Improved Feed Utilisation in Cage Aquaculture by Use of Machine Vision

by

Zelda Dunn

Thesis submitted in partial fulfilment
of the requirements for the Degree
of
MASTER OF SCIENCE IN ENGINEERING
(CHEMICAL ENGINEERING)

in the Department of Process Engineering
at Stellenbosch University

Supervised by
Prof. C. Aldrich, Department of Process Engineering
Mr. L.F. de Wet, Feed Technology Group, Division Aquaculture

December 2008

Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the owner of the copyright thereof (unless to the extent explicitly otherwise stated) and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: 5 December 2008

Synopsis

With the harvesting of fish and other aquatic organisms from natural waters having reached its upper limit, aquaculture is vital in providing for the ever increasing demand for fishery products (Boyd, 1999). Not surprisingly, aquaculture has seen considerable growth over the last decade or more.

With the rising importance of aquaculture, there is an increased emphasis on cost and reducing of waste for environmental reasons. Therefore, attempts to automate or increase efficiency of feeding are constantly being explored.

On an aquaculture unit approximately 60% of all costs are for feed; therefore high quality feeding management is essential for all fish farmers.

The rainbow trout farm at Jonkershoek Aquaculture Research farm near Stellenbosch currently have a feeding management system which makes use of traditional hand feeding. Handfeeding is not considered optimal, as the feed intake or pellet loss is not closely monitored resulting in higher operating costs.

Automation of aquaculture systems will allow the industry to produce closer to markets, improve environmental control, reduce catastrophic losses, minimize environmental regulation by reducing effluents, reduce production costs and improve product quality. The history of automated control in aquaculture has been brief; most of the systems have been custom-designed, personal computer systems.

A very popular approach for an automated feeding system is to monitor waste pellets beneath the feeding zone of the fish, with a feedback loop that can switch off the feeder if this waste exceeds a predetermined threshold. Other approaches use hydroacoustics to monitor waste pellets or demand feeders have also been implemented. These approaches however are not considered optimal as automatic feeders do not necessarily ensure optimal feed intake. Social dominance using demand feeders does not allow even feeding distribution among all sizes of fish.

In this project it was investigated whether an automated feeding system can be developed based on fish feeding behaviour. After facing problems with poor visibility at the Jonkershoek Aquaculture farm near Stellenbosch, video data were acquired from the Two Oceans Aquarium in Cape Town. Since it was a feasibility study, the focus was rather to investigate whether a predictive model could be generated for fish feeding behaviour in a more ideal environment which can form a foundation for further research.

The well-established multivariate methods of principal components analysis (PCA) and linear discriminant analysis (LDA) were used to extract informative features from the image data. These features were labelled with the corresponding behaviour they captured, namely the prefeeding, feeding and postfeeding behaviour of fish. By use of LDA, the three classes of behaviour could be identified with an accuracy of approximately 96%. Marginally better results were obtained with nonlinear models, such as classification trees and a nearest neighbour approach, using the LDA and PCA scores as inputs.

It was found during this study, that at the Jonkershoek aquaculture farm, external environmental factors would play a significant role in acquiring quality image data. These factors included turbidity induced by rain and considerable changes in lighting conditions.

The challenge of acquiring quality image data under these typically changing environmental conditions would have to be surmounted for the successful implementation of the proposed method.

Opsomming

Dit kan aanvaar word dat daar ‘n groeiende tekort is aan visbronne regoor die wêreld. Die rede daarvoor is dat daar nie meer enige grense aan moderne visvangtegnieke nie. Dus moet ons die behoefte vir visboerdery erken.

As gevolg van die groeiende visboerdery bedryf word daar al hoe meer klem gesit op koste besparing asook omgewingsbelange.

Visvoedsel bedra ongeveer 60% van die totale bedryfskoste, dus moet voeding baie effektief bestuur word.

Die reënboogforelplaas by Jonkershoek Akwakultuur Navorsingseenheid naby Stellenbosch gebruik huidig tradisionele handvoerders. Hierdie metode word nie as optimaal aanvaar nie, omdat viskosinname nie effektief gemoniteer word nie, wat dus lei tot hoër bedryfskostes.

Outomatisering van akwakultuurvoedingsisteme kan die industrie in staat stel om meer doeltreffend te produseer, omgewingskontrole te verbeter, katastofiese verliese te verminder, omgewingsbesoedeling te verminder deur minder afvalprodukte vry te stel en ook produkkwaliteit verbeter. Die outomatisasie van voedingsisteme is egter nog min en glad nie goed ontwikkel nie en meeste sisteme is doelgemaakte programme.

‘n Bewese benadering is om gebruik te maak van sensors wat die korrels wat nie geëet word nie onder die voedingsone op te tel en dan met terugvoerbeheer die outomatiese voerder af te skakel as dit ‘n sekere punt bereik. Ander benaderings is om gebruik te maak van akoestiese seine en selfvoerders, waar die visse self kos op aanvraag kan inneem. Hierdie metodes word nie gesien as optimaal nie, omdat dit nie noodwendig die optimale inname van viskos verseker nie, en met selfvoerders is daar altyd die probleem van sosiale dominansie, waar sekere visse gevoer word ten koste van die ander.

Met hierdie projek is ‵n alternatiewe benadering gevolg deur die haalbaarheid te ondersoek van ‵n outomatiese voedingsisteem gebaseer op waarneembare visvoedingsgedrag. As gevolg van troebel water by die Jonkershoek dam by Stellenbosch kon video-opnames van visgedrag by die plaas ongelukkig nie gebruik word vir ontleding nie. In die plek daarvan is video-opnames van visgedrag in die Twee Oseane Akwarium in Kaapstad ontleed.

Met behulp van hoofkomponentontleding en lineêre diskriminantanalise kon daar onderskeid getref word tussen die gedrag van die visse voor, gedurende en na voeding. Hierdie drie klasse van gedrag kon ongeveer 96% akkuraat onderskei word met behulp van lineêre diskriminantanalise. Marginaal beter resultate kon behaal word deur nie-lineêre modelle te gebruik, soos klassifikasiebome en naaste bure modelle, met die hoofkomponent and lineêre diskriminanttellings as insette,

Die data suggereer dus dat die benadering waar voeding gebaseer sou wees op die waarneembare gedrag van die visse haalbaar sou kon wees, mits goeie beeldmateriaal van die visse bekom sou kon word. Eksterne omgewingsfaktore sal waarskynlik ‵n noemenswaardige rol speel in die uitvoerbaarheid van die voorgestelde metode by die Jonkershoek Akwakultuur plaas. Eksterne faktore sluit in troebelheid van water veroorsaak deur reën, of roofdiere of enige ander stimulante wat bydra tot veranderings in gedragspatrone.

Dit word voorgestel dat verdere navorsing gedoen moet word om patrone te vind in visvoedingsgedrag deur gebruik te maak van beeldmateriaal verkry deur byvoorbeeld sonar, om sodoende die invloed van troebelheid op die kwaliteit van die beeldmateriaal te beperk.

Acknowledgements

I would like to thank the Department of Aquaculture for providing the financial support for this project. I would also like to thank Prof. C. Aldrich and Mr. L.F. de Wet for their support and guidance over the last two years.

I would specially like to thank Gorden Jemwa and John Burchell for their help and input into my project. I would also like to thank my family and friends for their support and our Creator.

Table of Contents

Chapter 1

Introduction

1.1 Background on Aquaculture

Aquaculture is the aquatic counterpart of agriculture and its origins extend back some 4000 years (Beveridge, 2004). The earliest cages were first used by fisherman as a convenient holding facility for fish until ready for sale, and were essentially modified fish traps or baskets. These traditional types of holding facilities were used in many parts of the world for generations. True cage culture, in which fish or other organisms were held for long periods of time until they increased in weight, was until recently thought to be a comparatively modern development. According to Li (1994), however, cage culture was established in China during Han Dynasty 2200 – 2100 years ago. Modern cage culture began in the 1950s with the advent of synthetic materials for cage construction. In the United States, universities did not begin conducting research on cage rearing of fish until the 1960’s (Beveridge, 2004).

Until recently agriculture and aquaculture has not developed since food in lakes and seas had always been abundant. World demand for fish, both as a source of food for human consumption and for reduction to fishmeal, has grown at a steady pace since the end of World War II. Previously demands were met by the expansion of capture fisheries, and indications are that capture fisheries will be the most important means of providing fish as a food source over the next 25 years (Beveridge, 2004). Fisheries and aquaculture aim to maximise the yield of useful organisms from the aquatic environment.

Aquaculture or fish farming is achieved through manipulation of a fish's life cycle and control of the environmental variables that influence it. In South Africa aquaculture production has increased with more that 30% per year from 3000 ton (value R51 million) in 1997 to 4030 ton (R146 million) in 2000 (van Zyl, 2006). Aquaculture has been the forerunner in agriculture production for the past three decades, with an average growth of 8.6% per year (van Zyl, 2006).

It is known that approximately 60% of all cost in fish farming is for fish feed (Foster et al*.*, 1995). Rainbow trout is a carnivorous species that requires high protein feeding and well oxygenated water. Most fish feed is manufactured from fish meal which is a commercial product made from both whole fish and the bones and offal from processed fish.

Depending on the quality of the diet and temperature it would take approximately 12 – 14 months for the fish to reach a marketable size. According to Tacon (1999), the higher the quality of fish meal, the more expensive the feeding pellets are; however it is more economical for the fish to reach a marketable size as soon as possible (Huet, 1975, Shakya, 2007).

The optimal conversion in fish farming or cage aquaculture can be achieved by applying high-quality feeding management. Feed management is having control of how much feed is to be given in order to achieve the optimal growth rate without overfeeding. Excess feeding leads to waste feed, which results in excess costs and poor water quality, which in turn could lead to stress of the fish. Many factors influence the appetite of fish, such as environmental conditions (including water temperature, oxygen concentration and water quality) and physiological factors (such as age/size, stress level and endogenous rhythms); therefore it is difficult to determine the exact quantity of feed that must be fed (Beveridge, 2004). Automated feeding systems are therefore designed with the aim to ensure an optimal conversion is achieved, which will be more economical for fish farming.

Several inventions and improvements have recently been developed in creating automated feeding systems. These emerging systems are known as "adaptive", "smart" or "intelligent" and develop an increased understanding of fish physiology, nutrition and behaviour. These systems are governed by process control, which is essentially using sensors and computer technology, which is better known as process control. A popular approach is to monitor waste pellets beneath the feeding zone with a feedback loop that can switch off the feeder if this waste exceeds a predetermined threshold (Myrseth, 2000). It is known that fish display behavioural characteristics that underlie an optimal rate of feed intake throughout their life cycle (Blyth et al*.*, 1999). However, it is not certain whether one can base a feeding system on these characteristics.

The objective of this project is to determine if an automated feeding system can be developed through image analysis of fish feeding behaviour using a submerged surveillance camera. The proposed method is to obtain video images of fish feeding behaviour and subsequently applying data analysis. The data analysis will entail image processing, followed by pattern recognition and machine learning methods. Currently there is no published literature using this approach.

Figure 1-1: Cage unit of the Jonkershoek Aquaculture Research Farm

The literature reviewed in this paper is needed to obtain a comprehensive background and understanding of automated feeding systems in aquaculture using methods as explained above. This project was to be conducted at Jonkershoek Aquacultural Research Farm (near Stellenbosch) however due to various difficulties (explained in chapter 3) the experimental part of this project was conducted at the Two Oceans Aquarium in Cape Town, South Africa.

Figure 1-2: Circular pond unit of the Jonkershoek Aquaculture Research Farm

1.2 Fisheries in Crisis

As an introduction to cage aquaculture we need to acknowledge the need for fish farming. This, however, is a debatable subject as some organizations believe that fish farming has more downsides than being a solution to over-fishing.

According to the World Wide Fund (2007) most marine fish are carnivorous, therefore if these carnivorous fish are being farmed, the aquaculture industry is using a large proportion of the fish caught in the world's oceans each year. Currently, one-third of the world's fish catch is used to produce fishmeal and fish oil. In 2004, the aquaculture industry used 87% of the world's fish oil and 53% of the world's fishmeal, with salmon farming alone using over half the global production of fish oil.

Many of the fish stocks used as feed - mostly anchovies, pilchards, mackerel, herring, and whiting - are already fished at, or over, their safe biological limit. So instead of relieving pressure on the marine environment, aquaculture is actually contributing to the overfishing crisis that plagues the world's fisheries (WWF, 2007).

It is now generally acknowledged that our fisheries around the world are in crisis (Greenpeace, 2007). The essential cause of the crisis is that the oceans' resources are considered infinite and deep. Rough and distant waters are no longer a barrier to modern fishing fleets, thus leaving no natural, safe haven for fish to escape to and replenish (Ogilvie, 2006).

Food from the oceans consists mainly of fish and shellfish. Over 90 million tonnes are caught annually, of which approximately 75% is eaten by people (half of which is preserved, usually in salt water and smoking, and the other half is chilled, frozen or sold fresh) and the remaining 25% is made into fish meal and oil used as pet and animal feed or fertilizer (Ogilvie, 2006; NOAA, 1998) .

In the distant past fish, as a resource, was used at a sustainable rate and people caught only as much as they needed. Since the 1970's, with the improvements in modern fishing technology, an increased number of people fishing and an increase in global population have caused the consumption rate to escalate exponentially. Modern fishing fleets make use of modern technology which includes airplanes, larger nets, radios and video sonar to locate schools of fish. With improved technology and the introduction to purse sein nets, long-line fishing, drift nets and factory trawlers, whole schools of fish are able to be caught with minimal effort (Ogilvie, 2006). Technologically advanced fishing fleets not only catch greater numbers of commercial species of fish, but they also catch millions of tonnes of unwanted marine life each year, as well as destroying coral and other sea-bed resources. This term is known as by-catch which can be defined as any catch of species (fish, sharks, dolphins, seals, turtles, sea birds, etc.) during fishery operations other than target species (Alverson et al*.*, 1994).

By-catch has two components, the non-target species catch that is retained and the non-target species that is discarded. By definition, by-catch is predetermined, while the decision to retain or discard may occur during the catching process, at some time later during the vessel trip, or in the harbours. Unwanted or undersized animals collected from a catch are discarded – thrown back into the sea, probably dying (Ogilvie, 2006, Alverson et al*.*, 1994).

In the National Oceanic and Atmospheric Administration State of the Coast Report (1998) it is estimated that approximately 27 million metric tons (30 million tons) of by-catch are discarded each year throughout the world's commercial fisheries, compared to a total of about 77 million metric tons (85 million tons) of valuable catch. Countless birds and other animals suffer and may die from injuries caused by swallowing or becoming entangled in discarded fish hooks, monofilament line and lead weights. Even the most considerate and careful fishers must share the blame, because every sport and recreational fisher's line will eventually become tangled in a tree branch and the line easily snaps during casting or gets snagged on the rocks (Ogilvie, 2006). Shrimpers, for example, catch 114 000 tons of shrimp per year but discard four times that in weight in mackerel, crabs and many more (Heilprin, 2005). In New Zealand, albatross population has decline 63% between the period 1973 and 1997. This decline is mainly ascribed to the fact that albatrosses are caught in the nets of fishing trawlers (WWF, 2007) each year.

Figure 1-3: Shrimp trawl catch. 95% of the catch in this photo that was not shrimp died on deck and was shoved overboard. (Norse, 2007)

Commercial fishing has improved vastly over the years. All improvements made in commercial fishing equipment are aimed at catching more fish, with little regard for ecological consequences. This has begun to change since biologists started tallying the loss of seabed ecosystems (crushed by repeated towing) and the vast unintended toll of sea turtles and unwanted fish swept into the gaping bags. Fishermen, too, begin to recognize that in capturing fish of all sizes they were undermining the health of the resource. Tightening laws have begun to shift and shape designs and practices that control the by-catch and ecological effect. Improvements in design may reduce ecological damage but will never prevent it entirely

(Ogilvie, 2006). The figure below gives an indication of fully exploited fisheries in 2002:

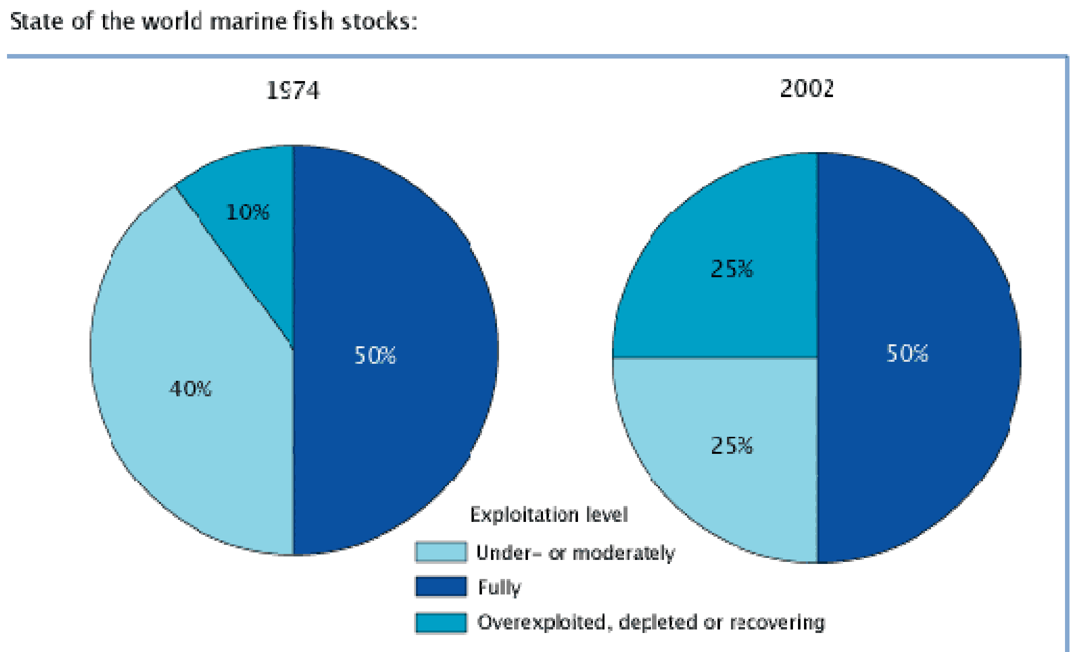


Figure 1-4: State of World Marine Fish Stocks (FAO, 2004)

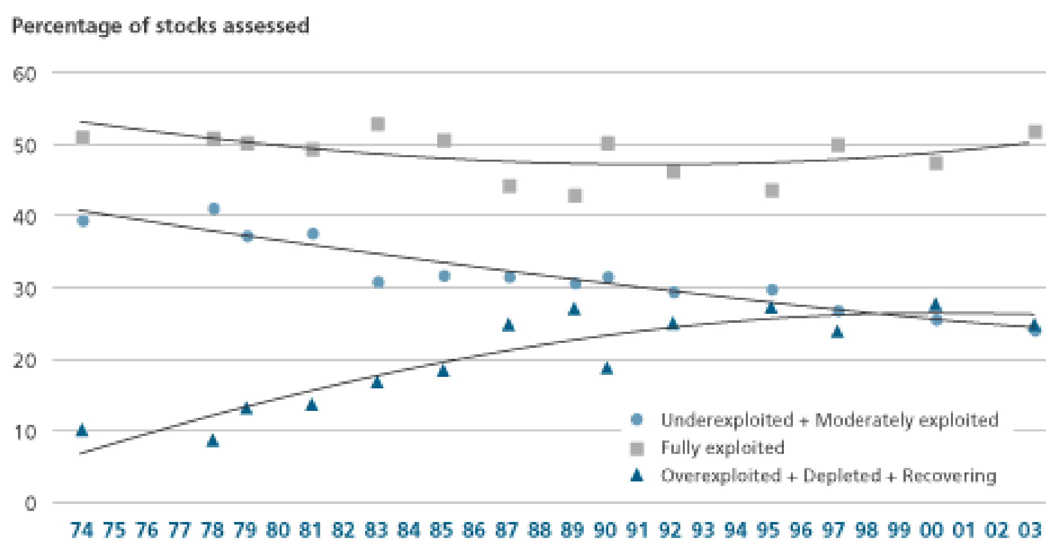


Figure 1-5: Global trends in the state of world marine stocks since 1974 (FAO, 2004)

Fish farming was developed to help alleviate the depletion of wild marine life from over-fishing and was also seen as a commercialising opportunity. While aquaculture may be an option for the rearing of some species of fish, there could possibly be inherent problems. The fish are believed to be unhealthy, because of the antibiotics, drugs and other chemicals used in fish farming. Releasing these antibiotics and chemicals into dams or ponds creates a perception that fish farms pollute. Some of the farmed fish are very expensive, the fish can escape due to damage to the net pens and fish farms can also spread diseases which can run out of control in densely packed fish farms. The idea behind aquaculture is that by keeping fish enclosed, production cycles can be manipulated and conditions optimized. This is done primarily by excluding predators and enhancing feed supply, but increasingly new biotechnological methods are being used. These methods include transgenic (transfer of genes from one fish to another), enhancements to improve growth, hormone therapy and vaccination to prevent diseases (Adelezi, 1998).

At present more than 220 species of fish are farmed, ranging from filter feeder to herbivores to carnivores (FAO, 1999). Aquaculture can alleviate pressure on wild fishery stocks either by increasing the production of popular fish (such as salmon), thus reducing prices, or by farming fish (such as tilapia) as alternatives to ocean fish. New technologies (such as hybridization), have been used in a number of species to make fish more profitable. This is done by increasing growth rate, improving flesh quality, increasing disease resistance and improving environmental tolerance. Techniques also allow sex to be controlled, and spawning to be induced. Aquaculture has also provided alternative employment in fisheries-dependent regions (FAO, 1999). In conclusion, fish farming may have its disadvantages, but it is the responsibility of fish farmers to manage their farms ethically and responsibly, ensuring that the advantages of fish farming outweigh the downsides, to provide a food source and employment and to be economically viable to their country.

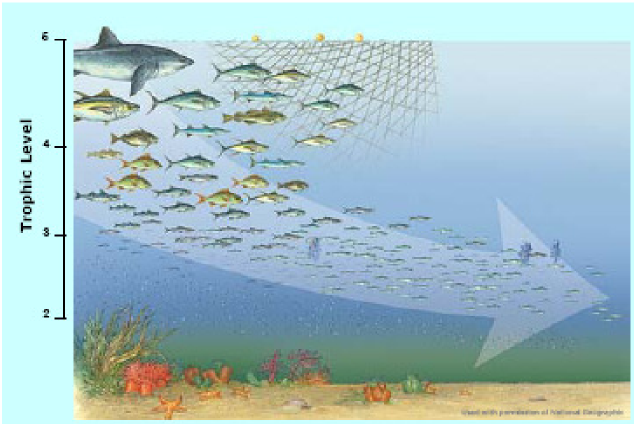


Figure 1-6: "Fishing down marine food webs" (Pauly et al., 1998)

"Fishing down marine food webs" (Pauly et al.,1998), illustrated by the blue arrow in the graph above, is the process whereby large, slow-growing predatory fish (with 'trophic levels' from 3.5 to 4.5) are overfished and gradually replaced in fishery landings, by smaller, fast growing fish and invertebrates (with trophic levels from 2.0 to 3.5). This process can be demonstrated through the decline of the mean trophic level of fishery landings, over time, and in different parts of the world.

1.3 The Development of Automated Feeding Systems

The factors giving rise to aquaculture production have been reviewed and the next step is to address the technological aspects of this project. The objective of the project is to find a way to develop an automated feeding system using a control system based on image analysis of fish feeding behaviour. In other words a machine will "learn" to distinguish between feeding behaviour and non-feeding behaviour. The control system therefore receives input (video image) which is encrypted (by a machine) to give you a desired output, illustrated as follows:

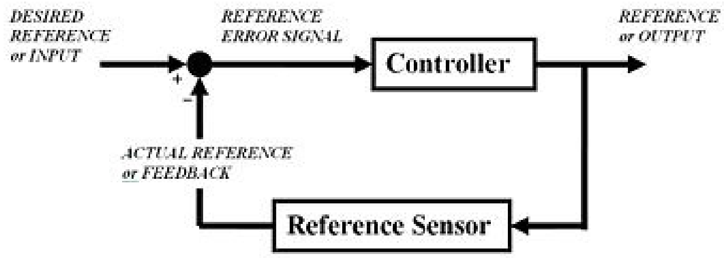


Figure 1-7: Illustration of a feedback control scheme (Franklin et al., 2002)

Process control is a statistical and engineering discipline that deals with architectures, mechanisms and algorithms for controlling the output of a specific process. In most modern processes large volumes of data are collected, which can be used in various ways to deal with the dynamic response of a system. Process data are also increasingly used in exploratory analysis aimed at the identification of relationships and abnormalities in multivariate systems, for process optimization and troubleshooting (Aldrich, 2001). Machine learning is concerned with the design and development of algorithms and techniques that allow computers to "learn" (Bishop, 2006). The major focus of machine learning research is to extract information from data automatically, by computational and statistical methods. Hence, machine learning is closely related to data mining, and statistics and theoretical computer science. Machine learning has a wide spectrum of applications including natural language processing, syntactic pattern recognition, search engines, medical diagnosis, bioinformatics and chem-informatics, detecting credit card fraud, stock market analysis, classifying DNA sequences, speech and handwriting recognition, object recognition in computer vision, game playing and robot locomotion (Alpayadin, 2004).

Some machine learning systems attempt to eliminate the need for human intuition in the analysis of the data, while other systems adopt a collaborative approach between human and machine. Human intuition cannot be entirely eliminated since the designer of the system must specify how the data is to be represented and what mechanisms will be used to search for a characterization of the data (Alpayadin, 2004). Machine learning can be viewed as an attempt to automate parts of the scientific method.

A machine learning algorithm addressed in this project is one of supervised learning in which the algorithm generates a function that maps inputs to desired outputs. One standard formulation of the supervised learning task is the classification problem: the learner is required to learn (approximate) the behaviour of a function which maps a vector into one of several classes by looking at several input-output examples of the function (Bishop, 2006).

A real appreciation for the data comes almost exclusively from exploratory graphical analyses of the data, which serves as a window into the essence of the system.

Pattern recognition is a sub-topic of machine learning. Pattern recognition aims to classify data (patterns) based on either a preceding knowledge or on statistical information extracted from the patterns. The patterns to be classified are usually groups of measurements or observations, defining points in an appropriate multidimensional space (Bishop, 2006). A complete pattern recognition system consists of a sensor that gathers the observations to be classified or described, a feature extraction mechanism (in our case scores) that computes numeric or symbolic information from the observations, and a classification or description scheme that does the actual job of classifying or describing observations, relying on the extracted features.

The classification or description scheme is usually based on the availability of a set of patterns that have already been classified or described which usually uses one of the following approaches: statistical (or decision theoretic), syntactic (or structural). Statistical pattern recognition is based on statistical characterisations of patterns, assuming that the patterns are generated by a probabilistic system. Syntactical (or structural) pattern recognition is based on the structural interrelationship features. A wide range of algorithms can be applied for pattern recognition, from very simple Bayesian classifiers to much more powerful neural networks. However, this section will be discussed in more detail in the methodology chapter, chapter 4.

