THE UNIVERSITY OF LIVERPOOL

Automated Industrial Measurement And Inspection Using Image Processing

Thesis submitted in accordance with
the requirements of the University of Liverpool
for the degree of Doctor of Philosophy
by

Richard Mark White

Department of Electrical Engineering and Electronics
March 1991

To my family

with love

Acknowledgements

I should like to thank my supervisor, Professor J.Lucas, and Dr. A.B.Parker for their support, guidance and encouragement during this research. I also wish to thank the members of staff of the Department of Electrical Engineering and Electronics, especially the staff of the mechanical workshop and stores, who have made the research possible. I should also like to express my gratitude to Dr. W.Lucas at The Welding Institute and the Science and Engineering Research Council for funding the research. Finally I wish to thank Philips Components Ltd. for providing me with information and samples for the on-line metrology and inspection work.

I should also like to thank my colleagues and friends, Greg, Simon, and Angie for their help, encouragement and friendship during the research and while writing this thesis. Finally I am indebted to my father for the great help he has given me in writing this thesis.

" This life's five windows of the soul
Distorts the Heavens from pole to pole,
And leads you to believe the lie
When you see with, not thro', the eye. "

William Blake (1757-1827)

Abstract

Industry has in the past considered image processing to be too slow for measurement and inspection. This thesis shows that by using new image processing technology with careful algorithm design, industry can now consider vision as a realistic tool. As well as being fast enough, an image processing sensor must be as reliable and acurate as the currently used techniques. For vision sensors to be considered by industry, they must offer advantages over current methods. These are primarily: the capability of measuring features previously uneconomic to measure on a production line, and the ability to inspect a product without touching it.

This thesis demonstrates through two industrially relevant case studies that image processing can be considered as a realistic sensor for manufacturing industry. A survey of some of the most popular image processing systems is undertaken to explain the choice of The University of Liverpool's Transputer Image Processing System (TIPS) as the vehicle for the case studies. By using the transputer, the same software developed for a single transputer can run on many to increase the processing speed. The use of the language Occam for parallel programming is described and applied to the main projects of the thesis. These show the development of a portable gauge to measure weld profiles, and the application of image processing to the on-line inspection of components in manufacturing industry, with particular reference to the automatic measurement and inspection of ferroxcube inductor cores.

Rigourous testing of the results at The University of Liverpool is shown for both the case studies in order to prove the success of the research. In addition, the portable weld measurement gauge is in use by The Welding Institute at Cambridge. Since The Welding Institute have had the gauge, it has been used to measure a wide variety of weld samples, giving greater accuracy in much reduced time than previous measurement methods. In one case it allowed thirty one weld samples to be measured in less than an afternoon where previously it would have taken about a fortnight.

Contents

Table of figures

Table of plates

1. Chapter 1 - Introduction

Sight offers man his most powerful physical sense. His sight has permitted him to gauge distances significantly greater and smaller than his own reach, to invent tools and to form organisations and systems which exploit and develop his environment to his advantage. Mechanisation of industrial processes has provided man with an ever better standard of living, by providing larger quantities of any product at lower cost. However, in automatic processes, inspection of every item is seldom viable; some features cannot be inspected without looking at them, and to employ people to check every item is not cost-effective. Automation of visual inspection has previously been regarded as both too slow and too complex for practical application, however recent developments in electronics and integrated circuit design has provided processors capable of running programs which can "understand" simple pictures within a usable period. Therefore, re-assessment of the application of automatic visual inspection to industrial production is now due.

The research described in this thesis investigates the application of image processing to automate two differing types of industrial inspection situation. The first situation is weld measurement, which typifies a situation where measurement is difficult, slow, inaccurate and subject to variation in human judgement, yet of vital importance. Secondly, the manufacture of ferroxcube inductor cores illustrates the inspection of a complex shaped component in volume production, where manual inspection conflicts with a desire to increase production rate at reduced component cost.

In order to perform the re-assessment, the research described in this thesis has undertaken to consider those vision systems commercially available and to assess their usefulness in industrial applications. From this the fastest and most flexible vision system was chosen on which to perform the two case studies. Additional electronic hardware was designed and built, specific optical arrangements and illumination developed, and new algorithms produced so that development systems could be constructed to allow rigourous testing. Analysis of the results show that automatic vision sensors can perform measurement and inspection, which is both accurate and fast enough for industrial use.

1.1 Industrial vision

Automation of process control in manufacturing industry is now widespread, but, more often than not, inspection for quality assurance remains a manual task. Quality inspection offers an area which is ripe for automation and is one area where many people are employed, often equipped with manual gauges, to check visually the quality and tolerance of a sample of the products. Current methods to inspect and measure butt and fillet welds illustrate manual inspection, and these are discussed in the case study in chapter four. Automation for quality assurance inspection is considered in chapter four in which a new automatic gauge is designed and evaluated.

As technology advances and electro-mechanical systems become more complex and powerful, there are new demands for greater accuracy and precision in manufacture, greater speed of operation and production, and this itself requires rising standards of quality and specification. But as the manufactured systems and machines become more complex and valuable, completed units, with all their added value, must not fail at the end of the production line because a defective component was not spotted before it was included. This principle is illustrated in chapter five where the inspection of ferroxcube transformer cores is discussed. Here an automatic visual inspection technique has been developed which has the potential to be applied to on-line production.

Most measuring sensors presently used in industry are tactile and therefore potentially damaging to the objects which they measure. Since these sensors require a mechanical movement to come into contact with the components, they are slow to operate. Electronic vision offers industrial processes a sensor which is non-contact and proves to be more accurate than the human eye.

1.2 Speed and cost effectiveness

Industry will only persue new technology and adopt new methods or tools if, by doing so, either the cost of performing known tasks is reduced, or, new, previously impossible tasks can be accomplished. Speed of operation does not of itself reduce costs; there is no point in performing tasks quickly, if it means that the new machine and, perhaps its operator, then have to stand idle while waiting for the supply of jobs to either catch-up, or for the finished work to be removed. The industrial vision system will be only a part of the whole manufacturing process if its function is to permit the task of inspection to keep pace with the rest of the production. The whole production system can work no faster than its slowest component. The advent of automated manufacturing machines has, in general, speeded up the industrial process, and, it can be argued that, manual inspection is presently amongst the slowest and most labour intensive of manufacturing operations. Commonly, this slow operation is offset by inspecting only a selection of items from a production batch in order to gain an assessment of the quality of the whole batch.

The vision system offers great potential for on-line inspection and quality control in manufacturing by performing the tasks faster than by manual methods, and more cheaply. Also it allows 100% inspection of the products to be made. The case study in chapter five illustrates how Philips presently rely on sample testing to keep pace with the overall production rate of ferroxcube transformer cores, but desire 100% inspection and an increased production rate.

The reliability and skill of an automatic vision system for inspection lies in both the software and the electronic circuitry. The development cost of the former is a single capital expenditure and the latter is comparatively cheap, whereas human skill is a continuing overhead and labour is relatively expensive. In many applications there is a good prospect for the recoupment of investment in industrial vision systems for inspection over relatively short periods.

1.3 The speed of image processing

A camera scans each frame, line by line, looking at each pixel sequentially, and assigning to each a numerical value for its shade and position. In Europe, most cameras complete the frame scan every 40 ms. For the purposes of this thesis, real-time processing is taken to mean that the processing is completed during the period of one frame.

There is no point in seeking to advance the processing rate in order to complete it much quicker than this period allows, while the frame scan rate of available cameras remains at the present standard; the rate for the whole system cannot be increased beyond the limitations imposed by other components of the system, in particular, the camera. This is discussed more fully in chapter two.

