB image not averagely (Church et. al).

Figure 5.4 illustrates the conversion of an abnormal RGB image into a grayscale image using luminosity method.

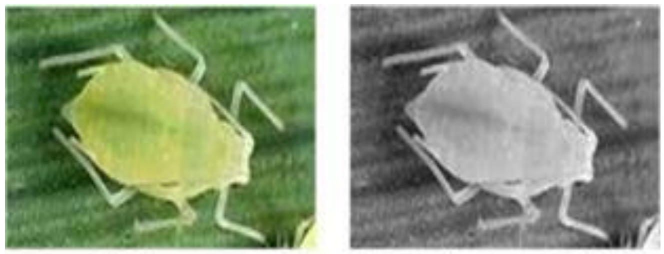


Figure 5.4: Grayscale conversion

5.3.2 Image smoothing using median filtering

Smoothing, so called blurring, is an adjustment and enhancement technique used to reduce the noise in an image to produce less pixelated and clearer image. Most smoothing techniques are based on low pass linear filters. It is mostly based on the averaging technique of the input image or the middle (median) value technique (Church et. al).

To perform a smoothing operation we will apply a filter to our image. The most widely recognized kind of channels is the non-linear filters, for example, median filter which is utilized as a part of our proposed framework. This filter is utilized to diminish indiscreet clamor or the salt-and pepper in a picture with safeguarding the helpful elements and picture edges. Middle separating is a direct procedure in which the yield of the being handled pixel is found by figuring the middle of a window of pixels that encompasses that concentrated on pixel. In short the median filter experiences every component of the picture and supplant every pixel with the middle of its neighboring pixels which are situated in a square neighborhood (piece) around the assessed pixel.

Figure below illustrates an example of a median filter and its mechanism to reduce the noise in an image by setting a kernel or window that goes through the whole matrix and find an output for the processed pixel by calculate the median of the pixels in the window (Church et. al).

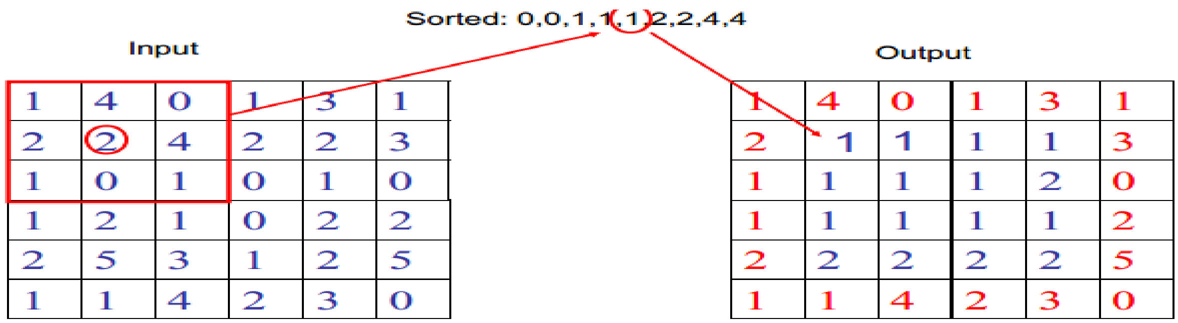


Figure 5.5: Median filter working principles

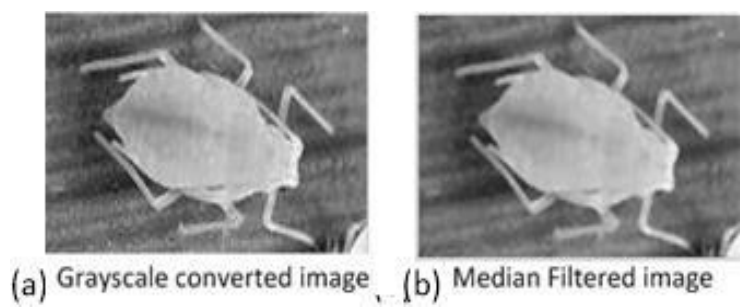


Figure 5.6: Median filtering

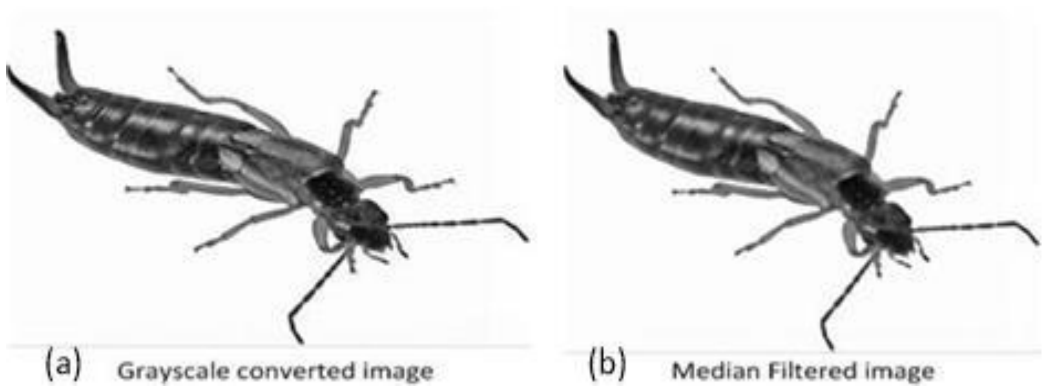


Figure 5.7: Median filtering of earwig

5.3.3 Gamma correction in medical imaging

With the end goal of expanding the image intensity and improve its resolution, the images experience intensity adjustment technique which is used to upgrade the difference of the image by expanding the intensity of its pixels or change the range of highest and lowest intensity in an image. During this operation, the intensity estimation of every pixel in the information image is changed utilizing an exchange capacity to frame a complexity balanced image (Gonzalez & Woods, 2004).

Ankit Aggarwal, R.S. Chauhan and Kamaljeet Kaur developed a system for the adaptive image enhancement technique preserving brightness level using gamma correction. Their proposed technique is that the weighted average of the histogram leveled, gamma corrected and the first picture are consolidated to acquire the upgraded processed image. The proposed calculation accomplish contrast enhancement as well as preserve the brightness level of images (Aggarwal et al., 2013).

In our case we selected the four parameters and gamma values in a way to obtain the best quality and contrast image.

The figure below represents the adjustment of an image and its effects in enhancing the image contrast.

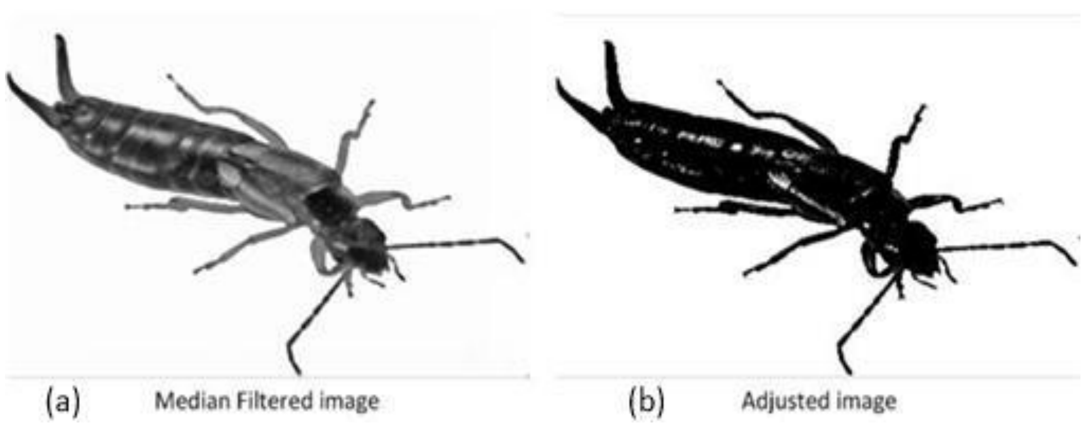


Figure 5.8: Adjusted image of an earwig

We can notice from the above image that the image adjustment operation has a great effect in enhancing the contrast and brightness of the image, so it is clearer and its features are more bright and shown. This helps in detecting the edges and features of the image in the next process.