Chapter 2

Literature Review of Automated feeding in Cage Aquaculture

2.1 History

There are many common tasks any fish farmer would find difficult to do, such as counting, sorting, measuring and weighing of fish without having to individually handle and stress the fish. These are critical needs in fish farming, as this is essential information for financing, insurance, stock-management and feeding activities. Past ongoing research has developed numerous fish feed monitoring, counting and measurement techniques without having to handle or stress the fish. For example, in the 1950's an electrical device was installed for counting salmon and kelts was developed, this operation was successful for periods of over twelve months (Juell et al*.*, 1991; Bjordal et al*.*, 1993).

In the 1970's developments in acoustic equipment and signal processing played an important role in advances in biological oceanography, as well as monitoring of a demand feeder. During the 1980's acoustic telemetric systems using 100 – 150 kHz transmitters were used to record data on fish such as position depth, temperature and heart rate. X-raying fish fed with spiked iron powder and rainbow trout feed spiked with radio-isotopes were used in nutritional studies. Microencapsulated fluorescent tracers in feed can directly monitor feed ingestion. During the 1990's hydro-acoustic techniques for monitoring uneaten feed dropping through the bottom of the sea cages were developed. Signal processing was found useful for counting and measuring fish without actually handling the fish. (Juell et al., 1991; Bjordal et al., 1993)

2.2 Current Status of Feeding Systems in Aquaculture

2.2.1 An Introduction to Feedback Control Feeding Systems

The conventional practice of hand-feeding is based on the use of feed tables, and the experienced eye of the feeder adjusting the feed quantity to suit the needs of the fish. Hand feeding is based on surface feeding behaviour or "feeding-frenzy". As cages and holding units have become larger and deeper, accurate visual observations of the fish have become more difficult. Two simple methods of improving information feedback of feed consumption are the airlift pump and the underwater video camera. Feed may be delivered by means of a feeder (for e.g. a feeding hopper), until a significant number of pellets are observed being drawn up through the pump by operators after which the feeder is turned off. More advanced systems may have pellet counters to provide an automatic feeder cut-off, and a facility for recycling the uneaten pellets (Juell, 1991; Blyth 1992; Bjordal et al., 1993; Myrseth, 2000; Beveridge, 2004). Alternatively, some farms now make regular use of submersible video cameras to observe the stock during feeding (Phillips et al., 1985; Kadri et al., 1991; Blyth, 1992; Thorpe and Huntingford, 1992). Again, this is usually to help the judgment of the person controlling the feeder. Further development in image analysis software could lead to "video footage" or "machine vision" being incorporated into automated systems, which is the aim of this project.

Figure 2-1: An example of a traditional Hand-Feeder (Gulf of Mexico offshore aquaculture consortium, 2003)

2.2.2 Adaptive Automated Feeding Systems

For a feedback system to be used in an automatic feeder, a controlling variable which can be analysed is needed to produce a feed control program. All systems rely on some method or another to detect uneaten feed, using a camera, a Doppler or an infrared detection system or a system that pumps water up from the floor of the cage. In a simple system, a farmer would simply use this information to stop feeding the fish, whereas in more sophisticated integrated interactive feedback feeding system, observations of uneaten feed are interpreted by software which will then control when the next meal will occur and the level at which the fish will be fed (Bulcock et al., 2001).

A different approach developed by several manufacturers uses hydro-acoustic sensors. These are usually suspended below a fish cage, facing the surface, to provide a sonar image of the cage contents (Juell et al., 1993). AKVA market a hydro-acoustic sensor which is claimed to distinguish pellets from other items in the cage (Blyth et al., 1993). The Peneye is marketed by Feeding Systems A.S., where in this case the hydro-acoustic sensor is optimised to measure the fish position and density within the cage (Bjordal, 1993). If feed is supplied when the fish are hungry, they will rise to the surface, descending when their appetite is diminished. The location of the fish therefore relates to changes in appetite, although it may be indicative of other events such as predator attack, changes in water quality and disease. The appearance of fish below the bottom net of a cage indicates over-feeding, which is attracting wild fish. Once again, analysis of the signals by software can provide the basis for controlling feeders, and could potentially be linked to alarm systems which are in turn will become anti-predator devices.

An early implementation by Simrad® estimated biomass and calculated the spread and distribution of fish size. Rapid changes in calculated biomass could be interpreted by software as indicative of a problem such as a hole in the net allowing fish to escape. The cost-benefit of these systems is strongly correlated to cage size, with the investment cost being easiest to justify on farms using larger cages.

2.2.3 Robots and Centralised Feeders

Inland based systems, particularly those with a large number of tanks (>30), Arvo-tec of Finland has developed a robotic system. The system consists of 1 to 4 feeder units, which move between tanks by means of an overhead track. A computer control system allows unattended operation, with the robot feeders guided by magnets in the track backed up with optical sensors. Each tank is fed according to the individual directions of its program. As the robot progresses, temperature and dissolved oxygen levels in the supply and discharge water can be measured and the values incorporated into the underlying feed model so that any required changes can be made to the feed supply. For example, if the oxygen level of the discharge water decreases to unsustainable levels, the system will begin to terminated feeding whilst sounding the alarm. The robot feeders are automatically refilled and this system offers a more cost-effective approach to placing a sophisticated feeder on each tank, more suitable for smaller feeding amounts and is more accurate that most centralised pneumatic systems (Euro Fish magazine, April 2001).

Centralised air-powered feed systems have been available for quite a few years. Feed is stored in central silos, from here the feed empties into a dosing unit from where it is transferred to an injector unit, main transport pipe and, via air from a blower, to its destination tank, pond or cage by a distribution valve and individual feed pipes. This method reduces labour required for feeding, but has a high capital cost. This method is also not suitable for widely spread or offshore sites (Euro Fish Magazine, April 2001).

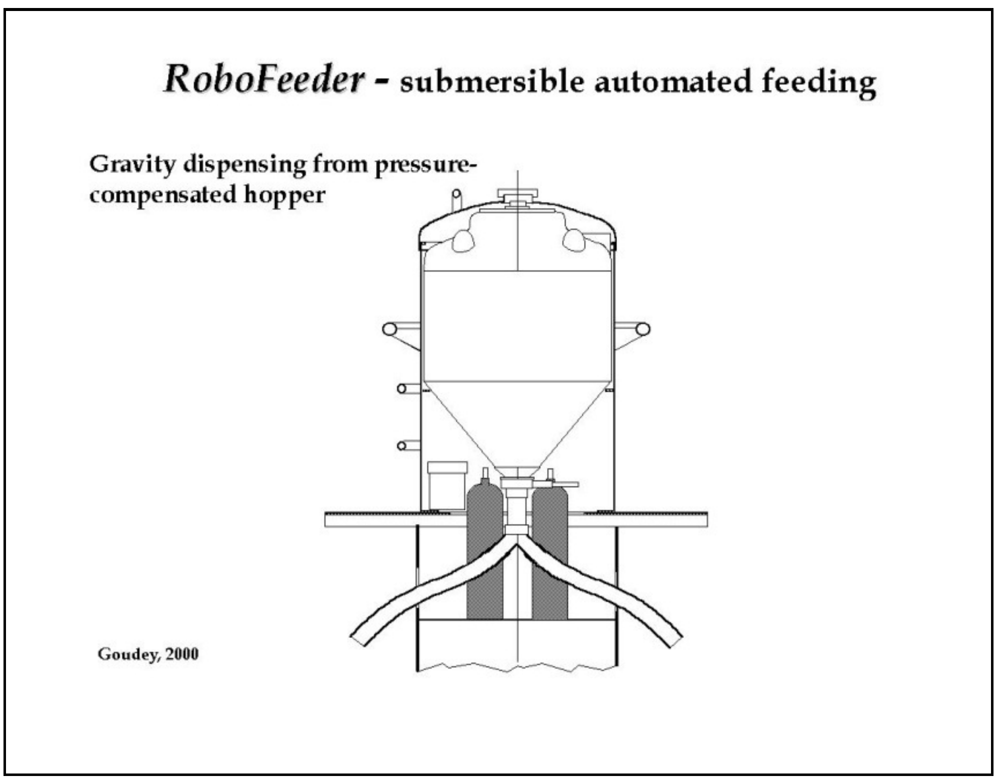


Figure 2-2: An example of a Robofeeder (Gulf of Mexico offshore aquaculture consortium, 2003)

2.3 Current status of Automated Feeding Systems

As it can be seen from the following articles, only certain areas in aquaculture engineering have been studied intensively. Research done in utilising improved feeding strategies has appeared to fall into three categories, what most researchers have found to be the best way to try and optimise the feeding conversion rate of caged aquaculture. These three categories are as follows: (1) Some researchers have discovered that by studying the wasted pellets below the feeding zone will indicate if the fish are satiated or not (2) Other researchers use hydroacoustic monitoring to study either the fish behaviour before, during and after feeding or to also detect feed pellets using hydroacoustics, and lastly, (3) feedback control systems have been developed in order to automate the feeding process. The next section explains these categories in more detail.

2.4 Detection of Uneaten Pellets below the Feeding Zone

2.4.1 Detection and Counting of Uneaten Feed Pellets in a Sea Cage Using Image Analysis (Foster et al., 1995)

The purpose of this study was to detect and count feed pellets in a sea cage using underwater video cameras. Using a light-compensating camera positioned straight down in the water column, extruded pellets appear white. Manual counting of feed pellets from video replay is arduous so algorithms were developed for detection and counting of the feed pellets. The algorithms were implemented on a computer based image processing system using actual feeding situations. The focus of their study fell on the counting of uneaten feed pellets. Knowledge of pellet loss is essential for assuring proper dosage delivery when the pellets contain flesh pigmentation agents, vaccines, vitamins or antibiotics. Therefore a feedback system was required to provide data on how much feed is not being eaten.

The objective of this study was to determine the validation of a manual video camera based pellet detection and counting system, and to develop an automatic version of this manual system using image analysis algorithms. The automatic pellet counting algorithm was developed using recorded image sequences. The sequences consisted of detecting the feed pellets and correctly tracking feed pellets from one image to the next.

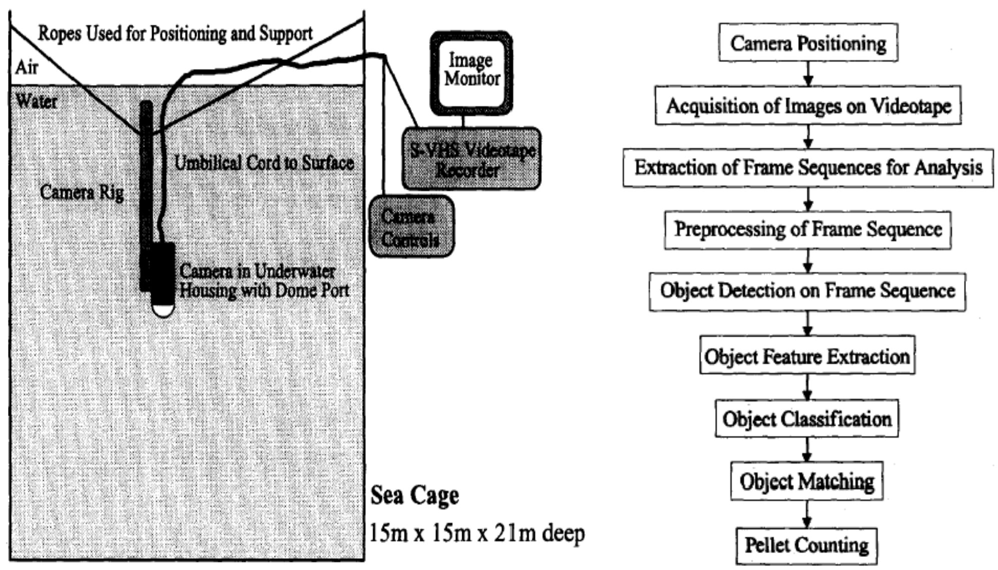


Figure 2-3: Experimental Set-up and Pellet Counting Process (Foster et. al., 1995)

The first test run was to count the feed pellets in the fish cage with no fish in the water in order to determine the accuracy of the counting algorithm. A computer error of approximately 10% occurred. These errors occurred during object detection, which was caused when feed pellets fell too close to the front of the camera. The camera blocked the light reaching the pellet, and therefore the pellet was not detected. Object division was another major cause of counting errors. Object classification was not very accurate; in very murky water this task will become even more difficult.

The algorithms which were developed in this project represented the first stage in development of commercial automatic pellet counting systems.

2.4.2 Control of Feed Dispensation in Sea Cages using Underwater Video Monitoring: Effects on Growth and Feed Conversion (Ang et al., 1997)

These researchers realized that uneaten feed is a detriment to farming, because it accumulates underneath in a cage where it is subject to microbial spoilage and attracts wild fish. The objective of this study was to develop a feed control system using underwater video cameras. The fish were fed in two experimental trials. The trials were conducted on Atlantic salmon (*Salmo salar L.*). Fish fed by mechanical feeders were monitored using an underwater video camera placed below the feeding zone. Data were collected as feeding response, growth and feeding conversion rate which was incorporated into a video sample and a standard statistical method was used to analyse the data. Environmental conditions were also taken into consideration. Fish in camera-monitored cages achieved significantly better feeding conversions and lower mortality rates due to more efficient feeding by visually observing their behaviour and feeding fish accordingly. During periods of poor water clarity, fish apparently had difficulty detecting pellets and therefore ate more poorly. In camera-monitored cages, feed dispensation rate varied depending on lighting conditions and broadcast method. Subsurface feeding activities and pellet loss constituted a better indicator of satiety in Atlantic salmon than did submerged feeding activities.

2.4.3 Detection of feeding rhythms in sea caged Atlantic salmon using new feeder technology (Blyth et al., 1993)

They found that fish display preferential feeding patterns that relate to both endogenous and exogenous factors. This new technology feeder is called the Aquasmart Adaptive Feeding System, designed to automatically feed fish by regulating feed input based on the levels of feed waste detected beneath a feeding zone. The system used this information to establish the feeding behaviour of the fish. The system fed a cage of Atlantic salmon, *Salmo salar* (2 – 3 kg) daily to satiation for four months during winter. This system is very similar to the systems mentioned above, except that a sensor is used to detect the pellets below the feeding zone. Large savings can be made by fish farmers by using these feeding control systems.

2.5 Hydroacoustic Monitoring of Feed Pellets during Feeding

2.5.1 Hydroacoustic monitoring and feeding control in cage rearing of Atlantic salmon (Salmo salar L.) (Bjordal et al.,1993)

An echo-sounder, linked to an upward-facing transducer mounted under the cage, provides echo signals from the caged fish. The signals are processed by a PC-based echo integrator which monitors the change in echo intensity at different depths in the cage. Before feeding the highest fish densities are found at medium depths. When feeding starts, fish density and thus echo intensity close to the surface increase significantly and stay at a high level as long as the appetite remains high. When the echo intensity in the upper layer of the cage decreases to a certain preset threshold, the feeders are automatically shut off. The downward migration of the fish is shown by a reduction in echo intensity and is thus used as an indicator of satiation.

Besides the direct feeding control application, the amount of feed is logged, and the software allows different feeding strategies, for example, with respect to the number of daily feedings, feeding intensity, total daily feed limits, and the level of automation to be implemented. This system was designed to monitor and control twelve cages and feeder units. Other useful applications as continuous monitoring of the fish with possible alarm functions, observation of dead salmon and wild fish, as well as possible biomass estimation can be implemented using this method. This information is very useful in learning about fish feeding behaviour.

2.5.2 Demand Feeding in Salmon Farming by Hydroacoustic Feed Detection (Juell et al., 1993)

A new method for demand-feeding of salmon in sea cages where automatic feeders are controlled by a feed detector was investigated. Hydroacoustic detection of feed pellets at 2.5 m depth was used as an indicator of reduced appetite. Feeding was terminated when echo energy from feed pellets sinking through a 360º acoustic beam exceeded a preset threshold. In an 83-day full-scale test, the feed intake and growth of salmon whose feeding was controlled by this method (detector group) was compared with those of fish fed in accordance with growth rate estimates (control group). The specific growth rates (% wet weight/day) were 1.01 and 0.71 in the detector and control groups respectively. This difference in growth was mainly explained by a considerably higher feed intake in the detector group. The results indicate that demand feeding by hydroacoustic feed detection automatically adjusts the feed ration to fish appetite, so that feed waste is avoided and the growth potential of the fish is utilized. In other words, as soon as feed pellets were wasted the feeder switched off. Thus the group of fish that was in detector group, showed a significantly better feeding conversion ratio.

2.5.3 Observing behaviour and growth using the Simrad FCM 160 fish cage monitoring system (Dunn & Dalland, 1993)

Instrumentation that uses underwater acoustics to monitor the behaviour and growth of salmon and trout in open water fish farms was successfully developed. The vertical distribution and behaviour of the fish in cages were monitored which could provide an indication of the well-being of the fish. Estimates of biomass, average size, size distribution and growth trends for the fish in each cage can be provided from this instrumentation. This information is very valuable to any fish farmer as mentioned before.

2.5.4 Acoustic characteristics of two feeding modes used by brown trout (*Salmo trutta*), rainbow trout (*Oncorhynchus mykiss*) and turbot (*Scopthalmus maximus*) (Lagardere et al., 2004)

Under conditions of intensive culture, the acoustic signals produced by fish during feeding depend on their feeding mode. Exclusive suction, used by turbot, is characterized by a maximum acoustic energy in the frequency rage 7 – 9 kHz and a sound duration of about one minute depending of time duration of pellet distribution. Suction feeding in conjunction with forward swimming, as employed by brown trout and rainbow trout, had a maximum acoustic energy in the frequency range 4 – 6 kHz and feeding sounds were measurable only for short periods (less than 1s) in between tow pellet distributions by hand. The brevity of these feeding sounds requires adapting the turbot acoustic-detection systems to actively feeding fish for developing automated feed distribution systems feasible in trout aquaculture.

2.6 Feedback Control Systems used in Cage Aquaculture

2.6.1 Automatic Feeding Control System for Fish (Kimura et al., 1993)

Marine farming systems such as those used in Japan, use radio equipment to provide telemetric control of sound emission and bait feeding. One of the biggest technical problems in these marine farming systems is not being able to transmit high-quality signals from fish finders within the signal transmission bandwidth allowed by the radio wave laws of the country. Therefore sizes of schools of fish are overestimated because of significant reduction in the signals received. The amounts of feeding are determined according to the measured sizes of schools of fish; so that over-estimation of the school sizes causes excessive feeding. Excessive feeding not only makes the system less economical, but also creates the serious ecological problem of marine pollution owing to residual baits. A method was therefore developed to enable signals to be transmitted from a fish finder without being affected by the legal limit signal transmission bandwidth of the country. In a country where these methods are used, this method may be very useful in estimating school sizes off-shore.

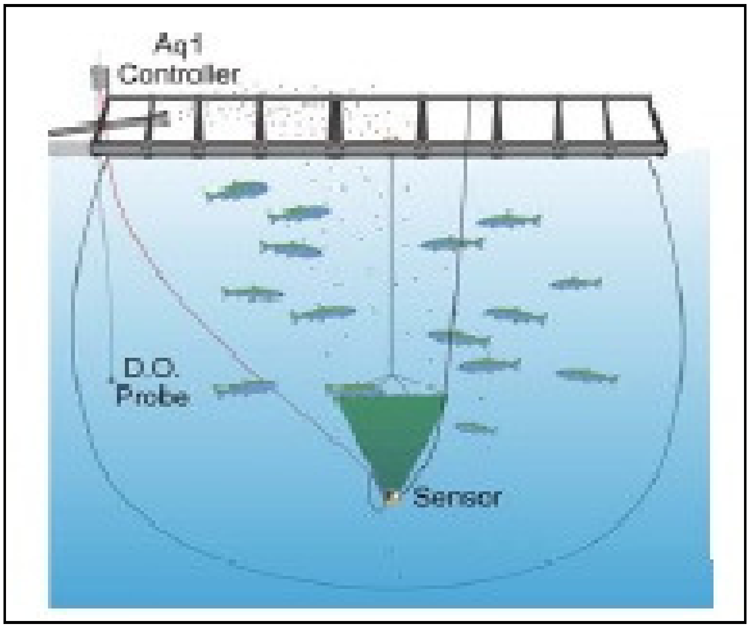


Figure 2-4: AQ1 system (www.AQ1systems.com)

2.6.2 An Automated Rearing Chamber System for Studies of Shellfish Feeding (Smith et al., 1998)

Producing large volumes of high quality microalgae to feed shellfish and other organisms is a limiting factor in the development of the aquaculture industry. Feeding regimes yielding the highest conversion efficiencies of algal feed to molluscan growth are required to maximize the return on algal culture investments. In the past, twelve specialized, manually controlled molluscan rearing chambers have been used to study nutritional requirements and growth of oysters, clams, and scallops. A computer-controlled, solenoid-valve system was added to automate seawater flow, volume of microalgal feed delivered, and feeding duration independently for each chamber. Each chamber represents a model for a programmed nursery system. These systems can benefit shellfish production enormously, as shellfish has become a very limited feed resource.

2.6.3 Automation of Feeding Management in Cage Culture (Myrseth, 2000)

An experiment involving three automated systems, the Aquasmart, Ecofeeder and Lift Up, was conducted in order to investigate how these systems compare in the improvement of control of feeding and help to improved feeding efficiency. The three systems tested gave feed conversion rates of 0.94 for Lift Up, 0.93 for Ecofeeder and 1.05 for Aquasmart. Thus, all these systems provide good control and any system could be implemented to assist in feed control.

2.6.4 Denitrification in aquaculture systems: Example of fuzzy logic control (Lee et al., 2000)

Nitrification in commercial aquaculture systems has been accomplished using many different technologies (e.g. trickling filters, fluidized beds and rotating biological contractors), but commercial aquaculture systems have been slow to adopt denitrification. Denitrification (conversion of nitrate, NO_3 to nitrogen gas, N_2) is essential to the development of commercial, closed, recirculating aquaculture systems in order to create an environment, as it would have occurred naturally. The problems associated with manually operated denitrification systems have been incomplete denitrification. These problems could be overcome by the development of a computer automated denitrifying bioreactor specifically designed for aquaculture. A fuzzy logic-based expert system replaced the classical process control system for operation of the bioreactor, which is continuingly optimizing denitrification rates. The fuzzy logic rule base was composed of more than 40 fuzzy rules, and took the slow response of the system into account. The fuzzy logic control system decides which process to follow taking into account a certain number of variables. This is a good idea in simplifying control of certain processes. This is not necessarily relevant to feeding utilisation, however this is an alternative approach which could be considered for future reference.

2.6.5 An automated feeding system for intensive hatcheries (Papandroulakis et al., 2002)

It is imperative to continuously meet metabolic demands via continuous online feed supply for successful aquaculture. This is especially important for small larvae, which have a relatively high metabolism, with a long photophase and require continuous feeding. Such requirements cannot be easily met using classic manual feeding methods due to logistic problems. A computerized system was consequently developed for feed management in intensive hatcheries. The daily plankton requirements of larvae, organized in feeding tables together with a distribution patter, was used for the development of the required hardware and software to control feeding. The system computes the feed required (plankton organisms) and activates a peristaltic pump and solenoid valves for distribution to tanks. The system offers the option of feeding according to tables or manually, depending on the concentration of plankton in the rearing tanks.

2.6.6 Design and Analysis of an automated feed buoy for submerged Cages (Fullerton et al., 2004)

A research prototype was developed to supply a submerged net pen at an exposed site south of the Isles of Shoals. The Isles of Shoals is a group of nine small islands situated approximately 16 km off the east coast of the

USA. The system, designed for a quarter ton feed capacity, consists of a surface feed buoy, rubber tether moorings attached to a submerged grid, a feed transfer hose, feed dispensing machinery, and telemetry/control components. The buoy is taut-moored above the cage by flexible members in order to allow for tidal range and large storm waves. The feeding mechanism uses a small, electric powered pump to actively force feed slurry down to the cage. A wind generator and solar panels provide power to the various pumps on a user-set schedule and also monitors the operation of the electric power system. This is not necessarily relevant to feeding utilisation; however this approach to automation can be a very good idea for future reference.

2.6.7 Development of an Intelligent Feeding Controller for Indoor Intensive Culturing of Eel (Chang et al., 2005)

This development was conducted by observing the gathering behaviour of eels using an infrared photoelectric sensor with a digital signal output. Timer-controlled automatic feeders with a rotating plate and scrubber were used widely in this form of culturing which was designed to run at preset times for a preset duration with no other control.

An intelligent feedback control system was developed in this project. The feeder equipped with such a sensor and governing control strategy is able to stop feeding according to the gathering behaviour of the eels. The control strategy is based on six user-adjustable parameters with default values. If the sensor fails to detect the gathering behaviour for three consecutive trials, the feeder will stop feeding until the next wake-up, thus reducing the risk of polluting the water. In the case of eels, their gathering behaviour is the key component to their satiation point. The control of this feeding controller is very important, and if this project is feasible, this method should be taken into consideration.

2.7 Fish Behaviour as a basis for an Automated Feeding System

The most common instruments used to monitor fish behaviour are with the use of tagging, X-radiography or video recording.

In nature, competition between individuals for limited resources (feed, feeding sites, and hiding places etc.) often results in the development of dominance hierarchies. This is a common feature in fish and leads to disparity in feed intakes, growth rates and survival. Aggression is known to increase in dominant fish, especially when feeding, while stress and fin damage increase in subordinate fish (Smith et al*.*, 1993).

Feeding patterns in any fish vary due to a number of factors, for example in a cold fresh water fish like rainbow trout, it is known that the fish feed very poorly during the summer seasons as the water temperature is too warm at the surface, and during winter months when the water is cold (usually below 14°C) the fish eat well as this is an optimal temperature to feed (De Wet, 2007). Feed intake varies depending on how stressed fish are due to poor water quality or predator behaviour. However, fish feed poorly when the water clarity is poor as the fish cannot detect their feed (Smith et al*.*, 1993). These events are supported by the following articles discussed:

2.7.1 Daily and Seasonal Patterns in the Feeding Behaviour of Atlantic salmon in a Sea Cage (Smith et al., 1993)

Smith et al*.* (1993) researched the variation in behavioural indications of appetite in Atlantic salmon in a sea cage. The variation in appetite was related to environmental variables and fish swimming activity during the seasons from autumn to spring. They discovered a marked seasonal variation in feeding behaviour which indicated a reduction in appetite from autumn to winter and a rapid increase in appetite from late winter onwards. Seasonal variation in behavioural indications of appetite was more closely related to how long a day was and the change in how long a day was than to other environmental variables including water temperature. A feeding regime based on the assumption that the water temperature is the most important environmental determinant of the appetite of salmon in sea cages could lead to feed wastage in autumn and early winter and under-feeding in late winter and early spring. The responsiveness of salmon to feed varied significantly throughout the day, but the overall pattern of appetite was different at different times of year. They also discovered that no marked morning and evening peaks of appetite occurred and prefeeding swimming speed was not closely related to appetite which is in contrast to some other studies.

2.7.2 Diurnal and Seasonal variation in the Feeding Patterns of Atlantic salmon in a Sea Cages (Blyth et al., 1999)

Atlantic salmon in sea cages had been known to exhibit feeding patterns that vary both diurnally and seasonally. Previous studies had been conducted on this section; however, no data reporting on the feed rate of a complete annual cycle had been recorded. From this project it was discovered that a major feeding peak regularly occurred soon after dawn, and feeding rates remained high for approximately one hour. Over the remainder of the day, the fish fed at a lower but steady rate. Relative feed intake varied over the trial, being initially high in summer followed by a sharp decline in autumn, and then further declining until fish reached harvest size at the beginning of the following summer. However, they recommend further investigation of the relationship between variation in annual feeding patterns and environmental factors should be carried out.

