Acknowledgements

I would like to express my sincere thanks to Paul Whelan for his help and guidance during the course of this project.

Thanks also to Professor Charles McCorkell whose advice and interest in my project was motivating.

I would also like to thank my colleagues and members of the vision group, particularly Derek Molloy, Gabriel Duffy, Peter Sutton, Ken McClannon, Kevin Clarke and Niall Dorr. Without their help and ideas this project would not have been possible.

Thanks also to the *Block 2* postgrads and technicians who were always there to help out when called upon.

My profound thanks to the members of the blind community for their advice and ideas which ultimately determined the course of this project.

Many thanks also to Dublin City University College Bar for many memorable evenings.

Last but not least, I would like to thank my parents for their endless encouragement and support.

Abstract

This project examines the possibility of applying machine vision research to aid visually impaired persons with the recognition of day to day objects. Background research into everyday problems for visually impaired persons was carried out to see how best machine vision could be developed to tackle these problems. Methods of extracting, storing and classifying shape and colour information were examined, implemented and tested. The best of these methods were combined into the prototype system which was then tested rigorously. The results obtained along with all the problems encountered are discussed in detail. The feasibility and recommendations for further development of this project are also discussed.

Table of Contents

Table of Figures

1. Introduction

The aim of this project is to develop a machine vision system that is capable of identifying common objects for visually impaired persons. This project is a part of ongoing work for the visually impaired being carried out by the vision systems group at Dublin City University. This report describes in detail the various stages of development of this project, from initial research to the testing of the final prototype. Throughout this report, this project will be referred to as MAVVIP(**MA**chine **V**ision aid for **V**isually **I**mpaired **P**ersons).

1.1 Overview and objectives

During the course of this project, a system was developed which was capable of recognising objects from shape and colour information. There are many ways this technology can be of benefit to visually impaired persons. Identifying common household objects (e.g. credit cards, train and bus tickets, postcards, stationary, books, foodstuffs, medicines) is a difficult task for the visually impaired. MAVVIP will also be capable of providing extra information about the objects recognised (e.g. ingredients, cooking instructions, nutritional information, warnings) provided this information is added during the training procedure.

An Apple Macintosh computer and an Intelligent Camera were used as the platform to create the prototype. In total the hardware and software used cost approximately IR£10,000. It is expected that a second laptop based prototype would cost in the region of IR£2,000 to IR£4,000 to develop. The hardware required to develop the prototype is expensive and this prototype will not become economical until the software is improved and customised hardware is created for MAVVIP. This project is concerned mainly with developing a prototype which will be evaluated in terms of usefulness, reliability and feasibility.

Methods used in the various stages of MAVVIP had to be simple and use minimal processing time. The reasons for this are as follows :

- Less complex methods would be easier to implement on any portable or hand-held devices created.
- In general simple methods would require less time for training and recognition than more complex or computationally intensive methods.

There are constraints which MAVVIP would have to obey and they are summarised as follows :

- The features extracted would have to require relatively small amounts of storage space. This means that it is feasible to store large numbers of items in the database.
- MAVVIP must be capable recognising items regardless of changes in size or rotations. This is important so that an item can be successfully recognised regardless of the distance and position of the object from the camera.
- Extraction methods must require minimal processing. This is important in order to make the training and recognition procedures as quick as possible.

The long-term objectives of MAVVIP are summarised below :

- Should be capable of reading text.
- Should be capable of recognising colours and patterns of clothes.
- Should be capable of reading LCD’s (Liquid Crystal Displays).
- Should be created as a portable hand-held device.
- Should be financially affordable to the majority of visually impaired persons.

These long term goals are discussed in detail in *section 6.3*.

1.2 Motivation and justification

In recent times, modern society has undergone many technological changes. High technology gadgets with GUI’s (Graphical User Interfaces) and LCD’s are advancing user interfaces as we know them. These advancements in technology seem to have neglected the needs of visually impaired people (Gill(a), 1993).

When personal computers and Microsoft DOS became popular in the 1980’s, a new lease of life was given to visually impaired people, who with the help of screen readers could successfully use computers. At the end of the 1980’s with the advent of

Windows and graphical user interfaces, a new set of problems arose for the visually impaired community. Accessing GUI's has been tackled and there are packages now available that allow users to navigate through these environments. Access to pictorial information was another problem that arose and to this day the problem has been tackled and only met with limited success.

Another problem is that the buttons on washing machines, cookers, video recorders, stereo systems and microwaves have been replaced with touch pads and LCD's. This has made life considerably more difficult for visually impaired people.

There are many problems which exist which are not necessarily related to the current technological improvements. For example it can sometimes be difficult to distinguish between certain foodstuffs and domestic items because the packaging is similar. This could have severe consequences if harmful substances were mistaken for foodstuffs. Similarly there are many colour problems facing visually impaired persons. For example trying to distinguish between black and white socks can be a difficult task. These problems are just a few examples of the many problems which exist.

In the past aids for the visually impaired have tended to be simple in design and use. This simplicity of use has led to their popularity. Examples of such devices are templates for cheques and templates for identifying denominations of money. Unfortunately in this ever increasing technological world, simple aids will not be enough to combat the new problems faced by the blind. In the author's opinion, easy to use high technology devices will begin to play a more important role.

There are two main areas which cause visually impaired people problems. They are navigation or the ability to move around, and access to information (e.g. signs, posters, books, newspapers, computer screens, tickets, bank statements and train timetables). It is in the second area where this project will make a contribution.

The justification for this project lies in the need for distinguishing between items and obtaining relevant information from them. Visually impaired persons have methods of labeling items but have no means of recognising the items first. Once an item is trained into MAVVIP, it will be able to permanently recognise these objects. This will provide a visually impaired person with more independence for carrying out daily tasks.

1.3 Report outline

Chapter 2 describes the background research that was carried out for MAVVIP. This research involved studying the lifestyles and everyday problems faced by visually impaired persons. The conclusions of a survey for suggestions and feedback on MAVVIP is provided. This survey provided important clues regarding the requirements of MAVVIP. The research carried out for each of the component sections of MAVVIP is discussed. More specifically the areas of image acquisition, lighting, shape recognition, colour recognition, classification and speech synthesis are also discussed.

Chapter 3 describes the design of the main component parts of the system. Four shape recognition methods, and seven classification methods are described along with the colour recognition method chosen and the approach taken for creating speech synthesis.

Chapter 4 describes the implementation of all the unit modules described in *Chapter 3*. Experimentation was carried out on the shape recognition methods and the classification methods and the best of each method was implemented. The implementation of all the units is described as is the final stage which consisted of the assembly all the units into the final prototype.

Chapter 5 describes the final testing of MAVVIP. It was essential to thoroughly test MAVVIP for the following reasons :

- To test the accuracy and reliability of MAVVIP.
- To find operating limitations and the conditions under which MAVVIP might fail.
- To examine any problems which could arise and to test how well MAVVIP could handle them.

Chapter 6 concludes this report with the lessons learned from MAVVIP and the advantages and disadvantages are discussed. A detailed proposal for a future strategy of development for this project is provided.

2. Review of Related Work

2.1 Introduction

This chapter reviews the background research carried out for the development of the machine vision system for the visually impaired (MAVVIP).

Research was carried out in the area of visual impairments. Some general facts and statistics are provided to give an insight into the enormity of the problems faced by visually impaired persons. More specifically day-to-day problems for visually impaired persons were examined along with the methods and technologies used to combat these problems. A feasibility study was also carried out to obtain feedback from the blind community on MAVVIP.

The areas of lighting techniques, image acquisition, shape recognition, colour image processing, classification techniques and speech synthesis were also examined during this project because MAVVIP incorporated all these areas into the development of the prototype.

2.2 General information on the visually impaired community

When referring to visually impaired persons we are referring to people who have poor vision or no vision at all. To give some idea of a visual disability, statistics provided by Dr. John Gill of the Royal National Institute for the Blind (RNIB) state that of the one million blind or partially sighted people in the United Kingdom, 75% of them have sufficient residual vision to read a newspaper headline (Gill(b), 1993).

Visual impairment is one of the most frequently reported chronic health problems in the United States (NIDRR, 1992). Estimates provided by the National Centre for Health Statistics and other authorities stated that approximately 1.4 million Americans have severe visual impairment (meaning they are unable to read ordinary newsprint); half of these people are registered as legally blind (Edward’s, 1989). According to

Judith Heumann (1994) there are approximately ten million other Americans that have some visual impairment which cannot be further improved with corrective lenses.

World-wide there are almost 45 million visually disabled persons, most of whom are living in developing countries (Abror, 1992). Altogether the visually impaired community may represent a small percentage of the worlds population, but it is predicted that the number of visually disabled persons in the world will double in the next fifteen years and double again in the following fifteen years (Gill, 1992). Most of this increase is likely to be in poorer countries but the increase in first world countries will be due to the increasing number of elderly people. At present in the United Kingdom 88% of the visually disabled are over sixty years of age (Gill(b), 1993).

Most sighted people associate blind people with a white cane or a guide dog. Additional statistics provided by Dr. John Gill (1993(b),1994) show that this is not always the case. Of the one million blind or partially sighted people in the United Kingdom nineteen thousand can read Braille. This is approximately 2% of those registered as blind or partially sighted. Only thirteen thousand actually do read Braille and of these nine thousand can write Braille. There are approximately ten thousand white cane users, four thousand guide dog users and approximately three or four thousand electronic aids have been sold. The latter statistics represent less than 1% of the visually disabled persons in the United Kingdom.

The vast majority of visually disabled persons are in some way dependent on others. In first world countries most of the visually disabled are elderly. In the poorer third world countries there is less financial support to aid the blind so they have less chance of receiving proper education or employment opportunities. Mr Frank Abror (1992) states that in his home country of Ghana there are over 200,000 blind people. Over 90% of these people are illiterate and unemployed. In a country with such a large population of blind people there are only two schools for the blind. Their combined pupil population is three hundred and ten and there are one hundred and twenty writing frames between both schools. More than 90% of the students have no canes and none of the students use either tape recorders or Braille writers.

There are many problems for the visually impaired people to cope with nowadays. According to Prof. L. Kay the two most important areas where the blind still demonstrate severe limitations are mobility or the general ability to move around an environment and access to information or more specifically reading (Kay, 1984).

Firstly let us examine the problems of navigation. During the 1950's a group of enthusiasts in association with the Veterans Administration Hospital, Valley Forge developed what is now commonly known as the long cane (Kay, 1984). Although the long cane was very slow to gain acceptance it is now recognised as the primary travel aid for blind persons. It has enabled many blind travelers to travel independently through their community. Unfortunately there are many problems with this form of travel. For example the long cane does not give information about things at head height such as lorry wing mirrors and overhanging trees. Guide dogs would guide blind travelers by such obstacles but they to have there own disadvantages. They cannot be taken everywhere and they require care. Visually impaired persons need time to become familiar with a new environment, when necessary, such as new furniture in a house or a new route to work etc.. Sudden changes in environments can be difficult to deal with (e.g. roadwork's). There are other problems which arise when carrying out housework, cooking, taking medication, gardening and taking part in social activities such as sport.

Secondly, there are restrictions on information due an inability to read. It is in this area that this project will make the main contribution. There has been more success in this area in recent years than in the area of mobility. Many books, magazines and newspapers are now accessible to visually impaired people through the use of scanners, speech synthesisers, Braille terminals, reading machines, tape recorders, CD Roms, magnifying equipment etc.. Screen readers and Braille terminals have to a certain extent allowed visually impaired persons to interact with computers. Despite these successes there is still information inaccessible to visually disabled persons. Posters, signs, labels on packages such as foodstuffs and medicines, bus and train timetables are examples of some of the information that can be difficult to obtain for visually impaired persons. It is this particular problem which this project will address.

2.2.1 Alternatives to MAVVIP

MAVVIP identifies everyday items for visually impaired persons. Objects such as domestic containers, packets of foodstuffs, credit cards, bus and train tickets, denominations of money etc. are identifiable. The following paragraphs describe how visually impaired persons cope with these objects on a day to day basis.

Identifying household objects is a problem faced by visually impaired people on a daily basis. The most common method of identifying these objects is to label them with embossed labels. Plastic embossing sheets such as 3M, Dymo or Braille-On can be purchased and then Brailled on using a Braille wheel or a Perkins Brailler (Crabb; Cook; Adams; Blair; Ford, 1995; Ratcliffe; Bryant; Mendham; Ford, 1996). This method of labeling items is simple, cheap and very effective. The only drawback of this system is that every time a new item is purchased a new label is required. So it can be expensive labeling all the foodstuffs that pass through the house (Lieberg, 1996). Another alternative is to label a certain section of a press specifically for one item. But this may not work if sharing a house with others. Another method is to reuse labels. This can be done using elastic bands, magnetic labels or masking tape (Murtha, 1996; Mimms, 1996; Ford, 1996).

One of the drawbacks of this method is that each time an item is labeled the item must first be identified. In most cases this requires help from a sighted person. However with MAVVIP the item must only be identified the first time it is purchased. Once it is trained into the system no more help from sighted users will ever be required to identify that particular object. Although the proposed MAVVIP will be expensive it could possibly be a good investment, saving money on labels in the long-term and also placing the visually impaired person in a more independent position requiring less assistance from others. Another drawback of labels is that information such as ingredients, nutritional information and cooking instructions cannot realistically be kept on labels and therefore another system is required to store this information.

The problem of identifying money is another problem which visually impaired people are constantly faced with. In most countries the denominations of money are different sizes and colours so there are inexpensive aids to help identify them. In the United Kingdom and the Republic of Ireland there are plastic templates which are placed over a note and by comparing the width of the note to the template, the value of the note (Gill(b), 1993) can be estimated. In India, the system is simpler again, the folded note is stuck between finger and thumb and the amount it sticks out by can tell the denomination (Raman, 1994). However, in the USA the problem is different. All dollar bills are of the same size and colour, so this makes recognition more difficult. It is said that the US government are currently looking into a redesign of the currency in order to beat forgers (Raman, 1994), so there is hope that if new dollar bills are designed they will be easier to identify for people with visual disabilities. The newest Canadian currency is an exception as it is bar coded, (each bar is approximately 0.4cm x 1.0cm) (Trong Quoc, 1994). When Richard Nixon was campaigning for the United States Presidency, he said that if elected, he would review the currency situation and would recommend that Braille markings be included on future designs. Once elected, nothing ever came of his statements (Samwick, 1994). The existing technology for identifying dollar bills is rather expensive. According to a catalogue containing information on aids for the visually disabled released in 1993, there are three bill identifiers available for the USA (Gill & Peuleve, 1993). The cheapest of these is $395(US) and the other two are over $600(US). There is also a single device mentioned for identifying Canadian dollars which has a price tag of $250(CAN) (Gill & Peuleve, 1993).

Labels and Bar code readers do not provide a suitable solution for identifying bills. MAVVIP cannot identify United States dollars because MAVVIP relies on shape and colour information but it is possible to develop specific software to enable MAVVIP to identify American dollars.

A bar code system is another alternative. The technology for scanning bar codes has been perfected so this presents a realistic option. A bar code reader system does have its drawbacks.

- Firstly the position and orientation of the bar codes varies with each item. There is no standard for positioning of bar codes and some items have proven difficult and time excessive to identify with bar code readers (Fowle; Rundle, 1995).
- Secondly not all items contain bar codes. Some examples are denominations of money, credit cards, bus and train tickets, bank cards, identification cards etc..

According to a survey carried out by the Royal National Institute for the Blind (Fowle; Rundle, 1995) the three most important pieces of information a blind person associates with an item are the use by dates, the price and the cooking instructions. Although none of the methods can tell the use by dates or the prices directly, a machine vision system will be the most likely method to determine this textual information automatically. Due to time restraints there is no plans to develop MAVVIP to read textual information but the potential is there for further development.

2.2.2 Feasibility of MAVVIP

Just to recap on the objectives of this project. The aim of this project is to develop a machine vision system which is capable of identifying household objects. A small hand-held device would be ideal and that is the eventual aim of the project. For the moment a laboratory prototype will be set-up. It is not likely that this prototype will be transformed into a hand-held device during the course of this project.

A survey was carried out on three blind mailing lists on the Internet for information. The three mailing lists were the BLIND-L (Blind-L Mailing List, 1996), EASI (Easi Mailing List, 1996) and BLIND NEWS DIGEST(Blind News Digest Mailing List, 1996) mailing lists. The results of the survey were extremely useful and encouraging although doubt and skepticism was expressed by some. The main results of the survey are summarised as follows :

- The system should be portable.
- The system should be cheap. Suggested retail prices for a working model ranged from $200 to $1000(US).

- The system should communicate by speech synthesis.
- The system should have the facility to record information such as ingredients, cooking instructions, nutritional information, storage instructions, recipes, manufacturer information etc..
- Tell whether a book is upside down or not so they can replace it on the shelf correctly.
- A fixed camera would be useful in a store such as Hallmarks that sells postcards. The system which could describe postcard pictures and other information associated with it.
- Identifying books or postcards in a bookshop. Not all books or postcards have bar-codes and blind people like to buy books as presents for others.
- Have a stationary system in a library that students could identify books on it. It might be quicker than using a scanner.
- Make it read text. This would offer extra information such as the price and use by date, the two most important pieces of information a blind person associates with an object (Fowle; Rundle, 1995).
- Make it identify other objects such as colour coded resistors or microchips (IC's), pens and markers of different colours, the colour of items of clothing etc.

Although MAVVIP will not fit all these criteria it is hoped that MAVVIP will provide a stepping stone for further development of this idea. The three main criticisms of MAVVIP were as follows :

- It was felt that bar code readers present a more feasible alternative.
- MAVVIP is not capable of reading text.
- It is not portable.

In the authors opinion these criticisms are justified but the benefits which could be obtained from further development of MAVVIP outweigh the short term setbacks.

2.3 Lighting techniques

The first stage in a vision system is the image acquisition stage. Illumination plays an important role when capturing images. Intensity, directionality and spectral distribution are the main parameters characterising a lighting system (Hill, 1985). There are many other characteristics which must be taken into consideration when preparing illumination for a machine vision system. Lighting set-ups are application dependent (Whelan, 1992) and frontal lighting was chosen, the reasons for the decision are explained in *Chapter 4.* It is beyond the scope of this report to explain in detail all the considerations taken, but there are many excellent texts which cover the areas in considerable detail (Holland *et al.,* 1979; Schroeder, 1984; Batchelor, 1985, 1987; Novini, 1988; Uber & Harding, 1991).

2.4 Overview of shape recognition methods

This section aims to give a brief overview of a variety of different types of shape recognition techniques that were overviewed. Most of these methods were considered for MAVVIP and the reasons for and against are discussed. Our research was restricted to techniques for recognising two dimensional silhouettes of shapes. It must also be emphasised that there are a huge variety of methods available for shape recognition and it was not possible to discuss them all. Therefore this section just covers the more important techniques.

2.5 Shape recognition methods used

2.5.1 Feature extraction

The use of shape features for recognising shapes was chosen after careful consideration. The main reasons for the choice was that they are easy to compute and require a small amount of storage space. The features chosen can be recognised despite changes in scale or rotation with no extra processing overhead. The simplicity and flexibility of these features allowed for a straightforward implementation of the features chosen.

There are many feature extractors in existence which have been used for recognition of shapes. They are not responsible for approximating an object but rather provide a set of features representing an object which are concise, compact and easily accessible (Hogg, 1993). The main characteristic required is that the features must be invariant to translation and changes in size or orientation. Matching shapes is done by comparing the similarity or dissimilarity between feature values. These comparisons are known as shape measures (Ansari & Delp, 1990). Some examples of useful feature descriptors are as follows :

- Number of corners
- Number of holes
- Shape factor
- Length width ratio
- Ratio of maximum to minimum radii

The eight feature descriptors chosen and used for MAVVIP are explained in detail in *Chapter 3*. There are many other feature descriptors not mentioned here, some of which are reviewed in other texts (Ballard & Brown, 1982; Levine(a), 1985; Vernon(b),1991; Sonka *et al.*(a), 1993; Okada *et al.*, 1994).

2.5.2 Radial coding

Radial Coding is a simple method which uses several measurements of radii to represent the object (Batchelor(a), 1991). This method is explained in detail in *Chapter 3*. Several interesting variations of this method exist (He & Kundu, 1991; Sekita *et al.*, 1992; Bala & Wechsler, 1993).

2.5.3 Moments

Another popular method for shape recognition is known as moments. To overcome problems of rotation, scaling a set of central moments can be used (Vernon(b), 1991). More importantly there exist moment invariants which are linear combinations of the normalised central moments that are invariant with position, orientation and scale changes (Vernon(b), 1991) and therefore are ideal for shape recognition. Several interesting techniques have used moments in different ways to recognise shapes (Mingfa et al, 1989; Mertzios & Tsirikolias, 1993). Moments were not used directly but they were used indirectly from the image processing functions built into the Intelligent Camera to calculate the centroids and axis of least moment of inertia.

2.6 Shape recognition methods considered

This section describes several of the methods which were considered for MAVVIP but were decided against. These methods vary from simple to very complex techniques.

2.6.1 Template matching

Template Matching in its simplest form is a well known method of shape recognition. It involves moving a template over and back across an image until a match is found as shown in figure 2.1. It is very slow and it will not find a match on the image if the item being searched for has been rotated or scaled in size.

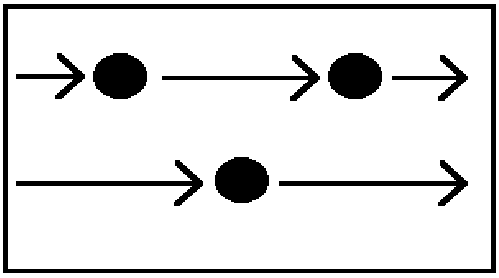


Figure 2.1 Diagram showing a circle moving across all possible positions on the image searching for a possible match.

Improvements to this method (Boyle & Thomas, 1988; Davies(a), 1990; Vernon(a), 1991; Sheela *et al.*, 1993) add to the accuracy of the method, but at the cost of processing time. Template matching can compare one template at a time so if a database of one hundred objects exists, the time required to check each template is large. Therefore it is not feasible to use this method for MAVVIP.

2.6.2 Chain coding

Chain Coding is a well known method of shape representation. Four or eight possible directions can be used for the chain codes and they are shown in figure 2.2.

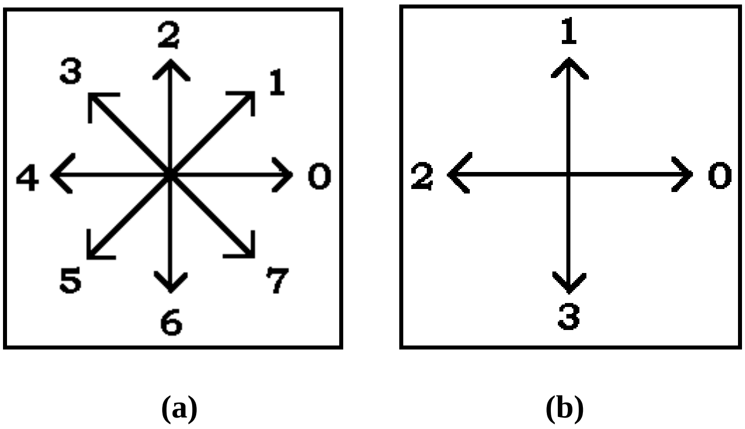


Figure 2.2 (a) Diagram showing eight possible directions for a chain code and a (b) diagram showing four possible directions for a chain code.

A starting point is picked on the edge of an object. The algorithm travels around the complete perimeter of the object and at each pixel along the way the direction of the next pixel is estimated and then stored in the chain code. Therefore the end result is a code of integers between 0 and 7. The shape can be reconstructed from this information.

An improvement to allow for rotation of shapes is required (Vernon(b), 1991). The frame-grabber for MAVVIP captures images of 256x256 resolution. Perimeters for large objects in this image could be as large as 600 or 700 pixels in length. The corresponding chain code will be too large to use as an effective representation for a single object. Another disadvantage with chain codes is that information relating to the curves can sometimes be lost when representing boundaries with chain codes (Leavers(a), 1992).

2.6.3 Ψ - s curve

The Ψ - s curve is a more detailed version of the chain code method. Ψ is the angle made between a fixed line and the tangent to the boundary at that point. It is plotted Ψ against s where s is the boundary length. Generally speaking the boundary length has to be computed and the most popular method is to assume horizontal and vertical sections of the boundary are one unit in length and the diagonal sections are $\sqrt{2}$ units in length. This is not an accurate method and more detailed suggestions are discussed in (Davies(b), 1990). For shapes with a closed boundary the function is periodic with a discontinuous jump from 2*p* to 0.

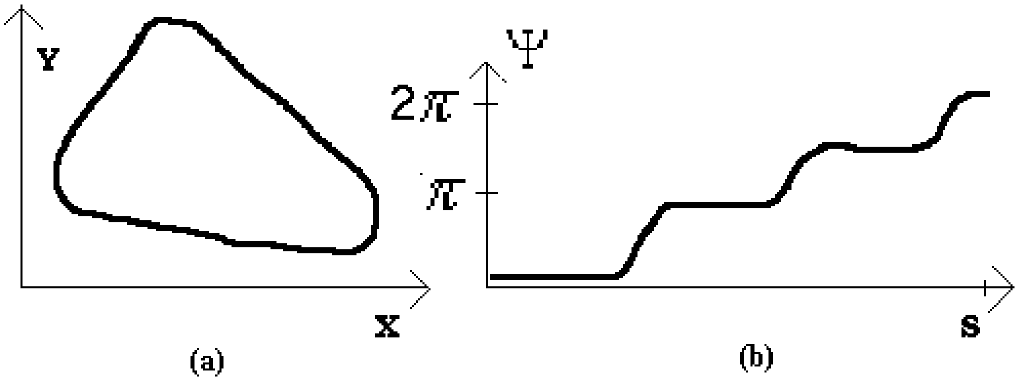


Figure 2.3 (a) Approximate triangular shape and (b) Ψ - s curve showing regions of high curvature (Ballard, 1982).

This method is further complicated by changes in scale. This can be overcome at a cost of more processing. The fact that the starting point is picked randomly on the boundary also increases the complexity because a template has to be moved along the curve until a best fit situation has been determined.

2.6.4 Low level recognition methods

Another method (Ryall & Sandor; 1989), involves checking every pixel in the image and counting the number of 2,3 and 4-neighbour pixels for each pixel. The 4-neighbours of a pixel (Batchelor(b), 1991) are shown in figure 2.4. If a pixel has four neighbours then that pixel is in the shape. A pixel with two neighbours represents a pixel on a corner and a pixel with three neighbours represents a pixel on a straight boundary part of the shape.

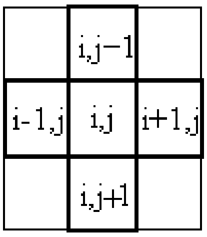


Figure 2.4 Diagram showing the four neighbourhood pixels for the pixel (i,j).

This process is carried out on both the image and the image template. A mathematical expression is used to obtain a single result which represents the degree of similarity between the two objects. A threshold value is selected and depending on whether the result is above or below the threshold, a match between the image and the image template is determined.

Several authors represent shapes as points, carefully selected on the perimeter of the shape(Lee & Park, 1989; Cootes *et al.*,1991; Cootes *et al.*, 1992) and have various methods of classifying and recognising these objects.

A polygonal approximation can be successfully used to match images. One approach (Chang & Leou, 1992) takes several images and using the polygon approximations, obtains an average polygonal approximation. The approximation creates an average of the shape models produced and this is to be used as the shape reference. Similarity measure is calculated and the threshold is set as 0.1. Weights are calculated at the learning stage so that less time is wasted at the matching stage. The author decided against these methods because it was unsure how well these methods would adapt to changes in scale or rotations.

2.6.5 Fourier transforms

Fourier transformation is one of many linear orthogonal transformations. When working in the Fourier domain, features can be extracted from the frequency components and it is possible to set up a classification system for Fourier features. Low frequency channels provide information regarding the general shape while higher frequencies indicate aspects of the boundary detail (Levine(a), 1985). Advantages include a sound theoretical base and Fourier transformations are information preserving. Derived properties remain invariant to rotations and scale changes. Several interesting techniques utilising Fourier transforms are described (Wechsler & Zimmerman, 1988; Huang & Chen, 1992) but the main reason Fourier approaches were dismissed was because of the large amounts computation involved in calculating Fourier features.

2.6.6 Hough transforms

Hough transforms are another popular method used for shape recognition. The basic concept in Hough transform is point line duality (Davies(c), 1990). This can be observed when a straight line is considered as a set of points. Take all possible straight lines which pass through one of these points. If one of these lines satisfies the other points then the equation of this line represents the line in question. This is shown graphically in figure 2.5 below.

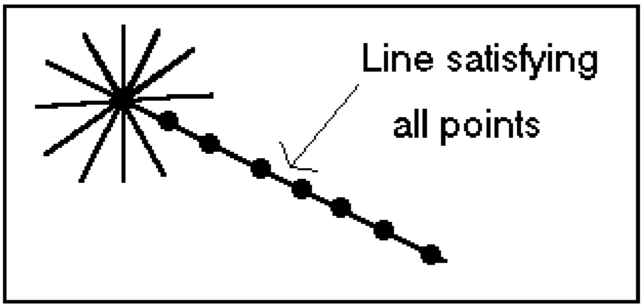


Figure 2.5 There is one line which passes through the leftmost point which satisfies all the other points. The equation of this line represents the line found.

The Hough transform can be extended to search for curves, circles, squares, corners, defects in shapes and many other shapes or features (Davies(d)(e)(f)(g), 1990). The algorithms mentioned are only capable of searching for one particular shape.

The Hough transform can also be generalised to detect arbitrary shapes. One strategy (Ballard, 1981) is to assign a reference point to the object. Then the r and f values for each point x on the boundary are calculated. These values are stored in the R-table.

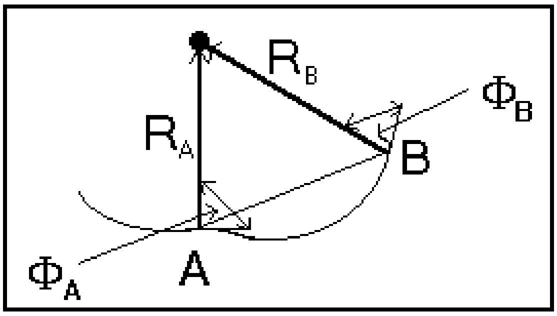


Figure 2.6 Diagram showing points A and B on the boundary of a shape and how the values for R and f are obtained for these points.

A database exists for storing the R-tables of trained objects. To match an object a single point is obtained on the boundary and the r and f values are calculated. Then the neighbouring pixels of the point are examined and any pixels likely to be on the border also have their r and f values calculated. This process continues and the results are compared with the R-tables until a match is found.

This system can be improved to take into account invariance, rotation and scaling. Also the system can be weighted so a particular part of a boundary can be detected more easily than other parts of the boundary (Ballard, 1980). These complex features come at the expense of computation time. This method was rejected on the basis that it was too complex. The R-table database required would be too large as well as the excessive computational time required processing the shape and searching for matches in the database. Other implementations of the Hough transform were also examined (Davies(d)(e)(f)(g), 1990; Leavers(a), 1992; Pao et al, 1992) along with the radon transform (Leavers(b), 1992).

2.6.7 Alternative transforms

There are also alternative transformations, the Karhunen-Loeve transformation or the singular value decomposition (SVD) which give better results than the Fourier transform at the cost of more processing time. The Hadamard transform is another common transformation which decomposes an image into a set of rectangular waveforms. Hadamard transform and most other alternatives to Fourier transform offer the advantage of being quick to compute but lack natural interpretation in terms of frequency spectrum which makes the Fourier transform easy to work with (James,1987).

Medial axis transformations (MAT) reduces a shape to a skeleton. There are several MAT thinning algorithms available (Sirjani & Cross, 1991). There are disadvantages with the medial axis transforms, as the skeleton does not give an accurate representation of the shape boundary and does not produce any unique features (Leavers(a), 1992) and the MAT is very sensitive to noise (Pavlidis, 1978).

2.6.8 Neural networks

There are many types of artificial neural networks available which serve a variety of different purposes. In the last few years activity in this area has increased exponentially (Lisboa, 1992). There are a variety of ways in which neural networks can be adapted for pattern recognition. The example of an XOR gate in figure 2.7 shows how one, two and three layer networks can be used as classifiers.

The three layer artificial neural network is commonest type and its ability as a classifier is obvious from the above diagram. A comparison was carried out (Lisboa,1992) between a neural network and six common classification methods. The test was carried out on the classification of ten digits. The neural network in each case was more accurate than the alternative classifiers.

It is also possible to successfully adapt a neural network to be invariant to translation, rotation and scale change of the input pattern within the field of view. An example of this was created by (Widrow & Winter, 1990) who constructed a pre-processor section for their neural network which computes weight matrices for the output of their system which is known as an adaline slab. Objects have to be trained in at different rotations for weights to be computed for all possibilities. Extra processing and extra training is required.

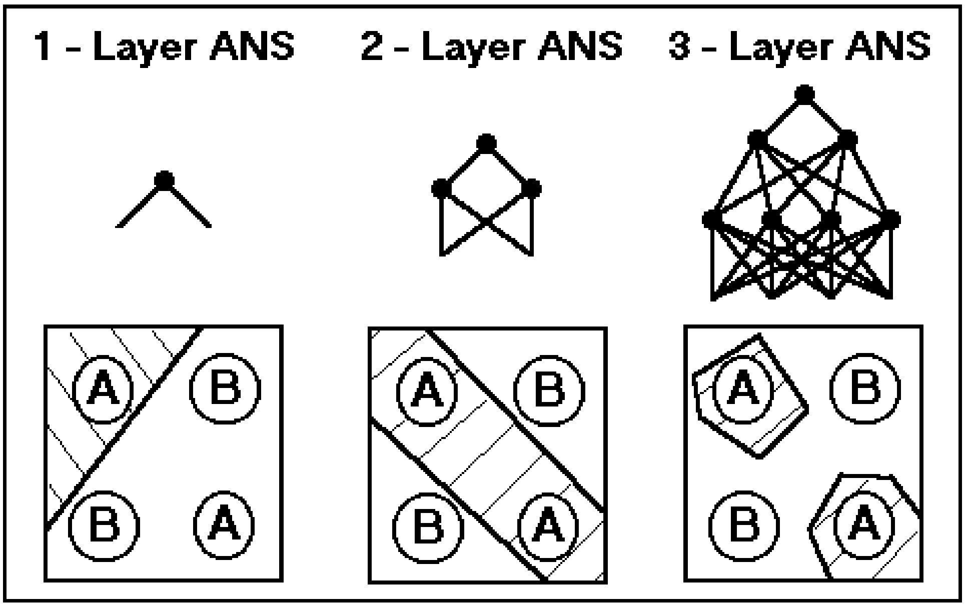


Figure 2.7 Example of three artificial neural network systems used to classify output of an XOR gate.