What has been achieved by the study of the problems considered in this thesis is to transfer the limitations of the vision system either to the camera or to the device that receives the data. Of course, more complex problems require more processing, but this can usually be contained within the same period by dividing the processing between more processors.

1.4 The Transputer Image Processing System (TIPS)

The Inmos transputer is a fast, single chip microprocessor which has been designed to process several programs at the same time and "in parallel", taking only a short time to switch between the programs. Parallel processing allows programmers to structure their software more closely to the function that it must perform, rather than to tailor it to suit the limitations of sequential execution; this concept is discussed in chapter three. Furthermore, the transputer has been designed to operate with many other transputers, each processing a part of the complete program and passing information rapidly between them. This parallelisation at all levels permits software which has been written with algorithms operating in parallel for one transputer to be split up to run faster on several transputers.

Image processing requires a great deal of processing to produce any results for the scene in view. Consequently, fast processors are needed if the results are to be achieved fast enough to be useful to manufacturing industry. The transputer can allow the processing of the images to be divided between several processors; however, special electronic circuits are necessary to achieve this division. The Transputer Image Processing System, TIPS, has been designed to utilise the parallel capabilities of the transputer in order to speed up image processing.

1.5 Preliminary work

A prototype system for "An optical automated weld profile monitor" to measure certain parameters of T.I.G. butt welds had been developed by Dr. P.B.Terry [1], at the University of Liverpool in 1987. It was based on an IBM-AT computer acquiring the information about the weld from a picture captured into an external 256 x 256 frame store from a camera and laser arrangement. The laser light was provided by a helium-neon gas laser, linked by a fibre optic to a collimator, to shine across the weld. The profile was viewed by a vidicon camera inclined at an angle to the weld. The device was essentially a bench device, and developed using the technology at that time available.

The principle formed the basis of the research to produce a portable electronic gauge to measure weld parameters which is described in this thesis. A number of modifications were introduced to the Terry system during the initial stages of the study and some of these have remained through to the portable weld measurement gauge.

- ❑ The vidicon camera was replaced by a CCD camera
- ❑ The He-Ne gas laser was replaced by an automatically switched semiconductor laser and the fibre optic arrangement replaced by a directly connected collimating lens.
- ❑ Replacing the external frame store by a 'plug-in' framestore, but this was again replaced by the total change in the processing concept which is described here.
- ❑ Band pass filters were attached to the camera lens.

Many of these changes were made to make the monitoring head smaller, more robust, less complicated and portable. The camera used in this application was a miniature solid state CCD camera and it was fitted with a 780 nm wavelength bandpass filter to eliminate the background light from the laser stripe. The filter gave an improved contrast, but did not overcome all problems of background light in the system and this became of greater importance when the semiconductor laser was used which produced much less power. It became important to design the weld gauge head so that background light was excluded as far as possible. The solid state laser, being placed much nearer to the lens and weld, operated successfully with much less intensity and applied much less power, which was considered a useful safety advance if the device was to be used industrially.

Demonstrations of the Terry system to industry highlighted the need for a device which was able to measure the weld width as well as the parameters for which it had been already programmed. The proposed gauge would have to be able to measure welds between plates at any angle and not just butt welds. There was also a need for even smaller monitoring heads to be developed which could automatically inspect both internal and external pipe welds. These suggestions became the considerations throughout this research.

Other early work, which proved less profitable, involved the transfer of the Terry system to a Matrox, PIP-1024 [2], framestore. All the interface programs in 82086 assembly language were rewritten, tested and implemented, and a change to the "C" code was necessary in order to accommodate the increased size of the frame buffer (256 x 256 to 512 x 512 and the increased intensity level range from 6 to 8 bits). New methods were sought to solve the problems produced by the lower light intensity of the solid state laser which produced a less sharp edge than the gas laser, but all these software methods ran much slower on the single processor system.

The transfer to the transputer became the main alternative and the systems developed as a result of this change are described in the later chapters of this thesis. Completely new algorithms have been developed to overcome the shortcomings of earlier software. A new strategy was developed to find the edges of the weld metal, which would be more reliable, cope with shallow and sharply cornered welds, and allow parallelisation to be implemented more successfully. Further, new methods of measuring the weld features were necessary to allow both fillet and butt types of weld to be included. This necessitated the use of a two dimensional calibration.

1.6 Preview of chapters

This thesis is structured to introduce the reader to the ideas and techniques of image processing software and to the hardware on which it is performed. Two case studies then demonstrate the application of image processing to automated measurement and inspection showing that vision sensors are now realistic tools for both on-line and off-line use.

1.6.1 Chapter two - Vision systems

An overview of how a vision system works is described, outlining the basic elements and the various considerations which must be made when designing or choosing a vision system. With the understanding of what a vision system is, how it works, and what it is required to do, a survey of some of the most popular commercial systems is undertaken in order to explain the choice of TIPS as the vehicle for the case studies.

1.6.2 Chapter three - Image processing

This chapter introduces many of the methods commonly used to enhance images, and discusses techniques of feature extraction as well as describing the programming language, Occam, which has been used for the case studies.

1.6.3 Chapter four - Portable weld measurement gauge

This chapter describes the main project of the thesis. The design and development of a portable system which measures weld profiles is detailed. New algorithms have been developed in order to provide robust and accurate measurements of all weld types. Details of the exhaustive test which are required for industry are given to prove the validity of using image processing as an industrial sensor.

1.6.4 Chapter five - On-line metrology and flaw inspection

Chapter five considers the use of a vision system to replace the manual inspection of ferroxcube inductor cores allowing 100% inspection with a six fold increase in production rate. Currently, the central hole in the core is tested by a 'go-no go' gauge and examined manually for damage. In addition to these tests accurate measurement of other component dimensions is desired. This chapter then shows how the best optical arrangement was chosen to give a reliable image. The operation of the software is then described to show how careful design achieves both speed and robustness. In order to prove the reliability and accuracy of the system a number of specimen cores with known defects were exhaustively tested, and the results from these tests are discussed and presented at the end of the chapter.

1.6.5 Chapter six - Conclusions and future work

The results of testing in both the case studies of chapters four and five show that image processing can be successfully applied as a sensor for automated industrial measurement and inspection. The future holds the expectation of great advance in electronics and consequently, as the processing of images takes less and less time, more complex algorithms will be implemented. Image processing will undoubtedly become a common tool in manufacturing. The portable weld profile measurement gauge is now in use by the Welding Institute and undergoing serious consideration for development into a production model. Its potential for on-line control is likely to be pursued in future work at the University of Liverpool.

2. Chapter 2 - Vision systems and their capabilities

Chapter two describes the general uses to which vision systems may be applied in an industrial role, and proceeds to explain vision system hardware, and the considerations required when designing or selecting one. A number of commercially available products and systems are reviewed, which fall into two categories. Firstly, those which permit ordinary computers to be used for image processing, and secondly specific systems designed for the task. Chapter two concludes by explaining the choice of vision system used in the case studies, detailed in chapters four and five, and its facilities.

2.1 Vision systems for industrial application

Requirements for vision systems may be divided into either the manual, interactive movement and enhancement of images on computer screens for desktop publishing, or the more complex need for these systems to be applied to manufacture and process control robotics to give them "sight". It is this latter application to which this thesis is directed.