2.7.3 Patterns of feed intake in four species of fish under commercial farming conditions: Implications for feeding management (Talbot et al., 1999)

Meal durations and feed ingestion rates were measured in sea caged Atlantic salmon, rainbow trout, yellowtail and Red Sea bream; which were fed dry extruded feed in discrete meals. At a specific population level, satiation times in yellowtail, salmon and trout were typically about 15 – 25 min, but in red sea bream time to satiation was longer; about 60 – 90 min. In all these species, feed ingestion rates declined progressively during the course of the meal as fish became satiated. The water temperature had little effect on ingestion rates, possibly because the fish were fed 1 to 3 meals per day, but may have standardized hunger levels at the start of meals. Yellowtail ingested feed at approximately 3.5 kg feed per ton fish per minute at temperatures of 18°C and 28°C, whereas red sea bream ingested feed at rates of 0.6 and 1.4 kg feed per ton fish per minute at 26.5°C and 18°C respectively. In rainbow trout no waste feed was collected during the first quarter of any meal, so initial feed ingestion rates were restricted by the feed delivery rates to 0.5 – 09.kg feed per ton fish per minute. The fish were eating approximately two to four pellets per minute at the start of the meals.

2.7.4 Studying visual cues in fish behaviour: A review of ethological techniques (Rowland, 1999)

A variety of approaches are available to fish ethologists to study the role of visual cues in fish behaviour. Examples of these varieties is using live fish, mirror images, dummies (i.e. models), or video playback as stimuli to investigate fish behaviour. These examples represent a diversity of functional categories of behaviour exhibited by fish, including aggression, courtship, schooling behaviour, parent-offspring, predator-prey and cleaner-host interactions. These specific techniques have been used by fish biologists to control or manipulate body shape, size, posture, morphological structure, colour and making patterns or movement of fish. Vision is the dominant sense of many fish; much of what is known about visual communication in fishes comes from observing freely interacting subjects. The fish interact lively, or to a mirror image, dummies or video playback while being observed by the researcher. A specific characteristic that is of importance to this project is the movement or feeding behaviour of fish. This study suggested that the least intrusive way to study movement cues is to record the behaviour of freely interacting fish and to test for correlation among the behaviour each performs. This is a type of statistical approach to investigate how a display of a couring guppy elicits response from its partner.

2.8 Summary

Currently aquaculture farms are feeding fish by means of traditional hand feeders and automated feeding systems using hydroacoustics and sensory feedback systems. These approaches however are not considered optimal as automatic feeders do not necessarily ensure optimal feed intake. Social dominance using demand feeders does not allow even feeding distribution among all sizes of fish.

An alternative approach to developing an automated feeding system will be investigated in this project. The aim of the project is to investigate whether an automated feeding system can be developed based on characteristic feeding behaviour of fish, whereas most other automated feeding systems operate on other variables, like detecting uneaten feed pellets as reviewed in the literature.

Video footage of fish feeding behaviour before, during and after feeding will be analysed using a computer software program Matlab®. From the image analysis done on the video footage, it is anticipated that characteristic feeding patterns before, during and after feeding should indicate the level of appetite of the fish. For example, famished fish could swim more aggressively and closer to the surface just before feeding, while a more satiated fish swims more calmly and closer to the floor of the cage. If this can be proven viable, an automated feeding system could be developed using the models obtained from the video data.

The aim of this project will be to explore the feasibility of developing an automated feeding system by studying the feeding behaviour of rainbow trout in the Jonkershoek Aquaculture research dams in Stellenbosch. This will in turn is anticipated to optimise the feeding management system of the farm, thereby reducing fish food waste as well as preventing underfeeding of the trout. Automated feeding systems based on characteristic feeding behaviour of fish would be more profitable to an aquaculture farm as it could possibly replace hand feeders and could possibly feed the fish more accurately based on their behaviour. Many factors will influence the feeding behaviour of the fish and all environmental factors will be taken into account. Since there are only females species present in the cages, no mating behaviour will influence the feeding behaviour of the trout. Male rainbow trout are essentially smaller and are only needed for breeding purposes.

From the literature review a lot is learnt about fish behaviour and on what basis other automated systems work. Problems encountered in other researcher's projects bring awareness to problems which could be encountered in this project. Since aquaculture is growing so fast and has such a large role in agriculture production it is important to learn as much as possible about aquaculture. Since 60% of all expenses on an aquaculture farm are due to feeding, any contributions made to improving a feeding management system will make an aquaculture farm more economically feasible

Chapter 3

Experimental Setup

3.1 Introduction

The experimental set-up can be divided into two sections, namely the experimental set-up at an aquaculture dam and the experimental set up at an aquarium. Initially data were to be obtained from Jonkershoek Research Aquaculture farm, where a light-compensating camera in a water-proof casing was lowered into the cage and video footage was acquired. However, due to the difficulties explained in section 3.3, an alternative experimental set-up was sought.

Different options for the experimental set-up were considered. The first option considered was to acquire video data from the Katze Dam in Lesotho, this however did not realise as the suitable equipment (i.e. appropriate video camera) could not be attained. Another option would also be to observe fish in a home aquarium (i.e. goldfish in a fish tank), however it was anticipated that this experimental set up would not truly replicate the conditions of the aquaculture farm.

Therefore it was decided to acquire video footage from the Two Oceans Aquarium at the V&A Waterfront in Cape Town, South Africa, as the conditions were more similar to the aquaculture farm (i.e. where traditional hand feeding methods are used and feeding patterns are already established).

Clearance was arranged with the senior aquarist, David Vaughn. Video footage was taken of fish behaviour before, during and after feeding. The fish in the aquarium are fed once a day and the sharks are fed once a week, whereas on an aquaculture farm, fish are often fed more than once a day so as to obtain a maximum feed intake and hence growth rate. The fish in the aquarium are fed sardines and squid and on fish farms they are fed a balanced diet of balanced feed pellets. In this project, it not assumed that all fish have the same feeding behaviour characteristics. As it will be explained below, the focus of the investigation was on the movement of the fish and not on their morphological characteristics; therefore having different species of fish was not a concern.

3.2 Experimental Set-up at Aquaculture Farm

The video data acquired in this project is in the form of video or *.avi format files. Data were acquired from the Rainbow Trout farm at the Jonkershoek cold-water aquaculture research unit (Stellenbosch, South Africa) using a light compensating surveillance camera, as indicated in Fig. 3.1. A waterproof casing was constructed in order for the analogue camera to be lowered into the water, which is then connected to a portable battery and an analogue-to-digital converter (Pinnacle Studio 500 USB®), as shown in Fig. 3.2. The analogue converter is then connected to an external USB port, which in turn is connected to a portable computer where the data are captured to a hard drive. Real, live video footage of the submerged cage is displayed on the portable computer’s monitor. The camera can be submerged up to 20 m deep; however the visibility is highly dependent on water clarity. The portable battery must be charged before the camera is connected (as explained above) and then the camera is lowered into the cage. Once all is connected, recording starts and the video footage is captured to the computer’s hard drive.

- The video surveillance system uses Black and White CCD camera with 41 IR LED's.
- Operates on a 12 volt DC battery.
- Camera works in total darkness.
- Use for bait monitoring, predator hunting, wildlife observation.

Figure 3-1: Ultrec Video Surveillance Camera

Figure 3-2: Portable 12V Battery and Analogue to Digital Converter (Frame Grabber)

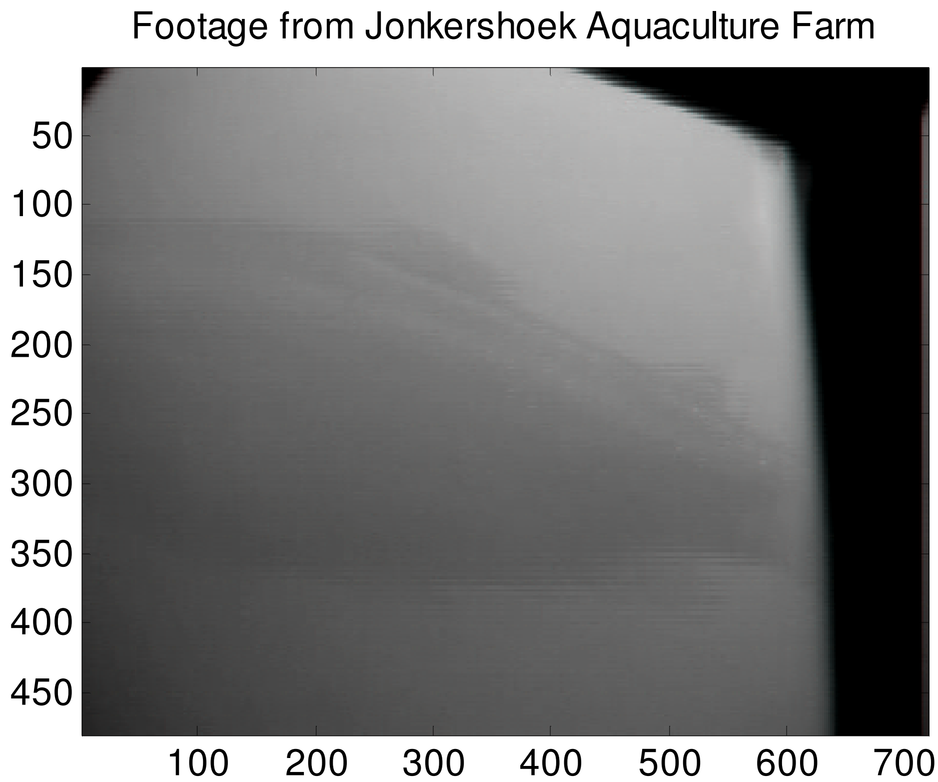


Figure 3-3: Video footage from Jonkershoek Aquaculture Farm

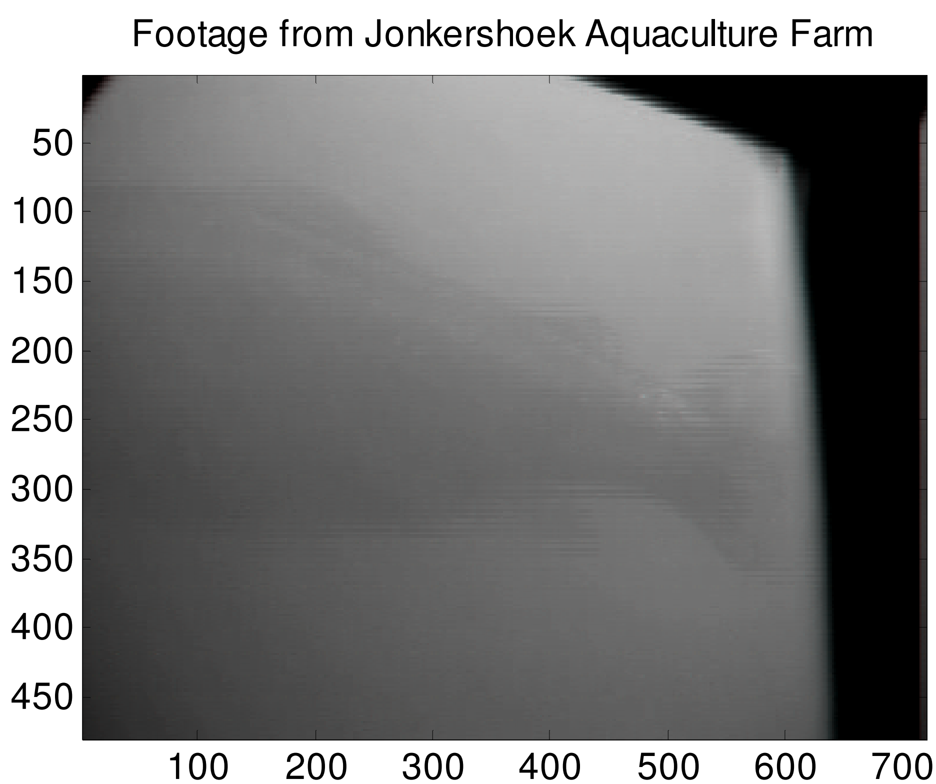


Figure 3-4: Video footage from Jonkershoek Aquaculture Farm

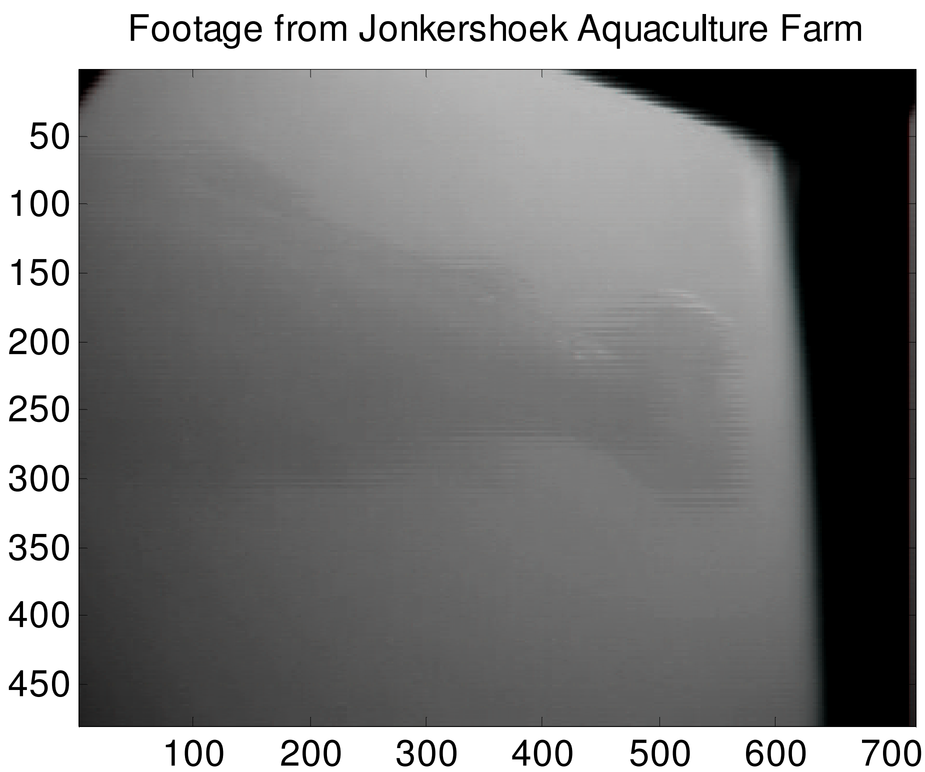


Figure 3-5: Video footage from Jonkershoek Aquaculture Farm

The surveillance camera captures approximately 30 frames per second. The frames shown in figures 3.3 to 3.5 are frames captured in intervals of approximately 10 seconds. As it can be seen from the frames, the visibility is very poor, and almost no changes can be detected from the frames.

3.3 Experimental Set-up at Aquarium

The video data from the Two Oceans Aquarium were captured using a three charge coupled device (3CCD) Panasonic® PV-GS150 digital video camera. The camcorder was set up on a tripod in an indoor aquarium, where no external environmental factors (such as changes in light) could influence the video data. Each data set consisted of approximately 2×10^4 image frames making up a video clip of approximately 15 to 20 minutes.

Each image frame contained the pixels of the image in a 480 x 720 matrix. Each of these frames was converted to matrix, the entries of which represented the intensities of the pixels, i.e. colour information was discarded, was the focus of the investigation was on the movement of the fish and not on their morphological characteristics. Matlab® was subsequently used for further analysis of the data. Typical prefeeding, feeding and postfeeding behaviour are illustrated in Figs 3.7 – 3.9 respectively. (The species of fish in the fish tank are cob, yellow tail, stumpnose and steenbras).

The reason for not using the same camera was firstly because the digital video camera was easier to use inside the aquarium, and the data could be transferred to a computer with ease. The surveillance camera consists of heavy and lengthy cables, as well as portable batteries which is required for the camera to function, which is difficult to manage inside the aquarium.

Figure 3-6: Panasonic Digital Camcorder

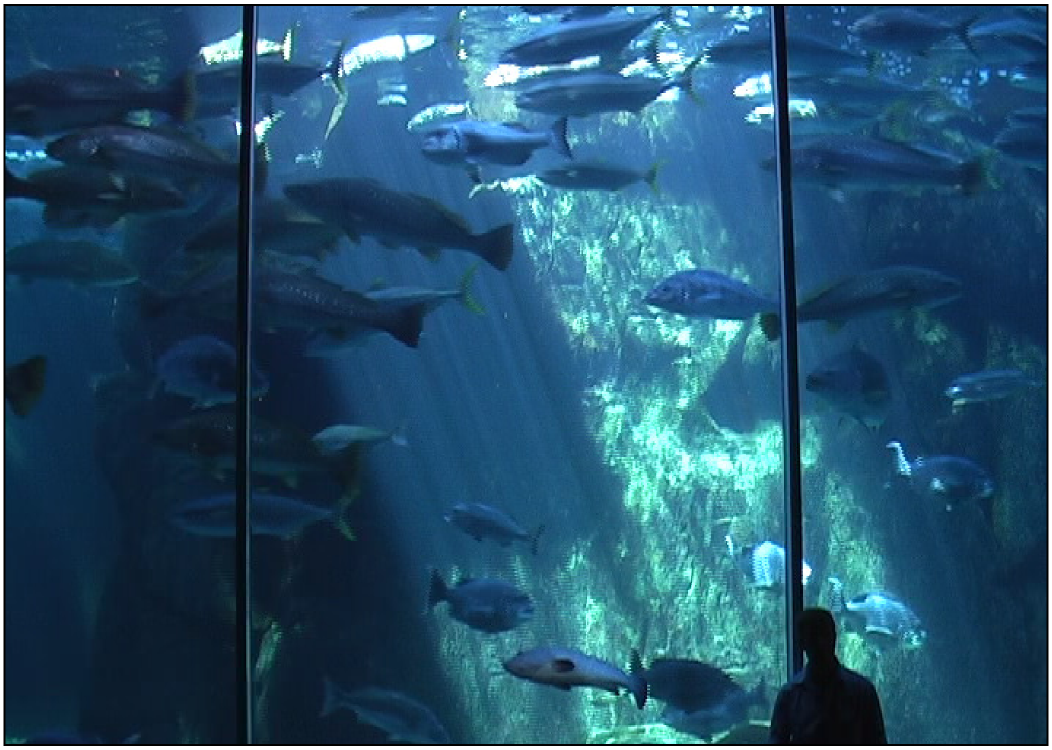


Figure 3-7: Prefeeding Behaviour

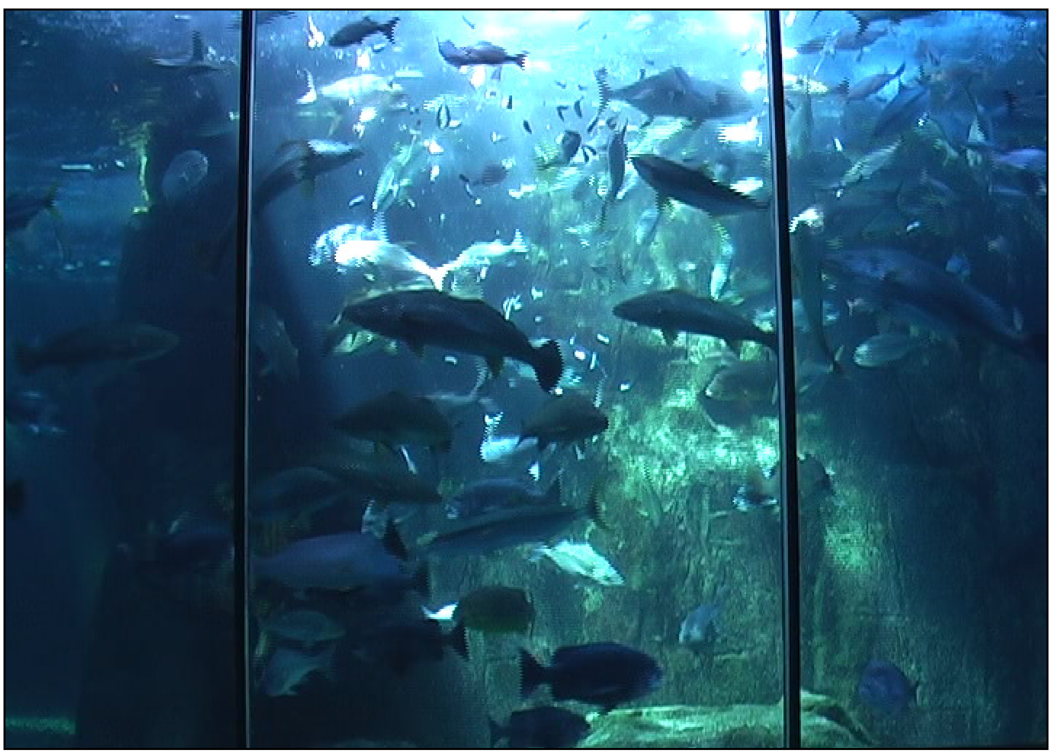


Figure 3-8: Feeding Behaviour

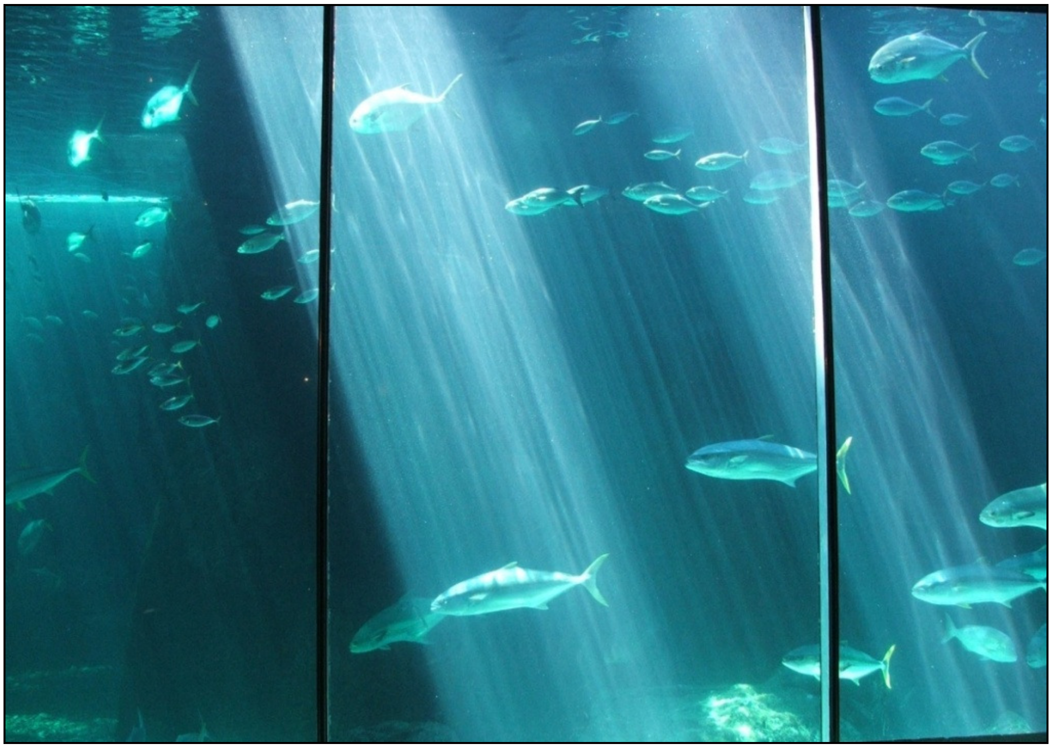


Figure 3-9: Postfeeding Behaviour

3.4 Practical Issues

3.4.1 Water Turbidity

The Jonkershoek Aquaculture research farm has murky water throughout the year. At the majority of aquaculture farms, the water becomes clearer during winter, but as Jonkershoek has high rainfall during the winter season the dams remain murky due to runoff from the surrounding hills. Video footage acquired from the Jonkershoek Aquaculture farm had almost zero visibility and therefore the video data could not be used.

A Secchi disc reading is a measurement of water transparency. A disc is mounted on a pole, and lowered into the water. The depth at which the pattern on the disc is no longer visible is taken as a measure of the transparency. Therefore, the higher the Secchi disc reading, the clearer the water.

The table below shows the results of Secchi readings measured at agriculture research dams in the Stellenbosch vicinity from 2004 – 2007. As it can be seen from the table, the Secchi readings vary considerably which will make it difficult to use the proposed video footage as a basis for fish feeding behaviour, and an alternative method should be sought to overcome turbidity issues.

Table 1: Secchi Disc Readings in cm ARC Research Dams (Direct Visibility Readings from Nietvoorbij dam).

	WRC research dams			
	NV 2004	**NV 2005**	**NV 2006**	**NV 2007**
Jan		84		
Jan		88	175	180
Feb	173	90		
Feb	165	70	170	200
Mar	156	85		
Mar	168	60	200	200
Apr	200	56		
Apr	94	53	110	125
May	124	50		
May	194	55	27	110
Jun	215	70		
Jun	314	83	30	85
Jun	286		105	
Jul	152	200		88
Jul	124		70	
Aug	105			43
Aug	126	110	95	65
Sept	114	105	130	105
Oct	113	170	110	95
Nov	83			
Nov	111	80	165	
Dec	150			
Dec	90		190	
Dec	110	120		

The only visibility data available for Jonkershoek Aquaculture dam is the following:

Jonkershoek:

18.10.06 40 cm

28.3.07 42 cm

27.06.07 200 cm and more

The readings given above do not provide enough information to form an opinion; however one can see the variability in turbidity readings at these times. Jonkershoek does not have a lot of phytoplankton, so the high turbidity of the dam is mostly influenced by sediment inflow and stirrup.

3.4.2 Video Recording Equipment

In order to obtain the desired images from the video data, provision should be made for the poor light under the water using a light compensating camera (infrared, LED's). An important aspect to bear in mind when recording video images is to maintain a desirable distance from the fish (to capture the cage as a whole); moreover it may occur that one fish will block the entire view of the camera.

Due to the turbidity of the water, the visibility length is approximately 30 cm. The camera was dropped directly into a cage; therefore there was no distance between the camera and the fish, where in some cases one fish could block the entire view of the camera. It would be more ideal to drop the camera a distance away from the cage in order to perceive a better image, but since the visibility length is only approximately 30 cm, it was not feasible in this instance.

3.4.3 Processor Memory

Another constraint to this project is the volume of the data sets. The computational algorithms required to describe the fish feeding behaviour rely on Matlab® software. The minimum specification of the computer memory required was 512 MB RAM, which was thought to be sufficient to perform the computation of the algorithms. However, this amount of memory proved to be insufficient to compute a significantly large enough data set. The hardware was constrained, and therefore Matlab® was only able to process approximately only 100 image frames from a complete data set, constituting approximately 20 000 image frames. Therefore an alternative software tool had to be incorporated to process the algorithms to overcome memory limitations. The software tool Gist® was recommended (the operation of Gist® is explained chapter 4). It is recommended that in the future processors with at least 2 GB RAM (or the maximum quantity of RAM obtainable) should be considered for the computation of these large multivariate data sets.