Suppose someone threw a small projectile to you. Most likely you would catch it. Tracking the state of dynamic systems is a nontrivial task for computers. Humans estimate the speed, trajectory and weight to catch it in real time. Human brains are significantly slower than computers for specific tasks yet humans can evaluate the situation and catch the projectile and in real time. The reason for this is parallel processing. The massive parallelism of our brain gives much processing capability in real time (Nelson & Illingworth, 1991). Neural networks work more efficiently as a parallel process, because working sequentially it is much harder to detect patterns than the situation where the entire picture is visible at once.

There are many types and variations of neural networks (e.g. back propagation, feed forward, adaptive systems, self-organising systems, hopfield neural networks etc.). Time was not available to research in detail all the different types of neural networks. The reasons why neural networks were not chosen for use by MAVVIP are explained in the following paragraphs.

With all the advantages of neural networks there are also disadvantages. Learning is difficult. There are no standards for learning algorithms or training networks. Off-line training is difficult and the number of trials needed to train a system can be quite large (Nelson & Illingworth, 1991).

Parallel processing is needed to fully exploit neural networks. Simulating this on sequential computers slows down the training and recognition of the system. It was decided that neural networks would not be exploited to there full potential unless the appropriate hardware was available to use them. Despite the speed and accuracy offered by neural networks their main drawback is in their training. This is the main reason behind the decision not to use them. Training neural networks is too slow and too delicate a process. If the prototype of MAVVIP is to be used practically, items must be trained in quickly as well as being recognised quickly.

2.7 Colour recognition

It was not the authors intention to delve deeply into the area of colour recognition. Work has been carried out previously in this area by colleagues of the author (Clarke, 1993; Batchelor & Whelan, 1993; 1995). It was hoped to incorporate some of their work into the project and build on their work.

2.7.1 Colour spaces

There are many colour spaces available to specify and name colours. RGB (Red, Green & Blue) colour space detects the red, green and blue components of an image separately. These three components can be represented as vectors in 3D space. The YIQ colour space is used by the American NTSC colour television systems. Y represents the luminance signal of the colour. The I (in-phase) and Q (Quadrature) components represent the chrominance amplitude and chrominance angle respectively. HSI (Hue, Saturation & Intensity) colour space is another common colour space which provides users with a more intuitive means of understanding and analysing colours than the RGB space. The hue of a colour is the name of the colour. The saturation of a colour refers to the degree of whiteness in a colour. The intensity refers to the lightness of the colour. These three methods are similar in that it is possible to switch from one colour space to another. The following equations show how to change from the RGB colour spaces to either HSI or YIQ colour spaces.

$$H = Cos^{-1}\left\{\frac{\frac{1}{2}\left[(R-G)+(R-B)\right]}{\sqrt{\left(R-G^{2}\right)+(R-B)(G-B)}}\right\}$$

$$S = 1-3\left[\frac{MIN(R,G,B)}{R+G+B}\right]$$

$$I = \frac{R+G+B}{3}$$

Figure 2.8 Equations to convert from RGB to HSI

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.144 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.523 & 0.312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Figure 2.9 Equations to convert from RGB to YIQ

This report is not the appropriate place to discuss colour systems or how the human visual system processes colour information. There are some good texts available which go into considerable detail on these topics (Levine, 1985, Jain(a), 1989, Travis, 1991).

2.7.2 Colour recognition hardware

The hardware available for this project is an Intelligent Colour Camera developed by A. P. Plummer of Image Inspection Ltd.. This system uses RGB inputs to perceive the colour information. One advantage of the RGB hardware is that it relies on cheap memory devices rather the custom IC’s or real-time divider circuits (Batchelor & Whelan, 1995).

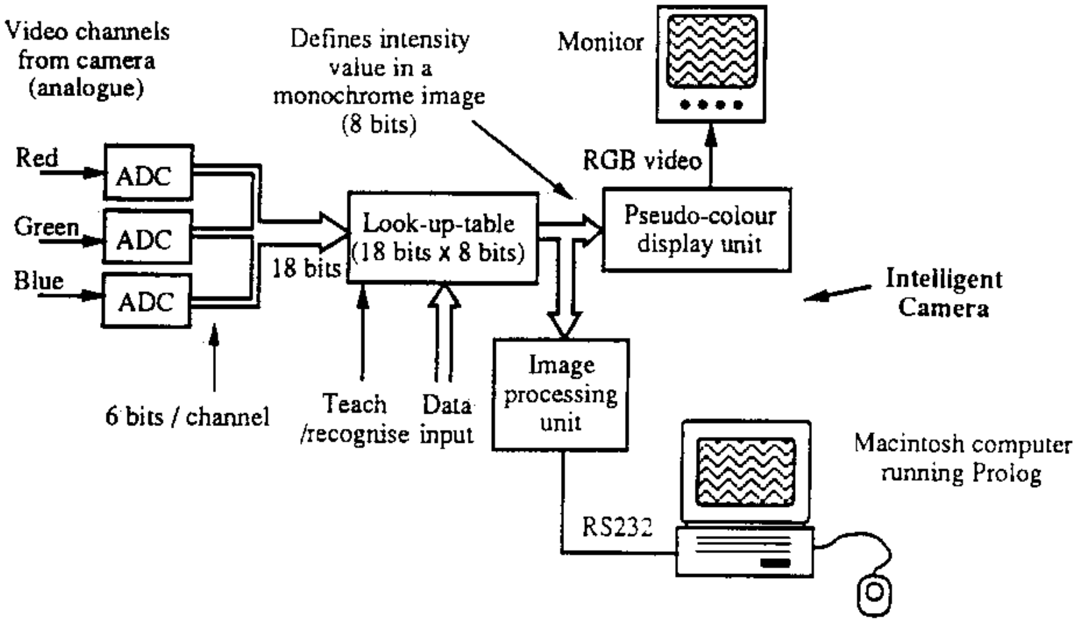

Video channels from camera (analogue)
Red
Green
Blue
ADC
ADC
ADC
6 bits / channel
Defines intensity value in a monochrome image (8 bits)
18 bits
Look-up-table (18 bits x 8 bits)
Teach /recognise
Data input
Monitor
RGB video
Pseudo-colour display unit
Intelligent Camera
Image processing unit
RS232
Macintosh computer running Prolog

Figure 2.10 Block diagram of colour camera (Batchelor & Whelan, 1995).

Figure 2.10 shows the set-up of the hardware of the colour intelligent camera which will be used. The three input channels are digitised using six bit analog to digital converters. Plotting these inputs on a three dimensional graph we obtain what is known as the colour cube. The point (0,0,0) is black and (63,63,63) is white, each an equal mix of the three primaries. The three input values are compared against a look up table to match a colour to the inputs. The colour cube is shown in figure 2.11

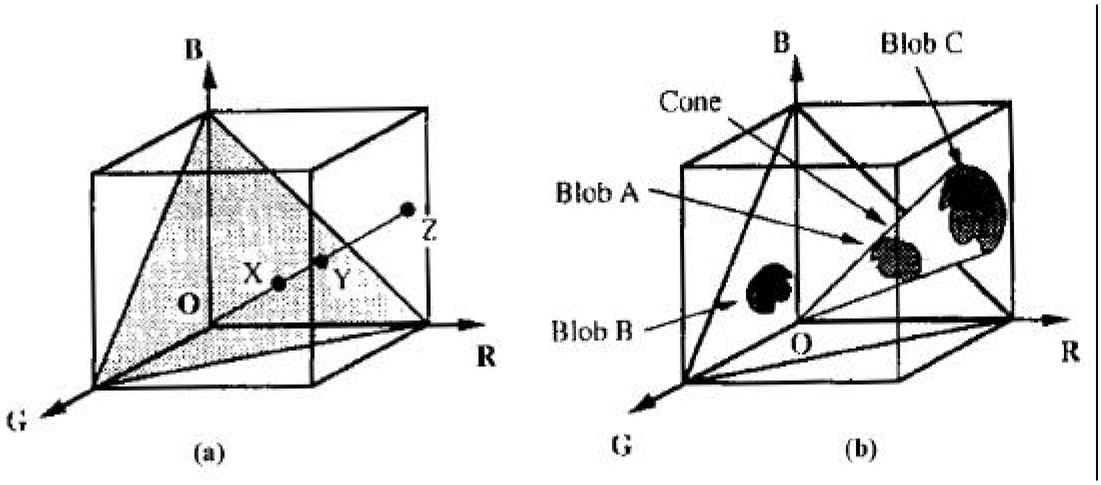


Figure 2.11 Colour cube(Batchelor & Whelan, 1995).

A plane joining the R,G and B points of the cube forms an equilateral triangle. This equilateral triangle forms what is known as a colour triangle. Each point on the triangle represents a specific hue and saturation. When the colour from a scene is taken the RGB vectors are projected onto the colour triangle creating a colour scattergram of the object.

2.7.3 Programmable colour filter (PCF)

Blobs on a colour scattergram represent the colour of an item and can be used to train the PCF(Programmable Colour Filter). This blob or blobs on the colour scattergram represent the colours contained within the image. There are many applications of Colour scattergrams. For example if some samples of ripe oranges are used to train the scattergram then this scattergram can be loaded into the PCF and can be used to check other oranges for their ripeness. Once the PCF is loaded with the scattergram for oranges, all other colours can be neglected so the colour camera can only see orange. In a case where oranges are passing along a conveyer belt only the ripe oranges can be spotted. Also the colour scattergram can be saved and used as a reference for identifying objects using their colour information.

2.8 Colour recognition method used

The author chose a variation of the colour generalisation methods used by (Batchelor et Whelan,1993: 1995; Clarke, 1993) because of the existing hardware and expertise available. Colour Generalisation involved obtaining colours from an object and plotting them on a colour triangle. This areas marked on the colour triangle were recorded and these marked areas were associated with the object as a reference. Therefore each object would have a different set of colours associated with it. The author decided to use a simpler approach to this method. Instead of taking the colours from the object and superimposing them on the colour triangle, this approach segmented the colour triangle into seven distinct sections, loaded this colour triangle into the PCF and then went and segmented the colours in the object and replaced these colours with the appropriate pseudocolours[1] from the colour triangle. In this way there

[1] Pseudocolour or false colour is created when a camera captures an image an replaces some or all of the colours on the screen with different colours. This may be done to provide a more striking colour contrast to attract the attention of the viewer (Jain(b), 1989).

is always eight colours in the object. The percentage surface area of each of the eight colours is what is used to represent the colour information of each item. This is explained in more detail in *Chapter 3*.

2.9 Colour recognition methods considered

2.9.1 Colour generalisation

An item of a certain colour is placed under the camera and its colour content is extracted and marked on a scattergram. Noise is then removed from the scattergram using common image processing techniques. This area is then dilated and then eroded to make the blob on the scattergram smooth. The blobs remaining on the scattergram are recorded and used as a reference for the recognising the object at a later stage. In the authors opinion, due to the wide variety of coloured objects used in MAVVIP, this and the following methods would be too difficult to implement successfully.

2.9.2 Colour thresholding

Thresholding can be used as a means of clustering specific colour groups of an object. The results of using thresholds to obtain clusters is similar to results obtained by using colour generalisation. The main reason why this method was not used was because of the wide variety of items which would be used in MAVVIP. Some items might produce distinct clusters and other objects might not provide any distinct clusters. Defining the clusters is a problem since a rule which identifies clusters for some items might not work for other items. There is too much inconsistency when using colour clustering or colour generalisation so these methods were decided against.

2.9.3 Colour clustering

Another more complex approach of colour clustering for image segmentation by (Celenk, 1990) was examined. This method relied on histograms to determine the prominent segments in an image. The results were impressive on complex images but offered little advantage over colour generalisation on simple scenes.

2.9.4 RGB analysis

A system developed by (Strachan, 1993) obtains the colours in a different way. Strachan developed a system to recognise fish using colour and shape. He segmented each fish into 38 sections and obtained the average R, G, and B values for all of these sections. Using a colour classification technique, the 114 variables are represented by three numbers which correspond to the front, middle and end parts of the fish respectively.

2.9.5 Similarity measures of colours

A simpler method used by (Harrell *et al.*, 1989) for automatic fruit picking was to obtain the average pixel colours of each fruit. Then the hue and saturation were calculated and if both hue (J) and saturation (r) were within a specific threshold {($J_{\min} \leq J \leq J_{\max}$) and ($r_{\min} \leq r \leq r_{\max}$)} then the fruit was ripe for picking. A similar approach used by (Gibbons & Williams, 1991) obtains the HSI values and calculates a similarity measure d using the following equation :

$$d = |i_t - i| + |s_t - s| + \min(|h_{t-}h|, h_{\max} - |h_t - h|)$$

where h_t, s_t, i_t are the reference values for hue, saturation and intensity and *h,s,i* are the measured values for hue, saturation and intensity respectively. The hue value of a pixel lies on a cyclic scale, as hue is an angular measure in colour space. The difference between two hues has to be adjusted to allow for this.

2.10 Classification methods

A typical object recognition system such as MAVVIP is a made up of a three stage process. Firstly the image acquisition stage, then the feature extraction stage and finally the classification stage. The image acquisition stage is responsible for obtaining a good quality image from which the feature extractor can extract the raw data from. The classifier is responsible for examining the raw data and identifying the object. Accurate classification is required in order to obtain a successful system. A radar system for an airport is an example of where there are lives depending on the correct classification of the radar signals.

Bayes classification sets the standard of optimum classification performance (Tou & Gonzalez, 1974, Fukunaga(a), 1990) and this method is widely used in statistical pattern recognition. While reviewing statistical pattern recognition methods, Bayes theory and the nearest neighbour method were examined along with several other methods (Ryall & Sandor, 1989; Chang & Leou, 1992). This section will review some of the statistical methods considered.

To sum up all classification techniques and categorise them, there are two distinct types. The two types of classification are supervised and non-supervised (Jain(c), 1989). We will only concern ourselves with supervised learning. Supervised learning or supervised classification comprises of two types of classification, statistical and distribution free. Distribution free methods do not require any knowledge of a priori probability functions, but rely on reasoning and heuristics. Statistical methods rely on probability distribution functions and may be parametric (e.g. Gaussian distributions) or non-parametric.

2.11 Classification methods selected for further examination

There are many methods used for statistical pattern recognition and it was not possible to evaluate all the methods that are available. During the course of the research into the classification it was observed that the most popular methods are the nearest neighbour method and the maximum likelihood ratio. It was decided to experiment with these two methods and variations of these methods to see how they perform on MAVVIP.

2.12 Classification methods examined

2.12.1 Popular classification techniques

A description of nearest neighbour algorithm is provided in *Chapter 3*. The nearest neighbour method is widely considered as a decision-making process close to one of a human being (Fukunaga(b), 1990). The nearest centroid method is also explained along with the two variations of these programs used. The Bayes classifier is explained also in *Chapter 3*. When distributions of random vectors are given the Bayes classifier is theoretically the best classifier which minimises the probability of classification error (Fukunaga(b), 1990). Theoretically for Bayes rule to provide accurate results the distributions must be normal. Although the Bayes rule has been shown to be optimal in the sense that it minimises the cost or the probability of error, it requires a large number of samples to give accurate results (Fukunaga(a), 1990). However the distributions obtained by MAVVIP are not normal because there are only a small number of samples available so non-parametric classifiers are more important in these situations (Chorafas, 1990). There are methods of estimating probability density functions when the number of available samples is small but these methods are quite complex and the author decided to use non-parametric methods instead. *Chapter 3* describes six non-parametric methods and the maximum likelihood ratio which is a parametric method. A comparison of these methods is presented in *Chapter 4*.

There are simpler processes available and the most common of these are linear, quadratic or piecewise classifiers (Fukunaga(b), 1990). These are variations of the Bayes classifier. All the variations of these classifiers have been analysed and their advantages and disadvantages have been noted. Although there are many options available for classifiers, most of these classifiers are only accurate for a small number of classes. If a new class is added to a database then a new classifier is required to take into account the new samples that have appeared. Not only that, but in some cases the complexity of the classifier is also increased with the addition of new samples. Therefore it was decided not to use complex classifiers and work with simpler classifiers because of the flexibility available (e.g. nearest neighbour and the maximum likelihood ratio test).

2.12.2 Non supervised classification

Non-supervised learning or clustering was dismissed as an option for MAVVIP. In the authors opinion it would be too difficult selecting clustering techniques which would be suitable considering the huge variety of shapes, textures and colours which would have to be classified.

2.12.3 Bayesian nets

Bayesian nets as described by Blake (1992), was in the authors opinion not a suitable method. This method relied on a item represented in a tree like structure. Each subpart was either PRESENT or NOTPRESENT. Although this system was used for describing scenes it would be possible to adapt this method for recognising shapes. A slightly different approach using trees for shape recognition (Batchelor(c), 1991) segmented an item into subparts and constructed a tree. Each node on the tree represented a specific sub-part. Semantic nets (Sonka *et al.*(b), 1993) go one step further providing relational descriptions with the subparts of an object and their descriptions. Trees were dismissed as an option because it was felt that segmenting an image into subparts was not feasible because of the wide variety of textures and colours which would be submitted for examination. Image processing techniques which work well for segmenting one particular set of colours and textures may not work so well with other textures or colours.

2.13 Speech

A method of communication with visually impaired persons was crucial for this project to succeed. Braille and speech synthesis were chosen as the two main candidates. Speech synthesis was eventually chosen, with the main reasons being it was cheaper and more practical to implement and the majority of visually impaired persons cannot read Braille. The following paragraph discusses some of the issues that were examined before deciding which of the two methods were to be used. This section discusses the options for communicating, that were available along with the advantages and disadvantages of each.

Braille has many advantages over speech so this section will highlight some of them (Lauer, 1989, Duran, 1995).

- Braille leaves the ears free. Therefore while reading Braille one can also listen to a conversation or take part in activities such as study groups etc..
- Hard copy Braille requires no machine to read it, all it needs is the human body. It can also be read on Braille displays which are similar to terminals for the sighted. Braille devices allow the user to write as well as read. In a situation where a visually impaired person wants to write a document speech allows for reading of the document only. There are several speech recognition packages available commercially but they are expensive and have not yet been perfected.
- Because Braille is usually written on paper, note taking, file making and accessibility are greatly facilitated. For many tasks Braille is preferred even though the reading speed is slow.
- Braille is preferred for non-textual work as most blind computer programmers prefer to read machine code in Braille. Equations, formulas, music and maths are nearly always more directly rendered in Braille.
- Written material sometimes loses some of its meaning when heard. More specifically precise spelling, foreign words, format and punctuation are lost for users relying on speech.
- Communications instructors have noticed that visually impaired people who have been relying on audio recordings for many years experience more deterioration in spelling skill than Braille readers.
- Deaf-blind people can use Braille but not recordings. Statistics provided by Dr. John Gill (1993(b)) state that 35% of the visually impaired persons in the United Kingdom have some form of hearing disability.

Speech has also many advantages over Braille and some of these are listed below.

- As mentioned earlier, less than two percent of those who could be registered as blind or partially sighted in the United Kingdom are capable of reading Braille (Gill, 1994). Therefore communicating solely through Braille would mean that only a small percentage of visually impaired people would be catered for.
- Braille reading is slow. The Gray-Todd Study carried out in Britain in the 1960s found people to be reading up to 200 words per minute but the average to be about

60. Visually impaired persons can absorb spoken information at an average of 120 words per minute (Gill, 1994). Many blind users can listen at accelerated word rates to the synthesised speech. When listening to familiar messages, such as prompts, some people claim that they can understand speech at word rates as high as 500 words per minute (Foulke, 1992, Blenkhom, 1994).

- A common cause of blindness in older people is diabetes, which often results in neuropathy, leaving fingers unable to read Braille efficiently. Many occupations also impair tactual perception.
- Tape recorders have become cheap and plentiful in our society. However they are not as reliable and precise for those who require them as tools.
- Recordings have become less bulky. It is now possible to fit more information onto existing disks and cassettes.
- Recordings can be hard to index, but the new tone- and voice indexing features of newer machines help a lot. With voice indexing, reference books, including recorded dictionaries and encyclopedias, have become feasible.
- Recent improvements in speech technology have meant that better quality speech synthesisers are now available. Some also have special features which are capable of adjusting the sound quality and speed (McGowan, 1994).

Some of the factors that had to be considered when choosing a method of communication are presented above. It is the authors opinion that the best solution would be to have all information communicated simultaneously on a Braille terminal and a speech synthesiser. This would leave the user free to choose from either of these facilities or perhaps to use both of them. Using the two media together is quite popular among blind people. According to Harvey Lauer (1989) the two media, Braille and speech, complement each other like the fists of a boxer.

It is intended to develop a speech synthesiser for communicating because the software is readily available at an affordable price. If at some later stage there is sufficient funds, then a Braille terminal will be incorporated into the system.

2.14 Speech synthesiser chosen

Several speech synthesisers for the Apple Macintosh were examined to see if it was possible to interface one of these synthesisers with the MacPROLOG environment. Several packages were examined (Fitzgerald, 1993; Haines, 1993; Shamsi, 1993; Takeuchi, 1993; Weidl(a)(b), 1993; Watte, 1993, Bender, 1994). After researching these packages it was decided that they could not be interfaced with MacPROLOG easily. But it was also noticed that several of these packages required a memory resident program called speech manager. On further investigation it was discovered that speech manager is a program provided free of charge by Apple Computer Inc. to help software developers incorporate speech into their software with relative ease. The licence agreement (Apple Computer Inc., 1993) and the speech manager manual were obtained and examined. The speech functions of speech manager could only be accessed through C or Pascal. MacPROLOG was capable of interfacing with C or Pascal programs. A C or Pascal program was required which could be run directly from the MacPROLOG environment and this program in turn could access the speech manager. Therefore it was decided to adopt speech manager in accordance with the licence agreements. The following two chapters describe in detail how this was carried out.

2.15 Speech synthesis methods examined

Several techniques were examined for implementing a speech synthesiser. Firstly it was decided to search for a product which could be bought for the MacPROLOG[2] environment. After extensive searching it was decided that there was no product available to suit MAVVIP’s needs so another alternative was required.

2.15.1 Developing a speech synthesiser

Creating a speech synthesiser was considered. Research was carried out to see what was involved in creating a speech synthesiser. There are several techniques for creating synthesised speech but the most common method used was a two stage process. The first of these two stages was to convert text into an intermediary state known as phonemes and the second stage was to map the phonemes to a set of sounds

[2] MacPROLOG (Johns, 1990) is a Prolog environment on Apple Macintosh Computers available from LPA Ltd.

which produced the speech. Various processes were examined (Apple Computer Inc., 1986; Ward, 1986) to see how feasible it was to attempt such a project with the time limitations available. In the authors opinion it was to big a task to undertake so this approach was disregarded.

2.15.2 Voice record/playback devices

The next option was using recorded speech. Firstly a chip voice record/playback device (Information Storage Devices, 1994) was examined. The particular device examined was quite expensive and could only contain ninety seconds of voice recording. This was not enough for the purposes of our project. It was assumed that similar devices with much greater storage would prove too expensive, so this approach was not implemented.

2.15.3 Apple Macintosh voice record/playback software

Apple Macintosh computers contain voice record/playback software. After testing it was found that this software could be used easily from within the MacPROLOG environment. Unfortunately each second of recording time required approximately twenty kilobytes of disk space for storage. This meant that a large amount of hard disk space was required in order to store a large amount of speech. This software was dismissed because using large amounts of hard disk space is not feasible. The main reason for this is that ideally it is hoped that this system if successful will be transformed into a cheap hand-held device and therefore storage space must be minimised.

2.16 Discussion

Image acquisition, feature extraction and classification are the three main stages to a pattern recognition system. Several methods were selected from the literary search from these areas mentioned and each of these methods was tested and compared in detail until the final methods were chosen. The following two chapters explain all the methods examined, how they were implemented and the results of testing and comparisons.

3. System Design

3.1 Introduction

This chapter describes the design of MAVVIP. The system is made up of six distinct parts and these are shown more clearly in figure 3.1. Several of these sections are described in this chapter and the remaining sections are discussed in the next chapter.

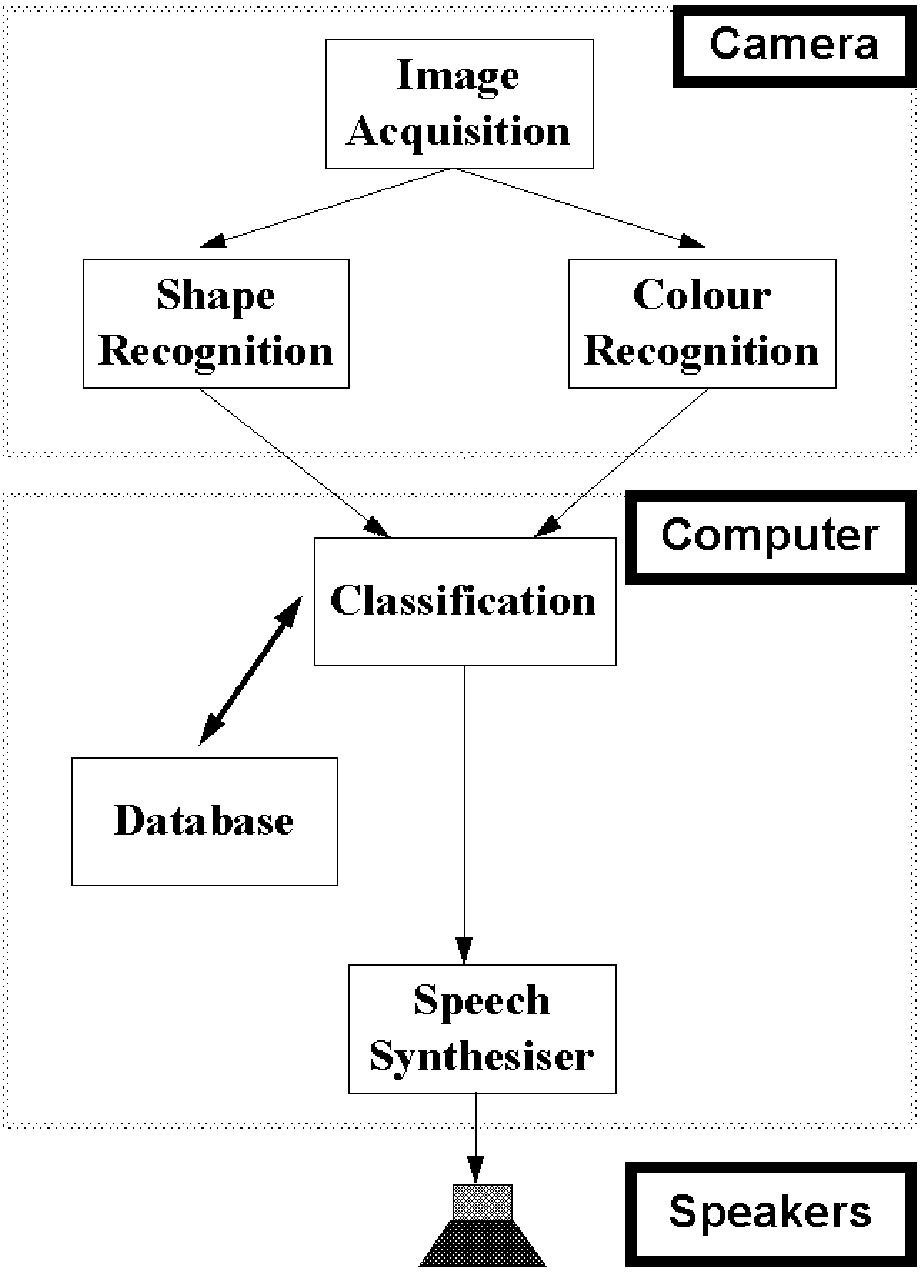


Figure 3.1 Block diagram of MAVVIP

3.2 System hardware

The prototype system was developed on an 'Apple Macintosh Centris 650' and an Intelligent Colour Camera. The camera overlooks a lighting table and there are four additional lights looking down on the lighting table from forty five degree angles. There is an additional colour monitor for observing all image processing. A diagram of the setup is shown in figure 3.2.

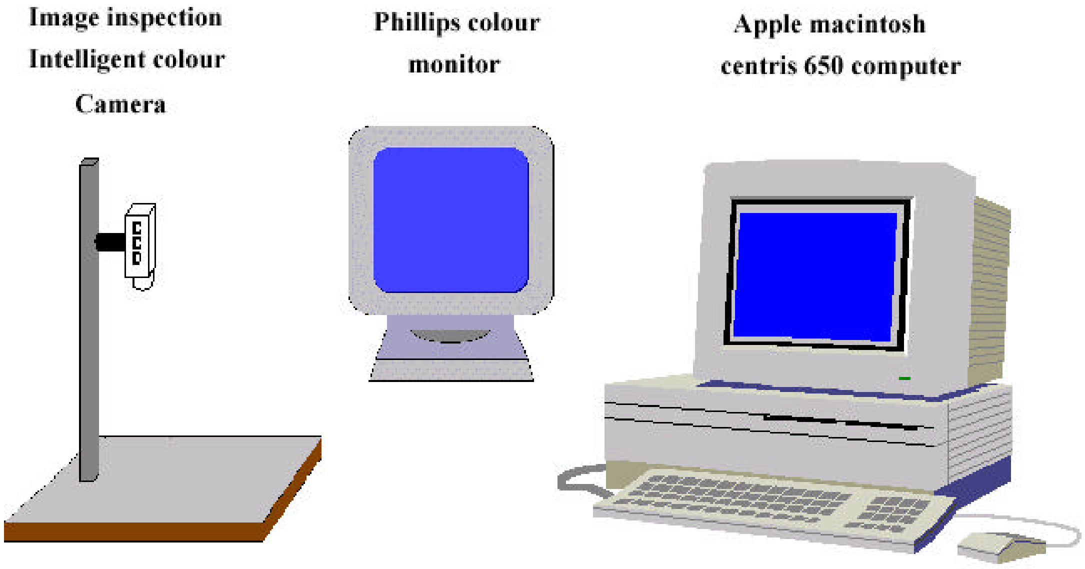


Figure 3.2 Prototype setup

3.3 Extracting information from shapes

Four methods were chosen for further examination. The following four sections describe these methods .

3.4 Feature extraction method

Feature extraction is a very common technique in machine vision systems. There are many different feature extraction techniques which can be used (Vernon, 1991; Hogg, 1993; Okada, 1994). For the purposes of this project, eight methods have been chosen to represent the object. All eight methods chosen, in theory, are not affected when the shape being examined undergoes changes in size or rotation. There are in reality limits to how much a shape can change and still be recognised. *Chapters 4 & 5* include experiments which test these limits. A brief description of the eight features are contained in the following sections.

3.4.1 Number of corners

This is a simple but effective feature for shapes. It cannot provide conclusive evidence for recognition but when combined with other feature values it becomes an important part of the recognition system. Figure 3.3 shows an example of the corners obtained by the Intelligent camera image processing software for a specific item.

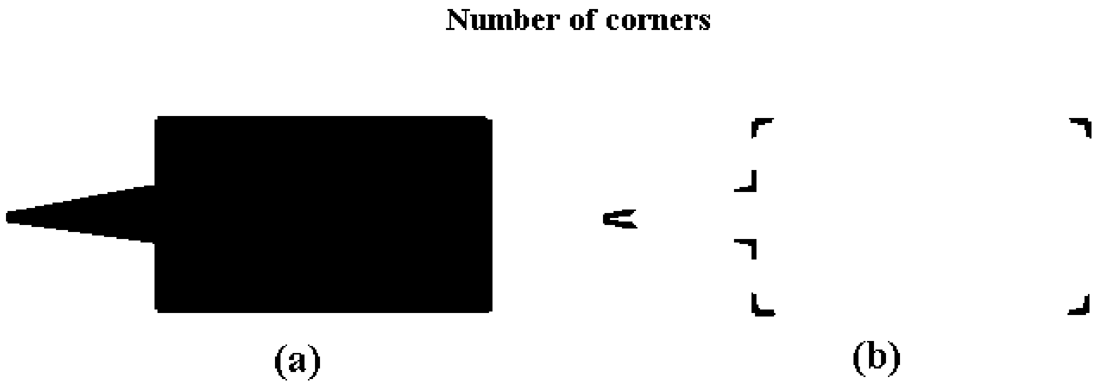


Figure 3.3 (a) The silhouette of a tin of oil and (b) the seven corners obtained for that object.

This is a reliable feature to use with smooth objects. Most of the objects trained into MAVVIP are plastic containers and have smooth perimeters. The number of corners obtained for the objects trained in rarely exceeded six corners. Occasionally there were shapes which had a rough outline and the corner function detected many corners. At different rotations this number of corners varied, sometimes considerably but the classification method was flexible enough to adapt to the changes. This feature is also unique as it is only one of two features which can have values greater than one. Six of the eight features can never have a value greater than one.