Automatic systems are required to have the ability of making decisions based upon the information seen in an image without the influence of an operator. When this type of vision system is coupled to a robotic system the image processing software must usually deliver the result from processing within a specified time, although not all automatic systems have such time constraints, Hedengren [3]. For example, in a system for tracking along a seam for welding the decision may be as simple as "move the welding torch to the left by $^1/_2$ mm", as is discussed in chapter four. In this case another operation, the welding, is dependent on the result from the vision system and there is a need for this result to be delivered within a specified critical time or the weld may be made in the wrong place, Clark *et al* [4].

A demand for a "non-contact" sensor, such as a visual one, is growing in the manufacturing industry where robots are now commonly used. These sensors can remove the limitation on a robot's speed imposed by a "contact" sensor which must touch a sample to acquire information about it. A robot's movement may be slowed down to prevent damage to either the sample or to the sensor. A robot which can "see" the component which it is to manipulate can accommodate a wider variation in the positioning of that component, than one which locates solely by knowing a pre-defined location. Consequently, vision sensors have the added advantage of reducing the overall cost of a production line by allowing a more flexible mechanical arrangement.

There are many functions which can only be solved practically by having a picture of an object. In addition to robot guidance, other applications include inspection for quality assurance, measurement and sorting. Furthermore, the information acquired by such systems can then be saved, analysed and used to improve production performance.

2.1.1 Robot guidance

A vision system used for robot guidance must assess, from the scene it is viewing, the direction in which the robot must move and also deliver this result within a limited time to the robot controller. The level of guidance may be as simple as "move left" or "move right" when directing a robotic arm to pick up an object. It may also be as complex as guiding a vehicle along a road; Morgan *et al* [5]. As the scope of guidance widens, the more decisions must be taken and more processing of the image is required.

In order to acquire the images a camera may be mounted in two places on a robot; either mounted on the arm which is to be guided or remote from the arm so that the arm is in the camera's field of view. The latter case is similar to the way in which man uses sight to guide his arms and hands to pick up an object. The solution chosen will depend on the application as each method has its different merits.

Positioning the camera at the point of movement (for example on the robot's arm) maximises the camera's resolution to give the most precise result to the control system operating the robot. In this case, with image processing used for seam following, Clark *et al* [4], the resolution allows the tracking head to deliver results with a precision of up to 0.02 mm for an image of 10 mm wide as shown in figure 2.1. This arrangement is similar to that used for the weld guidance and measurement in chapter four.

2.1.2 Inspection for quality assurance

Quality is one of the important factors relating to the success of any product in the market place, Stout [6]. Often, customer confidence in a product is given primarily in response to its appearance, then to its feel and finally to its reliability. Vision systems can help maintain these factors and have a direct effect on a product's saleability by guaranteeing its quality to the customer. Money can also be saved in production by correcting or rejecting substandard components early, such that further investment is not made in them as they progress along the production line. However, if the design of the product is fundamentally flawed, inspection cannot improve it.

As methods of production improve, quality improves and the customer comes to expect and demand ever increasing standards. If the customers receive badly finished or faulty goods it encourages them to look elsewhere for a better alternative and the manufacturers incur the additional cost of correcting the defects in the goods which were originally supplied. Quality may be quantitive as well as qualitive and many documents are produced to define the quality of a product, the most popularly known are national standards such as the British Standards [7] which are applied in chapter four to form the basis of weld inspection.

2.1.3 Measurement

When a person looks at an object he may be able to gauge its size, but only to a low degree of accuracy and not to a consistent level of precision. If viewing the whole object with a vision system which makes a measurement of the object's length, the accuracy is determined by size of the field of view, the orientation of the object in the frame and the resolutions of both the camera and the frame store. Typically the resolutions of the camera and frame store are 512 x 512 pixels. If the object was 1 metre across and almost filled the whole frame, the accuracy of the measurement would be about 1 metre divided by 500 pixels (given that the subject filled 500 pixels across the image). But, if the object was 1 mm long and filled the same 500 pixels across the image the accuracy would increase correspondingly. The case studies in chapters four and five examine the measurement of ferroxcube inductor cores and weld profiles.

2.1.4 Object identification for sorting

A production line may have a number of different components travelling along the same line. These could be sorted into boxes containing similar components by a vision system, which could identify the different types and pass on the information to another system. This would have great potential when extended to label the packaged components with bar codes automatically. Bar codes can be read by a vision system and using a camera with image processing software, which would then dispense with the flying laser techniques commonly found in supermarkets.

2.1.5 Data logging

Information from any inspection could be saved and used to give a better appreciation of how the manufacturing system is performing. If a manual system of component sorting is in use, the information showing the proportion of each batch which is too large, too small, or substandard is much less available for use in an analysis of the production system than if the information were already used by an automatic computerised system. For example in chapter five, the measurement of ferroxcube cores may reveal that when the raw material is purchased from one supplier the cores are more likely to be undersize or have chipped corners. Such information could be used to vary the amount of raw material used in the process, or combined with other data for the material to give better control over the process.

Data logging could be used to have a tighter control over the process by coupling it directly in a continuous process to "tweak" the manufacture. In the case study of welding, the measurement of the weld profile shortly after the bead is laid down could be used to modify the weld procedure while the bead continues to be laid. Additionally, The Welding Institute, as part of the portable weld measurement gauge project in chapter four has decided that stored information of weld profiles is valuable as a record of any inspected weld sample.

Data logging may be required for the following reasons:

- ❑ Historical record for future information
- ❑ Validation against specification requirements
- ❑ Counting numbers of components made in a batch or time period which failed inspection
- ❑ Reasons for failing inspection
- ❑ Distributions of results of size or flaws and types of flaws
- ❑ Linking logged data with time of day / week / year, temperature, humidity, raw material, supplier operator, new management decisions

A better knowledge of the process makes for better control of it and for a more reliable production.

2.2 Vision system design and consequent limitations

A vision system is a combination of electronic hardware and software which is capable of digitising a picture from a standard video source and which, by processing, allows the manipulation of a picture into the desired form or permits the extraction of some feature or features from it, Gonzalez & Wintz [8]. Vision systems, therefore, are made up of a number of connected parts:

- ❑ The image digitising (capture) circuitry
- ❑ Memory storage for the digital image
- ❑ A processing or computing element to operate on the image including program memory
- ❑ Circuitry for the reconstruction of the processed image back to an analogue signal for display.

This basic configuration is shown in block form in figure 2.2 .

2.2.1 Electronic parts of a vision system

The method by which images are transmitted has become standardised by television. A picture is displayed on a television by an electron beam scanning across the screen, line by line, starting at the top and finishing at the bottom. This is called a raster display. Since the television requires the image line by line, it is transmitted in this manner. An increased resolution of the image requires more lines and more information in each line and consequently takes longer to display each picture. In the UK the British Broadcasting Corporation (BBC) transmit pictures with a resolution of 625 lines. However, the response of the human eye is fast enough to discern that the display is scanned and not continuous at this resolution (the picture appears to flicker). To overcome this flicker the BBC transmit the picture in two parts, called fields, such that all the even lines are displayed first followed by the odd ones. Thus reducing the scan time from top to bottom by half and outside the response of the human eye. The standard used by the BBC is called Phase Alternating Line - PAL $_{UK}$, Cole [9], which defines the transmission of colour and intensity of television pictures. It is an analogue signal.

2.2.1.1 Image digitising circuitry

The image digitising circuitry of a vision system, figure 2.3 converts a standard video signal, such as that described above, to a stream of digital data representing each discrete point (called a pixel) of the image. The analogue signal is continuous through each line but the digitising circuitry samples this signal to produce the discrete points of the resolution allowed by the size of the image memory. For the applications described in this thesis, this is 512 pixels wide by 512 lines. The video signal consists of a synchronising pulse denoting the start of the field, followed by bursts of analogue voltage representing each line. The line intensity information is separated by horizontal synchronising pulses indicating the start of each line.