3.4.4 Feasibility of the Project

An automated feeding system (able to classify feeding state automatically using live images of aquaculture cages) would be able to replace hand feeding and possibly other automated feeders such as demand feeders which are not as effective. Live images of aquaculture aid in the monitoring of aquaculture for diseases, predators and water quality. At Jonkershoek aquaculture farm it would be very difficult to implement a feeding system based on an underwater camera system, seeing as the visibility length is only 30 cm. Cage aquaculture is growing at a considerable rate and therefore any research contributing to the improvement of automated feeding systems is an asset. However, it is important to note that in the case of automated feeding systems, based on visual observation and analysis, the water quality of the aquarium or aquaculture farm ought to comply with the requirements as mentioned in the previous section.

It is recommended that for aquaculture farms having poor visibility, an adaptive automated feeding system (such as the AQ1 adaptive feeding system, as described in chapter 2) are used. The adaptive feeding system for cultured species includes: (a) providing (i) a sensor able to detect feed particles passing through a sample area, and (ii) a control unit, including computer data storage age media in communication with the sensor, and (b) detecting and discriminating feed particles that pass through the sample area; wherein the control unit is able to process information obtained from the sensor and regulate subsequent feed output based on algorithm parameters (said algorithm parameters determine the instantaneous feed rate of the cultured species to adjust and match the preferred feed values meted to the cultured species at any given time).

Chapter 4

Image Processing and Data Analysis

The primary objective of this project was to generate a predictive model for fish feeding behaviour. Very little work has been done on developing an automated feeding system based on this method. However, these feeding behaviour patterns are in all probability strongly associated with the controlled environment in which they exist. In most cases, fish will are conditioned to their surroundings, therefore in a controlled environment such as an aquaculture farm or in an aquarium, fish become conditioned to know when and how they will be fed. Therefore in this experiment, it was not possible to observe fish feeding behaviour where there is an abundance of fish feed or how fish will behave when they are fully satiated.

The aim of this chapter is to describe the techniques used to analyse the data in this project. In the methodology it is firstly explained how to acquire the data and how it is pre-processed or prepared for data analysis, then the analysis is explained followed by the classification. The analytical techniques used in this project were principal component and linear discriminant analysis.

In this chapter principal component analysis is first reviewed. At a later stage, in order to optimise the classifier, kernel principal component analysis is discussed. The main differences between standard PCA and kernel PCA is that standard PCA is an orthogonal linear transformation and KPCA is a non-linear transformation. This will be discussed in more detail in section 4.4 and section 4.5.

4.1 Introduction

The analytical procedure consisted of four steps, namely data acquisition, image processing, feature extraction and classification of the condition of the fish based on the features extracted. The four steps are discussed in more detail below.

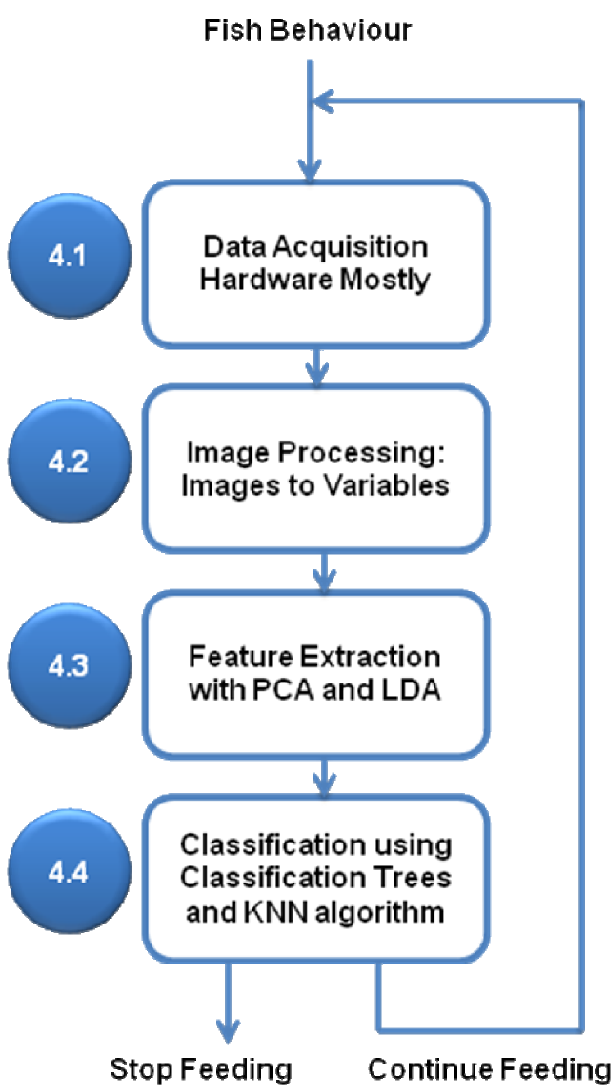


Figure 4-1: Methodology

4.2 Data Acquisition

Four 16 to 20 minute video clips were analyzed, each containing approximately 20 000 image frames. Each image frame consisted of a matrix of 480 x 720 pixels. Each of the video clips documented a prefeeding, feeding and postfeeding stage. The prefeeding stage was recorded for approximately 7-9 minutes. The feeding stage lasted approximately 2 minutes, while the postfeeding stage was also recorded for approximately 7-9 minutes.

4.3 Image Processing

A series of 1500 image frames with a resolution of 480 x 720 pixels was extracted for each of the three stages of feeding behaviour of the fish. Instead of analyzing the frames themselves, the differences between successive frames were considered, i.e. these differenced frames contained the change in pixel intensities over time (0.033 seconds).

Moreover, as indicated in Figure 4-2, the image frames were not processes in their entirety. Instead, a 240 x 200 zone was selected in each image to get rid of unwanted stationary scenery or image features not informative as far as the feeding behaviour of the fish was concerned. The rows in each of these zones were subsequently concatenated to yield a 1 x 48 000 row vector for each image. The aggregated data matrix **X** therefore had the dimensions 4500 x 48 000 for all three classes. It is this matrix that was decomposed by principal component analysis, as described in more detail in the next section.

Since these experiments were conducted in an indoor aquarium, typical image processing challenges such as changes in the brightness and contrast of image data were not a factor. Thus, these factors were not compensated for during image processing and may need to be considered under different experimental circumstances.

4.4 Feature Extraction with Principal Component Analysis

The data matrix **X** $\in \Re^{4500x48000}$ was decomposed by means of principal component analysis (PCA). PCA relies on an eigenvector decomposition of the covariance or correlation matrix of the process variables, i.e. for the data matrix **X** with n rows and m columns, **C**, the covariance matrix of **X** is defined as

$$C = \frac{1}{M}\sum_{j-1}^{M} x_j x_{j\ k}^T$$ Equation 4-1

Note that C is positive definite, and thus can be diagonalized with non-negative eigenvalues:

$$\boldsymbol{C}(\boldsymbol{X})\boldsymbol{p}_i$$ Equation 4-2

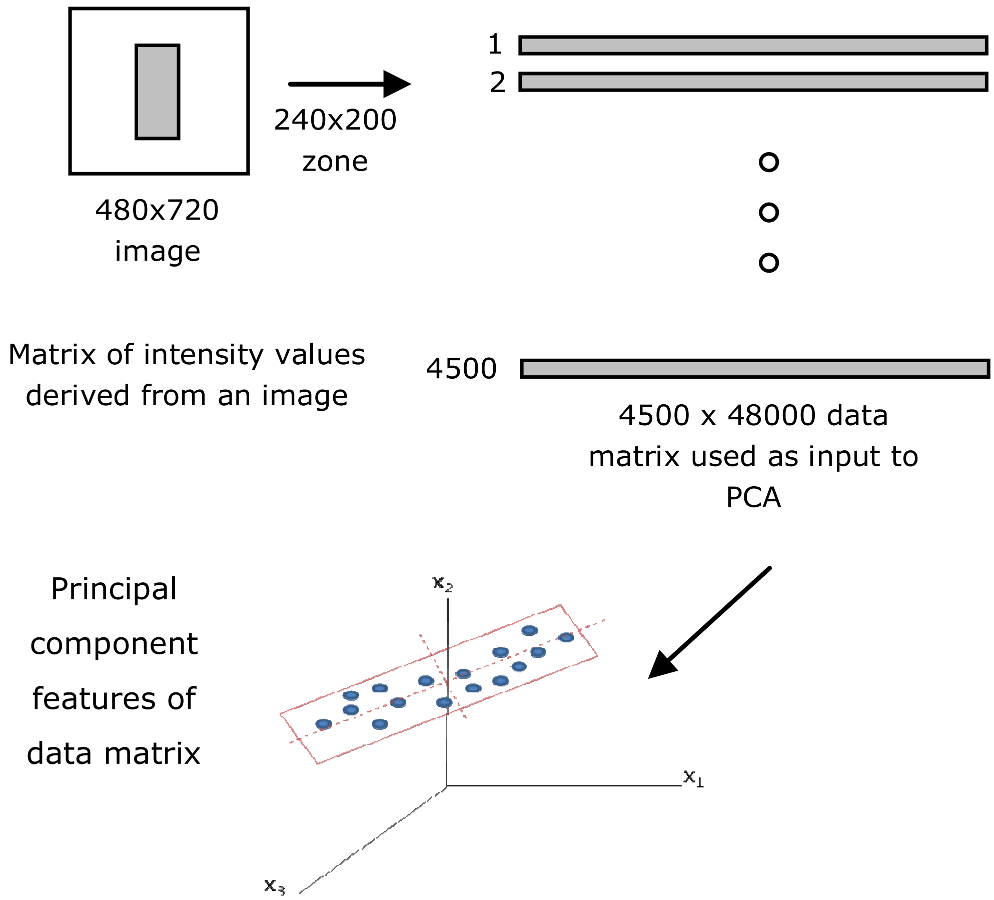


Figure 4-2: Construction of data matrix from digitized video images

4.4.1 Feature Extraction

Feature extraction has long been an important topic in pattern recognition and has been studied by many authors. Linear feature extraction can be viewed as finding a set of vectors which effectively represent the information content of an observation while reducing the dimensionality. In pattern recognition, it is desirable to extract features which are focused on discriminating between classes. Although a reduction in dimensionality is desirable, the error increase due to the reduction in dimensionality must be constrained to be adequately small. Finding the minimum number of feature vectors which represent observations with reduced dimensionality without sacrificing the discriminating the accuracy of the classifier is an important problem in field of pattern analysis.

In the case of this project, the features extracted are the scores computed by applying PCA. The scores is the representation of X in the principal component space. As mentioned in the previous section the scores correspond with the eigenvectors. The first eigenvector captures the most variability and the subsequent eigenvector the second most variability and so on. The first few scores is a good representation of approximately 50% – 80% of the variability in the data which should be adequately classified.

4.4.2 Principal Component Analysis

4.4.2.1 Background

Principal component analysis transforms a set of variables into a substantially smaller set of uncorrelated variables containing most of the information of the original set of variables. A small set of uncorrelated variables is easier to understand and work with than a large set of correlated variables (Aldrich, 2001).

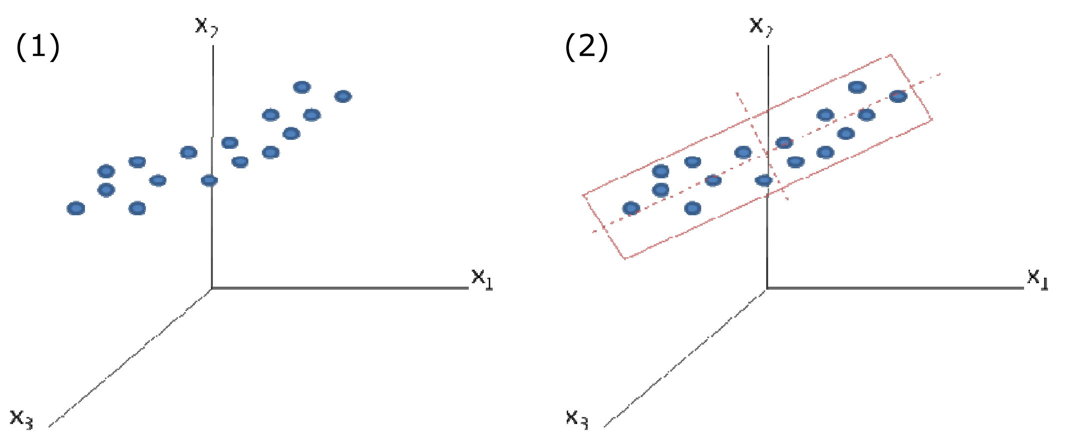


Figure 4-3: Geometric representation of principal component analysis (1) Data points in the observation space, (2) Plane defined by principal components.

Principal component analysis is a technique used to reduce multidimensional data sets to lower dimensions for analysis. The applications of PCA include exploratory data analysis data and generating predictive models. PCA can also be used to visualize multivariate data sets, so that outlying or atypical observations can be detected. PCA is mathematically defined as an orthogonal linear transformation that transforms the data to a new coordinate system such that the greatest variance by any projection of the data comes to lie on the first coordinate (called the first principal component), the second greatest variance on the second coordinate, and so on. An advantage of PCA is that once you have found these patterns in the data, and you compress the data, i.e. by reducing the number of dimensions, without much loss of information (Bishop, 2006).

There are two commonly used definitions of PCA that give rise to the same algorithm. PCA can be defined as the orthogonal projection of the data onto a lower dimensional linear space, known as the principal subspace, such that the variances of the projected data are maximized (Bishop, 2006). Equivalently, it can be defined as the linear projection that minimizes the average projection cost, defined as the mean squared distance between the data points and their projections (Bishop, 2006).

In the next section we will first review the mathematical perspective of the standard PCA algorithm.

4.4.2.2 Mathematical Perspective

Consider a set of data, with m variables and n observations, represented by the matrix $\boldsymbol{X} \in \mathbb{R}^{NxM}$.

Before we begin PCA, we need to centre the data so that it varies around zero. This is done by calculating the mean values of each of the variables and then subtracting these values from each measurement of a variable. $\boldsymbol{x_k} \in \mathbb{R}^N, \boldsymbol{k} = \mathbf{1}, \ldots, \boldsymbol{M},\ \sum_{j=1}^{M} \boldsymbol{x_k} = \mathbf{0}.$

PCA relies on an eigenvector decomposition of the covariance or correlation matrix of the process variables. In this work we will use the convention that rows of a data matrix X correspond to samples while columns correspond to variables.

For a given data matrix X with m rows and n columns the covariance matrix of X is defined as

$$C = \frac{1}{M} \sum_{j=1}^{M} x_j x_{j\ k}^T$$

Equation

4-1

$\boldsymbol{x_j} \in \mathbb{R}^N, \boldsymbol{j} = 1, \ldots, \boldsymbol{M}, \sum_{j-1}^{M} \boldsymbol{x_j} = \mathbf{0}$, Note that C is positive definite, and thus

can be diagonalised with non-negative eigenvalues:

$$\boldsymbol{C}(\boldsymbol{X})\boldsymbol{p}_i$$

Equation 4-2

For eigenvalue λ≥0 and eigenvectors $p \in \mathbb{R}^N\{0\}$, as

$$\boldsymbol{\lambda_i p_i} = \boldsymbol{C}(\boldsymbol{X})\boldsymbol{p_i} = \frac{1}{\mathrm{M}} \sum_{\mathrm{j}=1}^{\mathrm{M}} (\mathbf{x_j}\ \mathbf{p})\mathbf{x_j}$$ Equation 4-3

It can be shown that a line a space can be represented by a vector **p** of unit length, which means that per definition $\mathbf{p}^{\mathbf{T}}\mathbf{p} = \sum_{i=1}^{m} \mathbf{p}_i^2 = 1$. It is thus algebraically possible to define the principal components to be the linear combination

$$\boldsymbol{X}\boldsymbol{p_i} = \boldsymbol{t_i}$$ Equation 4-4

of the original variables that maximizes the variability which is usually affected by means of Lagrange multipliers (not dealt with here).

This is the same as saying that principal component analysis decomposes the data matrix X as the sum of the outer product of vectors t_i and p_i plus a residual matrix E:

$$\boldsymbol{X} = \boldsymbol{t_1}\boldsymbol{P_1^T} + \boldsymbol{t_2}\boldsymbol{P_2^T} + \cdots + \boldsymbol{t_k}\boldsymbol{P_k^T} + \boldsymbol{E}$$ Equation 4-5

Here k must be less than or equal to the smaller dimension of X, i.e. k ≤ {m,n}. The $\mathbf{t}_i$ vectors are known as scores and contain information on how the samples relate to each other. The p_i vectors are eigenvectors of the covariance matrix, i.e. λ_i is the eigenvalue associated with the eigenvector $\mathbf{p}_i$. In PCA the $\mathbf{p}_i$ are known as loadings and contain information on how variables relate to each other. The $\mathbf{t}_i$ form an orthogonal set ($\mathbf{t}_i^T\mathbf{t}_j = 0$ for $i \neq j$), while the $\mathbf{p}_i$ are orthonormal ($\mathbf{p}_i^T\mathbf{p}_j = 0$ for $i \neq j$, $\mathbf{p_i}^\mathbf{T}\mathbf{p_j} = 1$ for $i = j$).

Note that because the score vector $\mathbf{t}_i$ is the linear combination of the original data defined by $\mathbf{p}_i$., the $\mathbf{t}_i$, $\mathbf{p}_i$ pairs are arranged in descending order according to the associated λ_i. The λ_i is a measure of the amount of variance described by the $\mathbf{t}_i$, $\mathbf{p}_i$ pair. In this context, we can think of variance as information. Because the $\mathbf{t}_i$, $\mathbf{p}_i$ pairs are in descending order of λ_i, the first pair captures the largest amount of information of any pair in the decomposition. In fact, it can be shown that the first $\mathbf{t}_i$, $\mathbf{p}_i$ pair captures the greatest amount of variation in the data that it is possible to capture with a linear factor. Each subsequent pair captures the greatest possible amount of variance remaining at the step (Wise and Gallagher, 1996).

To summarize, the j'th column of *principal component scores* is the linear combination $\mathbf{t}_j=\mathbf{X}\mathbf{p}_j$, which has the greatest sample variance for all $\mathbf{p}_j$ satisfying $\mathbf{p}_j^T\mathbf{p}_j = 1$ and $\mathbf{p}_j^T\mathbf{p}_i = 0$ $(i < j)$. The *principal component coefficients* $\mathbf{p}_j^T$ are given by the elements of the *eigenvector* corresponding to the j'th largest eigenvalue λ_j (the variance of $\mathbf{t}_j$) of **C,** (the sample covariance matrix of the original data matrix **X**). The *loading* of the k'th original variable $\mathbf{x}_k$ on the j'th principal component $\mathbf{p}_j$ is defined by $\mathbf{p}_{jk}(\lambda_j)^{1/2}$, where $\mathbf{p}_j^T = (p_{j1}, p_{j2}, .., p_{jm})$. The *score* of the i'th individual or sample point on the j'th principal component is defined as $t_{ij} = \mathbf{p}_j^T\mathbf{x}_i = p_{j1}x_{i1} + p_{j2}x_{i2} + \ldots + p_{jm}x_{im}$ $(j = 1, 2, \ldots m)$.

4.5 Linear Discriminant Analysis

In order to view the data projected on a different plane we can apply linear discriminant analysis. Linear discriminant analysis or LDA is a popular linear method for dimensionality reduction and it achieves this by minimizing class scatter, a measure of the variance within each class, and maximizing the class distances so as to improve class separation. In the supervised learning context it is possible to make use of the class information, contained in target vector ***t***, to obtain an optimal mapping into a lower dimensional space to simplify the classification task and to gain insight into class separability. PCA was used to embed the image data and then LDA was used to extract discriminating features from the embedded data. This was followed by classification of the features. The third party software used to perform LDA was an updated version of Gist® which was updated by Jemwa (2007). The Matlab® interface program can be found in the Appendix A.

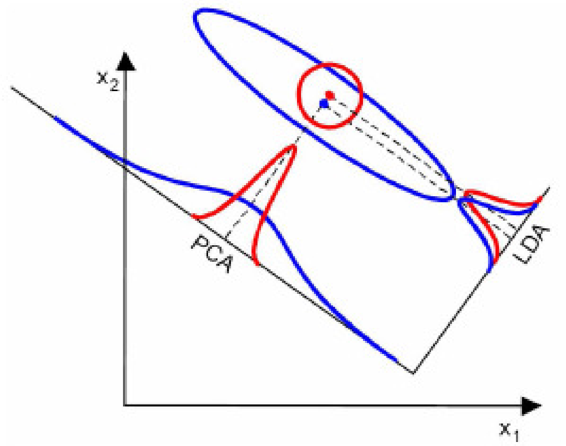


Figure 4-4: Illustration of difference between LDA and PCA

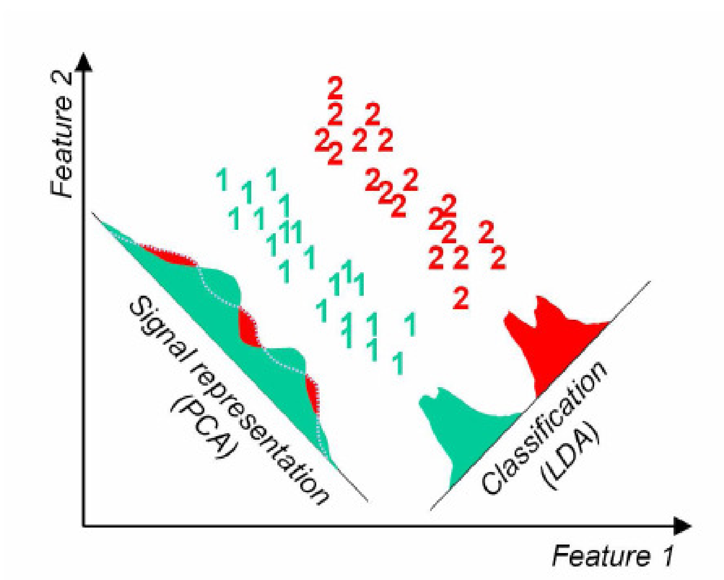


Figure 4-5: Illustration of differences between PCA and LDA

4.6 Classification

The features extracted from the image data are used as predictors for classification models designed to identify the feeding behaviour of the fish. There are a large number of models that can be used for this purpose, but in this thesis only two methods were considered, namely a classification tree and a nearest neighbour method. Both are nonlinear approaches. The classification tree was used to classify the behaviour of the fish based on features extracted with PCA, while the nearest neighbour method was used to classify the features extracted from applying LDA. The purpose of this investigation was not necessarily to find the optimal classification model, but to assess the feasibility of the features to classify fishing behaviour. A brief description of each classifier follows below.

4.6.1 Classification Tree

There are a variety of straightforward, but widely used, models that work by partitioning the input space into cubical regions, whose edges are aligned with the axes, and then conveying a straightforward model (for example, a constant) to each region. They can be observed as a model combination method in which only one model is responsible for making predictions at any given point in input space. The process of selecting a specific model, given a new input x, can be described by a sequential decision making process corresponding to the traversal of a binary tree i.e. one that splits into two branches at each node (Bishop, 2006).

The inductive classification model is a form of empirical learning through which general conclusions are inferred from specific examples. In particular, the training algorithms form generalized rules linking features, or independent variables, of a set of exemplars to the predefined classes or dependent variable of each of the exemplars. This decision tree is correspondent to a set of IF-THEN rules, which can be interpreted by the data analyst, or can be incorporated into an expert system shell to provide as a knowledge-based decision support system (Aldrich, 2001).

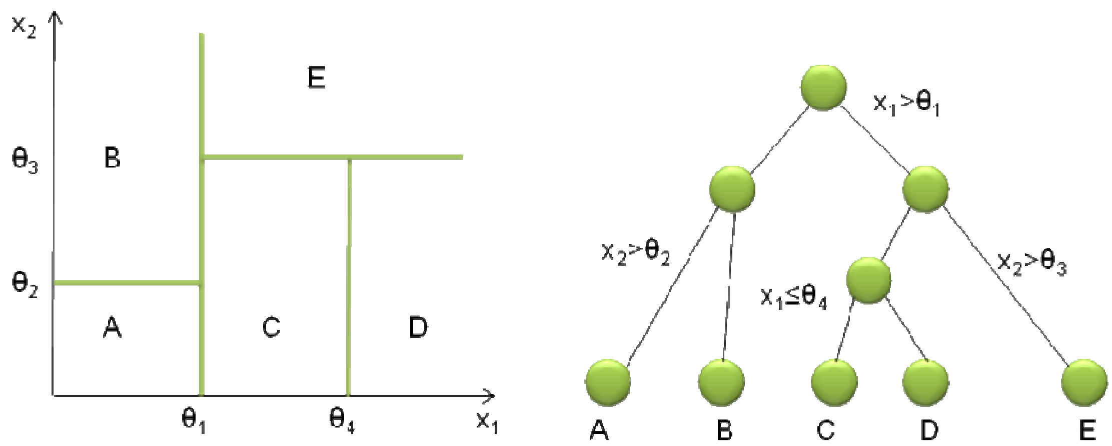


Figure 4-6: Illustration of a two-dimensional input space that has been partitioned into five regions using axis-aligned boundaries, next to it a binary tree corresponding to the partitioning of input space shown (Bishop, 2006).

The process starts with a training set consisting of pre-classified records. Having pre-classified records means that the target field or dependent variable has a known class or label. The goal is to build a tree that distinguishes among the classes. For simplicity, assume that there are only two target classes and that each split is binary partitioning. The splitting criterion easily generalizes to multiple classes, and any multi-way partitioning can be achieved through repeated binary splits. To choose the best splitter at a node, the algorithm considers each input field in turn. In essence, each field is sorted. Then, every possible split is tried and considered, and the best split is the one which produces the largest increase in homogeneity of the classification label within each partition. This is repeated for all fields, and the winner is chosen as the best splitter for that node. The process is continued at the next node and, in this manner, a full tree is generated.

Pruning is the process of removing leaves and branches to improve the performance of the decision tree when it moves from the training data (where the classification is known) to real-world applications (where the classification is unknown and it is what you are trying to predict). The tree-building algorithm makes the best split at the root node where there are the largest number of records and, hence, a lot of information. Each subsequent split has a smaller and less representative population with which to work. Towards the end, peculiarity of training records at a particular node display patterns that are peculiar only to those records. These patterns can become meaningless and sometimes harmful for prediction if you try to extend rules based on them to larger populations.

Pruning methods solve this problem, they let the tree grow to maximum size, and then remove smaller branches that fail to generalize.

Since the tree is grown from the training data set, when it has reached full structure it usually suffers from over-fitting (i.e. it is "explaining" random elements of the training data that are not likely to be features of the larger population of data). This results in poor performance on real life data. Therefore, it has to be pruned using the validation data set. There are numerous methods to validate the performance of the classifier; the methods used in this project are explained in the next section.

4.6.2 Classification Performance Evaluation

Unbiased evaluation of classification methods is important. The goal of classification tree analysis, simply stated, is to obtain the most accurate prediction possible, the strategies used to select the right-sized tree is as follows:

4.6.2.1 Cross-validation

Cross-validation, sometimes called rotation estimation (Ron, 1995), is the statistical practice of partitioning a sample of data into subsets such that the analysis is initially performed on a single subset, while the other subset(s) are retained for subsequent use in confirming and validating the initial analysis. The initial subsets of data are called the *training set*; the other subset(s) are called *validation* or *testing sets*.