3.4.2 Shape factor

This is another very important method of extracting information. The formula for calculating the shape factor is as follows :

$$Shape\ Factor = {Area}\big/{Perimeter^2}$$

This feature was measured correct to four decimal places. The more digits in the answer the less chance there will exist two shapes having the exact same shape factor. Of the eight shape features used the shape factor in the authors opinion is the best. The reason is that it is the only feature which provides a truly unique value for every shape. Any of the other features can have an equal values for different shapes(e.g. several different objects have four corners). Shapes which have small areas and large perimeters(e.g. star shaped objects) are exceptions to the rule and tend to have similar shape factors. No star shaped objects were encountered during the experimentation stages because this experiment concentrated on common household objects and the author was not able to aquire any star shaped objects in his household.

3.4.3 Ratio of maximum to minimum radii

There is no command available within the intelligent camera for obtaining the maximum to minimum radii of shapes. The first stage of the process is locating the centroid of the shape. After this is done a circle is expanded about this point.

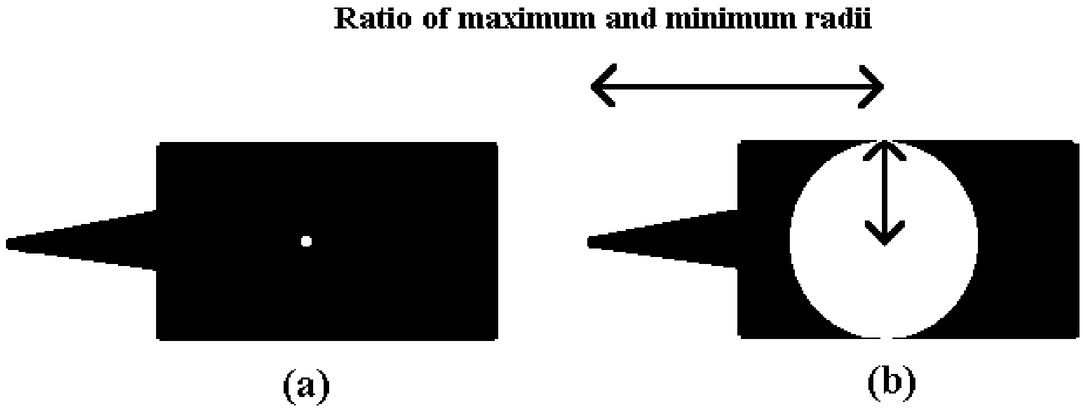


Figure 3.4 (a) Shape with centroid marked and (b) Shape with expanded circle showing the point where the minimum radius was obtained. The maximum radius is also marked in the diagram.

To the camera this circle appears as a hole in the shape as shown in figure 3.4 (a). Once the circle expands to a certain size and connects with the perimeter it is no longer a hole in the object because it has connected with the area around the shape as shown in figure 3.4 (b). At this point the radius of the circle is equal to the minimum radius of the object and is thus recorded. The circle continues to expand until the entire shape has been covered, at that point the radius is equal to the maximum radius of the shape and this is also recorded. The ratio is calculated using the equation

$$Max.\,Min.\,Ratio = {}^{Minimum\ Radius}\!\big/_{Maximum\ Radius}$$

By putting the smaller number on top, the ratio will always be between zero and one. This will be a standard setup for shapes features used during this project.

The maximum to minimum radii ratio is very similar to the length width ratio. For most objects the results from the two are very similar. The results are only different for some irregular shaped objects or more specifically objects which are wider at the ends than they are at the middle. When referring to irregular shaped objects we are assuming that triangles, squares, rectangles, parallelograms, circles and ellipses are regular objects and all other shapes are irregular. It is possible that for irregular shaped objects there will be sufficient information to distinguish them using the other seven features so this feature may not be essential. This method was implemented and re-evaluated at the experimentation stage in *Chapter 6* to check its necessity.

The aspect ratio is not a problem with this feature provided that each object is trained in several times as shown in *Chapter 4*. By training in the object horizontally and vertically in front of the camera, the two results will differ slightly due to the aspect ratio. The flexibility of the classification technique allows for successful classification and recognition despite the aspect ratio. This same condition applies to most features used in MAVVIP.

3.4.4 Number of bays (convex deficiencies)

A convex hull is plotted around the shape as shown in figure 3.5 (a). This is similar to placing an elastic band around the shape. The gaps which exist between the shape and the convex hull are referred to as bays or convex deficiencies and these bays are then counted. This is a useful feature when examining irregular shaped objects but for regular shapes such as squares, triangles and rectangles it is of little use because there are no bays. The diagram shown in figure 3.5 (b) shows the bays more clearly.

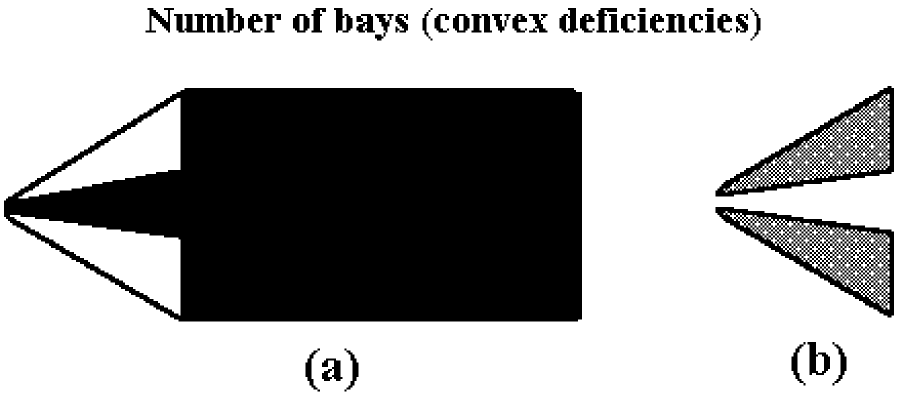


Figure 3.5 (a) Shape with convex hull plotted around it and (b) the existing bays are highlighted.

The feature is one of only two features which can have a value greater than one. The number of bays is a useful feature but it is not sufficient on its own to recognise an object. It is a useful feature for grouping objects and most common objects would fit into approximately five groups(e.g. group 1 - 0 bays, group 2 - 1 bay etc.).

3.4.5 Ratio between area of bays to area of object

Assuming that the area of the bays with most items in the household are smaller than the area of the objects themselves, then the formula for this ratio is as follows :

$$Bay\ Shape\ Ratio = {}^{Area\ of\ \ Bays}\!\big/_{Area\ of\ \ Shape}$$

This feature will produce values similar to *section 3.4.6*. The main difference between these features will be noticed when examining irregular shaped objects. This feature will be re-evaluated at the experimentation stage in *Chapter 6* to see how necessary it is.

3.4.6 Ratio of area of object to area of smallest rectangle which surrounds object

The first stage in measuring this feature is obtaining the area of the shape. The next stage is obtaining the axis of the least moment of inertia of the shape as shown in figure 3.6 (a). Once this is obtained the shape is rotated so that the axis is either horizontal or vertical. In this position the smallest possible rectangle is plotted around the shape which completely covers the shape as shown in figure 3.6 (b). The area of this rectangle is measured and the ratio between the shape and the rectangle is obtained from the following formula :

$$Ratio = {}^{Area\ of\ \ Shape}\big/{}_{Area\ of\ \ Rectangle}$$

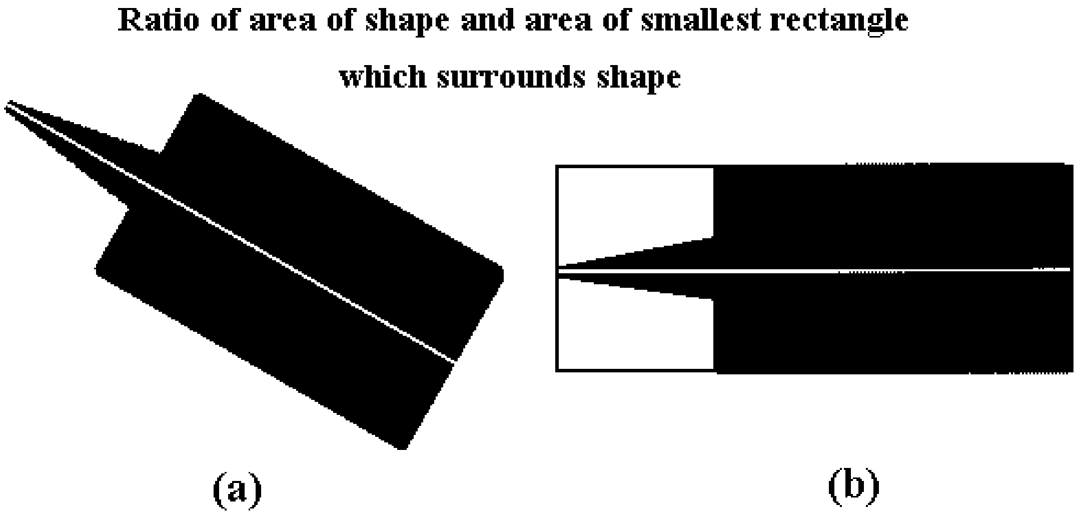


Figure 3.6 (a) The shape with its axis of least moment of inertia clearly marked on it and (b) the shape with the smallest fitting rectangle drawn around it.

As explained in the previous section this feature only differs from the previous feature when analysing irregular shaped objects and this feature will be reassessed during the final experimentation.

3.4.7 Symmetry

An example of a symmetrical shape is a shape when folded over on a certain axis exactly matches the other side. There is no specific measurement of symmetry so this method is an attempt to associate a number with symmetry. If a shape is symmetrical then the axis on which it is symmetrical, is the axis of the least moment of inertia. Therefore the first stage is dividing the object in two parts along its axis of least moment of inertia. The shape factor of each of the two parts is taken and referred to as *Shape Factor No.1* and *Shape Factor No.2* respectively. The difference between the two measurements and the average of the two measurements is calculated and the symmetry is measured as follows:

$$Symmetry \quad = \quad \left| \frac{Shape\ Factor\ No.1 - Shape\ Factor\ No.2}{\left[Shape\ Factor\ No.1 + Shape\ Factor\ No.2\right] \Big/ 2} \right|$$

With this method the more symmetrical the object is, the closer the result will be to zero. If the two shape factors are identical then the result will be an invalid number. After considerable experimentation it was decided that it would be highly unlikely to obtain two identical shape factors so this method was a reliable feature to use. If in the unlikely case the two numbers were equal then the shape would have to be retrained. However after considerable testing and usage, the two shape factors were never found to be equal.

3.4.8 Ratio of maximum width to maximum length

In order to obtain the maximum width and length of a shape the smallest fitting rectangle must be plotted. A description for this procedure was given in *section 3.4.3*. Once this rectangle is plotted then the sides of this rectangle represent the maximum width and maximum length and the ratio is calculated as follows :

$$Ratio = {}^{Maximum\,Width}\Big/_{Maximum\ Length}$$

As explained in section 3.4.3 this feature only differs from the ratio of maximum radii to minimum radii when the objects are irregular. This feature will also be re-evaluated at the experimentation stage.

3.5 Bisection method

The previous method relied on eight different features for recognising shapes. This approach takes one feature and distinguishes between shapes on the strength of this one feature. The feature used was the shape factor and the main reasons for choosing this feature are as follows :

- After considerable experimentation it was noted that the shape factor was different for almost every object unlike some of the other features (e.g. number of corners, bays).
- Easy and quick to calculate using the Intelligent Camera.

The bisection method is carried out in three stages. The first stage involves taking the shape factor of the object and storing the result in the first position of a data array. The second stage divides the shape in two along its axis of the least moment of inertia and measures the shape factor of the two individual parts. These two measurements are then entered into the data array, largest number first.

The third stage then divides each of the two shapes along their axis of least moment of inertia respectively. The shape factors for the four shapes are then calculated and sorted, largest one first. These numbers are then added to the array which completes an array of seven numbers. These seven numbers represent the shape. A diagram showing a example of a the bisection method performed on a rectangle is shown in figure 3.7.

The main benefit of this method is the speed. This method was the fastest of the four methods chosen. The reason is that in total only two important functions are used. These functions are obtaining the axis of least moment of inertia and the shape factor. Both of these functions are quick to compute when using the Intelligent Camera.

The idea is that this method relies on the objects shape factor. If there are objects which have similar shape factors then the second two shape factors are used to distinguish the objects and if they are not enough then the last four shape factors are used.

The main disadvantage with this system was that they was no way of accurately distinguishing between the two sections of a shape after they were split. Depending on what way the object was rotated at the start, the object is split and the two parts of a shape are not necessarily going to be analysed in the same order every time. It was decided to use size of the shape factor as a reference to their order but after experimentation it was found that this was not a completely accurate method of storing the right parts in the right order. This means that the order of the areas in the array is not always the same as the order of the areas in the array to be recognised. This can lead to occasional inaccurate classifications.

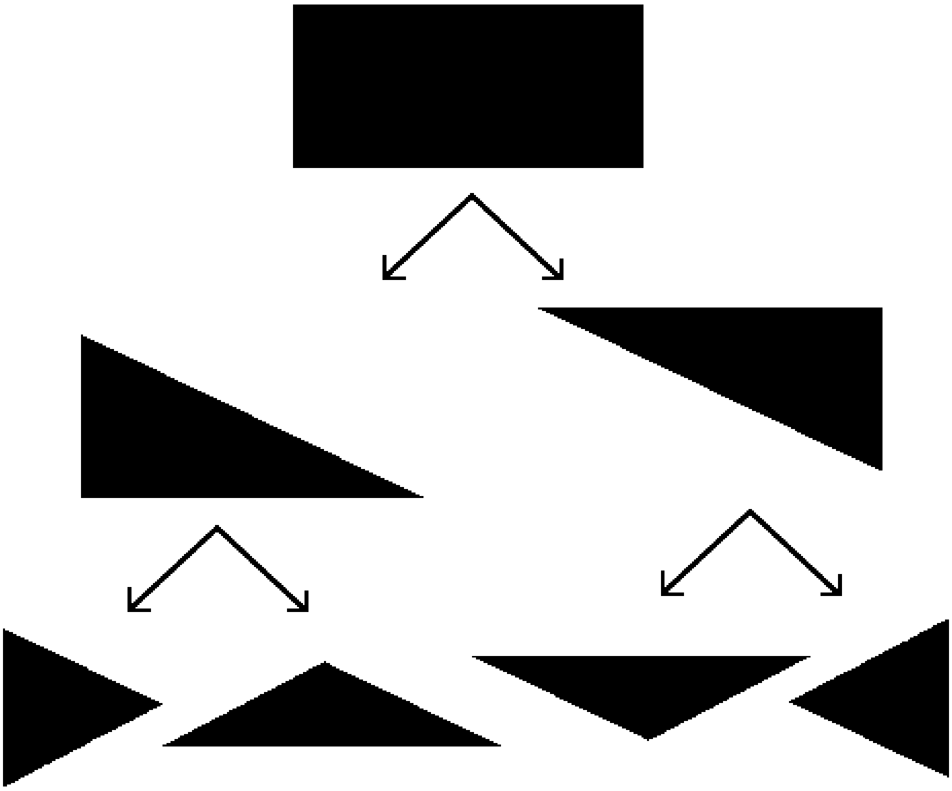


Figure 3.7 Diagram showing how a rectangle is divided up by the bisection method.

3.6 Radial coding

Radial Coding is a popular technique used for image processing. The centroid of the object is obtained and eight straight lines radiate out from this point at equal angles between them. A example showing this can be seen in figure 3.8. The areas of the eight sections are then recorded and stored and are used as a reference for recognising the object. The only problem with this method is that it does take into consideration shapes which are rotated or shapes which change in size.

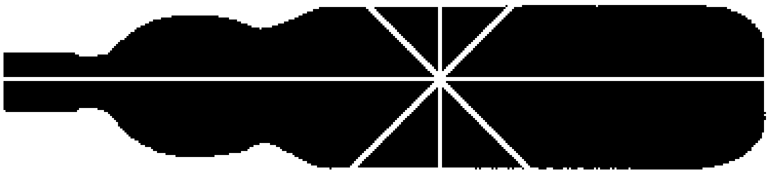


Figure 3.8 Diagram shows a bottle split into eight sections using the radial coding technique.

The solution to the problem of rotation is overcome in the following way. Using the axis of the least moment of inertia and the centroid as reference points, the shape is divided into eight sections using straight lines separated at forty five degree angles. The eight areas are obtained and sorted in order of size, largest one first. When stored in order of size the problem of shapes being rotated is solved because regardless of which way the shape has been rotated, the biggest section will always appear first in the list and so on.

To solve the problem of shapes changing in size, it was decided to store the list of areas as a list of ratios. This was done by dividing each number in the list by the biggest number in the list. This meant the first number in the list became one and the rest of the numbers lay between zero and one.

There were two other reasons why the eight area values were stored in an array, in order of size, largest value first as opposed to an array of area values in their respective order. The reasons are as follows :

- Finding a starting point is a difficult problem.
- If the object is turned upside down and examined then the order the area values measured will be backwards thus complicating the problem a little more.

The main disadvantage with this method is in situations where the centroid is near the perimeter of the object (e.g. biro). Depending on the position of the biro when the centroid is calculated, sometimes an area is measured and the value recorded and when the biro is rotated to a different position, that same area might appear as zero in size. Therefore like the bisection method, the order of the areas in the array is not always the same as the order of the areas in the array to be recognised which in turn can lead to occasional inaccurate classifications.

3.7 Rectangular coding

Rectangular coding is a variation of the radial coding method. The axis of the least moment of inertia is obtained first. This axis is divided into eight equal sections and lines perpendicular to the axis are drawn through these points. The areas of each of these sections are then measured and recorded. The eight values are then sorted and all numbers are divided by the biggest number. This is the same method of describing the numbers as discussed in the previous section.

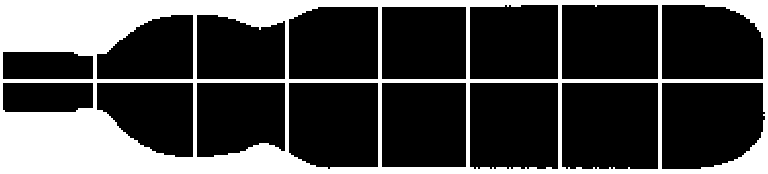


Figure 3.9 Diagram showing a bottle divided into eight sections using the rectangular coding method.

The main disadvantage with this method was splitting the length of the shape into eight equal sections. If the access length was not evenly divisible by eight, the leftover length was added on to the narrowest end of the object.

3.8 Extracting colour information from a scene

It was not the intention of the author to carry out extensive research in the area of colour recognition and classification so the approach taken for the colour recognition part of this project was a continuation of work carried out previously by others members of the Vision Systems Group (Clarke, 1993; Batchelor & Whelan, 1993, 1995). This section will provide a quick background into some of the theory behind the methods used for extracting colour information and the equipment available for implementing them.

3.8.1 Colour

The work carried out by Batchelor and Whelan concentrates on learning the characteristics of coloured information presented to it. It was their intention to train the system to recognise specific colour groups. An example is the colour group banana yellow. This group represents most of the possible shades of yellow which a banana could have. It was decided to use a variation of this method for MAVVIP. Instead of learning specific colours for each item, it was decided to set up eight colour groups and each item could be assessed by breaking down the colour content of each item into the various colour groups. Eight colour groups were required because the three primary colours, three secondary colours, black and white were necessary. The intelligent camera has a programmable colour filter as described in *section 2.7.3* which is capable of filtering out specific colours.

3.8.2 Pseudocolour

The technique used to represent the colours is known as pseudocolour. Firstly a colour triangle was drawn in RGB space. The colour triangle was divided into seven sections. Each of these sections was assigned a colour. The section in the middle of the colour triangle is assigned the colour white and this is the only section within the triangle which does not touch the boundary. The other six sections are assigned the colours red, yellow, green, blue, cyan and magenta. Black is also a colour which is recognised by the colour camera even though it does not appear within the colour triangle. The vertices of the colour triangle represent the primary colours Red, Green and Blue. In theory all points in between these three points represents a mixture of the three primary colours. The proportions which the three primary colours are mixed is directionally proportional to the distances between the three vertices. In theory all colours except black can be produced by mixing the three primary colours so the colour triangle represents all colours except black. The colour triangle used for MAVVIP is divided into seven sections and this colour triangle is fed into the programmable colour filter. When activated the programmable colour filter produces a pseudocolour image on the monitor. Pseudocolour is a false colour where all the colours in the image are replaced by the colours from the respective colour groups in the colour triangle that they belong to. The pseudocolour image is then grabbed and the eight distinct colours are measured and recorded.

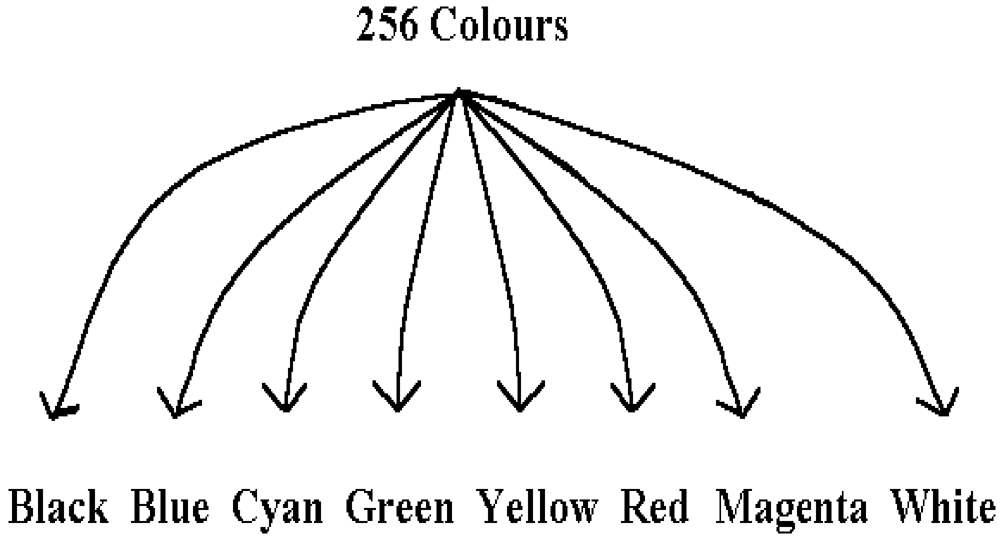


Figure 3.10 The 256 colours are all allocated into eight colours groups.

3.9 Classification techniques

There were seven classification methods chosen for further experimentation. The most suitable of these methods would be applied to MAVVIP. Four of the methods chosen are well known classification techniques and the other three methods are variations of these methods. The following sections explain these classification methods in more detail.

3.9.1 Method 1 - A simple nearest neighbour approach

This method is the simplest of the classification methods. This method was first implemented in an earlier program which identified common shapes (e.g. circles, squares, ellipses, triangles, and rectangles) (Molloy *et al.*, 1994). Although this is a simple classification method it was decided to experiment with it further because it would serve as a useful reference when comparing the results to more complicated methods such as the nearest neighbour classification and the maximum likelihood classifier.

$$\boxed{\begin{array}{l} shape\_data('\,ITEM\_1', A_{71}, A_{72}, A_{73}, A_{74}, A_{75}, A_{76}, A_{77}, A_{78}) \\ shape\_data('\,ITEM\_1', A_{81}, A_{82}, A_{83}, A_{84}, A_{85}, A_{86}, A_{87}, A_{88}) \\ shape\_data('\,ITEM\_2', B_{11}, B_{12}, B_{13}, B_{14}, B_{15}, B_{16}, B_{17}, B_{18}) \\ shape\_data('\,ITEM\_2', B_{21}, B_{22}, B_{23}, B_{24}, B_{25}, B_{26}, B_{27}, B_{28}) \end{array}}$$

Figure 3.11 A small section from the Prolog database. This represents how the shape recognition data is stored within the database.

To illustrate this method, assume that the parameters in the above section of the database are results from the feature extraction method of shape recognition. The classification test is performed eight times, once for each parameter in the shape data. For the purposes of demonstrating this method, one of the eight classifications will be shown. Drawn graphically the first parameter of each item can be seen in figure 3.12.

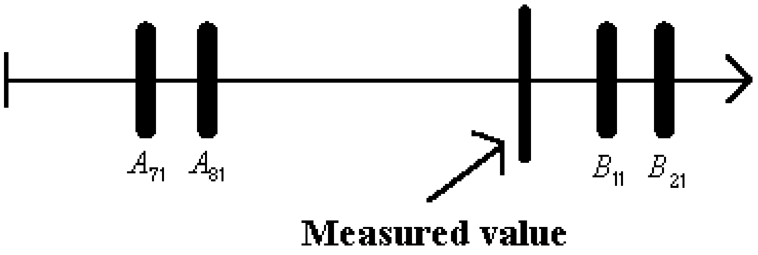


Figure 3.12 Parameters values for four points represented on a straight line.

When an object is measured the corresponding parameters are placed on their corresponding graphs. It is highly unlikely that two measurements when placed on the graph will be identical. Therefore in order to match a measured item with items already on the graph a tolerance must be allowed. This means that space is allocated either side of the measured item on the graph and any other items on the graph which lie within this region are possible matches with the measured item. This is shown more clearly in figure 3.13.

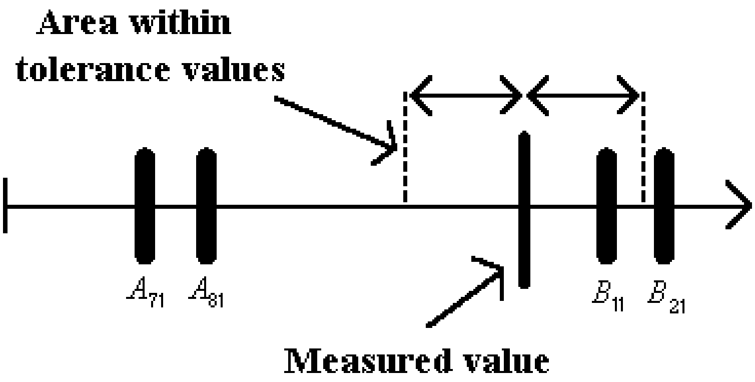


Figure 3.13 Measured value and region within tolerance

Parameter values will vary considerably with every object so a standard method for choosing tolerances is required. The method decided on was to take the maximum value plotted on the graph and assume this is a standard reference. Then all tolerances will be a percentage of that value. One major disadvantage with this method is that the tolerances will not remain the same permanently. When new items are trained into the database the tolerances might increase. Therefore the more items added to the database, the bigger the tolerances get and the less accurate this method of classification becomes.

3.9.2 Method 2 - Method relying on means and standard deviations

In order to operate this method each item must be trained into the database more than once. Figure 3.11 shows a sample piece of code from a Prolog database which shows the format of items trained in more than once.

Therefore eight tests are required for the data shown in figure 3.11. The reason items are trained in more than once is that the data can then be expressed in the form of a Gaussian distribution as shown in figure 3.14.

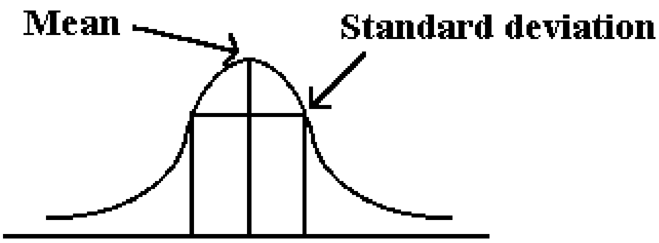


Figure 3.14 An ideal Gaussian distribution.

Gaussian distributions can be setup for each set of parameters and a mean and standard deviation can be associated with each Gaussian distribution. Therefore when an item is placed under the camera, the measured values obtained can be plotted on the same graphs as their respective Gaussian distributions. A matching item is found when the measured value lies under the boundary of a Gaussian curve as shown in figure 3.15.

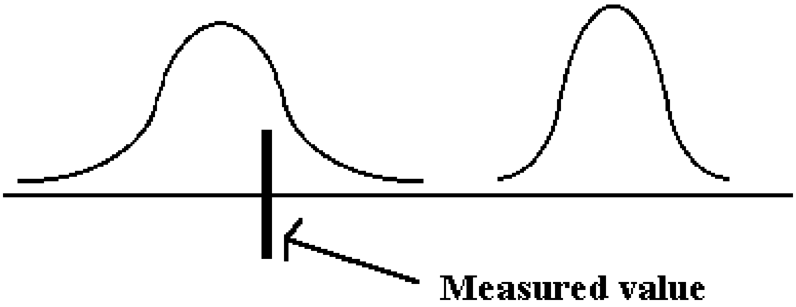


Figure 3.15 A measured value is marked on the axis and it is obvious that it belongs to the group whose Gaussian distribution is on the left hand side of the graph.

The main problem with this using Gaussian distributions is determining the edges of the boundary. The edges of a Gaussian curve can be varied so it is effectively acts as a tolerance. The standard deviation is a useful reference associated with Gaussian curves so the edges of the curve will be referred to in terms of standard deviations from the mean. According to J. R. Parker (1994), 68% of all samples lie within one standard deviation of the mean.

The formula used for the standard deviations is as follows :

$$s = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \bar{x})^2}$$

3.9.3 Method 3 - Nearest neighbour method performed eight times in one dimension

The nearest neighbour method is a popular classification technique. If each item in the database has one value associated with it then the diagram in figure 3.16 demonstrates the nearest neighbour method in one dimension. The distances between the measured value and all the other points are calculated and the shortest of these distances means the most likely match for the object.

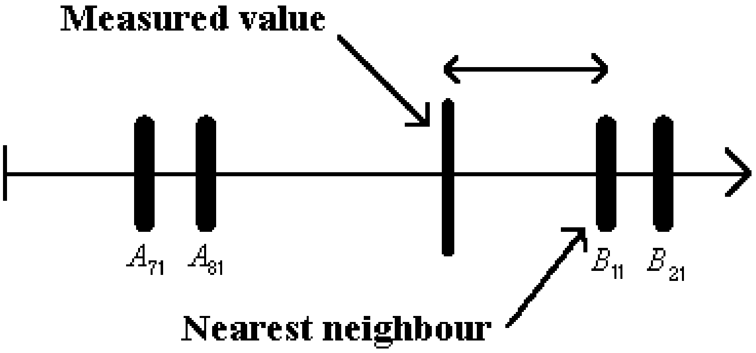


Figure 3.16 The measured value is plotted along an axis which contains four values. The measured value lies closest to the value B_{11}. Therefore B_{11} is the nearest neighbour.

In reality each item in the database will have approximately eight associated data values, so this test is performed eight times, once on each set of data. For example if the number of corners of an object is obtained then this is compared with the number of corners of each of the objects in the database. Similarly the shape factor is compared with all the shape factors stored in the database and so on.

3.9.4 Method 4 - Nearest centroid method performed eight times in one dimension

A similar method to the nearest neighbour method is the nearest centroid method. Instead of calculating the distances between a measured item and all other points, the centroid is obtained for the data associated with each item which has been trained into the database more than once. This reduces the quantity of measurements taken. Only measurements between the measured value and the centroids are required. This is shown graphically in figure 3.17. Like the previous method this is performed eight times in one dimension.

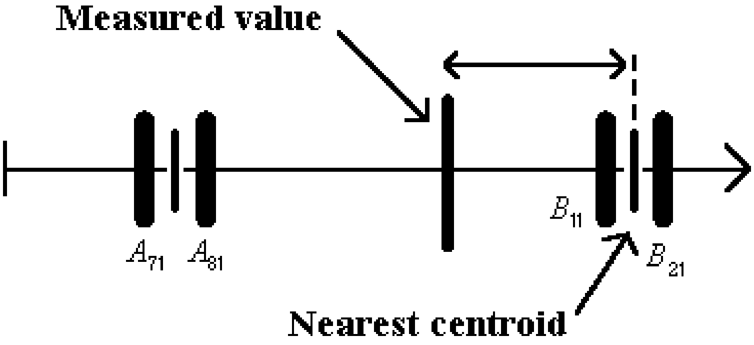


Figure 3.17 The measured value is plotted along an axis which contains four values and the two centroids representing those values. The measured value lies closest to the centroid representing B_{11} and B_{21}. Therefore the *B* centroid is the nearest centroid.

3.9.5 Method 5 - Nearest neighbour method

If there are two values of information associated with each item then the nearest neighbour method becomes a two dimensional problem as shown in figure 3.18.

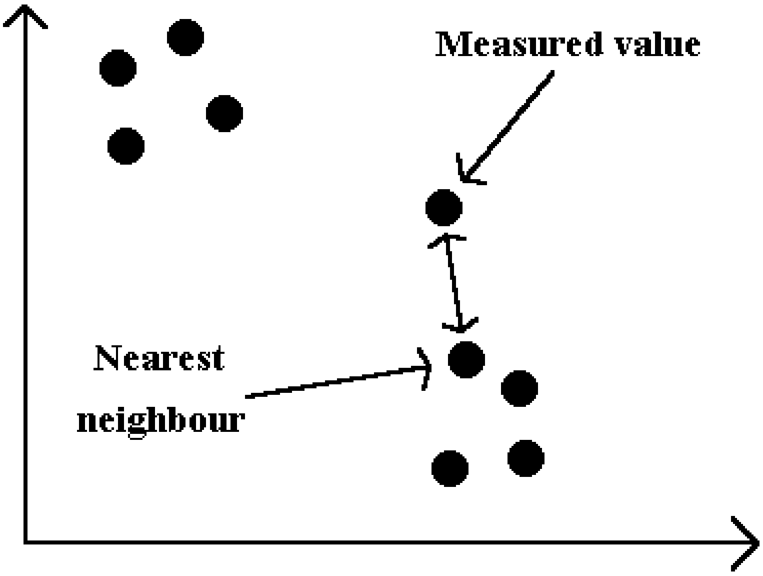


Figure 3.18 This is another example of the nearest neighbour method except it is in two dimensions.

Similarly if there are eight pieces of information associated with each item then the nearest neighbour problem is an eight dimensional problem. The nearest neighbour method was used in two different ways during the classification experiments. The first variation of the nearest neighbour method splits the problem into eight individual problems as described in section 3.9.3. This method is essentially the correct implementation of the nearest neighbour method.

3.9.6 Method 6 - Nearest centroid method

This method is the nearest centroid method and explained in section 3.9.4. The difference between the nearest neighbour method and the nearest centroid method is discussed in section 3.9.5.

