R&D REPORT NO. 31

Intelligent Vision System for Real-time Automatic Inspection of Alimentary Products (ALINSPEC)

September 1996

Campden & Chorleywood
Food Research Association

Campden & Chorleywood
Food Research Association

Director-General
Prof. C. Dennis
BSc, PhD, FIFST

Chipping Campden
Gloucestershire
GL55 6LD UK
Tel: +44 (0) 1386 842000
Fax: +44 (0) 1386 842100

R&D Report No. 31

Project No. 12978

Intelligent Vision System for Real-time Automatic Inspection of Alimentary Products (ALINSPEC)

Compiled by Ade Ade-Hall

September 1996

Preface

This report is an edited version of the project's final technical report submitted to the EC by the project co-ordinator. The report includes contributions from the following:

- Consorzio CEO (Centro Eccellenza Optronica), Italy (project co-ordinator)
- Officine Galileo, Italy
- Coren-Frigolouro, Spain
- Campden & Chorleywood Food Research Association, UK
- Bull, France
- Florence University, Italy
- Vigo University, Spain
- Strathclyde University, Scotland

VISION SYSTEM FOR REAL-TIME

AUTOMATIC INSPECTION OF ALIMENTARY PRODUCTS

(ALINSPEC)

EXECUTIVE SUMMARY

compiled by

Ade Ade-Hall

September 1996

Introduction

ALINSPEC has resulted in the building of a "toolbox" of image analysis based techniques for the real time, on-line quality assessment of meat, demonstrating them on pig legs and chickens. The developed techniques have value, but their limitations and especially their costs must be taken into account.

The development of ALINSPEC was supported by the European Commission under contract BRE2-CT92-132 of the Brite-Euram programme. Campden & Chorleywood Food Research Association (CCFRA) was a member of the consortium responsible for developing ALINSPEC. CCFRA's work in the project was partially supported by the UK Meat and Livestock Commission.

The project began in October 1992 and was completed in March 1996.

Objectives

The objectives of the project were to develop techniques (architecture, image and data processing software, and expert system) to:

- look at the meat and perform 100% on-line examination using multiple existing sensor technologies (visible, x-ray, Infrared)
- recognise features (fractures, bruises, shape ...)
- deduce significance, taking into account other information (weight, plunge probe, production schedule ...)
- deduce and initiate appropriate action (classify meat, issue warnings ...)

and to construct industrial demonstrators working on pig meat and chickens.

The demonstrators were to operate on pig legs, deciding their suitability for Spanish dry cured ham, and on chickens, assigning a quality grade. However, for pig legs, the priority was not in the recognition of meat features required for on-line examination, but in the recognition of features which would be beneficial to those developing systems for automating the pig butchery process.

The demonstrators were for development, proving, and demonstration purposes only, and were not to take the form of prototypes. They were to recognise and reason on the basis of features generally relevant to the assessment of pig meat and poultry, demonstrating the long term and broad applicability of the tools and techniques developed.

Another objective was to find ways in which cheap industrial versions of the demonstrator could be used to deliver the performance required by the meat industry.

Achievements

Typical machine vision systems measure properties of an object such as colour, dimensions, shape, and surface texture. Where the objects are rigid and well defined, as in most manufactured products, deducing quality is easy: the measurements taken are simply compared with pre-determined "perfect" reference values. For natural objects, such as meat carcasses, it is more difficult to reproduce the judgement of a human expert.

In the absence of a "perfect" set of reference values, simple measurements were not enough. Multiple defects and features with ill-defined sizes, shapes and colours must be recognised against a background which is itself heterogeneous. The absence, or presence, and nature of these features must be balanced against subjective requirements to deduce quality in a manner typical of humans.

At the end of the project, a collection of "tools" were developed for assessing the quality of natural products, in particular chicken carcasses and pig legs. A flexible, adaptable, high speed demonstrator system was built to examine chicken carcasses in a manner similar to humans.

Tools were developed for acquiring images of meat samples using multiple existing sensor technologies such as visible (RGB), infrared and x-ray. Studies of the most appropriate sensors were carried out and their application in other areas within the food industry were considered, especially within the meat industry. Safety issues and the effect of production environments on the sensors were also considered. Although cheap "off the shelf" infrared and x-ray sensors were not available, the cost of including these sensors was reduced by designing alternative sensors which made use of the components from existing "off the shelf" sensors.

Effective and generally applicable techniques were developed to recognise and identify meat features by processing the images acquired from sensors:

1. Chicken features on the skin surface such as Bruises, Litter Spots, Ammonia Burns and Stains were identified by processing RGB images.

2. The sub-surface chicken feature Fat Blister was detected by processing infrared images .

3. Pig leg features on the skin surface such as Haematoma (Bruises) and Burns were identified by processing RGB images.

4. The measurement of "fat thickness" and the identification of the muscles on the cut-surface of a pig leg were carried out by processing RGB images of the cut-surface.

5. The bones of pigs' legs were identified by processing x-ray images.

Although an analysis of the combination of x-ray and visible light (RGB) images showed promise for identifying bone and joint positions, and detecting fractures and dislocations, they were not promising for soft tissue defects such as abscesses. However, infrared images detected such sub-surface features on chickens, but not on pig meat where the skin was thicker and the temperature differences were smaller.

A "user definable" rule based reasoning concept was developed for assessing the quality of meat samples using the meat feature information provided by the image processing algorithms. A simple means of editing the rules was provided, thus allowing the user to adjust quality assessment criteria.

The communication mechanisms for integrating ALINSPEC with other factory computers were developed.

A computer hardware platform was specially designed to enable the processing of images at real-time production speeds. Using this platform, a single RGB sensor, and the software tools for image processing and quality assessment, a chicken ALINSPEC system ("demonstrator") was constructed at the Bull laboratories in Milan.

Although the real-time speeds were not achieved in the demonstrator, its performance could be scaled up by adding more computer processors. For example, by using four C40 digital signal processors instead of the single one used in the demonstrator, the real-time target of examining one chicken in half a second would be easily achieved.

A strategy for down-sizing the ALINSPEC demonstrator was identified. An industrial version of ALINSPEC demonstrator consisting of one RGB camera could be produced at a cost of about £22,000. However, not included in this price is the "once only" development cost estimated to be about £500,000, and the mark-up on each system.

The modular architecture developed for ALINSPEC allowed a system to be easily configured to suit the different industrial requirements. If required, additional "tools" such as infrared and x-ray sensors could be added, but the cost of including such sensors must be weighed against the benefits.

Conclusions

The development of the image processing algorithms were carried out by successfully integrating traditional image processing techniques with artificial intelligence based techniques.

Reasoning and decision taking on the basis of the information was developed and shown to be flexible and feasible at real-time production speeds. The concept used in assessing quality of the sample would allow meat industry companies to maximise their profits by enabling them to assess quality according to the specific needs of individual customers. It also provides a quick and consistent evaluation of the quality of the incoming raw material in order to provide a fair and justifiable method of paying the supplier.

The technology was commercially appropriate to the chicken industry where there were common requirements and little manual handling. However, a commercially feasible application is unlikely to use infrared or x-ray systems which, at present, are too expensive to be justified by the added benefits.

Although a pig leg demonstrator was not built, the various "tools" developed were demonstrated off-line. The results obtained from developing the pig leg tools included image processing algorithms for detecting the most important meat features required in assessing the suitability of pig legs for Spanish dry cured ham.

The techniques developed for identifying pig leg features were highly relevant to the progress of automation in the pig meat industry. However, once again, the costs of the individual techniques must be carefully weighed against the benefits. The interest in the ALINSPEC technology would come from those developing automation systems rather than direct from slaughter houses or factories. The cost of the proposed industrial version (£22,000) should help in promoting the potential of the ALINSPEC technology, especially as the original estimate at the beginning of the project was (£100,000) for a basic multiple RGB sensor system.

During the course of the project, other areas where the ALINSPEC technology could be applied were identified. Advanced systems for the automatic butchery of meat carcasses are currently under development. These systems are also computer vision based systems which analyse the shape of a carcass by using statistical methods to decide how to trim the carcasses, the methods used so far have proved unreliable. However, the approach used by ALINSPEC would be appropriate for such applications (i.e. fusing RGB and X-rays images in order to detect internal bones and principal muscle groups, and deduce the appropriate action based on the features identified). The image processing and reasoning modules could be used in the positioning and control of automatic knives for trimming.

The Expert System module classified production samples intelligently using the features reported. Although the ES can deduce trends and causes of defective samples, making it a useful module in its own right, it is difficult to imagine any system other than an intelligent computer vision system, supplying it with the required meat feature information at the required speed.

The libraries of images used in the development of ALINSPEC represent a useful resource which could be used in developing utilities such as computer based training aids.

The wide range of techniques developed would also be useful to those developing "added-value" systems in other industries where there is a need to assess quality of natural products in a similar manner.

Contents . Page

1. INTRODUCTION

This is a report by Campden & Chorleywood Food Research Association (CCFRA) on the intelligent vision system for the real-time automatic inspection of Alimentary products (ALINSPEC).

The report details the objectives, the approach used in developing the system, and the results obtained.

1.1 BACKGROUND

Campden & Chorleywood Food Research Association (CCFRA) was a partner in the ALINSPEC project which was supported by the European Community under contract BRE2-CT92-132 of the Brite-Euram programme. The UK Meat and Livestock Commission (MLC) partially supported CCFRA's work in the project.

The ALINSPEC project team (consortium) consisted of several European Companies and Organisations. The other members of the consortium were :

CEO	Consorzio CEO (Centro Eccellenza Optronica), Italy (Project Co-ordinator)
OG	Officine Galileo, Italy
COREN	Coren-Frigolouro, Spain
BULL	Bull, France
DIE	Florence University, Italy
VIGO	Vigo University, Spain
STRATH	Strathclyde University, Scotland

The project began in October 1992 and was completed in March 1996.

1.2 SCOPE

The information contained in this report was compiled from the official final technical report submitted to the EC by the project co-ordinator (CEO). This report is not a replacement for the final report, but a revised version of it.

Although CCFRA have reported the achievements of all the partners, the partners have not co-authored the report and responsibility for its contents, especially where these are opinions, rests with CCFRA.

1.3 TERMINOLOGY

CEO	CONSORZIO CEO, Centro di Eccellenza Optronica
BULL	BULL S.A.
CCFRA	CAMPDEN & CHORLEYWOOD FOOD RESEARCH ASSOCIATION
COREN	COREN / FRIGOLOURO
DIE	UNIVERSITY OF FLORENCE, Dipartimento di Ingegneria Elettronica
OG	OFFICINE GALILEO S.P.A.

STRATH	UNIVERSITY OF STRATHCLYDE, Electronic and Electric Engineering Department
UVIGO	UNIVERSITY OF VIGO, ETSIT Group
C40	TMS320C40 DSP computer processor
ES	Expert System Module
IVS	Intelligent Vision System Module
MMI	Man Machine Interface Module
PPP	Parallel Processing Platform.
GLYPH	A Khoros icon representing a complete image processing routine
VIFF	The standard data file format used by Khoros for storing images
COOKED IMAGE	Synthetic image to be used for image processing purposes, e.g. multi-dimensional kernels used in morphological operations

2. OBJECTIVES

Within the food industry in general, and the meat industry specifically, the importance and difficulty of inspection is increasing. Public awareness of and interest in the quality of food is also increasing. The market is becoming more discriminating, reflected in increasing pressure by retailers for producers to supply products of consistent and assured high quality. Such pressure increases the need for producers to examine their products consistently and effectively.

As quality becomes more important and profitability is under pressure, there is a corresponding trend in the need for accurate classification of carcasses and meat cuts in order to maximise returns. It is also important that the classification criteria used must respond to changes in requirements and market conditions.

The level of automation in the meat industry is rapidly changing. Line speeds of 6,000/hour are common for poultry, with speeds up to 20,000/hours likely in the near future. Speeds for larger animals, typically 600/hour for pigs, are lower, but still high enough to make human visual inspection difficult and ineffective.

The introduction of automated butchery methods increases the importance of effective examinations, as well as introducing additional requirements. Automated butchery needs accurate images of external conformation and internal structure. Failure to detect internal defects, especially abscesses, can result in extended interruption of automated lines.

The need for a high speed, consistent, verifiable examination of meat, including the detection of internal defects, makes human examination increasingly unsatisfactory and difficult to manage.

Inspection systems in the meat industry are predominantly human visual systems. Butchers and line operators look at the meat and make subjective judgements. The failings of this procedures include ineffectiveness, especially at high line speeds, and inconsistency. Sometimes there is control by managers on a Quality Control sampling basis, but this results in only partial Quality Control with associated inspectors "going native", and problems in paying by results.

The Senior Management of COREN/FRIGOLOURO identified improved quality management as a major policy objective in order to maintain or increase their market share. They wished to improve the consistency and homogeneity of the quality of their various product types, and to improve their ability to tailor their production to individual customer requirements. Their commitment to a programme of improved quality management was evidenced by their introduction of the ISO9000, their installation of new lines, and their commitment to ALINSPEC. Their expectation was that automated inspection systems would allow them to accurately assess the quality parameters of their raw materials (chickens and un-cured pig legs) resulting in:

- more consistent and appropriate allocation of pieces of meat to a product type
- the ability to modify quality parameters associated with product types in response to customer demand
- the supply of more complete and objective quality information to customers
- more accurate and demonstrably fair payments to suppliers.

Therefore, the objectives of the project were to develop techniques (architecture, image and data processing software, and expert system) to:

- look at the meat and perform 100% on-line examination using multiple existing sensor technologies (visible, X-ray, Infrared)
- recognise features (fractures, bruises, shape, ...)
- deduce significance, taking into account other information (weight, plunge probe, production schedule, ...)
- deduce and initiate appropriate action (classify meat, issue warnings, ...)

and to construct industrial demonstrators working on pig meat and chickens.

The demonstrators were to operate on pig legs, deciding their suitability for Spanish dry cured ham, and on chickens assigning a quality grade. The demonstrators were for development, proving, and demonstration purposes only, and were not to take the form of prototypes. As such they were targeted towards the specific needs of Coren-Frigolouro. However, they were to recognise and reason on the basis of features generally relevant to the assessment of pig meat and poultry, demonstrating the long term and broad applicability of the tools and techniques developed.

Another objective was to find ways in which cheap industrial versions of the demonstrator could be used to deliver the performance required by the meat industry.

3. REQUIREMENTS

Although the functionality of the ALINSPEC system was primarily influenced by the requirements of COREN/FRIGOLOURO, the requirements of the meat industry as a whole were identified.

3.1 Meat Industry

A market survey of automated systems in the meat industry was carried out by CCFRA to determine the price the industry was prepared to pay for a commercial ALINSPEC system. The survey gave an indication of the meat industry's willingness to pay considerable sums of money (of the order of £350,000) for high technology systems when these contributed directly to earnings, either by allowing compliance with legislation, or by direct reduction of waste or "give away". However, such systems should demonstrate a return on the investment required.

In addition, OG BULL and CCFRA made enquiries at companies and organisations in the meat industry in order to identify their needs, and their level of interest in ALINSPEC. This resulted in part sponsorship of CCFRA by the MEAT AND LIVESTOCK COMMISSION (MLC), UK.

The result of the enquiries were summarised as follows.

1. Modern chicken lines were similar to each other. There was considerable interest in a system to provide on-line, objective measurement of quality. The techniques developed within the ALINSPEC project for intelligent, adaptive recognition of defects, and intelligent reasoning, would go a long way to making up the deficiencies in previous systems. The most important determinants of quality were on the surface, and were detectable with visible light, and the breast area was far more important than other views.

2. Pig lines were largely manual, with a high degree of skilled manual intervention. There was very little interest in visual quality assessment systems in pig factories. Where an interest was shown, the requirements were inconsistent. However, there was a strong demand for increased automation of the pig butchery process. The automation efforts were increasingly moving into areas where artificial "vision" was needed.

3.2 Coren-Frigolouro

The specific requirements defined by Coren-Frigolouro showed a clear need for a system such as ALINSPEC for:

1. ***on-line quality control***, in order to provide their customers with products of a consistent quality.
2. ***incoming quality control tests***, in order to provide a quick and consistent evaluation of the quality of the incoming raw material, thus providing a fair and justifiable method of paying the supplier.

3.3 Conclusion

COREN/FRIGOLOURO's requirements were somewhat different from that of the northern European meat industry. The raw materials e.g. live chickens and pigs, arriving at the factories in northern Europe were of such high and consistent quality, that the cost of an on-line quality assessment system would be extremely difficult to justify. Notwithstanding the differences, the poultry industry showed an interest in ALINSPEC right from the beginning of the project. However, as the precise functionality of a chicken prototype demonstrator emerged, the real interest was not in a "marketable" ALINSPEC system for on-line quality assessment, but in how the ALINSPEC technology could be used in "added value" systems e.g. in an automated system for the identification and trimming of fat off chicken portions.

In the pig meat industry, there was a strong demand for increased automation on pig slaughter lines. The automation effort seemed most effective when concentrated on particular unit processes, rather than attempt to produce a wholly automatic line. Automation efforts had been most effective when using existing, proven, sensors technology. All this militated against a "marketable" prototype incorporating ALINSPEC technology. The interest in ALINSPEC would come from those developing automation systems rather than direct from the factory.

Therefore, the ALINSPEC system was required to:

- acquire multiple images of the meat from visible and non visible image sensors
- fuse and analyse those images to recognise relevant features and defects
- deduce the significance of those feature in the context of other available information, including marketing considerations
- interact with other systems in the factory to obtain and pass on relevant information
- initiate appropriate actions for an effective consistent 100% real-time examination of subjectively assessed food products such as chickens and Spanish dried cured ham, according to the requirements of COREN/FRIGOLOURO.

The meat features to be recognised by ALINSPEC from images of the carcasses were:

Chickens	Pig legs
Bruise	Leg Weight
Ruptured Vein or Artery	Skin Colour
Torn Skin	Hematoma
Fracture	Petechiae
Dislocation	Luxation
Breast Blister	Fracture
Skin Scald	Marks on Skin
Ammonia Burn	Fat Colour
Skin Colour (yellow/white)	Wrinkled Skin
Weight (size)	Leg Conformation
Lividity	Trotter defects
Abnormal Skin Colour	Stamp
Litter spot	Exposure of bone
Toad Skin	Scars and bites
Scratches	Torn muscle
Conformation (including deformity and thinness)	Burned Skin

For chickens, the requirement was to allocate a quality grade to each chicken inspected (in decreasing order of value):

- Grade A - highest quality
- Grade B - could be cut and sold as portions

- Grade C - condemned chicken

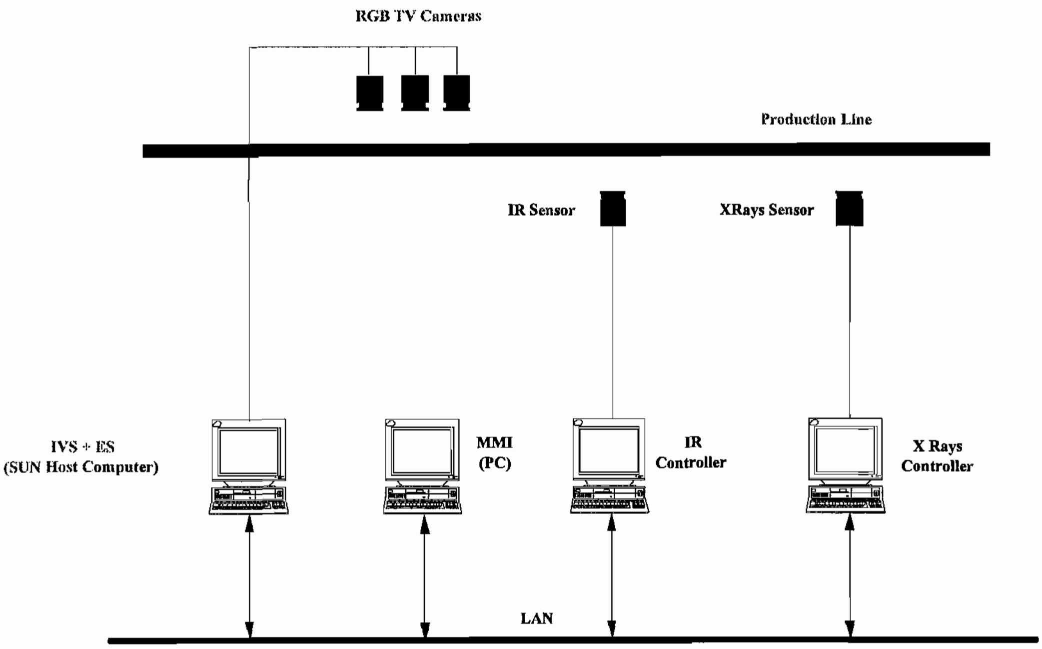
For pig legs, the requirement was to suggest to the trimmer what each leg should be used for (in decreasing order of value):

- Serrano - best quality dry cured ham
- Rounded with Trotter - also dried and cured
- Rounded without Trotter
- York - skinned and sold as fresh meat suitable for boiled ham
- Fresh meat - skinned, de-boned and sold as fresh meat
- Condemn, do not trim

Equally as important was the requirement that the quality assessment criteria be modifiable, allowing COREN/FRIGOLOURO to vary the quality of their products according to the needs of their customers. Although, these precise functions were not required by the Northern European poultry industry, the techniques developed would be beneficial to those developing "added value" systems. The technology could also be applied to other industries where there was a need to inspect and assess the quality of natural products in a similar manner.

4. APPROACH

The approach was to develop a "tool box" from which appropriate methods could be selected according to the performance and cost requirements of particular applications. This led to a modular and distributed architecture shown by the figure below.



The imaging sensors of the ALINSPEC system would be situated along a piece of production line (test bed). As the meat samples were conveyed along the line, they would pass in front of a series of imaging sensors. An on/off trigger signal from an optical sensor would start the image acquisition process. The digitised images would be processed by the IVS and the features identified communicated to the ES. The ES would process this information and the resulting grades would be sent to the MMI. The IVS, ES and MMI modules would communicate over a Local Area Network (LAN).

In addition to sending the results to the MMI, a signal could also be sent to a mechanical switch which would then separate the samples according to the grade allocated by the system.

If both visible and invisible(infrared/x-ray) sensors were used, the non-visible images of the samples would be acquired and processed by "infrared controller" and "x-ray controller" modules. Since RGB and infrared/x-ray sensors cannot acquire their images simultaneously, the infrared and x-ray controller modules would process the infrared/x-ray images, translate the features identified into appropriate symbols and send these symbols to the IVS. The IVS would fuse the symbolic information with the information already obtained from processing RGB images and send to the ES. However, fusing of information acquired at different times would require a means of identifying and tracing each sample. This would be necessary so as to allow information for each sample to be correlated even if acquired at different points on the production line. Appropriate bar-code readers and transponders for the identification and tracing of samples were identified by COREN/FRIGOLOURO and BULL.

The development of the modules that make up the ALINSPEC "tool box" were carried out as follows:

- Test beds were constructed at COREN/FRIGOLOURO factories so as to present the samples (chickens and pig legs) to appropriately positioned imaging sensors under the appropriate illumination.

- An evaluation of existing commercial RGB TV cameras was carried out by OG and BULL in order to determine the most appropriate RGB sensor.

- A detailed evaluation of the Infrared sensors available commercially was carried out by OG taking into account the cost of the equipment, the image resolution and integration time required.

- A detailed evaluation of the X-ray sensors available commercially was also carried out by OG (in collaboration with IRVIN ELECTTRONICA) taking into account the cost of the equipment, the optimal layout, safety and accident prevention factors.