The synchronising pulses are removed by a sync splitting circuit, figure 2.3, leaving just the line data to be digitised by a fast analogue to digital converter which samples the signal produces a value for each pixel representing its brightness. This shade, for monochrome video, is usually resolved to 8 bits or 256 levels giving black as 0 and white as 255. Each pixel value is placed logically in the memory to build up an image which can be accessed by a processor.

2.2.1.2 Processor access to the image

The fastest technique of allowing a processor access to the image memory is to decode it into the memory map of the processor. The shorter the access time of the memory used, the faster each pixel can be read or written. When considering whether to base the portable weld measurement gauge, chapter four, on a portable IBM PC a problem became apparent. Owing to historical processor design limitations, the disk operating system, DOS, used by the IBM PC does not recognise memory in excess of 1 Megabyte and requires special basic input-output system calls (BIOS calls, Norton [10]) to access this area of memory. Framestores which are mapped into this area of the memory map have a slow pixel access since this BIOS software must be used. In an attempt to overcome this limitation some frame stores are designed with at least three registers which are mapped into the input-output map, IO map, of the processor allowing the reading and writing to any pixel by specifying its x and y coordinate in two of the registers and then accessing the pixel via the third. Although this is inherently slow, it is faster than using the BIOS software. When designing a vision system without these historic limitations, sharing the memory between the processor, the capture and the display circuitry is beneficial.

Only one of the three influential circuit blocks (capture, process or display as shown in figure 2.2) may access any one pixel at a time. Where the memory has only one method of access, via one address bus, only one of the competing circuits may access the memory at any time. Thus, with "single port" or normal memory, Hitachi [11], processing must stop during the capture or the display of each pixel while these operations take control of the memory's address bus. The design allocates a lower priority for the processor to have access to the memory than for capture and display.

In order to achieve real-time image processing, in which images are captured, processed and displayed within the period of one frame cycle, it is essential to lose as little processor time as possible during the capture of the next image and the display of the last. Special memory, which has two methods of access and is called "dual port" memory (video memory), Hitachi [11] has been designed to facilitate independent access by the capture and display circuitry and the processor. In this case the processor may only lose a very small amount of time due to being held up by a capture sequence.

It is inadvisable, if not impossible in some systems, to try processing an image while it is being captured, since the data is changing.

2.2.1.3 Construct the processed image for display

The display process can be seen as the reverse of capture. The "x" and "y" address counters are incremented to recreate the lines of data in digital form at the right time for a digital to analogue converter to reproduce an analogue signal with sync pulses added. Thus a PAL standard signal is reproduced for display on a compatible monitor.

Since most vision systems have enough memory available for two or four images, it is usual to share parts of the same circuitry (the x and y address counters) for display as that employed for capture, except for selecting a different frame buffer for each. To maintain continuous display, the circuitry continually increments the x and y counters transmitting the image in a chosen buffer, and only captures into another buffer when required. If it is connected, it is usual to use the synchronising pulses from the video source (camera) for a capture sequence. For real time processing, capture occurs every frame, as does display, and these are simultaneous and continuous.

2.2.2 Limitations and design considerations

The over-riding limitation of vision systems is having a very large quantity of data and insufficient time in which to process it. This problem can be reduced in a number of ways by considering the software as well as some hardware techniques for speeding up the processing.

2.2.2.1 Use a faster processor

Probably the vision system designer will have selected the fastest processor available, but, short of waiting for a better one to be developed, he will have to adopt design techniques which will extract the best performance from that which has been chosen.

2.2.2.2 Reduce the amount of processing required from the processor

Division of the processing between several processors, all working at the same time, is not possible if one is designing for an existing single processor computer such as the IBM PC. If each processor is connected together such that they all have access to the same image, then parallel image processing becomes possible.

Such an arrangement could be designed using parallel bus structures like the Versa Memory Europa, VME [12]. The limitation of this method of parallelisation of image processing is the time taken for each processor to access the image, holding up other processors while it does so and, therefore, creating a finite number of processors which could be connected together.

The parallel processing capability of the VME bus is very much limited by the number of processors trying to access the bus and the time taken to switch between tasks. The minimum time for a VME system using a 68020 microprocessor [13] running at 10 MHz to switch tasks is made up from saving 31 registers, each of 32 bits and picking up another 31 for another task. Each access takes 6 clock periods making the time for task switching greater than 25 µs. Added to this time is the execution time for the software which is needed to control the task switching. The time is almost eliminated by use of transputers as these have been designed for parallel processing both on one chip and on many communicating with each other. The time taken to switch tasks is between 600 ns and 1.1 µs.

2.2.2.3 Design considerations

(1) Resolution

Convenient ways to speed up the processing time are to use less data by reducing the x , y resolution or by disregarding a number of shades of grey. This can be implemented in hardware at the design stage, but it is better considered in software (if the need arises) leaving the system flexible.

The constraining factors on image resolution come from two directions, the cost of memory (but this always tends to go down) and the resolution of the video source. However, when vision systems are to be used for measurement, higher resolutions are often required and this necessitates employing greater processing capacity. The highest resolution from a camera is about 750 x 750, Pulnix [14], but these are very expensive and can only provide an effective resolution of 625 line using standard transmission.

Another factor worthy of consideration is the shape of the pixels. It is important that the equation of a circle should produce a circle on the screen and this is considered further in chapter five when describing the ferroxcube core measurement. Square pixels are favoured for industrial applications, especially if measurement is involved. Square pixels can be achieved in two ways, either by capturing a square portion of the whole image from the camera or by having a rectangular image of $^4/_3$ aspect ratio, with, typically, a resolution of 640 x 480, Matrox [15], and thus making it the same as the camera. A square image is the more convenient when writing the software, but either solution is acceptable.

(2) Colour

The description of vision system design has so far been concerned with monochrome sources and displays. Most vision systems, which offer image processing of monochrome images, provide output through three video signals for red, green and blue, incorporating a look-up table on each of these outputs, figure 2.4. The 256 grey levels can be displayed on a colour monitor as 256 different colours which can be selected from a palette of 256^3 or 16 777 216 shades, Niblack [16].

A colour image can be represented in two different ways in a computer; either as a combination of the three primaries (red, green and blue) or as a combination of intensities and the colour. The method of describing each pixel with three numbers representing the intensity of red, green and the blue is very similar to that with monochromatic image processing. The three colours are transmitted as three separate signals and the electronics for a Red-Green-Blue (RGB) vision system can be considered as a three-fold mono system. This type of system is useful if the application requires the extraction of information on one of the primary colours. However, when a combination of the primary colours is required, more processing is necessary than if the information is displayed in the alternative technique of "chrominance and luminance". The extra processing is incurred as three arrays of values (representing the red, green, blue images) must be processed rather than only two (chrominance and luminance).

Chrominance and luminance offers the colour information as two numbers, the first being a number corresponding to colour and the second to its brightness. This system can use algorithms which operate on the intensity alone, like monochrome vision systems, but also it offers the potential of extracting features which lie within a range of colours. The colour information (chrominance) is then represented as a number representing the colour of that pixel which is commonly selected from the chromaticity diagram, Niblack [16].

Colour is not necessary for many applications as the intensity gives enough information to extract the desired features. For example, a vision system designed to inspect biscuits in order to reject the burnt or broken ones, need not be a colour one, since broken biscuits will be the wrong shape and burnt biscuits will be darker than those correctly cooked. But if iced biscuits were being checked, such that the brightness or intensity of a pink iced biscuit was the same as a non-iced one, the colour information would be needed to show pink from brown.