The theory of cross-validation was inaugurated by Seymour Geisser. It is important in guarding against testing hypotheses suggested by the data, especially where further samples are hazardous, costly or impossible to collect.

Validation techniques are motivated by two fundamental problems in pattern recognition: model selection and performance estimation.The method applied to cross-validating in this project is called the leave one out cross-validation approach. As the name suggests, leave-one-out cross-validation (LOOCV) involves using a single observation from the original sample as the validation data, and the remaining observations as the training data. This is repeated such that each observation in the sample is used once as the validation data. This can be illustrated as follows:

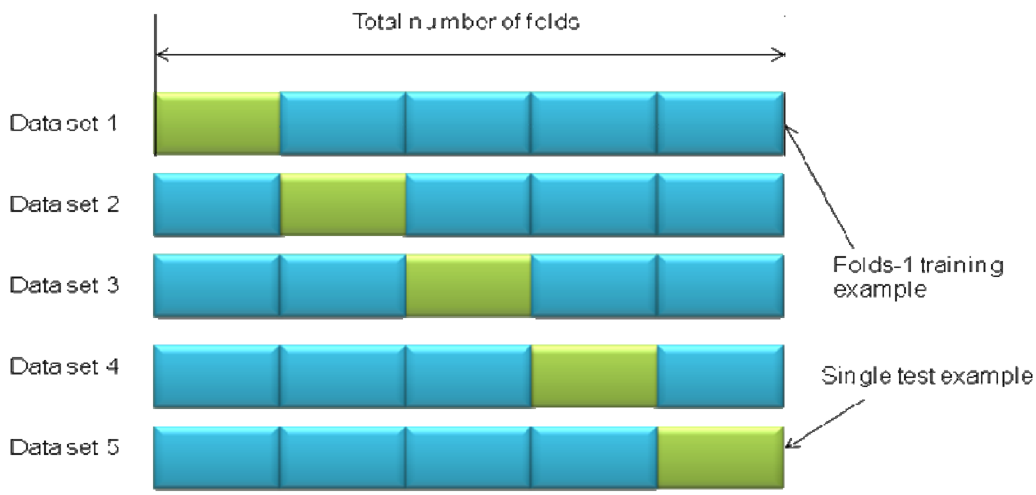


Figure 4-7: An illustration of leave one out cross validation with 5 folds

The true error is estimated as the average error rate on the test examples

$$E = \frac{1}{N}\sum_{i=1}^{N} E_i \qquad \text{Equation 4-6}$$

4.6.2.2 Random sub-sampling

Each split randomly selects a fixed number of examples without replacement, for each data split we retrain the classifier from scratch with the training examples and estimate E with the test samples.

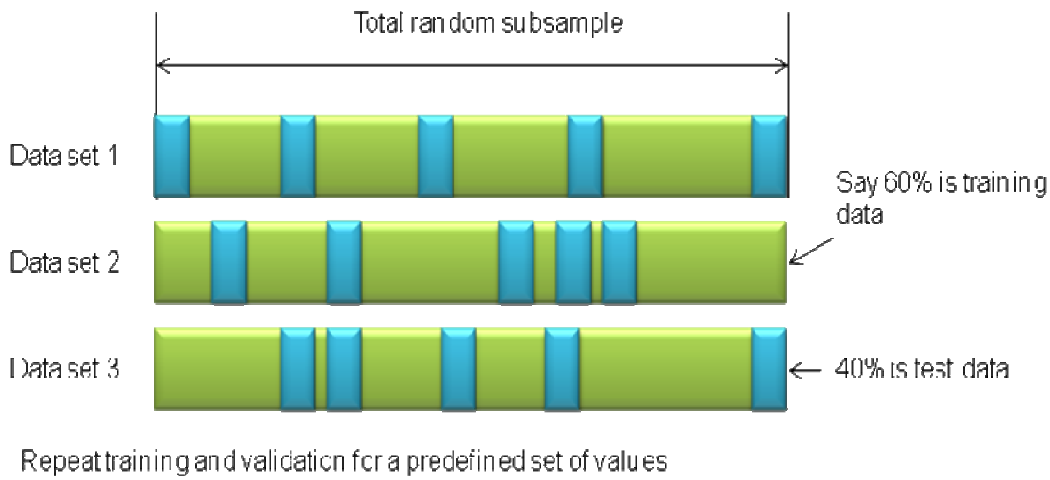


Figure 4-8: An illustration of random sub-sampling

The true error is estimated as in equation 4-8.

4.6.3 K nearest neighbour classification

The *k*-nearest neighbour (KNN) is amongst the simplest of all machine learning algorithms. The classifier will be used as an alternative model to assess the ability of the features to discriminate between various feeding behaviours of the fish.

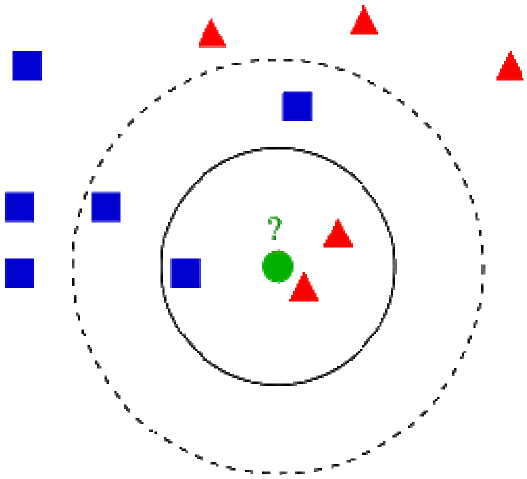


Figure 4-9: The KNN classification. The test sample (green circle) should be classified either to the first class of blue squares or to the second class of red triangles. If *k = 3* it is classified to the second class because there are 2 triangles and only 1 square inside the inner circle. If *k = 5* it is classified to first class (3 squares vs. 2 triangles inside the outer circle) (Dasarathy, 1991).

A new sample is classified by calculating the distance to the nearest training case; the sign of that point then determines the classification of the sample. The k-NN classifier extends this idea by taking the k nearest points and assigning the sign of the majority. It is common to select k small and odd to break ties (typically 1, 3 or 5). Larger k values help reduce the effects of noisy points within the training data set, and the choice of k is often performed through cross-validation (Robotics Research Group, 2007). This is illustrated in Figure 4-9:

4.7 Software Tool

Gist® is a convenient software tool which is available for free on the internet. It is convenient, because the programming code is already written to carry out PCA or kernel based PCA. Gist® is a software tool for support vector machine classification and for kernel principal component analysis, but only the standard features were used. An important consideration in the use of Gist® was the fact that it could deal with relatively large sets that would otherwise have caused problems in Matlab®, owing to memory limitations.

This program has the option of selecting “standard” PCA. Firstly, ‘gist-kpca.exe’ is run in Matlab in order to train the PCA model (feeding behaviour), and secondly, ‘gist-project.exe’ is used to project the data. Thus, projecting the scores of the trained PCA model (feeding behaviour) and projecting the test data onto the PCA model (pre and postfeeding behaviour). More detail on the software is provided in the Appendix.

Chapter 5

Results and Discussion

5.1 Preliminary Analysis of Fish Feeding Behaviour

As a first basic approach to see if significant differences exist in the three classes observed, the following method was followed as illustrated below:

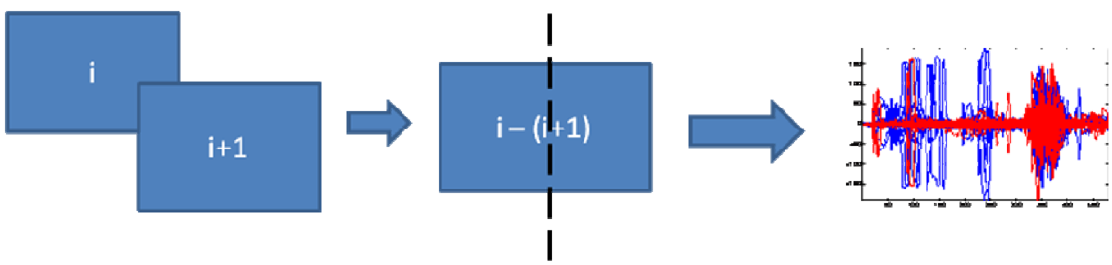


Figure 5-1: First Basic Approach to Detect Differences in Three Classes

The image frames are read into Matlab® where the each subsequent (i+1) image frame is subtracted from the previous frame (i) and the differences between these image are then displayed in figures 5.2-5.4. As shown in figure 5.1 the changes across the midpoint column of each frame is detected using this method. This method is a very simple way to detect if significant differences exist in these classes, this is not a very efficient method and therefore improved techniques had to be applied. From this straightforward first technique applied the changes detected in the different classes were significantly different for post-feeding behaviour than for pre-feeding and feeding behaviour. This indicates that predicaments may arise when classifying pre-feeding from feeding behaviour and post-feeding behaviour would seemingly be more effortlessly classified from the other two classes. At this early stage of data analysing it would seem better two rather have two classes than three classes of feeding behaviour due to the complexity of categorizing prefeeding from feeding behaviour.

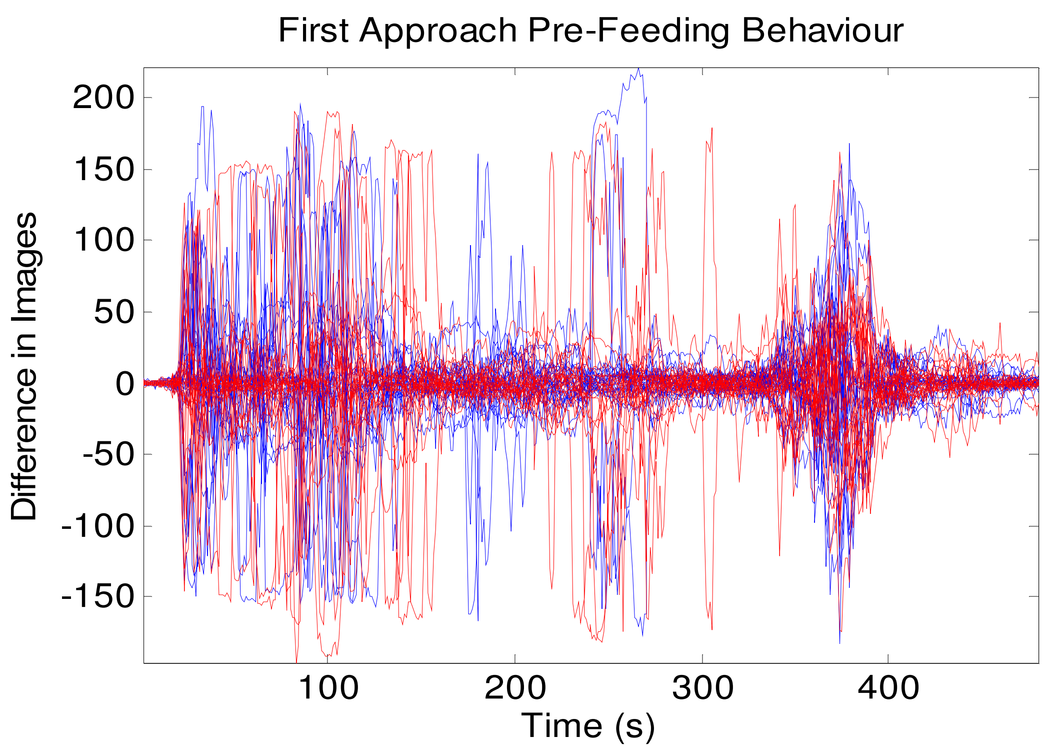


Figure 5-2: Prefeeding Behaviour

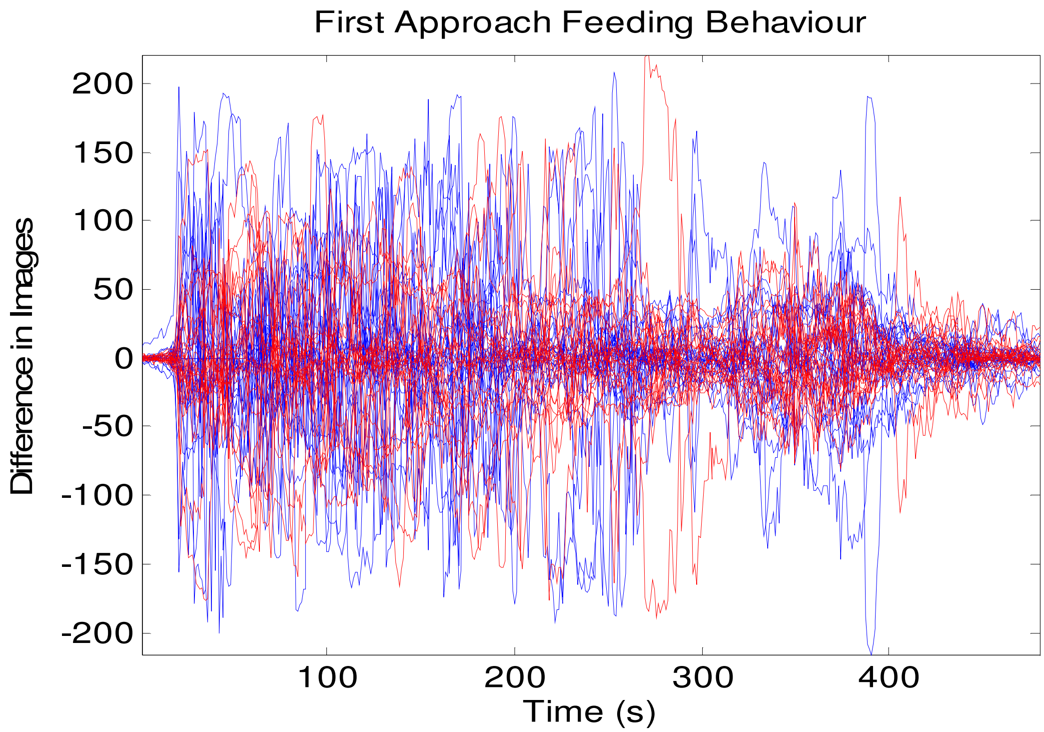


Figure 5-3: Feeding Behaviour

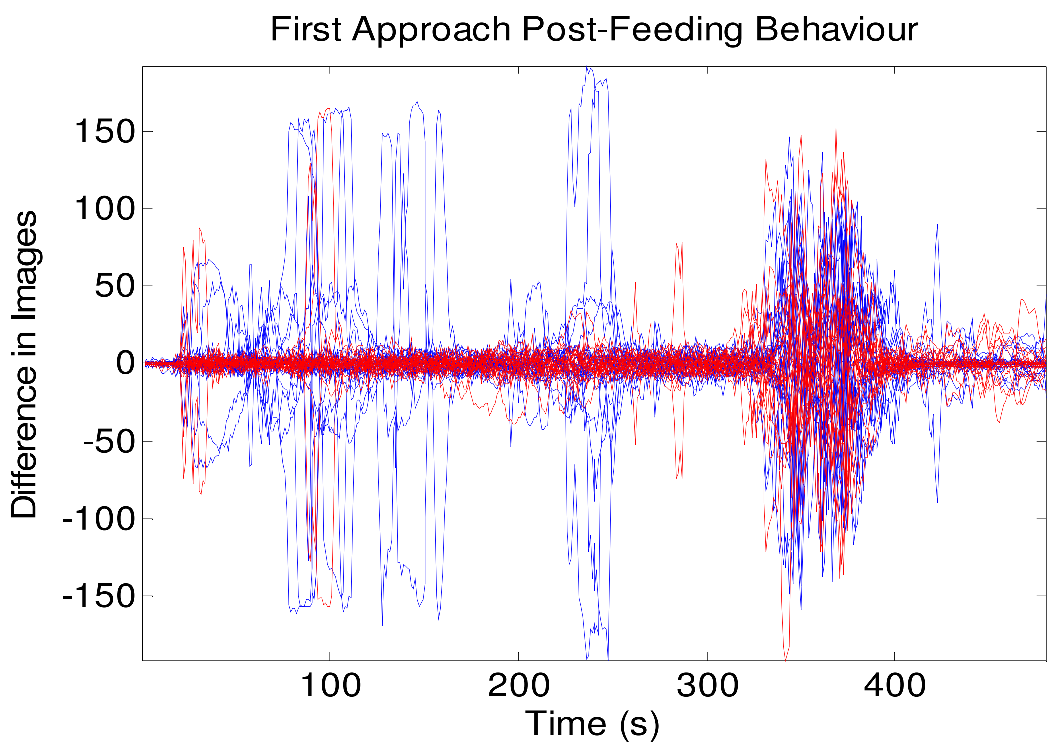


Figure 5-4: Post-feeding Behaviour

5.2 Principal Component Analysis

Video data of fish feeding were processed in Matlab® as a series of 240x250 matrices representing the pixels of the greyscale image derived from a 420x720 truecolour image. The reference data consisted of integrating 1500 image frames from each class, i.e. 4500 in total.

To compress the data, we can then choose to transform the data only using say 200 variables, which is equal to 92% of the total variance. In Figure 5-5 the bars represent the variance and the line represents the cumulative variance. As shown in the scree plot (Figure 5-5) the total variance of the first ten scores is equal to approximately 40%. If the original data are reproduced, the images would have lost some of the information. However, in this project it is not necessary for us to reproduce the data, but only to see how well the data can be classified.

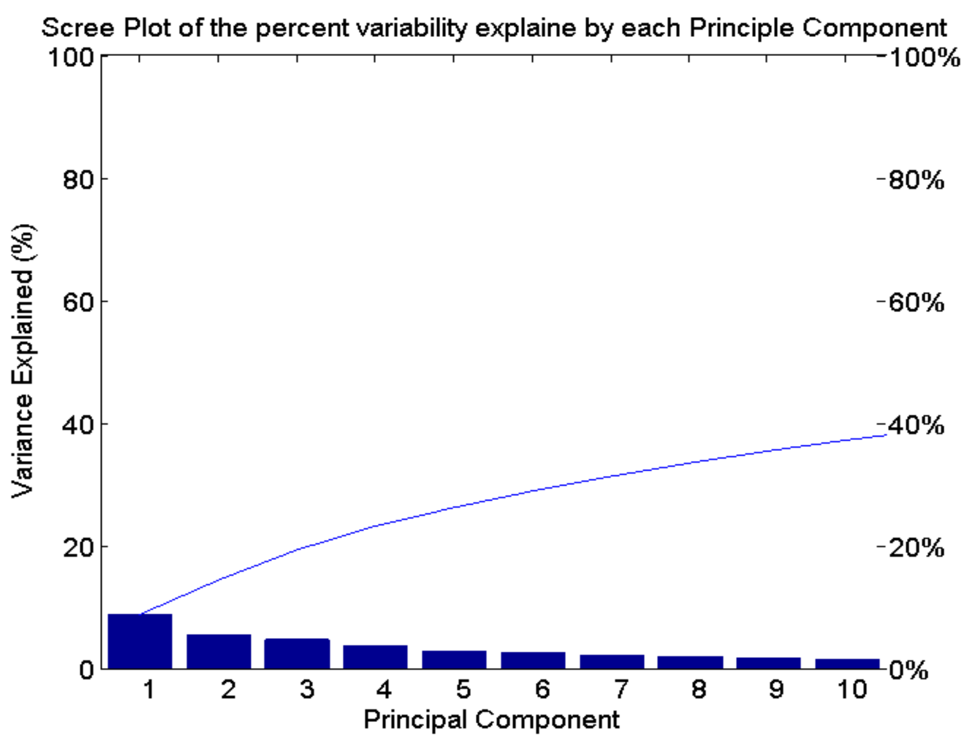


Figure 5-5: Scree plot for eigenvalues derived from performing PCA on movie 1

One of the difficulties inherent in multivariate statistics is the problem of visualising the data that has many variables. To visualise the principal components we can plot the relationship between the first two principal components, as well as the first three components, retaining most of the variability in the data.

The feeding class is indicated by the blue markers, prefeeding class by red markers and postfeeding by green markers. As explained in chapter 4.4, the feeding behaviour is trained to generate the PCA model. Prefeeding and postfeeding behaviour scores are then subsequently projected onto the PCA model. It can be seen in the first two and three principle component dimensions (Figure 5-6 and Figure 5-6) that there is overlapping between the features of prefeeding, feeding and postfeeding behaviour. However, feeding and non-feeding behaviour could be classified with an overall accuracy of 99% using the classification tree model.

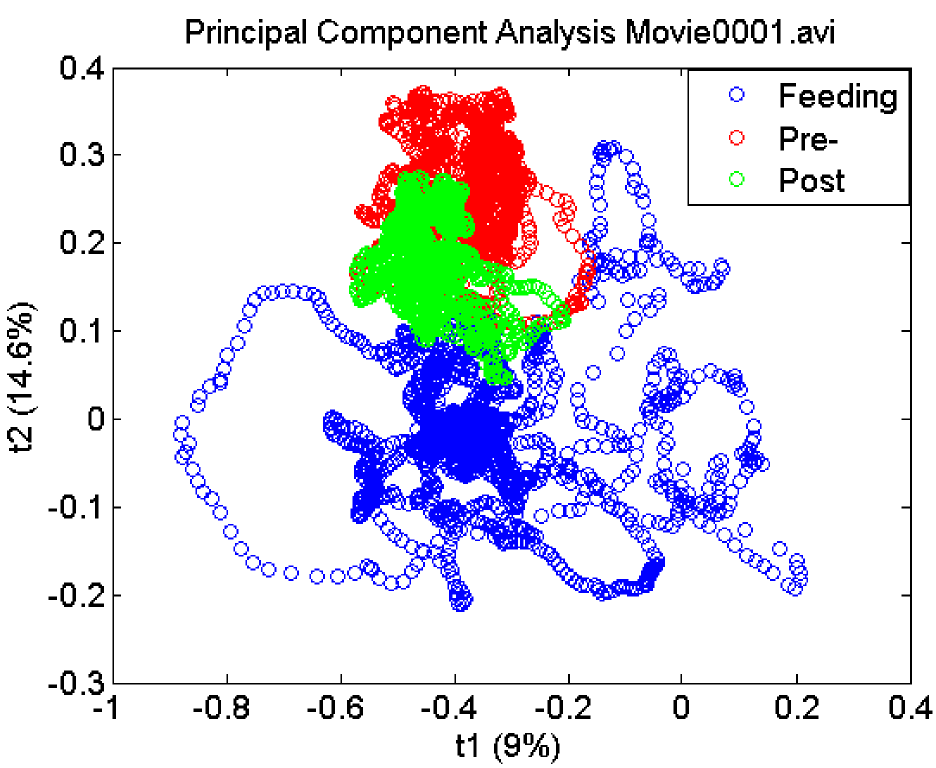


Figure 5-6: First two principal component scores (Movie0001.avi)

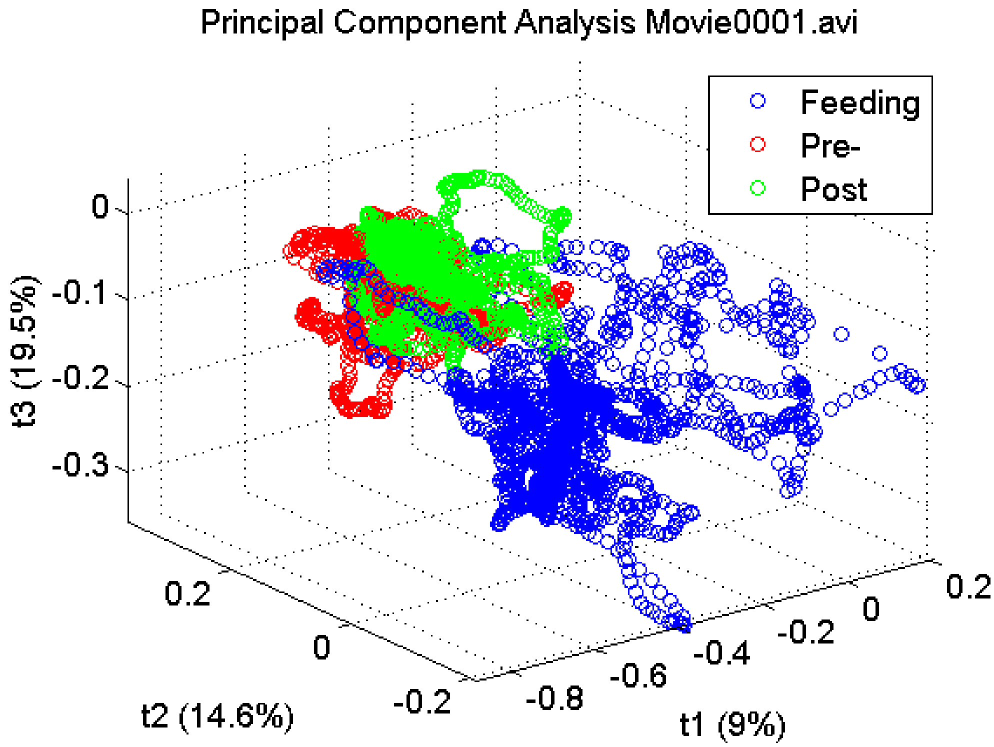


Figure 5-7: First three principal component scores (Movie0001.avi)

5.3 Linear Discriminant Analysis

We can also view the data on a different plane using linear discriminant analysis as explained in chapter 4.5 which is illustrated in the figure 5.9 below. LDA was performed also using the third party software as in the case of PCA, however, this software program was written by Jemwa (2007), which is an updated version of Gist®. The program is run using Matlab as in the case with PCA. Firstly a LDA model is trained used all three classes, of 1000 frames each, in order to train the model to discriminate between the three classes, this is illustrated in figure 5.8. The motivation for using only 1000 frames from each class was due to limitations from process performance of the computer. Secondly a second set of independent image frames are then projected onto the model so as to test the performance of the model, which is shown in figure 5.9.

It can clearly be seen in figure 5.8 that significant separation occurs between all three groups. However, the true performance of this projection can be substantiated using the test data as shown in figure 5.9.