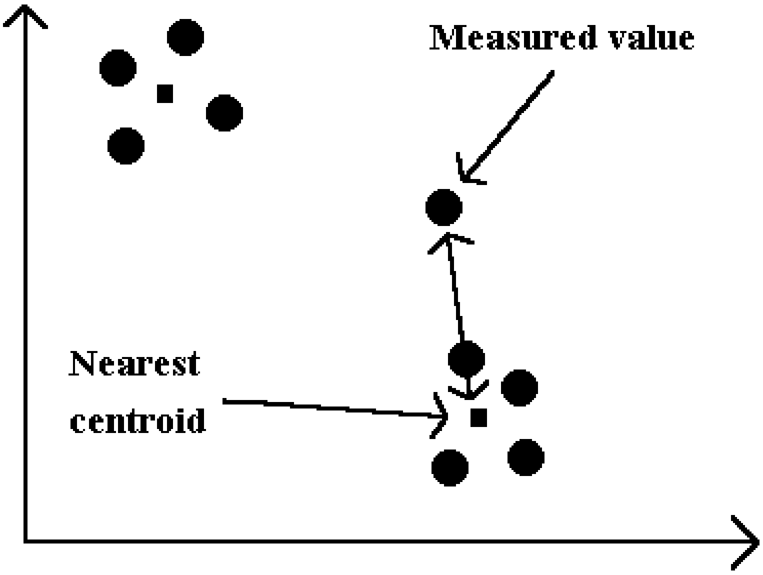
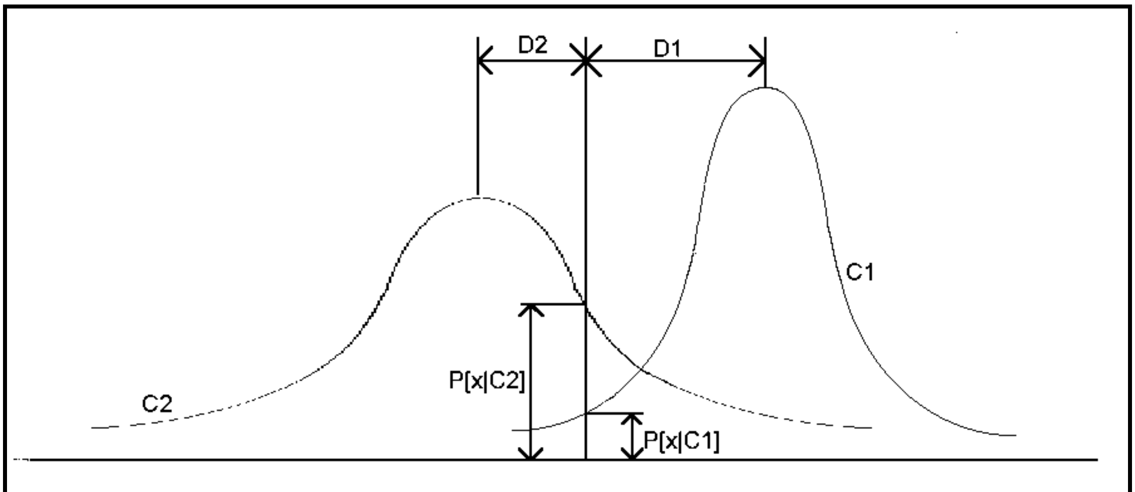


Figure 3.19 The nearest centroid method in two dimensions.

3.9.7 Method 7 - Maximum likelihood theorem

The final method and the most complex of the methods implemented was the maximum likelihood theorem. This is a measure of probabilities that an object of a given class will have a given feature. The measured value is marked on the same axis as the Gaussian distribution graphs. Unlike the nearest neighbour method the most likely match is not the nearest mean value.



P[x|C2] = Probability that a feature x will occur given that the object belongs to class 2.

P[x|C1] = Probability that a feature x will occur given that the object belongs to class 1.

Figure 3.20 The measured value between two distributions with the important features marked.

The probability, P[x|C1] or P[x|C2], which has the greater value, determines which Gaussian distribution the measured value is more likely to belong to.

3.10 Speech synthesis

The Speech Manager package was created by Apple Computer Ltd. to provide Apple Macintosh users with better speech facilities. Speech Manager is accompanied by the file Macintalk Voices. This file provides the Speech Manager with nine additional simulated speech voices. The Speech Manager package also includes another text to speech program known as PlainTalk Text to Speech and the four voices which accompany it. PlainTalk Text to Speech will be discussed in the next section.

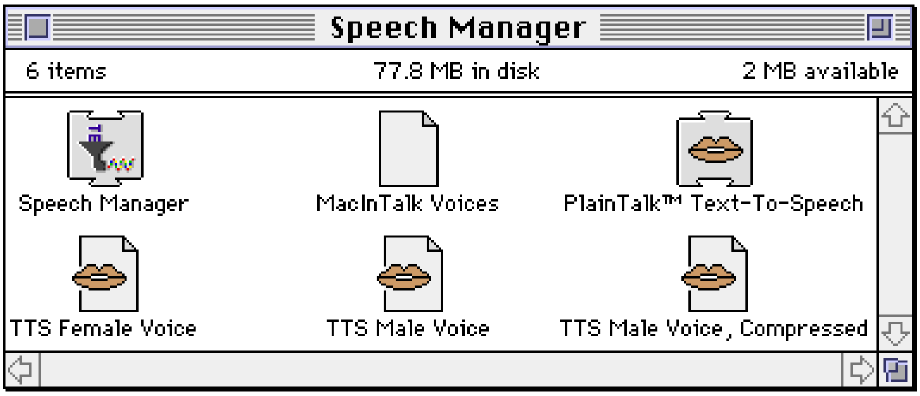


Figure 3.21 List of files contained in the Speech Manager Package.

Speech Manager is a memory resident program which is responsible for passing text from applications to the speech synthesiser where the text is converted into sound and sent to the speakers as the actual audio output. Speech Manager will only run on Macintosh computers which have System 6.0 or better and it requires 150 Kbytes of system memory. Speech Manager is not a stand alone application, it is a collection of PASCAL functions which can be called from within the PASCAL and C environments. It is up to an application to call these functions to create the required speech. With these functions, it is possible for a software developer to create customised speech packages.

On a simple level, speech synthesis from text input is a two stage process. Firstly the text is converted into phonemes. The second stage corresponds to the resulting sequence of phonemes being converted into audible sounds by mapping of the individual phonemes to a series of wave forms, which are sent to the sound hardware to be played. By default, speech synthesisers expect input in normal language text. However the input mode in Speech Manager can be controlled by embedded speech commands. The commands are explained in more detail in (Molloy *et al.*, 1994). These commands enable the user to input normal text or phonemic text or a combination of both directly into the speech synthesiser. The sample code regarding phonemes and pronunciations, provided in McGowan (1994) refer to the Speech Manager voices, they do not necessarily apply to the PlainTalk voices. The reason for this is that the PlainTalk Text to Speech package is more advanced than Speech Manager so it does not make as many mispronunciations.

3.10.1 PlainTalk text to speech

The PlainTalk Text to Speech program is an additional package which provides Speech Manager with extra voices. PlainTalk Text to Speech is now standard with new Apple Macintosh Computers. PlainTalk contains male and female voices, compressed and uncompressed, which produce better quality speech than the Speech Manager voices. This program requires System 7.0 or better to function correctly and also, PlainTalk requires 600 Kbytes of system memory. PlainTalk can be accessed by the same functions that call Speech Manager routines.

PlainTalk also contains a more sophisticated text to phoneme converter. An example of PlainTalk's speech generation is the sentence "He earned over £2,000,000 in 1990". It would be preferable to say "He earned over two million pounds sterling in nineteen ninety." rather than "He earned over pound-sign, two, comma, zero, zero, zero, comma, zero, zero, zero in one, nine, nine, zero.". During the text to phoneme stage, number strings, abbreviations and special symbols must be detected and converted into appropriate words before being converted into phonemes. To produce the desired spoken output automatically, knowledge of these sorts of constructs is built into the synthesiser.

The phoneme to sound stage is also complex. Phonemes themselves are not enough to describe the way a word should be pronounced. For example the word "object" is pronounced differently depending on whether it is used as an object or a verb. When it is used as a noun, the stress is placed on the first syllable. As a verb, the stress is placed on the second syllable. In addition to stress information, phonemes must often be augmented with pitch, duration and other information to produce intelligible, natural sounding speech.

3.11 Discussion

This chapter described in detail the shape and colour recognition, classification and speech synthesis sections of MAVVIP. Four methods of extracting information from shapes were discussed. The implementation and initial experimentation of these four methods is described in *Chapter 4*. A method for extracting colour information from objects was described also. Implementation and initial results of this method are provided in *Chapter 4*. Seven classification techniques were described and again the implementation and comparison stages are described in the next chapter. Some general information was provided about the Speech Manager package and the implementation of the speech synthesis is described in the next chapter.

4. Implementation of MAVVIP

4.1 Introduction

This chapter describes the implementation of all the individual modules of MAVVIP. Several methods for shape recognition and classification were implemented. Experimentation was carried out to select the best of these methods. The results of the experiments to decide the best shape recognition and classification methods are shown in this chapter. A detailed description of how these modules were combined carefully into one system to produce a working prototype is described in this chapter. The modules will be described in the order which they are used within the system.

4.2 Lighting set-up and image acquisition

The implementation and accuracy of MAVVIP depended on the lighting set-up. In the authors opinion it would be hard to develop shape and colour recognition software if there were errors which were a result of bad lighting. By eliminating possible errors from sources such as bad lighting it is easier to analyse the effectiveness of the recognition software. Due to the time scale involved with this project it was not intended to divert excessive amounts of time experimenting with lighting arrangements.

After experimenting with several lighting setups, the setup chosen was four tungsten lights, two on either side of the light table shining down on the light table from an angle of approximately forty five degrees. This set-up provided adequate light to extract colour information from the scene. A light table was also used in conjunction with the tungsten lights. Objects were placed on the light table and this allowed for an accurate silhouette of the shape to be obtained. Although lighting is a major problem when considering the feasibility of this project, it was not our priority. The main objective was to develop software for recognising objects.

4.2.1 Background lighting

Background lighting can effect the analysis of shape and colour so attempting to reduce the deviations of background lighting as much as possible meant reducing the error rate of the overall system. More specifically varying amounts of light during the day and at night added an extra variable in the recognition procedure. It was decided to minimise the use of daylight in the system. Therefore the strategy decided upon was to add as much artificial light to the lab as possible. The more artificial light in the laboratory meant the less dependent the system was on daylight. This in turn meant that daylight would have less of an effect on the shape and colour analysis which in turn would improve the accuracy of the system. In order to maximise the artificial lighting in the laboratory it was decided that when experimentation was being carried out that all lights in the lab would have to be on as well as the lights in the lighting setup of MAVVIP. Also all the window shutters in the laboratory were closed to shelter MAVVIP from the immediate effects of direct sunshine.

4.2.2 Image acquisition

This is the smallest module within MAVVIP but it is crucial to the overall performance of the system. The image acquisition program is responsible for obtaining an accurate outline of the shape of the object and obtaining an accurate colour representation of the object. The shape is done by obtaining a silhouette. The intelligent camera is located on a stand overlooking a light table. The camera grabs an image and thresholds it to obtain the shape outline. A colour triangle is loaded into the programmable colour filter of the Intelligent Colour Camera. The colour filter then extracts pseudocolour information from the object. The colour information extracted relies on the colour triangle as a reference. This was prepared after testing it on an ad-hoc basis to see which level of saturation provided the best colour information. The final colour triangle is shown in figure 4.3. The images in figure 4.1 show the quality of two sample shapes obtained from the image acquisition program.

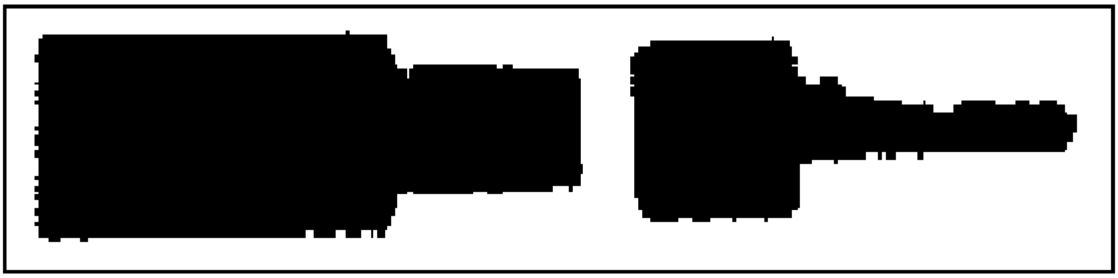


Figure 4.1 Shape templates obtained for bottle of *tipex* and *key*. The images are approximately to scale.

The level of thresholding was done after trial and error testing. An example of a *moro bar* captured in real colour and pseudocolour is shown in figure 4.2.

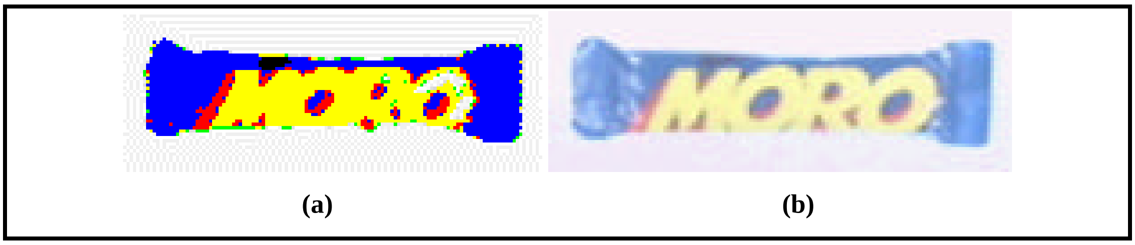


Figure 4.2 (a) *Moro bar* in pseudocolour and (b) a real colour image of the same moro bar. Due to hardware restrictions it was not possible to grab pseudocolour images directly. Therefore grey scale images were grabbed and converted into pseudocolour images manually. As a result of this all pseudocolour images in this report are close approximations of the real pseudocolour images.

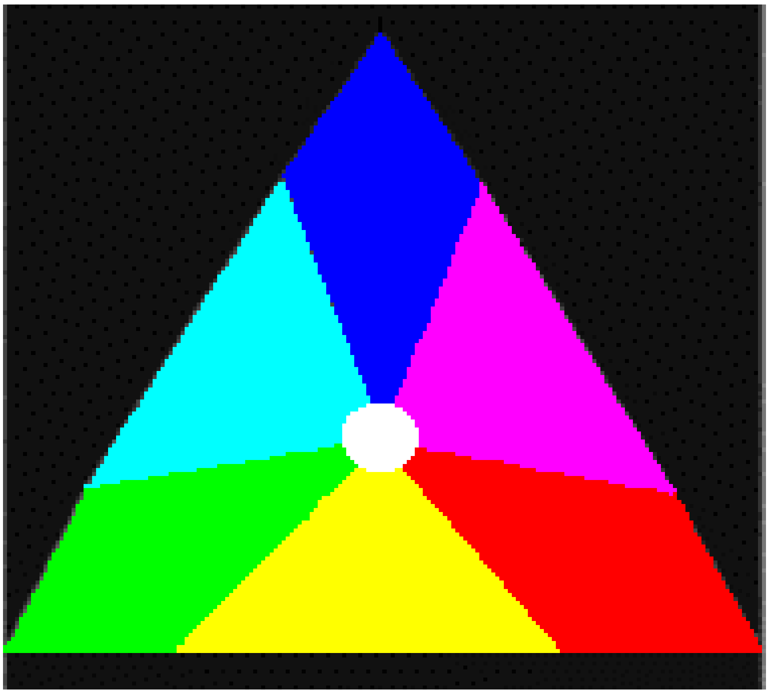


Figure 4.3 Colour triangle used to program the colour filter in the intelligent colour camera.

4.3 Implementation of shape and colour extraction software in prolog

Four methods of shape extraction and a method of colour extraction were described in the previous chapter. In order to compare these methods at a later stage, they first have to be implemented so objects can be trained into the system and the raw data can be recorded for further analysis.

4.3.1 Prolog

Prolog was chosen as the main language for implementing this project. C was also used to supplement the Prolog especially when it came to writing mathematical routines.

Prolog offers several advantages over lower level languages such as C, Pascal and Basic. Being a higher level language, Prolog is an ideal artificial intelligence language. Prolog also allows backtracking in programs and the database facilities within the Prolog environment are excellent.

A secondary package known as Prolog+ developed by Bruce Batchelor was also available which provided an interface to the intelligent camera along with many image processing functions. This package proved to be a useful stepping stone and was customised by the author. All non essential functions were removed and some new functions specific to the authors needs were added.

4.3.2 Implementation of shape extraction techniques

Implementation was a straight forward procedure but a few problems arose. It must be emphasised here that the experimentation described in this chapter is preliminary experimentation to choose the shape extraction and classification methods. More detailed experimentation on the final prototype is described in *Chapter 5*. Some of the problems encountered during implementation are as follows :

- The **feature extraction** method created no real problems but the third test of the eight was particularly slow. Expanding circles on the monitor was very slow and there are several ways which the speed of this third test could be improved.
- The **bisection method** obtained seven shape factors and placed these numbers in an array. When objects appear at different rotations there are slight changes in shape factor values. These deviations in values meant that the order of the measured values might not be the same order as the values for the same object stored in the database.
- The **radial coding** method produced a problem. When small or very thin objects were rotated the centroid position varied slightly. The smaller the object the more inaccurate the result will be. Similarly as before an area might be measured as zero in size and when rotated and measured again might have a small area. As the order of the array is sorted according to size this effect reducing the accuracy of classification.
- The **rectangular coding** method caused slight problems. The axis of the moment of inertia was split into eight sections. If the length of this axis was not a multiple of eight then this presented a problem. It was decided to add the surplus piece to the section at the narrowest end of the object. This would lead to extra inaccuracy with one of the eight section measurements. The smaller the object the more inaccurate the results will be.

Once the four shape recognition methods were implemented the twelve shapes in figure 4.4 were trained in. The necessary information was extracted and stored in the database. For consistency while training objects, each object was trained in eight set positions. The eight positions are shown in figure 4.5.

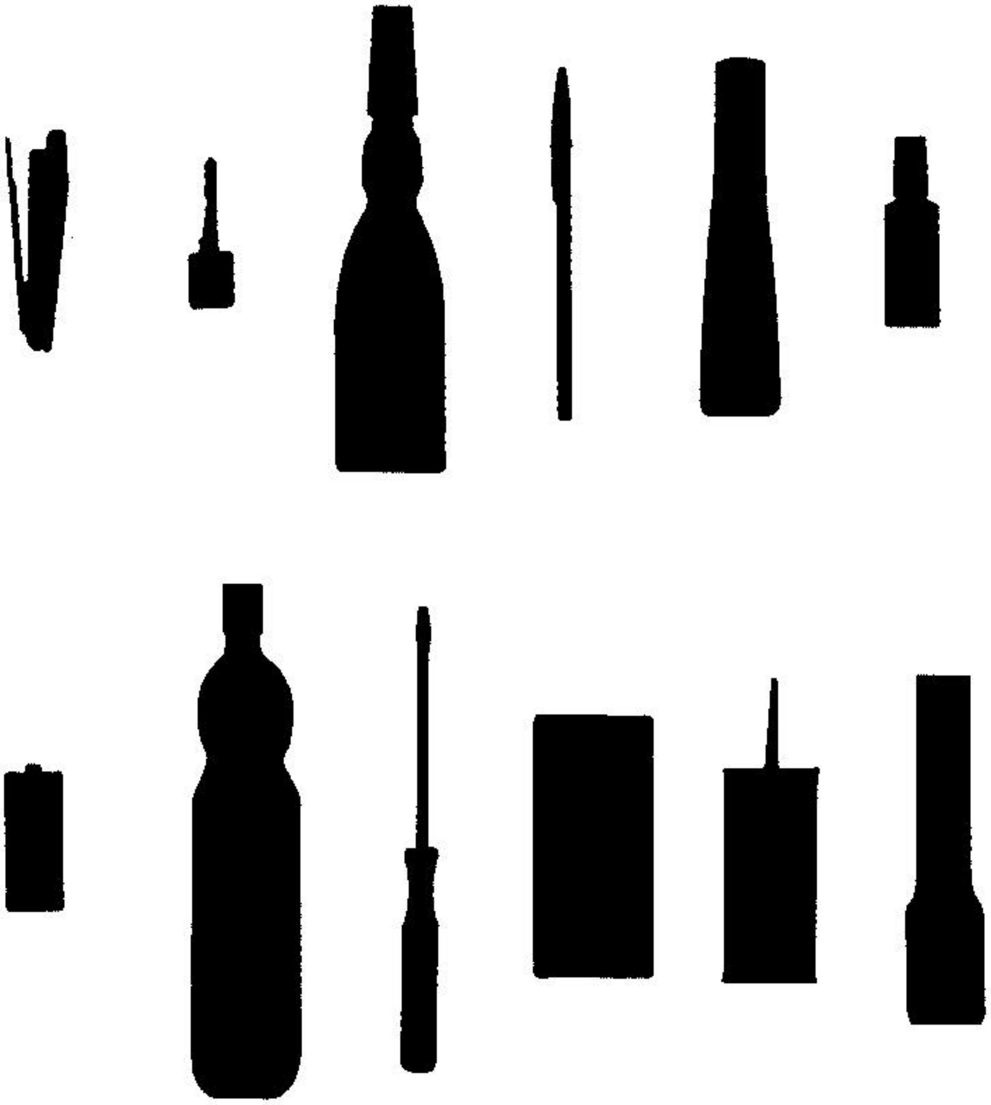


Figure 4.4 The twelve items which were used for the shape recognition experiments. Although no circular objects were used during the experimentation, MAVVIP is capable of recognising circular objects and this is shown in *Appendix A*.

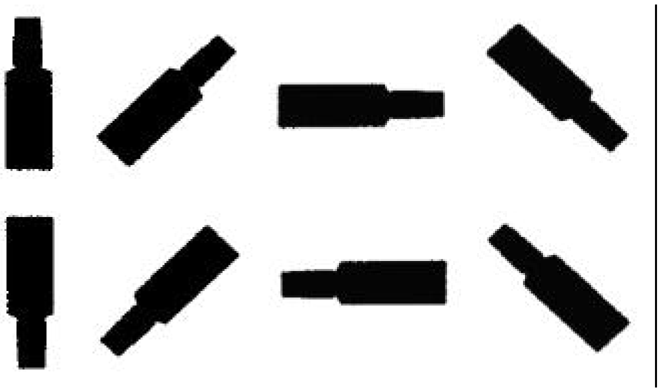


Figure 4.5 The eight different positions which each of the shapes were placed under the camera.

Test data was also needed to compare against the database. Therefore each shape was trained into the database an extra three times. The three positions are shown in figure 4.6.

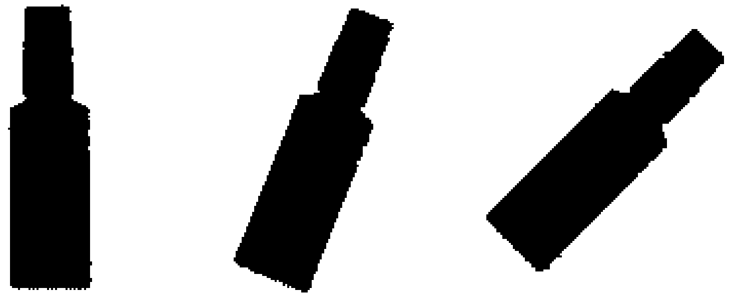


Figure 4.6 The three positions which each item was placed under the camera to obtain shape information to be compared with previous results.

4.3.3 Implementation of colour extraction method

In the authors opinion the colour extraction method was more reliable so it was decided to train in each shape four times as opposed to eight times in the previous methods. There are several reasons why the objects used in the shape classification were not used for the colour experimentation.

- The two experiments were carried out at different stages during the course of the project and some of the objects which were available for the shape experiments were not available for the colour experiments.
- There were 3 cylindrical objects trained in for shape experimentation and since cylindrical objects have no obvious front or back, they cannot be regarded as reliable when extracting colour information.
- When objects have similar shapes, colour recognition becomes important. Therefore it was decided to use objects with similar shapes. Several cards from the authors wallet were chosen for this demonstration and the six cards, front and back, are shown in figure 4.7.

The four standard positions which were used for training the cards are shown in figure 4.8. Once again the cards were trained in an extra three times to obtain test data which could be compared with the database. The three positions each card was trained in at are shown in figure 4.9.

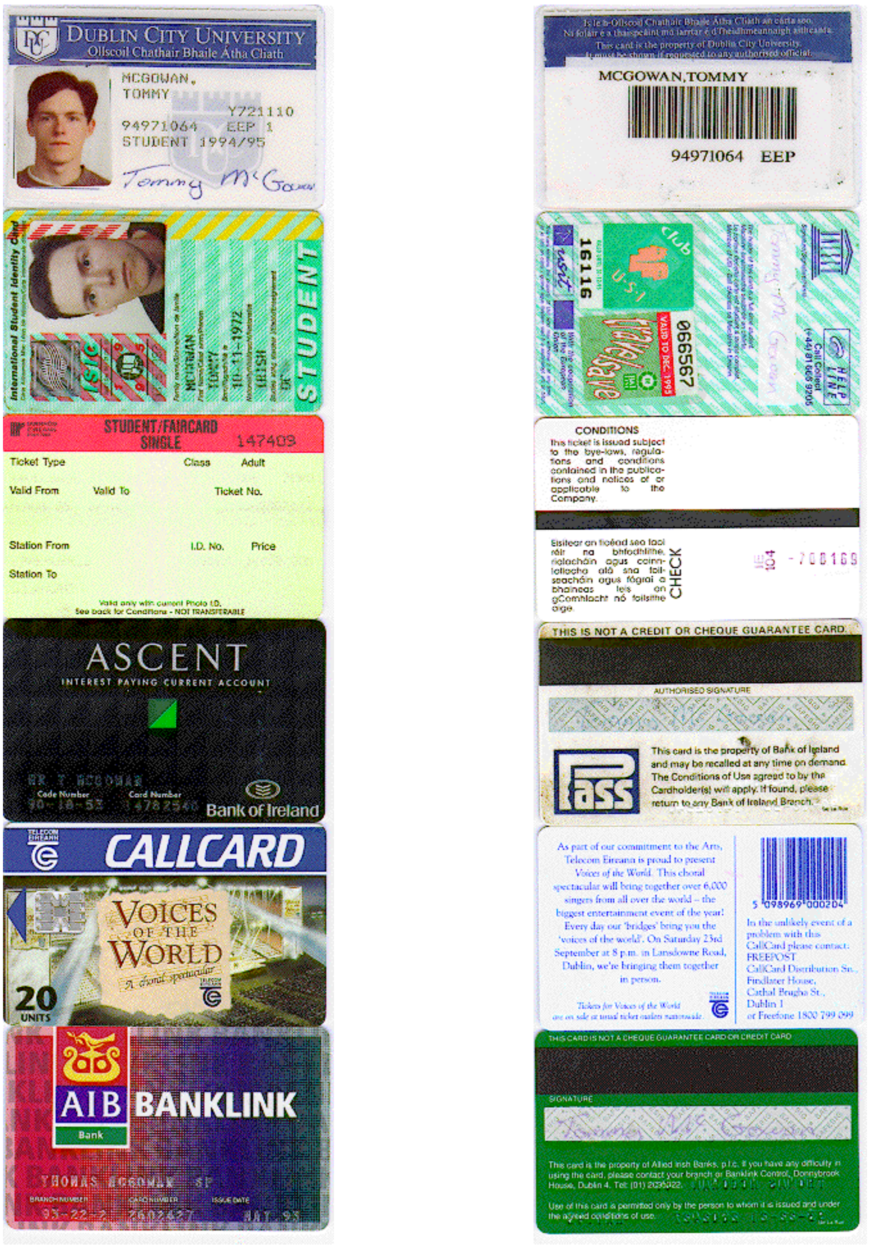

DUBLIN CITY UNIVERSITY
Ollscoil Chathair Bhaile Átha Cliath
MCGOWAN, TOMMY
94971064 EEP
STUDENT
STUDENT/FAIRCARD SINGLE
CONDITIONS
ASCENT
INTEREST PAYING CURRENT ACCOUNT
Bank of Ireland
THIS IS NOT A CREDIT OR CHEQUE GUARANTEE CARD
CALLCARD
VOICES OF THE WORLD
20 UNITS
AIB BANKLINK
THIS CARD IS NOT A CHEQUE GUARANTEE CARD OR CREDIT CARD

Figure 4.7 The front and rear sides of the six cards which were used for testing the colour information.

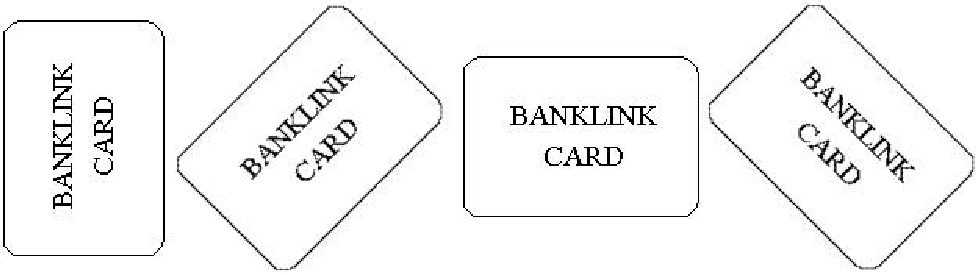


Figure 4.8 The four different positions each item was placed under the camera while training the database with the colour information.

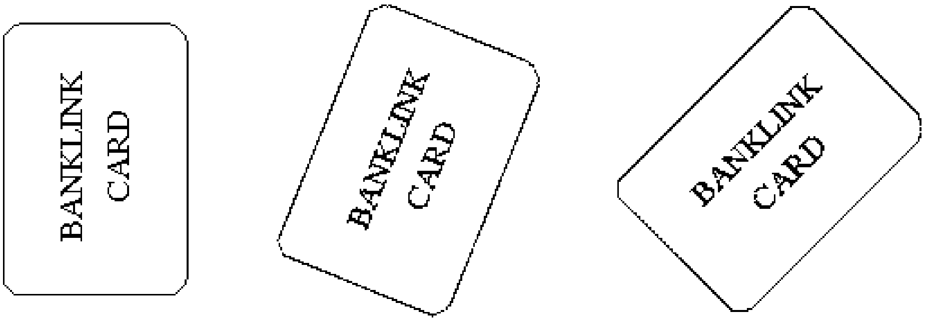


Figure 4.9 The three different positions each item was placed under the camera to extract colour information to compare with the database.

4.4 Implementation of classification methods and results of comparisons

Due to the complexity of the classification techniques, it was decided not to implement these methods in Prolog. Another means was required to implement these techniques firstly for evaluation. A mathematical software package called Matlab was chosen to implement and test the classification techniques using the data obtained in the previous section. Only the best of the classification methods would be implemented in Prolog and incorporated into the prototype.

4.4.1 Classification of shape information

Seven classification methods were implemented and were each tested with the data obtained from the four shape information extraction methods. Therefore a total of twenty eight tests were carried out. The results of those tests will now be discussed.

Of the twenty eight tests carried out, only four tests achieved thirty six out of thirty six correct. This can be seen in figure 4.10. The bisection method, the radial coding method and the rectangular method each obtained thirty six out of thirty six correct when the first classification (A simple nearest neighbour approach) method was used. The feature extraction method obtained thirty six out of thirty six correct when the second classification method (Method relying on means and standard deviations) was used. So we will narrow our search down to these four methods.

A further explanation is required for the four methods which have been singled out. When we say that an object has obtained thirty six correct matches out of thirty six correct tests, this does not mean that the correct shape was chosen in each case. In certain cases two shapes might have been selected and of these two shapes, only one of the shapes is the correct one. Similarly three, four, five, six or even seven shapes might have been selected but only one of these shapes is the correct one. Of the four methods chosen for further examination their results are shown in a little more detail in figure 4.11.