5.3.4 Tresholding

Thresholding is the separation of region of images into two regions. One region corresponds to the foreground region, in which it contains the objects that we are interested in. The other region is the background, corresponds to the unneeded objects. This provides segmentation of the image based on the image different intensities and intensity discontinuities in the foreground and background regions. The input of this method is usually a grayscale or color image, while the output is a binary image representing the segmentation. The black pixels refer to background and white pixels refer to foreground. The segmentation is achieved by a single parameter known as the intensity threshold. This is set by analyzing the histogram of the image which represents the intensity distributions of the image. During Thresholding, each pixel is compared to that threshold value. If the pixel value is greater than that threshold, then this pixel is considered as
26 foreground pixel (white). If the pixel value is lower than that threshold value, then the pixel is considered as background pixel (black) (Gonzalez & Woods, 2002).
Figure 5.9 illustrates a breast cancer image that undergoes thresholding.

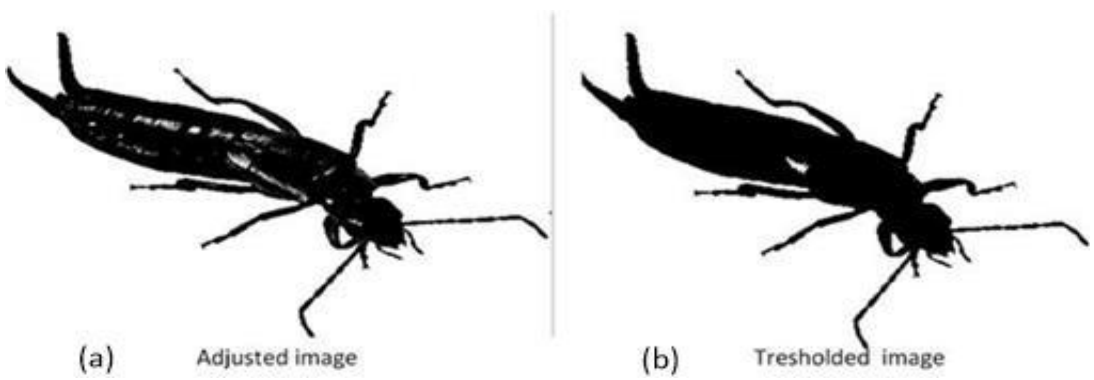


Figure 5.9: Tresholding

5.3.5 Canny edged based Segmentation

Segmentation can be defined as grouping of the image parts into many regions. The goal of such image processing operation is to represent some meaningful and needed areas of the image, such as tumors, faces etc…

In other words, the segmentation is to combine the interesting regions in the image to make two different areas: foreground regions of interest and background regions. The background region is to be ignored using some techniques such as tresholding. Thus, the pixel values that are lower than the threshold are considered as 0's (black or background), while the pixel values higher than the threshold are considered as 1's (white or foreground) (Shapiro & Stockman, 2004).

Pixel edges are associated with some intensity changes or discontinuities; therefore, edge detection is the process of identifying such sharp intensity contrasts (i.e., discontinuities) in an image. Classical edge detection operators Sobel and Prewitt uses 3×3 kernels which are convolved with the original image to calculate approximations of the derivatives - one for horizontal changes, and one for vertical. In this developed framework, edges were identified utilizing canny edge operators. This system is the most popular recognized and utilized technique for identifying edges and fragmenting the image. The Canny edge identifier is considered as one of the best as of now utilized edge finders since it gives great commotion insusceptibility and recognizes the genuine edges or intensity discontinuities while keeping a minimum error. Canny operator has been used for such algorithm with regard to the following criteria (Saif et al., 2012):

1. To maximize the signal-to-noise ratio in an image.
2. To ensure that the detected edge is localized as accurately as possible.
3. To minimize multiple responses to a single edge.

The steps of Canny algorithm in order to segment an image into many regions are as follows:

1. Smoothing: it means blurring an image in order to remove.

2. Finding gradients: Since the edges must be highlighted in the places where the computed gradients has large magnitudes, we have to find the gradient of the image by feeding the smoothed image through a convolution operation with the derivative of the Gaussian filtering both the vertical and horizontal directions.

Its magnitude value can be obtained using the following formula:

$$|G| = |Gx| + |Gy| \tag{5.2}$$

Where Gx and Gy are the gradient values computed by using Sobel mask in x and y direction respectively and edge direction can be computed by using the following formula

3. Non-maximum suppression: this technique is to find the local maxima in the direction of the gradient, and suppress al others, in order to minimize the false edges.

4. Perform double thresholding: in this stage, the potential or strong edges are found using thresholding, therefore, instead of using a single static threshold value for the entire image, the Canny edge algorithm provided what is called hysteresis thresholding, in which two threshold values are used. These two threshold levels are T1 or high threshold and T2 or the low threshold, where T1>T2.

5. Final Edge tracking by hysteresis: this is the final stage, in which the final edges are tracked by suppressing all edges that are not connected to a very strong edge (Saif et al., 2012).

These are high threshold and low threshold. The pixel having the quality more prominent than high threshold is situated as an edge pixel and the pixel having the worth more noteworthy than the low limit and is having a way to the edge pixel is safeguarded though pixel having slope esteem more noteworthy than low edge and is not associated with edge is stifled.

Figure 18 illustrates segmentation of a background image using canny edge detection.

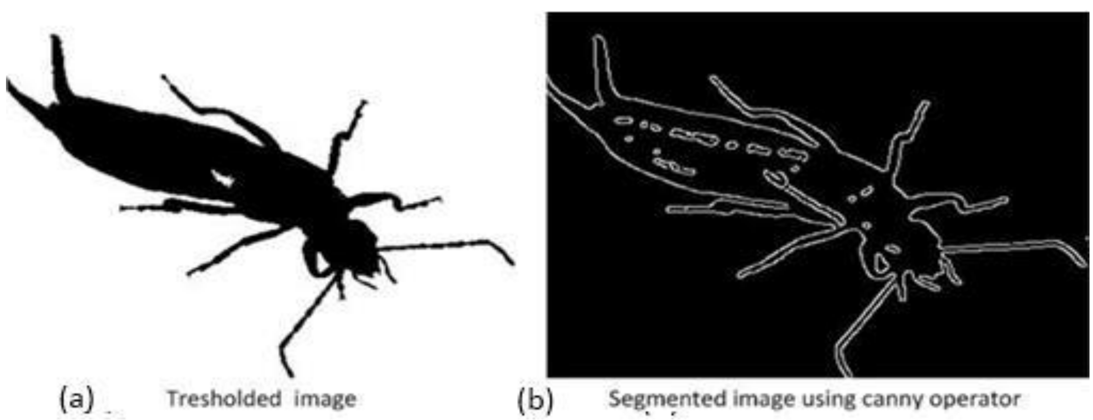


Figure 5.10: Segmentation using canny edge detection

Figure.5.10 illustrates segmentation of the adjusted image using canny edge detection by analyzing and visualizing these two segmented images, we can notice that the segmentation result is better when segmenting the adjusted image since the edged are clearer.

Advantages of using canny operators:

- It reduces immunity in an image.
- It detects the true edges or intensity discontinuities.
- It smooths edges
- It is in a completely different class from Sobel and Laplacian. It is much smarter and accurate since it includes a bunch of post processing whereas Sobel and Laplacian are simply high pass filter outputs followed by linear binary thresholding (Saif et al., 2012).

5.3.6 Features extraction and rescaling using pattern averaging

After the segmentation process using the canny edge detection, the images size ought to be lessened keeping in mind the end goal to be fed into the neural system. To decrease the span of images while keeping the valuable and required elements extricated by the already utilized techniques, we utilized patter averaging. This stool is characterized as taking the average of the characterized sections of the image by choosing a kernel of 4*4 fragments that are found the middle value of. Consequently, each concentrated pixel is then computed as the average of the 16 neighbor's pixels in the chose window. Consequently, we think of an averaged image with similar elements and properties of the first one for the reasons for quick preparing and simple figuring.

An intelligent blood cell identification system was developed by (Khashman, 2008) for the identification of the three blood cells. The authors used this technique to reduce the size of the blood cell images while preserving the needed features.

Below in Figure.5.11 is shown some of processed rescaled images.