- Following the evaluation of RGB cameras and the purchase of an appropriate camera, two image libraries were created by UVIGO using the test bed. The libraries consisted of images of defective and non-defective samples found at the COREN/FRIGOLOURO chicken and pig factories.

- Infrared images of chickens and x-ray images of pig legs were acquired by OG.

- Several image processing software algorithms were developed by DIE, UVIGO, OG, BULL and STRATH using RGB, infrared and x-ray images. The algorithms were for:
  - identifying visible chicken features - DIE
  - identifying visible pig leg features - UVIGO
  - identifying sub-surface chicken meat features - OG
  - identifying bones in pig legs - OG
  - identifying muscle groups and measuring the fat thickness on the cut-surface of a pig leg - BULL

- An Expert System module was developed by CCFRA to grade a meat sample using the meat feature information reported by the IVS, and a set of simple user definable rules called a Quality Profile. The ES deduced significant changes in quality of meat samples examined, and maintained a log of all the information generated for each sample (i.e. the features reported and the grades assigned to each sample). The ES also produced detailed and summary reports of the logged information.

- The Man Machine Interface (MMI) was developed by OG as a software module running on a DOS/Windows™ platform. The MMI was developed using a windows application development tool called **LabWindows/CVI™** from NATIONAL INSTRUMENTS.

- A Parallel Processing Platform was developed by BULL to carry out the image acquisition and processing functions of the IVS at real-time production speeds. All the sub-modules of the Parallel Processing Platform were developed according to the TIM-40 specification defined by TEXAS INSTRUMENTS. This allowed hardware modules produced by different manufacturers to be plugged into standard computer motherboards.

- Communication between the different modules of ALINSPEC were carried out using fixed length records stored in NFS mounted files (Network File System on a LAN/Ethernet TCP/IP network). Two files were used for communicating each type of message. One file was used for storing the records, and an associated file (index) was used for storing the number of the last record written.

5. METHODS AND RESULTS

5.1 TEST BED

Test beds were constructed at COREN/FRIGOLOURO factories in order to present the samples (chickens and pig legs) to appropriately positioned imaging sensors under the appropriate illumination.

Appropriate bar-code readers and transponders for the identification and tracing of samples were identified by COREN/FRIGOLOURO and BULL.

The images used in developing the image processing algorithms were acquired from the test beds using RGB sensors.

5.2 RGB Sensors

Existing commercial RGB TV cameras were evaluated by OG and BULL taking into account the following requirements:

Field of view	=	300 mm (horizontal) by 400 mm (vertical)
Resolution (min. defect size)	=	1.0 mm by 1.0 mm
Maximum working distance	=	600 - 700 mm
Electronic shutter	=	Yes
Video out	=	RGB and video composite output signals

The following cameras were evaluated:

- Panasonic WV-CL700
- Panasonic WV-CL350
- JVC TK-1070E
- SONY DXC107P

The JVC TK-1070E camera was selected as the most appropriate for the ALINSPEC system.

5.3 Infrared Sensors

A detailed evaluation of the Infrared sensors available commercially was carried out by OG taking into account the cost of the equipment, the image resolution and integration time required.

Infrared systems with the required performance characteristics were expensive (in excess of US $10,000). They also needed expensive and/or inconvenient cooling systems. OG produced a detailed design of a camera/image capture system using line scan imaging techniques which rely on the movement of the object in order to generate a picture. The linear array sensor initially chosen (the M2105-256-1 from Litton) was withdrawn from the market. In its place, another sensor (the Graseby Infrared Sensor from the UK) with similar characteristics and performance was selected.

The resulting low cost, high performance system was estimated at US $86,550 for the first prototype, and US $17,300 each for a series of 50. This type of sensor was used in the development of image processing algorithms for detecting the sub-surface chicken meat feature “Fat Blister”.

The evaluation also resulted in the following:

- a suitable optical design
- the design of an industrial waterproof container to house the sensor, optics and the electronic interfaces
- the focusing mechanism
- the electronic interfaces
- the computing and image processing architecture.

5.4 X-RAY SENSORS

A detailed evaluation of the x-ray sensors available commercially was also carried out by OG (in collaboration with IRVIN ELECTTRONICA) taking into account the cost of the equipment, the optimal layout and safety factors.

Two types of sensors were investigated:

- An image intensifier based solution, where a whole image frame was intensified and captured by a TV camera. The image intensifier solution resulted in images where the noise increased with shorter integration (imaging) times. An integration time of more than 0.25 seconds was needed for a satisfactory noise level. However, high integration times resulted in blurred images because of the movement of the sample. Therefore, the image intensifier technique would require a means of holding the sample stationary when imaging.

- A linear diode array (line scan), where a line of detectors captured one line of an image, and the movement of the sample was used to generate a picture.

The cost for both types of sensors were similar, in the order of US $70,000 for an industrial version.

The image intensifier approach gave better images, but the samples would have to be stopped before imaging. The line scan approach could image moving objects, but produced an image of lesser quality. The line scan approach was selected and a demonstrator system constructed by modifying an existing x-ray luggage inspection system.

The pig leg image from a line scan sensor varied in quality. The top and bottom part of the image were of better quality than the middle part. This was due to the variation in the thickness of the leg. The automatic gain control found on most x-ray equipment could not be appropriately tuned to cater for the variation in the thickness of the leg, so improvements in image quality were obtained by manually adjusting the intensity. This problem was reflected in the image processing algorithms developed for identify the bones. The bones were reliably identified in the top and bottom parts of the leg, but not in the middle part of the leg.

Further development of x-ray equipment would result in a clearer image, which would also improve the accuracy of the algorithms for identifying bones, especially in the middle part of the leg.

5.5 RGB IMAGE LIBRARIES

Following the evaluation of RGB cameras, and the purchase of the appropriate camera, two image libraries were created by UVIGO using the test beds. The libraries consisted of images of defective and non-defective samples found at the COREN/FRIGOLOURO factories. The image libraries were used in developing image processing algorithms and the Expert System module.

5.5.1 CHICKEN IMAGE DATABASE

250 chicken samples were imaged from 3 different views resulting in images of 25 different chicken meat features. The location of features imaged and comments were recorded in the "header" of each image.

Meat Feature	Comments
CF1. Bruise:	Included old and recent bruises, old bruises are marked with the code '1' in the image header.
CF2. Rupture of vein or artery:	Feature was associated with "badly stunned chickens".
CF3. Torn skin:	Skin torn by factory machines were marked with the code '1' in the image header
CF4. Fracture:	Included fractures with or without bruises. Associated bruises were marked as separate features.
CF5. Dislocation:	Included dislocations with or without bruises. Associated bruises were marked as separate features.
CF8. Scratch:	
CF9. Blister:	Included blisters with or without scabs. Associated scabs were marked as separate features.
CF10. Feathers:	
CF11. Skin scald:	Only the ones considered as important by the factory were marked.
CF13. Ammonia burn:	Included burns with or without scabs. Associated scabs were marked as separate features.
CF15. Skin colour:	Marked in the header as 'W' (White) or 'Y' (Yellow).
CF22. Stub:	Only the ones considered as important by the factory were marked*.
CF25. Blue-red livid skin areas:	Associated with incomplete bleeding after slaughter.
CF27: Ragged skin edge	Associated with a badly cut neck.
CF28: Abnormal colour	Abnormal colour maybe due to disease.
CF29: Bulging craw	
CF30. Fat blister:	
CF31. Toad skin:	Included all areas of curly skin.
CF34. Scab:	Usually associated with blisters or ammonia burns. Scabs on the hock and lower legs were marked in the special set of images*.
CF35. Skin rubbed off:	Only the ones considered as important by the factory were marked.
CF36. Bites:	
CF37. Dirty nails:	Only marked in a special set of images*.
CF40. Rickets	
CF41. Deformation**	
CF42. Thinness**	

* 20 samples were imaged from different views in order to capture the features present on the neck and feet.

** These features were reported by the image processing algorithms as CF38 (conformation).

The proportion of features imaged was influenced by how often these features occurred, and how important the features were to the overall quality of the sample. The following table shows the distribution according to the three views imaged.

Feature Code	Number of samples	Number from View1	Number from View2	Number from View3	% of Total Number of samples
CF01	138	61	40	37	18.3
CF02	26	12	8	6	3.4
CF03	38	4	20	14	5.0
CF04	43	19	10	14	5.7
CF05	35	17	10	8	4.6
CF08	33	0	13	20	4.4
CF09	10	7	2	1	1.3
CF10	52	17	16	19	6.9
CF11	35	12	12	11	4.6
CF13	27	25	1	1	3.6
CF22	23	3	10	10	3.0
CF25	38	13	13	12	5.0
CF27	22	8	8	6	2.9
CF28	30	10	10	10	4.0
CF29	25	11	4	10	3.3
CF30	15	0	7	8	2.0
CF31	22	8	7	7	2.9
CF34	28	11	9	8	3.7
CF35	29	11	9	11	3.8
CF36	20	0	10	10	2.6
CF37	11	4	4	3	1.5
CF40	30	10	10	10	4.0
CF41	3	1	1	1	0.4
CF42	16	6	6	6	2.4

5.5.2 PIG LEG IMAGE DATABASE

102 pig leg samples were imaged from 5 different views. Additional samples were imaged from just 2 views, 10 images of samples before trimming, and 10 after trimming. These latter images were used in developing image processing algorithms for the measurement of fat thickness on the cut-surface of a leg. All together, images of 17 different pig meat features were acquired. The location of features imaged and comments were recorded in the "header" of each image.

Meat Feature	Comments
PF11. Skin colour:	Only the ones considered as important by the factory were marked.
PF13. Bruise:	Bruises associated with fractures were marked with the code '1' in the image header, and '0' when associated with dislocations.
PF14. Petechiae:	
PF15. Luxation:	
PF16. Fracture:	
PF17. Hair:	
PF25. Dirt marks on skin:	
PF28. Leg-fat thickness	Only marked in the latter images.
PF34. Meat colour:	There were a couple of laboratory-checked examples of PSE and DFD meat. Marked as '0' or empty.- PSE Marked as '1'.-DFD
PF35. Lividity (incomplete bleeding):	

Meat Feature	Comments
PF36. Leg conformation:	
PF37. Trotter defect:	
PF40. Bad Stamp:	
PF42. Leg bone exposed:	
PF50. Swelling:	Included 4 different kinds depending on the origin: Marked as '0'- callus in the bone, Marked as '1' - abscess of pus, Marked as '2' - subcutaneous swelling, Marked as '3' - accumulation of serum liquid due to fracture or luxation.
PF51. Muscular tear:	
PF53. Burned skin from the scalder:	

The proportion of features imaged was influenced by the availability of defective samples. However, unlike chickens the most important pig features occurred less frequently in the factory. The table below shows the distribution according to four out of the five views imaged.

Feature Code	Number of samples	Number from View1	Number from View2	Number from View3	Number from View4	% of Total Number of samples
PF11	6	0	2	2	0	2.0
PF13	95	12	18	29	17	30.3
PF14	9	4	2	0	0	2.9
PF15	2	0	1	1	0	0.6
PF16	60	2	2	29	25	19.2
PF17	5	5	0	0	0	1.6
PF25	4	1	0	1	1	1.3
PF28	10	0	0	0	10	3.2
PF34	10	0	0	5	5	3.2
PF35	10	0	0	5	0	3.2
PF36	26	4	6	6	5	8.3
PF37	6	0	3	0	0	2.0
PF40	2	1	1	0	0	0.6
PF42	10	1	3	1	1	3.2
PF50	44	2	19	2	1	14.0
PF51	3	0	0	1	1	0.9
PF53	1	3	4	2	0	3.5

5.6 INTELLIGENT VISION SYSTEM

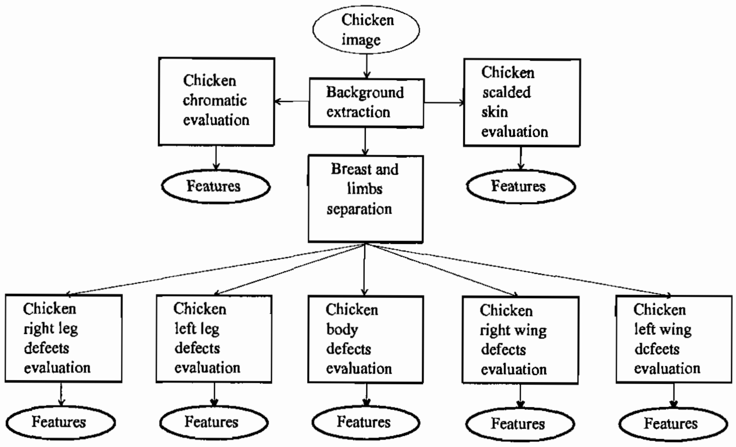
The IVS was developed in a modular fashion. The IVS consisted of a communications module and one or more image processing modules. The communications sub-module was developed by UVIGO to handle the communication between the IVS and the MMI and ES. The image processing modules were developed by DIE, UVIGO, BULL and OG using the Khoros image processing development tool.

For chickens, modules were developed for processing RGB and infrared images. For pig legs, modules were developed for processing RGB images and x-ray images. The module for processing chicken RGB images was translated into a C program and compiled to run on a purpose built Parallel Processing Platform.

5.6.1 CHICKEN RGB IMAGE PROCESSING MODULE

The chicken RGB image processing module (as shown by the figure below) consisted of a series of image processing algorithms for:-

- extracting the chicken silhouette
- segmenting it into its sub-parts (body, legs and wings)
- identifying skin areas containing a meat feature
- classifying the areas identified as a particular meat feature



The algorithms were written as Khoros glyphs:

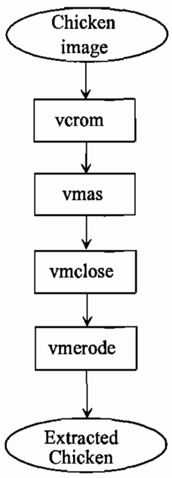
- VCROM - chromatic transformation.
- VMAS - masked threshold.
- VRESKER - chicken body kernel re-size.
- VCHKSS - chicken dissection.
- VARB - chicken features assembling and correlating.
- VLREGOUT - image chromatic registration.
- VDIST - Mahalanobis distances calculation.
- VCOLORTH - image colour threshold.
- VBGROW - blue defects finder.
- VMOD - modal filter.
- VMOD1R - rubbed skin filter.
- VGCLAX - chicken chromatic classification.
- VSCALD - scalded skin classification.
- VCLAS - defect classifier for body.
- VLEGCLAS - defect classifier for legs.
- VWINCLAS - defect classifier for wings.

The glyphs above made use of the following standard Khoros glyphs:

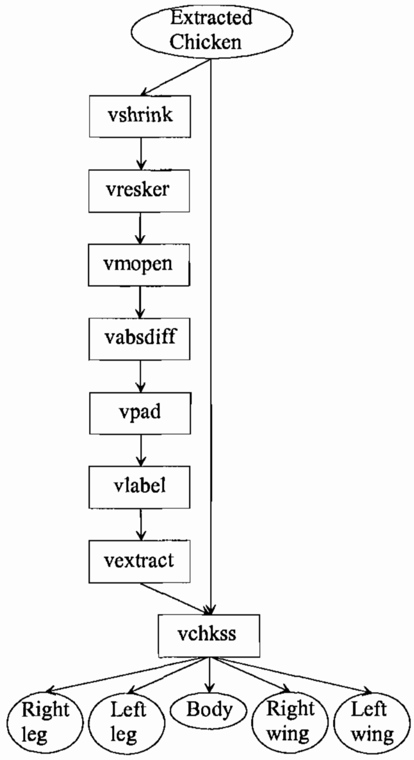
- VMERODE - morphological erosion.
- VMOPEN - morphological opening.
- VMCLOSE - morphological closing.
- VMDILATE - morphological dilation.
- VM180 - kernel rotation 180 degrees.
- VLABEL - image labelling.
- VXOR - logical XOR.
- VAND - logical AND.

- VOR - logical OR.
- VCONVERT - conversion between different data types (not ported to parallel processing platform).
- VTHRESH - image threshold.
- VHMED - histogram mediation.
- VSTATS - statistical computation (not ported to parallel processing platform).
- VSHRINK image reduction.
- VABSDIFF - image absolute difference.
- VPAD - image padding.
- VEXTRACT - image extraction.
- VARVIFF - image pixel to variable extraction (not ported to parallel processing platform).

The background extraction and separation of the body was implemented as follows:

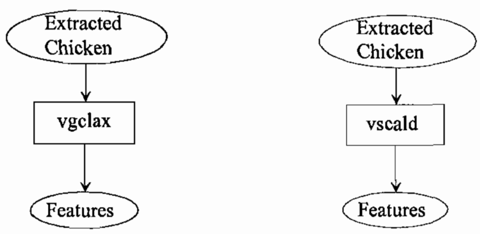


Background Extraction Section

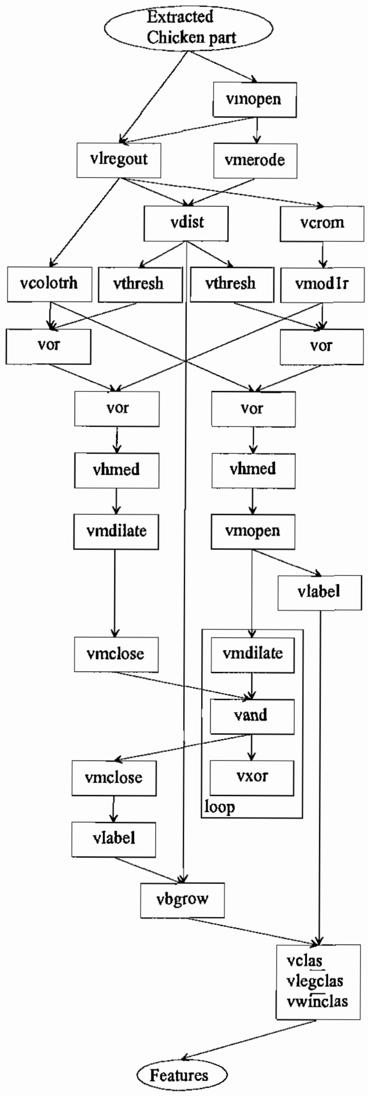


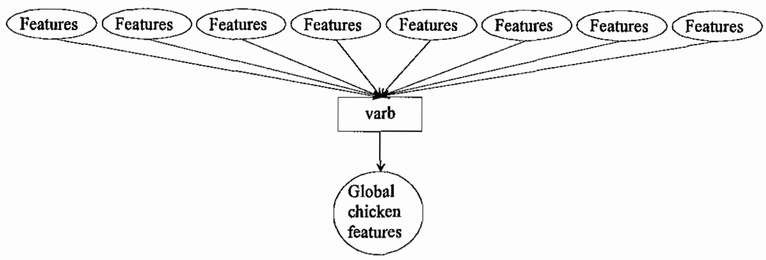
Separation of Limbs Section

The "chromatic" evaluation of each chicken in order to extract the meat feature "Skin scald" and all others was implemented as follows:



The classification of features detected, and the collection and correlation of all features detected was implemented as follows:





The Khoros glyphs (or executables) were executed from a UNIX shell script. Data was passed from one glyph to another using temporary files.

5.6.2 EVALUATION OF CHICKEN IMAGE PROCESSING ALGORITHMS

An evaluation of the image processing algorithms was carried out by CEO in order to optimise the efficiency of the algorithms. The algorithms produced by DIE for processing chicken RGB images were evaluated before porting to the parallel processing platform.

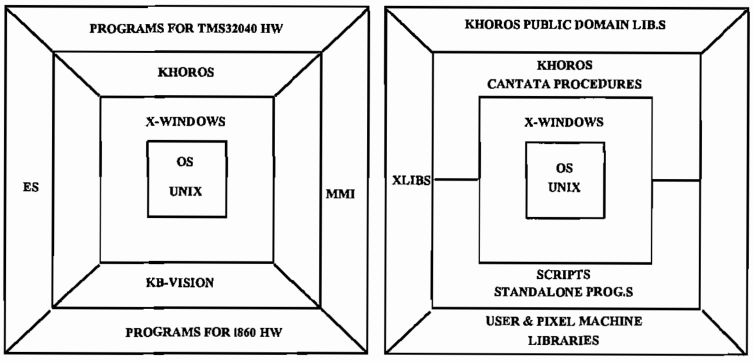
The huge "number crunching" involved in evaluating the algorithms led to the selection of the following hardware configuration:

- Workstation Solbourne 5/602 with 2 SPARC CPU's, 32 Mbytes of RAM, 1.1 Gbytes Hard Disk, Colour Monitor 19", 150 Mbytes backup unit, running the Solbourne OS/MP 4.1A UNIX operation system (compatible with SunOS 4.1.1).

- AT&T Pixel Machine Model 964D with 1280x1024 high resolution 21" 60 Hz non interlaced display monitor. The Pixel Machine consisted of a parallel array of 64 AT&T DSP32 Digital Signal Processors (20 MHz 5 MIPS, 10 MFLOPS). Each DSP (capable of 32 bit floating point arithmetic) consisted of 4 Kbytes "on-chip" memory, four 40 bit floating point registers, twenty-one 16 bit integer registers and, parallel and serial I/O ports with DMA. The overall processing power of the Pixel Machine was about 820 MFLOPS.

The software architecture of the evaluation environment (shown below) consisted of the following:

- a standard UNIX and X-11 Windows kernel
- Khoros image processing development tool
- Windows, Khoros and Pixel Machine software libraries
- non standard software libraries (written by CEO and OG) to act as an interface between Khoros and the Pixel Machine.



ALINSPEC SYSTEM ENVIRONMENT　　　EVALUATION ENVIRONMENT

Prior to evaluating the algorithms, routines were written for:

- the transfer of images from Khoros to the Pixel Machine
- calling the Pixel Machine libraries routines from the Khoros environment
- the transfer of image processing results from the Pixel Machine back to Khoros (numeric values or processed images produced by the Pixel Machine)

The evaluation procedure (see next figure) consisted of the following:

1. images from the image library were selected according to a pre-defined criteria (sorting rule, e.g. all the images of chickens, or all the images of chickens with a bruise on the breast, or all the images of chicken with a broken wing)

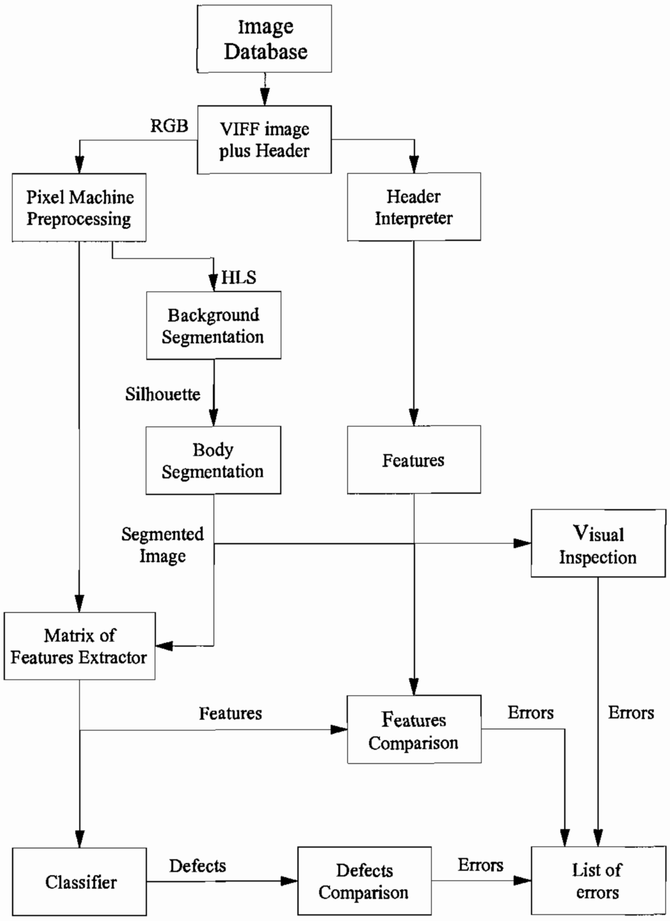
2. for each image selected, the meat features marked during image acquisition and recorded in the header were extracted

3. the images were transferred to the image processing module (algorithms) being evaluated

4. the results obtained from processing the images were compared with the list extracted from the headers

5. the results were stored in an ASCII file

6. the above steps were repeated for another set of images selected according to the next criteria



The Evaluation Loop

The evaluation was carried out by separating the chicken meat features into 3 main classes: global "chromatic" features, local "chromatic" features and shape features. The algorithms for detecting each class were fed a set of images of white chickens from the image library. The process was then repeated for yellow chickens. The algorithms were developed by using the image library "header" information as reference data. An image header contained meat features marked during the image acquisition phase. The image library contained 230 images, 147 were images of white chickens.