Classification Methods

No.1 - A simple nearest neighbour approach

No.2 - Method relying on means and standard deviations

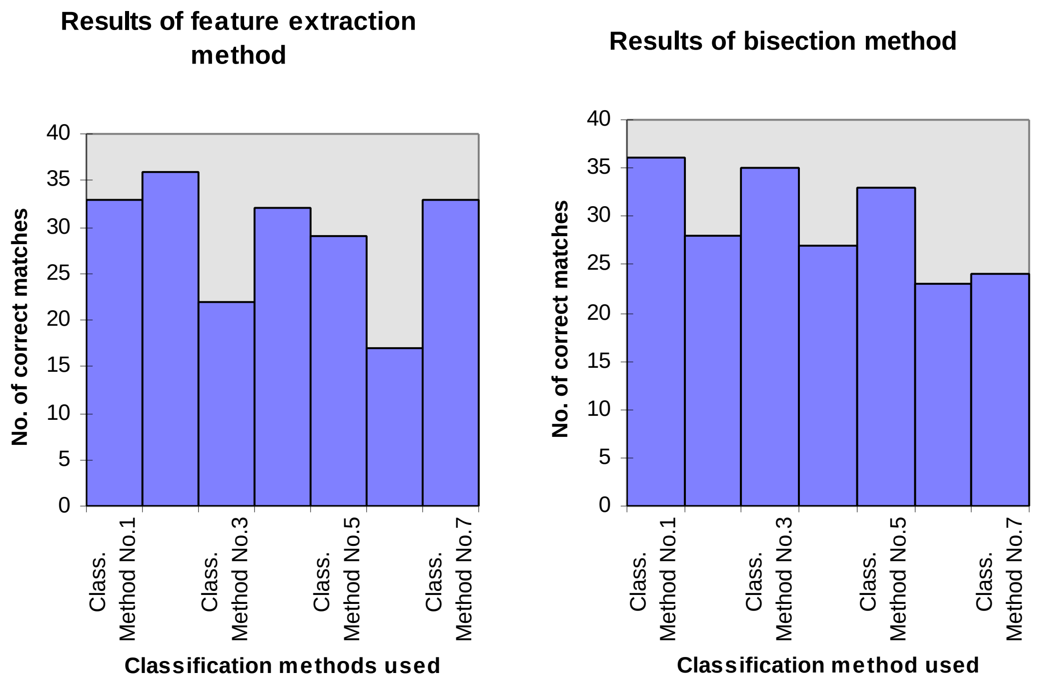
No.3 - Nearest neighbour method performed eight times in one dimension

No.4 - Nearest centroid method performed eight times in one dimension

No.5 - Nearest neighbour method

No.6 - Nearest centroid method

No.7 - Maximum likelihood ratio



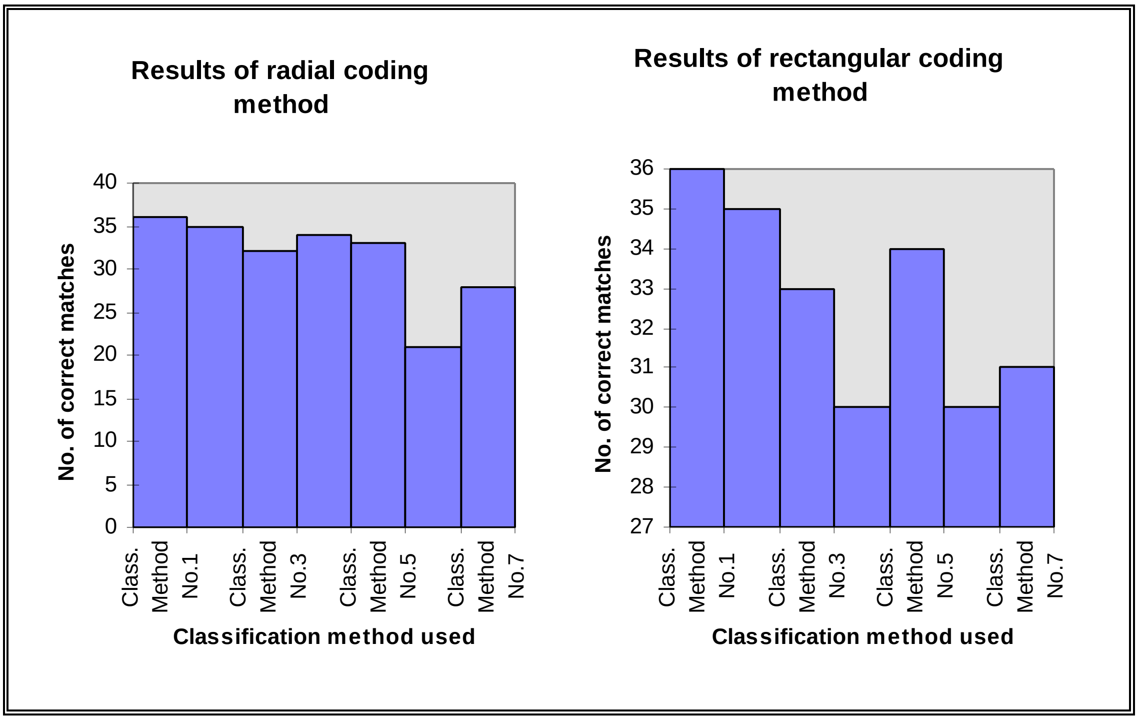


Figure 4.10 Graph showing the number of correct results for each of the thirty six tests.

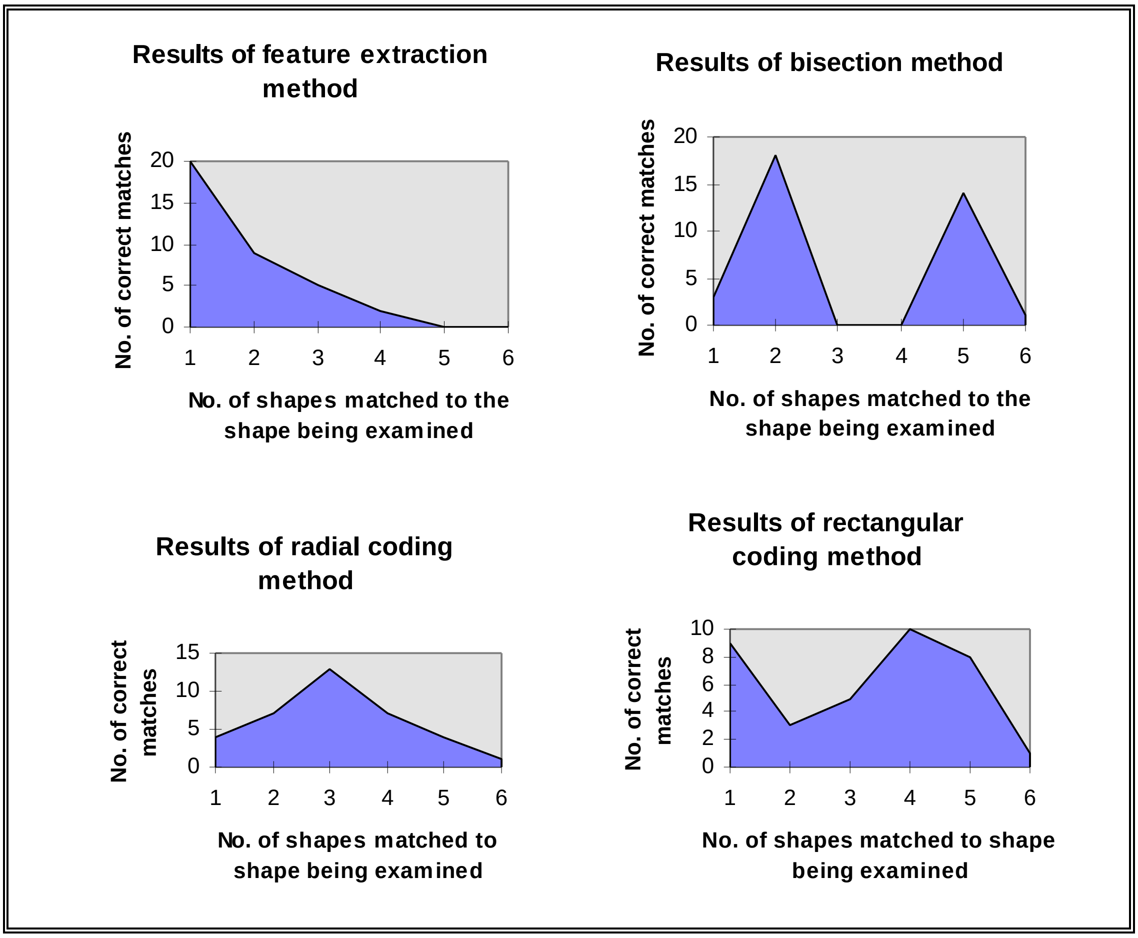


Figure 4.11 Graphs showing the number of shapes matched to the shape being examined using four different methods.

The ideal results of the above graph in figure 4.11 would be if the number in column one was thirty six and the rest were zero. This ideal situation would mean that each shape was correctly matched to one shape in the database. Unfortunately obtaining ideal results is very difficult. Therefore the best results are from a method which begins at a high level and drops to zero quickly. Of the four methods examined in figure 4.11, the first method is clearly the best. The first method in the graph uses feature extraction to obtain the shape information and uses the second classification method which relies on means and standard deviations and is explained in section 3.9.2.

At this point it was decided that feature extraction and method 2 (method relying on means and standard deviations) of the classification techniques would be incorporated into MAVVIP. It was necessary to see if the classification could be optimised to produce better results. Several experiments were run using this classification method and the tolerance in units of standard deviations were varied. The following graph in figure 4.12 shows the results of this same experiment using different tolerances.

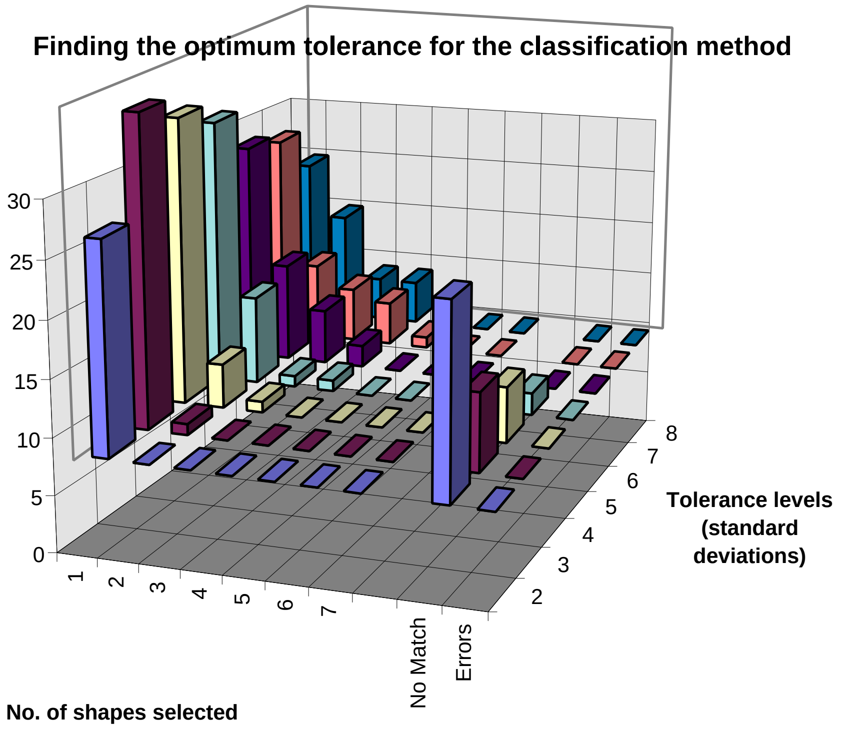


Figure 4.12 Graph showing the results of a single classification method with varying tolerances.

To start with the first tolerance of two standard deviations in the graph, it can be seen that nineteen shapes were identified correctly and the other seventeen could not be matched with the database. It can be seen that as the tolerance increases the number of no match situations decreases. The no match and error situations must be distinguished correctly. A no match situation occurs when the classifier cannot match a shape to any of the shapes in the database and an error occurs when the classifier match’s a shape incorrectly with another shape in the database. From the graph it can be seen that as the tolerances increase, the no match situations decrease but the number of possible shapes selected starts to increase. Therefore a balance must be decided on between the two. It was decided to use a tolerance of six standard deviations because this was the smallest tolerance which was without any no match situations.

An added bonus exists for this method over most of the other methods examined. Across all the tolerances tested there has been no errors and this reinforces the authors opinion that this is the most reliable of all the methods examined.

4.4.2 Classification of colour information

The colour data obtained from training the cards in figure 4.7 was incorporated into the classification techniques. The classification techniques used are the same as those used in the previous section for the shape data. The colour data was tested with the seven classification methods. The results of this experiment are shown in figure 4.13. As can seen from the graph, the colour information provided better results than any of the shape recognition methods. Four of the seven classification methods scored thirty six correct matches out of possible thirty six tests. The four best methods are methods 1, 2, 4 and 5. It was decided to examine these results a little more closely and this is done in figure 4.14.

Classification Methods

No.1 - A simple nearest neighbour approach

No.2 - Method relying on means and standard deviations

No.3 - Nearest neighbour method performed eight times in one dimension

No.4 - Nearest centroid method performed eight times in one dimension

No.5 - Nearest neighbour method

No.6 - Nearest centroid method

No.7 - Maximum likelihood ratio

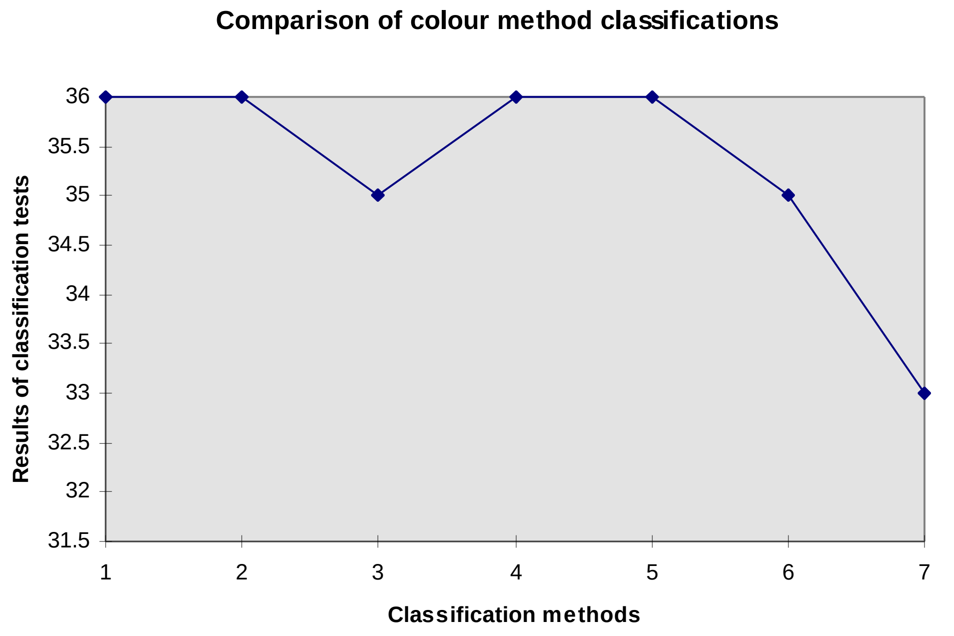


Figure 4.13 Graph showing the results of the seven different classifiers using the colour data.

From the graph it can be seen that methods two and five are the best. It can be seen that in these cases, one object is identified correctly in each of the thirty six tests. Classification method no. 2 and method no. 5 were the two methods left to choose from. It was decided to use method no. 2 because this was the method chosen to classify the shape information. Since this method had already been implemented the workload was reduced. The efficiency of MAVVIP is likely to be improved because one classification method will be used instead of two different ones.

Classification Methods

No.1 - A simple nearest neighbour approach

No.2 - Method relying on means and standard deviations

No.4 - Nearest centroid method performed eight times in one dimension

No.5 - Nearest neighbour method

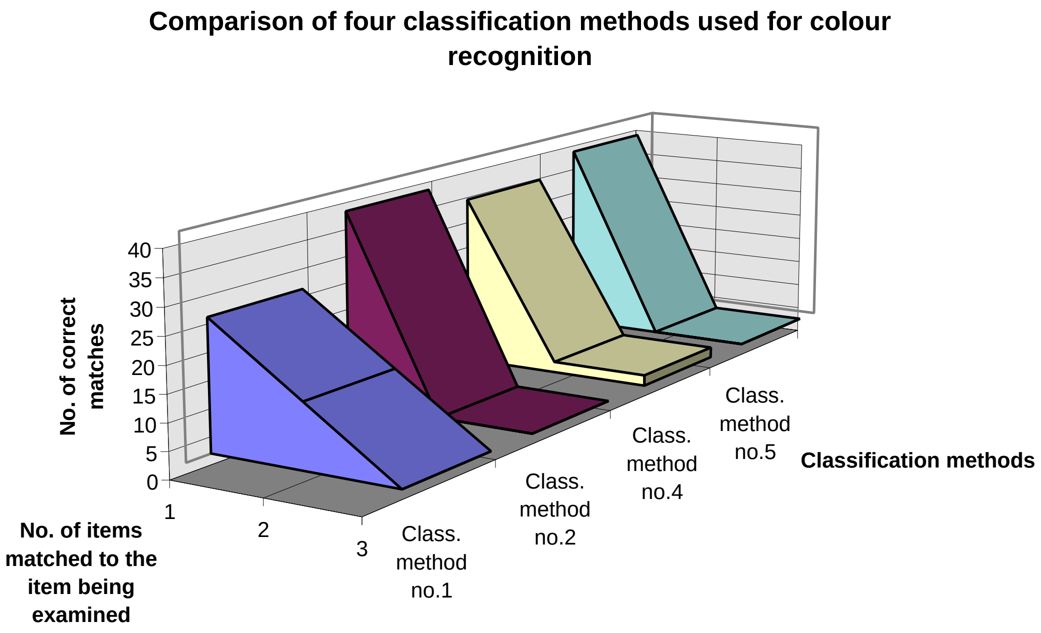


Figure 4.14 Graph showing the number of times an item was matched to one or more items in the database.

The present results of method two are ideal. It was decided to lower the tolerance to see if ideal results could be reached with a lower tolerance. As can be seen from figure 4.15, when the tolerance was dropped to five standard deviations only thirty four out of thirty six tests were correct and there were two no match situations. Similarly when the tolerance was dropped further there were even more no match situations. Therefore it was decided to use a tolerance of six standard deviations.

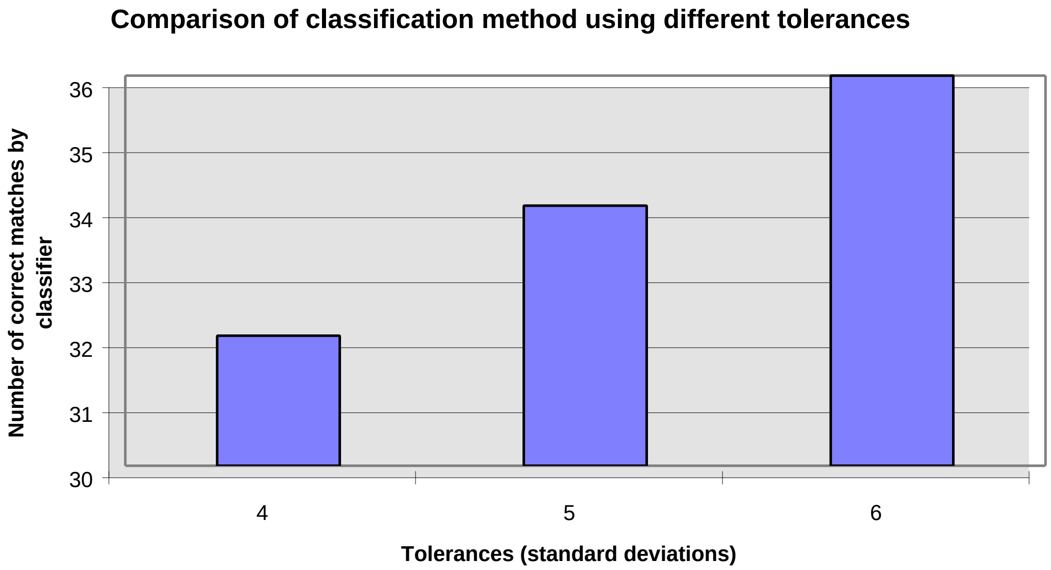


Figure 4.15 Graph showing the number of correct matches using three different tolerance levels.

4.4.3 Implementation of classification method no. 2 (method relying on means and standard deviations)

Implementing classification method no. 2 is a difficult task in Prolog because Prolog is a high level artificial intelligence language. Prolog is not suited to mathematical calculations. Therefore it was decided to incorporate the use of C code resources into the Prolog program. C is an ideal language for performing complex mathematical routines. MacPROLOG has the ability to interface with C or Pascal using code resources. A more detailed explanation on C code resources is provided in the next section on speech synthesis.

A C code resource was written which could be activated at any stage from the Prolog environment. Measured values obtained from the training procedures were passed to the code resource, which calculated the mean and standard deviation of the numbers before returning these new values back to the Prolog environment. The mean and standard deviations were then stored in the database along with other information on the object in question.

4.5 Implementing speech synthesis

A brief summary of the speech synthesis section, more specifically the problems which have to be tackled, will be given now before discussing the design approach and the implementation of this section of the project. The following sections also describe some features of the MacPROLOG Text to Speech package. A detailed description on the various stages of development of the MacPROLOG Text to Speech package is provided by McGowan (1994).

The project involves creating a Prolog program which will provide text-to-speech facilities for any Prolog user. It has been decided to use Apple's new Speech Manager package to carry out the actual text-to-speech process. There are two separate stages contained within this project. The first stage is to create an interface between Speech Manager and the MacPROLOG environment. The second stage is to develop a suitable interface in the MacPROLOG environment to provide the user with easy access to the text-to-speech facilities. It must be pointed out that this user interface will probably not be required for the finished prototype but it is merely for providing easy access to speech for testing and experimentation purposes. A block diagram representing the two stages is shown in figure 4.16.

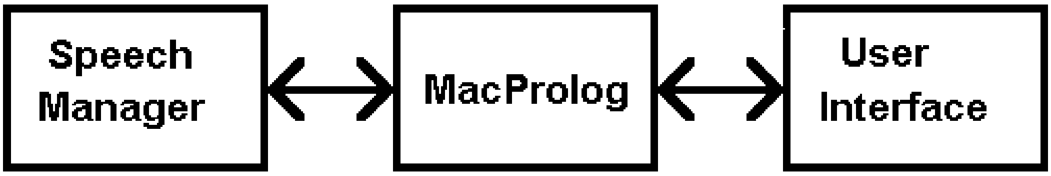


Figure 4.16 Block diagram representing speech synthesis section of MAVVIP.

4.5.1 Design approach

Speech Manager is not a stand alone application, it is a collection of PASCAL functions which are used by applications written to take advantage of them. These functions provide a means for software developers to create customised speech synthesiser packages. The Speech Manager functions cannot be called from within the

MacPROLOG environment, they can only be called from within the C or PASCAL environments.

MacPROLOG is capable of interfacing with C code resources or PASCAL code resources. Code resources are pieces of either C or Pascal code which are called from within the MacPROLOG environment using the *call_c* and *call_p* predicates. A C code resource which would call the Speech Manager functions was created. This C code resource acts as a go between for interfacing MacPROLOG and Speech Manager. The block diagram for this interface is shown in figure 4.17.

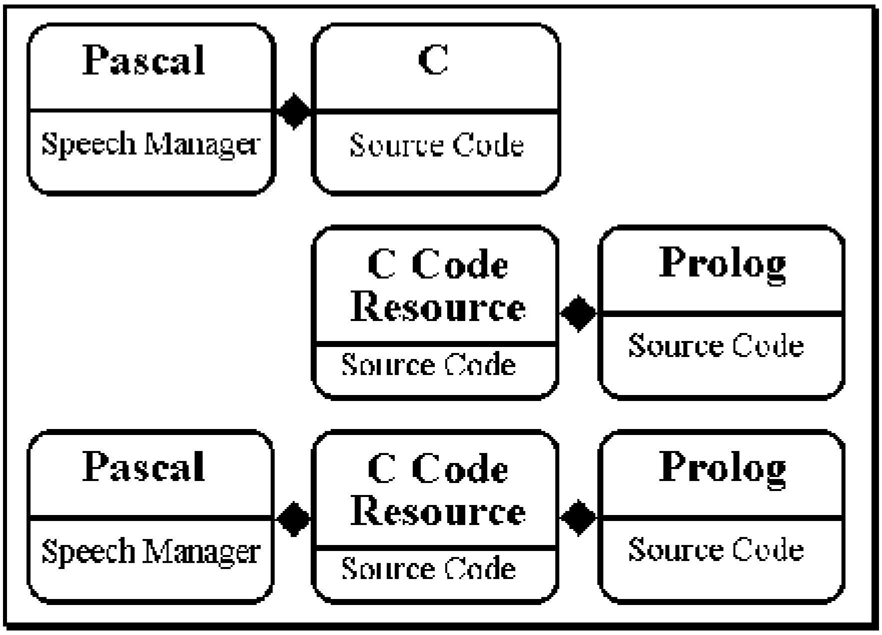


Figure 4.17 This diagram shows the block diagram for the MacPROLOG - Speech Manager interface.

4.5.2 C code resources

Code Resources cannot use ANSI functions (e.g. *printf, scanf* etc.) because the code resource is being run from inside the MacPROLOG environment. Therefore the standard input / output functions of C are of no use anyway. Full instructions for interfacing the C code resource to the MacPROLOG program are given in the MacPROLOG 3.0 Reference Manual (Johns, 1990). Parameters can also be passed to and from Prolog programs and C code resources.

4.5.3 Debugging C code resources

The main disadvantage with C code Resources is that they are very difficult to debug. Standard debugging methods such as tracing and setting breakpoints do not work with code resources.

The best strategy for this problem was first to develop the program in C and when this was working correctly, convert the C code into a C code resource. This was done by adding some extra code to the program and this is explained in the MacPROLOG 3.0 Reference Manual (Johns, 1990). Implementing the code in C first meant that all syntactic and programming errors were eliminated before the code resource was created. Once the code resource was created, it was tested. If the code resource was not working correctly, then the *print_char* or *print_str* commands were entered into the code resource in various parts of the program. These commands printed a character or a string in the Output Window of MacPROLOG when executed. Therefore when a specific character or string appeared in the Output Window, then it was known that the C code resource had reached that point in the C code resource successfully.

4.5.4 Approach taken for developing the software

It was decided to take a modular approach to writing the speech software. This required developing a number of individual programs instead of developing one main program piece by piece. It was hoped to have a number of individual modules instead of one big program. When sufficient modules had been created, they were combined into one large program.

If the software is presented as individual modules then there is more flexibility when integrating these programs together to create a text-to-speech synthesiser specific to a person's needs. This would also be useful for others in the Vision Systems Group who would require a customised text-to-speech synthesiser.

4.5.5 Error messages

Error messages have been incorporated into the code as a means of information for the user. Every command executed in the C code resource has an error message associated with it. Therefore if and when an error occurs in the code resource, the user is notified by means of an error message in the Output Window of MacPROLOG. If an error occurs and the code resource is terminated, an error message appears on the screen notifying the user of the mistake. An example of this would be if you sent an empty string of text to the code resource. There is no text for Speech Manager to speak so the code resource terminates and the following message appears on the screen.

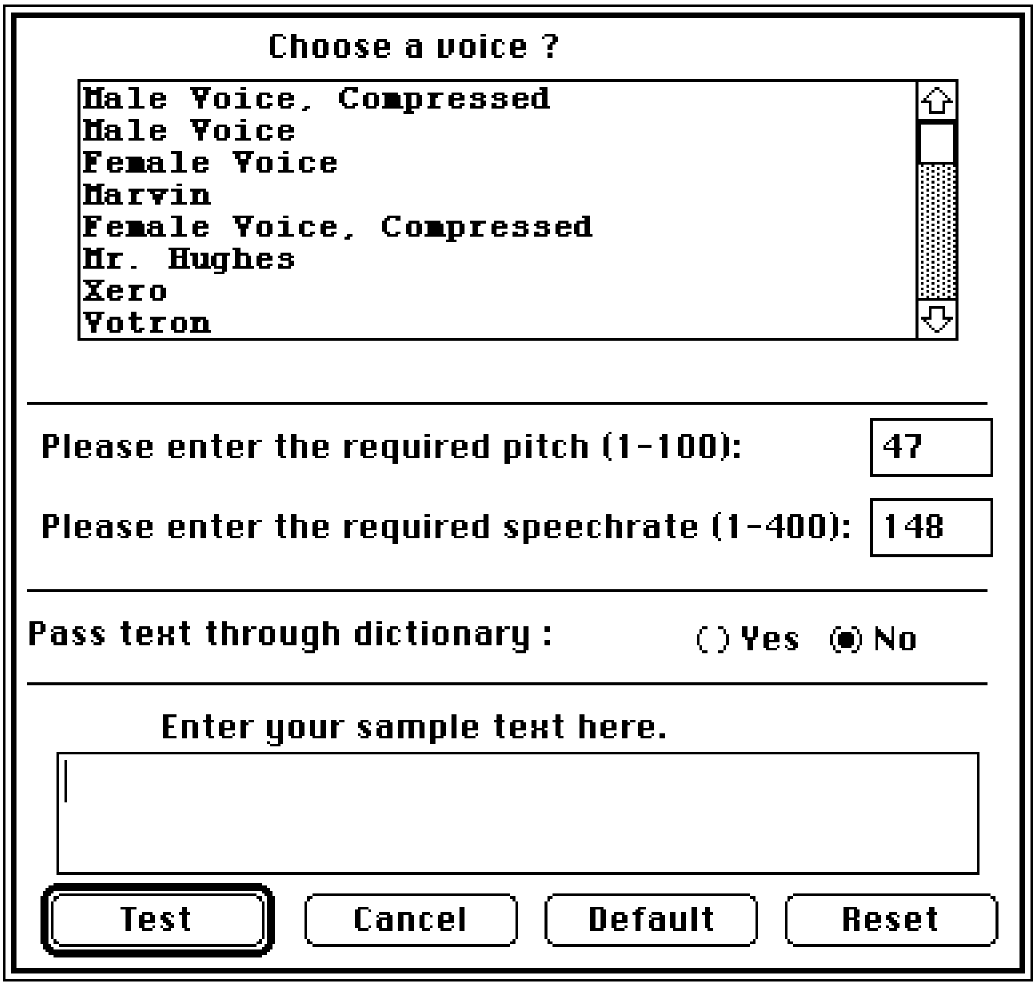
"Invalid string was sent to C Code Resource."
"C Program terminated."

The C code resource does not always terminate if an error occurs. An example of this is when an invalid pitch rate is sent from the Prolog program. In this case the following message appears in the Output Window.

"Invalid Pitch Rate was sent to C Code Resource."
"Default Pitch Rate used."

4.5.6 Choose voice menu option

There are several options available in the speech menu. The first of these options is the *Choose Voice* menu option and the window for this option is shown in figure 4.18. This window offers the user the opportunity to create a voice to suit their specific tastes. A list of voices is provided and it is up to the user to select one. Once a suitable voice has been selected, the pitch rate and the speaking rate can be altered to improve the voice further.



Choose a voice ?

Male Voice, Compressed
Male Voice
Female Voice
Marvin
Female Voice, Compressed
Mr. Hughes
Xero
Votron

Please enter the required pitch (1-100): 47

Please enter the required speechrate (1-400): 148

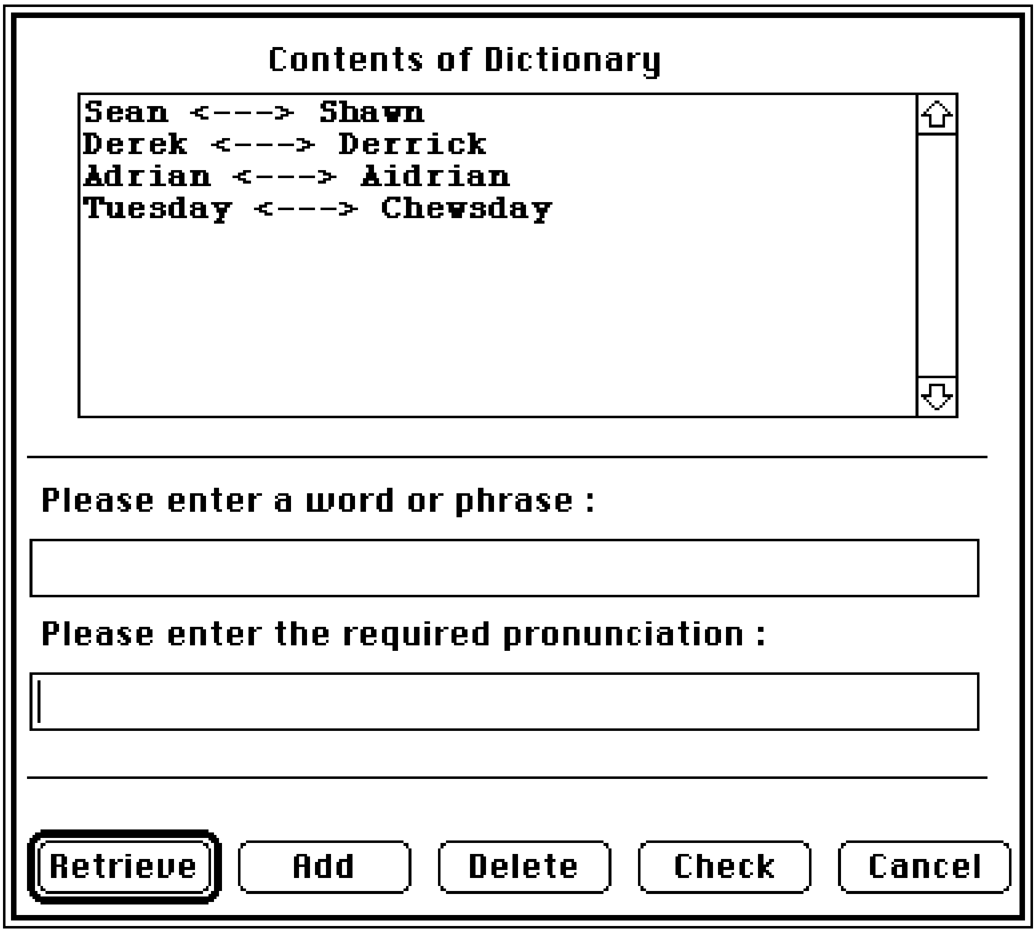
Pass text through dictionary : ○ Yes ● No

Enter your sample text here.

Figure 4.18 Choose Voice menu option.

4.5.7 Dictionary menu option

The next menu option of importance is the dictionary. The dictionary window is shown in figure 4.19. This provides a means for the user to enter in an alternative spelling for a word or phrase. Remember that the dictionary is not responsible for correcting spelling mistakes, it offers an alternative spelling which will give a better pronunciation for a word. It is up to the Prolog user to enter in whatever words in their opinion require a better pronunciation.


Contents of Dictionary
Please enter a word or phrase :
Please enter the required pronunciation :
Retrieve
Add
Delete
Check
Cancel

Sean <---> Shawn
Derek <---> Derrick
Adrian <---> Aidrian
Tuesday <---> Chewsday

Figure 4.19 Dictionary menu option.

The dictionary program is an essential part of the text-to-speech synthesiser. This dictionary does not correct spelling mistakes like a dictionary in a typical word processing package. This dictionary contains an alternative spelling for a word and it is this alternative spelling, which produces a better pronunciation of the word. Therefore it is helpful if text is passed through a dictionary before being spoken. Everybody has different opinions on how speech sounds, therefore it is up to the individual MacPROLOG Text to Speech users to enter in their own alternative phonetic spellings for words.