(3) Frame buffers

Frame buffers are areas of memory used to store each image. Since it is not possible to process a frame which is currently receiving an image, as it is not yet there, it is useful to have at least two frames. This allows processing of one image while the next is being captured. If a monitor is to be used to display whatever the image processing has found, it is useful to show an image on which the processing has been completed.

In a vision system with two frame buffers, the frame which is being processed will be displayed. At the start of the processing this will show a "raw" or unprocessed image which will build up to the completed picture. Thus, the complete image will be displayed briefly before the frame buffers are swapped and a new raw image is shown.

A better arrangement is three or four buffers, one of which is being displayed, one is receiving a new image leaving the processor to read the image in the third buffer, either drawing the results into the fourth buffer or back on top of the image being processed. This method becomes of greater importance as the image processing is made to run more quickly, until for every frame cycle a new image is being processed. In this case of real-time image processing the sequence of frame buffers may be as follows.

Capture frame	Process Frame	Display frame
0	1	2
2	0	1
1	2	0

The table above shows the order in which the frame buffers may be used in a vision system with three frame buffers, numbered from 0 to 2. The last frame to be captured is being processed, the last frame which has been processed is now being displayed and a new image is being captured into the remaining buffer.

This now shows the fastest response time of an image system to be:-

Response time	=	Capture time	+	Process time
	=	40 ms	+	40 ms

The response time is governed by the processing time which in conventional systems is much greater than the indicated 40 ms, but often of the order of minutes or hours.

(4) The interlaced video signal

The term "interlace" describes how the image is transmitted as an analogue signal between the source, the vision system and the display. Also it describes how it is clocked out of the vidicon tube or Charge Coupled Device (CCD) (see section (6)). An interlaced signal is one where all the odd lines of the image are transmitted as the first field followed by all the even lines to make up a complete image. The reason for doing this lies with the response time of the human eye, as discused in the introduction to this section. If the whole frame was transmitted with odd and even lines successively from top to bottom, the screen would have dimmed by the time the line was next scanned by the electron beam and the display would be seen to flicker. The flicker is less apparent to the viewer if all the odd fields are displayed and then all the even ones. The flicker is still clear when single pixel lines are drawn horizontally on the image.

(5) Non-interlaced cameras

Non-interlaced cameras are available which either reduce the vertical resolution and transmit the whole image in one frame of 20 ms or by using a higher frame rate to allow more lines to be transmitted in the same period. The higher rate is 31.5 kHz as opposed to the 15.6 kHz rate of the interlaced display signal, which overcomes the flicker problem of the display monitor.

2.2.3 The video camera

In conclusion to this section on vision system hardware and design, a discussion of the video source is included. There are two types of electronic camera commonly in use, those based on a photoconductive tube and those which employ a micro-electronic chip of charge coupled devices. The latter design is now becoming the most widespread as it is more robust and has a longer life.

2.2.3.1 Photoconductive tube camera (vidicon)

There are three main parts to the vidicon camera, the electron gun, the scanning system and the photoconductive target onto which the scene is focused, King [17]. The electron gun produces a beam of electrons which are focused, aimed, and scanned across the photoconductive target in a similar manner as in a display tube. The image of the scene is focused on to the target which consists of a layer of photoconductive material having the property of exhibiting a high resistance when low light levels are incident on it and an increasing conductivity as the amount of light falling on it increases. The scanning action of the electron beam across the target makes a circuit where the voltage at the electrode attached to the target varies according to the resistance of the target. This voltage fluctuation is the analogue video signal.

The major drawback of the tube is that a constant image slowly "burns" itself permanently on to the photoconductive layer. Since the vidicon camera is based on a delicate cathode tube design care must be taken when in use to handle it with this in mind. Most commercial versions require a 240 volt or 110 volt supply and therefore do not lend themselves to incorporation in portable systems. The design is also physically large, typically 150 mm x 100 mm x 75 mm.

2.2.3.2 CCD cameras

These micro chip cameras are small, robust, reliable, operate on low voltage and have a low power consumption. They are typically 100 mm x 50 mm x 50 mm, but can be as small as two standard match boxes back to back. The CCD chip is not damaged by long periods of constant and unchanging image nor very bright light. As VLSI technology improves the resolution of these devices increases to beyond that which was possible by focusing an electron beam. It is possible to get CCD cameras with resolutions of 1024 x 1024.

CCDs can be considered as analogue shift registers, Stafford [18]. When used as photodetectors in an array, organised into rows and columns with an optical image focused onto them produce an electronic picture, also known as an image. The CCD array can be regarded as an area of closely spaced capacitors which hold charge corresponding to the amount of light falling on them. Each charge is amplified and read out as an analogue voltage corresponding to a point and assembled to form the lines of a video signal.

2.3 A review of commercial image processing systems

Many frame stores are available which are intended for use with "PC bus" [19], VME [12] and other host systems. As with all "high tech" electronics, there is a great deal of marketing "hype" to sift through before the actual potential or power of a system can be evaluated. Most systems offer libraries of routines performing image subtractions, thresholds, edge detectors and histograms. These can be of use to make the image processing simpler and are all described in chapter three, but, more often than not, they add an enormous amount of processing time before the feature extraction can begin.

Many of the systems offer real-time frame capture; any system not offering this facility builds up the image over several frames and requires a much greater stability from the picture and, of course, takes longer to capture it. Some systems offer accelerator boards which perform the library functions quicker or use Digital Signal Processing (DSP) hardware. These can be of benefit, however, there is no compromise for having fast access to the image and the capability to deliver results back within a reasonable time. The period depends on the application; if it is off-line, leaving a program running overnight to process a single frame may be acceptable, but for any form of robotics, in which control is required, such a response is inadequate. In robotics a sub-second response is necessary. The ultimate goal is to guarantee results in less than the time taken to capture the next image, thus making the limitations lie with the video rate and the camera rather than the software.

Most systems cannot come close to achieving "real-time" processing and are intended for the graphics type of applications. In this respect, image processing for desktop publishing is currently the largest market for vision systems, but few of these meet the criteria required for industrial applications.

One guide to the speed of a system is simple pixel access time, but this does not give fast processors credit for the processing done off the frame store. A better guide is to take a general purpose algorithm and compare its performance on different systems.

2.3.1 IBM compatible based image processing systems

The IBM personal computer is an adequate machine on which to base an image processing system if the speed of response need not be sub-second. There are many plug-in frame-stores which enable image processing to be performed.

2.3.1.1 Matrox's frame-stores

Matrox Electronic Systems [15] is a Canadian company specialising in graphics, image processing and alpha-numeric display controllers for Multibus [20], VME bus and IBM PC bus application. All their frame-store boards offer real-time frame capture capability. (See section 2.1.1.1) They also offer 256 (8 bit) levels of grey, from black to white, and resolution of up to 512 pixels wide by 512 pixels high.

The IBM PC bus frame stores are limited by the processor in the PC and by the access time specification of the PC bus. Since these frame stores are mapped as input/output devices the access time for one pixel is as follows:

Pixel access time = Number of processor cycles x cycle time

However the time needed to read a pixel and then write it back again is double the pixel access time; the time taken to read all the pixels in one buffer and copy them to another was measured to be 1.21 second for an image of 512 x 512 using the library function supplied with the PIP-1024 framestore.

The Matrox PC bus frame stores offer the potential of performing image processing algorithms to solve most applications. However, since the times considered are for the fastest access to the frame store, they cannot be improved and do not enable image processing to be used in on-line applications in which results are demanded sub-second.