The meat features detected by evaluating the chicken RGB image processing algorithms (mapped class) were compared with the information contained in the image headers (true class). The results were presented as tables.

Although, algorithms were developed to detect three kinds of global "chromatic" meat features: Scalded skin (CF11), Incomplete bleeding (CF25) and Abnormal colour (CF28), an additional meat feature (CF41) was introduced because of the difficulty in distinguishing between CF25 and CF28. The classification of CF25 and CF28 as one kind of feature was acceptable by the human experts as both features are global and the presence of either results in a very low grade, i.e. grade C "Condemn".

Although, there were cases where global features such as CF41 were detected as a local feature Bruise (CF1), the effect on the quality of the sample was found to be the same. This also applied to the global feature Scalded skin (CF11) which was sometimes detected as the local features Rubbed skin (CF35) or Blister (CF9-CF13). Some large sized local features had the same effect on quality as a global feature.

The algorithms for detecting global and local "chromatic" meat features were evaluated by processing:

a) images of white chicken breasts:

Classification of white chicken breasts as healthy or defective.

	Mapped class		
True class	**Healthy**	**Defective**	**Number of test images**
Healthy	83	5	88
Defective	5	60	65
$P_e = 6.5\%$ P(false alarm) = 5.7% P(missing a feature) = 7.7%			

Detection and classification of meat features on white chicken breasts.

	Mapped Class							
True class	**Healthy**	**CF42**	**CF1**	**CF9**	**CF11**	**CF34**	**CF25**	**Number of samples**
Healthy	80	1	1	2	1	0	1	86
Stain (CF42)	0	2	1	0	0	0	0	3
Bruise (CF1)	1	0	8	1	0	0	1	11
Blister (CF9)	2	0	0	19	0	0	0	21
Scald skin (CF11)	1	0	0	0	4	0	0	5
Litter spot (CF34)	0	1	0	2	0	6	0	9
Global features (CF25/CF28/CF41)	1	0	2	0	0	0	15	18
$P_e = 12.4\%$								

The P_e (probability of an error) from classifying the features detected was higher (second table) than classifying the breast as healthy/defective (first table). The most frequent causes of such errors were the probability of missing a breast blister ($P_e = 9.5\%$), not classifying a bruise ($P_e = 27\%$), not classifying a global feature such as "abnormal colour" ($P_e = 16\%$), and the classification of a litter spot as a blister ($P_e = 22\%$). For some features such as litter spot, there were only 9 examples in the image library.

b) images of white chicken legs:

Classification of white chicken legs as healthy or defective.

	Mapped class		
True class	**Healthy**	**Defective**	**Number of test images**
Healthy	190	5	195
Defective	2	16	18
$P_e = 3.3\%$ P(false alarm) = 2.6% P(missing a feature) = 11.1%			

Detection and classification of meat features on white chicken legs.

	Mapped Class					
True class	**Healthy**	**CF1**	**CF34**	**CF43**	**CF35**	**Number of samples**
Healthy	190	0	0	3	2	195
Bruise (CF1)	1	9	0	2	0	12
Litter spot (CF34)	0	0	1	1	0	2
Reddish area (CF43)	1	0	0	3	0	4
Rubbed skin (CF35)	0	0	0	0	0	0
$P_e = 4.7\%$						

An additional meat feature CF43 was introduced during the development of the algorithms in order to report the reddish skin areas associated with the skin texture feature “Toad skin”. Although the P_e values were low, as mentioned ealier there were very few examples of this feature in the image library.

c) images of white chicken wings:

Classification of white chicken wings as healthy or defective.

	Mapped class		
True class	**Healthy**	**Defective**	**Number of test images**
Healthy	144	34	178
Defective	12	26	38
$P_e = 21.3\%$ P(false alarm) = 19.1% P(missing a feature) = 31.6%			

Unlike the algorithms developed for detecting features on the breast and legs, the image “headers” could not be used as reference data during the development of the algorithms for wings. The presence of shadows in the images of wings made it extremely difficult to distinguish relevant meat features from normal skin. An examination of the images on a computer screen was carried out by human experts to identify the features. This proved to be difficult as well. However, they were able to classify the wings as defective or normal, rather than identify individual features. Thus, the algorithms were developed using the human experts' classification and not the information recorded in the image header.

d) images of yellow chicken breasts:

Classification of yellow chicken breasts as healthy or defective
(without module for detecting the meat feature rubbed skin).

	Mapped class		
True class	**Healthy**	**Defective**	**Number of test images**
Healthy	50	2	52
Defective	10	21	31
$P_e = 14.5\%$ P(false alarm) = 3.8% P(missing a feature) = 32.2%			

Detection and classification of meat features on yellow chicken breasts
(with module for detecting the meat feature rubbed skin).

	Mapped Class							
True class	**Healthy**	**CF42**	**CF1**	**CF9**	**CF11**	**CF35**	**CF25**	**Number of samples**
Healthy	40	5	2	3	0	0	0	50
Stain (CF42)	0	4	0	0	0	0	0	4
Bruise (CF1)	1	0	0	3	0	0	0	4
Blister (CF9)	1	0	0	4	0	0	0	5
Scald skin (CF11)	1	0	0	3	1	0	0	5
Rubbed skin (CF35)	0	2	0	6	0	0	0	8
Global feature (CF25/CF28/CF41)	1	0	0	2	0	0	4	7
$P_e = 36\%$								

Although the introduction of the module for detecting the meat feature rubbed skin reduced the probability of missing a feature, the P_e value for yellow chicken breasts was greater than that of white chickens. This was due to the smaller number of yellow chickens, and the variety in the type of yellow chickens samples imaged during the image acquisition phase.

e) images of yellow chicken legs:

Classification of yellow chicken legs as healthy or defective
(based only on the algorithm for detecting the meat feature rubbed skin).

	Mapped class		
True class	**Healthy**	**Defective**	**Number of test images**
Healthy	132	13	145
Defective	6	15	21
$P_e = 11.4\%$ P(false alarm) = 8.9% P(missing a feature) = 28.6%			

f) images of yellow chicken wings:

Classification of yellow chicken wings as healthy or defective.

	Mapped class		
True class	**Healthy**	**Defective**	**Number of test images**
Healthy	140	5	145
Defective	6	15	21
$P_e = 6.6\%$ P(false alarm) = 3.4% P(missing a defect) = 28.6%			

The algorithms for detecting "shape" meat features e.g. size/weight (CF23), conformation (CF38), fractures (CF4) and dislocations (CF5) were evaluated by processing all the images in the library (white and yellow chickens).

Detection of fractures and dislocations on the legs.

	Mapped class		
True class	**Healthy**	**Defective**	**Number of samples**
Healthy	447	4	451
Defective	1	8	9
$P_e = 1.08\%$ P(false alarm) = 0.89% P(missing a defect) = 11.11%			

Detection of fractures and dislocations on the wings.

	Mapped class		
True class	**Healthy**	**Defective**	**Number of samples**
Healthy	433	8	441
Defective	2	17	19
$P_e = 2.17\%$ P(false alarm) = 1.8% P(missing a defect) = 10.52%			

5.6.3 CHICKEN INFRARED IMAGE PROCESSING MODULE

The use of Infrared sensors for the on-line automatic inspection of chicken carcasses was investigated by OG. Infrared sensors were used in developing algorithms for detecting subcutaneous features not visible in RGB images, such as Fat Blisters (CF30).

Prior to developing the algorithms, infrared images were acquired at OG laboratories. The sample target temperature was set at 50° C to represent the chicken carcass emerging from the hot water used in the plucking process on a chicken production line. The requirements were to:

- detect temperature variations of 0.1-0.2 ° C in order to detect subcutaneous masses cooling at different rates because of the different specific heat capacities.
- detect emissivity differences of about 1% in order to detect skin damage, where the emissivity of exposed subcutaneous tissue was different from that of skin.

An infrared sensor of a spectral band of 8-12 micron was chosen instead of the 3-5 micron sensor used in detecting high temperature targets.

Chicken carcasses with subcutaneous defects were immersed in a 50 ° C water bath for 3 minutes. After a short period, the samples were imaged using the infrared sensor (the same samples were also imaged using an RGB sensor).

When imaged after approximately 30 seconds:

- fat and meat regions were distinguishable on the infrared image (grey level ratio of about 1.8).
- scratches were clearly more visible on an infrared image than on a corresponding RGB image. This was due to the difference in emissivity between skin and internal tissues, and wrinkled skin around the scratch increased its cooling rate.

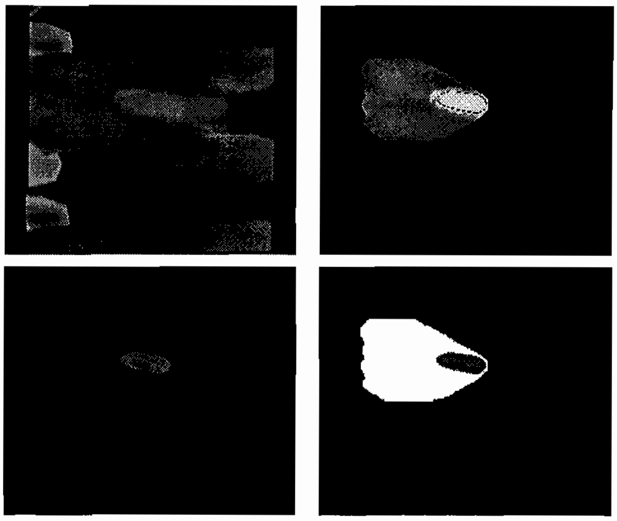
When imaged after approximately 60 seconds, fat blisters were visible in the infrared image. The meat feature had to be larger than 4 sq. cm to be reliably distinguished from artefacts such as water droplets which had a similar grey level.

The development and evaluation of algorithms for detecting Fat Blisters were carried out on a UNIX workstation using Khoros.

The algorithms developed consisted of two parts:

- The first part extracted the breast region from the rest of the image. This algorithm was developed using techniques similar to that developed by DIE and CEO for processing chicken RGB.
- The second part identified and separated the regions containing the fat blisters.

The example below shows the results of processing an infrared image, each image is the output from each stage in the process.



The results obtained proved to be better than expected despite the use of images of "old" chicken samples instead of the fresh samples found on a typical production line. However, only a small number of samples were imaged because of the difficulty in obtaining samples of chickens with fat blisters. Such samples were extremely rare in Italy where OG's laboratories were situated.

5.6.4 PIG LEG RGB IMAGE PROCESSING MODULE

The pig leg RGB image processing module consisted of a series of image processing algorithms for:-

- extracting the pig leg silhouette
- segmenting it into its sub-parts (hip, leg and trotter)
- identifying skin areas containing a meat feature
- classifying the meat features identified

The techniques for extracting the pig leg silhouette and segmenting it into its sub-part (hip, leg and trotter) were very similar to those developed for chickens. However, a slightly different technique was used for feature detection. The method used for chickens could not be used for pig legs because of the relatively poor quality of the pig leg images, and the considerable variation in the colour of "healthy" pig skin. The colours associated with normal "healthy" skin varied between samples and within a sample. However, once the features were identified, their classification was carried out using the same technique as that of chicken.

The poor quality of the images were caused by the choice of background during the image acquisition phase. Unlike chickens which were hung and imaged against a blue background, the pig legs were placed on a black conveyor belt. This resulted in images with a relatively constant average brightness (luminance). This was caused by the automatic adjustment of the aperture of the RGB sensor (TV camera) according to the amount of light entering the camera. As the background was black, the light entering the camera was proportional to the area represented by the non-black object, i.e. the pig leg sample. Big objects such as pig legs meant more light entered the camera, resulting in the adjustment of the aperture so as to reduce the luminance in the image. The result was a library of images with a constant average luminance value.

An attempt was made to improve the quality of the image by developing a pre-processing algorithm. This consisted of converting the images to a colour space in which the "brightness" component was separated from the colour components. The "brightness" was then adjusted using values from a reference image, whilst still maintaining the values of the original colour components. Although this technique only brought about a slight improvement in the quality of the images, the pre-processed images were used in the development of some of the feature detection and classification algorithms.

The segmentation of the pig leg into sub-parts included an additional algorithm for identifying the cut-surface of the leg in one of the views (5 views per pig leg, one view included the cut-surface). The algorithm identified the areas of skin, fat and meat, where the cut-surface consisted of only fat and meat.

The variation in the colour of features between samples and within a sample meant that the function which defined healthy and non-healthy skin areas was quite different from that of chicken. The variation in colour also resulted in more regions of overlap, i.e. a pixel would be associated with both healthy and non-health skin areas.

The algorithms for detecting features were written in a modular manner with each module (called an expert) dedicated to identifying regions where a particular feature was present. The "experts" were developed using a Neural Network" approach.

The image processing algorithms were only developed for identifying features that were considered by COREN/FRIGOLOURO to be of utmost importance in assessing the quality of their pig legs. The features of utmost importance were:

- Bruise
- Petechiae
- Trotter defect
- Leg bone exposed
- Stains
- Burns
- Bad stamp

Algorithms were developed for the reliable detection of the features above with the exception of Petechiae which was unreliable given the low resolution of the images. Images of a higher resolution would be needed in order to develop image processing algorithms for identifying regions of Petechiae.

The table below shows the comparison of the output (mapped class) from the pig leg RGB image processing module with that obtained from humans (true class). The output was obtained from processing 3 views of 63 pig leg samples.

	True class						
Mapped class	**Healthy**	**Bruises**	**Stains**	**Burns**	**Trotter**	**Leg bone**	**Stamp**
Healthy	NA[3]	0	11	0	0	0	0
Bruises	8	98	1	1	NA[2]	0	0
Stains	19	2	136	1	0	0	0
Burns	2	0	0	15	NA[2]	0	0
Trotter	9	NA[2]	1	NA[2]	26	0	0
Leg bone	0	0	0	0	0	1 [1]	0
Stamp	2	1	0	0	0	0	57

1. There was only one example of the feature "Leg bone exposed" in the 3 views used.
2. NA = Not Applicable. The presence of the feature "defective trotter" depended on the presence of bruises or burns or regions of abnormal colour on the trotter, i.e. bruises or burns detected on the trotter were also classified as trotter defect.
3. Not applicable.

By adjusting certain parameters defined in the image processing algorithms, lower values of P(missing a feature) (probability of "missing a feature") were obtained. However, making the adjustments resulted in higher values of P(false alarms) as shown below.

	Bruises	**Stains**	**Burns**	**Trotter**	**Leg Bone**	**Stamp**
P(false alarm)%	9.2%	13.9%	11.7%	5.5%	0%	5%

The accurate assessment of the quality of a pig leg cannot be done without detecting the presence of additional meat features such as broken coxal bones and hematoma on the cut-surface.

5.6.5 PIG LEG X-RAY IMAGE PROCESSING MODULE

The use of x-ray sensors for the on-line automatic inspection of pig legs was investigated by OG. X-ray sensors were used in developing algorithms for identifying the bones in a pig leg.

Prior to developing the algorithms, x-ray images were acquired at the OG laboratories using a line scan x-ray sensor. The images from the line scan sensor varied in quality. The top and bottom part of each image were of better quality than the middle part. This was caused by the automatic gain control found on most x-ray equipment which could not be appropriately tuned to cater for the variation in the thickness of the leg. As a result, image processing techniques for adjusting the intensity were applied to the images.

The linear array based equipment used in acquiring the x-ray images of pig legs was based on an L shaped sensor (derived from a standard luggage inspection system). Although, the L shaped sensor provides a quasi-3-dimensional view of a luggage item, this was not necessary for pig legs. As a result, the images were pre-processed in order to extract the image of the pig leg from the original image as shown below.

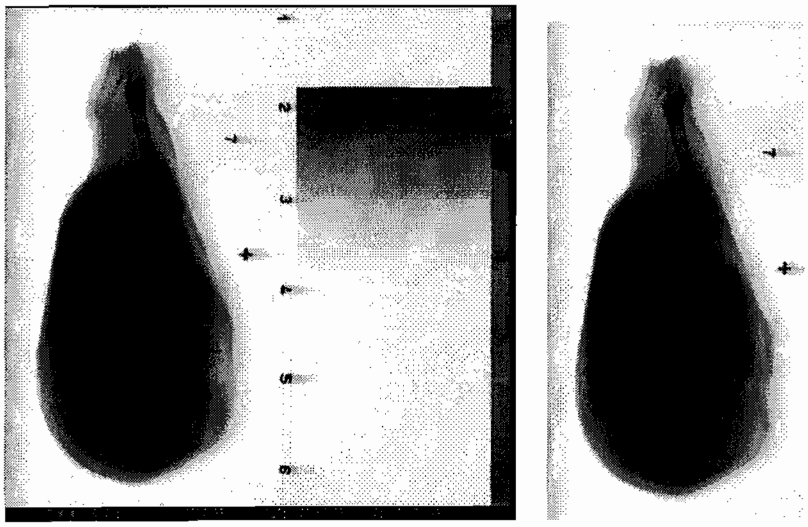
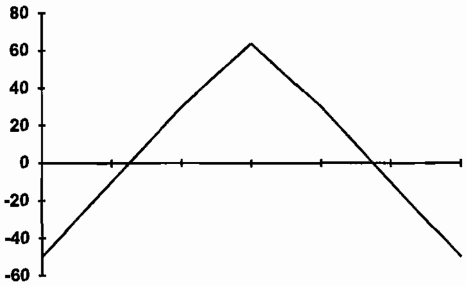


Image processing algorithms were developed to identify the region containing the bones, and to emphasise the bone edge. These algorithms were developed on a UNIX workstation using Khoros. The algorithms consisted of two functional modules. One module processed the x-ray image in order to enhance the contour of the bones, whilst the other extracted the region containing the bone.

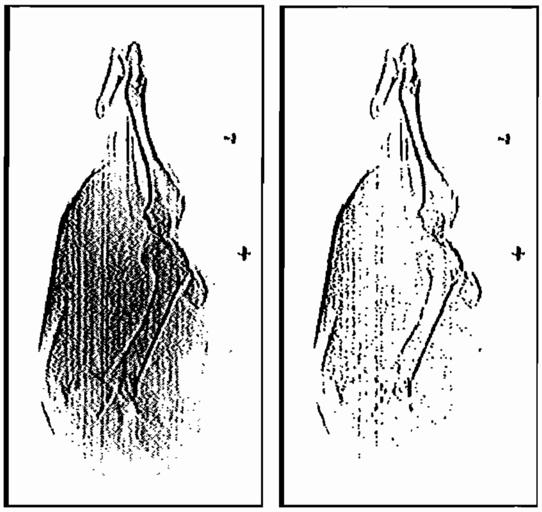
The module for enhancing the edges of the vertical bones performed a one-dimensional 7x1 "TOP-HAT" convolution kernel containing the following values:

k_1	k_2	k_3	k_4	k_5	k_6	k_7
-50	-10	30	64	30	-10	-50

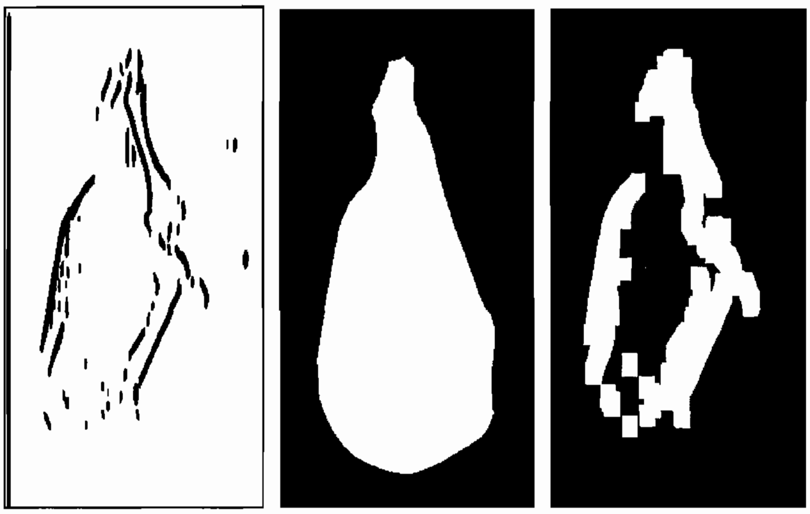
and with the following shape:



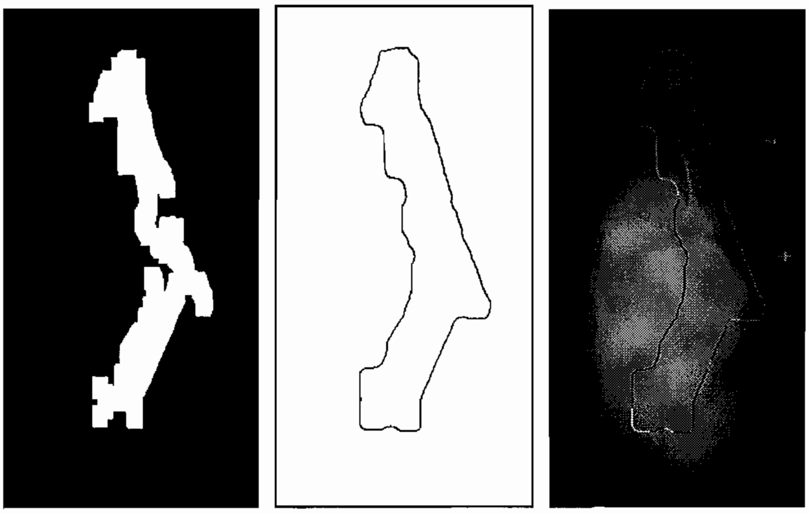
Since the edges were located in the horizontal (X) direction, they were detected by using a one-dimensional convolution operation performed in the X direction. The kernel enhanced the very smooth edges present in the x-ray images, but the led to an increase in noise.
A threshold was applied to the "noisy" image followed by 3x3 median filter process to remove the spots (noise) as shown below.



A convolution operation was performed to enlarge the enhanced contours in the vertical direction. The resulting image was manipulated by performing an "AND" operation with a "binary eroded" image of the pig leg in order to extract the regions containing the enhanced contours. A morphological operation was then performed to thicken the extracted contour regions as shown below.



The next module extracted the region containing the bones by merging "connected" pixels. Smoothing of the edge of the region extracted was performed using "dilation" and morphological operations. The final image was obtained by performing a binary addition of the extracted region with the original image of the pig leg as shown below on the right.