There were several options available for implementing the dictionary. The dictionary could be implemented in Prolog, C or a combination of C and a ResEdit resource file. Also Speech Manager has a facility for creating and maintaining a dictionary. Implementing a dictionary in C is quite a large task, because C is a low level language. For this reason a dictionary in C would probably be more efficient than a similar one created in a high level language such as Prolog. It was uncertain if there would be enough time to create a dictionary using C. One disadvantage of using C would be that all dictionary items would have to be continuously passed between MacPROLOG and the C code resource. This continuous passing of parameters would slow the package down. For this reason alone, one of the aims of this project was to develop as much of the software as possible in the MacPROLOG environment and to use the code resource as little as possible.

The second option of implementing a dictionary in Prolog was the one chosen. Prolog is a high level language and it was felt that implementing a dictionary would not require as much time as it would using C. Speed was the most important factor in deciding which method to use implementing a dictionary. It is important that a dictionary be extremely fast. There was no way of telling which method would prove faster so it was decided to go for the easier method of the two which was using Prolog.

Implementing a dictionary proved more difficult than expected. To compare words or phrases, they first had to be converted into a list of ASCII characters and operated on using list handling commands. The list handling commands in MacPROLOG are good, but are still quite limited. The final program proved to be quite inefficient. The final dictionary program consisted of three loops one inside another. This proved to be quite slow and to make the problem worse, Prolog supports automatic backtracking. This means when the end of the program is reached, the program backtracks line by line back to the start of the program. This contributed greatly to slowing the program down. One other problem which occurred while using Prolog was the use of the *fail* command. Some of the variables in the program are assigned their values while the program is backtracking, but if a *fail* command is encountered while backtracking the variable is not assigned its value. The solution to this was that every time the program edited the sentence, it would write the new sentence over the old sentence in the database. Similarly before the sentence was edited it was read from the database.

Overall the dictionary program worked correctly but was quite slow. As the dictionary facility was not a priority for MAVVIP, a second more efficient dictionary program in C was not implemented.

4.5.8 Text to phonemes menu option

A facility for converting text to and editing phonemes is also provided under the *Text to Phonemes* menu option. The Text to Phonemes window is shown in figure 4.20.

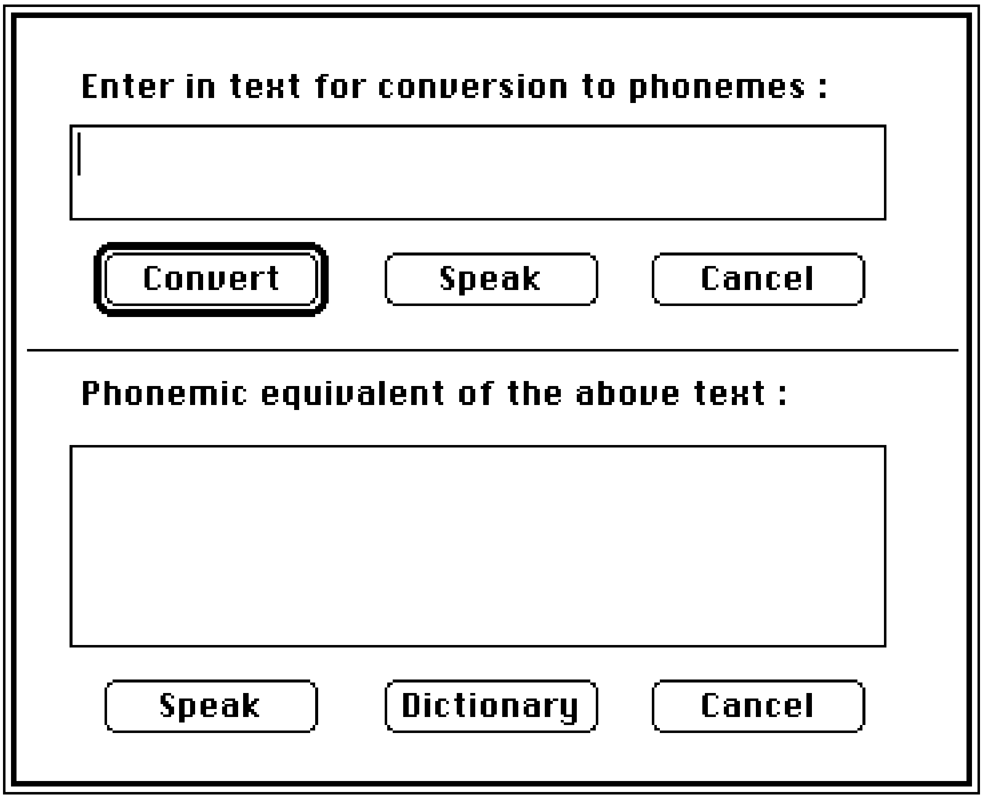

Enter in text for conversion to phonemes :
Convert
Speak
Cancel
Phonemic equivalent of the above text :
Speak
Dictionary
Cancel

Figure 4.20 Text to phonemes menu option.

4.5.9 Phonemes

Phonemes represent an intermediate stage in the conversion of text to speech. Most modern text-to-speech synthesisers consist of a two stage process, the text to phoneme stage and the phoneme to speech stage. Phonemes correspond to specific sounds, when input to the speech synthesiser are sent to the sound driver and the resulting output to the speakers is in the form of speech. By converting text to speech, the user can edit the phonemes to produce the best possible speech synthesis of the text.

To give a simple example of phonemes, take the letter A. This letter A is pronounced several different ways depending on letters which surround it. The following diagram in figure 4.19 shows four words which have different pronunciations for the letter A. The corresponding phonemes for the letter A in each case are also provided.

Phoneme	**Example**
AE	b**a**t
EY	b**ai**t
AO	c**au**ght
AX	**a**bout

Figure 4.21 Table showing phonemes corresponding to four different pronunciations of the letter A.

A phoneme represents a sound corresponding to a letter. Speech is created by putting together several of these phonemes. A complete table of phonemes is provided by McGowan (1994).

4.5.10 Prosodic controls

Phonemes alone are not sufficient to create natural sounding speech. Others factors have to be included before natural sounding speech is created. Prosodic controls are symbols used to control pitch, emphasis and stress on certain words and phrases. They are also responsible for creating short pauses after full stops, commas, question marks, semicolons etc.. A complete table of prosodic controls is provided by McGowan (1994).

The word "object" is pronounced differently depending on whether it is being used as a noun or a verb. As a noun the stress is placed on the first syllable, as a verb the stress is placed on the second syllable. Therefore by placing the symbol for stress before either the first or second syllable, the user can determine how the word is pronounced. More detailed information on phonemes and prosodic controls are given in *section 6* of the MacPROLOG Text to Speech User Manual (McGowan, 1994).

4.5.11 Prolog commands

Several Prolog commands were created which enable the user to incorporate the speech facilities into their own programs. These commands are as follows :

1. speak_text
2. use_dictionary
3. add_to_dictionary
4. delete_from_dictionary
5. speak_contents_of_dictionary
6. text_to_phonemes
7. call_c_program

Of the seven Prolog commands available, the *speak_text* command is probably the most important. It is this command that actually produces the speech. This command is capable of speaking any string of two hundred and fifty six characters or less. There are extra parameters which are required by this command in order to select a voice. If the second argument in the command is ‘default’ then the MacPROLOG Text to Speech default voice is used. Similarly, if the second argument is ‘current’ then the last voice which was selected from the Choose Voice window is used. Another alternative for the *speak_text* command is to select a voice number, pitch rate and speaking rate and insert them as the second, third and fourth arguments respectively in the command. A piece of sample code illustrating different uses of the *speak_text* command is shown in figure 4.22.

```
father('Bob','Pat').
father('Gabriel','Jimmy').
father('Andrew','Peter').

sample_program_1:-
writenl('Enter in a name enclosed in single quotes ?'),
read(Name),
next_stage1(Name).
```

```
next_stage1(Name):-
father(Father,Name),
concat(Name,' has a father called ',String),
concat(String,Father,FinalString),
writenl(FinalString),
speak_text(FinalString,1,42,130),!.

next_stage1(Name):-
father(Name,Son),
concat(Name,' has a son called ',String),
concat(String,Son,FinalString),
writenl(FinalString),
speak_text(FinalString,'current'),!.

next_stage1(Name):-
String = 'No one by that name exists.',
writenl(String),
speak_text(String,'default').
```

Figure 4.22 Sample code demonstrating the use of the *speak_text* command.

An abbreviated version of the Prolog commands also exist and they to are explained in the MacPROLOG Text to Speech User Manual (McGowan, 1994).

4.5.12 Embedded speech commands

Embedded speech commands represent a set of commands that can be included into any text which is sent to Speech Manager. These commands will be executed at the point in the sentence which they are placed. These commands are used to change certain characteristics of the speech after the speech has started. For example, if you wish to change the pitch rate midway through a sentence, include the embedded speech command for pitch and select a new pitch rate. A piece of sample code is shown in figure 4.23 below, which demonstrates the use of an embedded speech command to change the pitch in the middle of a sentence.

```
begin3:-
String = 'This is normal pitch [[ pbas 90 ]] This is high
pitch.',
speak_text(String,4,47,148).
```

Figure 4.23 Sample code demonstrating an embedded speech command to change the pitch.

Similarly, there are embedded speech commands which can alter the speaking rate and volume of a voice. There are commands to set the input mode to read phonemes, read words, character by character and to read numbers, digit by digit. There are also embedded commands to reset voice parameters, comment on other commands and to add delays into a sentence if needed.

4.5.13 Concluding remarks

This section gave an outline of the MacPROLOG Text to Speech package. It showed the user interface windows available and gave brief descriptions of their capabilities. This chapter also mentioned the Prolog commands and explained how they can be incorporated into Prolog programs. The uses of Phonemes, Prosodic controls and embedded speech commands are also discussed in order to demonstrate various methods of improving synthesised speech.

4.6 Implementation of final prototype

4.6.1 Feature extraction

The ease at which the intelligent camera can be manipulated from the Prolog environment meant that the shape extraction software was put together and tested in many different ways in order to find the optimum method. Once the software was put together and tested for the last time, the code was converted from Prolog to Intelligent camera commands and these were then downloaded into the camera’s EPROM (Erasable Programmable Read Only Memory). When run from within the camera the process is considerably quicker. The reason is that valuable time was being wasted sending individual commands along a serial cable to the camera. Unfortunately only the image processing modules can be run from the camera. However the later stages of this project run from the computer and don’t require information being passed over and back along the serial cable.

4.6.2 Colour extraction

Implementing the colour extraction part of the experiment was quite difficult. There is no right or wrong settings for adjusting the colour extraction software. This was done on an ad-hoc basis until the author felt he had achieved the best results. A colour scattergram was created and tested repeatedly using variations on the saturation allowed. The final colour scattergram is shown in figure 4.3.

The colour scattergram was programmed into the colour filter and all objects were viewed with the colour filter. Objects viewed through the colour filter appeared on the monitor as pseudocolour images. The pixels within the shape area were counted and classified into eight distinct colour groups. The percentage representations of the colours were computed and the eight percentages calculated represented the colour characteristics of the shape which were then stored in the database. Figure 4.2 shows an object in true colour and in pseudocolour.

4.6.3 Database

The database consisted of two parts. There was a temporary database and a permanent database. The temporary database was only used while training objects into the system. While training an object, all the data is stored in the temporary database. When the training of the object has been completed, the data in the temporary database is extracted and passed to a C code resource which calculates the relevant means and standard deviations. The object along with the name of the side of the object just trained in, all necessary means and standard deviations and additional information (e.g. ingredients and cooking instructions) was then stored in the permanent database.

4.6.4 Speech synthesis

The speech synthesiser was scaled down in size before being put into the final prototype. Only the essential parts were included because it was necessary to minimise the MAVVIP prototype as much as possible. The reason for this is that MAVVIP has a limited memory evaluation space so non-essential code was removed for efficiency.

4.6.5 Closeness of Match Value

Two closeness of match values have been incorporated into the program. One for the colour recognition and one for the shape recognition. The closeness of match value consists of the sum of the differences between the measured values and their corresponding mean values in the database. Therefore the smaller the closeness of match value then the more likely a possible match has been found. Closeness of match values will be provided every time an object is successfully recognised. These values will have no direct influence towards the classification of items but are merely an extra indication of how well the classification technique is performing.

4.7 Problems with implementation

There was only one significant problem which had to be tackled during the implementation of the prototype. This problem occurred intermittently with the analysis and classification of the colour information.

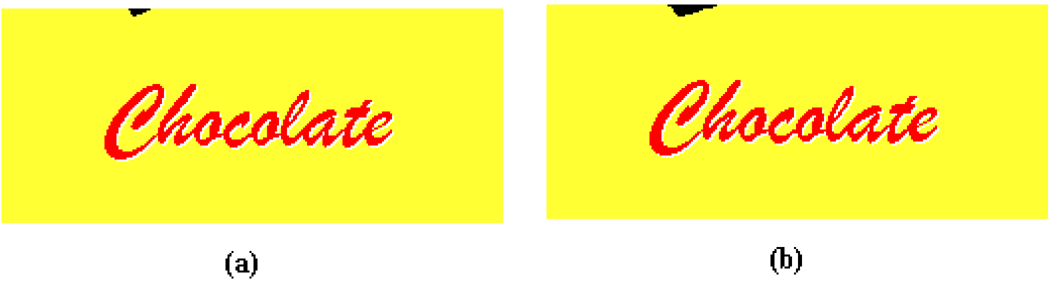


Figure 4.24 (a) A sample *chocolate bar* and (b) the same sample bar after the wrapper had been moved slightly

Take for example the *chocolate bar* in figure 4.24 (a). When training this object there was a small black mark on the side of the wrapper which partially overlapped onto the top of the wrapper. At a later stage when it came to recognising the *chocolate bar* more of the black patch was visible as shown in figure 4.24 (b). Assume this is because the chocolate melted at some stage and the wrapper moved slightly. Therefore the chocolate bar now has more of its surface area occupied by black and less of its surface area occupied by yellow. There was approximately a one percent decrease in the surface area occupied by yellow and approximately a two hundred percent increase in the area occupied by black. The classification system matches the red and yellow components of the object successfully but the recognition fails because the black patch is not matched correctly. The point of this example is that colour matches are more likely to fail due to a fluctuation of a colour which occupies a small percentage of the surface area. In order for the classification system to fail this fluctuation would have to occur after the training stage was complete.

In reality this problem only occurred for colours which occupied considerably less surface area than the example shown in figure 4.24. The solution was to provide all colours which had a very small or even zero percentage surface area with a minimum tolerance. This meant the system did not fail for relatively insignificant colour changes for colours occupying very small percentage surface areas.

4.8 Discussion of final prototype

Overall the implementation went well with no major setbacks. One surprise however was the results of the classification tests. Theoretically it was expected that the maximum likelihood classification would prove more accurate than the other classification methods. In the authors opinion the poor performance of this popular method is due to the small number of samples taken to form the Gaussian distribution. With just eight samples taken to train the shapes and four samples while training the colour data, this was not nearly sufficient to form a proper Gaussian distribution. An example of a set of samples from the shape training is shown in figure 4.25. In a situation where many more samples were taken and an ideal Gaussian distribution was created, then the maximum likelihood classification would probably be more appropriate. Realistically, it is not feasible to train in objects many times because this would take too long for the end user of MAVVIP.

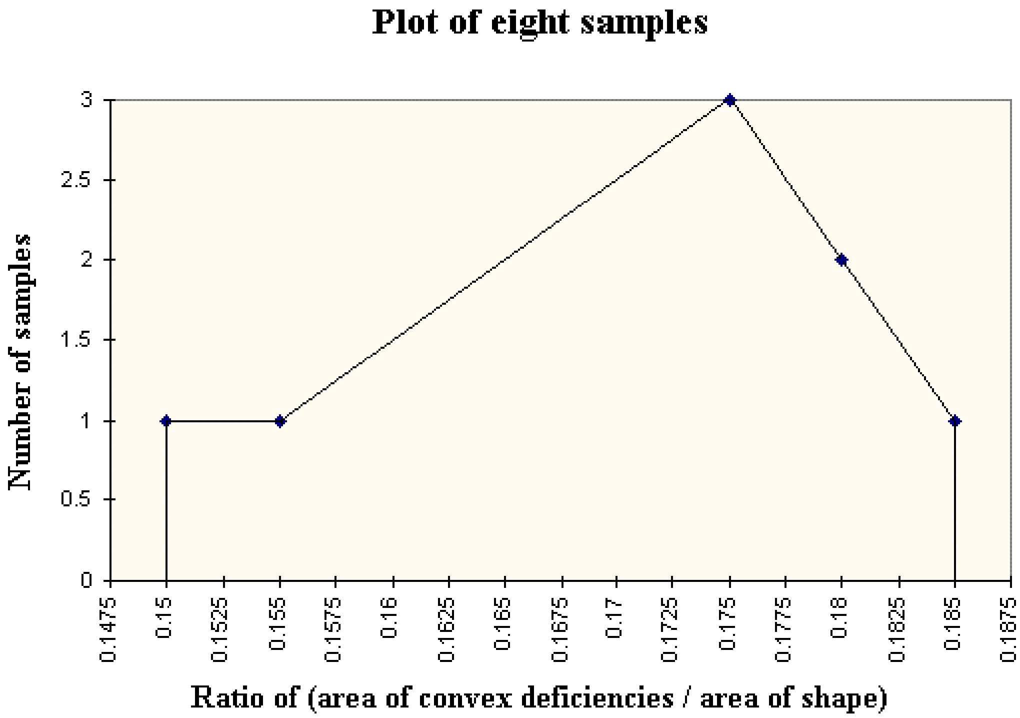


Figure 4.25 A plot of eight samples. This graph does not represent an ideal Gaussian distribution.

Overall the implementation stage was very successful and initial results from the system were promising. A detailed analysis of MAVVIP is described in the next chapter along with a discussion of the results.

5. Experimentation and Results

5.1 Introduction

This chapter consists of five experiments designed to test MAVVIP extensively. The main objectives of these experiments are as follows :

- Optimise the training procedure to give most accurate results.
- Push MAVVIP to its limits of operation.
- Finding the weaknesses in MAVVIP.
- Finding new ways of improving the performance of MAVVIP.

This was done using the following five experiments.

- Experiment 1 - Optimising training procedures.
- Experiment 2 - Checking flexibility of MAVVIP for scale changes.
- Experiment 3 - Recognition of items elongated both vertically and horizontally.
- Experiment 4 - Cylindrical objects.
- Experiment 5 - Testing groups of objects.

5.2 Experiment 1 - Optimising training procedures

Using eight household items, each item was trained into the database eight times. After ad hoc testing at an earlier stage in the project it was decided to train objects into the database eight times. This was not experimentally proven to be the optimum number of times to train an object into the database. Therefore it was necessary to experimentally find the optimum number of times to train an object into the database. To be more precise, optimum training is minimising the number of times an object is trained into database without losing accuracy in recognition stage.

Therefore each item was trained into the database in seven different ways. Firstly each item was trained into the database using a combination of the item in two different positions. Secondly each item was trained into the database using a combination of the item in three different positions. Similarly each item was trained in at an increasing number of positions until the maximum of eight different positions was reached. The diagram in figure 5.1 shows seven rows showing the seven sets of positions which were trained into the database for each item.

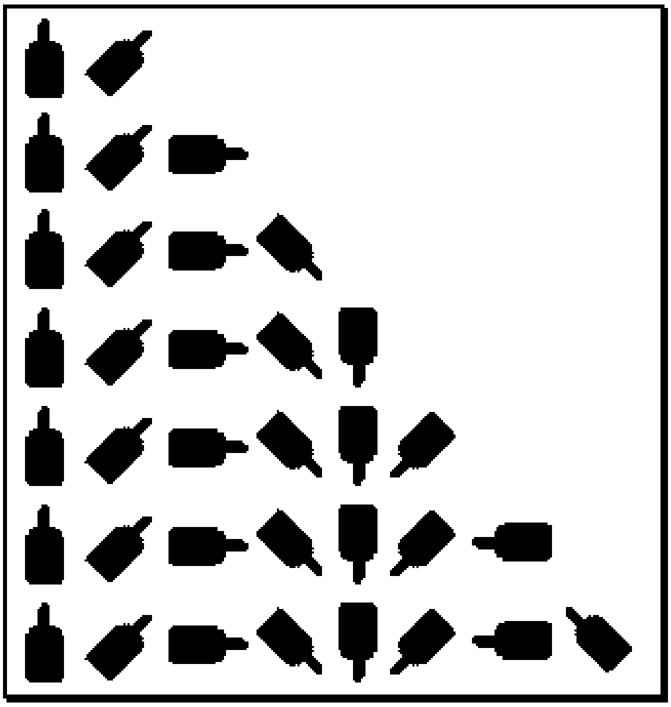
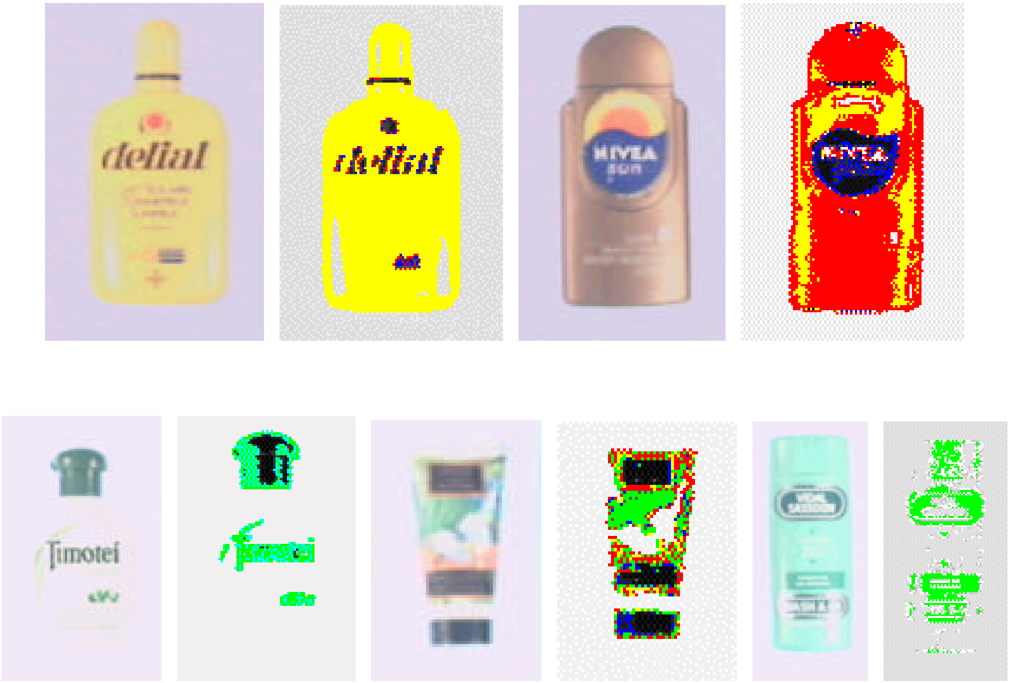


Figure 5.1 Seven rows showing the seven different sets of positions used to train each item into the database.

The eight items used for this first experiment are shown in figure 5.2 along with their respective pseudocolour images.



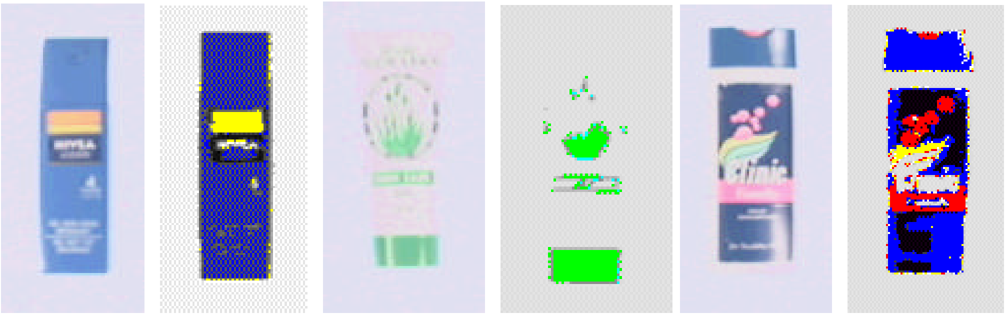


Figure 5.2 Eight items used for the first experiment. The appropriate pseudocolour image for each of the items is also included.

Once the database had been trained, each item was tested to see if a match could be found in the database. Each item was tested three times and the three positions are shown in figure 5.3. Two of the positions shown in figure 5.3 were previously used to train the items into the database. The third position (the position in the middle of the three positions in figure 5.3) was not used to train the object so it was expected that this would not be as easily recognised during the tests.

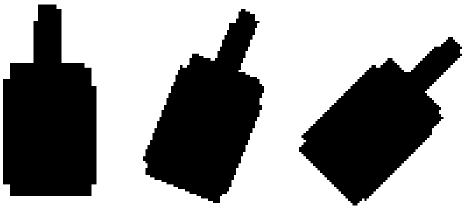


Figure 5.3 The three positions used to test each item.

5.2.1 Results

Only the front side of each item was tested so altogether there were a total of twenty four tests performed. Three of these tests could not find any match with the database. These three items were moved several millimetres and retried. These three items were successfully matched on the second attempt. Four other items were poorly recognised and upon retrying, these four items were matched with better accuracy. The diagram in figure 5.4 attempts to show the results of this experiment. The brown area is the area where correct matches were obtained for the items tested. It can be observed that the best results were achieved when the items were trained into the database between four and eight times. In the authors opinion it is not necessary to train an object in 8 times. Using the graph below as a reference it was decided that training an object four times is the optimum amount to train any item. These results are quite accurate but are by no means conclusive because only eight different objects were used. The final experiment in this chapter trains a larger amount of objects and the results there also suggest that training each item four times is sufficient.

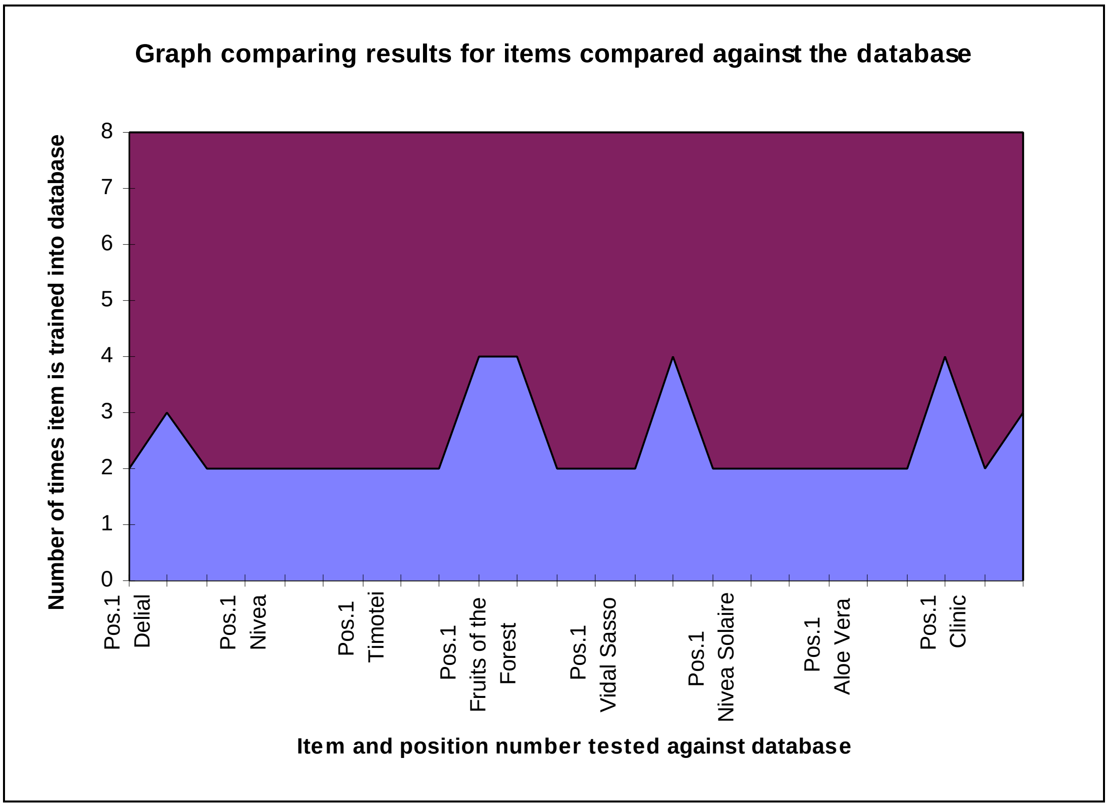


Figure 5.4 Graph showing results of recognised items trained into database in varying amounts. The brown region indicates the correct matches. Each item was tested from three different positions and only the first position for each object is marked on the graph.

5.2.2 Concluding remarks

A closeness of match was also obtained for every object tested. This closeness of match value was similar to a nearest neighbour estimation. The closeness of match value for any item was the sum of the differences between the measured data and the corresponding mean values of an item in the database. In general it was noticed that these values were lower when the object was tested in a position which was similar to a position the object was trained in, but were considerably larger when the object was tested in a position which it was not trained in. Using the *delial* bottle as an example, there was approximately a 75% increase on average of colour match values between the two positions tested. For shape values there was an average of approximately 67% increase between the two positions tested. This is shown graphically in the following two diagrams, figure 5.5 and figure 5.6.

The purpose of the following two graphs is to point out that it is more probable that a positive match will be found if an item is tested in a similar position to one which was used when training the object. This is not to say that MAVVIP is incapable recognising objects tested in unusual positions. The flexibility in the classification technique used by MAVVIP should in theory be able to cope with all positions. In reality there have been some occasions when positive matches were not found for objects. An example of this was the *Nivea Solaire* bottle which has a slightly curved surface and when this bottle was placed in a new position a large green blemish appeared on the bottle due to the lighting set-up. This blemish only appeared on the bottle while it was resting at one particular position. This blemish was significant enough that a positive match could not be found for the object. It must be pointed out that only a very small percentage of items are affected in this way.

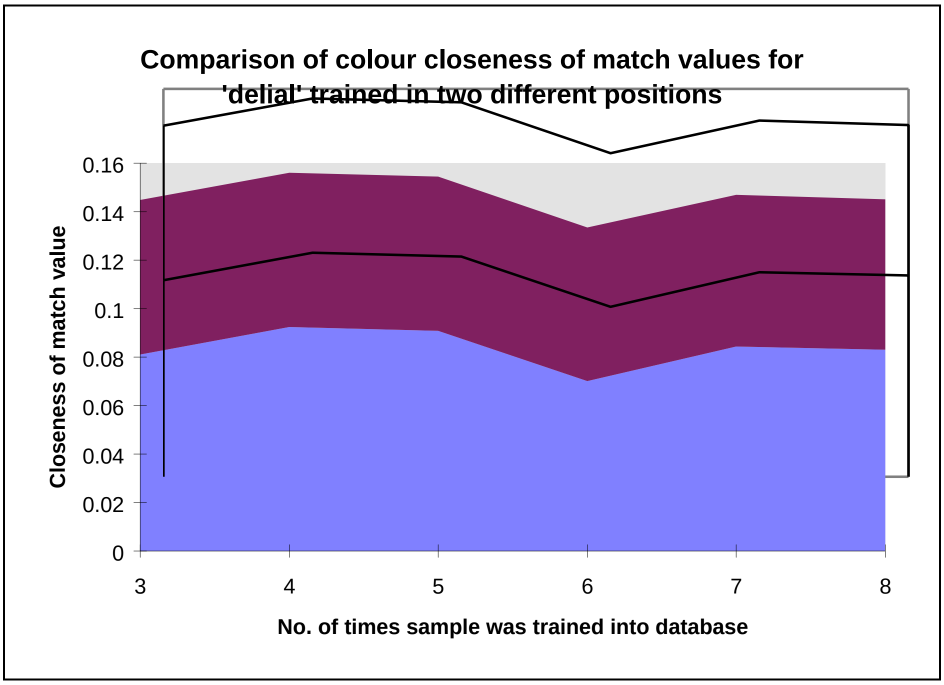


Figure 5.5 Graph showing the differences between colour match values of a *delial* bottle when tested in a position similar to one used while training the object compared with a position which had not previously been trained into the database.

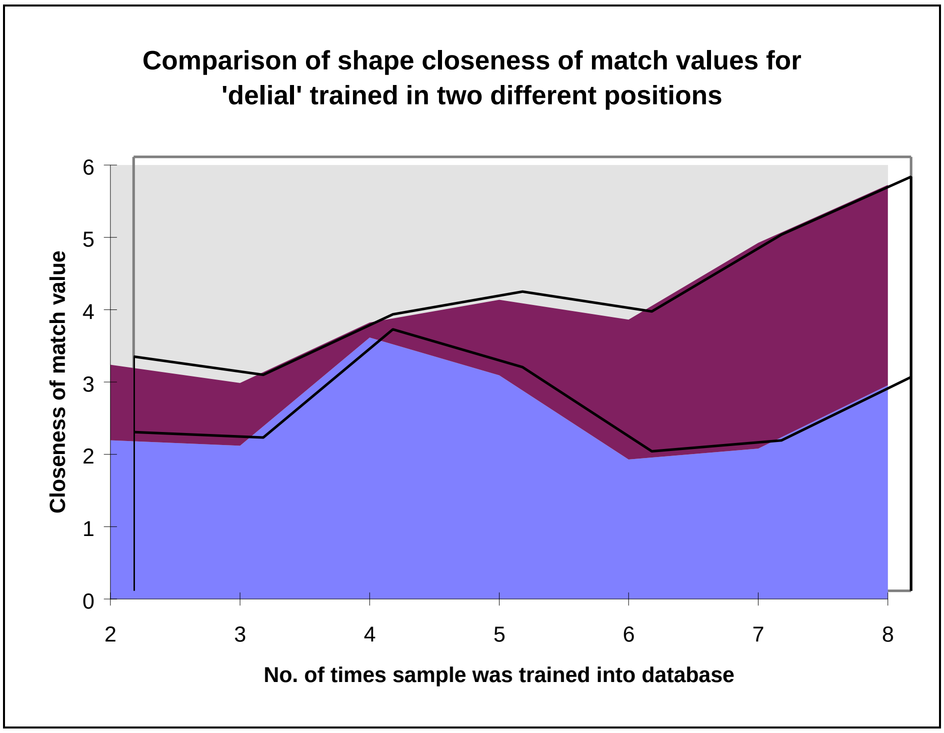


Figure 5.6 Graph showing the differences between shape match values of a *delial* bottle when tested in a position similar to one used while training the object compared with a position which had not previously been trained into the database.