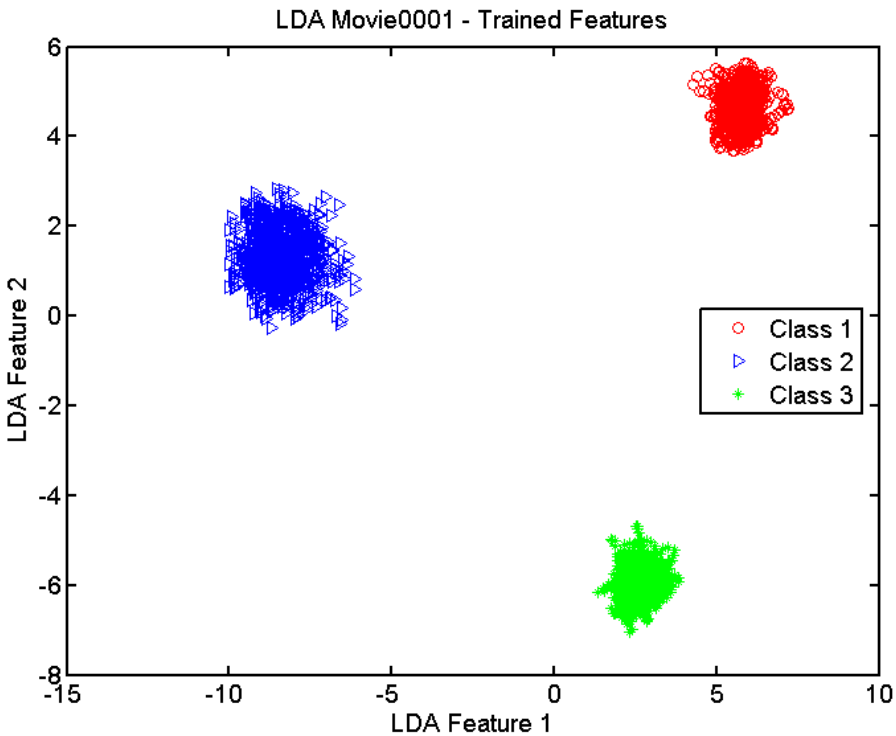


Figure 5-8: Component Scores using LDA (Trained Features)

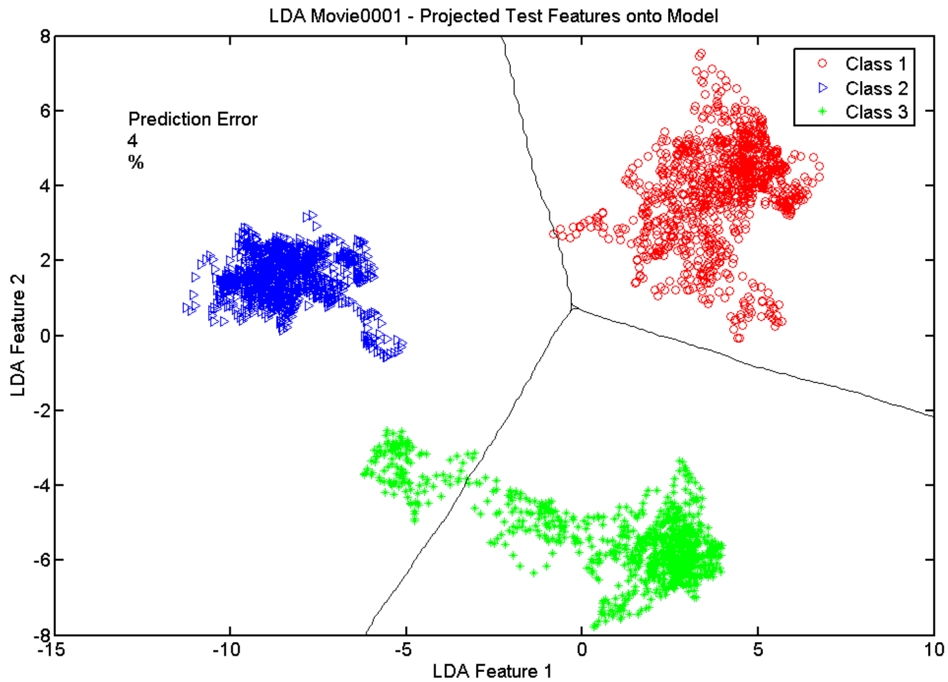


Figure 5-9: Linear Discriminant Analysis Movie0001 (Projected Test Features)

In figure 5.9, class 1 represents pre-feeding behaviour, class 2 represents feeding behaviour and class 3 represents post-feeding behaviour. In figure 5.9 it can clearly be seen that the test features are more scattered than with the trained features shown in figure 5.8. In comparison with the PCA figure 5.6, it can be established that the LDA shows better class separation than with PCA. The objective of PCA is to perform dimensionality reduction while preserving as much of the randomness (variance) in the high dimensionality space as possible. And the objective of LDA is to perform dimensionality reduction while preserving as much of the class discriminatory information as possible.

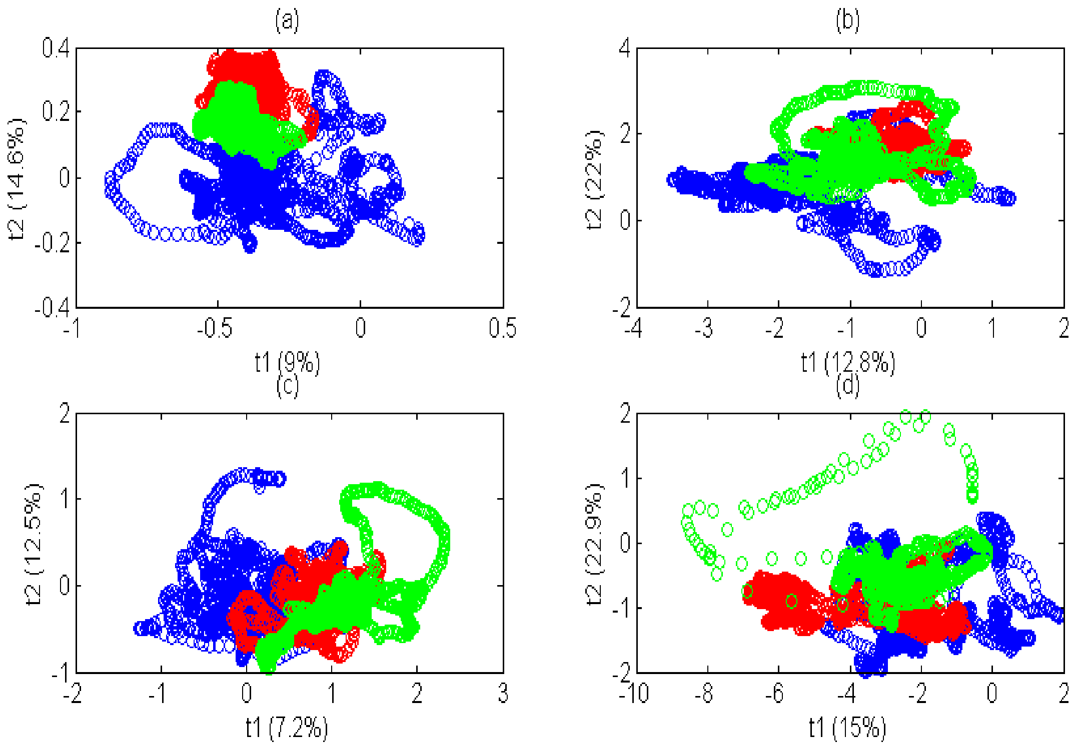


Figure 5-10: Component scores plots using PCA for all four movies (a) movie00011 (b) movie0002 (c) movie0004 (d) movie0005

The discriminatory quality of the LDA features was tested using a k nearest neighbour classifier. Quantitatively the three classes were classified with an overall accuracy of 96%. The results for PCA and LDA for all the video data that were analysed are shown in figure 5.10 and 5.11. Once again, it can clearly be seen that the features pre-feeding and post-feeding do not allow sharp discrimination in the two-dimensional plot, as the overlap between the groups are significant. Quantitatively the three classes were classified using a classification tree with an overall accuracy of 99%, 98%, 96% and 98% respectively.

This however will be discussed in more detail later on in this chapter. Any separation can thus be more clearly seen in a large three-dimensional plot. All other two dimensional and three dimensional plots of the data can be found in the Appendix B.

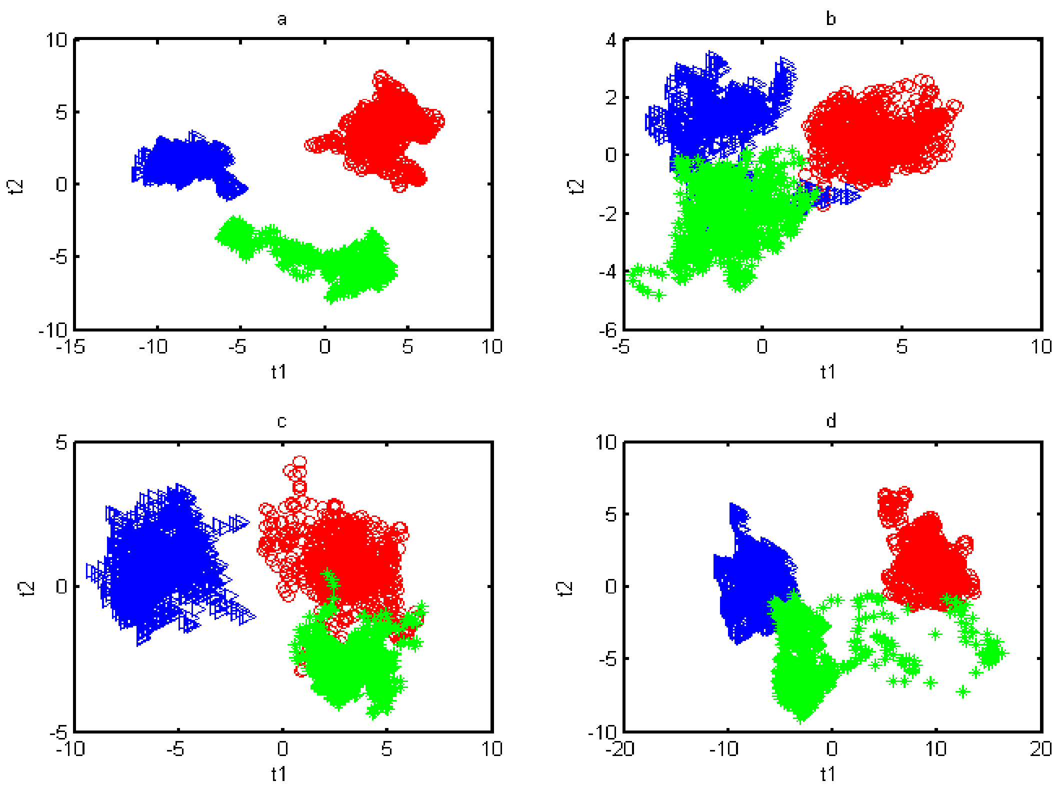


Figure 5-11: Component scores plots using LDA for all four movies (a) movie00011 (b) movie0002 (c) movie0004 (d) movie0005

5.4 Classification

It can clearly be seen in the first two principal component dimension (figure 5.11 (b), (c) and (d)) there is overlapping between the features, prefeeding, feeding and postfeeding behaviour. However, the three classes were classified with an overall accuracy of 96%, 89%, 95% and 94% respectively. The figures containing the plot of the KNN model is shown in the Appendix C. The within-class separation can be more clearly visualised using LDA.

The data that represent the relationships or processes to be modelled are typically divided into representative training and test data sets. Some prefer to divide the data into a training set and two data test sets. The first set is used repeatedly during training, i.e. for cross-validation of the performance of the model, while the second set of test data are only used once the model has been developed, to assess the accuracy of the model. In this classification, as a first approach, a training set containing 50% of randomized features extracted from all 3 classes was used to train the classification tree; the other 50% was used as the test data set. An illustration of the decision tree is shown below for movie0001, the classification trees for the movie002, movie0004 and movie0005 can be found in the Appendix D.

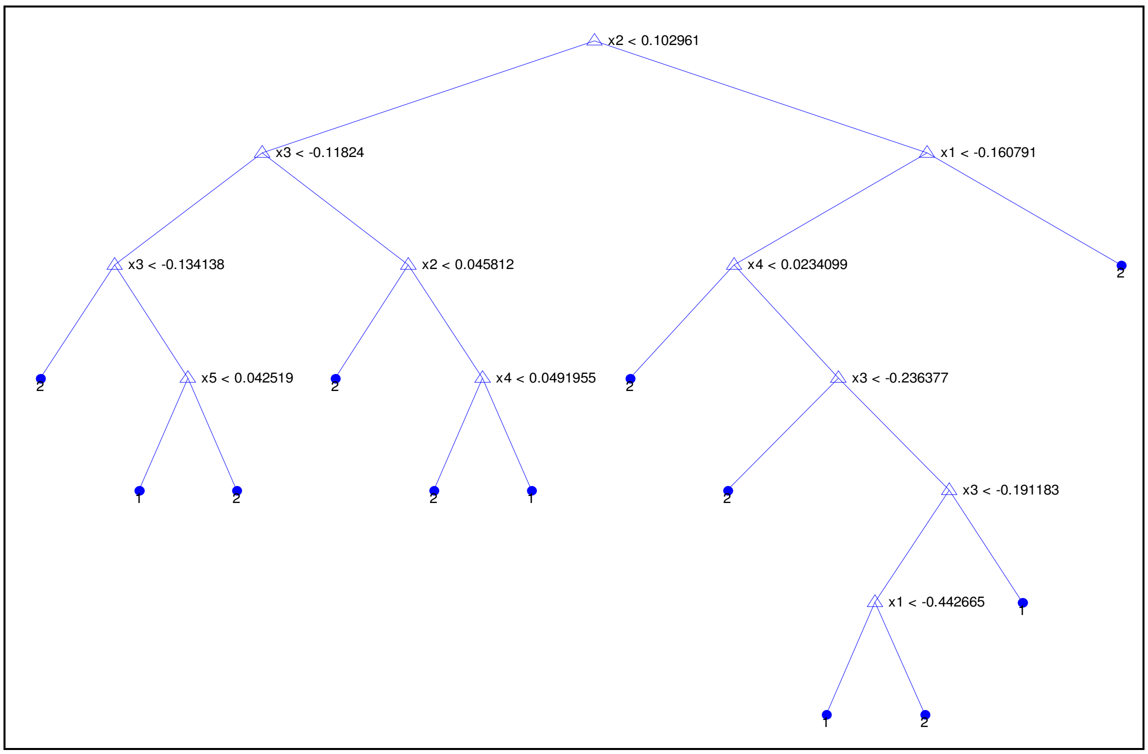


Figure 5-12: Classification Tree Movie0001.avi

The classifier distinguishes between two classes, feeding (features from PCA model) and not feeding (features projected onto the PCA model). A data set is made up of 1500 features from each class which is randomized.

The training set error rate can be highly misleading and is usually an overoptimistic estimate of performance. Inaccuracies are due to the over-fitting of a learning system to the data. When multiple random test and train experiments are performed, a new classifier is learned from each training sample. The estimated error rate is the average of the error rates for the classifiers derived for the independently and randomly generated test partitions.

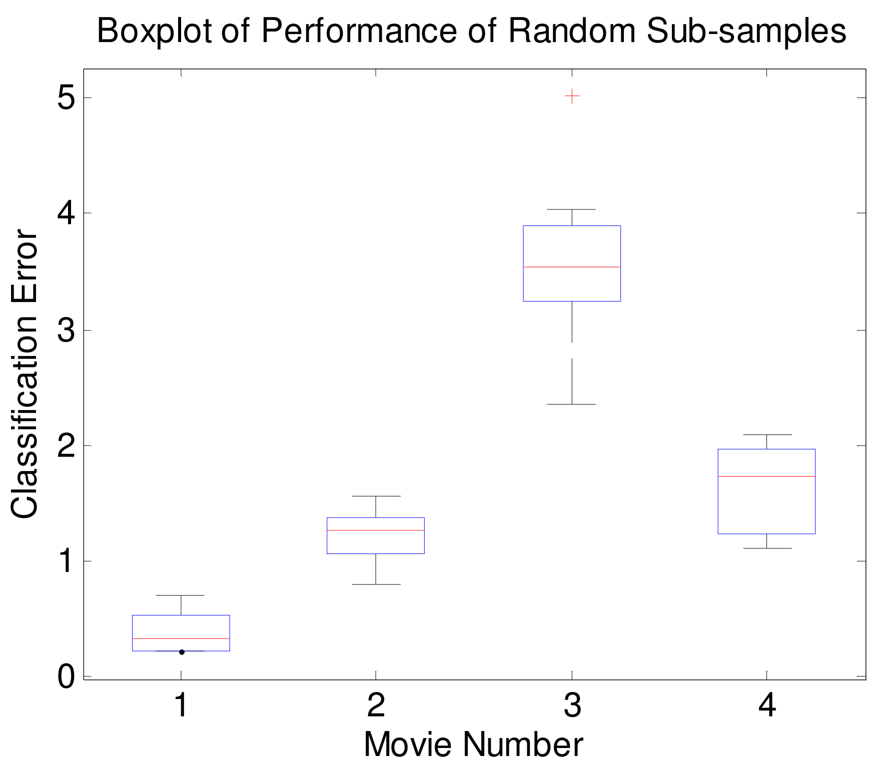


Figure 5-13: Boxplot of the performance of different movies. Each movie was randomly sampled ten times.

Each movie was randomly tested and trained 10 times; the extracted features from movie0001 were classified the most accurately with a classification error of 0.39%. The extracted features from Movie0004 were classified the least accurate with a classification error of 3.6%.

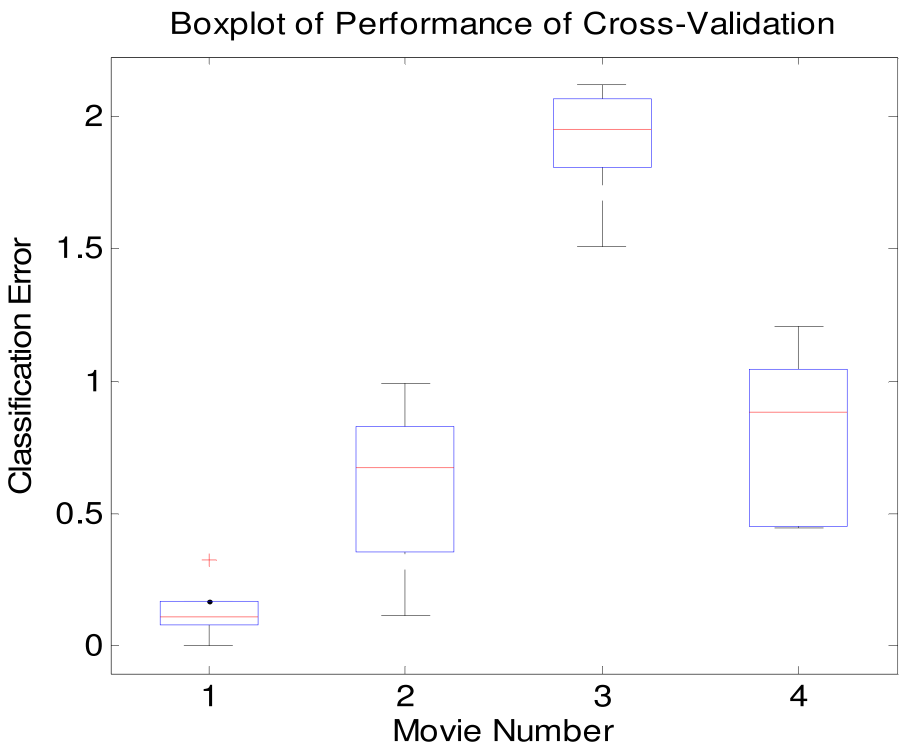


Figure 5-14: Cross-Validation of the performance of different movies. Each movie was validation 5-Fold

A 5-fold cross-validation was performed on each classification of the movies. The features from Movie0001 was classified the most accurately with a classification error of approximately 0.20% and the features extracted from movie0004 were classified least accurately which corresponds well with the validation of the random sub-sampling method. These are two of the simplest techniques for "honestly" estimating error rates, and both validation tests shows similarity which is an indication of a good classifier.

Classification of Video Data for all 4 Videos:

Better representations of the data are to combine all four video's data and to see how well they classify using both methods namely random sub-sampling and cross-validation.

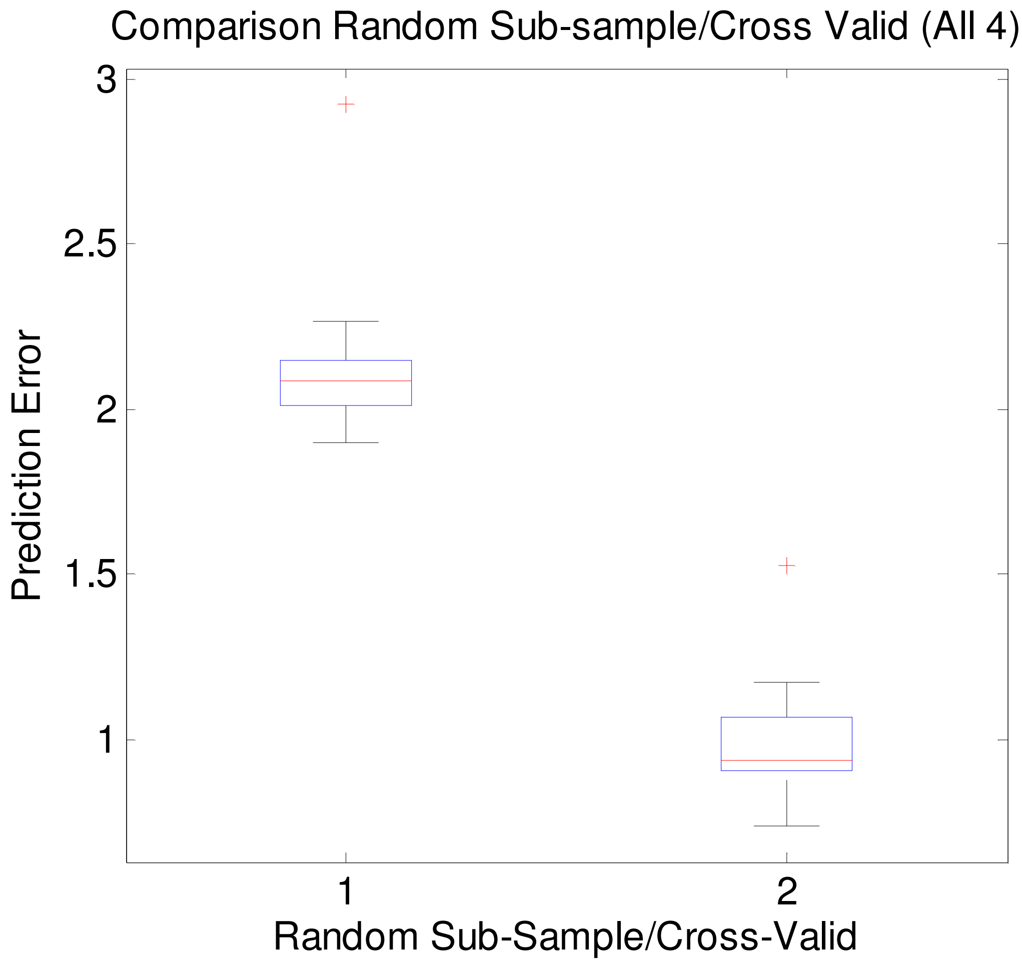


Figure 5-15: Boxplot for Random Sub-samples and Cross Validation of all four videos

The video data is firstly be normalised in order to combine the video data from four different videos. The features for all four videos combined were classified with an overall accuracy 2% and 1% respectively.

5.5 Concluding Remarks

The two and three dimension principal component plots for all four videos showed some overlapping between the features extracted from prefeeding, feeding and postfeeding behaviour. The features in movie0001 allow for most reliable discrimination between feeding and non-feeding conditions. Quantitatively, the two groups could be classified with an overall accuracy of 99%. movie0002, movie0004 and movie0005 could be classified with overall accuracies of 98%, 96%and 98% respectively. The differences in the overall classification accuracies of the different movies are probably not statistically significant.

The features extracted using LDA showed better separation between the three classes namely, prefeeding, feeding and postfeeding behaviour. The performance of the LDA model was tested using the k nearest neighbour classifier. Quantitatively, the features extracted from movie001 were classified with an overall accuracy of 96%. Movie0002, movie0004 and movie0005 could be classified with overall accuracies of 89%, 95% and 94% respectively.

In comparison with the PCA figure 5.6, it can be established that the LDA shows better class separation than with PCA. The objective of PCA is to perform dimensionality reduction while preserving as much of the randomness (variance) in the high dimensionality space as possible. And the objective of LDA is to perform dimensionality reduction while preserving as much of the class discriminatory information as possible.

It was demonstrated that, in principle, classification of feeding behaviour in the aquarium via PCA and LDA was feasible for "ideal" conditions for example in an aquarium. From the video data from Jonkershoek Aquaculture farm (shown in chapter 3) it can clearly be seen that the video footage was unable to be used to for analysis. It is recommended that in the case of an aquaculture farm other methods be utilised to observe feeding behaviour and other methods be investigated to overcome visibility problems in order to analyse the video data.

Chapter 6

Conclusions

In this thesis, the feasibility of observing fish behaviour as a basis for optimized automated feeding is considered. Although useful digital images from fish under actual aquacultural conditions at the Jonkershoek farm in Stellenbosch could not be used, the feeding behaviour of fish in the Two Oceans Aquarium in Cape Town was analysed instead. Video images of the fish before, during and after feeding were used as the basis for analysis. The analytical methodology was based on a viewpoint of "detection of change" in the data.

The well-established multivariate methods of principal components analysis (PCA) and linear discriminant analysis (LDA) were used to extract informative features from the image data. These features were labelled with the corresponding behaviour they captured, namely the prefeeding, feeding and postfeeding behaviour of fish.

The extracted features were used to train classification models to distinguish between the abovementioned classes of behaviour. More specifically, classification tree models were developed using PCA features, and K nearest neighbours models were trained using LDA features. The performance of these models was evaluated on the basis of how accurately they could assign the correct class of behaviour to an independent set of test features.

By using either PCA or LDA features, the behaviour of the fish could be classified very reliably. Since this could be achieved with very little preprocessing of the data, it suggests that the observed behaviour of fish could form the basis of a cost-effective automated feeding system.

It was found during this study, that at the Jonkershoek aquaculture farm, external environmental factors would play a significant role in acquiring quality image data. These factors included turbidity induced by rain and considerable changes in lighting conditions.

The challenge of acquiring quality image data under these typically changing environmental conditions would have to be surmounted for the successful implementation of the proposed method.

Chapter 7

References

Adelezi P (1998); Biotechnology in Aquaculture: The Future of Fish Farming; The Agbiotech Infosource; Published by AG-West Biotech Inc. Issue 33, February 1988. (www.agwest.sk.ca/publications)

Aldrich C (2001); Making Sence from Histroric Process Data; University of Stellnbosch.

Alpayadin E (2004); Introduction to Machine Learning (Adaptive Computation and Machine Learning), MIT Press, ISBN 0262012111.

Alverson DL, Freeberg MK, Muawski SA & Pope JG (1994). A global assessment of fisheries bycatch and discards. FAO Fisheries Technical Paper No. 339. Rome

Ang KP, Petrell RJ (1997); Control of feed dispensation in seacages using underwater video monitoring: effects on growth and feed conversion, Aquaculture Engineering, May 15, 1996, pp. 45-62, vol. 16, Elsevier Science, New York, NY.

Beveridge M (2004); Cage Aquaculture; Third Edition, Oxford, UK.

Bjordal A, Juell JE, Lindem T, A. Ferno A (1993); Hydroacoustic monitoring and feeding control in cage rearing of Atlantic salmon, Fish Farming Technology, pp. 203-208, Balkema, Rotterdam.

Bishop C (2006); Pattern Recognition and Machine Learning, Springer, ISBN 0-383-31073-8.

Blyth PJ, Purser, GJ & Russell JF (1992) Boosting profits with adaptive feeding: letting the fish decide when they're hungry. AustAsia Aquaculture, Vol. 6 pp 33-38.

Blyth PJ, Purser GJ & Russell JF (1993); Detection of feeding rhythms in sea-caged Atlantic salmon using new feeder technology, Fish Farming Technology, pp. 209-216, Balkema, Rotterdam.