The algorithms developed identified the regions containing the bones successfully in 8 out of the 10 samples tested. The bone edges of 2 of the samples were easily identified in the top and bottom parts of the leg, but not reliably identified in the middle part of the leg, even though image enhancement allowed a good visual interpretation of the image. It must be noted that the algorithms could fail if metals were present in the samples imaged

Further development of x-ray equipment would result in clearer images. Clearer images would ultimately lead to more reliable techniques for identifying bone edges, especially in the middle part of the leg.

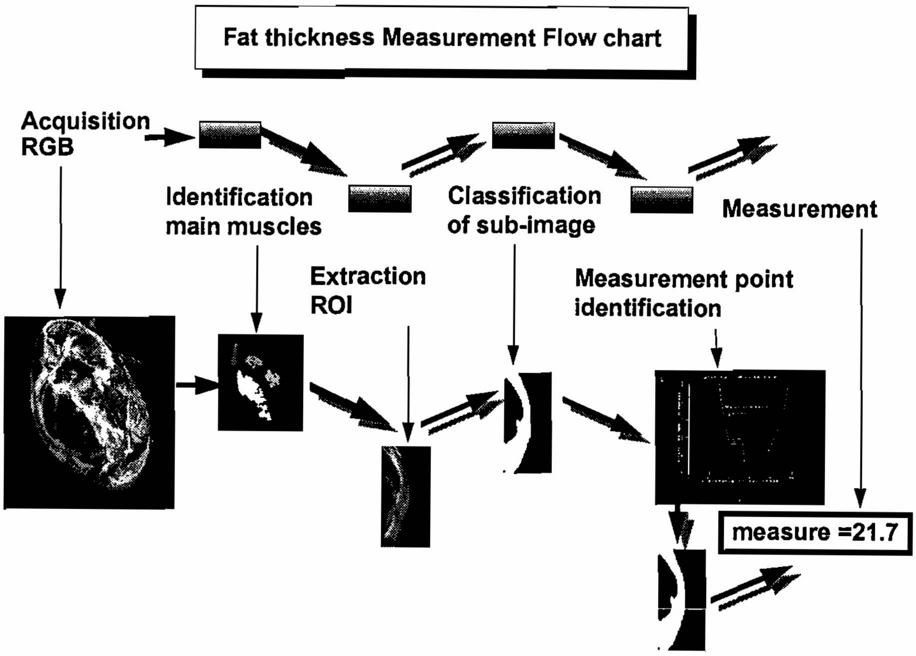
5.6.6 Pig Leg Fat Measurement Image Processing Module

The fat thickness measured on the cut-surface of a pig leg was found to be very important in determining the suitability of the leg for Spanish dried cured ham products. The trimmers (or experts) at COREN/FRIGOLOURO measured the fat thickness by a visual inspection of the cut-surface. For a leg to be suitable, the fat thickness must be between 5 and 20 mm.

Thus, the assessment of the quality of pig legs required the development of an image processing module for measuring fat thickness. The module was developed by BULL in partnership with DIE and UVIGO. A special set of images of the cut-surface were acquired during the image acquisition phase by UVIGO. The fat thickness of the samples imaged were measured by the human experts at COREN-FRIGOLOURO and recorded in the image headers.

The image processing module was developed to:

- identify the muscle areas
- extract the region of interest
- identify the fat in the region extracted
- determine the measurement point
- measure the fat thickness



The identification of the main muscles was carried out by:

- processing RGB sampled images (sampling ratio of 6)
- using the Khoros "*vkmeans*" clustering algorithm
- retaining the cluster corresponding to meat areas.

The extraction of the region of interest was carried out by:

- identifying the region with the largest number of pixels
- computing the inclination of the region's principal axis (the slant of the main muscle determines if it is a right or left leg)
- extracting the region of interest based on the above.

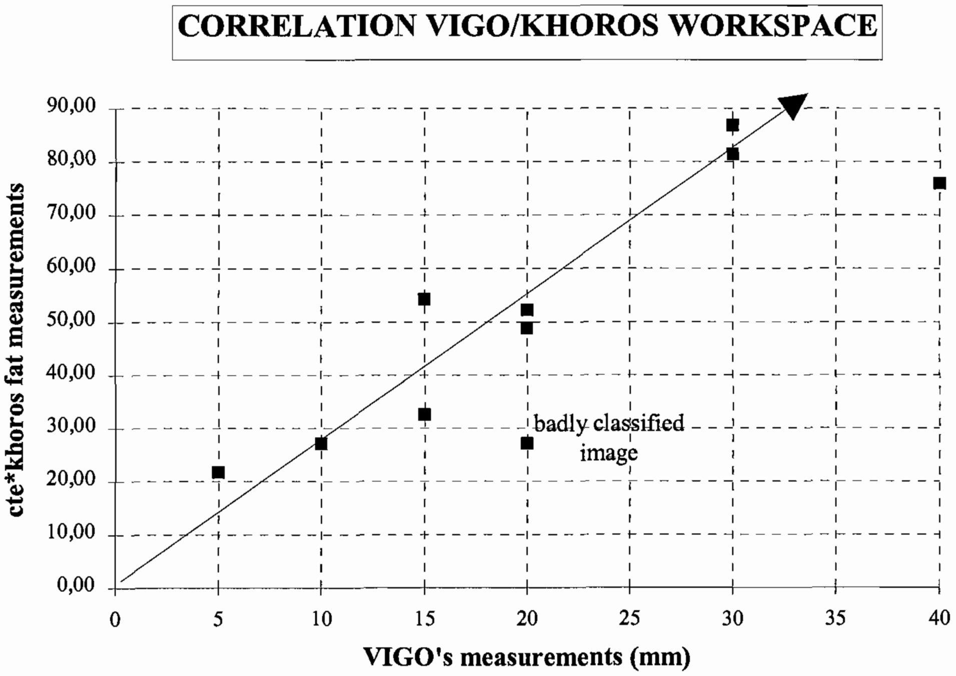
The classification of the extracted region into background, meat and fat areas was carried by using a 4 level colour "quantization" on a 2 banded image (saturation & value) (the sampling ratio with respect to the original image was only 3).

The identification of the measurement point (deducing the summit of the second muscle starting from the top of the extracted image) was carried out by:

- calculating the distance along the y-axis from the left side of the extracted image to the meat/fat boundary
- smoothing and normalising the distance calculated

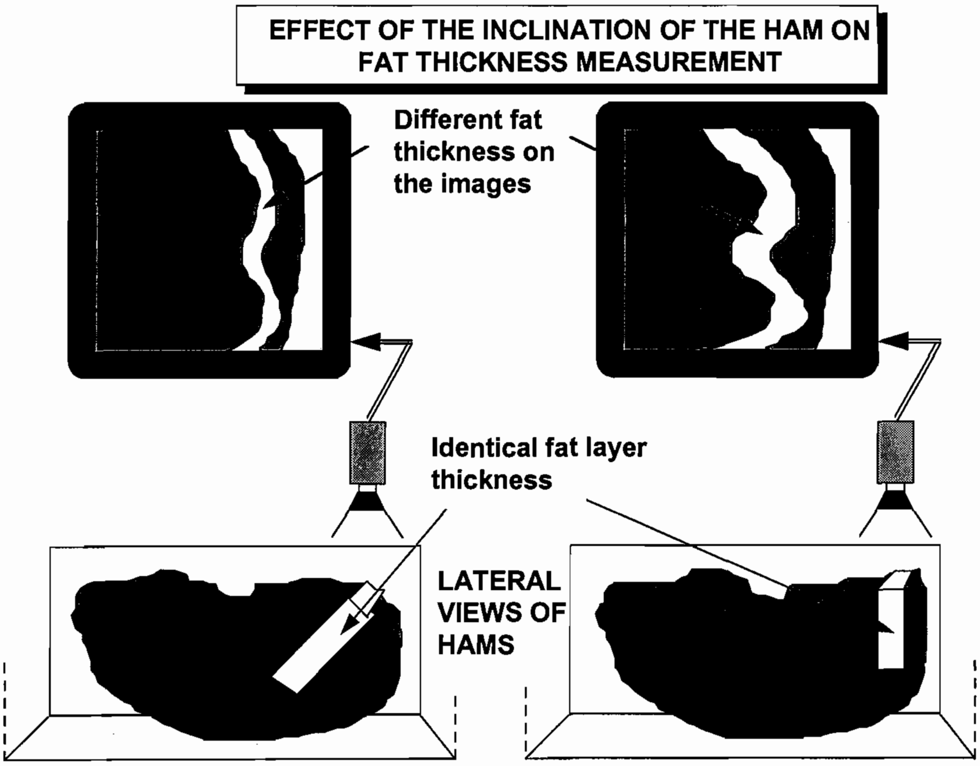
The measurement at the point identified was carried out by counting the "white" pixels along a horizontal sampling line.

The image processing algorithms were evaluated using the 10 images acquired during the image acquisition phase. A correlation of the measurements obtained (expressed in number of pixels) from the image processing module with the "human" measurements recorded in the headers of the 10 images by UVIGO is shown below.



The classification of pixels into areas of fat and meat was critical to the process of measuring the fat thickness. For the 10th sample (marked above as a "badly classified image"), there were more shadows in the fat regions of the image, and bright areas representing light reflected from meat regions. The pixels corresponding to shadows in fat were not classified as fat pixels, and pixels corresponding to areas of reflection in meat were not classified as meat. To improve the process of classification, particular attention must be paid to the image acquisition phase in order to maximise the light "uniformity".

An exact correlation between the measurements obtained from the module and the "human" measurements was not possible because of variations in the inclination of the surface of the cut and the RGB TV camera during the image acquisition phase. In order to obtain more consistent measurements, a sensor for assessing the inclination of the cut surface could be placed in front of the main camera. The sensor would feed its images to a separate image processing module which would calculate the inclination of the leg.



5.7 EXPERT SYSTEM

The function of the Expert System was to take the features of the meat samples recognised by the Intelligent Vision System and deduce the appropriate action for ALINSPEC to take.

The Expert System should:

- recognise patterns of features related to a single cause ("characteristics or syndromes").
- given the features and syndromes identified, decide on what should be done with the meat sample ("sample strategy").
- "remember" features and syndromes seen in earlier samples and recognise changes in the frequency with which they occurred ("trend detection").
- having recognised significant changes in the incidence of particular features, deduce their cause ("upstream causes").

In addition, the Expert System should determine when there was a need to "tune" the imaging and image analysis systems. When this happened, the Expert System should deduce the appropriate changes to be made.

The Expert System should communicate its decisions and actions to factory personnel via the Man Machine Interface module. The Expert System should communicate with the other modules via an Ethernet TCP/IP network.

The functions described above made up the objectives of the project work programme WP6: Expert System. In addition to these functions, the Expert System should maintain a detailed log of meat information processed by ALINSPEC. The information logged should include the features and characteristics seen in each sample, and the decision as to what the sample should be used for. The information should be used in producing the following reports:

- a detailed report listing the features/characteristics and decisions for each sample. This report should be stored in a format suitable for processing by spreadsheets and other report generators
- a summary report which gives the factory personal some indication of the quality of a batch of samples.

The Expert System should produce the reports when requested by the factory personnel (via the Man Machine Interface).

The final Expert System delivered to the ALINSPEC system integrator (BULL) consisted of two modules coded entirely in the ANSI C programming language. The modules ran as separate processes and communicated with each other via files. The Expert System was developed as two separate modules so the functions of the system could be run at different priorities by modifying the priority of each process. The function of "sample strategy" required a higher priority than that of "trend detection".

The first module carried out the functions defined by the work programmes WP6.1 Characteristics, WP6.2 Sample Strategy, WP6.5 Vision System Tuning and WP6.6 Requirements for Integration. These were:

- communication of the Expert System with the other two modules, i.e. the Intelligent Vision System and Man Machine Interface.
- assessment of quality and grading meat using a custom inference engine, quality profile(s) and the meat feature information from the vision system.
- tuning of the vision system when ALINSPEC was off-line.

The "sample strategy" function was developed outside the Nexpert environment making the Expert System more efficient and portable.

The second module carried out the functions defined in the work programmes WP6.3 Trend Detection and WP6.4 Upstream Causes. This was the deduction of any significant changes in the quality of the meat inspected by ALINSPEC, and the event which caused the change. The module also logged all appropriate meat sample information, and when required, produced a detailed and summary report from this information.

The final Expert System software was only tested for the chicken inspection application. The pig leg application could not be tested as there was insufficient IVS pig leg feature information for developing a quality profile.

The ES software module was written in the C programming language to allow it to be easily ported onto a parallel processing platform. The ES was developed as two separate modules, thus allowing its functions to be executed at different priorities.

- The ES Shell was responsible for communicating with other modules, grading meat samples and tuning the IVS when necessary
- The Background Function was responsible for deducing significant changes in quality of meat samples examined, and producing detailed and summary reports.

The ES also maintained a log of all the information generated for each sample (i.e. the features reported and the grades assigned to each sample).

5.7.1 CHARACTERISTICS (SYNDROMES)

The data fusion and recognition modules (part of the Intelligent Vision system) recognised features of each sample e.g. bruises, torn skin, fractures etc., whilst the Expert System recognised patterns of features related to a single cause (syndromes or characteristics). For example, the Expert System should deduce that torn skin, fracture and bruising are all features of a fractured limb. CCFRA and DIE defined the format in which meat features recognised by the Intelligent Vision System were made available to the Expert System. The rules required in deducing characteristics from meat features were elicited from the human experts (domain experts) at COREN/FRIGOLOURO, and occasionally from experts at slaughter houses in the UK.

Meat feature information recognised by the Intelligent Vision System should be passed to the Expert System. The format for transferring this information was defined by the C structure shown below. One record of this structure reported one meat feature, i.e. the feature code, the location, colour and size. The conformation of the meat sample was reported as the aspect ratio of the major body parts, e.g. for chickens, the aspect ratio of the breast, wings and legs were reported.

```
struct es_features_rec
      {
       unsigned char type_of_message;
       unsigned char message_id;
       unsigned char line_type;
       unsigned char line_no;
       unsigned long sample_no;                  /* meat sample number */
       char comment[16];
       unsigned char feature_id;                 /* meat feature code */
       unsigned char colour_id;                  /* colour */
       unsigned char d_confidence;
       unsigned char c_confidence;
       unsigned long feature_present;
       unsigned long feature_size_or_value_1;    /* size  or parameter */
       unsigned long feature_size_or_value_2;    /* additional parameter
                                                 e.g. if reporting aspect-ratio*/
       char location[8];                                /* location code */
      };
```

The location of meat features were reported as codes which were defined by CCFRA in the project handbook.

5.7.2 PER SAMPLE STRATEGY.

Given the features and characteristics identified for a particular piece of meat, the Expert System should decide how that piece of meat should be treated Three methods were investigated in the development of the "Sample Strategy" function of the Expert System. The methods also reflected the change in requirements of ALINSPEC from a more general meat inspection system, to a system tailored to meet the specific requirements of

COREN/FRIGOLOURO, i.e. inspect and assess the quality of chickens and pig legs leading to the allocation of a grade, and the ability to change the assessment criteria.

1. **Nexpert Inference Engine**:- Knowledge bases consisting of rules were built using the Expert System development tool "Nexpert". Using meat feature information, the Nexpert Inference Engine evaluated the rules in the knowledge base to arrive at a conclusion. Two prototypes were developed. The chicken prototype produced a quality grade, and the pig leg prototype made a suggestion as to how a leg should be trimmed.

This method was found to be inadequate in meeting COREN/FRIGOLOURO's requirement for modifying the quality assessment criteria. Any changes made to the knowledge base would have had a much wider impact than anticipated because of the structure of the rules. Also, any changes to the knowledge base would require some expertise of Nexpert.

However, the knowledge bases were maintained and used as a means of storing knowledge elicited from the COREN/FRIGOLOURO human experts.

2. **Mathematical Model**:- A statistically based mathematical model of the quality assessment and grading process was developed. A functional form of the model and the data required to fit the model were defined. The Intelligent Vision System output data for white chickens was fitted to the model and its validity checked using statistical methods. This method would have provided a simple means of modifying the quality assessment criteria. However, the data in terms of quality and volume needed to improve the model required additional resources to be allocated to the image capture and analysis tasks of the project.

3. **Custom Inference Engine**:- An inference engine was designed to evaluate a set of "simple" user definable quality assessment rules. These rules were grouped together to form a "quality profile". Quality profiles were specified by the operator via the Man Machine Interface screens. The inference engine evaluated the rules using meat feature information reported by the vision system, and this led to a deduction of the grade or decision for the sample inspected. The rules were simple, making the modification of the quality assessment relatively easy.

The custom inference engine was written in the standard ANSI C programming language, eliminating the need for a Nexpert run-time license or Nexpert expertise. The result was an Expert System which was easily portable from one computer system to another.

The development of software to carry out the "sample strategy" function was carried out by building prototypes. Each prototype used a different technique for the assessment and grading of meat.

a) chicken prototype using Nexpert.

The domain of the chicken Expert System was modelled in terms of classes, objects and properties. Features, characteristics (syndromes) and decisions (actions) were represented as classes. When a feature was reported for a sample, or a characteristic deduced, an object was created under the appropriate class. The object automatically inherited the properties of the class, e.g. the properties of an object created under the class "feature" were size, colour and location. A set of rules was evaluated to deduce the characteristics resulting in the creation of "characteristic" objects. Another set of rules was evaluated to deduce the action to be taken on the sample.

The rules were fired by the Nexpert inference engine using the backward chaining mechanism. The rules were chained together, with the evaluation of one rule requiring the evaluation of another rule. By evaluating the first rule in the chain, the appropriate rules to the left were evaluated until the hypothesis of the first rule was either true or false. The firing of the rules ultimately led to one or more conclusions (decisions).

Using both fictional and real meat feature information (derived from the image library "header" files) as input data, the output characteristics and decisions were compared with that of a human expert.

For the "Characteristic Identification" function:

	Number correct	**%**
50 "real" chickens	44	88
40 "fictional" chickens	34	85
20 "fictional" pig legs	13	65

For the "Sample Strategy" function:

	Number correct	**%**
44 "real" chickens	31	70
34 "fictional" chickens	24	71
13 "fictional" pig legs	6	46

The "real" chicken data was extracted from image library headers and as such did not represent the output from the Intelligent Vision System. This prototype was developed before meat feature information was made available by the team working on the Intelligent Vision System.

Although these initial results were encouraging, this method was found to be inadequate in meeting COREN/FRIGOLOURO's requirement for modifying the quality assessment criteria. Any changes to the knowledge base could have a much wider impact than anticipated because of the way the rules were chained together. Also, any changes to the knowledge base would require some expertise in Nexpert.

b) pig leg prototype using Nexpert.

The pig leg prototype was very similar to the chicken prototype.

The domain of the pig leg Expert System was also modelled in terms of classes, objects and properties. An additional class "set-up" was introduced to allow the user to modify a fixed number of parameters used in the evaluation of rules. The rules were written such that certain properties of "meat feature" objects could be compared with corresponding properties of "set-up" objects. This would allow the operator to modify the quality assessment criteria or grading mechanism of the system by changing the properties of "set-up" objects. The rules were also chained together and fired by the Nexpert inference engine using the backward chaining mechanism.

Using the Intelligent Vision System output for pig legs as input data, the output grades were compared with that of a human expert:

	ALINSPEC						
Real	**Serrano**	**Rounded with trotter**	**Rounded without trotter**	**York**	**Fresh meat**	**Condemn**	**Total**
Serrano	0	0	0	0	0	0	0
Rounded with trotter	1	0	0	2	0	0	3
Rounded without trotter	0	0	0	1	0	0	1
York	2	2	0	31	8	0	43
Fresh meat	1	1	0	8	1	0	11
Condemn	0	0	0	0	0	0	0
Total	4	3	0	42	9	0	58

	Number of samples	**%**
Same Grade	32	55
One grade out	18	31
Two grades out	4	7
Three grades out	3	5
Four grades out	1	2
Total	58	100

These results were encouraging given the quantity and quality of the input data available at the time. Although this prototype provided a mechanism for changing the quality assessment criteria, changes made could have a wider impact than anticipated because of the way the rules were chained together. To guarantee the integrity of the knowledge base, the effect of all the possible changes would have to be evaluated and this would require additional resources in terms of time, effort and test data. Although the development of the chicken and pig leg Nexpert prototypes were discontinued, the knowledge bases were maintained and used as a means of storing knowledge elicited from the COREN/FRIGOLOURO human experts.

c) chicken prototype using a mathematical model.

A more conceptually simple and easily modifiable reasoning mechanism was developed using a statistically based mathematical model of the quality assessment process.

Photographs of the images of chickens in the image library were assessed by the human experts from COREN/FRIGOLOURO, and the quality scored on a continuous scale of 0 to

10 ("real grades"). The vision system output available at the time meant that only the quality of the breast area of each chicken image was assessed by the human expert.

A functional form of the model was defined using the Regression Analysis tool in the statistical package "Minitab". Regression Analysis utilises the relationship between two or more quantitative variables so that one variable can be predicted from the other. A regression of the "real grades" quality scores against the vision system output revealed the following relationship.

$$\text{Quality} = 6.35 - 0.000995\ \text{CF1\_BLUE} + 0.000209\ \text{CF1\_PURP} - 0.000524\ \text{CF1\_RED} - 0.00326\ \text{CF42} - 0.00128\ \text{CF9} - 1.77\ \text{CF41} - 0.00183\ \text{CF34} - 5.75\ \text{CF28}$$

Where

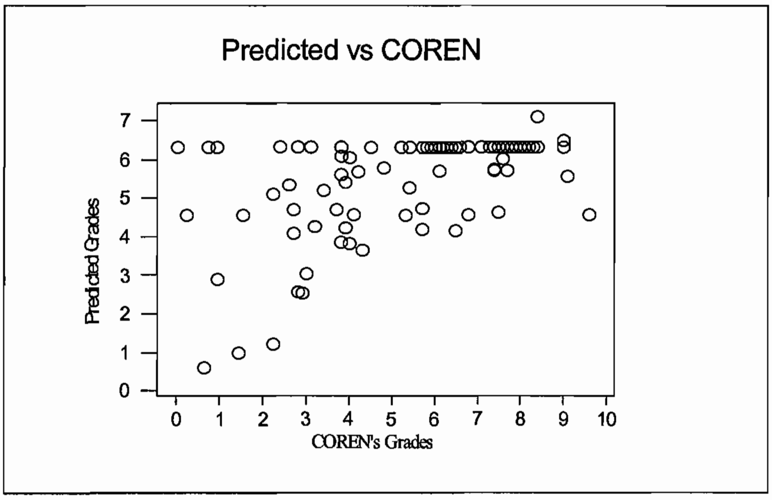
CF1_BLUE is the size of a blue bruise
CF1_PURP is the size of a purple bruise
CF1_RED is the size of a red bruise
CF42 is the size of a stain on the skin
CF9 is the size of a breast blister
CF41 is the size of a stain or skin area of abnormal colour (in some cases the IVS may not be able to distinguish between the two)
CF34 is the size of a litter spot
CF28 is a flag to indicate if the colour of the entire chicken is abnormal

Using this equation to predict the quality and comparing the grades obtained with that of the human expert:

	ALINSPEC			
Real	**Grade A**	**Grade B**	**Grade C**	**Total**
Grade A	7	14	0	21
Grade B	0	45	0	45
Grade C	0	37	1	38
Total	7	96	1	104

	Number of samples	**%**
Same Grade	53	51
One grade out	51	49
Two grades out	0	0
Total	104	100

These results indicated that this approach was feasible, but the data, in terms of quality and volume, needed to improve the model required additional resources to be allocated to the image capture and analysis tasks of the project. As a result, the development of this prototype was discontinued. The following graph represents the output from the Regression Analysis tool in Minitab. The plotting of predicted scores against the real scores revealed that the model tended to predict a higher grade, indicating that the vision system output was inadequate for generating a fully functional model.



d) the final Expert System using a custom inference engine.