5.3 Experiment 2 - Checking flexibility of MAVVIP for scale changes

The theory behind MAVVIP allows for an object to be enlarged or reduced in size and still be successfully recognised. It was necessary to test this theory to see in reality how well the theory worked. The reason for this flexibility is that if MAVVIP or a derivative of it was commercially available and was being assembled for use in different locations, this flexibility might be a necessity. If the system was being assembled at the user end or was being moved around then it is possible that the camera might be at slightly different distances from the objects it was attempting to recognise.

Figure 5.7 shows a picture of a *callcard* taken at two different zoom levels. This experiment was carried out in two stages to maximise the zoom range. Firstly the camera was zoomed out as far as possible (figure 5.7 (a)) and the objects were trained into the database. The camera was zoomed in (figure 5.7 (b)) and the same objects were tested against the database. Similarly when the camera was zoomed in (figure5.7 (b)) the objects were trained into the database and the camera was zoomed out (figure 5.7 (a)) and again the same objects were compared against the database.

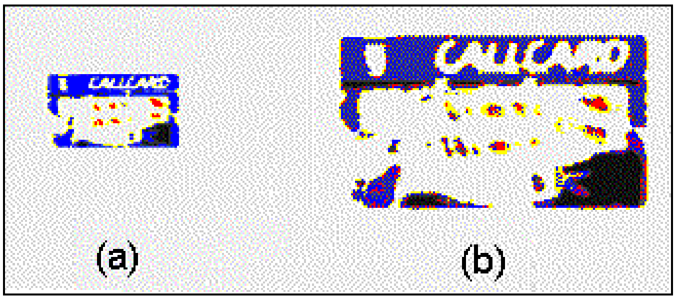


Figure 5.7 Pseudocolour image of *Callcard* taken at points of minimum and maximum zoom.

Eight credit cards were used for this experiment. Each credit card had both sides trained in four times. Focus was another variable which had to be considered for this experiment. When objects were to tested, they were tested first with the same focus as they were trained in with and they were also tested with a new focus which was most appropriate to their current zoom.

5.3.1 Results

The first experiment involved training each of the cards into the database when the camera was zoomed out (figure 5.7 (a)). The camera was zoomed in (figure 5.7 (b)) and the front and back of all cards was attempted to be recognised. There were six successful recognition's when the camera was unfocused and there were ten successful recognition's when the camera was focused.

Similarly the cards were zoomed in (figure 5.7 (b)) and trained into the database. The camera was zoomed out (figure 5.7 (a)), the cards were tested, and there were only three successful recognition's when the camera was unfocused and focused.

In all sixty-four tests were performed. In all cases the shape recognition matched the shapes correctly. And in total only twenty two tests out of sixty four matched the colours correctly.

5.3.2 Concluding remarks

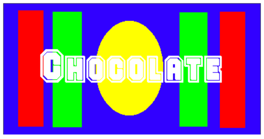
The reason for the poor recognition of colour information was due to the discrete nature of the CCD array. When zooming out from an item small patches of colours were reduced in size and occasionally disappeared. They were replaced by the more dominant colour which surrounded the colour patches in question. On manual inspection of the results it was observed that it was mostly minority colours which were being affected.

Improving recognition in such circumstances could be achieved by increasing the tolerances in the classification process. A better solution would be to incorporate a weighted tolerance system into the classification system. Higher tolerance values would be allocated to colours which take up small percentages of object surface areas.

5.4 Experiment 3 - Recognition of items elongated both vertically and horizontally

Another problem which exists is that of geometric invariance. If an item is not placed directly underneath the camera then the captured image might be slightly distorted. Therefore it was decided to study the effects of an item which was severely elongated, both horizontally and vertically. For the purposes of this experiment an item was created in an art package. This item was trained into the database and then stretched horizontally and vertically. The elongated versions were then tested. It must be pointed out that the items used may not accurately represent images distorted by geometric invariance but this test will provide an indication as to how the test will perform under geometric invariant positions.

The first test involved stretching the item vertically. Figure 5.8(a) shows the item trained into the database. Figure 5.8(b) shows the three items which were then tested during the first test and figure 5.8(c) shows the items tested during the second test. The *chocolate bar* was purposely designed using five different colours so colour recognition of the elongated bar could be observed also.



(a)

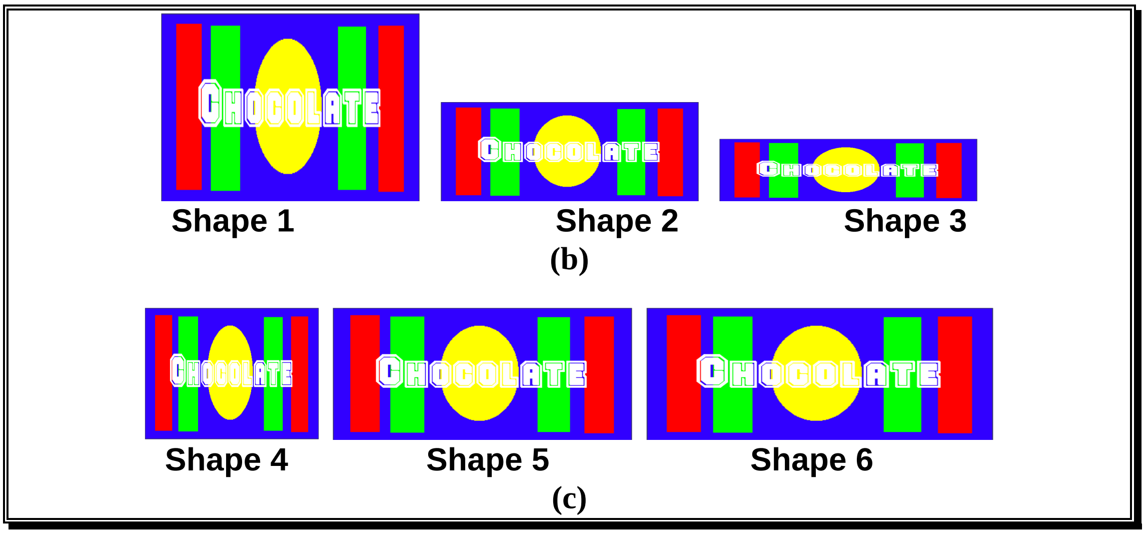


Figure 5.8 (a) The purpose built *chocolate bar* which was trained into the database. (b) The three vertically elongated bars which were tested. (c) The three horizontally elongated bars which were also tested.

5.4.1 Results

In all six tests there were no matches found for the shapes. Four of the six tests found matches for the colours. The four items whose colour was matched correctly are *shapes 1,2,5 and 6*. The colour on *shape 3* was not matched due to a 22% reduction in the surface area occupied by blue. Similarly the colour on *shape 4* could not be recognised due to a 18% decrease in the surface area occupied by blue and a 56% increase in surface area occupied by yellow.

5.4.2 Concluding remarks

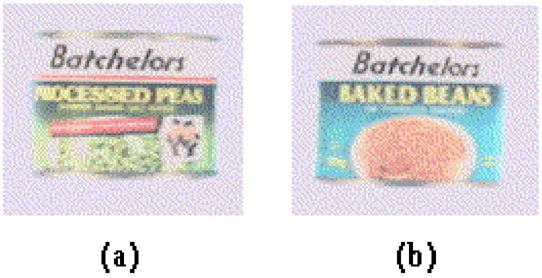
The elongation's of the items tested are extreme situations. MAVVIP will never be in a situation where it will have to recognise items which have been elongated by that magnitude. Even though MAVVIP will never have to deal with examples similar to those used in this experiment, this experiment was useful because it demonstrated how flexible the shape and colour recognition methods of MAVVIP are for elongated items.

There were two particular shape tests which prevented the shape recognition tests to match any of the distorted items. The tests were the ratio of maximum and minimum radii and the maximum length width ratio. A solution to this problem would be to simply increase the tolerances for these two particular features during the classification stage.

5.5 Experiment 4 - Cylindrical objects

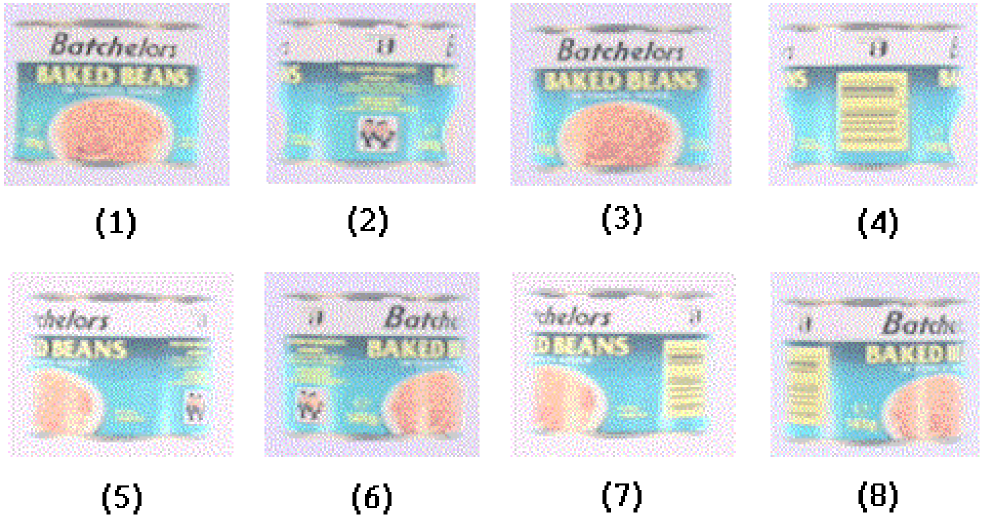
It was expected that cylinders would pose a problem for MAVVIP. Therefore it was decided to test MAVVIP out with some cylinders to see if it was capable of training and recognising cylinders. It was also necessary to see how feasible it is to train cylinders in with the rest of the objects in the database.

The two cylinders chosen for the experimentation were a tin of *batchelors beans* and a tin of *batchelors peas* and these are shown in figure 5.9. Each tin was trained in up to eight times. The eight positions used to train the object are shown in figure 5.10. The tins were trained in a similar fashion to the way items were trained in section 5.2. It was intended to find the optimum number of times to train cylinders into the database.



(a) (b)

Figure 5.9 (a) Tin of batchelors peas and (b) tin of batchelors beans.



(1) (2) (3) (4)

(5) (6) (7) (8)

Figure 5.10 The eight different positions in their correct order used to train all cylinders.

5.5.1 Results

After training the tins into the database they were both tested in eight different positions. In all sixteen tests the items were matched correctly. The results from the tests are shown graphically in figures 5.11 and 5.12. It can be seen from these graphs that the optimum number of times to train in the tins is six times. When the tins were trained in six times there were two tests out of the sixteen which also matched the tin of peas to a tin of beans.

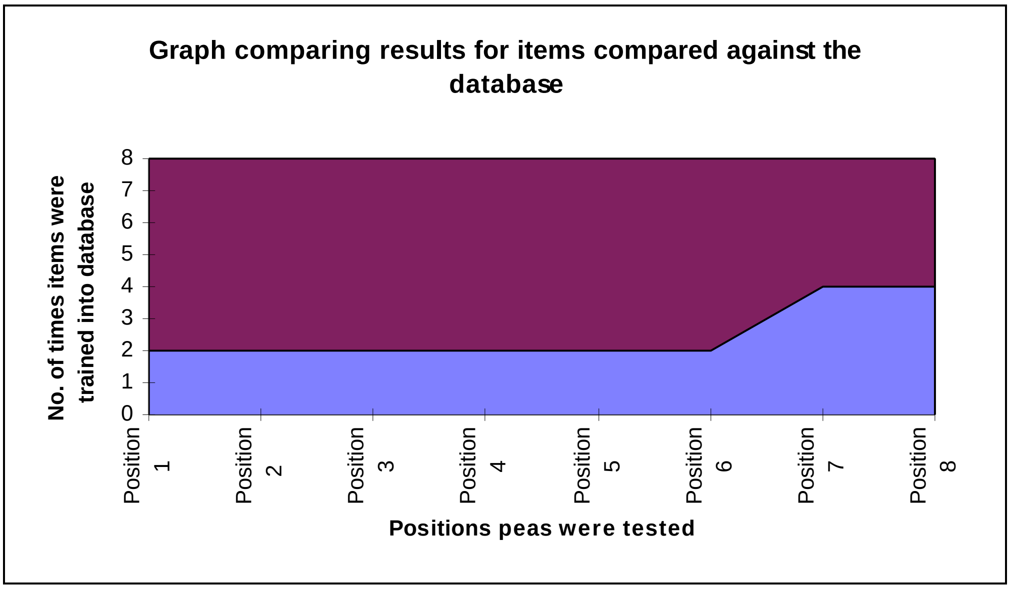


Figure 5.11 The area in brown shows the correct matches for the tin of peas.

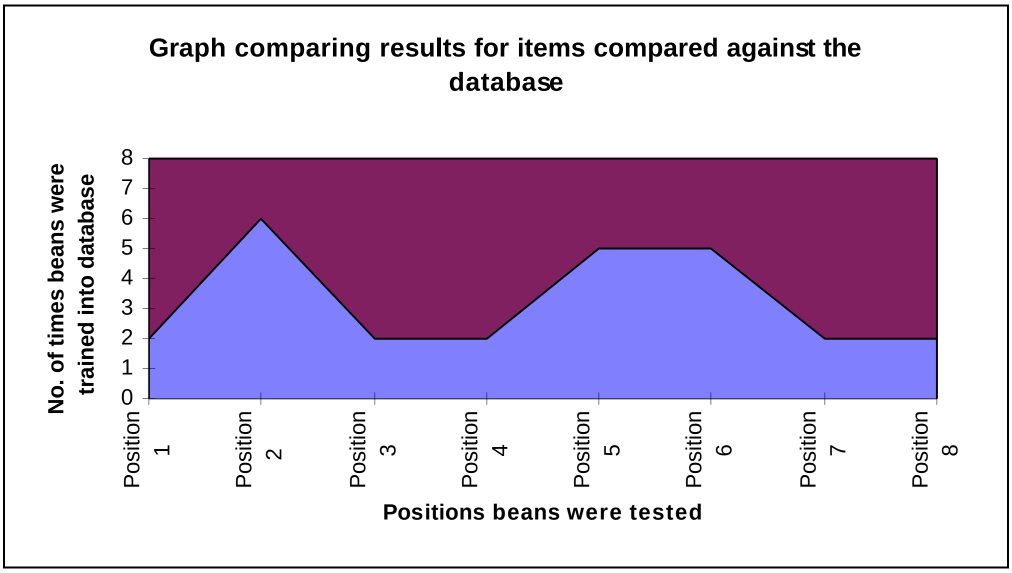


Figure 5.12 The area in brown shows the correct matches for the tin of beans.

5.5.2 Concluding remarks

As expected there were large standard deviations associated with the tins. This would present problems within a database of ordinary objects. Because of the large standard deviations, any object with colours similar to the tins could very easily be matched to the tins. Therefore, the shape recognition part of the system would play a more important role to discard the item but this was difficult because, like most items in the database, most cylindrical objects when viewed from the side will have rectangular silhouettes. This can be seen in figure 5.9 and figure 5.10.

The simplest method to overcome this problem would to reduce the tolerances allowed for cylindrical objects during classification. This solution would only work some of the time. An example of a situation where this would fail is described in the following paragraph.

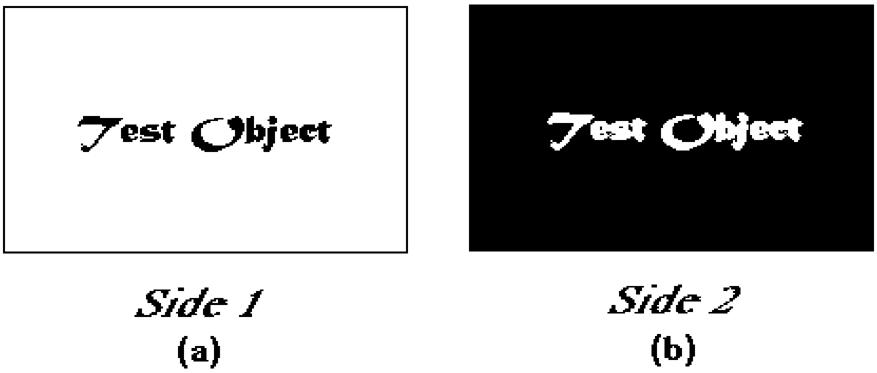


Figure 5.13 (a) *Side 1* is the image seen when looking at a cylinder from one side and (b) *Side 2* is seen looking at the opposite side of the same cylinder.

Take for example a cylindrical *test object* whose curved surface area is made up of two colours, black and white. The two views from opposite sides of the cylinder are shown in figure 5.13. Lets call them *Side 1* and *Side 2*. The breakdown of colours for *Side 1* is approximately 95% white and 5% black. Similarly the breakdown of colours for *Side 2* is 95% black and 5% white. The overall mean value measured for each of the colours is 50% of the surface area. The minimum tolerance required in order to successfully match either of the items is + or - 45% of the surface area. If this tolerance is reduced then neither side will be matched correctly.

Another point worth mentioning is that if the mean value of black and white trained into the database is 50% each and the tolerances for these colours is ±50% then any item which consists of black and white will be matched to this test object. This is another reason why it is not recommended that we train cylinders into the database unless a better method can be found.

Incorporating the nearest neighbour method into MAVVIP is another solution for reducing the number of matches with cylindrical objects such as tins of beans and peas. This method is discussed in more detail in section 5.6.3. In the authors opinion this method is far from perfect. The following example shows why this method is not full proof. Figure 5.14 shows two sample cylinders which cause a problem.

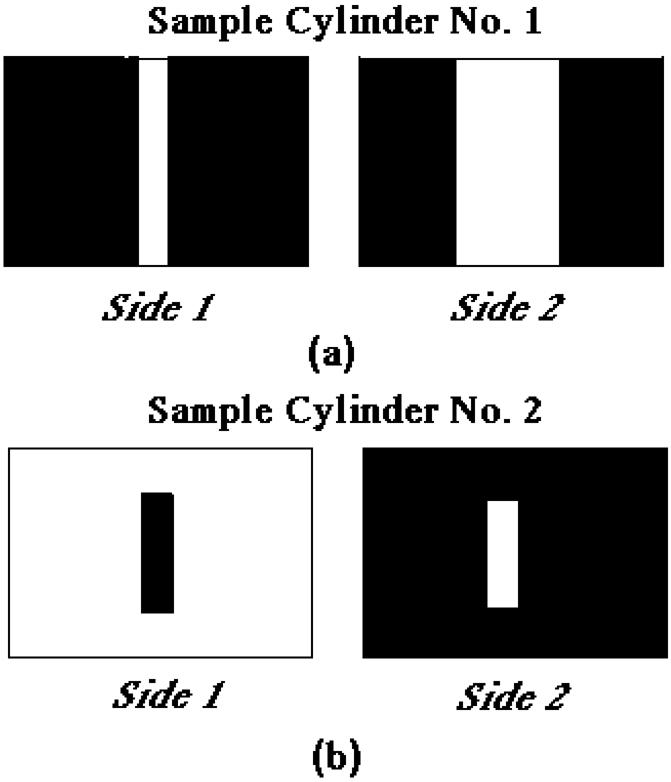


Figure 5.14 (a)(b) Two sample cylinders used to demonstrate conditions where nearest neighbour classification method fails.

Sample Cylinder No. 1

Side 1	white area	0.1	*Side 2*	white area	0.3
	black area	0.9		black area	0.7

mean white value = 0.2 mean black value = 0.8

Sample Cylinder No. 2

Side 2	white area	0.95	*Side 2*	white area	0.05

black area 0.05 black area 0.95

mean white value = 0.5 mean black value = 0.5

Assume the total area of each cylinder side (i.e. visible part in above diagrams 5.14(a)(b) is 1 unit in area.) is 1 unit and all measurements are expressed in terms of this unit value.

Figure 5.15 Statistics for the *sample cylinders*.

Assume *Side 2* of *Sample Cylinder No.2* is tested and values of 0.95 and 0.05 are obtained for black and white respectively. After performing the nearest neighbour algorithm it was observed that a nearest neighbour value of 0.3 was received when compared with *Sample Cylinder No. 1* as opposed to a nearest neighbour value of 0.9 when compared with *Sample Cylinder No. 2*. The lower the nearest neighbour value is, the more likely that object is similar to the object being tested. Using the nearest neighbour method on this occasion provided an incorrect answer.

There is a reliable and simple method of including cylindrical objects into the database and this method does not require any extra work. The solution is instead of assuming that the cylinder has one side, train the cylinder from several different viewpoints and assume each of these viewpoints is a different side. Therefore the cylinder is stored in the database as a several sided object.

This experiment with the cylinders was carried out again only this time assuming that the cylinders had four distinct sides as opposed to one side on the previous attempt. The first side consisted of training figure 5.10 (1)&(5) and side two consisted of training figure 5.10 (2)&(6) and so on. When comparing the results of the tin of beans from the two experiments there was an average sixteen percent reduction in standard deviation values and an average fifty four percent decrease in closeness of match values. This new method was a success because of the reduction in tolerance levels and a significant increase in accuracy. In the authors opinion, further experimentation of this method could improve the results considerably.

5.6 Experiment 5 - Testing groups of objects

The final experiment performed on MAVVIP was performed for the following three reasons.

- To see if, in a large database of items, there would exist any items or groups of items which would be matched as similar. An item matched to one or more items will be referred to as a double matched item for the rest of this chapter.
- To test the accuracy of MAVVIP on a large group of items after each item had been optimally trained into the database.
- To see if new problems would arise and how well MAVVIP could handle these problems.

There were forty-five items chosen for this test. They can be classified into the following groups :

- Common household products
- Credit, bank and identification cards
- Music CDs
- Music Cassettes
- Chocolate Bars
- Denominations of Money
- Stationary

There was a reason why specific groups of items were chosen in preference to forty-five random objects. Visually impaired persons can distinguish between a bar of chocolate and a credit card but they cannot distinguish between different credit cards or between different chocolate bars. Therefore this experiment also aimed to emphasize the fact that MAVVIP could distinguish between items which visually impaired persons have trouble recognising.

5.6.1 Results

Each item was trained into the database, front and back. Each object was then tested front and back with the exception of the stationary which had no front or back so was only tested once. In all 85 tests were carried out. The results of the eighty-five tests are shown in figure 5.16.

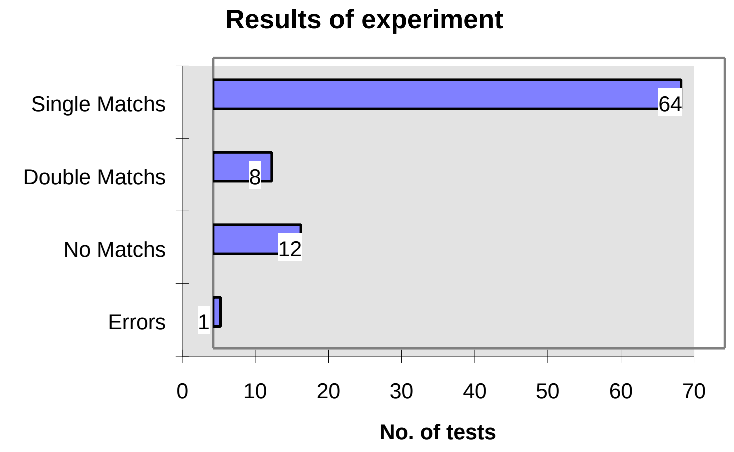


Figure 5.16 The results of the eighty-five tests carried out.

The twelve no match results and the incorrect match were retried. Of the thirteen retries the results are shown in figure 5.17.

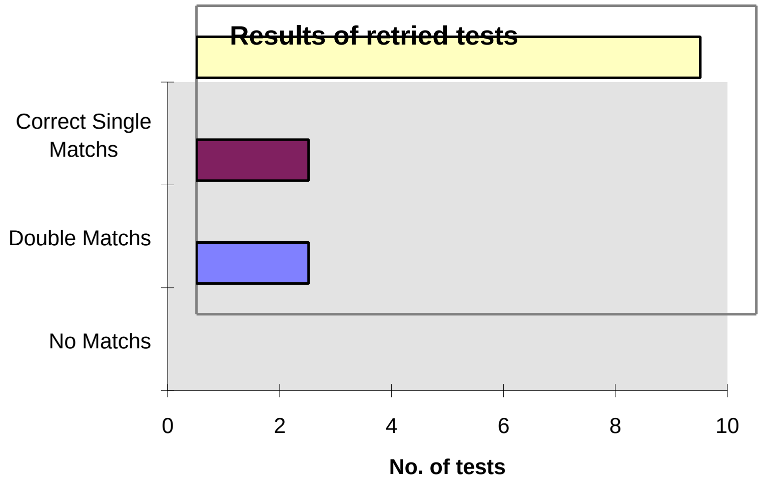


Figure 5.17 The results of the thirteen retried tests.

The two remaining no match tests were tried again. The first object was a snickers bar which was retried a further six times at different positions. A match was not found after eight attempts so further attempts were abandoned. The reason this item could not be matched is explained in section 5.7.1. The second no match item was a container of *aloe vera*. After a third attempt this was successfully matched to one of two objects (*aloe vera* or *black biro*). Matching of *aloe vera* and a *black biro* was regarded as unusual since there is very little colour similarity between the container of *aloe vera* and the *black biro*. Coincidentally it was the *black biro* which was the only incorrect match of this experiment. After further examination of the results it was observed that the three items mentioned above, the *snickers bar*, the *aloe vera* container and the *biro* each created different problems for MAVVIP. These three problems were previously unencountered and were unanticipated. The problems are explained in detail in section 5.7 below.

5.6.2 Effectiveness of MAVVIP with regard to double matches

The following is a list of all eleven double matches obtained during the final experiment.

The Hit Pack cassette	Popman cassette
U2 Zooropa cassette	Popman cassette
Twirl Bar	Black Biro
H5 Energy Bar	Black Biro
10 pound note	5 Pound Note
5 Pound Note	Train ticket
Blue Biro	Black Biro
Red Marker	Red Biro
Aloe Vera	Black Biro
Simon & Garfunkels Greatest Hits	Paul Simons Graceland cassette
Aloe Vera	Black Biro

Figure 5.18 Table displaying all double matches obtained after ninety-eight tests.

Of the eleven double matches, five items have been matched to *biros*. The transparent effects of the *biro* caused a problem in the recognition stage. This problem is discussed in detail in section 5.7.1. If this problem were corrected or improved this would mean a substantial reduction in double matches. Even despite this problem, a visually impaired person using a system similar to MAVVIP is not necessarily in trouble if that person gets a double match. For example if MAVVIP recognises an item as either a *twirl bar* or a *black biro* it is possible to distinguish between these items just by touch alone. Similarly this is true of the *H5 Energy bar* and a *biro,* a *marker* and a *biro,* a container of *aloe vera* and a *biro* and last but not least a *five pound note* and a *train ticket*.

Another set of double matches encountered were that of the backs of music cassette boxes. On three occasions there were double matches and it is not possible to distinguish between cassette boxes by touch. The similarities between music cassette boxes is shown in figure 5.19.

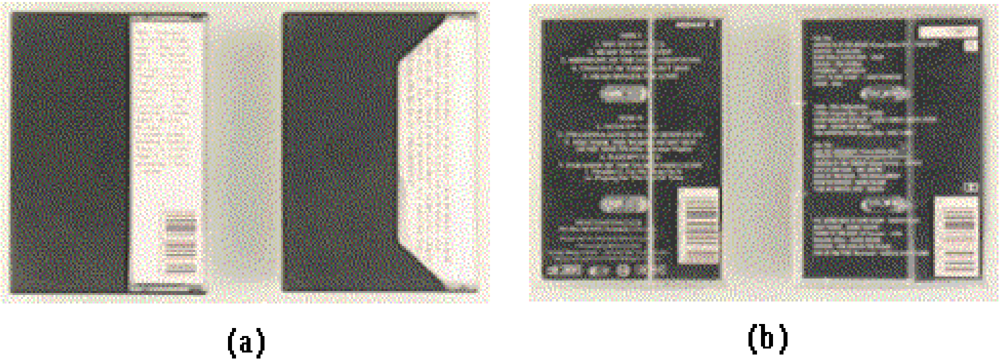


Figure 5.19 (a)(b) Two pairs of similar cassette boxes.

The solution to this problems is simply to turn the *cassette box* over and test the other side of it. Although *music cassettes* can have similar backs, their front covers are usually unique.

The two remaining double matches cannot be distinguished by touch alone. A *ten pound note* and a *five pound note* are quite similar in texture and size. Similarly a *blue biro* and *black biro* are identical to touch. These matches can be overcome by testing the items in question at slightly different positions or turning them upside down in the case of the money and examining the other side.

5.6.3 Comparison of closeness of match values

The closeness of match values were examined for each of the double match items. In all thirteen cases of double matches the colour closeness of match values were lower for the correct item. However after examining colour closeness of match values from all eighty-five tests, it was observed that six were incorrect.

If a visually impaired person was told that an item is one of two things, the author would not recommend using the colour closeness of match value to try and distinguish between the two. There is a 14.2% chance that the wrong answer would be given.

When dangerous substances are being recognised then this statistic is to high a risk especially when someone’s health is at stake. The colour closeness of match values could still be put to good use in a safe way. It is not recommended that this method replace the existing method for colour classification but that it assist it. The highest colour closeness of match value recorded during these experiments was 0.3649 for a *twirl bar*. So assume that a threshold is set up at approximately 0.4 and any items with a closeness of match value above this threshold should be discarded.

In the case of the shape closeness of match values, there were only nine out of the thirteen double matches where the shape closeness of match value was lower for the correct item. It is to be expected that the shape closeness of match value is not a reliable measure for distinguishing between objects. The reason is that many items have exactly the same shape so when trained in they should very have similar shape features. Take for example a credit card, depending on what angle the credit card is tested at it is equally likely to be closely related to any one of eight credit card sized items in the database. It would be useful however to use the shape closeness of match values to assist the existing classification system in the same way as described above for the colour values.

5.7 New problems arising from tests

5.7.1 Snickers Bar

The ends of chocolate bar wrappers such as a *snickers bars*, are rather flexible. MAVVIP can only cope with such bars if the flexible ends are moved slightly while training the object. The reason for this is that in order for MAVVIP to adapt to objects which can alter in shape these alterations have to be trained into MAVVIP. If the flexible ends are not moved around during training, MAVVIP can still recognise objects most of the time, but not always, as in the case of the *snickers bar*. When training in the shape, the colour is extracted from both sides but the shape is only extracted from one side because it is assumed that the shape is the same on both sides. In the case of the *snickers bar*, the ends were pointing slightly upwards during training. Therefore the shape information was extracted from a shape which had its ends facing upwards. While testing the object there was no problem with the object when it was facing upwards. However if the bar is facing downwards then the ends of the wrapper are facing downwards and this can create a different silhouette which is what happened to the *snickers bar*. This is shown in figure 5.20.

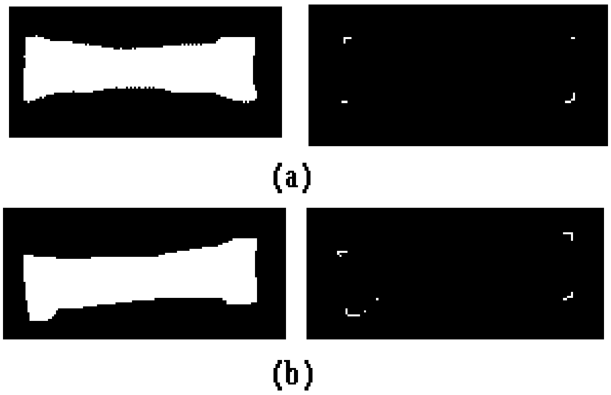


Figure 5.20 (a) Silhouette of *snickers bar* facing upwards and the four corners obtained from the corner detection algorithm and (b) the silhouette of the *snickers bar* facing downwards and the six corners obtained from the corner detection algorithm.

5.7.2 Aloe vera

The problem with the *aloe vera* container was a combination of two things.

- It contained a dense liquid which had very little viscosity.
- The container was an unusual shape.

Therefore if the container was laid to rest in a certain position for a length of time the liquid would rest in a set position. When laid out flat the liquid would remain in its old position for several minutes and as a result the container would rest in different ways producing different silhouettes depending on how the liquid had settled within the container. Figure 5.21 shows three different silhouettes produced by the *aloe vera* container.

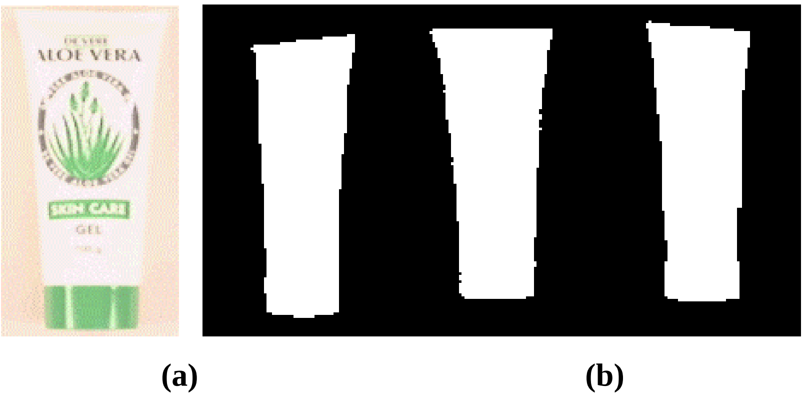


Figure 5.21 (a) *Aloe vera* container and (b) three different silhouettes obtained for the *aloe vera* container.