Blyth PJ, Kadri S, Valdimarsson SK, Mitchell DF (1999); Diurnal and Seasonal variation in feeding patterns of Atlantic Salmon, Salmo salar L., in sea cages; Aquaculture Research, pp. 539 – 544, Vol. 30, Blackwell Science Ltd, Australia.

Boyd, C.E. 1999. Aquaculture sustainability and environmental issues. World Aquaculture, 30(2) 10-13.

Bridger CJ (2003); Feedfish Offshore, Gulf of Mexico aquaculture consortium college of marine sciences (www.masgc.org/oac)

Bulcock, P., Bostock, J. C., Beveridge, M. C. M. and Telfer, T. C. (2001). The evolution of aquaculture feed supply systems. Eurofish 2, 74-76.

Chang CM, Fang W, Jao RC, Shyu CZ, Liao IC (2005); Development of an intelligent feeding controller for indoor intensive culturing of eel; Aquacultural Engineering, July 20, 2004, pp. 343 – 353, Vol. 32, Elsevier Science, New York, NY.

De Wet L (2007); Department of Aquaculture, University of Stellenbosch. – nie gepaste reference nie

Dunn M & Dalland K (1993); Observing behaviour and growth using the Simrad FCM 160 Fish Cage Monitoring System; Fish Farming Technology, pp. 269-274, Balkema, Rotterdam.

Euro Fish Magazine (April 2001); The evolution of Aquaculture Feed Supply, http://www.eurofish.dk/indexSub.php?id=797&easysitestatid=-460526522

FAO (1999); Aquaculture Production Statistics 1988-1997 (Food and Agriculture Organisation, Rome, 1999).

FAO (2004) Fisheries Department; The State of World Fisheries and Aquaculture, 2004, Food and Agriculture Organisation, Rome, 2004.

Foster M, Petrell R, Ito MR & Ward R (1995); Detection and Counting of Uneaten Food Pellets in a Sea Cage Using Image Analysis, Aquacultural Engineering, Aug 8, 1994, pp. 251-269, vol. 14, No. 3, Elsevier Science, New York, NY.

Franklin et al. (2002) Feedback control of Dynamic Systems, 4, New Jersey, Prentice Hall ISBN 0-13-032393-4.

Fullerton B, Swift MR, Boduch S, Eroshkin O, Rice G (2004); Design and Analysis of an automated feed-buoy for submerged cages; Aquacultural Engineering, May 12, 2003, pp. 95 – 111, Vol. 32, Elsevier Science, New York, NY.

Greenpeace (2007); www.greenpeace.org/international/

Gulf of Mexico Offshore Aquaculture Consortium (2003); www.masgc.org/oac/gallery05.htm; College of Marine Scineces; USM; Ocean Springs.

Heilprin J (2005); Wasted bycatch detailed in study, Nation World, 1 December 2005.

Huet (1975); Textbook of Fish Culture, Breeding and Cultivation of Fish. Fishing News (Books), Surrey, England.

Jemwa G.T. PhD (2007). Department of Process Engineering, University of Stellenbosch.

Juell JE (1991); Hydroacoustic Detection of Food Waste--A Method to Estimate Maximum Food Intake of Fish Populations in Sea Cages; Aquacultural Engineering, May 28, 1991, pp. 207-217, vol. 10, Elsevier Science, New York, NY.

Juell JE, Furevik DM & Bjordal A (1993); Demand Feeding in Salmon Farming by Hydroacoustic Food Detection; Aquacultural Engineering, Sep. 30, 1992, pp. 155-167, vol. 12, Elsevier Science, New York, NY.

Juell JE, Holm JC, Beltestad AK (1996); In situ acoustic observation of food consumption in caged Atlantic mackerel, Fisheries Research, Dec. 9, 1995, pp. 131-135, vol. 27, Elsevier Science, New York, NY.

Kadri, S., Metcalfe, N.B. Huntingford, F.A. & Thorpe J.E. (1991) Daily feeding rhythms in Atlantic salmon in sea cages. Aquaculture, vol. 92, pp. 219 – 24.

Kimura R, Kawaguchi S, Sugitani S (1993); Automatic feeding control system for fish; Fish Farming technology, pp. 217 – 225, Balkema, Rotterdam.

Lagardere JP, Mallekh R, Mariani A (2004); Acoustic characteristics of two feeding modes used by brown trout (Salmo trutta), rainbow trout (Oncorhynchus mykiss) and turbot (Scophthalmus maximus); Aquaculture, Sept. 8, 2003, pp. 607 – 616, vol. 240, Elsevier Science, New York, NY.

Lee PG, Lea RN, Dohmann E, Prebilsky W, Turk PE, Ying H, Witson JL (2000); Denitrification in aquaculture systems: an example of a fuzzy logic control problem; Aquacultrueal Engineering, Nov. 6, 1998, pp. 37 – 59, Vol. 23, Elsevier Science, New York, NY.

Li, SF (1994); Fish culture in cages and pens. In: Freshwater Fish Culture in China: Principles and Practices (Ed. By S Li & J Mathias), pp. 305 – 346. Elsevier, Amsterdam.

Myrseth B (2000); Automation of Feeding Management in Cage Culture; In Cage Aquaculture in Asia: Proceedings of the First International Symposium on Cage Aquaculture in Asia (ed. IC. Liao and C.K. Lin), pp. 151 – 155. Asian Fisheries Society, Manila, and World Aquaculture Society – Southeast Asian Chapter, Bangkok.

NOAA (1998) National Oceanic and Atmospheric Administration – Ecological Effects of Fishing (Brown S, Auster PJ, Lauck, Coyne M) NOAA's Staate of the Coast Report, Sivler Spring MD.

Ogilvie D (2006); So what's wrong with Fish? Vegetarian Network Victoria; www.vnv.org.au/Articles.

Papandroulakis N, Dimitris P, Pascal D (2002), An automated feeding system for intensive hatcheries, Aquacultural Engineering, Aug 12, 2001, pp. 13 – 26, Vol. 26, Elsevier Science, New York, NY.

Pauly D, Alder J, Bennet E, Christensen V, Tyedmers P, Watson R (1998); The future for fisheries. Science, Vol. 302, pp. 1359 – 1361.

Phillips, M.J., Beveridge, M.C.M & Ross L.G. (1985). The environmental impact of salmonid cage culture on inland fisheries: present status and future trend. Journal of fish Biology, Vol. 27 pp 123 – 37

Rowland DW (1999); Studying visual cues in fish behaviour: a review of ethological techniques; Environmental Biology of Fishes; Dec 29, 1998, pp. 285 – 305, Vol. 56, Kluwer Academic Publishers, Netherlands

Shakya G (2007); Rainbow Trout Fish Farming; The Rising Nepal (website: www.gorkapatra.org.np)

Smith BC, Wikfors GH (1998); An automated rearing chamber system for studies of shellfish feeding; Aquacultural Engineering; Jan. 29, 1997, pp. 69 – 77, Vol. 17, Elsevier Science, New York, NY.

Smith IP, Metcalfe NB, Huntingford FA & Kadri S (1993); Daily and seasonal patterns in the feeding behaviour of Atlantic salmon (Salmo salar L.) in a sea cage; Aquaculture, June 4, 1993, pp. 165 – 178, Vol. 117, Elsevier Science Publishers B.V., Amsterdam.

Tacon A (1999); Estimated Global Aquafeed Producation and Aquaculture in 1997 and projected growth; International Aquafeed, Vol. 2 p.5.

Talbot C, Corneillie S & Korsoen O (1999); Pattern of feed intake in four species of fish under commercial farming conditions: implications for feeding management; Aquaculture Research, pp. 509 – 518, Vol. 30, Blackwell Science Ltd., Australia.

Thorpe JE & Huntingford FA (1992); The importance of Feeding Behaviour for the Efficient Culture of Salmonoid Fishes. World Aquaculture Society, Baton Rouge, LA.

Van Zyl J (2006); "Die visse teel soos konyne", FinWeek Magazine 14 September 2006, South Africa.

Wise MW & Gallagher NB (1996); The process of chemometrics approach to process monitoring and fault detection; Journal of Process Control, Vol 6.; pp 329-348; Elsevier Science Ltd.; Great Britain.

World Wide Fund (2007): www.wwf.org

Appendix

A. Initialising Gist

Gist Version 2.31 is available from Gorden Jemwa (2007)

Gist® Code using Matlab Interface

>> system('gist-kpca --help')

Usage: gist-kpca [-train <filename> (required)]

[-zeromeanrow]

[-varone]

[-nonormalize]

[-constant <value> (default=10)]

[-coefficient <value> (default=1)]

[-power <value> (default=1)]

[-radial]

[-widthfactor <value> (default = 1)]

[-width <value> (default = use widthfactor 1)]

[-adddiag <value> (default=0)]

[-matrix]

[-nocenter]

[-numeigens <value> (default = all)]

[-eigenthresh <value> (default = 0.000001)]

[-eigenvalues <file>]

[-rdb]

[-kernelout]

[-notime]

[-precision <value>]

[-verbose 1|2|3|4|5 (default=2)]

Secondly to test the data using Gist Project:

>> system('gist-project --help')

Usage: gist-project [-train <filename> (required unless test set is a kernel matrix)]

[-learned <filename> (required)]

[-test <filename> (required)]

[-selftrain <filename>]

[-selftest <filename>]

[-rdb]

[-kernelout]

[-notime]

[-precision <value> (default = 6)]

[-verbose 1|2|3|4|5 (default=2)]

B.Gist Manuals

Gist-kpca - Manual

Description: Compute kernel-based eigenvectors for a set of training examples.

Usage: gist-kpca [options] -train <filename>

Input: -train <filename> - a tab-delimited, labeled file of training examples. The first column contains labels, and the remaining columns containing real-valued features.

Output: A tab-delimited matrix in which each column corresponds to an eigenvector. Eigenvectors are normalized so that the dot product of the eigenvector with itself equals the reciprocal of the corresponding eigenvalue. In the output, the eigenvectors are sorted by increasing magnitude.

Options:

- -zeromean - Subtract from each element in the input data the mean of the elements in that row, giving the row a mean of zero.
- -varone - Divide each element in the input data by the standard deviation of the elements in that row, giving the row a variance of one.

By default, the base kernel function is a dot product. In this case, the kernel-pca will give the same results as a 'standard' principal component analysis. If desired, this kernel can be modified using the following options. The operations occur in the order listed below.

- -nocenter - PCA requires a centered matrix, in which the sum of each column is zero. This centering operation can be performed in kernel space, and is done by default. The -nocenter option disables this operation. This option is only useful in conjunction with the -kernelout operation, to produce an intermediate matrix.
- -adddiag <value> - Add the given value to the diagonal of the kernel matrix.
- -nonormalize - Do not normalize the kernel matrix. By default, the matrix is normalized by dividing K(x,y) by sqrt(K(x,x) * K(y,y)).
- -constant <value> - Add a given constant to the kernel. The default constant is 10.
- -coefficient <value> - Multiply the kernel by a given coefficient. The default coefficient is 1.
- -power <value> - Raise the kernel to a given power. The default power is 1.
- -radial - Convert the kernel to a radial basis function. If K is the base kernel, this option creates a kernel of the form $\exp[(-D(x,y)^2)/(2\ w^2)]$, where w is the width of the kernel (see below) and D(x,y) is the distance between x and y, defined as $D(x,y) = \mathrm{sqrt}[K(x,x)^2 - 2\ K(x,y) + K(y,y)^2]$.
- -widthfactor <value> - The width w of the radial basis kernel is set using a heuristic: it is the median of the distance from each training point to the nearest training point. This option specifies a multiplicative factor to be applied to that width. The default is a width factor of 1.

- -width <value> - Directly set the width w of the radial basis kernel. If set, this option overrides the -widthfactor option.

If the supplied kernel functions are insufficient, the user can supply as input a precalculated kernel matrix using the following option

- -matrix - By default, the base kernel function is a dot product. This option allows that function to be replaced by an arbitrary function supplied by the user (for many commonly used kernels, see the options listed above). If supplied, the software reads kernel values, rather than raw feature data, from the file specified by -train. The matrix must be an n+1 by n+1 tab-delimited matrix, where n is the number of training examples. The first row and column contain data labels. The matrix entry for row x, column y, contains the kernel value K(x,y).

The remaining options (except for -rdb) affect the output of the software.

- -numeigens <value> - Include in the output at most the specified number of eigenvectors (subject to the next constraint). By default, all are included.
- -eigenthresh <value> - Include in the output only eigenvectors whose corresponding eigenvalues are greater than the specified value. Default value is zero (all eigenvectors).
- -eigenvalues <file> - Create a file with the given name and store the eigenvalues there as a space-separated array of numbers.
- -rdb - Allow the program to read and create RDB formatted files, which contain an additional format line after the first line of text.
- -kernelout - Compute and print the kernel matrix to stdout. Do not compute the eigenvectors.
- -notime - Do not include timing information in the output header.
- -precision <value> - Number of digits after the decimal place in the output file. The default is 6.
- -verbose 1|2|3|4|5 - Set the verbosity level of the output to stderr. The default level is 2.

Gist-project - Manual

Description: Project a given set of data onto a given set of eigenvectors.

Usage: gist-project [options] -train <filename> -learned <filename> -test <filename>

Input: -train <filename> - an tab-delimited file of training data. The first column contains labels, and the remaining columns containing real-valued features.

- -learned <filename> - a tab-delimited file of eigenvectors, as produced by kernel-pca. Each column corresponds to an eigenvector.
- -test <filename> - a tab-delimited file of test data to be projected onto the training data's eigenvectors.

Output: A tab-delimited matrix in which the given data has been projected onto the given set of eigenvectors.

Options:

- -selftrain <filename> - Read from the given file a series of n values of the form K(x,x), where K is the base kernel function and x is an element in the training set. This option is only necessary if the base kernel function is supplied from a file and the kernel is normalized or radial basis. The input file should be in tab-delimited format, with data labels in the first column and values in the second column.
- -selftest <filename> - Similar to '-selftrain', but for the test set.
- -rdb - Allow the program to read and create RDB formatted files, which contain an additional format line after the first line of text.
- -kernelout - Compute and print the kernel matrix to stdout. Do not compute the classifications.
- -notime - Do not include timing information in the output header.
- -precision <value> - Number of digits after the decimal place in the output file. The default is 6.
- -verbose 1|2|3|4|5 - Set the verbosity level of the output to stderr. The default level is 2.

C. Matlab Code

1. Get_framedata.m

```
% user-defined variables

fname = 'train.dat'; % change to correspond to all_classe
below
fname_labels = 'train.lab'; % change to correspond to
all_classes below
all_classe = [2];%#ok %% which class to include in matrix
fname. 1|2|3 (class 1: pre-feeding, class 2: feeding, class 3:
post-feeding)


num_segments = 20;
num_frames= 50; % avoid num_frames > 100

% to avoid appending onto existing files. Otherwise rename
fname and
% fname_labels (the variables and NOT variable_names)
warning('off') %#ok
delete(sprintf(fname));
delete(sprintf(fname_labels));
warning('on') %#ok


start_frame = [1 9501 15001];
class_id = [];

select_rows=[1 240];
select_cols=[251 450];


%Open an handle to the AVI file
% first clear existing files

 if exist('avi_hdl','var')
        dxAviCloseMex(avi_hdl);
 end

 if exist('grsclfrms','var')
        clear grsclfrms
 end

[avi_hdl, avi_inf] = dxAviOpen('MOVIE0004.avi');
p=avi_inf.Width;
q=avi_inf.Height;
%last_counter = 1;
```

```
total_segments = 0;

for class= all_classe %#for classes 1 to 3,

    for segment=1:num_segments,
        start_frame_ind = start_frame(class)+(segment-
1)*num_frames;
        grsclfrms=zeros(q,p,num_frames);

        for frame_num =
start_frame_ind:start_frame_ind+num_frames-1;
            %Reads frame_num from the AVI
            pixmap = dxAviReadMex(avi_hdl, frame_num);
            grsclfrms(:,:,frame_num-start_frame_ind+1) =...

rgb2gray(reshape(pixmap/255,[avi_inf.Height,avi_inf.Width,3]))
;
        end
        total_segments = total_segments+1;
        current_counter = (total_segments - 1)*num_frames + 1;


        % create a new matrix

grsclfrms=grsclfrms(select_rows(1):select_rows(2),select_cols(
1):select_cols(2),:);

grsclfrms=reshape(grsclfrms,numel(grsclfrms(:,:,1)),num_frames
)';

 % dlmwrite('test.dat', grsclfrms,'delimiter','\t','-append');

grsclfrms=[(current_counter:current_counter+(num_frames-1))'
grsclfrms]; %#ok
        %last_counter = (total_segments - 1)*num_frames + 1;
        % write matrix to fname (text document in directory)


   %    dlmwrite(fname, num2str('label'),'delimiter', '\t','-
append');
   %
dlmwrite(fname,1:size(grsclfrms,2),'delimiter','\t','-
append');


%Prepare formatting string (labels in text document)
        if (((class==1 || length(all_classe)==1)) &&
        segment==1),
            %%

            str = 'label\t';
```

```
            n=size(grsclfrms,2);
            for i=1:n-2,
                str = [str,'X%d\t'];%#ok
            end

            str = [str,'X%d\n'];%#ok

            % Write using fprintf
            % tic
            fid = fopen(fname,'wt');
            fprintf(fid,str,(1:n-1));
            fclose(fid);
        end



        dlmwrite(fname, grsclfrms,'delimiter','\t','-append');

dlmwrite(fname_labels,class*ones(size(grsclfrms,1),1),'-
append','delimiter',' ');

    end % for segment
end % for class

 %Cleanup
dxAviCloseMex(avi_hdl); clear avi_hdl
```

2. Write Frame Labels (Supervised Learning/Target Vector)

```
function enc_labels = ind2mat(labels,assigned_class,max_class)

% simple 1-of-N class encoding


  if nargin<3,
      max_class = [];
  end


  if nargin<2,
      assigned_class = 0;
  end

  classes = unique(labels);


  if ~isempty(max_class),
      classes = min(1,min(classes)):max_class;
  end
```

```
   enc_labels =
   assigned_class*ones(length(labels),length(classes));

for i=1:length(classes),
      ind = ( labels == classes(i) );
       enc_labels(ind,i) = 1;
end
```

3. Gist_train_KPCA.m

```
function
dist_kpca_train(fname,pca_model_name,numeigens,eigenval,eigent
hresh, verbose)
% GIST_KPCA_TRAIN(fname,pca_model_name,numeigens,eigthresh,
verbose)
%
% train PCA model using gist v2.3
%INPUTS
%        fname - (required) filename containing data (obtained
from diff_using_directXread2.m)
%        pca_model_name - name of file to save trained model
[default: fname_numeigens.pca-model]
%        numeigens - number of eigenvectors and eigenvalues to
be retained [less than min(numrows,numcols)]
%        eigthresh - default eigenvalue threshold
%        verbose - level of info from gist [1(minimal),
2(default), 3, 4, 5(maximal)]
%verbose, save)

%define the program arguments

if nargin<1,
    system('gist-kpca --help')
    return
end
if nargin<2,
    pca_model_name = sprintf('%s.pca-model',fname);
end

if nargin<3,
    numeigens = 0; % default = 0 (all)
end

if nargin<4
    eigenval = sprintf('%s.eigval-dat',fname);
end

if nargin<5,
```

```
    eigenthresh = 0.000001; % default eigenvalue threshold
end

if nargin<6,
    verbose = 5; % 1|[2]|3|4|5
end


call_func = sprintf('gist-kpca -train %s -verbose %d -
numeigens %d -eigenvalues %s -eigenthresh %4.10f > %s',...

fname,verbose,numeigens,eigenval,eigenthresh,pca_model_name);
system(sprintf('%s',call_func));
```

4. Gist_test_KPCA.m

```
function
gist_kpca_test(train_fname,modelfile,test_fname,scores_fname,v
erbose)
%GIST_KPCA_TEST(train_fname,modelfile,test_fname,scores_fname,
verbose)
%
%project test data onto PCA model using gist v2.3
%INPUTS
%       train_fname    - (required) filename containing
training data
%       modelfile      - (required) filename containing
(kernel-)PCA
%                           model
%       test_fname     - name of file containing test data to
be
%                          projected. If empty, train_fname
is used
%       scores_fname   - name of file to save principal
componets or scores
%       verbose        - level of info from gist [1(minimal),
2(default), 3, 4, 5(maximal)]

%define some program arguments

if nargin < 2,
    system('gist-project --help')
    return
end
if nargin < 3,
    fprintf(1,'Test data not specified. Training data to be
used\n');
    test_fname = train_fname;
end
```

```
if nargin <4,
    scores_fname = sprintf('scores-%s',test_fname);
end

if nargin <5,
    verbose = 2;
end

call_func = sprintf('gist-project -train %s -learned %s -
verbose %d -test %s > %s',...

train_fname,modelfile,verbose,test_fname,scores_fname);
system(sprintf('%s',call_func));
```

5. Gist_gda_train.m (no manual available)

```
function a=gist_gda_train(fname,classes_file,varargin)
%function gist_kpca_train(fname,varargin)
% GIST_GDA_TRAIN(fname,varargin)
%
% train gda model using gist v2.31
%INPUTS
%        train_filename - (required) filename containing data
%  varargin: paired inputs of the form ('str',str_input)
%            where 'str'
%        model_file - name of file to save trained model
[default:
%                         fname_numeigens.pca-model]
%        classes -  (required) filename containing data
%                                    categories in matrix form.
%        rowmeans -  (required) filename to save training row
%                              means.Useful for efficient
matrix centering
%                              when projecting test data
%        verbose - level of info from gist [1(minimal),
2(default), 3, 4, 5(maximal)]
%verbose, save)
%define some program arguments
if nargin<2,
    system('gist-gda --help')
    return
end

% remove extensions from training file name
sep = '.';
%new_sep = '-';
ind_sep = find(sep==fname);
if ~isempty(ind_sep),
    fname2 = sprintf('%s',fname(1:ind_sep-1));
```

```
end

% set defaults
a.train_fname=sprintf('%s',fname);
a.classes = sprintf('%s',classes_file);
a.model=sprintf('%s.gda-model',fname2);
%a.model_eigs_fname = sprintf('%s.gda-model-eigs',fname2);
a.rowmeans = sprintf('%s.gda_row-means',fname);
%a.feats = 0;
a.verbose = 2;
a.trained = 0;

if ~isempty(varargin),
    for i=1:length(varargin),
        if strcmpi(varargin{i},'model_file'),
            a.model_fname=sprintf('%s',varargin{i+1});
        end
        if strcmpi(varargin{i},'rowmeans'),
            a.rowmeans=varargin{i+1};
        end
        if strcmpi(varargin{i},'verbose'),
            a.verbose=varargin{i+1};
        end

    end

end

%a.model_fname=sprintf('%s-%d_ret-
eigvals',a.model_fname,a.num_eigs);
%a.model_eigs_fname = sprintf('%s-%d_ret-
eigvals',a.model_eigs_fname,a.num_eigs);


call_func = sprintf('gist-gda -train %s -class %s -rowmeans %s
-nonormalize -verbose %d > %s',...

a.train_fname,a.classes,a.rowmeans,a.verbose,a.model);
system(sprintf('%s',call_func));
a.trained=1;
```

6. Gist_gda_test.m

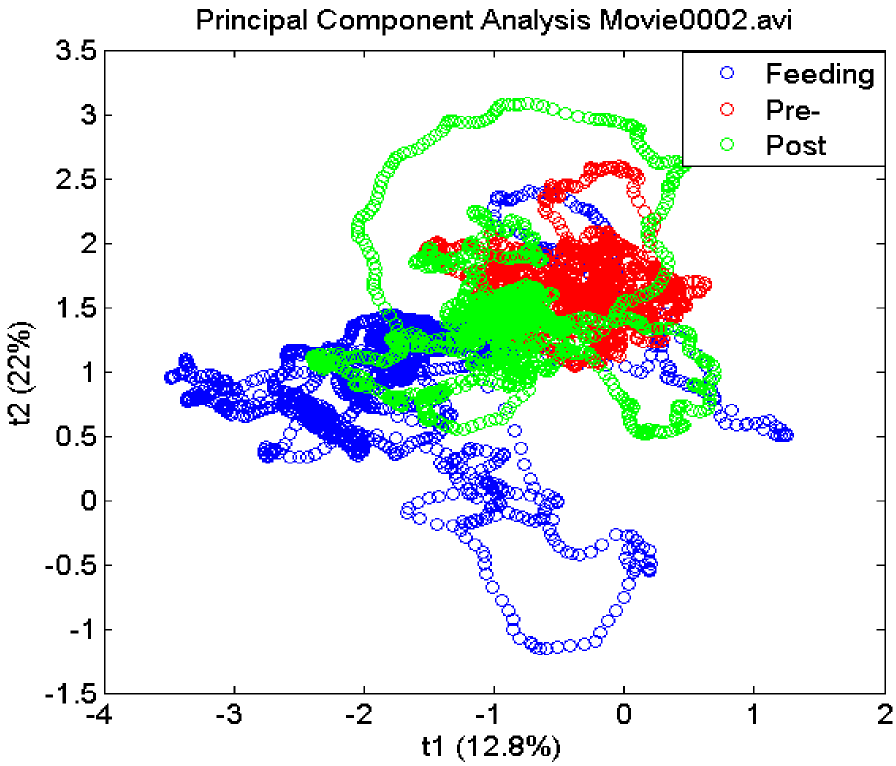
```
function a=gist_gda_train(fname,classes_file,varargin)
%function gist_kpca_train(fname,varargin)
% GIST_GDA_TRAIN(fname,varargin)
%
% train gda model using gist v2.31
%INPUTS
%       train_filename - (required) filename containing data
```