Further intensive interaction with COREN/FRIGOLOURO human experts revealed the precise modifiability requirements and this led to the definition of a quality profile in terms of the maximum extent of a meat feature/characteristic acceptable in a given category. The requirement to code the knowledge elicited from human experts in a manner which made it easily modifiable led to the following simplifications:

1. Multiple occurrences of the same feature within the same location, have the same effect on quality as one feature of size equal to the sum of the individual feature sizes.
2. Each feature on its own can limit the grade of the sample.
3. The effect of reported features on the grade of a sample were independent of each other.
4. The grade of the sample is the lowest of the grades attributable to each feature.

The use of the "OR" logical operator in chaining Nexpert knowledge base meant that any changes made to one rule could have a much wider impact than anticipated. By avoiding the "OR" logical operator, the rules in the chicken Nexpert prototype knowledge base were simplified, with each rule leading to a conclusion which is independent of the other.

For example the Nexpert rule:

If the following properties of the chicken feature object bruise are:
    location is precisely equal to "breast"
    size is greater than or equal to 0 mm^2
    size is less than 442 mm^2
    colour is precisely equal to "blue"
then characteristic grade_1_bruise_on_the_breast is present.

if the characteristic grade_1_bruise_on_the breast is present
OR this other characteristic is present.....
OR ....
:

:
then the sample is grade A

became

If the chicken feature/characteristic bluish-bruise is present on the upper breast, and its size is between 0 and 442 mm^2, then the sample's grade cannot be higher than 0 (Grade A).

If the chicken feature/characteristic bluish-bruise is present on the mid breast, and its size is between 0 and 442 mm^2, then the sample's grade cannot be higher than 0 (Grade A).

If the chicken feature/characteristic bluish-bruise is present on the lower breast, and its size is between 0 and 442 mm^2, then the sample's grade cannot be higher than 0 (Grade A).
:
:
The sample is assigned the lowest of the grades resulting from the evaluation of each rule.

A format for storing these rules was defined by OG and CCFRA. The rules were grouped together to form a quality profile, with each profile representing a set of quality assessment criteria. The Man Machine Interface module designed by OG provided the operator with screens for creating or modifying rules of a quality profile and stored in the format shown below.

```
RULE
   Feature ......... = Bluish Bruise
   Location ........ = Upper Breast
   Print ........... = No
   Nmb of ITEMS .... = 3

   ITEM #1
      Low .......... = 0 mm²
      High ......... = 441 mm²
      Grade Level .. = 1

   ITEM #2
      Low .......... = 442 mm²
      High ......... = 1261 mm²
      Grade Level .. = 2

   ITEM #3
      Low .......... = 1262 mm²
      High ......... = 4000 mm²
      Grade Level .. = 3
```

A method was developed for processing meat feature information against quality profiles in order to determine the grade of the meat sample (a quality profile also contained a list of possible grades).

- The meat sample was initially assigned the highest grade in the profile.
- For each meat feature reported for the sample:
  - The appropriate rules for that feature were evaluated resulting in a new grade.
  - If the new grade was of a lower quality than the previous grade, the sample was assigned the new grade, else it retained the previous grade.

If no meat features were reported for a sample, then the sample's initial grade would be its final grade.

The inference engine allowed the simultaneous processing of feature data against several quality profiles, thus allowing a sample to be graded against multiple, different, quality assessment criteria, e.g. if two profiles were defined, the Expert System's output would be two grades for each sample.

A quality profile of the current COREN/FRIGOLOURO chicken quality assessment criteria was produced as part of the development of the custom inference engine. The chickens in the image library were assessed on a computer screen and graded by the human experts. The vision system output available at this stage of the project meant that the entire front view of each chicken image was assessed by the human expert.

Using the human expert grades, meat feature information from the vision system, and a prototype of the Man Machine Interface, the profile was produced by first converting some of the chicken Nexpert knowledge base rules to simpler quality profile rules. The profile rules were then modified in an iterative fashion to give the closest correspondence between the human and ALINSPEC grades. The following results were obtained at the end of this process:

	Predicted			
Real	**Grade A**	**Grade B**	**Grade C**	**ALL**
Grade A	42	3	2	47
Grade B	11	55	4	70
Grade C	4	2	23	29
ALL	57	60	29	146

	Number	**%**
Same Grade	120	82
One grade out	20	14
Two grades out	6	4

The six samples, two grades out, were due to the skin surface features (e.g. "Toad Skin") not reported by the Intelligent Vision System, hence, the appropriate rules were not evaluated by the engine. The 82% success rate obtained was a marked improvement on the rates obtained from the earlier prototypes, i.e. 51% and 55%. The success rate depended, to a large extent, on the quality and quantity of information reported by the Intelligent Vision System.

The final Expert System satisfied COREN/FRIGOLOURO's requirements that the quality assessment criteria be easily modifiable and this could be done via the Man Machine Interface screens for creating and modifying quality profiles.

The custom inference engine was written in the standard ANSI C programming language. This eliminated the need for a Nexpert run-time license, and a Nexpert specialist in order to modify the knowledge base rules. It also made the Expert System more efficient and easier to port from one computer to another, e.g. by compilation to run on a purpose built parallel processor platform.

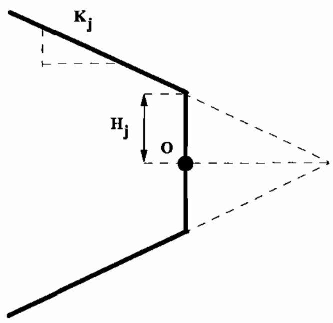
5.7.3 TREND DETECTION AND UPSTREAM CAUSES.

The Expert System should remember features and characteristics seen in earlier samples and recognise changes in the frequency with which they occurred. Having recognised a significant change in the incidence of a particular feature, the Expert System should deduce its cause.

Knowledge elicited from the human experts (domain experts) at COREN/FRIGOLOURO slaughter houses revealed that standard statistical quality control techniques were adequate for carrying out the functions of "trend detection".

The Cumulative-Sum (CUSUM) statistical quality control method is used for detecting small shifts in a production process. This method was chosen as the most appropriate tool for detecting significant shifts in the occurrence of defects. The method could be used in detecting shifts in either direction from a target value. However, ALINSPEC only required the detection of an upward shift from a zero target value.

The CUSUM makes use of a "V-mask" to decide when a significant shift has occurred.



V-mask shape

Two parameters defined the "V-mask" for a defect j:

K_j	=	Slope of the arms of the "V-mask" for defect j.
H_j	=	Decision interval for defect j.

The tabular form of the CUSUM method was used as it was particularly useful in implementing the CUSUM on a computer. The CUSUM for each defect was calculated after each meat sample had been assessed and graded.

The upper CUSUM for defect j was calculated as follows:

If

$S_j(i)$ = Upper CUSUM for defect j after processing sample i
$S_H(i\text{-}1)$ = Previous Upper CUSUM for defect j
m_j = Pre-defined Target value for defect j.
x_i = Defect j present or absent in sample i
0 = absent
1 = present.

Then

$$S_j(i) = \max\,[0,\ x_i - (m_j + K_j) + S_j(i\text{-}1)]$$

Since target m_j was always 0, therefore

$$S_j(i) = \max\,[0,\ x_i - K_j + S_j(i\text{-}1)]$$

where K_j is the slope of the V-mask.

If $S_j(i) \geq H_j$, then a significant upward shift was detected. On detecting a shift, the CUSUM was re-set to zero.

Knowledge elicitation carried out with COREN/FRIGOLOURO human experts revealed that relatively standard statistical quality control techniques were adequate for detecting and signalling trends and changes in the frequency of occurrence of defects.

The criteria for deducing the cause of such changes were represented as relatively simple rules which were coded in a tabular form and stored in a text file as shown below. The defects to be monitored by ALINSPEC were also stored in this file. The defects could either be a meat feature or a combination of meat features, e.g. old bruises consisted of both blue and purple bruises.

Defect Name (as stated by COREN Factory Manager)	**Characteristic/Defects (Name used by ALINSPEC)**	**Location of Characteristic**	**Responsible for Defect**	**Tolerance per Batch (%)**
Rubbed Skin	Rubbed Skin	Breast Legs Wings	Factory - De-Feathering Machine.	6-8
Fractured Wing (no bruising, clean fracture)	Fracture	Wings	Factory - De-Feathering Machine.	1-2
Scalded Skin	Scalded Skin	Whole Body	Factory - Scalder	2-5
Old Bruises	Bluish Bruise Purplish Bruise	Breast Legs Wings Breast Legs Wings	Farm	1-2

Defect Name (as stated by COREN Factory Manager)	Characteristic/Defects (Name used by ALINSPEC)	Location of Characteristic	Responsible for Defect	Tolerance per Batch (%)
New Bruises	Reddish Bruise	Breast Legs Wings	Transportation, Gathering or Slaughtering Process	6-8
Bruises on Wings	Bluish Bruise Purplish Bruise Reddish Bruise	Wings Wings Wings	Transportation, Gathering or Slaughtering Process	0-3
Blister (with/without scab)	Black Litter Spot Dark Brown Litter Spot Brown Litter Spot Breast Blister	Breast Legs Wings Breast Legs Wings Breast Legs Wings Breast	Farm	0.5-1.0
Fractured leg	Fracture	Legs	Transportation, Gathering or Slaughtering Process	0-6
Third Level Chickens (condemned samples)	N/A	N/A	Farm	0.2-0.8
Fat Blister	* not detected by ALINSPEC	NA	Farm - Feed	5-12
Ammonia Burn at Feet	* not detected by ALINSPEC, ALINSPEC does not examine the feet.	Legs	Farm	10-15
Ammonia Burn on Breast	* not detected by ALINSPEC, ALINSPEC cannot distinguish between Ammonia Burn and Breast Blister	Breast	Farm	2-5
Dirty Nails	* not detected by ALINSPEC	NA	Farm	
Bites	* not detected by ALINSPEC	NA	Farm	
Scratches	* not detected by ALINSPEC	NA	Farm or Gathering Process	5-9

Other parameters to be monitored were the average weight of the samples and samples which were not assigned the highest grade.

Defect Name/Parameters (as stated by COREN Factory Manager)	Characteristic/Defect (Name used by ALINSPEC)	Location of Characteristic	Tolerance per Batch (%)
Toad Skin	* not detected by ALINSPEC	NA	N/A
Second Level (samples graded as B)	N/A	N/A	N/A
Average Weight	Weight	Whole Body	N/A

Deducing trends was done by using the Cumulative-Sum (CUSUM) statistical quality control method. The CUSUM method was used for detecting small shifts in:

- the number of samples having a particular defect;
- the average weight;
- the number of samples assigned a grade other than the highest possible quality grade.

The final Expert System consisted of two modules, the first module handled the communication and grading. The second module calculated the CUSUM for each defect after each sample had been graded. The criteria for detecting that a significant shift had occurred was defined by values of the CUSUM V-mask slope and the decision interval, which were stored for each defect in the table shown above.

When a shift was detected, the likely upstream cause of the shift was deduced from the information contained in the table, and a warning message transmitted to the operator via the Man Machine Interface module.

In addition to deducing significant shifts, the module also monitored the quality of a batch of meat samples by calculating the percentage of samples in a batch having a particular defect. This was done after a batch of samples had been processed. The value calculated for each defect was compared with its pre-defined "warning" and "alarm" levels. If the value exceeded the "warning" level, but not the "alarm" level, then a warning message was issued to the operator via the Man Machine Interface module. If the value exceeded the "alarm" level, then an alarm was raised.

The second module was also responsible for producing a set of reports for a specified batch of meat samples. The reports were produced on receiving a "report request" message from the operator via the Man Machine Interface module. The reports were of two types:

1. A detailed report consisting of fixed length records, with each record (or row) containing all the information generated for a sample. Each record consisted of the grades allocated by the Expert System and the sizes (or aspect ratios) of features reported by the Intelligent Vision System. The report was produced in a format which made it suitable for processing by computer spreadsheet packages.

2. A summary report (example shown below) summarising the information contained in the detailed report as percentages of features identified in a batch of samples. The report gave an indication of the overall quality of the batch.

Summary report on a batch of chickens.

Batch : DEFAULT
Date Produced : 1996 03 25 14:04:50.218
Total number of samples processed : 146
Average weight : 543.452

Defect/Parameter	% Present in BATCH	Min. Tolerance	Max. Tolerance
Rubbed Skin	3.42	6.00	8.00
Fractured Wing (no bruising, clean fracture)	1.37	1.00	2.00
Scalded Skin	3.42	2.00	5.00
Old Bruises	14.38	1.00	2.00
New Bruises	41.10	6.00	8.00
Bruises on Wings	40.41	0.00	3.00
Blister (with/without scab)	17.12	0.50	1.00
Fractured leg	0.68	0.00	6.00
Third Level Chickens (condemned samples)	31.51	0.20	0.80
Second Level (samples graded as B)	30.82	100.00	100.00

Although the operation of the second module was tested using meat feature information from the chicken image library (about 300 samples processed by the Intelligent Vision System), no meaningful results could be obtained as a larger data set was needed to determine appropriate values for the V-mask slope and decision interval for each defect.

5.7.4 VISION SYSTEM TUNING.

The original requirement was for the Expert System to determine when there was a need to "tune" the imaging and image analysis systems, and when this happened, the Expert System would deduce the appropriate changes to be made.

The architecture of the Intelligent Vision system software was developed in a manner which required it to be trained on a set of library images. Once the parameters defining its sensitivity to the recognition of features were fixed during the training period, they could not be changed without repeating the training process. As a result, the function of the Expert System changed from "on-fly" tuning to deciding which set of pre-defined parameters, the Intelligent Vision System should use.

The Intelligent Vision System was tuned by the Expert System when the ALINSPEC system was off-line (it was not processing samples). This occurred after the Expert System received a "New Batch of samples" message (which included the type of product) from the operator via the Man Machine Interface module.

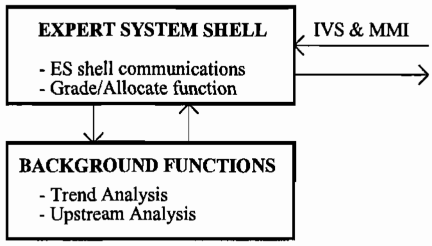
Since the parameters had been defined for each type of product (e.g. white or yellow chickens) during the development of the Intelligent Vision System, a file containing the appropriate parameters for a particular product was created. The name and location of the file was transmitted to the Intelligent Vision System prior to processing the new batch of samples.

5.7.5 REQUIREMENTS FOR INTEGRATION

The Expert System should take the meat features recognised by the Intelligent Vision System, process the information and then communicate its decisions and actions to the factory personnel via the Man Machine Interface module.

The modules were developed to run on separate computers. As a result, the Expert System should communicate with the other modules via an Ethernet TCP/IP network. A simulation of the communication of the Expert System with the other modules was carried out using the following methods: Sockets; Semaphores; and NFS (Network File System) mounted files. Although sockets and semaphores were initially chosen as the means of communication between modules running on separate computers, a NFS file based communication mechanism provided an additional advantage in that the files could be mapped directly to computer memory location if two or more modules were compiled to run on one computer.

The Expert System consisted of two modules, the first module, called the Shell, communicated with the Intelligent Vision System and the Man Machine Interface modules via a TCP/IP Network File System. The Shell also communicated with the second module, called the Background Process.



Integration of Expert System with other modules.

In line with the other modules, the Expert System Shell could be in any one of three major states:-

- STOPPED - The Expert System software was not running at all
- IDLE - The Expert System software was running, but NOT actively processing meat feature information. However, it was communicating with the Man Machine Interface and the Intelligent Vision System modules.
- RUNNING - The Expert System was actively processing meat feature information.

The Background Process could only be in the STOPPED or RUNNING state.

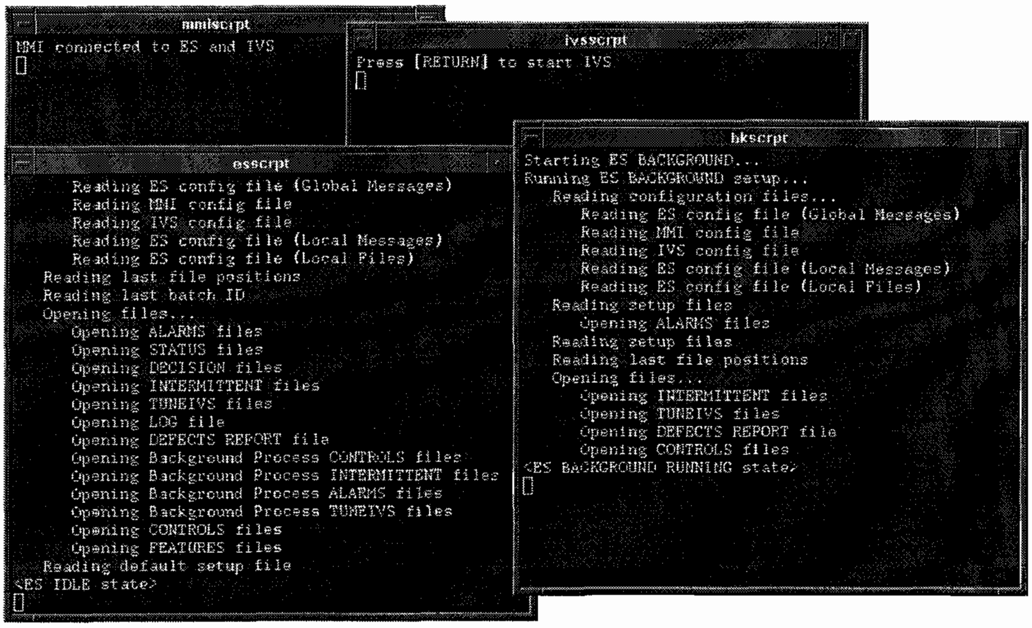
Once started, the Expert System Shell was controlled by messages received from the Man Machine Interface module. The Background Process, in turn was controlled by messages received from the Shell. The messages communicated between the ES Shell, IVS and MMI were:

Message	Description	Communication File Name	Associated Index File Name	Sent By
Status	Current Expert System state	status.dat	status.idx	ES Shell
Decisions	Grade of sample	decision.dat	decision.idx	ES Shell
Intermittent	Messages for the Operator when a trend was detected	intermit.dat	intermit.idx	ES Shell
Alarms/Errors	Expert System software errors	alarms.dat	alarms.idx	ES Shell
TuneIVS	Name of IVS parameter file.	tuneivs.dat	tuneivs.idx	ES Shell
Features	Meat feature information	features.dat	features.idx	IVS
Control	Control messages e.g. changing batch identifier, requests for reports, changing system states.	controls.dat	controls.idx	MMI

The messages communicated between the Shell and the Background Process were:

Message	Description	Communication File Name	Associated Index File Name	Sent by
Control	Control messages e.g. stopping background process, command to create a report.	esbkctl.dat	esbkctl.idx	ES Shell
Intermittent	Messages for the operator when trends are detected.	esbkint.dat	esbkint.idx	ES Background
Alarms/ Errors	Expert System internal software errors.	esbkalm.dat	esbkalm.idx	ES Background
TuneIVS	Name of IVS parameter file.	esbktun.dat	esbktun.idx	ES Background

To test the communication of the Expert System and how it integrated with the other modules, software was written to simulate the Man Machine Interface and Intelligent Vision System.



Simulation of Expert System communication.

The tests carried out and shown by the figure above, consisted of four windows, two of which were running the two Expert System modules, i.e. the window labelled "esscrpt" running the Shell, and "bkscrpt" running the Background Process. The other two windows, "mmiscrpt" and "ivsscrpt" were running simulations of the Man Machine Interface and Intelligent Vision System modules respectively.

Operator commands were specified by entering commands into the mmiscrpt window. Meat feature information stored in a file was read by the ivsscrpt simulator and communicated to the Expert System.

5.8 MAN MACHINE INTERFACE

The function of the Man Machine Interface (MMI) was to provide users (factory operators) with a WINDOWS based interface to the ALINSPEC system. The interface would be a smart terminal through which the users would:

- define the ***production parameters*** as:
  - stock parameters and codes
  - production strategies
  - inspection/classification strategies (quality profile)
- issue **START/ STOP COMMANDS**, allowing ALINSPEC to turn inspection ON and OFF
- ask for **Reports** (on a previous or current batch of samples), the information contained in the reports would be defined during the course of developing the ALINSPEC system.

The MMI was developed to provide the "end user" with the following information:

- the STATUS (stopped/idle/running) of the ALINSPEC modules
- real-time inspection results (grades)

- possible warnings/alarms messages (intermittent outputs) from the ES, e.g. when a particular trend was deduced
- test results
- summary and detailed reports on a batch of samples.

5.8.1 SOFTWARE ARCHITECTURE

The Man Machine Interface (MMI) was developed as a software module running on a DOS/Windows™ platform. The MMI was developed to run on a PC with the following hardware and software characteristics:

- Intel 486 DX2 66 MHz, 16 MBytes RAM
- EISA BUS and EISA expansion slots
- 500 MByte Hard Disk and 1.44 MByte floppy disk
- Super VGA 1280x1024 colour monitor
- RS-232 ports, 1 printer port and 1 mouse port
- Ethernet adapter board
- MS DOS™ 6.0 or later, MS Windows 3.1™ and PC-NFS™ for DOS™.

The MMI was developed using a windows application development tool called **LabWindows/CVI™** from NATIONAL INSTRUMENTS.

Communication to/from the MMI was implemented using fixed length records written to and read from NFS (Network File System) files mounted over an Ethernet TCP/IP network, with a different file for each type of message. As part of the overall ALINSPEC system start-up procedure, the IVS "exports" an NFS file system which the other modules "import". As a result, each module treats the file system as if it were local to it.

As part of each module's "start-up" procedure (including the MMI), it read its own communication ***Configuration File.*** The configuration file contained the information needed to open each communication file. This included the file name, location, and maximum size. The configuration file was set up as an ASCII (text) file to enable the system integrator (BULL) to easily reconfigure the communication network without modifying the software modules. The contents of the MMI configuration file are shown below:

```
# Configuration file for the MMI (Version 1 16/08/95)
#
# COMMON SECTIONS
# -------------------------------------------------------------------
# This section contains information needed to open or create files for
# the message types written by the Man Machine Interface
#
# The section has format 4 lines for each message type with the lines holding
#           * Name and location of index file
#           * Name and location of communication file
#           * Size of messages, in bytes
#           * Total number of message records allowed in a communication file
# -------------------------------------------------------------------
#
# [MMI STATUS FILE]
#
m:\mmi\config\status.idx
m:\mmi\config\status.dat
32
```

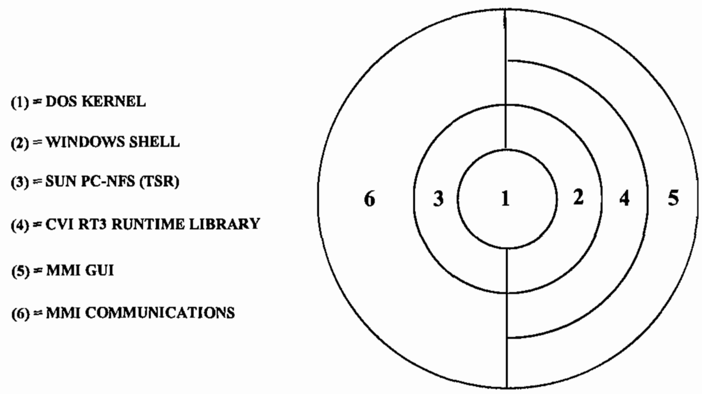
```
80
#
# [REQUESTS AND CONTROL Message Type (to the ES)]
#
m:\mmi\mmitoes\controls.idx
m:\mmi\mmitoes\controls.dat
28
80
#
# [FEATURE Message Type (to the ES)]
#
m: \mmi\mmitoes\features.idx
#m:\mmi\mmitoes\features.dat
48
500
#
# [REQUESTS AND CONTROL Message Type (to the IVS)]
#
m:\mmi\mmitoivs\controls.idx
m:\mmi\mmitoivs\controls.dat
28
80
```

The table below lists the names of the communication files used by the MMI in communicating with the ES and IVS.