The VME and Multibus systems are limited in a similar way to the PC bus, except that now it is possible to let more processors have access to the image memory. This provides a useful advance, given that more processor time is spent processing the pixels rather than accessing the image. If the latter is the case, processors will be left waiting while others are accessing the image. In most image processing applications the majority of the time is spent looking at the image.

2.3.1.2 Data Translation's frame-stores

Data Translation Ltd [21] primarily targets its imaging products towards the PC market, but offers a Micro-VAX based frame store as well. These frame stores suffer the same limitations as those produced by Matrox. The main distinguishing feature is the method by which it interfaces to the PC. Data Translation's frame stores are memory mapped, above the DOS 1 Mbyte boundary. The memory mapping technique offers the potential of simply reading from and writing to pixels as an area of memory defined as a two dimensional array. This gives a much faster access than the I-O mapped Matrox boards. However, since DOS does not recognise memory over 1 Mbyte, the access must be made through calls to the BIOS. The short access time achieved by memory mapping is now slowed down by the BIOS routines and proves to be slower for single pixel access than the I-O mapped design.

It is quicker if the whole image is read down into program memory as one block, than moving the whole of an I-O mapped accessed image, but the image processing time must then be added to this fundamental overhead.

Data Translation market a more advanced board with a processor which allows frame averaging, zooming and scrolling, frame addition and subtraction and Boolean functions, but it is still necessary to perform any specific feature extraction using the IBM PC.

2.3.1.3 Digithurst's frame-stores

Digithurst [22] offers a more graphics and desk top publishing oriented solution, which is designed for human interaction and is slower. Colour is a key feature, but this is not often required in industrial applications. A transputer accelerated PC based board offers the potential of higher processing speed, but this is only available for the library routines similar to those offered by Data Translation and Matrox, and the image processing is still performed on the IBM PC.

2.3.2 High performance vision systems

2.3.2.1 The Akebia 'PIPE'

In 1989 Akebia [23] launched the 'PIPE' (Parallel Image Processing Engine) which offered the possibility of much higher performance than those which were previously available and with the possibility of real-time processing.

The PIPE is based on the Inmos transputer and uses a high speed data broadcast bus to move the image in and out of the processors at 100 Mbytes / second. The system comprises:

□ Video input and output boards (V_{in} & V_{out})
□ Dual Transputer and Tripart Memory boards ($DTTM_0$)
□ Quad Transputer boards (QTs)
□ User Interface board (UIF)
□ Transputer Input and Output boards (TIO).

The video input & output boards send to each processor the part or the whole of the image which it requires, addressing each pixel to its destination board. Image processing can be performed on these boards where each transputer has direct access to its image memory. This system allows different transputers to 'look at' different parts of the image to speed up data or features extraction ready for the next image. The ability to divide up the image processing on to several processors gives the most effective route to achieving the real-time processing objective.

2.3.3 Do the commercial systems serve industry's needs?

All the systems reviewed are sold successfully, but mainly for commercial use in an inactive capacity. All except Akebia's PIPE fail to offer the processing power capable of meeting control time limits, and thus they are unable to be used in many manufacturing systems.

The Akebia PIPE fulfils the time criterion, but the method adopted to transport the images limits the expansion of the system and the number of processors which can be connected together. This system is also very expensive.

A cost effective and adaptable solution is still not available commercially, but a system which meets most of the criteria has been developed at The University of Liverpool. It is around this system that the case studies described in chapters four and five have been developed and implemented.

2.4 The Liverpool University Vision System - TIPS

The Transputer Image Processing System, (TIPS), Chambers [24], has been developed at the University of Liverpool in order to achieve the goal of real-time image processing for general application to solve industrial problems. This system achieves real-time by allowing many processors to have access to each image and by dividing the processing between them. This parallel solution to the processing has been made realistic by use of the transputer and its parallel programming features. Real-time, in terms of image processing, requires each frame to be processed before the next arrives. Using standard BBC transmission rates of 15.6 kHz with an interlaced frame, a complete 512 x 512 resolution image is achieved every 40 ms, giving 40 ms to produce the result.

TIPS was chosen for use in the case study in chapter five and compatibility was chosen for the portable weld measurement gauge in chapter four.

2.4.1 What makes TIPS useful for industrial processing?

TIPS moves the speed limitation of image processing from the processing to the capture rate of the camera, which makes the fastest sampling rate possible for getting results 50 Hz (the cycle frequency of the camera). In addition to the speed capability of the system it has been designed with industry in mind and is compatible with the 19 inch rack standard, Schroff [25]. The modular design permits the expansion of existing TIPS systems, to enable them to operate faster or to interface to a different robot, more easily than in the other popular bussed systems.

2.4.1.1 Overcoming the processing limit

TIPS has been designed around the 15.6 kHz interlaced, RS170 [26], PAL standard which gives a complete new image, or frame, every 40 ms or 25 Hz. However, when required, TIPS could be modified to accept images at twice that rate making it compatible with the 31.5 kHz non-interlaced standard. For all case studies shown in this thesis the 15.6 kHz standard was used.

How does TIPS manage to produce results at frame rate which few other commercial systems do? TIPS does not use a conventional processor designed for sequential processing of programs, but instead employs the Inmos transputer [27]. The transputer is a processor which allows the programmer to write code which is processed in parallel and can run on one, or more than one, transputer. By using parallel processing with several transputers, the processing of each image can be split to give each transputer a small part of the total processing required. Each transputer will then finish its smaller task quicker and since all the transputers are processing the image at the same time, the whole process is completed in a fraction of the time taken by a single processor.

The four frame buffers hold images of 512 pixels high by 512 pixels wide (totalling 262 144 pixels) with 256 grey levels. If every pixel has to be read and the system uses memory devices which take 150 ns to access, this operation of reading each pixel, with no processing involved, will take 39.4 ms. However, the transputer is a 32 bit processor and reads four pixel intensities at once, thus it only takes a quarter of this time (10 ms) to read the whole image. If the whole image is to be written back as well, the time is doubled to about 20 ms. Image processing requires some mathematical operation to be performed on the image. For an operation involving only one pixel the processor may read it, process it and write the new value back. Consequently the time to perform image processing is the sum of all the pixel reads with all the writes added to the sum of the processing. With only a small amount of processing, a single processor cannot complete its task within the time period for the new frame. If two processors were used, for example one looking at the left half and the other at the right half of the image, the amount of processing which could be performed would be doubled. Of course, in many industrial algorithms it is not necessary to look at every pixel.

The processing power which is made available using TIPS has another advantage which is that more complex and powerful algorithms can be implemented giving increased flexibility and reliability. Instead of placing tighter restrictions on the image being viewed (that is it must always be in "this" position or at "that" level of brightness), a more flexible approach can be made.

2.4.1.2 Expansion of TIPS at a later date.

Later expansion of TIPS is possible for two reasons; firstly the system has a very modular design, allowing additional boards and transputers to be added to increase the image processing power further. Secondly the method of writing the code using Occam, Hoare [28], with parallelism at all levels, makes splitting the program from one processor to several much easier. If more processing power is required, another image processing board can easily be added to the system, thus bringing another transputer to help achieve the desired speed. The existing program can then be reconfigured for some of it to run on the extra processor.