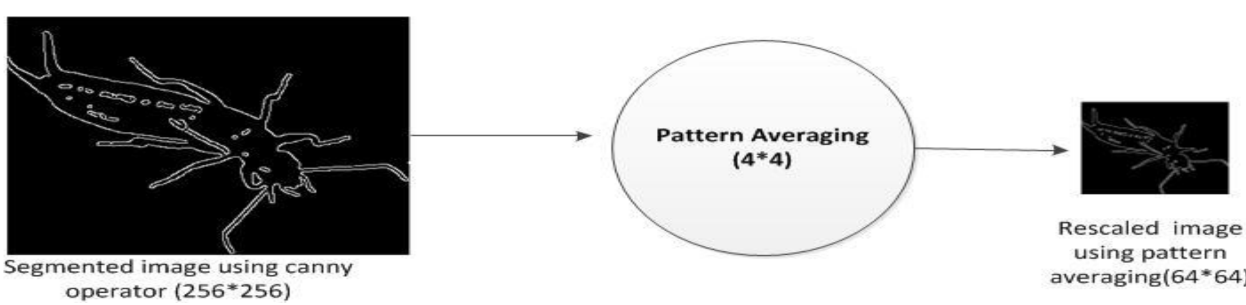


Figure 5.11: Pattern averaging

Pattern averaging is used to take the average feature vector from an image which represents the same features but with smaller size comparing to the original image. Averaging is a straightforward however effective technique that makes "fluffy" examples when contrasted with multiple"crisp" designs, which furnishes the neural system with important learning while decreasing processing cost. Also, design averaging defeats the issue of shifting pixel values inside the divided squares as an aftereffect of pivot or fluctuating scale, along these lines, giving a turn and scale invariant framework (Khashman, 2009).

Highlight extraction in this work utilizes normal estimations of non-cover sectioned image squares. Each arrived at the midpoint of square esteem is encouraged into a neuron in the system's input layer. In this way, the decision of piece size influences the aggregate number of squares speaking to an input image, and thus the quantity of input layer neurons. We utilize two piece sizes in this work: (4 × 4 and 8 × 8) pixels. In the event that by occurrence, images of two unique cells have similar worldwide dim level circulation (histogram), then their worldwide normal esteem will be indistinguishable, however their neighborhood pieces normal qualities will be distinctive, and subsequently the two cells can be isolated and distinguished by the neural system because of the utilization of nearby square normal values in this stage. Considering the measure of the input images (256 × 256) pixels, two piece sizes have been utilized and normal qualities speaking to the image were acquired, in this way bringing about 4096 normal qualities (utilizing 4 × 4 pixel squares) and 1024 normal qualities (utilizing 8 × 8 pixel squares). These piece sizes were picked in light of the fact that they give adequate representation of input images for significant learning while keeping up the extent of the neural system as little as could reasonably be expected, so as to decrease computational costs. The got arrived at the midpoint of qualities will be utilized as the input to the system for both preparing and testing.

5.4 Artificial Neural Network

Artificial neural network can be defined as a system consists of interconnected simple computational units called neurons or cells. It is an attempt to mimic the structure and function of the brain (Zurada, 1992).

The associated neurons are connected by links, and every link has all its numerical weight associated with it. Weights are the primary means of long-term memory in Artificial Neural Networks. The von Neumann's computer model is obviously faster and more accurate in computing but its lacks flexibility, and noise tolerance; it cannot always deal with incomplete data (Negnevitsky, 2005). The most important is the inability to raise the level of performance over time from experience. i.e. incapable of learning.

5.4.1 Backpropagation neural network algorithm

The back propagation training algorithm uses a feed forward process, a back propagation updating method, and supervised learning topology. This algorithm was the reason of neural networks development in the 80s of the last century. Back propagation is a general purpose learning algorithm. Although it is very efficient, it is costly in terms of processing requirements for learning. A back propagation network with a given hidden layer of elements can simulate any function to any degree of accuracy (Gupta, 2006).

The back propagation algorithm is still as simple as it was in its first days. That is due to its simple principle and efficient algorithm. The input set of training data is presented at the first layer of the network, the input layer passes this data to the next layer where the processing of data happens. The results after being passed through the activation functions are then passed to the output layers. The result of the whole network is being then compared with a desired output. The error is used to make a one update of the weights preparing for a next iteration. After the adjustment of the weights, the inputs are passed again to the input, hidden, and output layers and a new error is calculated in a second iteration and vice versa.

The mentioned process continues until achieving an acceptable level of the error so that the network can be considered has learned. Figure 5.9 presents the structure of the network with layers and back propagation process (Al-Milli, 2013).

- **Modeling of back propagation algorithm**

The back propagation is an algorithm that uses the theory of error minimization and gradient descent to find the least squared error. Finding the least squared error imposes the calculation of gradient of the error for each iteration. As a result, the error function must be continuous derivable function. These conditions lead to the use of continuous derivable activation functions as they are the precedents of error calculation. In most of cases, the tangent or logarithmic sigmoid functions are used (Dimililler, 2012). The sigmoid function is defined by:

$$o(x) = \frac{1}{1+e^{-ax}} \tag{5.3}$$

Where the variable a is a constant controlling the slope of function. Where the derivative of the sigmoid function is given by:

$$o^{\backslash}(x) = f(x)(1 - f(x)) \tag{5.4}$$

The equations describing the training of the network can be divided into two categories:

1- Feed forward calculations: used in both training and test of the network.
2- Error back propagation: used in training only.

In the feed forward process, the output or total potential can be given by:

$$TP = \sum x_n \omega_n + b_n \tag{5.5}$$

Where, x_n is the input vector, w_n is the weight matrix, and b_n is the bias values vector. The total potential obtained in each layer must be passed by an activation function. The activation function can be either linear or non-linear function. An example of a linear function that is mostly used in neural networks is the sigmoid function given in equation (5.6). Another example is the tangent sigmoid given by:

$$o(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{5.6}$$

It is important to notice that this function is also continuous and derivable. The derivative of this function is given by:

$$o^{\backslash}(x) = 1 - \frac{(e^{x} - e^{-x})^2}{(e^{x} + e^{-x})^2} \tag{5.7}$$

The output of the last activation function is the actual output of the neural network. This output is then compared with the goal of training to generate the error signal. The error signal is actually defined by equation (5.8). The goal of the training of neural network is always to minimize that error.

$$E = \sum (T - o)^2 \tag{5.8}$$

Where, T signifies the target output. An error function is then defined based on the value of E such that:

$$\Delta_j = (T_j - o_j)o_j(1 - o_j) \tag{5.9}$$

This value is propagated back to the network using the next equations to update the weights and biases of the different layers. The weights are then updated using the next equation:

$$\omega_{jhnew} = \omega_{jhold} + \eta\Delta_j o_h + \alpha(\delta\omega_{jhold}) \tag{5.10}$$

Concerning the hidden layers, their weights are updated using the error update defined by:

$$\Delta_h = o_h(1 - o_h)\sum \omega_{jh}\Delta_j \tag{5.11}$$

The new weights values are then given by:

$$\omega_{hinew} = \omega_{hiold} + \mu\Delta_h o_i + \alpha(\delta\omega_{hiold}) \tag{5.12}$$

The values of α and η are the well-known momentum factor and learning rate. At the end of weights update, a new feed forward iteration is done again. The error is being calculated at each iteration until it arrives to an accepted error value.

5.5 The Classification Phase

In this phase, the images of the different insects are classified into 8 classes using a supervised backpropagation neural classifier. One database was used for training and testing the different networks. However, two learning schemes were used. Learning scheme 1 consists of 80 images; 10 for each class of insect for training and 5 images for each insect for testing, 80:5. Similarly learning scheme contains 80 images; 20 for each class of insects were used for training but 14 of each insect were used for testing, 80:14. The two networks were first trained using the database. Hence, 80 images were used for training the networks; 10 images for each class type. The remaining images were used for testing purposes.

The input layer of the BPNN1 organize comprises of 4096 neurons since every image is rescaled to 64*64 bitmap utilizing design averaging. The hidden layer comprises of 40 neurons, while the yield layer has 8 neurons since we have just 8 yield classes: ants, aphids, caterpillars, earwigs, grasshoppers, mealybugs, creepy crawlies, and whiteflies. BPNN2 includes 1024 neurons since every image is rescaled to 32*32 bitmap utilizing design averaging

Figure 5.12 demonstrates the neural system topology of our proposed recognizable proof framework for the BPNN1 and BPNN2.

Table 5.2 shows to the input parameters setting of the framework Table 5.3 demonstrates every one of the parameters utilized when preparing the systems with the database training images. The systems kept running for 5000 greatest iterations number with a learning rate of 0.3, a momentum rate of 0.5 and a base mistake of 0.001.