FROM	TO	SUB DIRECTORY	FILENAME	MESSAGE TYPE
		C:\ALINSPEC\	ALINSPEC.EXE	MMI Main Program
MMI		M:\MMI\CONFIG\	CONFIG.DAT	MMI Configuration File
MMI		M:\MMI\CONFIG\	STATUS.DAT*	Status Report
MMI	ES	M:\MMI\MMITOES\	CONTROLS.DAT*	Commands from the MMI
			FEATURES.DAT*	Meat Features from the MMI
			SETUP.DAT	MMI set-up file for the ES
ES	MMI	M:\ES\ESTOMMI\	STATUS.DAT*	Status Report
			ALARMS.DAT*	Alarm and Warning Messages
			DECISION.DAT*	Per Sample Decision Messages
			INTERMIT.DAT*	Intermittent Output Messages
			REPORTS.DAT	Report Messages
MMI	IVS	M:\MMI\MMITOIVS\	CONTROLS.DAT*	Commands from the MMI
IVS	MMI	M:\IVS\IVSTOMMI\	STATUS.DAT*	Status Report
			ALARMS.DAT*	Alarm and Warning Messages
			FEATURES.DAT*	Sample Identifier

* This communication file had an "Index File" associated with it. The index file was located in the same directory as the communication file, and was the same name but with the extension .IDX.

Communications between the MMI and the other modules of ALINSPEC were managed in a totally transparent way by the SUN PC-NFS software package. This package was loaded as a "Terminate and Stay Resident (TSR)" program when the PC was switched on. The figure below shows the software environment in which the MMI module operated.



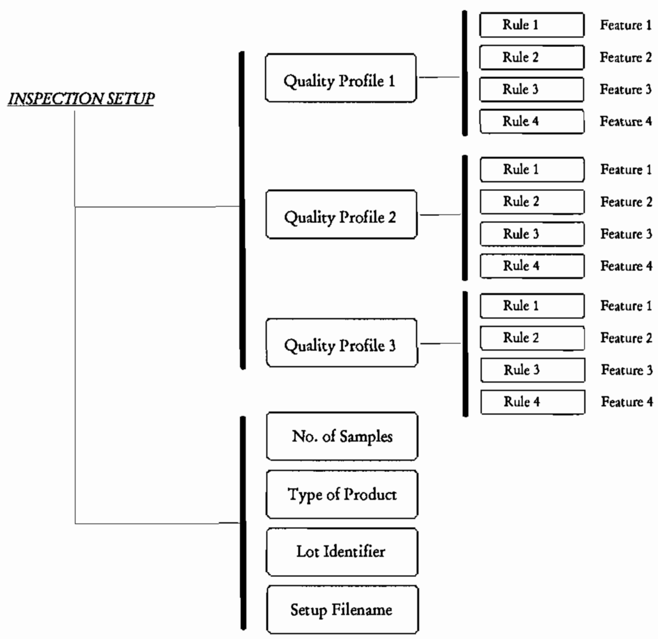
- The lowest level was the DOS operating system (1).
- sitting on top of the DOS kernel were the Windows 3.1 operating system (2) and the SUN PC-NFS TSR software (3)
- the CVI RT3 Runtime Library (4) from National Instrument operated within Windows 3.1
- the Graphical User Interface of the MMI module was managed by the CVI RT3 layer
- the communication module of the MMI was managed by the PC-NFS layer.

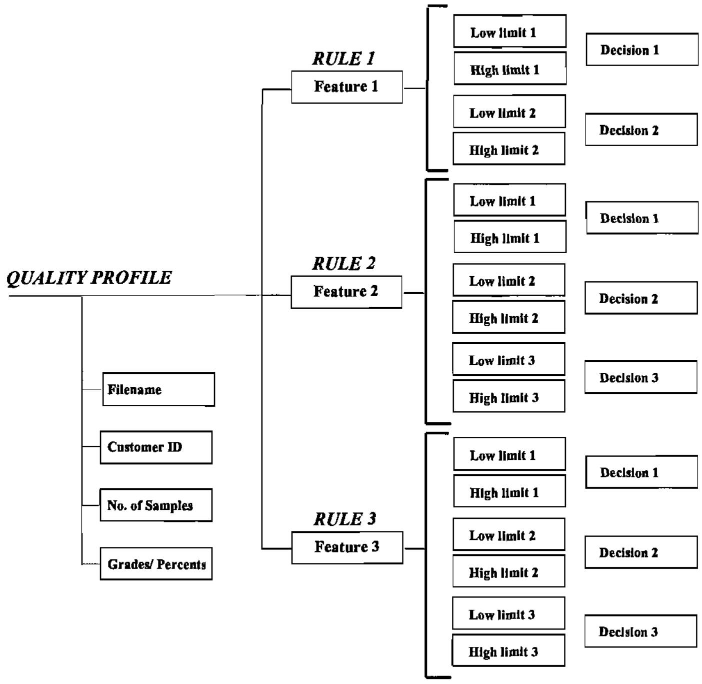
5.8.2 THE USER INTERFACE

The User Interface was developed using common features of Windows interfaces such as pop-up menus, menu bars, and windows. The interface was developed to enable the operator to perform tasks such as defining the inspection set-up, reading inspection results, reading or printing reports, and issuing commands for starting or stopping the inspection process.

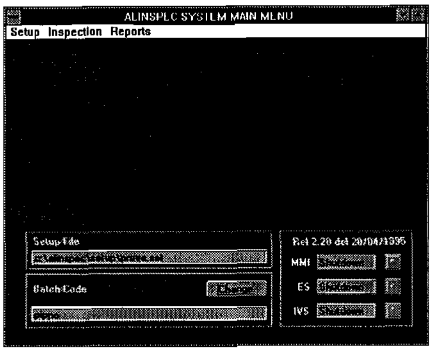
One of the important functions of the MMI module was to provide the operator with a simple means of specifying the ***"INSPECTION SET-UP"***. An inspection set-up (as shown by the next figure) consisted of **quality profiles** which defined the rules used by the Expert System in assessing the quality of meat samples. The interface was designed such that up to 5 quality profiles could be specified for an inspection, thus allowing the ALINSPEC system to:

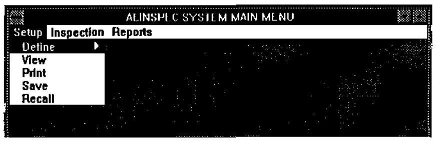
- assess the quality of a batch of samples arriving at the factory from a farm, this provides the factory with a consistent assessment of the quality of meat from their suppliers (i.e. the Farmers)
- assess quality and grade meat samples inspected according to the quality requirements of different customers (i.e. a quality profile represented the quality requirements of a particular customer)

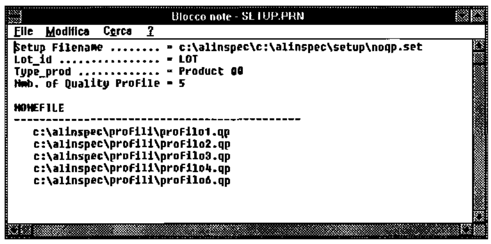


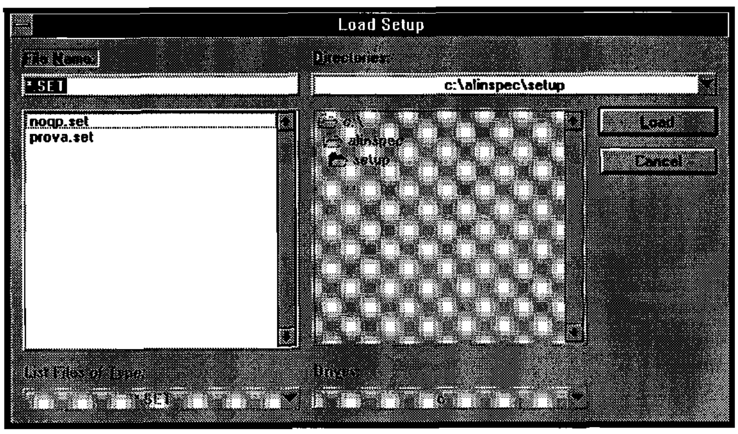


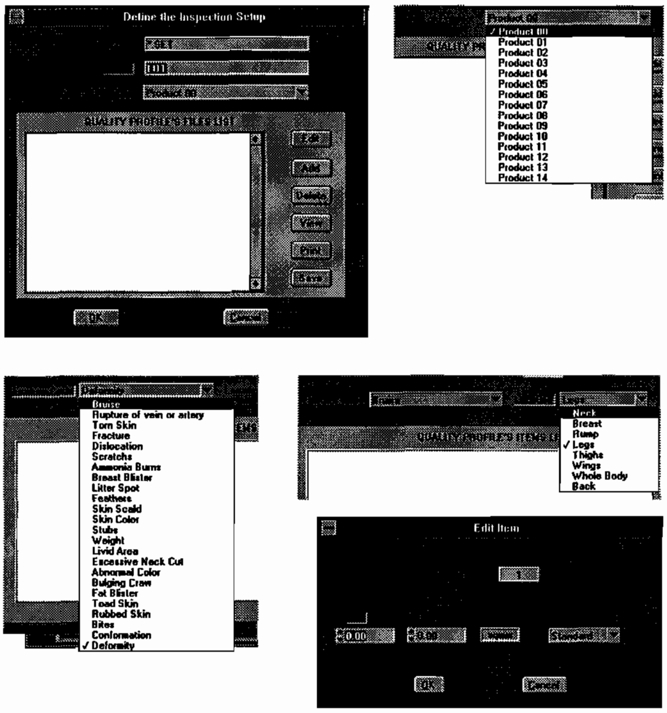
In addition, the interface provided menu options for the retrieval, storage, editing and printing of inspection set-ups and quality profiles as shown by the figures below.











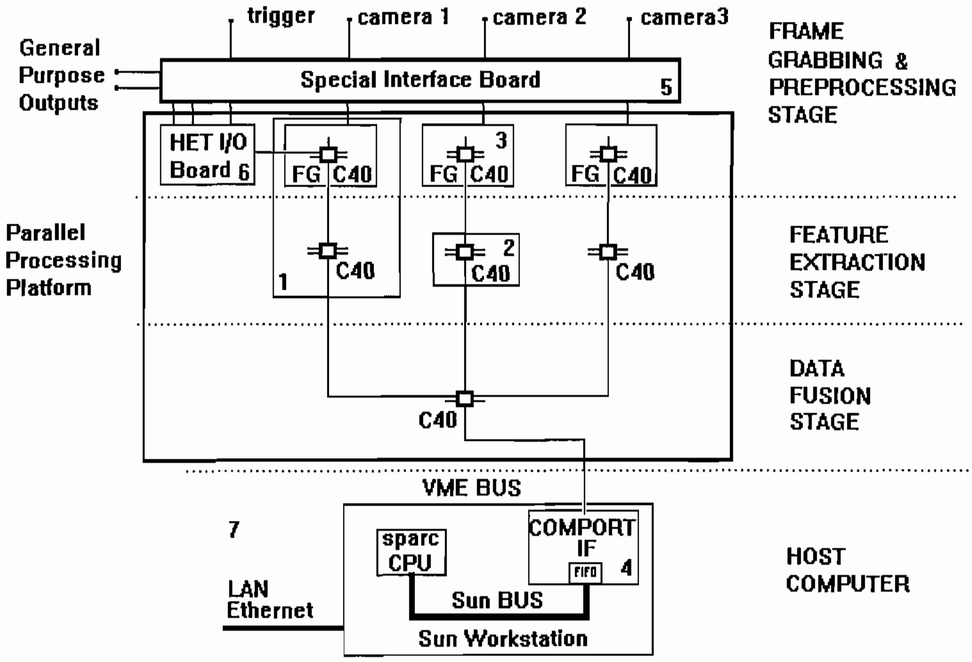
5.8.3 COMMUNICATIONS

The MMI module was designed such that its communication procedure was executed in an asynchronous manner (i.e. a separate program from the user interface and running in the background). The communication procedure was executed via an operating system software interrupt generated every 0.5 seconds.

However, in order to synchronise the communication of the MMI with the other modules, a time delay was introduced after the interrupt was generated. This delay was set up as a configurable parameter whose value depended on the hardware characteristics of the PC. For example, a delay of 0.8 s was required for a 66 MHz Intel 486 processor, and 0.3 s for a 100 MHz Pentium processor.

5.9 PARALLEL PROCESSING PLATFORM (PPP)

A Parallel Processing Platform (PPP) (shown below) was developed by BULL to carry out the image acquisition and processing functions of the Intelligent Vision System (IVS) at real-time production speeds.



All the modules of the Parallel Processing Platform (part numbers 2,3,4,5) were developed according to the TIM-40 specification defined by TEXAS INSTRUMENTS. This allowed hardware modules produced by different manufacturers to be plugged into a standard motherboard, such as the HEV40 (part number 1).

5.9.1 TIM-40 MOTHERBOARD

The HEV40 produced by HUNT ENGINEERING was a VME slave card that allowed up to 3 TIM-40 modules to be connected to it, thus providing an interface between the VME BUS and the TIM-40s.

- The HESB40 (part number 4) is a SBUS DVMA Master card which provided an interface between a SBUS based workstation (IVS SUN Host UNIX Computer) and a system of TIM-40s.

- A special VME Interface Board was designed to connect the sensors to the motherboard. The VME Interface Board was also used in controlling an opto-electronic sensor.

- The digital inputs/outputs of the IVS were provided by a custom built input/output (I/O) board produced by HUNT ENGINEERING.

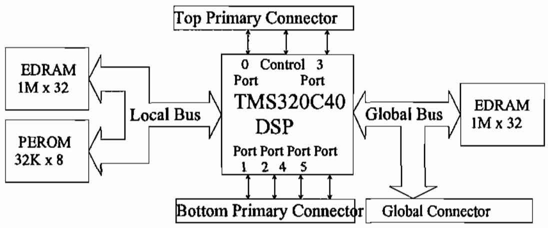
- The HET I/O (part number 6) was a TIM-40 without a processor. This was used as an interface to a TMS320C40 module via a communication port (COM port).

- The HET I/O ports were used in driving two relays. The ports were electrically isolated from the current sinks by two “optocouplers”, thus protecting the HET I/O device from damage by an external electrical surge.

The design of the PPP was based on the Texas TMS320C40 Digital Signal Processor (DSP), thus a Frame Grabber Board (part number 3) based on the same processor was used.

5.9.2 TIM-40 TRANSTECH MODULE

The Parallel Processing Platform made use of 4 commercial "transputer" modules (part number 2), each one was equipped with a TEXAS INSTRUMENTS TMS320C40 Digital Signal Processor as shown below.

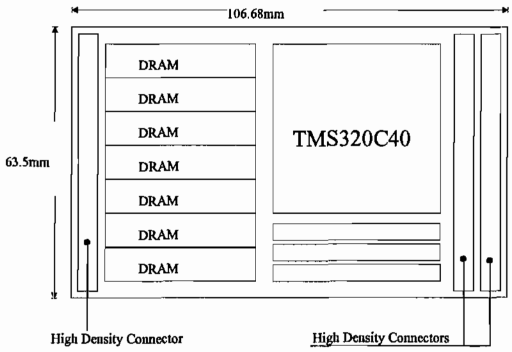


The TRANSTECH TDM411 TIM-40 module was selected for its price/performance ratio. The TMS320C40 is one of today's fastest processors. It offers very high performances in executing intensive computational ("number crunching") and communication tasks. It also allows:

- up to 275 MOPS
- up to 50 MFLOPS
- six 20 MBytes/sec communication links
- two 100 MBytes/sec memory buses

Tests were performed on the module to confirm the access times stated in the technical specification.

Although the TMS320C40 was particularly suitable for image processing applications and was used in the construction of the demonstrator, it consisted of unusual parts making it a very complex and expensive hardware module. In order to reduce the cost of the Parallel Processing Platform, BULL designed a proprietary hardware module based on the TMS320C40 (shown below). The module was designed according to the TIM-40 specifications defined by TEXAS INSTRUMENTS. The dimensions of the module were chosen such that the module area offered the best compromise between board size and ease of manufacture. The module was also designed such that it could be installed on a variety of standard motherboard expansion busses, e.g. the AT bus or VME bus of a PC.



The height chosen for the module made it possible to stack one or more modules vertically along the length of most motherboards, including that of a PC.

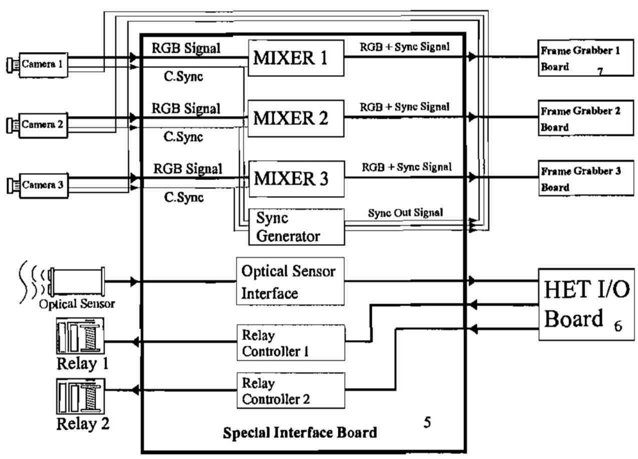
5.9.3 PARALLEL PLATFORM TO HOST COMPUTER CONNECTIONS

The HESB40 (part number 4) is a SBUS DVMA Master card which provided an interface between a SBUS based workstation (IVS SUN Host UNIX Computer) and a system of TIM-40s. It was designed by HUNT ENGINEERING and supplied to BULL by TRANSTECH as the TDMB420 module.

5.9.4 SPECIAL VME INTERFACE BOARD

A special VME Interface Board (shown below) was designed to connect the sensors to the TIM-40 motherboard. This front-end was needed to cater for the lack of a "sync" input on the C40 Frame Grabbers (standard cameras supply a composite sync as a separate signal from the RGB signal, whilst Frame Grabbers need at least one colour signal combined with a composite sync).

This interface circuit mixed the composite sync signal with the RGB signals. The signals from the RGB sensors (cameras) were collected on this board, and after mixing with the composite signal, were sent to the respective C40 Frame Grabber boards. The VME Interface Board also controlled an opto-electronic sensor and its associated relays. These are used in synchronising the image acquisition process with the movement of the meat sample. The trigger signal from the optical sensor determined the precise moment an image was acquired (i.e. when the sample was situated in front of a camera).

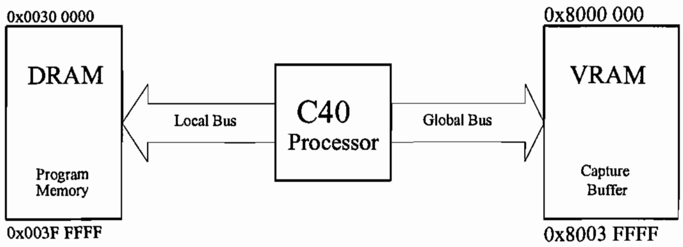


VME Interface Board

5.9.5 THE FRAME GRABBER

As the design of the Parallel Processor Platform was based on the TMS320C40 Digital Signal Processor, the Frame Grabber Board was fitted out with the same processor. The Frame Grabber selected was the TRANSTECH TIM-40CFG module.

The Frame Grabber Board consisted of 4 MBytes of page mode DRAM on the Local Bus and 1 MByte of VRAM on the Global Bus. The DRAM was used for software application program code and data, whilst the VRAM was used as the display buffer (i.e. storage of the acquired image). The following figure shows the layout of the memory of the TIM-40CFG Frame Grabber Board.



The Frame Grabber board was a size 2 TIM-40 plug-in module that provided frame grabbing for a C40 based system. It operated in two modes: 24 bit true colour and 8 bit monochrome modes. The 24 bit true colour mode was used in the Parallel Processing Platform, enabling the digitisation of a full colour image of up to 512x512 pixels.

A Frame Grabber program was written to test and configure the image acquisition process. The program was developed as a "diagnostic" tool, and not to be used by the ALINSPEC end-user. The program provided the following options via a main menu for testing the image acquisition process:

- **RESET** - restore the frame grabber to the same start-up configuration
- **CONFIGURE FRAME GRABBER** - display the configuration menu
- **TEST BED START/STOP** start or stop the Test Bed
- **LIGHT ON/OFF** - turn the lights on or off
- **FILL MEMORY** - set the memory area where the acquired image is stored to a specified 32 bit hexadecimal value, i.e. simulate the acquisition of an image.
- **SNAP** - acquire a single image
- **START/STOP AUTO ACQUISITION** - start or stop the automatic process of acquiring images, i.e. the frame grabber waits for a trigger signal from the optical sensor before capturing the image
- **WRITE REDUCED VIFF FILE** - write the image captured (stored in the "capture buffer"), in a "reduced" VIFF format to a file. The default "reduce" factor was 50%, but can be changed using the SET_RED REDUCING FACTOR option
- **WRITE 3 FILES VIFF R, G, B** - write the image in VIFF format to three files, one file for each colour plane (i.e. red, green and blue)
- **WRITE 1 FILE VIFF RGB** - write the image in VIFF format to one file
- **QUIT** - quit the frame grabber program.

The **CONFIGURE FRAME GRABBER** option activated a sub-menu listing the following options:

- **SET ACQ MODE** - set acquisition mode to Black-White (B/W) or Colour. Eight bit images were captured in B/W mode, and twenty-four bit images in colour mode. The default was the colour mode.
- **SET INTERLACED OR NOT INTERLACED MODE** - set the Frame Grabber in Interlaced or Not Interlaced mode. The Frame Grabber was set in interlaced mode by default.
- **SET SYNC SIGNAL ON R/G/B** - select the RGB channel to be used as input for the sync signal from the Frame Grabber. The green channel was used by default.
- **SET C40 OSCILLATOR** - select the method used by the digitiser in generatating the pixel clock. This could either be a 30 MHz Oscillator fitted as standard (Socket Clock) or the clock of the C40 processor. The default configuration was the Socket Clock (30 MHz)
- **SET DIVIDE RATIO** - select a factor, the Clock frequency above was divided by a factor to obtain the pixel frequency used by the frame grabber in digitising the images. The default divide ratio was 2, so the default pixel frequency was 15 MHz. However, the pixel frequency must not be greater then 20 MHz.
- **SET FIELD REVERSED MODE** - select the mode in which the fields of an interlaced image are grabbed, the fields may be rounded in a wrong way. The options were to either capture the fields in "Reverse" or "Normal" mode. The default mode was the Reverse mode.
- **SET REFERENCE VOLTAGE** - set the "Top Reference Voltage" for the three RGB channels. The Frame Grabber digitised analogue voltages signals between 0V and the value of the "Top Reference Voltage" in steps of 256. The maximum range was 0-1.2V. The default value for the "Top Reference Voltage" was 0.7V. It should be noted that PAL and NTSC video signal standards allow for a maximum voltage of 0.7V.
- **SET REDUCING FACTOR** - select the ratio by which a VIFF formatted image was reduced (see WRITE REDUCED VIFF FILE option above). The "Reducing Factor" could either be 75% or 50%, the default was 50%.
- **VIEW CONFIGURATION** - display the current configuration on a computer screen
- **VIEW REGISTER -** display the contents of the Bt254, Bt261and IIOF registers for debugging purposes
- **ENABLE/DISABLE TRACE** - enable or disable logging of "trace" information in a file. This was used for debugging purposes.
- **ENABLE/DISABLE PRINT** - enable or disable the display of "trace" information on a computer screen. This was used for debugging purposes.
- **EXIT** - return to main menu option

6. INTEGRATION

The hardware and software tools that make up the ALINSPEC system were designed and developed by different partners. The methods used and rules governing development were agreed and adhered to by all the partners. This was carried out by adopting standards such as the UNIX and DOS operating systems, Khoros image processing development tool, ANSI Standard C programming language, LabWindows© Windows development tool and Network File System (NFS) for file sharing. BULL as system integrator was responsible for the integration of the hardware and software modules developed by each partner, and necessary for the construction of an ALINSPEC demonstrator.