5.7.3 Biros

The glass section of the *biros* caused a problem. The *biros* tested were hexagonal in shape and the problem is caused by the transparency of the glass. Depending on what way the *biro* was resting, sometimes the camera could see the entire *biro* and other times only the lid of the *biro* was visible. The diagrams in figure 5.22 show some of the silhouettes obtained for *biros*. These huge differences in shape features create enormous standard deviations for classifying *biros*. This is similar to the problems faced for cylindrical objects as discussed in section 5.5.

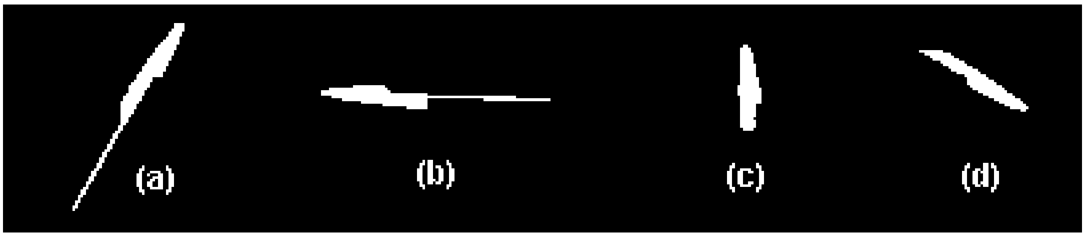


Figure 5.22 (a)(b)(c)(d) Examples of silhouettes obtained for *biros*.

This also has a similar knock on effect for the colour recognition of *biros*. Take the shape for figure 5.21 (c). A colour analysis of this object will observe that the entire area is blue (or black depending on the colour of the lid). Similarly a colour analysis of figure 5.21(b) will only have approximately half its area covered with blue.

5.8 Discussion

The four objectives of the experiment were achieved. These objectives are summarised below.

(1) Firstly it was observed that the optimum number of times an object should be trained in was four times.

(2) Secondly limitations were highlighted with regard to the following :

- MAVVIP only met with limited success when handling cylinders.
- Similarly the limitations were shown for recognising objects which had significant scale changes and elongations.

(3) Among the weaknesses are the following :

- The classification technique was not adaptable enough to account for the problems faced by scale changes, elongations and cylinders.
- Colour information was significantly altered through scale changes.
- Shape information was altered when shapes were elongated.
- There were three significant problems which arose unexpectedly, and those problems occurred with *biros, snickers bars* and *aloe vera* containers.

(4) There were two main suggestions for improving MAVVIP and they are as follows :

- In order to train cylinders into the database they must be considered as objects with several sides.
- The classification system could be altered to take into account scale changes.

The main disadvantage with the MAVVIP prototype was the speed. It requires approximately between a minute and a minute and a half to perform a recognition. The bigger the object and the bigger the database means that it requires more time for the recognition. It must be highlighted that this is a prototype and it is possible to create a faster more compact system.

It must be pointed out that of the forty five objects tested, twenty two did not have any barcodes. If this is a typical statistic for household items then this is a major disadvantage for the use of barcode readers.

There were two pairs of shape features used in the prototype which produce similar values. The pairs are :

- Ratio of maximum to minimum radii and the maximum length to width ratio
- Ratio between area of bays to area of object and the ratio of the area of the object to the area of the smallest rectangle which surrounds the object.

The results of MAVVIP were examined closely, paying particular attention to irregular shapes because they produce different results for all four features. For both pairs there were shapes disregarded during recognition based solely on one of the above features. This implies that all four of these features are essential to MAVVIP.

6. Conclusion

6.1 Overall conclusion

Overall the development of MAVVIP was successful and all the objectives were achieved. The results were very promising and a lot was learned from this project. Many unexpected results were obtained during the experimentation and several new methods for improving MAVVIP were suggested based on these experimental results. Several unexpected problems were encountered along the way and all were overcome successfully. The problems encountered provided an interesting analysis of the operation and limitations of MAVVIP. The experimental results obtained and the problems encountered might be of benefit to others carrying out similar research.

6.2 Advantages and disadvantages of MAVVIP

This section summarises some of the advantages and disadvantages of MAVVIP. The advantages are as follows :

- MAVVIP can recognise objects that don’t have barcodes(e.g. bills of money, credit cards).
- MAVVIP can extract colour information from an object.
- A vision system is required in order to scan text so this is another reason against the use of barcode readers.
- A vision system is the best way forward for reading LCD’s.

Some of the disadvantages of MAVVIP are as follows :

- At present MAVVIP is slow and expensive.
- MAVVIP is very sensitive to light.
- Sometimes an object is matched to more than one object in the database.
- Objects such as medicine bottles are similar in shape and colour. The only difference is the text but MAVVIP cannot read text.

6.3 Future directions of research

6.3.1 Short term goals

An ideal recognition device would be hand-held and this is not feasible in the short term. There are still too many problems which have to be overcome before such a device can be made portable. The software must be improved significantly first before progressing. Going from MAVVIP to a portable hand-held device is not a straightforward task. There is a considerable amount of development work still needed and there will be several intermediate stages of development before a hand-held device is produced.

In the author's opinion the next logical step would be to develop a stand specifically for the purposes of MAVVIP. This stand would contain a camera overlooking a miniature light table and would also contain controllable lights beside the camera which shine down on the bright surface underneath. The software for MAVVIP could be rewritten to run entirely from a laptop computer. A laptop computer and a customised stand would take up considerably less room than the existing prototype for MAVVIP.

A laptop computer is too heavy for anybody to carry around. Therefore this new system will not be feasible for visually impaired persons to carry around. Such a system would not be feasible in shops or banks because of the difficulty in locating this system. Take for example a supermarket. After taking an item from a shelf the visually impaired person would have to find the recognition system, identify the item, and if the item was not required the visually impaired person would have to find the shelf where the item came from and replace it. Repeating this process for every item on a shopping list would be a time consuming process.

Therefore the author recommends providing some visually impaired users with systems for use in the kitchen. It is also recommended that two further utilities be incorporated into the system and they are as follows :

- a barcode reader.
- the machine vision system should be adapted to recognise colours and patterns of clothing.

The software should have a database for keeping track of how often each of the three systems are being used. After the evaluation period is over statistics should be created from the database and this information along with feedback and suggestions from the visually impaired users should be taken into consideration for future development.

6.3.2 Long term goals

There are some more long term goals for MAVVIP but there are a considerable number of difficulties which must be overcome in order to achieve the final objectives. Some of these problems are as follows :

- MAVVIP is very sensitive to light and currently requires a dedicated lighting system and light table. If a hand-held device is constructed, it will not be possible to control lighting, background lighting or the position and distance the item is viewed from the camera.
- Reading text is another problem. The price and best before date are the two most important pieces of information about any item. It is feasible for a machine vision system similar to MAVVIP to be equipped with a device for reading text. Reading text would add a considerable amount of complexity to MAVVIP. Nevertheless in the author's opinion this is vital to the long term development. It is not likely that such a system would ever replace the current methods of recognition used by MAVVIP. The reason is that not all items have text (e.g. *biros*, *markers*) and visually impaired persons will always require colour information(e.g. identifying the colour of clothing).
- A user interface for MAVVIP is a difficult task. Visually impaired persons require easy access to MAVVIP. It must be simple to use for the training and recognition of items. A means of communication between the visually impaired person and the computer further complicates the problem. To recap, British statistics provided by John Gill(1993(b)) stated that less than 2% of legally blind people are capable of reading Braille and 35% of visually impaired people have hearing disabilities. There is no universal method of providing information which will help all visually impaired people so the best solution might be to provide a speech synthesiser, a Braille terminal and a screen enlargement facility. For hand-held devices, a Braille terminal and screen enlargements tools provide problems because of their size. These issues regarding the user interface require an extensive amount of research.
- LCD displays are another major problem facing visually impaired persons in this technological world. Most modern domestic appliances contain LCD's (Liquid Crystal Displays)(e.g. washing machines, cookers, microwave ovens, deep fat

friers, televisions, stereo systems). It is important for MAVVIP to adapt its software to read LCD displays as one of its long term development objectives.

In the authors opinion it will not be feasible to develop a reliable hand-held device which is capable of training and recognition until the above problems have been resolved.

It is recommended however that after a stationary system has been developed which works with a laptop computer then a prototype hand-held device be constructed. There are many ways of implementing this device but the simplest would be to have a hand-held device which transmits an image back to a central computer, the central computer will recognise the item, and transmit a speech signal back to the hand-held device for the user. This is an important strategy to take because research of MAVVIP will be costly and the eventual commercial prototypes will be quite or very expensive. In Britain over 68% of blind persons are elderly and of those who are not elderly many are living on disability benefits and overall not all visually impaired people will be in a financial position to afford such a system. Therefore the strategy will be to ‘initially aim’ for shop owners. If shops train all their items into a central database then these systems could be provided to visually impaired customers.

Previous aids designed for the visually impaired have on occasions become popular among others who do not have visual impairments(e.g. screen enlargers for personal computers). This has ultimately led to a more wide spread use and price reductions with such products. If a hand-held recognition system was in use in a shop, there is the possibility that customers with poor or failing vision might also find MAVVIP useful. If this were to happen and MAVVIP became popular, this would justify the cost of the system. This could create a market for MAVVIP which would mean more research and development, reduce the cost and eventually MAVVIP would become cheap, reliable and useful enough to be of significant benefit to the visually impaired community.

6.4 Practicality of MAVVIP

Developing customised hardware and software for MAVVIP is an expensive undertaking. Therefore the author’s recommends that every effort must be made to utilise technologies which have already been developed.

There are many pocket size computers capable of performing a variety of tasks which could be evaluated for MAVVIP. Even laptop computers are quite small and reasonably priced. There have been great improvements of late in the area of hardware devices specifically customised for speech synthesis, image processing, neural networks, digital signal processing and many other areas which could be incorporated into MAVVIP. Similarly there are many cameras available which could be useful to a system such as MAVVIP. An example of such a camera is SHARP's version of the Apple Newton which has a built in CCD (Charge Coupled Device) camera and is capable of capturing images as well as performing other tasks (CERTEC (Centre of Rehabilation Engineering), 1994).

Off the shelf software packages are available in areas such as speech synthesis, databases, classification and image processing techniques. Unfortunately image processing software on its own is not sufficient to recognise objects. These functions still have to be developed further to create MAVVIP. The problems of lighting and geometric invariance mentioned in the previous section are the main problems inhibiting the development of MAVVIP. Although these setbacks are specific to MAVVIP and off the shelf software packages will not be available to solve these problems, all is not lost. It is possible that similar work might have carried out by others institutions around the world. Whether the objectives were for manufacturing, military purposes, satellite imaging or other reasons it is possible that ideas and methods could be obtained elsewhere to overcome our problems.

The Lund Institute of Technology is currently working on a system called ISAAC (CERTEC (Centre of Rehabilation Engineering), 1994). It consists of SHARP's version of the Apple Newton and a shoulder bag which contains some customised hardware. The hardware contains a GPS (Global Positioning System) and cellular phone lines for voice and data communication. Although designed to help mentally disabled persons, ISAAC could also be used by blind people. With the help of a support centre, ISAAC users can show the operators pictures of their locations or objects they need help recognising and the operator in turn assists in telling the user their current location or the name of the objects in question. Although this system is not automated it provides an alternative to MAVVIP. In the authors opinion, ISAAC is just one of many alternative ways available for implementing MAVVIP.

The problems faced developing MAVVIP are significant and the solution to our lighting and geometric invariance problems may not necessarily be software based. In the author's opinion an innovative hardware device might be more likely to solve the lighting and geometric invariance problems before a software solution is found.

Although constructing MAVVIP from several hardware devices and software packages may mean that MAVVIP will be initially expensive, it will severely reduce developments costs which in turn will mean MAVVIP will be commercially viable sooner rather than later. Once an accurate prototype is developed, the task of customising the hardware and software with the aim of reducing the cost is more realistic.

6.5 Contributions to date

The following are some of the contributions this project has made to date :

- A speech synthesis package called *MacPROLOG Text to Speech* has been created for the MacPROLOG environment. To the best of the author's knowledge no other speech synthesis package is available for MacPROLOG.
- An extensive library of image processing functions has been created for MacPROLOG. These functions range from shape analysis, colour analysis, image acquisition and classification techniques.
- Significant progress has been made in the area of applying machine vision research to aid the visually impaired. Hopefully the interest created in this project might also encourage other research institutes to carry out similar research.

Appendix A - Experiment demonstrating shape recognition on circles and ellipses

A circle and two ellipses were used for an experiment to show that the shape recognition methods work for circular objects.

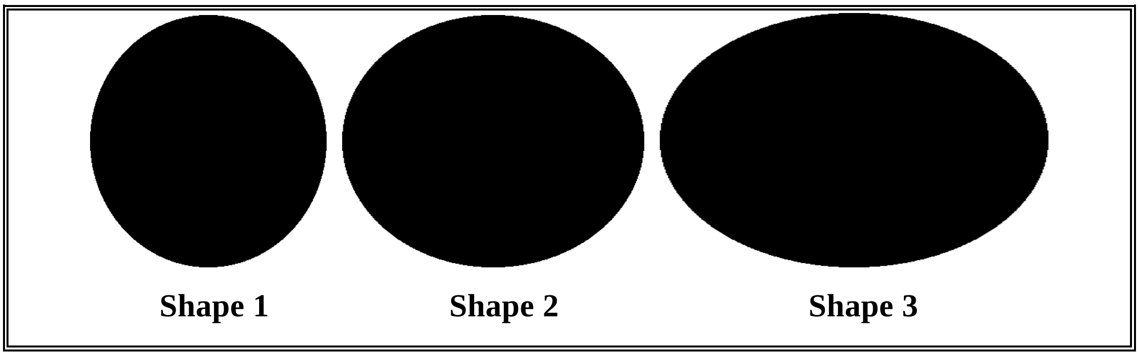


Figure 7.1 The circle and the two ellipses used in the experiment.

Each of the shapes was trained four times and then tested three times. All nine tests were correct. There were two single matches and seven double matches. The classification algorithm could not distinguish between *Shape 1* and *Shape 2*. This accounted for six of the double matches. The seventh double match occurred while testing *Shape 3*, and *Shape 2* was the second shape it was matched too.

A tolerance of six standard deviations was used during this test. A second identical test was performed with a tolerance of four standard deviations. There were only three double matches recorded as opposed to seven with the previous test.

It must also be pointed out that the aspect ratio of the CCD camera increases the difficulty of distinguishing between circles and ellipses. The aspect ratio can distort a circle in one position to look like an ellipse and an ellipse in another position to look like a circle. This must be taken into account when assessing MAVVIP on its ability to differentiate between circular and elliptical shapes.

References

Abror, F.E. Gill, J.M. "The needs of the developing countries." World Blind Union Research Committee. 1992; p.11 Priorities for Technical Research and Development for Visually Disabled Persons.

Adams, K. "Re: Marking Appliances." BlindFam Mailing List - 25 Jul 1995 to 26 Jul 1995;

Ansari N, Delp EJ. "Partial Shape Recognition: A Landmark-Based Approach." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1990;Vol. 12(No. 5):470-83.

Apple Computer Inc. *Inside Apple Macintosh.* 1986; Chap. 27, "Sound Manager." p. 473-502.

Apple Computer Inc. Apple Computer Inc. Software Licence. 1993.

Bala J, Wechsler H. "Shape analysis using genetic algorithms." *Pattern Recognition Letters* 1993;Vol. 14(No. 12):965-73.

Ballard DH. "Generalising the Hough Transform to Detect Arbitrary Shapes." *Pattern Recognition Letters* 1981;Vol. 13(No. 2):111-22.

Ballard DH, Brown CM. *Computer Vision.* Prentice-Hall, 1982; Chap. 8, "Representation of Two-Dimensional Geometric Structures." p. 231-63.

Batchelor BG. IFS (Publications) Ltd. 1985; Chap. 7, "Lighting and Viewing Techniques." p. 103-79.

Batchelor BG. Digital Image Processing in Industrial Applications, 1987;"Systems Considerations for Automated Visual Inspection." p. 15-32.

Batchelor BG. (a) *Intelligent Image Processing in Prolog.* Springer Verlag London Limited, 1991; Chap. 7, "Further Applications." p. 175-261.

Batchelor BG. (b) *Intelligent Image Processing in Prolog.* Springer-Verlag London Limited, 1991; Chap. 2, "Fundamentals of Prolog+." p. 15-76.

Batchelor BG. (c) *Intelligent Image Processing in Prolog.* Springer-Verlag London Limited. 1991; Chap. 8, "Advanced Inspection Techniques." p. 263-316.

Batchelor BG and Whelan PF. "Generalisation Procedures for Colour Recognition." SPIE Press; Boston USA. Sept 1993; *Machine Vision Applications, Architectures, and Systems Integration II.* Vol. 2046: 36-46.

Batchelor BG, Whelan PF. "Real-time colour recognition in symbolic programming for machine vision systems." *Machine Vision and Applications* 1995;Vol. 8(No. 6):385-98.

Bender T. Tex-Edit [computer program]. Ftp://archive.umich.edu/mac/sound/speech/texedit2.0.sit.hqx: 1994;

Blair, B.E. "Re: Marking Appliances." BlindFam Mailing List - 3 Aug 1995 to 4 Aug 1995.

Blenkhom P. “Designing products that speak:lessons from talking systems for blind people.” *Computing and Control Engineering Journal*. 1994;

Blind-L Mailing List. Listserv@uafsysb.uark.edu.

Blind News Digest Mailing List. LISTSERV@vm1.nodak.edu: Bill McGarry.

Boyle RD, Thomas RC. *Computer Vision : A First Course.* Blackwell Scientific Publications, 1988; Chap. 4, "Low Level Processing." p. 32-53.

Bryant, G.Y. "Re: Methods for Labeling Cans, Boxes and Jars." BlindFam Mailing List - 22 Jan 1996 to 23 Jan 1996.

Celenk M. "A Color Clustering Technique for Image Segmentation." *Computer Vision, Graphics, and Image Processing* 1990;Vol. 52:145-70.

CERTEC (Centre of Rehabilation Engineering) “ISAAC A Personal Digital Assistant for the Differently Abled” Sweden. Lund Institute of Technology. Lund University. 1994;

Chang Y, Leou J. "A Model-based approach to representation and matching of object shape patterns." *Pattern Recognition Letters* 1992;Vol. 13(No. 10):707-14.

Chorafas DN. *Knowledge Engineering.* Van Nostrand Reinhold, 1990; Chap. 10, "The Mathematics of Uncertainty." p. 191-212.

Clarke, K. "Colour Description for the Visually Impaired." Dublin City University. 1993;

Cook, V.A. "Re: Marking Appliances." BlindFam Mailing List - 25 Jul 1995 to 26 Jul 1995.

Cootes TF, Cooper DH, Taylor CJ, and Graham J. "A Trainable Method of Parametric Shape Description." Springer-Verlag; London. *Proceedings of the British Machine Vision Conference*. 1991; p. 55-61.

Cootes TF, Taylor CJ, Cooper DH, and Graham J. "Training Models of Shape from Sets of Examples." Springer-Verlag; London. *Proceedings of the British Machine Vision Conference.* 1992; p. 9-18.

Crabb, N. "Re: Marking Appliances." BlindFam Mailing List - 25 Jul 1995 to 26 Jul 1995.

Davies ER. (a) *Machine Vision: Theory, Algorithms, Practicalities.* Academic Press, 1990; Chap. 1, "Vision, the Challenge." p. 1-18.

Davies ER. (b) *Machine Vision: Theory, Algorithms, Practicalities.* Academic Press, 1990; Chap. 7, "Boundary Pattern Analysis." p. 168-88.

Davies ER. (c) *Machine Vision: Theory, Algorithms, Practicalities.* Academic Press, 1990; Chap. 8, "Line Detection." p. 191-206.

Davies ER. (d) *Machine Vision: Theory, Algorithms, Practicalities.* Academic Press, 1990; Chap. 9, "Circle Detection." p. 207-39.

Davies ER. (e) *Machine Vision: Theory, Algorithms, Practicalities.* Academic Press, 1990; Chap. 12, "Polygon Detection." p. 284-300.

Davies ER. (f) *Machine Vision: Theory, Algorithms, Practicalities.* Academic Press, 1990; Chap. 14, "Corner Detection." p. 323-40.

Davies ER. (g) *Machine Vision: Theory, Algorithms, Practicalities.* Academic Press, 1990; Chap. 15, "Abstract Pattern Matching Techniques." p. 345-68.

Duran, P. Braille Research and Literary Inc. "The Need for an Integrated Tactile System - A Call for Reason and Action." Cambridge. 1995;

EASI Mailing List. LISTSERV@sjuvm.stjohns.edu: EASI.

Edwards S. “Microcomputers and the Visually Impaired (Low-vision to No-vision)”. *OCLC Micro*. 1989; Vol. 5(No. 6)20-25.

Fitzgerald G. Welcome ! [computer program]. Ftp://archive.umich.edu/mac/sound/speech/welcome1.32.sit.hqx: 1993.

Ford, K. "Re: Marking Appliances." BlindFam Mailing List - 12 Dec 1995 to 13 Dec 1995.

Ford, K. "Re: Methods for Labeling Cans, Boxes and Jars." BlindFam Mailing List - 21 Jan 1996 to 22 Jan 1996.

Foulke, E. "Priorities for Development of New Audio Technology." World Blind Union Research Committee. 1992; *Priorities for Technical Research and Development for Visually Disabled Persons.* p. 30-40.

Fowle, L. "User-Evaluation of a Barcode Reader for Visually Impaired People - Results and Discussion." London. Royal Nation Institute for the Blind Information Services Department. 1995;

Fukunaga K. (a) *Statistical Pattern Recognition.* Academic Press, 1990; Chap. 4, "Parametric Classifiers." p. 123-80.

Fukunaga K. (b) *Statistical Pattern Recognition.* Academic Press, 1990; Chap. 1, "Introduction." p. 1-10.

Gibbons RJ and Williams DJ. "Colour and Texture Analysis for Automated Sorting of Eviscera." London; Springer-Verlag. *Proceedings of the British Machine Vision Conference.* 1991; p. 327-330.

Gill, J.M. (a) "A Vision of Technological Research for visually disabled people." 1993;

Gill, J.M. (b) "Access to Graphical User Interfaces by Blind People." GUIB Consortium. 1993;

Gill, J.M. and Peuleve, C.A. "Research Information Handbook of Assistive Technology of Visually Disabled Persons." The Tiresias Consortium. 1993;

Gill, J.M. "Introduction." World Blind Union Research Committee. 1992; p.2 Priorities for Technical Research and Development for Visually Disabled Persons.

Gill, J.M. Seminar by Dr. John Gill of the Royal National Institute of the Blind entitled Technology and the Visually Impaired. 1994.

Harrell RC, Slaughter DC, Adsit PD. "A Fruit-Tracking System for Robotic Harvesting." *Machine Vision and Applications* 1989;Vol. 2:69-80.

He Y, Kundu A. "2-D Shape Classification Using Hidden Markov Model." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1991;Vol. 13(No. 11):1172-84.

Heumann, J.E. "NIDRR Research Grant Info." EASI Digest mailing list. NIDRR (National Institute on Disability and Rehabilitation Research Office of Special Education and Rehabilitative Services Department of Education Washington D.C.20202). 1994;

Hill DA. IFS (Publications) Ltd. 1985; Chap. 4, "Illumination Equipment." p. 39-59.

Hogg DC. "Shape in Machine Vision." *Image and Vision Computing* 1993;Vol. 11(No. 6):309-16.

Holland SW, Rossol L, Ward MR. "Consight - I: A Vision-Controlled Robot System for Transferring Parts from Belt Conveyors." 1979; p. 81-100.

Huang C, Chen C. "Object identification using modified distributed associated memory." *Pattern Recognition Letters* 1992;Vol. 13(No. 8):569-79.

ISD (Information Storage Devices). "Applications Notes and Design Manual." 1994;

Jain AK. (a) *Fundamentals of Digital Image Processing.* Prentice Hall, 1989; Chap. 3, "Image Perception." p. 49-77.

Jain AK. (b) *Fundamentals of Digital Image Processing.* Prentice Hall, 1989; Chap. 7, "Image Enhancement." p. 262-4.

Jain AK. (c) *Fundamentals of Digital Image Processing.* Prentice Hall, 1989; Chap. 9, "Image Analysis and Computer Vision." p. 414-23.

James M. *Pattern Recognition.* BSP Professional Books, 1987; Chap. 4, "The Frequency Approach." p. 63-83.

Johns N. *MacPROLOG 3.0 Reference Manual.* LPA Ltd. 1990;

Kay PL. "Electronic aids for blind persons: an indisciplinary subject." *IEE Proceedings* 1984;Vol. 131(No. 7):559-75.

Lauer H. “Why One Medium Isn't Enough”. *OCLC Micro*. 1989; 5(6). 22-25.

Leavers VF. (a) *Shape Detection in Computer Vision Using the Hough Transform.* Springer-Verlag London Limited; 1992;

Leavers VF. (b) "Use of the Radon transform as a method of extracting information about shape in two dimensions." *Image and Vision Computing* 1992;Vol. 10(No. 2):99-107.

Lee H, Park R. "Relaxation algorithm for shape matching of two dimensional objects." *Pattern Recognition Letters* 1989;Vol. 10(No. 5):309-13.

Levine MD. (a) *Vision in Man and Machine.* McGraw Hill, 1985; Chap. 10, "Shape." p. 480-544.

Levine MD. (b) *Vision in Man and Machine.* McGraw Hill, 1985; Chap. 7, "Color." p. 313-70.

Lieberg, M. "Re: Methods for Labeling Cans, Boxes and Jars." BlindFam Mailing List - 21 Jan 1996 to 22 Jan 1996.

Lisboa PJG. *Neural Networks Current Applications.* Chapman & Hall, 1992; Chap. 1, "Introduction." p. 1-34.

McGowan, T. "A Text to Speech synthesiser for the MacPROLOG environment." Dublin City University. 1994;

Mendham, J. "Labeling cans boxes etc.." BlindFam Mailing List - 22 Jan 1996 to 23 Jan 1996.

Mertzios BG, Tsirikolias K. "Statistical shape discrimination and clustering using an efficient set of moments." *Pattern Recognition Letters* 1993;Vol. 14(No. 6):517-22.

Mingfa Z, Hasani S, Bhattarai S, Singh H. "Pattern recognition with moment invariants on a machine vision system." *Pattern Recognition Letters* 1989;Vol. 9(No. 3):175-80.

Molloy D, McGowan T, Clarke K, McCorkell C, and Whelan P. "Application of machine vision technology to the development of aids for the visually impaired." SPIE; 1994; *Machine Vision Applications, Architectures, and Systems Integration III, Proceedings.* SPIE 2347, 31 Oct-2 Nov, Boston USA. p. 59-69.

Murtha, C.A. "Re: Methods for Labeling Cans, Boxes and Jars." BlindFam Mailing List - 21 Jan 1996 to 22 Jan 1996.

Nelson MM, Illingworth WT. *A Practical Guide to Neural Nets.* Addison-Wesley Publishing Company, 1990; Chap. 2, "Net Questions: What and Why?" p. 14-25.

NIDRR (National Institute on Disability and Rehabilitation Research Office of Special Education and Rehabilitative Services Department of Education Washington D.C.20202). "Low Vision Rehabilitation." Rehab Brief. 1992; Vol. XIV, No. 4.

Novini A. "Optics, Illumination and Image Sensing for Machine Vision III." LumenX Company of Alltrista Corp. Proc. SPIE. 1988; Vol. 1005 p.131-136.

Okada S, Imade M, and Miyauchi H. "Automatic Identification of Conveyer-Transferred Parts through Image Data Processing." *The Industrial Electronics Society (IES) of IEEE*; Bologna, Italy. Sep. 1994; Vol. 2 p. 719-722.

Pao DCW, Li HF, Jayakumar R. "Shape Recognition Using the Straight Line Hough Transform: Theory and Generalisation." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1992;Vol. 14(No. 11):1076-89.

Parker JR. *Practical Computer Vision using C.* John Wiley & Sons, Inc. 1994; Chap. 4, "Classifying and Recognising Objects." p. 243-85.

Paul Mimms. "Re: Methods for Labeling Cans, Boxes and Jars." BlindFam Mailing List - 21 Jan 1996 to 22 Jan 1996.

Pavlidis T. "A Review of Algorithms for Shape Analysis." *Computer Graphics and Image Processing* 1978;Vol. 7:243-58.

Raman, T.V. "Identify Dollars for the Blind." The Blind News Digest. 1994; Issue #1072.

Ratcliffe, V. "Re: Methods for Labeling Cans, Boxes and Jars." BlindFam Mailing List - 22 Jan 1996 to 23 Jan 1996.

Rimey R, Brown C. ; Blake A, Yuille A, editors.*Active Vision.* Massachusetts Institute of Technology, 1992; Chap.14, "Task-Oriented Vision with Multiple Bayes Nets." p. 217-38.

Rundle, C. "Food Labeling Project: Initial Report." London. Royal National Institute for the Blind Information Services Department. 1995;

Ryall G, Sandor J. "Statistical Pattern Matching." *Pattern Recognition Letters* 1989;Vol. 9(No. 3):163-8.

Samwick, M. "Identify Dollars for the Blind." Blind News Digest. 1994; Issue #1072.

Schroeder HE. "Practical Illumination Concept and Technique for Machine Vision Applications." Robotics International of the Society of Manufacturing Engineers; *Robots 8 Conference*. 1984; Vol. 14. p. 27-43

Sekita I, Kurita T, Otsu N. "Complex Autoregressive Model for Shape Recognition." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1992;Vol. 14.(No. 4):489-96.

Shamsi A. Recite [computer program]. Ftp://archive.umich.edu/mac/sound/speech/recite1.0.sit.hqx: 1993.

Sheela BV, Rajagopal C, Padmanabhan K. "TEMPO - template matching by parametric optimization." *Pattern Recognition Letters* 1992; Vol. 14(No. 1):65-9.

Sirjani A, Cross GR. "On representation of a shape's skeleton." *Pattern Recognition Letters* 1991;Vol. 12(No. 3):149-54.

Sonka M, Hlavac V, Boyle R. (a) *Image Processing, Analysis and Machine Vision.* Chapman & Hall, 1993; Chap. 6, "Shape representation and description." p. 192-253.

Sonka M, Hlavac V, Boyle R. (b) *Image Processing, Analysis and Machine Vision.* Chapman & Hall, 1993; Chap. 7, "Object Recognition." p. 255-315.

Strachan NJC. "Recognition of fish species by colour and shape." *Image and Vision Computing* 1993;Vol. 11(No. 1):2-10.

Takeuchi K. Informer 0.1 B2 for Macintosh [computer program]. Ftp://archive.umich.edu/mac/sound/speech/informer0.1b2.cpt.hqx: 1993.

Tou JT, Gonzalez RC. *Pattern Recognition Principles.* Addison-Wesley Publishing Company, 1974; Chap. 6, "Trainable Pattern Classifiers - The Statistical Approach." p. 217-42.

Travis D. *Effective Color Display: theory and practice.* Academic Press, 1991; Chap. 2, "The Visual System." p. 32-67.

Trong Quoc, L. "Identify Dollars for the Blind." Blind News Digest. 1994; Issue #1070.

Uber GT, Harding KG. "Illumination and Viewing Methods for Machine Vision." *Machine Vision Systems Integration* 1991;Vol. CR36(SPIE Critical Review):20-33.

Vernon D. (a) Anonymous*Machine Vision : automated visual inspection and robot vision.* Prentice Hall, 1991; Chap. 6, "Image Analysis." p. 118-38.

Vernon D. (b) *Machine Vision : Automated Visual Inspection and Robot Vision.* Prentice Hall, 1991; Chap. 7, "An Overview of Techniques for Shape Description." p. 140-55.

Ward TA. *Programming C on the Macintosh.* Scott, Foresman and Company, 1986; Chap. 11, "Megaroids - A Large-Scale Macintosh Application." p. 352-85.

Watte J. Talking Clock Pro [computer program]. Ftp://archive.umich.edu/mac/sound/speech/talkingclockpro2.obo.cpt.hqx: 1993.

Wechsler H, Zimmerman L. "2-D Invariant Object Recognition Using Distributed Asociative Memory." *IEEE Transactions on Pattern Analysis and Machine Inteligence* 1988;Vol. 10(No. 6):811-21.

Weidl E. So to Speak [computer program]. Ftp://archive.umich.edu/mac/sound/speech/sotospeak1.0.sit.hqx: 1993.

Weidl E. SpokesDaemon and Spiricom [computer program]. Ftp://archive.umich.edu/mac/sound/speech/spokesdaemon1.01.cpt.hqx: 1993.

Whelan, P.F. "Machine Vision - A Basic Introduction to Industrial Vision Systems." Dublin City University. 1992.

Widrow B, Winter R. ; Zornetzer SF, Davis JL, Lau C, editors.*An Introduction to Neural and Electronic Networks.* Academic Press, 1990; Chap. 12, "Neural Nets for Adaptive Filtering and Adaptive Pattern Recognition." p. 249-71.

Publications arising from this project

"Automated recognition and classification for the visually impaired"

Tommy McGowan, Paul F. Whelan, Charles McCorkell

Ecarte 3; 1995*; European Conference on the Advancement of Rehabilitation Technology - ECART Proceedings*, 10-13 October 1995, Lisbon Portugal p. 41-43.

"Application of machine vision technology to the development of aids for the visually impaired."

Derek Molloy, Tommy McGowan, Kevin Clarke, Charles McCorkell, and Paul F. Whelan.

SPIE; 1994; *Machine Vision Applications, Architectures, and Systems Integration III, Proceedings.* SPIE 2347, 31 Oct-2 Nov, Boston USA. p. 59-69.