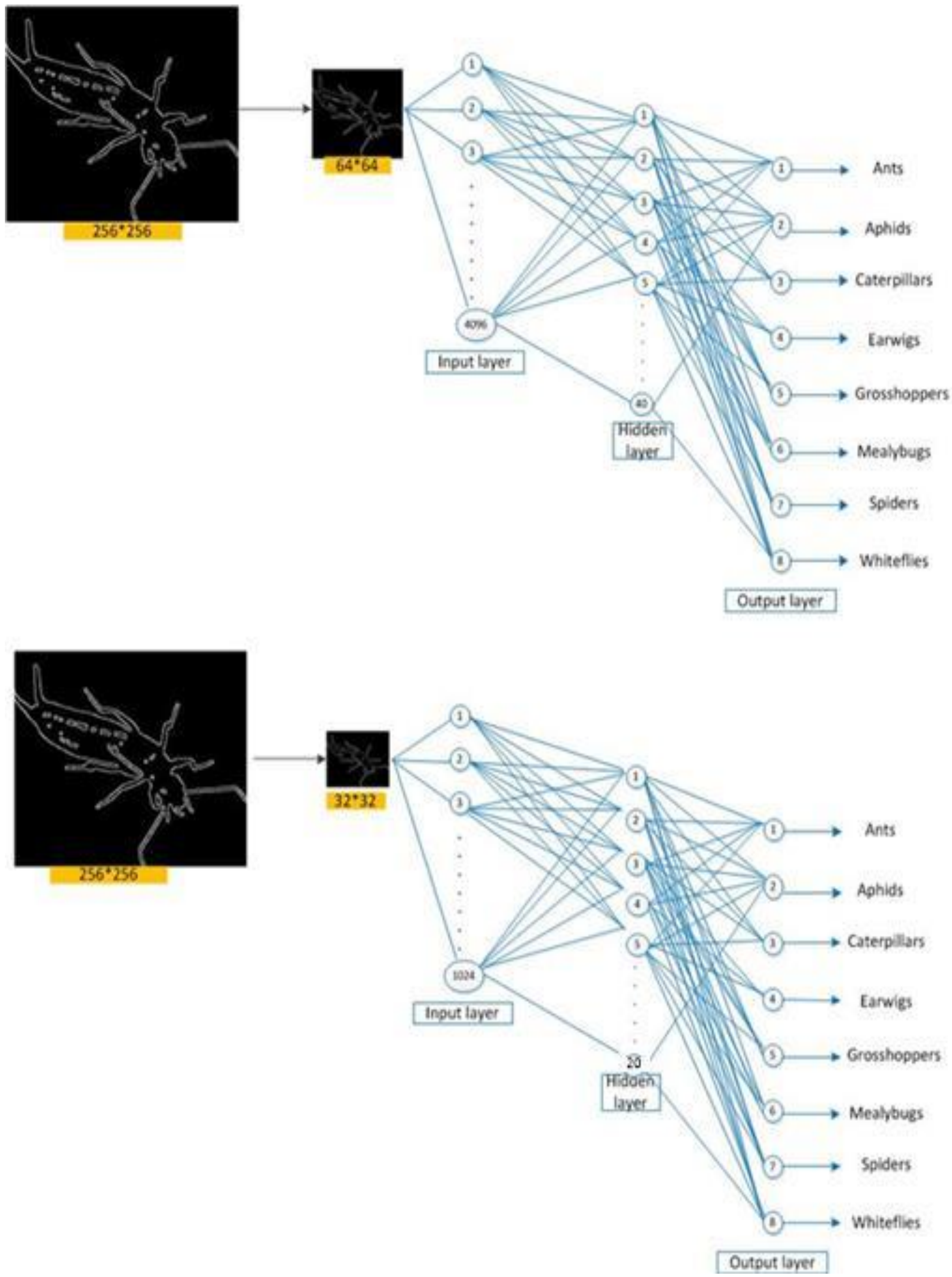


Figure 5.12.: BPNN topologies for BPNN1 and BPNN2

Table 5.2: Input parameters of BPNN1

Parameters	Value
Number of neurons in input layer	4096
Number of neurons in output layer	8
Number of neurons in hidden layer	40
Maximum Iteration number	5000
Learning rate	0.3
Momentum rate	0.4
Error	0.01
Activation Function	Sigmoid

Table 5.3: Input parameters of BPNN2

Parameters	Value
Number of neurons in input layer	1024
Number of neurons in output layer	8
Number of neurons in hidden layer	20
Maximum Iteration number	5000
Learning rate	0.3
Momentum rate	0.5
Error	0.01
Activation Function	Sigmoid

5.5.1 The system training

As previously discussed both networks were trained using 80 images from database. The network was simulated and trained on Matlab software and tools. We used eight different set of images of eight insects. The following is the training results of the two sets (learning curve) for both BPNN1 and BPNN 2 when trained with 80 images of 8 different insects.

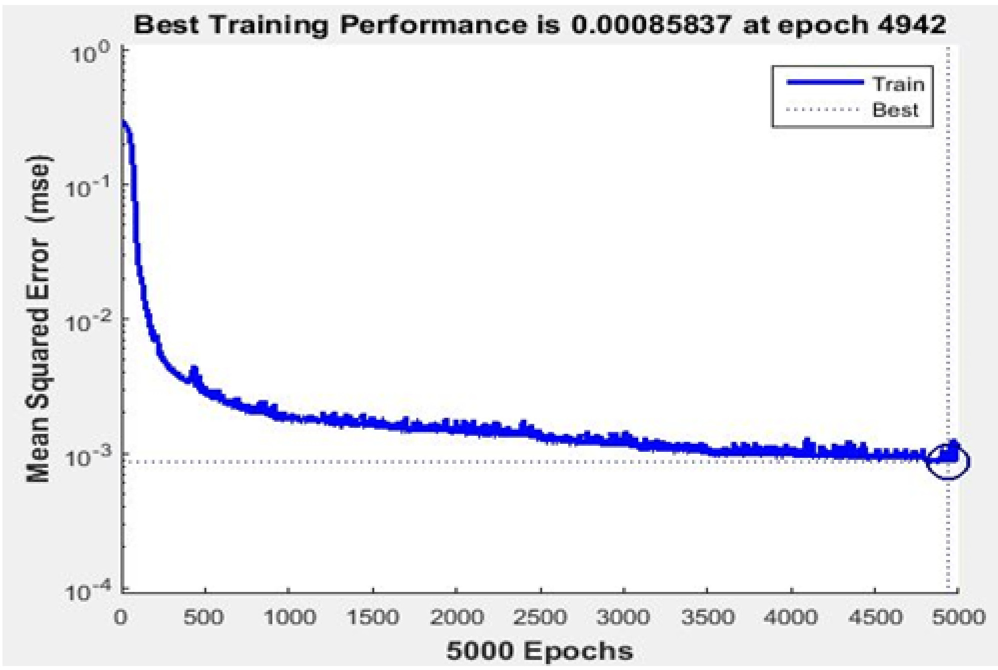


Figure 5.13: Learning curve for BPNN1

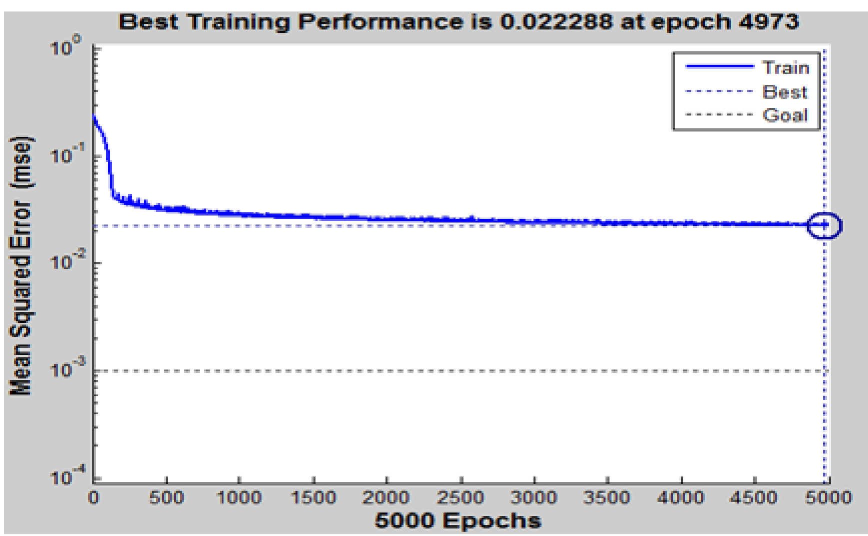


Figure 5.14: Learning curve for BPNN2

The Figures.5.13 and fig. 5.14 shows the learning curve of the two backpropagation neural networks when trained using the database images. It is remarkable that the BPNN1 learned better than the BPNN2 where they achieved 0.0008 and 0.022 respectively. This can be due to the larger size of the input data in the first one, where the averaged image size was 4096 for BPNN1. However, the averaged image's size used in the BPNN2 was 1024 which yields less convergence and learning of network.