2.4.1.3 Compatibility with the Industrial Environment

All the circuit boards which comprise TIPS are of standard double extended Eurocard size (233.4 mm x 220 mm), which are widely used and compatible with the 19 inch rack standard Schroff [25]. TIPS can therefore be conveniently packaged with any other special function boards to operate from a single unit box. The 19 inch rack standard allows a system to be housed to operate in most environments and meet the specifications of ruggedness for industrial operation. This is a most important consideration as some factory sites may require filtering from very dusty atmospheres or cooling from intense heat.

Compatibility with standard board sizes allows the use of many special function boards for control of a variety of different robotic actuators and sensors. In addition to a wide variety of commercial boards the University of Liverpool has designed a number of processing, robotic control and communications boards to facilitate the construction of complete industrial image processing and control systems. Chapters four and five show applications of its use.

2.4.2 The TIPS design

The system comprises three component boards, a frame capture board, a system display board and as many processing boards as speed and cost dictate. All are connected by the system back-plane. The system uses the Inmos transputer to allow full parallelisation of the image processing on and between the processors.

2.4.2.1 The Inmos Transputer

The transputer is a very fast single processor capable of 30 MIPS (million instructions per second) and 4.3 MFLOPS (million floating point operations per second) peak instruction rate. But its real power lies in the fact that many transputers can be networked together to form a parallel processing array of greatly increased performance. TIPS uses this parallelism to divide the image to be processed between many transputers, reducing the amount of data to be handled by each processor.

The transputers used in the system are the T800 and T400 series, which are both 32 bit processors with internal clock speeds of either 17, 20, 25 or 30 MHz. The T800 series transputers are pin processor compatible with the T400 series, but have an additional floating point processor as part of the chip. Full technical details of the T805, T801, T800, T425, T414 and T400 can be found in the Inmos transputer data book [27].

2.4.2.2 The TIPS architecture

Superficially, TIPS compares with the figures 2.1 and 2.2 for an overview of a general vision system and with capture, processing and display circuitry. TIPS is made up of three component boards which are described below and shown *in situ* in figure 2.8.

(1) The Frame Capture Board

The frame capture board incorporates a fast, flash, Analogue to Digital Converter, ADC, which converts the analogue video signal into a stream of digital pixels. Each of these pixels is defined with a number from 0 to 255 representing a shade of grey from black to white. These are then communicated to the image processing boards via the system back-plane in the pixel bus. The frame capture board provides all the necessary system timings for operation.

(2) The System Display Board

The system display board is responsible for the collation of pixels from the image processing boards, via the back-plane and the second pixel bus, and converts them back into an analogue signal for transmission and display. This board transmits an image as an RGB signal with a colour look-up table able to generate 255 different colours selected from a palette of 16 777 216 different shades.

The system display board has a transputer on it to direct system operations, but it can also be used as a root processor for the whole system. Since the display can have a resolution of 640 x 512, the display board also has an area of display memory of resolution 640 x 512 which can be used in addition to the image processing memory for graphics.

(3) The Image Processing Boards

Each of the image processing boards possesses the captured image and can store four complete images, each with a resolution of 512 x 512 square, 8 bit pixels. The memory is mapped directly into the transputer's memory map which allows fast access to the image. In addition to the 4 x 256 kbytes of image memory 1 Mbyte of program memory is available.

2.4.2.3 TIPS functioning.

As the image is digitised it is placed on the first pixel bus, see figure 2.8 and is received in parallel by all the image processing boards and, at the end of the frame cycle, leaves them each with duplicates of the whole image. At the same time, the second pixel bus is displaying the contents of one of the other frame buffers by choosing which pixel to display from each board.

Meanwhile, all the image processing boards have been processing the previous image ready for display at the next cycle.

2.4.2.4 Transputer networking

Once a program is divided to run on two or more transputers, these parts of the whole program, known as procedures, will need to be able to communicate data between themselves as well as the external systems. The transputer is designed to be networked by point to point communication lines called links. These are represented as channels in software such that two procedures running on the same transputer are seen by the program to communicate in the same way as two procedures running on two transputers connected by a link. See figure 2.5.

All transputers used in the system have four bi-directional links (except the T400, which has two) which allow the network to be configured in any way required for the algorithms to be implemented. Not all the links need be connected and the figure 2.6 shows three examples of configuration which may be used.

In configuration A on figure 2.6 , the processors could be mapped over an image to process a rectangular section of it, and to communicate any features to each other, or back to the system display board via links A B C D. Processor T_1 could be processing the top left corner of the image, T_2 the top middle section, T_3 the top right corner and so on. Alternatively, if the feature on the image was known to be moving from left to right, the processor may be more effectively networked in a line as in configuration B, where each transputer processes a vertical strip of the image.

The hardware and software are tightly bonded, but can easily be modelled on one processor producing results slower than on several. The discussion of the software implementation is developed further in chapter three.

The use of the transputer's networking capability can be applied beyond the image processing to allow development of very modular systems. TIPS could form a subsystem of a complete control unit which has its own network of transputers communicating with another network of transputers sited elsewhere in a factory. Figure 2.7 shows how this may develop. Since the links communicate in a serial manner, they may be connected easily by fibre optics and placed a long way from each other.

As well as the networking and modular communication the transputer compares well with other more popular processors. The transputer is a very powerful single processor which has the security of "military status" and is also cheap compared with its similarly powerful competitors. Interfacing to the transputer is also easier than the Motorola 68000 series or Intel 80x86 series processors. It offers dynamic memory support on chip and does not require any external support chips except a 5 MHz crystal oscillator.

2.4.3 TIPS conclusion

TIPS currently offers the most cost effective method of processing images at high speed with the capability of achieving a real-time rate. Only one other commercial system has this power, the Akebia PIPE, but this has a much more complex architecture and its cost is much higher. The TIPS design is innovative in its simplicity and performs well in practice.

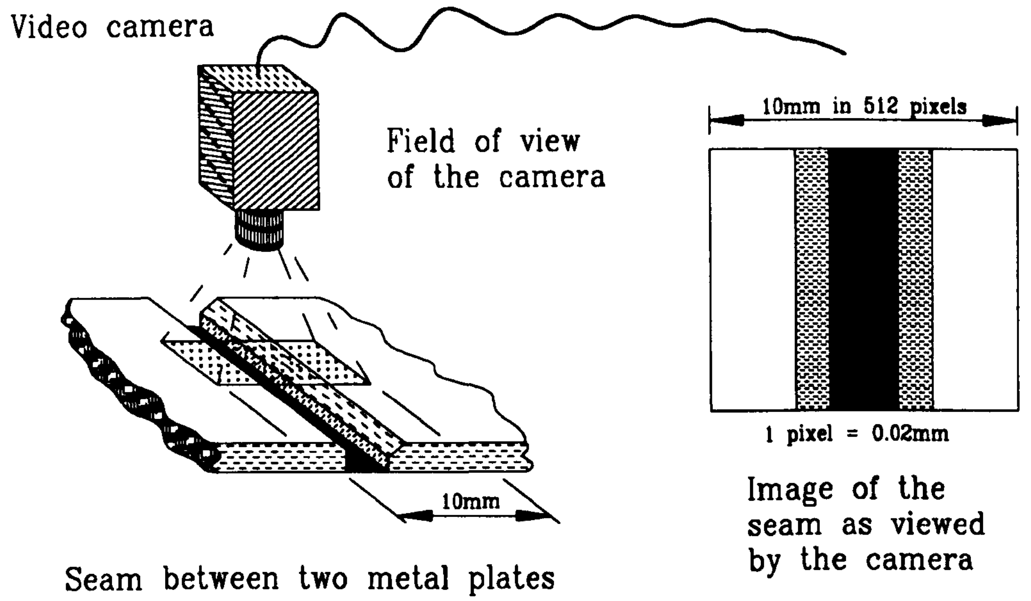


Figure 2.1 Video camera mounted over a seam between two metal plates showing a real resolution of 0.02 mm per pixel