The software modules which made up the ALINSPEC demonstrator were:

- IVS software module developed by DIE, OG and CEO
- IVS communication module developed by UVIGO.
- ES software module developed by CCFRA
- MMI software module developed by OG

The image processing algorithms developed by STRATH for detecting skin texture features such as "Toad Skin" in chickens were not included in the IVS module because they were developed using the KB-Vision image processing development tool. The KB-Vision environment could not be ported to the parallel processing platform given the resources available.

In order to fulfil one of the objectives of the project, i.e. develop a system to run at real-time production speeds, the image processing algorithms (IVS module) were ported from the development environment (single processor platform) to a custom built parallel processing platform (PPP).

6.1 PPP Software Development Environment

The parallel processor platform (hardware) needed a software development environment to allow for the compilation of the image processing algorithms (source code). To achieve this, the following tasks were carried out by BULL:

- a software tool was developed for compiling Khoros workspaces (image processing algorithms) to run on different UNIX workstations
- installation and validation of the parallel C programming language compiler (3LC), and the installation of the debugger for the 3LC
- installation of the Helios operating system (PPP UNIX operating system), and configuration of Helios according to the agreed standards.
- the specification and coding of a "Link" file system, this acted as an interface between software applications and the communication management functions provided with the 3LC package.

Installing a C compiler on the PPP enabled software developed in a single processor environment to be ported easily to a parallel processor one, i.e. by compiling the source code using the 3LC compiler.

6.2 Porting IVS Module To PPP

The IVS (Intelligent Vision System) module of the ALINSPEC system consisted of two sub-modules:

1. The chicken image processing sub-module was developed by DIE and CEO using Khoros. This "development" version consisted of a UNIX shell script and a collection of Khoros and C programs. The script executed the Khoros and C subroutines or functions.
2. The administration and communication sub-module managed the flow of data between the IVS image processing sub-module and ES and MMI main modules. The sub-module consisted of C programs only.

The architecture chosen for the IVS demonstrator was a SUN UNIX Host + PPP Workstation. The image processing would be executed on one of the C40s of the PPP, and the communications module on the SUN UNIX workstation. In order to port the image processing module to the PPP, the following tasks were carried out:

- analysis of the UNIX shell scripts from DIE and CEO
- development of a C program to replace the shell scripts
- pre-integration of the IVS image processing with IVS communications modules
- porting of the IVS image processing module to PPP, this included porting Khoros library routines.
- final integration of the IVS image processing module running on the PPP with the IVS communications module running on SUN Host workstation.

6.2.1 ANALYSIS OF IVS SOFTWARE

The image processing UNIX shell script analysed chicken images by extracting the silhouette, segmenting chicken into its sub-parts (i.e. body, legs and wings), and for each sub-part, performed a series of morphological operations in order to identify meat features present.

The scripts executed standard Khoros library routines as well as Khoros procedures developed by DIE and CEO. The routines/procedures were UNIX executables, and information was communicated from one procedure to another via temporary files. The output from executing the shell script was a file containing the meat features detected.

Following the analysis of the IVS architecture, and a better understanding of the subroutines and their functions, most of the processing time was spent in carrying out morphological operations, and the loading of images. As a result, the architecture of the IVS image processing module was rationalised, especially with the loading of "cooked" images and variables/parameters.

6.2.2 CONVERSION OF IVS SOURCE TO C SOURCE CODE

The conversion of the IVS scripts into a C program involved the reorganisation of all the subroutines and procedures, these included executables files and library subroutines. The executables (main programs) were converted into C function calls and data passed as arguments of a function, rather than using temporary files. The scripts were integrated into one single C main program calling C functions. This was carried out by converting the following Khoros and DIE subroutines into C functions:

- standard Khoros image processing functions

- special purpose DIE chicken image processing functions
- Khoros I/O and VIFF support functions
- public domain matrix algebra functions
- public domain FORTRAN support functions

The functions were isolated and grouped into the following libraries:

- the DIE/CEO chicken image processing library (DIE.LIB).
- the Khoros image processing library (VIPL.LIB).
- the Khoros I/O library (VUTILS.LIB).
- the matrix and complex number public domain library (LINPACK.LIB).
- the FORTRAN to C support library (F77.LIB).

The public domain LINPACK library was converted from FORTRAN77 to C using a package called **F2C**. The Khoros I/O library was built by only including the necessary parts from the original procedures. UNIX-specific code such as file locking, shared memory support and process synchronisation procedures were excluded.

However, the following Khoros functions were not converted:

- *vconvert* - this routine was not ported because of the unique data type (BYTE) of images processed by the IVS
- *vstats, varviff* - a completely new and more efficient version of these routines were re-written.

The Khoros and DIE procedures made use of UNIX system functions for allocating memory dynamically. Memory was allocated but not freed upon completion of a task. The development version of the IVS module relied on the UNIX operating system to release all system resources when a process (i.e. executable) exited or was terminated. By converting to a single C program, the management of memory dynamically would have to be performed by the program itself. The system functions for allocating memory could be called up to 70,000 times in one image processing operation.

The code for managing memory was excluded from the library functions converted. Instead, a general purpose memory allocation and de-allocation function was written. The function also kept track of what memory was allocated. In some cases, software "bugs" in the original Khoros source code were detected, and these were usually caused by the allocation of memory not previously freed.

Many of the routines in the development version loaded the same "cooked" image more than once, thus increasing the execution time. This was changed such that the reading of parameters and loading of cooked images were carried out once.

The routines in the development version also exchanged data (images) via temporary files. This was changed to the exchange of date between functions via memory so as to increase the speed of the process. A check for the overwriting of the same memory location was added.

In order to read the IVS configuration file, write the features detected to a file, and perform all non standard Khoros I/O (all non VIFF files) operations, functions were added to the DIE library. Some of these functions were written by converting the source code of utilities executed from the IVS UNIX shell scripts.

Efforts were made to keep the new source code as simple as possible. Once completed, the IVS source code was compiled to run on a SUN UNIX Workstation, tested to ensure the results were the same as that produced by the development version.

The IVS C program ran at about the same speed as the IVS script, i.e. about **90 seconds** for the image of a chicken sample.

Prior to porting the new C program to run on the PPP, the module was integrated with the communications module (delivered as a C program by UVIGO), and tested on a single processor platform. The communication between the IVS and the ES and MMI was also tested.

6.2.3 PORTING IVS C SOURCE CODE TO PPP

The new C program carried out the following functions:

- read the configuration file
- read the "cooked" images (either specified in the configuration file declared as constant)
- loaded a chicken image
- evaluated the chicken image
- wrote the features recognised to a file

The libraries of the C program were compiled using the 3LC compiler on the C40 (PPP) processor. In order to optimise the size of the IVS, the F77 library was modified so as to eliminate all I/O functions. The floating point exceptions were also modified so the process was not terminated when an error occurred.

However, due to the hardware configuration of the C40 processor (each C40 is equipped with 8 MBytes of memory divided into two equal parts, one part is accessed via the global bus and the other via the local bus), there was insufficient memory for loading both the image to be processed (3.5 MBytes), and the "cooked" images and other parameters.

To overcome this problem, the resolution of the images acquired for development were halved leading to a 2/3 reduction in the size of each image. This was accomplished by modifying several DIE Khoros procedures, and rewriting the Khoros function *vcolorth*.

This coupled with the TEXAS INSTRUMENT proprietary floating point format used by the C40 (the DIE/CEO development module used the IEEE format) led to differences in the size computed for the features recognised by the PPP IVS module. Shown below is a comparison between the results obtained from the PPP IVS (demonstrator) and the DIE/CEO IVS (development).

CHK054										
	BULL C Program					DIE / CEO Script				
Feature N	**Feature ID**	**Colour ID**	**Size/ value1**	**Size/ value2**	**Location**	**Feature ID**	**Colour ID**	**Size/ value1**	**Size/ value2**	**Location**
1	23	0	99504	0	G	23	0	98352	0	G
2	38	0	37168	1374	FB	38	0	36560	1391	FB
3	38	0	8832	2397	FRW	38	0	8432	2338	FRW
4	38	0	9232	1931	FLW	38	0	9296	2010	FLW
5	38	0	13952	2315	FRL	38	0	13568	2348	FRL
6	38	0	13616	2369	FLL	38	0	13616	2402	FLL
7	9	0	2004	0	FBBCC	9	0	1897	0	FBBCC
8	1	7	296	0	FWLM	1	7	244	0	FWLM

In most cases, the PPP IVS program detected the same features as the development version, but the value of the size reported differed. In cases where the size reported by the development version was very small, the feature was not detected by the PPP version. The differences in the size reported were caused by the following:

- floating point approximation
- difference between IEEE and TEXAS INSTRUMENTS floating point formats
- reduction in the resolution of the input image

Many Khoros routines were optimised so as to improve the performance of the PPP IVS, especially in the image scanning loops. Since the analysis of the image processing chain revealed that the processing time was spent in morphological operations, the "cooked" images and parameters were loaded into the C40's local memory so as to reduce the time. The image to be processed was loaded into the local memory.

As a result, the C40 version of the IVS processed each chicken image in about **21 seconds**.

This test was carried out using a TRANSTECH C40 with a clock speed of 40 MHz (C40s can run at 50 MHz). A detailed design of a 50 MHz TIM 40 module was produced by BULL and described in the project handbook document *Hardware draft project for a proprietary parallel processing platform*. A 50 MHz TIM 40 module would improve the performance of the PPP IVS module.

Further increases in the performance of the PPP IVS module could be obtained by increasing the performance of three Khoros classifying procedures *vclas*, *vlegclas* and *vwinclas*. These routines scanned the input image at least 50 times when the image was being processed.

6.2.4 INTEGRATION OF PPP IVS WITH COMMUNICATIONS MODULE

The PPP IVS module was integrated with the IVS communication module running on the SUN Host UNIX Workstation.

A simple communication interface was built between the two IVS modules. This consisted of two file based communication channels, one channel for commands from the Communications IVS to the PPP IVS, and one for meat features from the PPP IVS to the Communications IVS. Each channel consisted of a file for storing the data to be communicated and a "lock" file for signalling that data written to a channel should be read.

6.3 INTEGRATION OF IVS, ES AND MMI

The IVS module was integrated with the other main modules, i.e. the ES and the MMI.

The ES C source code produced by CCFRA was successfully compiled to run on the SUN UNIX Host Workstation. The Workstation was connected to a Personal Computer running the MMI software. The two computers were connected by a Local Area Network (LAN), and the NFS shared file system was configured to enable communication between the modules.

7. DEMONSTRATOR

A chicken ALINSPEC system was built to demonstrate the techniques and tools developed for the real-time inspection and assessment of the quality of chickens.

The chicken demonstrator was constructed using the test bed built at the COREN/FRIGOLOURO chicken factory. The test bed consisted of a section of the production line. The test bed was initially used for the image acquisition phase of the project at the COREN/FRIGOLOURO factory. The test bed was then dissembled and shipped to the BULL laboratories at Milan where it was reassembled and used in demonstrating the ALINSPEC system.

7.1 OBJECTIVES

The objective of the ALINSPEC Project was to develop a technology, and design an architecture to allow an effective, consistent 100%, real-time examination of subjectively assessed food products such as chickens, and Spanish dried cured ham. The objective was not to develop an industrial prototype, but to develop and investigate a wide range of techniques, and integrate them into demonstrators "tool boxes" from which appropriate methods could be selected according to the performance and requirements of particular applications.

The chicken demonstrator was built to demonstrate:

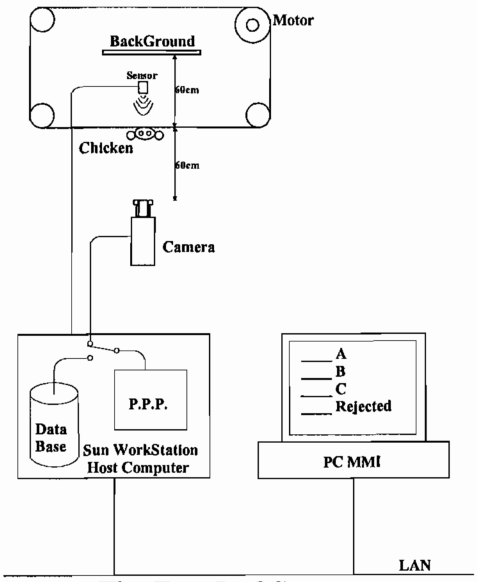
- the acquisition of images of chicken samples from a production lime running at a speed of 0.6 m/s
- the identification of chicken meat features and grading of the sample accordingly

As reported earlier, the industrial situation was very different for chickens and pigs. For this reason, the objectives set at the beginning of the project were modified to include the development of "tools" which would be beneficial to those automating the pig butchery process. As a result of reallocating resources to meet these objectives, an on-line real-time pig leg quality assessment demonstrator was not built. The results obtained from developing the pig leg tools have been reported earlier, these included image processing algorithms for detecting the most important meat features required in assessing the suitability of pig legs for Spanish dry cured ham. Other tools developed and tested, but not integrated to form a demonstrator were:

- identification of sub-surface features from infrared images of chickens
- identification of bones from x-ray images of pigs legs'
- measurement of fat thickness on the cut-surface of a pig leg.

7.2 CHICKEN IMAGE ACQUISITION DEMONSTRATOR

The demonstrator showed how images of chicken suitable for processing by the IVS were acquired at real-time production speeds using the test bed developed by COREN/FRIGOLOURO and reassembled at BULL's laboratories.



The Test Bed Structure.

The demonstrator as shown by the figure above consisted of the following:

1. the test bed with a rotating conveyor from which chicken samples were hung
2. a blue background placed behind the chicken so the image processing algorithms could extract the chicken shape from the background
3. a RGB TV camera for acquiring images of the sample as it passed in front of the blue background
4. a UNIX SUN Host Workstation and a PC.
5. the UNIX workstation was set up to either process images acquired directly from the camera via the Frame Grabber, or images loaded from the chicken image library (used in developing the image processing algorithms)
6. the MMI ran on a PC and allowed the operator to define an "inspection set-up", as well as displaying the grades assigned to each sample by ALINSPEC.

When a chicken sample crossed the sensor's field of view, the sensor triggered the acquisition process as the sample passed in front of the camera.

As the demonstration was carried out at BULL's laboratories, real chickens could not be imaged for sanitary reasons. As a result, BULL used fictional chickens made from foam rubber. The fictional chickens were made by cutting a large block of foam in the shape of a chicken, and grinding the surface to obtain a smooth and regular surface. Each fictional chicken was then plunged in a tank full of liquid latex. When dried, a thin film covered the chicken and this was painted in order to obtain a realistic chicken skin surface.

The relatively high speed of the production line (0.6 m/s) resulted in "blurring" of the images initially acquired. This was caused by the odd and even fields of an interlaced image, the information present in these fields do not represent the same image. The shift in the image was proportional to the production line speed.

The solution to this problem was the acquisition of only one of the two image frames (fields), leading to a reduction in the resolution of the image acquired, but with sufficient pixels (image information) for the statistical operations performed by the image processing module.

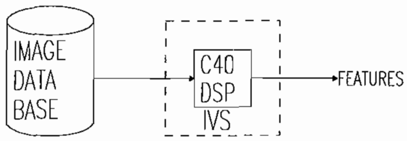
The table below shows the results obtained from timing the image acquisition process. This was obtained by timing the execution of the image acquisition software running on a the C40 based Frame Grabber board (Frame Grabber Program described in section 5.9.5).

Test /step	Acquisition process	Time
1	Automatic acquisition and display of a reduced image (256x256 Pixels)	
1.1	Snap image	37.5 ms
1.2	Copy the frame buffer and reduce the image size	110 ms
1.3	Write the reduced image on the disk	250 ms
2	Acquisition of a single image and writing of three (R,G,B) VIFF files:	
2.1	Snap image	37.5 ms
	the following two steps were repeated three times:-	
2.1	Frame buffer copy and image splitting	100 ms
2.2	Write one colour plane	1500 ms
3	Acquisition of a single image and writing of one VIFF file with 3 colour planes:	
3.1	Snap image	37.5 ms
3.2	Copy the frame buffer and reduce the image size	100 ms
3.3	Write one colour plane	3000 ms

The images acquired from the Frame Grabber were saved on the disk of the UNIX workstation. Further reductions in the acquisition could be obtained by transferring the data between the C40 of the Frame Grabber and the PPP directly. This could be carried by using the communications port of the C40 DSP (throughput of 9 MBytes/sec). The time could also be reduced by not displaying the image acquired on a computer screen.

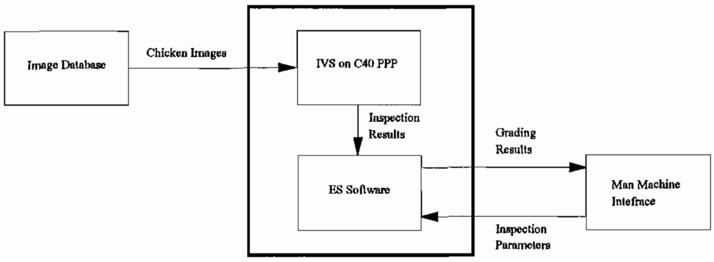
7.3 CHICKEN IMAGE PROCESSING AND GRADING DEMONSTRATOR

The demonstration showed how chicken images were processed by the PPP IVS. The features identified were sent to the ES, and the grade assigned by the ES was then relayed to the MMI. The demonstration was carried out using images of real chickens (not fictional) stored in the image library. The images of white or yellow chickens were selected randomly and processed by the IVS running on one of the C40s of the PPP as shown below.



This demonstrator consisted of the following (as shown by figure below):

- the PPP IVS on a motherboard plugged into a standard UNIX Workstation
- the Communications IVS running on the same UNIX Workstation
- the ES running on the same UNIX Workstation
- the MMI running on a PC and connected to the UNIX workstation by a Local Area Network (LAN) and using the TCP/IP protocol.



The demonstrator was expected to perform the following on images of white chickens (front/breast view only):

- analyse the shape of each chicken
- identify meat features if present and compute the size
- grade each sample (when the grades are compared with the corresponding human grades, a probability of error P_e of no more than than 20% should be obtained)
- display the grades on the screen of the MMI

The demonstrator was configured to identify the following meat features on white or yellow skinned chickens:

- CF01: BRUISES
- CF04: FRACTURES
- CF08: SCRATCHES
- CF13: AMMONIA BURNS
- CF34: LITTER SPOT
- CF15: SKIN COLOUR
- CF23: WEIGHT (the weight of chickens were estimated when analysing their shape)
- CF25: LIVID AREAS
- CF28: ABNORMAL COLOUR
- CF35: RUBBED SKIN
- CF38: CONFORMATION
- CF41: DEFORMITY
- CF42: THINNESS

The demonstrator was tested using two sets of images. One set contained images of white chickens only and the other yellow. In most cases, the features identified were the same as the ones obtained during the integration phase. By comparing the features identified with that of development version of the IVS, the size of the features reported differed by an average of 3%. As explained before, this was caused by:

- floating point approximation
- difference between IEEE and TEXAS INSTRUMENTS floating point formats
- reduction in the resolution of the input image

When the TEXAS INSTRUMENTS C40 proprietary floating point values were converted to the IEEE format, some of the significant decimals were lost. The cumulative rounding-off error increased with the amount of floating point operations resulting in a different final value for the size of features identified.

However, the most significant cause of the difference in the values was the reduction in the resolution of the input images used for the demonstrator. This would ultimately lead to a reduction in the accuracy of the size evaluated for each feature. This was unavoidable and as such, the choice of a suitable ALINSPEC system would be a trade-off between accuracy and processing speed. For faster ALINSPEC systems, the small meat features would not be identified. This was considered as acceptable as very small features such as skin imperfections or scratches or spots have very little influence on the overall quality (grade) of a sample.

In some cases, features identified by the demonstrator were classified incorrectly. For example, the table below shows the features identified by the demonstrator for sample CHK091 (columns on the left), and that identified by the development version (columns on the right). The seventh feature was classified as type 35 (Rubbed Skin) by the demonstrator, instead of type 9 (Breast Blister) as classified by the development version. This was also caused by the reduced image resolution, the colour of a single pixel was evaluated as an average value of the nearest four pixels. By reducing the number of pixels (reduced resolution), the resulting colour may be a little different from the original resulting in the wrong classification for the feature.

CHK091										
	BULL C Program					DIE / CEO Script				
Feature N	Feature ID	Colour ID	Size/ value1	Size/ value2	Location	Feature ID	Colour ID	Size/ value1	Size/ value2	Location
1	23	0	76944	0	G	23	0	75696	0	G
2	38	0	25360	1462	FB	38	0	24400	1472	FB
3	38	0	9424	1559	FRW	38	0	9328	1617	FRW
4	38	0	10496	1720	FLW	38	0	10240	1756	FLW
5	38	0	8752	4963	FRL	38	0	8784	5128	FRL
6	38	0	9872	4299	FLL	38	0	9840	4364	FLL
7	35	0	2380	0	FBBCC	9	0	3382	0	FBBCC
8	1	7	756	0	FRL	1	7	884	0	FRL
9	1	7	968	0	FLL	1	7	1192	0	FLL

As a result, the grading accuracy of the demonstrator was less than that obtained from the development version of the ES (i.e. 70% compared with the 82% obtained using the features identified by the IVS development version). No changes were made to the reasoning structure of the ES developed by CCFRA. The difference in accuracy was caused by the difference between the PPP IVS output and the DIE/CEO IVS output (development version), which as mentioned above was due to the reduction in the resolution of the input image. Increasing the grading accuracy would require an increase in the accuracy of the PPP IVS.

An average chicken inspection time of **21 seconds** was obtained by dividing the total processing time by the number of samples inspected.

Since the input image for the demonstrator was a quarter of the size of the original one and the C40 and the SPARC processors operated on the same clock frequency, the result obtained indicated that the performance of the C40 processor was comparable to that of the SUN SPARC processor when running similar C-compiled program code. Only one C40 was used for executing the image processing software.

The slower than expected inspection time could be improved on by further optimisation of the PPP IVS source code for speed. The resources for porting the IVS to the PPP were stretched because of the unforeseen memory allocation problems. As a result, BULL only carried out "macro-optimisations" in order to simplify the porting process and write "easy to understand" code.