5.6 System Performance

This work presents an intelligent 8 insects identification framework established image processing and neural classification. The images of insects are pre-processed first for the purpose of extracting their patterns of interests which represent their geometric shape using various image processing techniques. The segmented shapes of insects bear sample averaging to rescale them while keeping the extracted features.

This identification system was tested using MATLAB software and tools. Two testing ratios were used for testing both networks. The first uses a total of 40 images; 5 for each inset class. The second uses 112 images; 14 for each insect type. Both networks were tested using these two testing ratios to investigate the best learning scheme and the network that records highest accuracy. The results of both testing and training phases of both networks are included in the following Table 5.4.

Table 5.4: Total classification rate for BPNN1 and BPN2

Insects images	Image sets	Number of images	Recognition rate of BPNN1	Recognition rate of BPNN2
	Testing set (80:40)	40	95%	93%
Total recognition set	Testing set (80:112)	112	87%	89%

Table 5.4 shows the classification rate achieved upon training and testing both BPNN1 and BPNN2 with the two learning schemes: 80:40 and 80:112. It also represents the number of images used in each set for both databases, as well as the overall identification rate obtained which are 95% and 87% for BPNN1 and 87% and 89% for BPNN2 tested using two testing ratio respectively.

CHAPTER SIX
RESULTS DISCUSSION, COMPARISON AND CONCLUSION

6.1 Results Discussion

In this thesis, an intelligent plant pest identification system was developed. The system is based on both image processing and neural network classification. A large number of images of 8 different types of insects were collected from different plant pest databases. These images were used for training and later for testing the network. The system comprised of image processing phase where the images are processed and then some features are extracted using pattern averaging technique. Two networks and two databases were used in this work to investigate the effectiveness of difference in input size and learning schemes of the classification rate a neural network. For BPNN 1the input images were first rescaled to 64*64 and it was trained and tested using two learning schemes. For the BPNN2, the images were rescaled using pattern averaging to 32*32 and also it was trained and tested using the two learning schemes.

The networks were finally converged each with different recognition rate and were able to classify the 8 types of insects regardless of a small error using the backpropagation learning algorithm.

As previously discussed, two backpropagation neural networks were experimentally used for performing this classification task. The purpose of using two networks is to experience the effects of the huge number of neurons in the input layer on the effectiveness and time processing of the neural network. In addition, two learning schemes were used for training and testing the networks. The learning scheme 1 uses 80 images for training and 40 images for testing. However, the same number of images (80) was used per learning scheme 2 for training but the testing number of images was different; 112 images of 8 different insects. Therefore, each network was trained using 80 images but tested with different testing ratios. The BPNN1 has 4096 neurons in the input layer, while the BPNN2 has 1024 neurons.

The testing simulations in Figures 5.13 and 5.14 show that the training classification rate is better for BPNN1 where input images are of size 64*64 then that for BPNN2 where the images used are of less dimensions (32*32). The learning with database 1 was exponentially decreasing with the increasing of the number of iterations in Figure 5.13 while it was not in Figure 5.14. Moreover, the network 1 has achieved a very small minimum square error of 0.0008. However, BPNN2 couldn't achieve such minimum square error where it reached only 0.022.

Note that both networks were trained using same training ratio which uses 80 images of 8 different insects. The difference in accuracy is the averaging of the input image which allows the loss of some important features in BPNN1 since the images were rescaled to 32*32. Thus, it can be concluded that the network with highest input neurons performed better during training phase. Similarly, the networks were both tested using testing ratios 1 & 2 respectively. However, during the testing ratio 1 (80:40) the network BPNN1 which uses higher number of input neurons performed better when tested with 80:40 ration. This is due to the used learning scheme (80:40) which uses more training images than testing images which improves the generalization capability of network. In contrast, the network 2 BPNN2 which uses less input neurons performed also better with testing ratio 1, but it couldn't outperform BPNN1 which it clearly because of its number of neurons in the input layer.

Using testing ratio 2 (80:112), both networks (BPNN1 and BPN2) classification rate is less that that when testing ratio 1 was used. Obviously, this is due to the less number of images used in testing than training which reduces that capability of networks to generalize.

In addition, the training processing time of the BPNN1 is 57 seconds which is lower than comparing to the BPNN2 which takes 88 seconds to simulate the results with best recognition rate.

All of the above discussed results show that the use of 64*64 pixels for the input images helps in improving the effectiveness and efficiency of the network than using lower sizes 32*32 since some features may be cleared. However, what improves the accuracy more is the use of a learning schemes that involves more training images than those of testing which helps in obtaining higher classification rate and lower processing time than one in database 2 (80:80).

6.2 Results Comparison

This table below shows a comparison between different related works to our proposed system for the classification of insects using backpropagation neural network.

Many researches have been conducted in order to develop effective systems for the insects classification based on image processing and intelligent classifier such as neural networks. For each research, the authors created image processing algorithms for the insects' shapes and edges detection and then they started the classification process that is based on intelligent systems like neural networks or other classifiers.

In some papers, the authors used the whole processed insects' images to be fed into the neural network, while others extracted the important features that may indicate the geometric shape of insects and they used as input vector for a neural network.

In this thesis, the images are rescaled to 64*64 and 32*32 using pattern averaging to preserve the needed features and then they were fed into 2 neural networks to be classified into 8 classes.

Table 6.1: Results Comparison

Paper Title	**Authors**	**Methods used**	**Recognition Rate**
The Application of AdaBoost-Neural Network in Stored product Insect Classification	Hongmei Zhang et al.	AdaBoost-Neural Network	89 %
Image Classification of Paddy Field Insect Pests Using Gradient-Based Features-KNN and SVM	Kanesh Venugoban and Amirthalingam Ramanan	Gradient-Based Features-KNN and SVM	90% for SVM and 89% for KNN
Proposed system	Salah Zarrouk and Kamil Dimililer	BPNN	95%

6.3 Conclusion

In this thesis, we have illustrated insects' pest classification system that was developed with a backpropagation neural network. The developed system aims to classify the 8 different types of insects that can be found on plants in the agricultural field. The methodology of the developed framework requires the extraction of the features that represent the shape of the insect so that it can be classified using intelligent classifier such as neural network.

Therefore, pattern averaging technique was used in order to extract the geometric and shape features from the input images. The images were processed first using different image processing techniques such as median filtering, image adjusting, and then they were segmented using canny edge detection in order to be pertained to the next techniques which is the feature extraction using pattern averaging. After extracting features, the images were fed into the neural network to be classified into 8 types of insects.

During the training phase, two backpropagation neural networks were used to simulate the training results. For the first one (BPNN1), the images were averaged to 64*64 using a window of 4*4; while for the second one (BPNN2), the images were rescaled to 32*32 using a window of 8*8.

The simulation of both networks was achieved using one database which uses two learning schmes for testing. The training simulation showed that the BPNN1 which uses higher input neurons than BPNN2 showed better performance during training. However, the testing simulation with both ratios showed that the highest number of images used for training than testing lead to a better network performance and generalization in the testing phase. This learning schemes which included higher number of images during testing and greater image size allows the network to be more effective during training since it results in higher classification and very low minimum square error in addition to less processing time 57 minutes.

REFERENCES

Bashish, D., Braik, M., & Bani-Ahmad, S. (2011). Detection and classification of leaf diseases using K-means-based segmentation and. *Information Technology Journal, 10*(2), 267-275.

Baxt, W. G. (1995). Application of artificial neural networks to clinical medicine. *The lancet, 346*(8983), 1135-1138.

Beham, M. P., & Gurulakshmi, A. B. (2012). Morphological image processing approach on the detection of tumor and cancer cells. *In Proceeding of Devices, Circuits and Systems International Conference on* (pp. 350-354). Italy: University of Bologna.

Carino, F. O., Kenmore, P. E., & Dyck, V. A. (1979). The FARMCOP suction sampler for hoppers and predators in flooded rice fields. *International Rice Research Newsletter, 4*(5), 21-22.