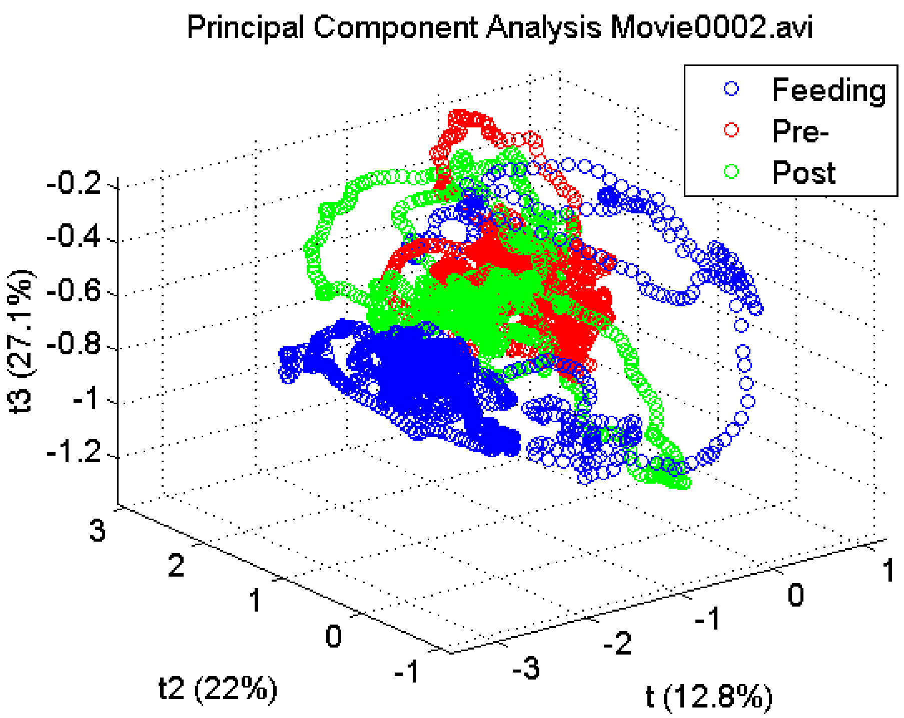
```
%  varargin: paired inputs of the form ('str',str_input)
%            where 'str'
%        model_file - name of file to save trained model
[default:
%                         fname_numeigens.pca-model]
%        classes -  (required) filename containing data
%                                  categories in matrix form.
%        rowmeans -  (required) filename to save training row
%                            means.Useful for efficient
matrix centering
%                            when projecting test data
%        verbose - level of info from gist [1(minimal),
2(default), 3, 4, 5(maximal)]
%verbose, save)
%define some program arguments
if nargin<2,
    system('gist-gda --help')
    return
end

% remove extensions from training file name
sep = '.';
%new_sep = '-';
ind_sep = find(sep==fname);
if ~isempty(ind_sep),
    fname2 = sprintf('%s',fname(1:ind_sep-1));
end

% set defaults
a.train_fname=sprintf('%s',fname);
a.classes = sprintf('%s',classes_file);
a.model=sprintf('%s.gda-model',fname2);
%a.model_eigs_fname = sprintf('%s.gda-model-eigs',fname2);
a.rowmeans = sprintf('%s.gda_row-means',fname);
%a.feats = 0;
a.verbose = 2;
a.trained = 0;

if ~isempty(varargin),
    for i=1:length(varargin),
        if strcmpi(varargin{i},'model_file'),
            a.model_fname=sprintf('%s',varargin{i+1});
        end
        if strcmpi(varargin{i},'rowmeans'),
            a.rowmeans=varargin{i+1};
        end
        if strcmpi(varargin{i},'verbose'),
            a.verbose=varargin{i+1};
        end

    end
```

```
end

%a.model_fname=sprintf('%s-%d_ret-
eigvals',a.model_fname,a.num_eigs);
%a.model_eigs_fname = sprintf('%s-%d_ret-
eigvals',a.model_eigs_fname,a.num_eigs);


call_func = sprintf('gist-gda -train %s -class %s -rowmeans %s
-nonormalize -verbose %d > %s',...

a.train_fname,a.classes,a.rowmeans,a.verbose,a.model);
system(sprintf('%s',call_func));
a.trained=1;
```

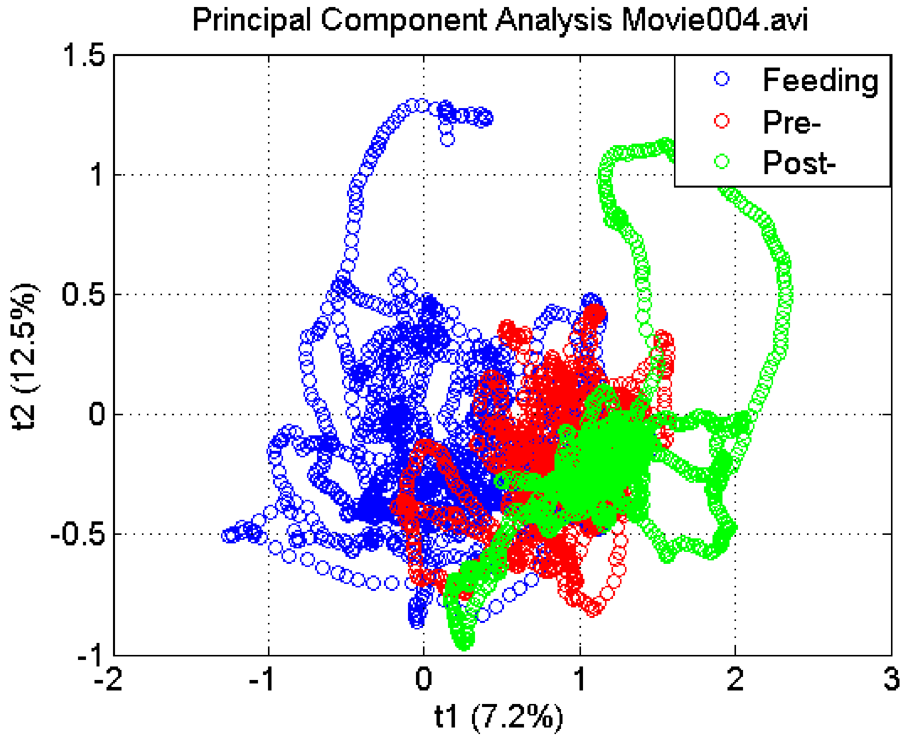
D. Results from Other Video Data for PCA

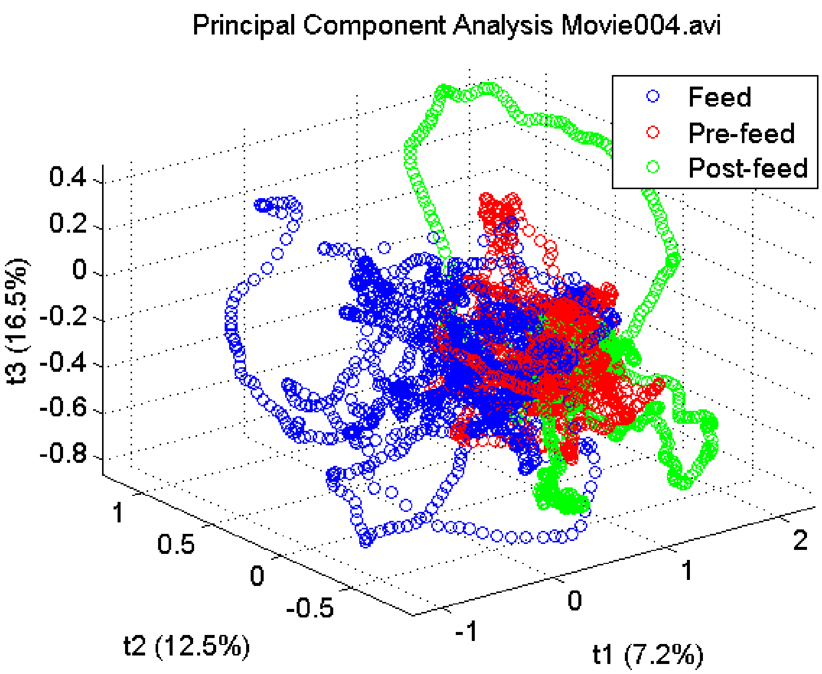
Movie0002.avi



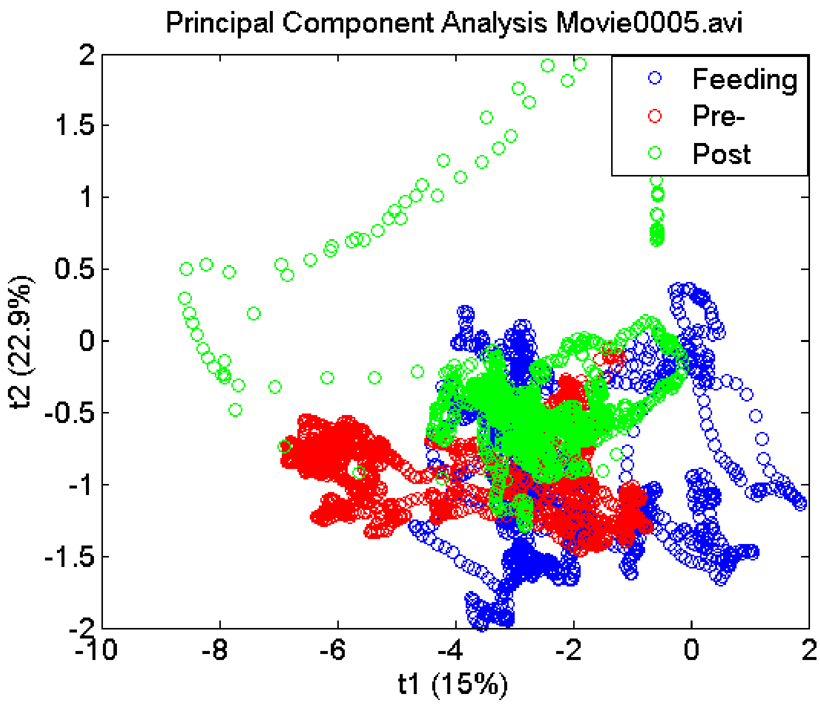


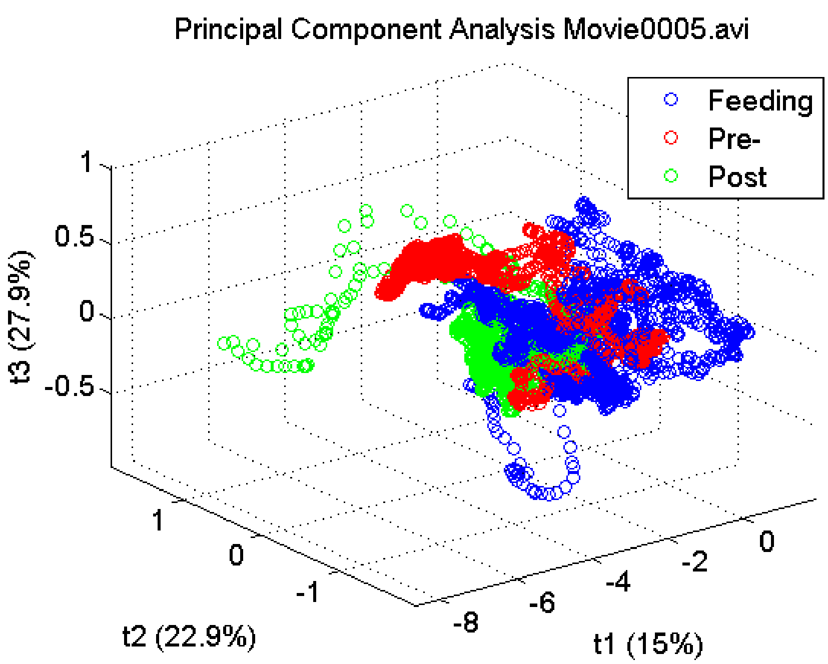
Movie0004.avi





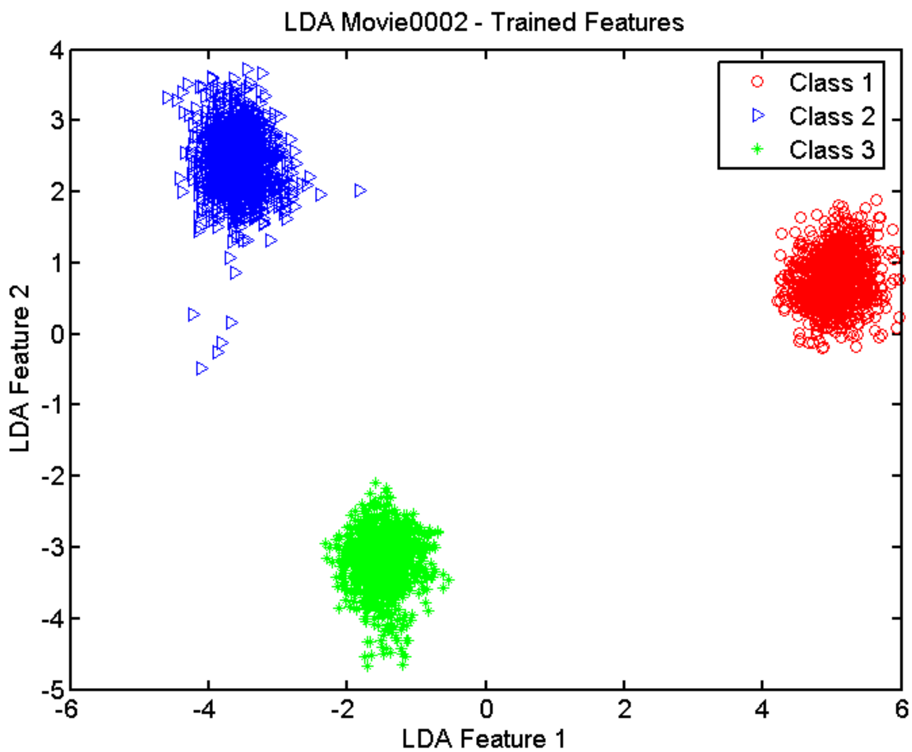
Movie0005.avi

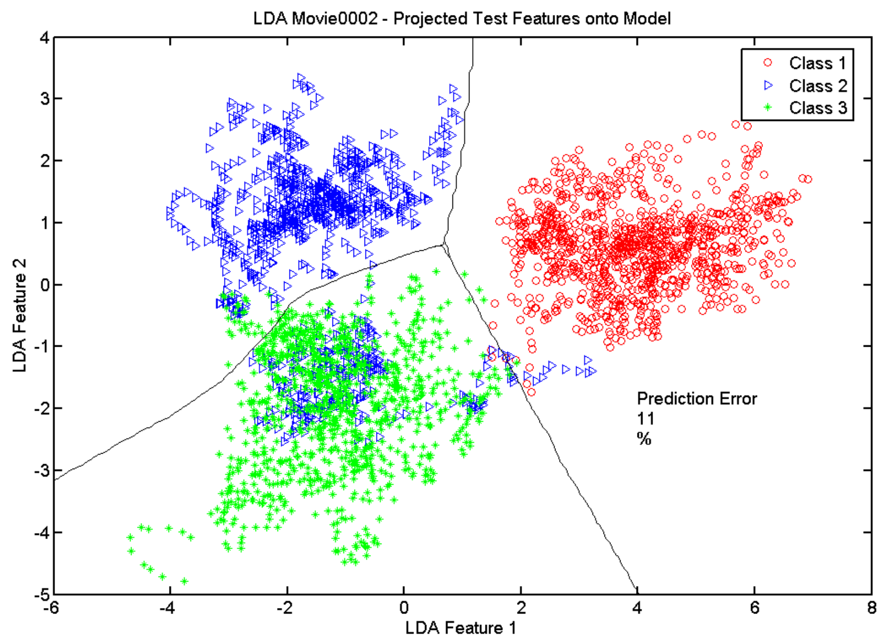




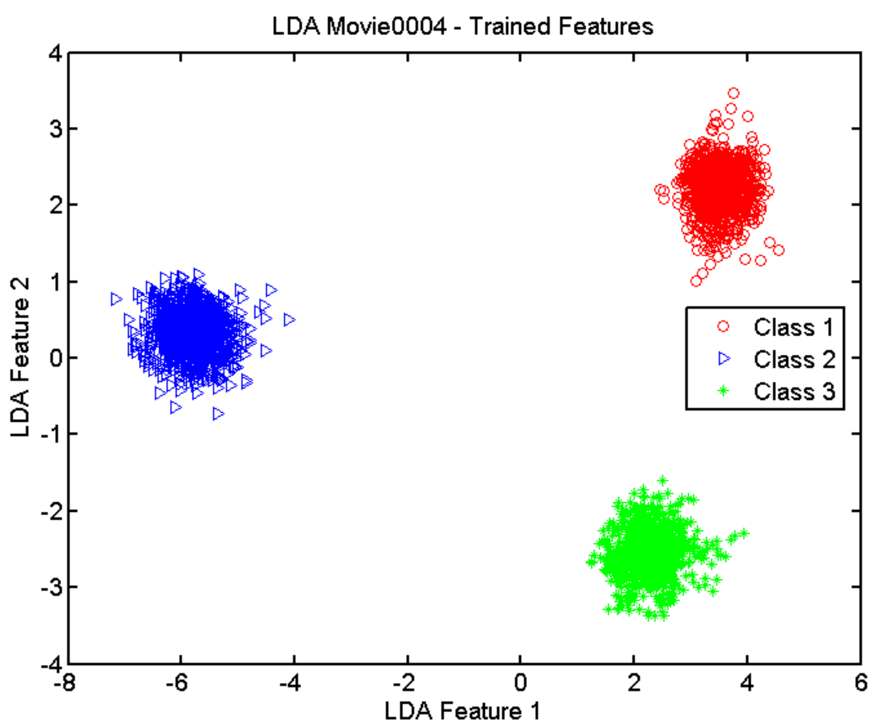
E. Results from Other Video Data for LDA

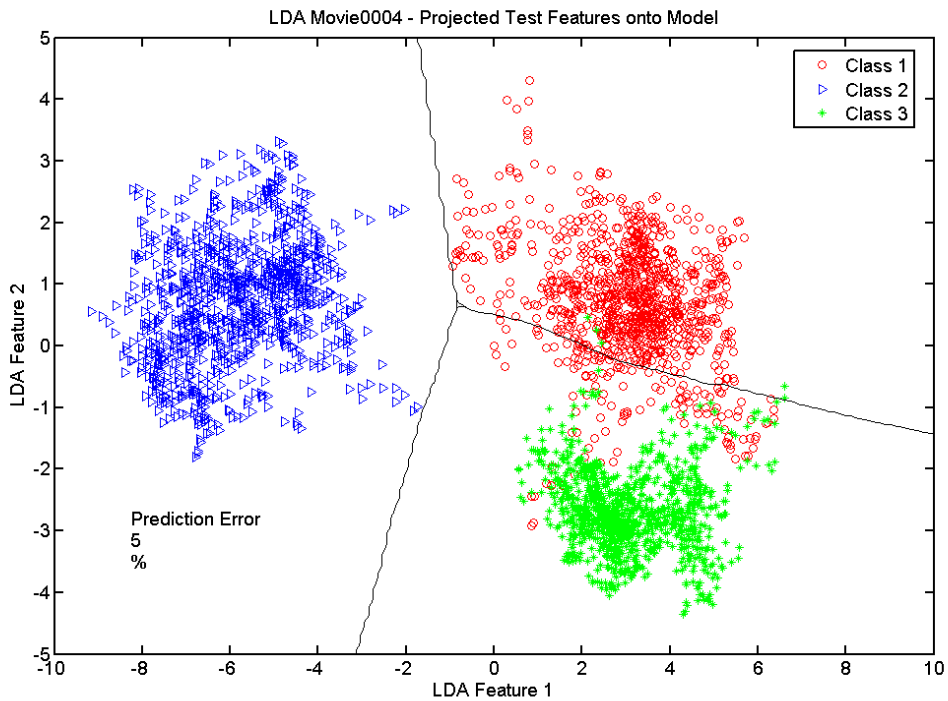
Movie0002.avi



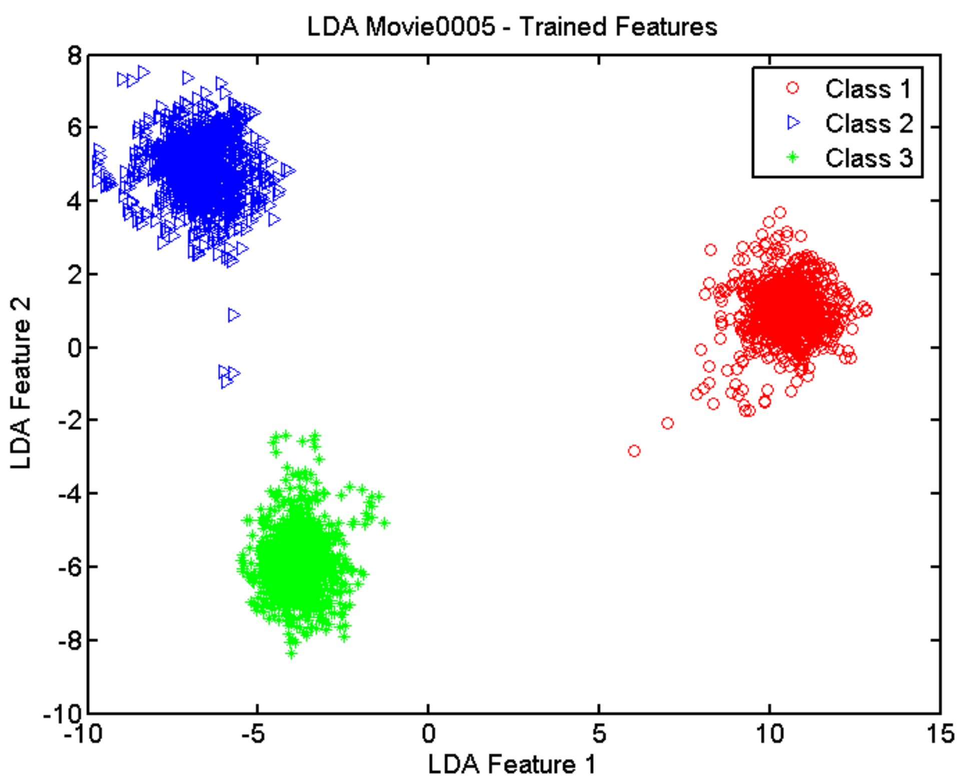


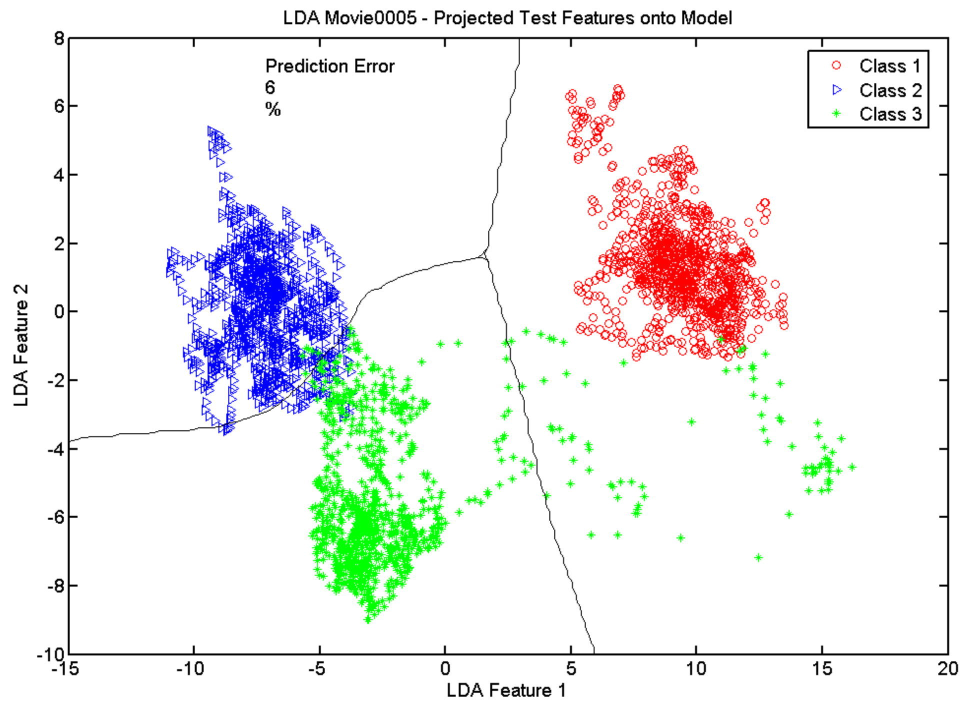
Movie0004.avi





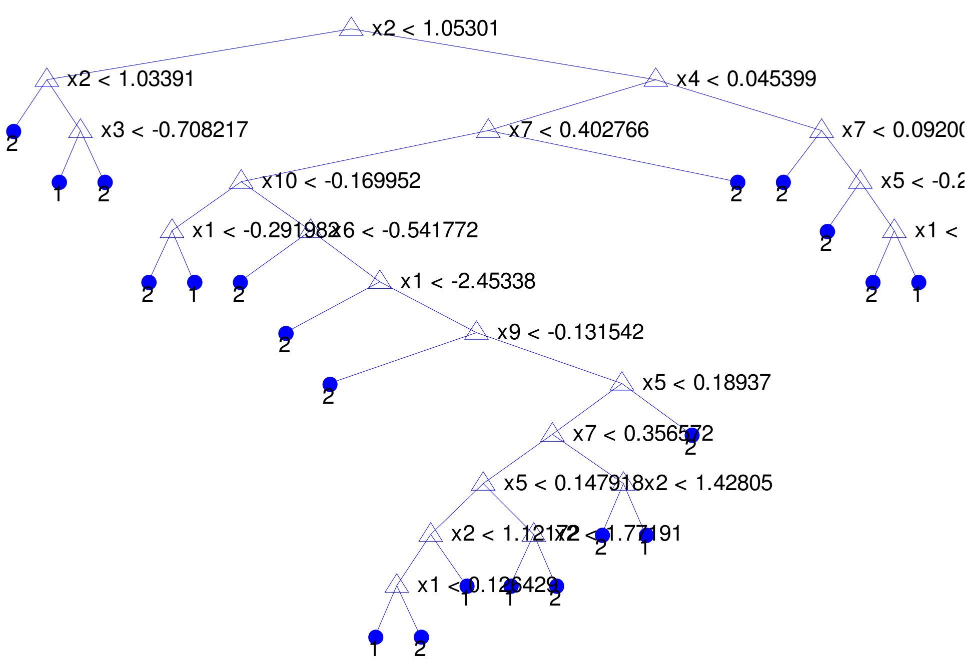
Movie0005.avi



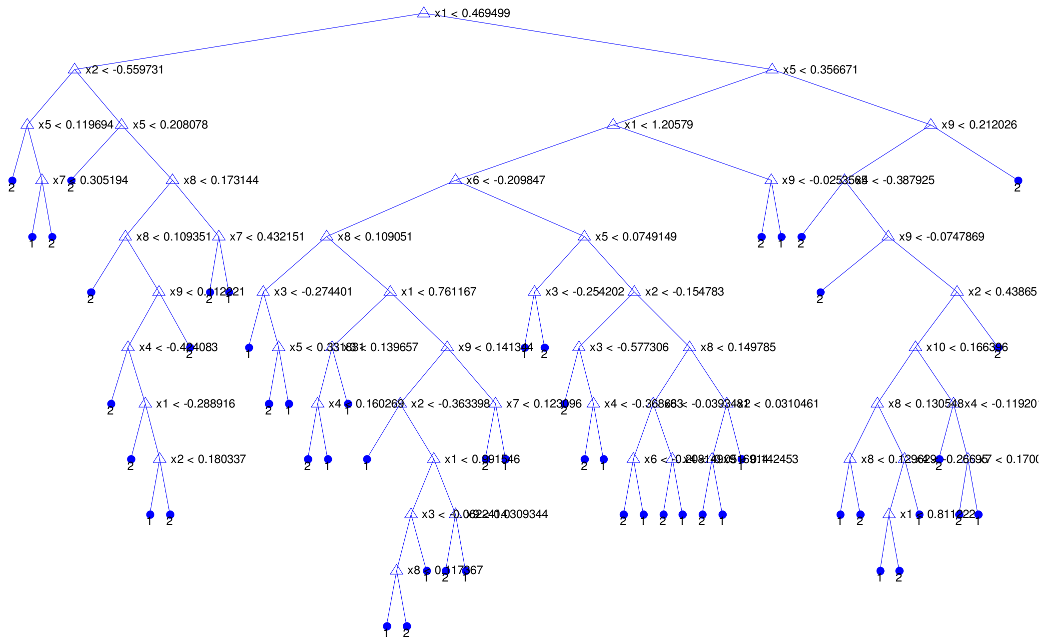


F. Classification Tree Data

Classification Tree Movie 0002



Classification Tree Movie 0004



Classification Tree Movie 0005