Nonetheless, the architecture of the demonstrator was designed such that future optimisations would be possible so as to achieve the necessary speeds. The optimisations could include the following:

- Image processing optimisations:- reduce the number of image scanning loops (some Khoros routines have more than 50), reduce the number of operations in the inner loops (especially the multiply operation), enhance image pixel addressing, and enhance the most frequently used functions and the heaviest "number crunching" algorithms.
- C40 specific optimisations:- static allocation of the small kernel images in the C40's internal memory, smart memory allocation of temporary images (optimising the usage of the C40 local and global busses), association of the multi-band images with their masks stored in the same memory space (using the fourth byte of a C40 word), optimise the register and stack usage, and replace the heaviest "number crunching" algorithms with functions written in the C40 assembly language.
- C40 parallel processing optimisations: allocate all the processing tasks to four C40 processors (i.e. background extraction and anatomical parts separation on the first C40, breast evaluation on the second, left wing and leg on the third and right wing and leg on the fourth)

By carrying out the first two optimisations, the performance of ALINSPEC could be increased by a factor of ten or more. However, by using four C40 digital signal processors instead of the single one used in the demonstrator, the real-time target of examining one chicken in half a second would be achieved.

8. CONCLUSIONS

Typical machine vision systems measure properties of an object such as colour, dimensions, shape, and surface texture. Where the objects are rigid and well defined, as in most manufactured products, deducing quality was easy; the measurements taken were simply compared with pre-determined "perfect" reference values. For natural objects, such as meat carcasses, it is more difficult to reproduce the judgement of a human expert.

In the absence of a "perfect" set of reference values, simple measurements were not enough. Multiple defects and features with ill-defined sizes, shapes and colours must be recognised against a background which is itself heterogeneous. The absence, or presence, and nature of these features must be balanced against subjective requirements to deduce quality in a manner typical of humans.

At the end of the project, a collection of "tools" were developed for assessing the quality of natural products, in particular chicken carcasses and pig legs. A flexible, adaptable, high speed, prototype ("demonstrator") system was built to examine chicken carcasses in a manner similar to humans.

8.1 TECHNICAL ACHIEVEMENTS

The "tools" developed were for identifying visible and non-visible chicken and pig leg meat features and assessing quality. At the end of the project, the activities and investigations carried out indicated that:

- the technology (sensors, computer hardware and software) is available to allow 100% automatic grading of chickens and pig legs at real-time production speeds
- the required image processing and classification algorithms (analysing RGB, infrared and x-ray images) have been developed to identify surface and sub-surface meat features, defects, and other characteristics necessary for assessing quality.
- an intelligent flexible reasoning based method for assessing quality has been developed
- communication and integration of ALINSPEC with other factory computer systems is possible
- the Parallel Processing Platform developed for the ALINSPEC project represents a considerable technical resource for BULL, it will be used in the development of other real-time computer systems
- transfer of the technology to other application areas is possible.

A constant objective during the project was to find ways in which cheaper systems could be used to deliver the performance required by the industry. As a result, a strategy for down-sizing the ALINSPEC demonstrator has been identified. A basic (single RGB camera) industrial version can be produced at a cost of about 25,000 ECU (£21,000 - £22,000).

The modular architecture developed would also allow a system to be easily configured to suit the different industrial requirements, e.g. the additional use of non-visible sensors such as infrared and x-ray. Although cheap "off the shelf" infrared and x-ray sensors were not available, the cost of including these sensors was reduced by designing alternative sensors which made use of the components from existing "off the shelf" sensors.

The concept used in assessing quality of the sample would allow the meat industry to maximise their profits by enabling them to assess quality according to the specific needs of individual customers. It would also provide a quick and consistent evaluation of the quality of the incoming raw material so as to provide a fair and justifiable method of paying the supplier.

The original objective of building the demonstrator to run at real-time speeds at the COREN/FRIGOLOURO factories was not achieved. On reflection, this objective was perhaps a little bit too optimistic and ambitious given the consortium's knowledge of the poultry and pig meat production processes at the beginning of the Project. The problems encountered as a result of the consortium's inexperience resulted in the partial optimisation and parallelisation of the image processing algorithms running on the PPP developed by BULL. This ultimately resulted in a demonstrator running at a much lower speed than originally anticipated.

However, the work carried out in porting the algorithms enabled BULL to identify a cheap parallel processing platform based on a TMS320C40 DSP, and the most suitable architecture for achieving the required real time performance. This was made possible by the modular "expert" based structure of the IVS module, and by using multiple C40s for feature identification (i.e. one "expert" per C40), the real-time speeds would be easily achieved.

The development of the image processing algorithms were carried out by successfully integrating traditional image processing techniques with artificial intelligence based techniques. These techniques have also been applied in other areas such as in Military and Defence applications.

The computer based survey carried out by CCFRA (over 400 references), the bibliographic research by CEO to identify any patents related to the ALINSPEC system, and the survey of French producers of computer vision based equipment by BULL, confirmed that the multi-sensory capability, fusion of multiple data, intelligent recognition of features and the reasoning method of ALINSPEC were unique.

Although the consortium was aware of similar systems previously developed, there was no evidence to suggest that these systems were commercially available:

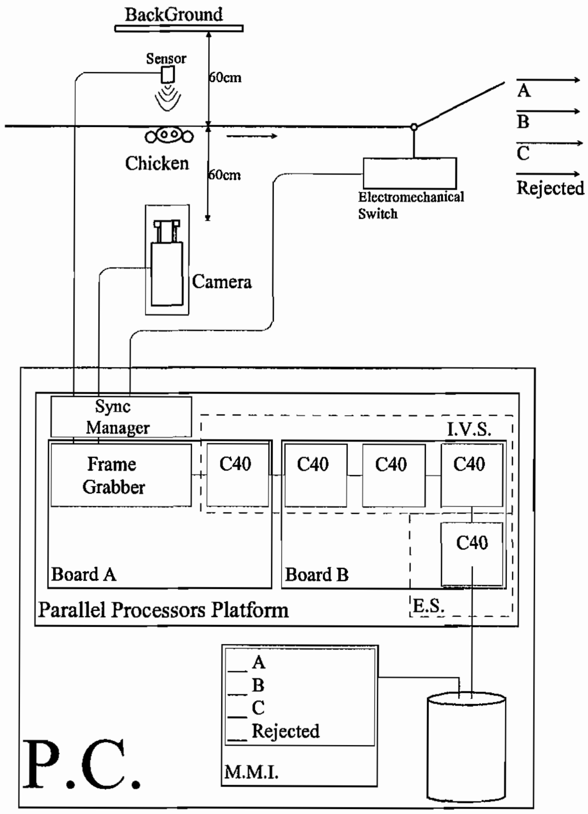
- the AQS system from STORK for on-line chicken inspection
- a (prototype) system from GTRI (Georgia Tech Research Institute) for on-line chicken inspection
- the Automatic Meat Grading System, from AUTOMATIX INTERNATIONAL UK Ltd for on-line pig meat inspection.

A new generation of intelligent inspection systems such as ALINSPEC would contribute towards:

- the development of European Standards for the Alimentary Industry
- improving the quality of meat products
- providing "added-value" to existing products

8.2 FUTURE IMPROVEMENTS

The architecture of the ALINSPEC system was developed using standard commercially available hardware components where possible so as to avoid incompatibility problems and expensive hardware/software modifications. However, this resulted in an expensive and slow system. To improve the cost-effectiveness of the ALINSPEC system, BULL have defined possible optimisation strategies. One such strategy would be to have ES and IVS modules running on a PPP plugged into a PC (as shown below), rather than the much more expensive UNIX Workstation.



A Future Improvement.

Further cost reductions could be achieved by using a set of proprietary C40 modules developed by BULL rather than the commercial TIM-40 modules.

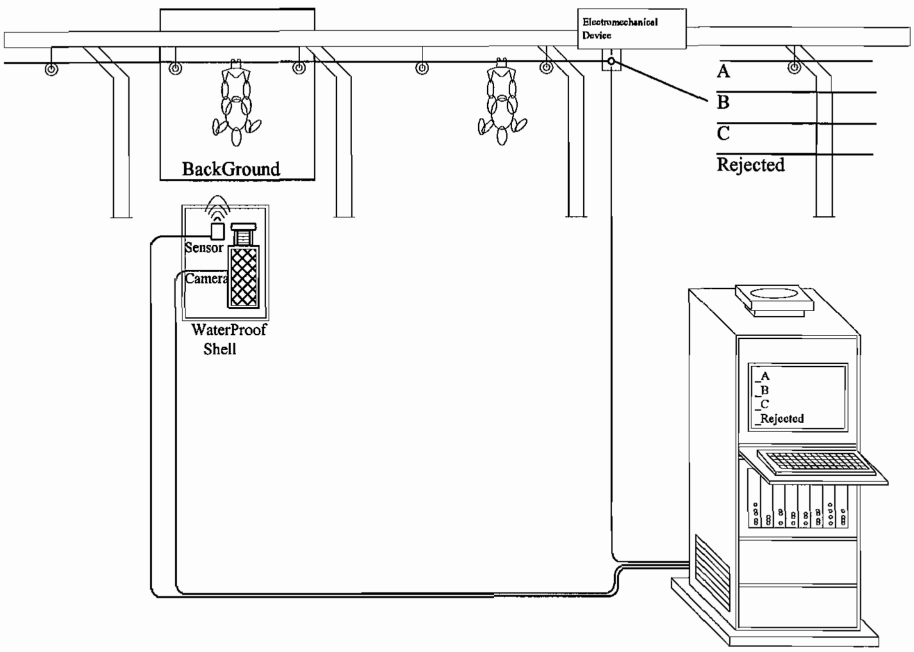
The performance of the image processing module could also be improved by re-writing the Khoros library functions, with the most frequently used functions written in the C40 assembler language. This would allow the full exploitation of architecture of the C40 processor.

At present, the time taken by the demonstrator to analyse a chicken image would be reduced to 2 seconds if a single BULL C40 was used instead of the TIM-40. The industrial requirement of examining 6000 chickens per hour (0.6 second per chicken) would be achieved by using four C40 to execute the Intelligent Vision System and another C40 to execute the Expert System.

8.3 INDUSTRIAL VERSION

The ALINSPEC system developed during the project was a prototype demonstrator. The architecture of the PC version proposed earlier would have to be modified to allow for operation in an industrial environment. The sensors would have to be enclosed in a water-proof shell. All electrical parts would have to be protected by a sealed enclosure.

The figure shown below is a proposal for an industrial ALINSPEC system for examining chickens.



A possible Industrial Solution

Standard industrial practices involve periodical cleaning of the production line. To avoid possible damage to the equipment, all the electrical devices would be protected by a waterproof cover. A transparent plastic shell could be used in sealing the camera and other optical parts. The shell would also protect the camera from dirt, blood and fat.

The commercial video camera used for the demonstrator (JVC TK1070E) was developed for use in the field of Television. The camera acquires two different half frame images, at different instances. If the object being imaged is moving, the resulting image is "blurred" and of poor quality for image processing.

This problem could be overcome in an industrial version by using a "non-interlaced" camera. The image would be acquired in one instance. However, these cameras are more expensive than the "interlaced" cameras.

The identification of skin texture meat features such as "Toad Skin" in chickens required images of a higher resolution, the extra cost associated with high resolution cameras would have to be weighed against the commercial benefits of identifying such features.

The cheapest industrial solution would be to acquire low resolution images using a low cost "interlaced" camera. The cheapest industrial version proposed would be priced as follows:

a) For the proprietary C40 proposed by BULL using the TMS320C40 processors and RAMTRON memory devices, the following prices were supplied by TEXAS INSTRUMENTS.

FIRM	DEVICE	CODE	PRICE (LIRA)
Texas	DSP	TMS320C40	375,000
Ramtron	Memory	DM2202J	575,000

The price of the memory devices are for the 16 pieces necessary for obtaining the required 8 MBytes of memory (4 MBytes of Global Memory and 4 MBytes of Local Memory).

b) For the development of the module, including routing and printed circuit production, CABRE (contractors for BULL) have supplied the following prices.

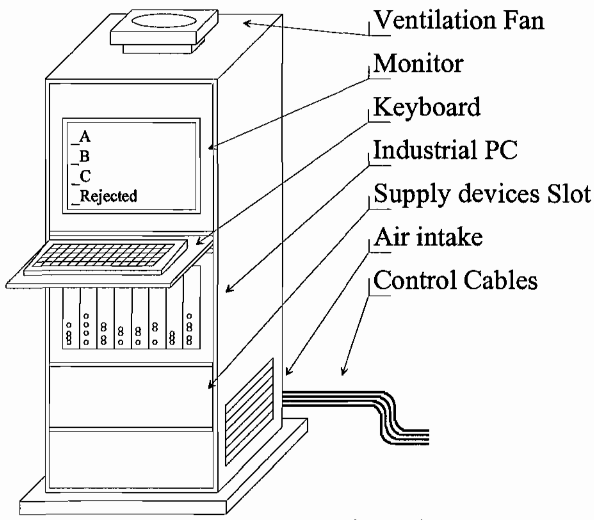
PROCESS	PRICE (LIRA)
Studying and routing	5,700,000
Prototype board	2,300,000
Electrical tests	3,700,000

The cost of manufacturing the modules was estimated at 3,870,000 Lira per module for a minimum of 50 modules, and 3,770,000 Lira for a minimum of a 1000 modules. However, during the last year of the project, other firms started producing C40 based modules at much lower prices. The cost of a single MDC40ED1-50 module produced by LOUGHBOROUGH SOUND AND IMAGES was about 4,700,000 Lira compared with 7,000,000 Lira for a TRANSTECH TDM411 module in 1993. The production of the BULL proprietary modules was estimated to be only profitable at a minimum unit of 50 modules.

The Special VME Interface Board developed for the ALINSPEC system was originally designed to fit into the VME slot of UNIX Workstations. The board was consequently re-designed in the PC ISA BUS format so it could be used in a PC based industrial version.

The proposed industrial version would consist of a MERCURY PCM601 (PC) supplied by ASEM with a SVGA 14" colour monitor, 486DX2 processor working at 66 MHz, 8 MBytes of RAM and 635 MBytes of Hard Disk. The monitor would be the MRACK 14"/I colour SVGA 14" suitable for assembly in a 19" Rack (container). The brightness and contrast controls of the monitor would be located on the front panel.

The industrial PC and all devices needed to drive the RGB camera could be placed in an industrial Rack such as the 19" Rack model PS4410 produced by RITTAL. The PC would be fitted out with two TIM-40 motherboards for mounting the Parallel Processing Platform and the Frame Grabber, a monitor to display the windows of the Man Machine Interface module, a keyboard mounted in a drawer (to be closed when not in use) and a suitable air ventilation system for maintaining the correct working temperature within the Rack. The figure below shows the proposed industrial version of ALINSPEC.



The proposed industrial Rack.

The following table lists the production cost of the proposed PC based industrial version.

Item	Description	Quantity	Price (Lira)	Total (Lira)
MERCURY PCM601 - ASEM	Industrial PC	1	3,640,000	3,640,000
MRACK 14"/I - ASEM	Monitor	1	750,000	750,000
QPC/C40B - LSI	Mother Board	2	2,520,000	5,040,000
MDC40ED1-50 - LSI	Tim-40 Module	5	4,700,000	23,500,000
TIM-40CFG-RGB - NEL	Frame Grabber	1	11,500,000	11,500,000
TK1070E - JVC	Camera+optics	1	3,737,000	3,737,000
PS4410 - RITTAL	Rack	1	2,000,000	2,000,000
FRK92/2 - IVO MALANCA	Optic Sensor	1	560,000	560,000
TOTAL				50,727,000

(£22,000)

The most expensive part of the ALINSPEC hardware was the Parallel Processing Platform. The overall cost could be reduced further by about 4,000,000 Lira using a set of five BULL proprietary C40 modules instead of the LOUGHBOROUGH SOUND AND IMAGES (LSI) modules.

The "once only" development costs necessary for further software optimisation and parallelisation were estimated to be about 1 billion Lira.

8.4 EXPLOITATION

Increasing automation led to the need for high speed, consistent, and verifiable examination of subjectively assessed natural products. The speeds required have made human examination increasingly unsatisfactory and difficult to manage. This led to the development of a prototype "demonstrator" for the subjective assessment of chickens and pig legs for demonstration purposes only, but targeted towards the needs of Coren-Frigolouro slaughter houses.

The requirements defined by the Coren-Frigolouro indicated a clear need for a system to provide quick and consistent evaluation of the quality of the incoming raw material. This requirement was somewhat different from that of the northern European meat industry where the raw materials e.g. live chickens and pigs, arriving at the factories were of such high and consistent quality, that the cost of an on-line quality assessment system would be extremely difficult to justify.

For chickens, the exact requirements of the industry in general have been somewhat misleading. At the beginning of the project (during the system analysis phase of the project), COREN/FRIGOLOURO's need for a system to provide their customers with products of consistent quality seemed to agree with the general needs of the poultry industry. However, towards the end of the project when the precise functionality of a chicken prototype "demonstrator" emerged, the interest shown by the poultry industry was not in a "marketable" ALINSPEC system for on-line quality assessment, but in how the ALINSPEC technology could be used in the development of "added value" systems, such as automated systems for identification and trimming of fat in chicken portions.

For pigs, the requirements were clear right from the beginning of the project. The slaughter lines in the pig meat industry were not as automated as poultry lines and as such the industry was in general not interested in a "marketable" ALINSPEC system. Where an interest was indicated, the requirements were inconsistent. However, there was a strong demand for increased automation of the lines, but the efforts at automation would be most effective when concentrated on particular unit processes on the line, rather than a wholly automatic line.

Although the requirement at Coren-Frigolouro pig factory, was for ALINSPEC to suggest to the operator how the leg should be trimmed, the techniques developed for such a system could be applied in future efforts at automating pig slaughter lines, e.g. identification of muscle groups for the positioning of automatic knives.

All this points to the conclusion that a functional production ALINSPEC system would be "ahead of its time". However, the techniques and tools developed would be useful to those developing "added-value" systems for the poultry industry, or pig line automated systems, or other industries where there is a need to assess quality of natural products in a similar manner.

The interest in the ALINSPEC technology would come from those developing automation systems rather than direct from slaughter houses or factories. The cost of the proposed industrial version (50,000,000 Lira) should help in promoting the potential of the

ALINSPEC technology, especially as the original estimate at the beginning of the project was (300,000,000 Lira) for a basic multiple RGB sensor system.

During the course of the project, other areas where the ALINSPEC technology could be applied were identified.

A proposal was submitted to the EROSKI Group in Bilbao (Spain) by BULL and OG for an automatic beef carcass grading system. The requirement was for:

- the *on-line* contactless measurement of the fat/lean ratio of meat cuts
- a colour based automatic grading of samples
- the automatic grading of carcasses and fresh meat cuts by evaluating their shape and colour.

OG and BULL also submitted a proposal to the largest Italian producer of baked products, the BARILLA Group. The requirement was for a computer vision based system for inspecting and grading up to 6000 samples per minute using multiple monochrome cameras. A prototype demonstrator software was produced using the Khoros tool.

OG carried out a feasibility study on the development of a real-time system for the on-line inspection of fresh oranges using RGB and infrared sensors. The study indicated that the techniques developed for ALINSPEC could be easily transferred to the identification of blemishes on fresh oranges.

Some advanced systems for the automatic butchery of meat carcasses are currently under development. These are also computer vision based systems which analyse the shape of a carcass by using statistical methods to decide how to trim the carcasses. The methods developed so far have proved unreliable. However, the approach used by ALINSPEC (fusing RGB and X-rays images in order to detect internal bones and principal muscle groups, and deduce the appropriate action based on the features identified) would be appropriate for such applications. The image processing and reasoning modules could be used in the positioning and control of automatic knives for trimming.

The Expert System module classified production samples intelligently using features reported by some means. Although the ES can deduce trends and causes of defective samples, making it a useful module in its own right, it is difficult to imagine any system other than an intelligent computer vision system, supplying it with the required meat feature information at the required speed.

The libraries of images used in the development of ALINSPEC represent a useful resource which could be used in developing utilities such as computer based training aids.

The wide range of techniques developed would also be useful to those developing "added-value" systems in other industries where there is a need to assess quality of natural products in a similar manner.

9. PUBLICATIONS AND CONFERENCES

9.1 PUBLICATIONS

1. Jewell, K. Scotter, C. N. G. "*A Market Survey of Automated Inspection Systems Available to the Meat Industry*", CCFRA Technical Bulletin No. 100, 1994.
2. Barducci, M. Barni, V. Cappellini, S. Livi, A. Mecocci, "*ALINSPEC Project: an Intelligent Vision System for Automatic Inspection of Alimentary Products*", Proceedings of 3rd Golden West Conference on Intelligent Systems, Las Vegas, 6-8 June 1994.
3. Mussa, T. S. Durrani, S. Livi, "*Intelligent Vision System for Real-Time Automation of Alimentary Products*", Proceedings of ICIP '94 Conference, Austin, November 1994.
4. Barni, V. Cappellini, A. Mecocci, "*Vision System for Automatic Inspection of Meat Quality*", Proceedings of ICIAP '95 Conference, Sanremo (Italy), September 1995.
5. Barni, A. W. Mussa, V. Cappellini, A. Mecocci, T. S. Durrani, "*An Intelligent Perception System for Food Quality Inspection Using Colour Analysis*", Proceedings of ICIP '95 Conference, Washington, October 1995.
6. Mussa, M. Barni, A. Mecocci, V. Cappellini, "*Intelligent Techniques for the Visual Inspection of Alimentary Products*", Proceedings of ICIP '95 Conference, Washington , October 1995.
7. Jewell, "*Computers spot the bruises*", Food Manufacture, Vol. 71, No. 4, p. 42, 1996.
8. Barni, V. Cappellini, A. Mecocci, "*Colour-Based Detection of Defects on Chicken Meat*". Submitted to "Image and Vision Computing".

9.2 CONFERENCES AND WORKSHOPS

1. CCFRA presented the ALINSPEC Project to the Brite-Euram Workshop on Handling Automation & Inspection in Manufacturing, Bristol, December 1992.
2. CEO presented the ALINSPEC Project to the Conference Come finanziare la ricerca industriale - Le opportunita' dei programmi comunitari ed internazionali, organised by FAST (Federazione delle Associazioni Scientifiche e Tecniche), Milan, 21-23 April, 1993.
3. Jewell, "Carcass quality assessment by image analysis", presentation at Image Analysis in the Food Industry conference held at FMBRA, Chorleywood, London, 11th October 1994.
4. CEO presented the ALINSPEC Project at "***5th EC Conference RTD on Industrial Technologies***", Brussels, 6-9 December 1994.
5. CEO, OG and CCFRA presentation at **TEKES (TEKNOLOGIAN KEHITTAMISKESKUS)**, at the Machine Vision seminar in ***Espoo,*** Helsinki, 29th September 1995.
6. L. Alba, L. Docio: "*CMHNN: A Constructive Modular Hybrid Neural Network for Classification*". Submitted to the 1996 International Conference on Neural Networks, to be held in June 1996 in Washington .
7. Ade-Hall, K. Jewell, M. Barni, A. Mecocci, "*Machine Vision for Assessing the Quality of Natural Products*", submitted for presentation/publication at "CAD/CAM Robotics and Factories of the Future" conference to be held at the Middlesex University in August 1996.

8. L. Alba, L. Docio, D. Docampo,"***Colour-based Recognition of Defects in Alimentary Products***", submitted to the 1996 European Conference on Signal Processing, to be held in September 1996 in Trieste (Italy).