Church, J. C., Chen, Y., & Rice, S. V. (2008, April). A spatial median filter for noise removal in digital images. *In Proceeding of IEEE SoutheastCon 2008* (pp. 618-623). Malaysia: University Malaysia Sabah.

Dimililer, K. (2012). Neural network implementation for image compression of X-rays. *Electronics World, 118*(1911), 26-29.

Dimililer, K. (2012). Neural network implementation for image compression of X-rays. *Electronics World*, *118*(1911), 26-29.

Dimililer, K. (2013). Backpropagation neural network implementation for medical image compression. *Journal of Applied Mathematics*, *29*(1), 10-18.

Dimililer, K., & İlhan, A. (2016). Effect of Image Enhancement on MRI Brain Images with Neural Networks. *Procedia Computer Science*, *102*, 39-44.

Dimililer, K., Ever, Y. K., & Ugur, B. (2016, September). ILTDS: Intelligent Lung Tumor Detection System on CT Images. *In Proceeding of the International Symposium on*

Intelligent Systems Technologies and Applications (pp. 225-235). Springer International Publishing.

Gasson, J., Gandy, S. J., Hutton, C. W., Jacoby, R. K., Summers, I. R., & Vennart, W. (2000). Magnetic resonance imaging of rheumatoid arthritis in metacarpophalangeal joints. *Skeletal radiology*, *29*(6), 324-334.

Gonzalez R.C., Woods R.E. (2001) Digital Image Processing. New York: CRC Press.

Haralick, R. M., & Shapiro, L. G. (1985). Image segmentation techniques. *Computer vision, graphics, and image processing*, *29*(1), 100-132.

Haykin, S. S. (2001). Neural networks: a comprehensive foundation. Tsinghua University Press.

Kim, Y. T. (1997). Contrast enhancement using brightness preserving bi-histogram equalization. *IEEE transactions on Consumer Electronics, 43*(1), 1-8.

Larios, N., Deng, H., Zhang, W., Sarpola, M., Yuen, J., Paasch, R, & Shapiro, L. G. (2008). Automated insect identification through concatenated histograms of local appearance features: feature vector generation and region detection for deformable objects. *Machine Vision and Applications*, *19*(2), 105-123.

Martin, V., & Moisan, S. (2004). Early Pest Detection in Greenhouses. *INRIA Sophia Antipolis Mediterrannee*, *2*(6), 27-35.

Mittal, A., & Dubey, S. K. (2012). Analysis of Rheumatoid Arthritis through Image Processing. *International Journal of Computer Science, 9*(6).

Mundada, R. G., & Gohokar, V. V. (2013). Detection and classification of pests in greenhouse using image processing. *Journal of Electronics and Communication Engineering, 5*(6), 57-63.

Radha, R., & Lakshman, B. (2013). Retinal image analysis using morphological process and clustering technique. *Signal & Image Processing*, *4*(6), 55-100.

Russell, D., Reed., R., Marks, J. (2011). Supervised Learning in Feed forward Artificial Neural Networks. New York, NY: university of York.

Seçil, M., Obuz, F., Altay, C., Gencel, Ö., Igci, E., Sagol, Ö., & Dicle, O. (2008). The role of dynamic subtraction MRI in detection of hepatocellular carcinoma. *Diagnostic and Interventional Radiology, 14*(4), 2004.

Souza, T. L., Mapa, E. S., dos Santos, K., & Menotti, D. (2011, September). Application of complex networks for automatic classification of damaging agents in soybean leaflets. *In Proceeding of the 18th IEEE International Conference on Image Processing* (pp. 1065-1068). India: University of India.

Villegas, V., Osiris, O., González, B. (2009). License Plate Recognition Using a Novel Fuzzy Multilayer Neural Network. *International Journal of Computer, 3*(1), 4-9.

Wang, Y., Chen, Q., & Zhang, B. (1999). Image enhancement based on equal area dualistic sub-image histogram equalization method. *IEEE Transactions on Consumer Electronics, 45*(1), 68-75.

APPENDIX

- **Source Code**

```
%ants

for k=1:35

pic = strcat('IMAGE\Ants\an', num2str(k), '.jpg');

f = imread(pic);

f=rgb2gray(f);

f=imresize(f, [256 256]);

m=medfilt2(f);

g2=imadjust(m,[0.5 0.75], [0 1]);   %% image adjustement, gamma =1, by default..convert the
%intensities btw 0.5 and 0.75 to values btw 0 and 1.

%% treshold the image

level = graythresh(g2);

bw = im2bw(g2,level);

bw = bwareaopen(bw, 50);

% Labeled the component(s), and plot the centroid on the original image

% Canny

S5 = edge(g2, 'canny', 0.25);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

m3=S5;

T = blkproc(m3, [4 4], @mean2);

T=reshape(T,[ ],1);

if k==1

e1=[ ];
```

```
h=[e1 T];% concatenate reshaped images horizontally

else

h=[e1 T];% concatenate reshaped images horizontally

end

e1=h;% safe images for reuse

s1=e1;% safe e1 into s1,incase

end

t1=ones(1,35);

t2=zeros(7,35);

b_target1=[t1;t2];

l1=h;

%ants=[e1;A_target];

ant_train_input=l1(:, 1:25);

ant_train_target=b_target1(:, 1:25);

ant_test_input= l1(:, 26:end);

ant_test_target=b_target1(:, 26:end);

%%%Aphids

for k=1:36

pic = strcat('IMAGE\Aphids\ap', num2str(k), '.jpg');

f=imread(pic);

f=imresize(f, [256 256]);

f=rgb2gray(f);

m=medfilt2(f);

g2=imadjust(m,[0.5 0.75], [0 1]);   %% image adjustement, gamma =1, by default..convert the
intensities btw 0.5 and 0.75 to values btw 0 and 1.
```

```
%% treshold the image

level = graythresh(g2);

bw = im2bw(g2,level);

bw = bwareaopen(bw, 50);

% Labeled the component(s), and plot the centroid on the original image

% Canny

S5 = edge(g2, 'canny', 0.25);

m3=S5;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

T = blkproc(m3, [4 4], @mean2);

T=reshape(T,[ ],1);

if k==1

e1=[ ];

h=[e1 T];% concatenate reshaped images horizontally

else

h=[e1 T];% concatenate reshaped images horizontally

end

e1=h;% safe images for reuse

s1=e1;% safe e1 into s1,incase

end

t1=zeros(1,36);

t2=ones(1,36);

t3=zeros(6,36);

b_target2=[t1;t2;t3];
```

```
l2=h;

%aphids=[e1;b_target];

aphids_train_input=l2(:, 1:25);

aphids_train_target=b_target2(:, 1:25);

aphids_test_input= l2(:, 26:end);

aphids_test_target=b_target2(:, 26:end);

%%%caterpillars

for k=1:45

pic = strcat('IMAGE\Caterpillars\ca', num2str(k), '.jpg');

f=imread(pic);

f=rgb2gray(f);

f=imresize(f, [256 256]);

m=medfilt2(f);

g2=imadjust(m,[0.5 0.75], [0 1]);   %% image adjustement, gamma =1, by default..convert the
intensities btw 0.5 and 0.75 to values btw 0 and 1.

%% treshold the image

level = graythresh(g2);

bw = im2bw(g2,level);

bw = bwareaopen(bw, 50);

% Labeled the component(s), and plot the centroid on the original image

% Canny

S5 = edge(g2, 'canny', 0.25);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

m3=S5;

T = blkproc(m3, [4 4], @mean2);
```

```
T=reshape(T,[ ],1);
if k==1
e1=[ ];
h=[e1 T];% concatenate reshaped images horizontally
else
h=[e1 T];% concatenate reshaped images horizontally
end
e1=h;% safe images for reuse
s1=e1;% safe e1 into s1,incase
end
t1=zeros(2,45);
t2=ones(1,45);
t3=zeros(5,45);
b_target3=[t1;t2;t3];
l3=h;
%caterpillars=[e1;b_target];
caterpillars_train_input=l3(:, 1:32);
caterpillars_train_target=b_target3(:, 1:32);
caterpillars_test_input= l3(:, 33:end);
caterpillars_test_target=b_target3(:, 33:end);
%%%earwigs
for k=1:46
pic = strcat('IMAGE\Earwigs\ea', num2str(k), '.jpg');
f=imread(pic);
f=rgb2gray(f);
```

```
f=imresize(f, [256 256]);

m=medfilt2(f);

g2=imadjust(m,[0.5 0.75], [0 1]);   %% image adjustement, gamma =1, by default..convert the
intensities btw 0.5 and 0.75 to values btw 0 and 1.

%% treshold the image

level = graythresh(g2);

bw = im2bw(g2,level);

bw = bwareaopen(bw, 50);

% Labeled the component(s), and plot the centroid on the original image

% Canny

S5 = edge(g2, 'canny', 0.25);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

m3=S5;

T = blkproc(m3, [4 4], @mean2);

T=reshape(T,[ ],1);

if k==1

e1=[ ];

h=[e1 T];% concatenate reshaped images horizontally

else

h=[e1 T];% concatenate reshaped images horizontally

end

e1=h;% safe images for reuse

s1=e1;% safe e1 into s1,incase

end

t1=zeros(3,46);
```

```
t2=ones(1,46);

t3=zeros(4,46);

b_target4=[t1;t2;t3];

l4=h;

%earwigs=[e1;b_target];

earwigs_train_input=l4(:, 1:32);

earwigs_train_target=b_target4(:, 1:32);

earwigs_test_input= l4(:, 33:end);

earwigs_test_target=b_target4(:, 33:end);

%%%Grasshoppers

for k=1:50

pic = strcat('IMAGE\Grasshoppers\g', num2str(k), '.jpg');

f=imread(pic);

f=rgb2gray(f);

f=imresize(f, [256 256]);

m=medfilt2(f);

g2=imadjust(m,[0.5 0.75], [0 1]);   %% image adjustement, gamma =1, by default..convert the
intensities btw 0.5 and 0.75 to values btw 0 and 1.

%% treshold the image

level = graythresh(g2);

bw = im2bw(g2,level);

bw = bwareaopen(bw, 50);

% Labeled the component(s), and plot the centroid on the original image

% Canny

S5 = edge(g2, 'canny', 0.25)
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

m3=S5;

T = blkproc(m3, [4 4], @mean2);

T=reshape(T,[ ],1);

if k==1

e1=[ ];

h=[e1 T];% concatenate reshaped images horizontally

else

h=[e1 T];% concatenate reshaped images horizontally

end

e1=h;% safe images for reuse

s1=e1;% safe e1 into s1,incase

end

t1=zeros(4,50);

t2=ones(1,50);

t3=zeros(3,50);

b_target5=[t1;t2;t3];

l5=h;

%grasshoppers=[e1;b_target];

Grasshoppers_train_input=l5(:, 1:35);

Grasshoppers_train_target=b_target5(:, 1:35);

Grasshoppers_test_input= l5(:, 36:end);

Grasshoppers_test_target=b_target5(:, 36:end)
```

```
%%%Mealybugs

for k=1:24

pic = strcat('IMAGE\Mealybugs\m', num2str(k), '.jpg');

f=imread(pic);

f=rgb2gray(f);

f=imresize(f, [256 256]);

m=medfilt2(f);

g2=imadjust(m,[0.5 0.75], [0 1]);   %% image adjustement, gamma =1, by default..convert the
intensities btw 0.5 and 0.75 to values btw 0 and 1.

%% treshold the image

level = graythresh(g2);

bw = im2bw(g2,level);

bw = bwareaopen(bw, 50);

% Labeled the component(s), and plot the centroid on the original image

% Canny

S5 = edge(g2, 'canny', 0.25);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

m3=S5;

T = blkproc(m3, [4 4], @mean2);

T=reshape(T,[ ],1);

if k==1

e1=[ ];

h=[e1 T];% concatenate reshaped images horizontally

else

h=[e1 T];% concatenate reshaped images horizontally
```

```
end
e1=h;% safe images for reuse
s1=e1;% safe e1 into s1,incase
end
t1=zeros(5,24);
t2=ones(1,24);
t3=zeros(2,24);
b_target6=[t1;t2;t3];
l6=h;
%Mealybugs=[e1;b_target];
Mealybugs_train_input=l6(:, 1:17);
Mealybugs_train_target=b_target6(:, 1:17);
Mealybugs_test_input= l6(:, 18:end);
Mealybugs_test_target=b_target6(:, 18:end);
%%%Spiders.mite
for k=1:24
pic = strcat('IMAGE\Spiders\sp', num2str(k), '.jpg');
f=imread(pic);
f=rgb2gray(f);
f=imresize(f, [256 256]);
m=medfilt2(f);
g2=imadjust(m,[0.5 0.75], [0 1]);   %% image adjustement, gamma =1, by default..convert the
intensities btw 0.5 and 0.75 to values btw 0 and 1.
%% treshold the image
level = graythresh(g2);
```

```
bw = im2bw(g2,level);

bw = bwareaopen(bw, 50);

% Labeled the component(s), and plot the centroid on the original image

% Canny

S5 = edge(g2, 'canny', 0.25);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

m3=S5;

T = blkproc(m3, [4 4], @mean2);

T=reshape(T,[ ],1);

if k==1

e1=[ ];

h=[e1 T];% concatenate reshaped images horizontally

else

h=[e1 T];% concatenate reshaped images horizontally

end

e1=h;% safe images for reuse

s1=e1;% safe e1 into s1,incase

end

t1=zeros(6,24);

t2=ones(1,24);

t3=zeros(1,24);

b_target7=[t1;t2;t3];

l7=h;

%spiders=[e1;b_target];
```

```
Spiders_mite_train_input=l7(:, 1:17);

Spiders_mite_train_target=b_target7(:, 1:17);

Spiders_mite_test_input= l7(:, 18:end);

Spiders_mite_test_target=b_target7(:, 18:end);

%%%Whileflies

for k=1:27

pic = strcat('IMAGE\Whiteflies\w ', num2str(k), '.jpg');

f=imread(pic);

f=rgb2gray(f);

f=imresize(f, [256 256]);

m=medfilt2(f);

g2=imadjust(m,[0.5 0.75], [0 1]);   %% image adjustement, gamma =1, by default..convert the
intensities btw 0.5 and 0.75 to values btw 0 and 1.

%% treshold the image

level = graythresh(g2);

bw = im2bw(g2,level);

bw = bwareaopen(bw, 50);

% Labeled the component(s), and plot the centroid on the original image

% Canny

S5 = edge(g2, 'canny', 0.25);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

m3=S5;

T = blkproc(m3, [4 4], @mean2);

T=reshape(T,[ ],1);

if k==1
```

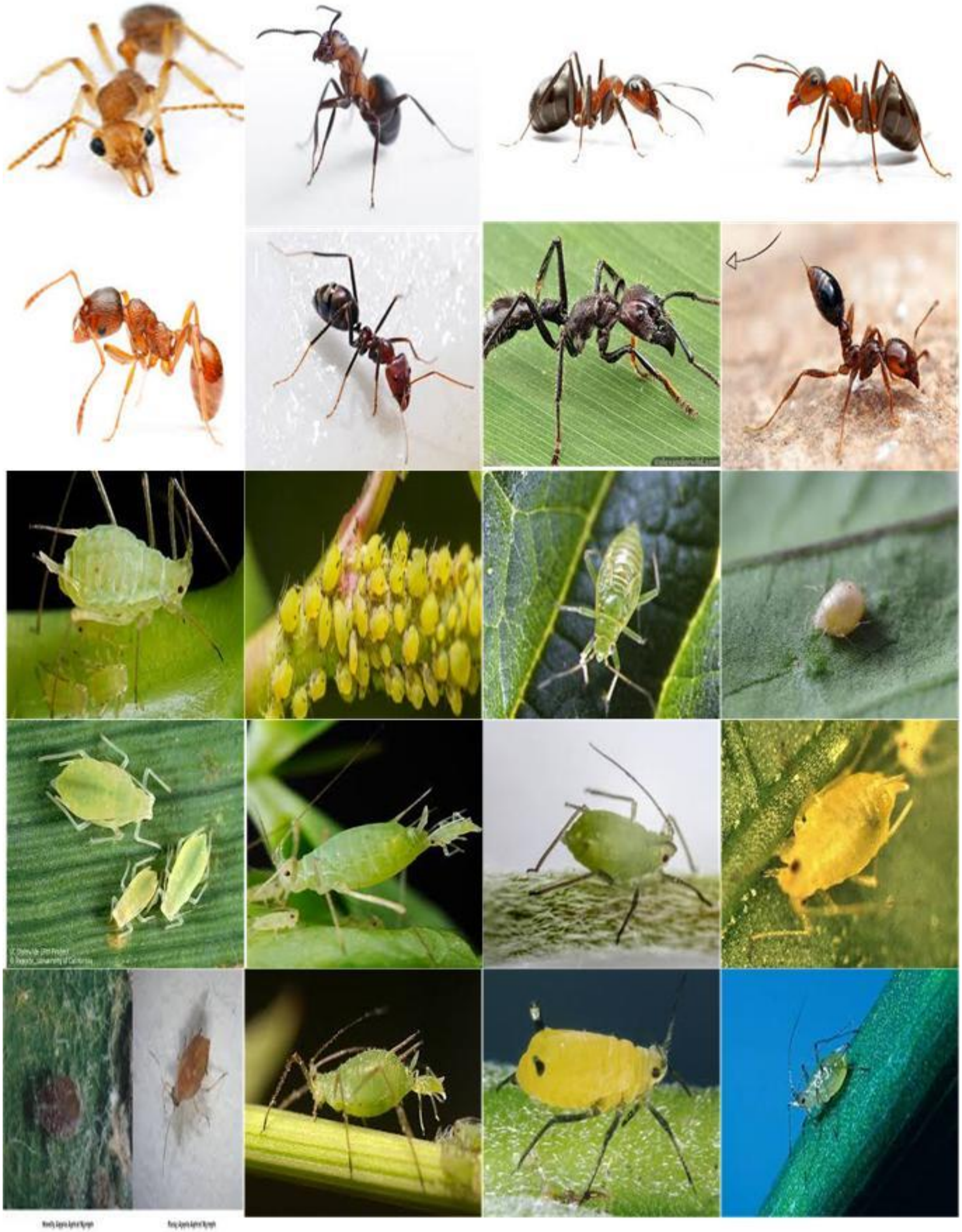
```
e1=[ ];

h=[e1 T];% concatenate reshaped images horizontally

else

h=[e1 T];% concatenate reshaped images horizontally

end

e1=h;% safe images for reuse

s1=e1;% safe e1 into s1,incase

end

t1=zeros(7,27);

t2=ones(1,27);

b_target8=[t1;t2];

l8=h;

% whiteflies=[e1;b_target];

Whileflies_train_input=l8(:, 1:19);

Whileflies_train_target=b_target8(:, 1:19);

Whileflies_test_input= l8(:, 20:end);

Whileflies_test_target=b_target8(:, 20:end);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%5%%%%%%%%%%
%%%%%%%%%%%

%%Concatenate all flys for training data

train_input=[ant_train_input, aphids_train_input, caterpillars_train_input, earwigs_train_input,
Grasshoppers_train_input, Mealybugs_train_input, Spiders_mite_train_input,
Whileflies_train_input];

train_target=[ant_train_target, aphids_train_target, caterpillars_train_target,
earwigs_train_target, Grasshoppers_train_target, Mealybugs_train_target,
Spiders_mite_train_target, Whileflies_train_target];

%%Concatenate all flys for testing data
```

```
test_input=[ant_test_input, aphids_test_input, caterpillars_test_input, earwigs_test_input, Grasshoppers_test_input, Mealybugs_test_input, Spiders_mite_test_input, Whileflies_test_input];

test_target=[ant_test_target, aphids_test_target, caterpillars_test_target, earwigs_test_target, Grasshoppers_test_target, Mealybugs_test_target, Spiders_mite_test_target, Whileflies_test_target];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Solve a Pattern Recognition Problem with a Neural Network

% Script generated by NPRTOOL

%

% This script assumes these variables are defined:

%

% CREATING AND INITIATING THE NETWORK

net = newff(minmax(train_input),[35 8],{'logsig','logsig'},'traingdm');

% TRAINING THE NETWORK

net.trainParam.lr = 0.47;  % Learning Rate.

net.trainParam.show = 300; % Frequency of progress displays (in epochs).

net.trainParam.epochs =5000;% Maximum number of epochs to train.

net.trainParam.mc = 0.85 % Momentum Factor.

[net,tr] = train(net,train_input,train_target);
```

- **Dataset**

1. Training Data


University of Florida




A
Photo Credit: Ben Gray
Photo 1. Adult soybean aphid (wingless)

Photo: Male earwig (by G. Bradley)
www.uksafari.com


5402874
Photo: Meadow grasshopper (by G. Bradley)
www.uksafari.com


UF


Spider mites on
a lemon plant

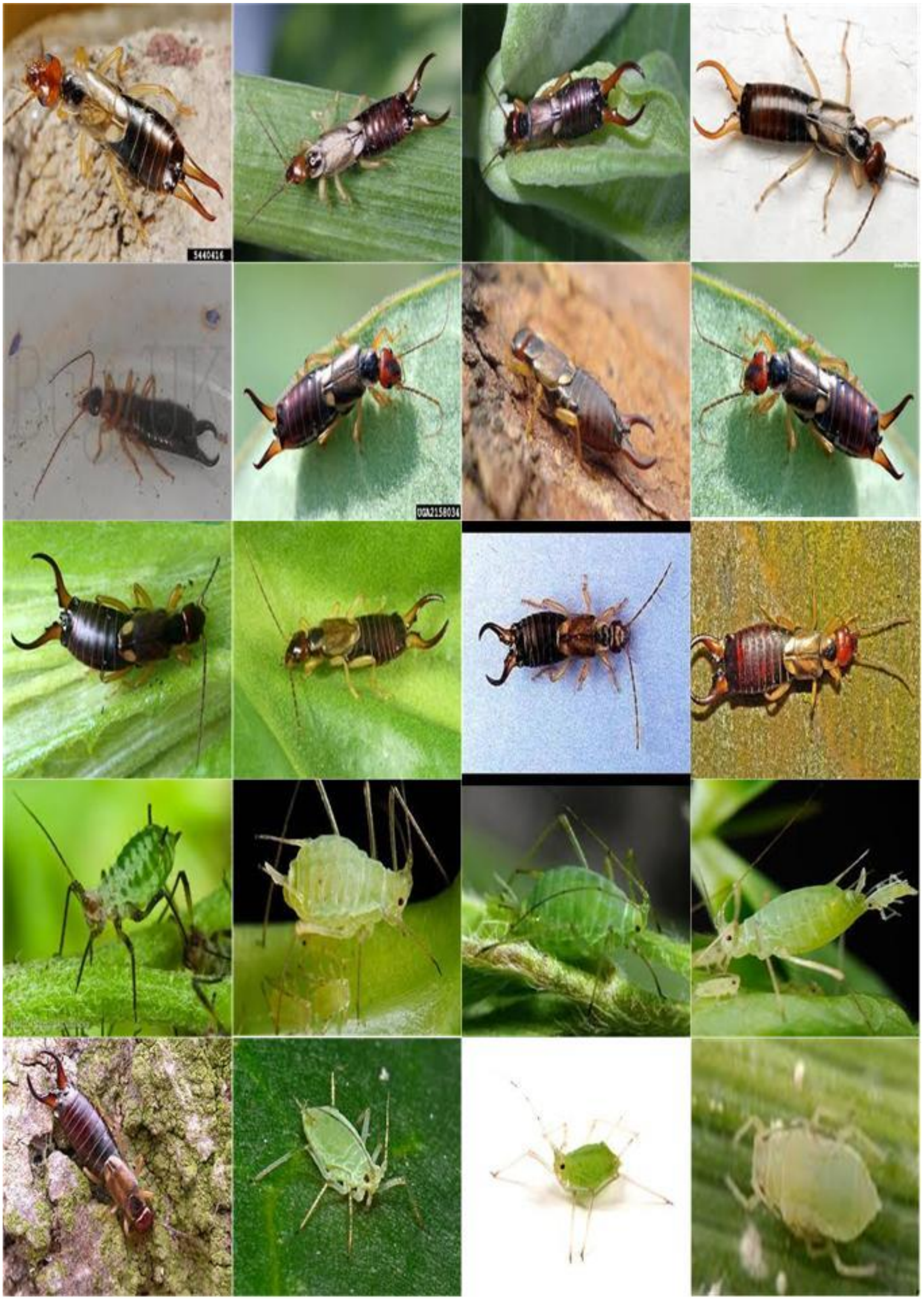

UGA1148020


University of Nebraska
Department of Entomology

2. Testing Data


Whitefly


UF
Baldo Villegas
Forficula auricularia male, cleaning antenna
www.bumblebee.org


5440416
UGA2158034