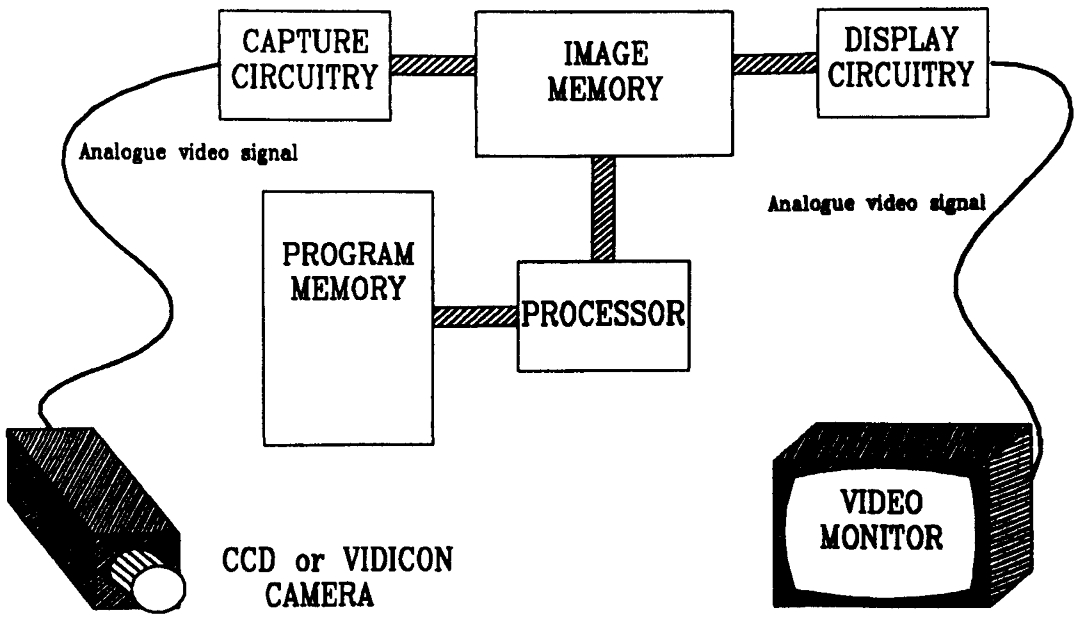


Figure 2.2 Overview of a vision system

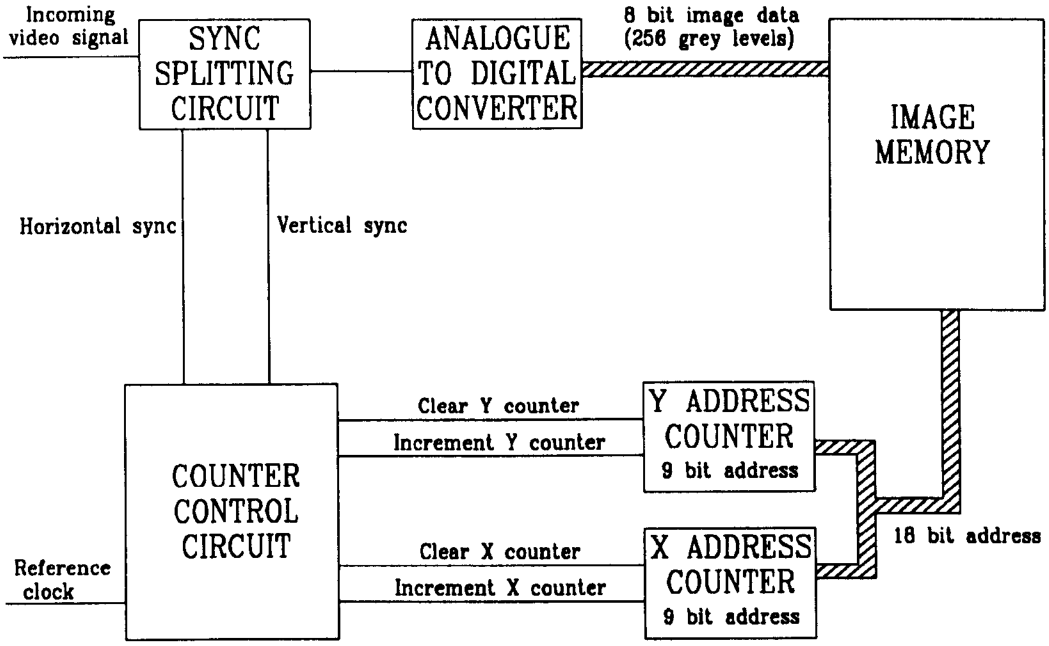


Figure 2.3 Elements of the capture circuit of a vision system

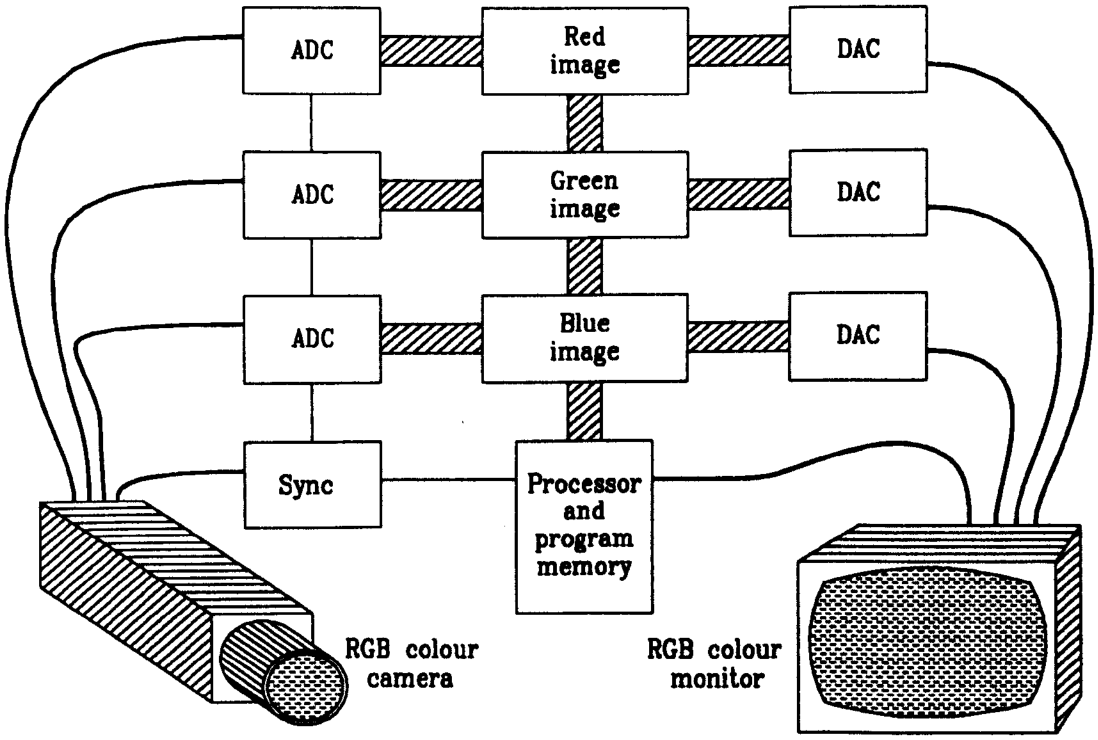


Figure 2.4 Two concurrent procedures running on a single transputer and on two transputers connected by links

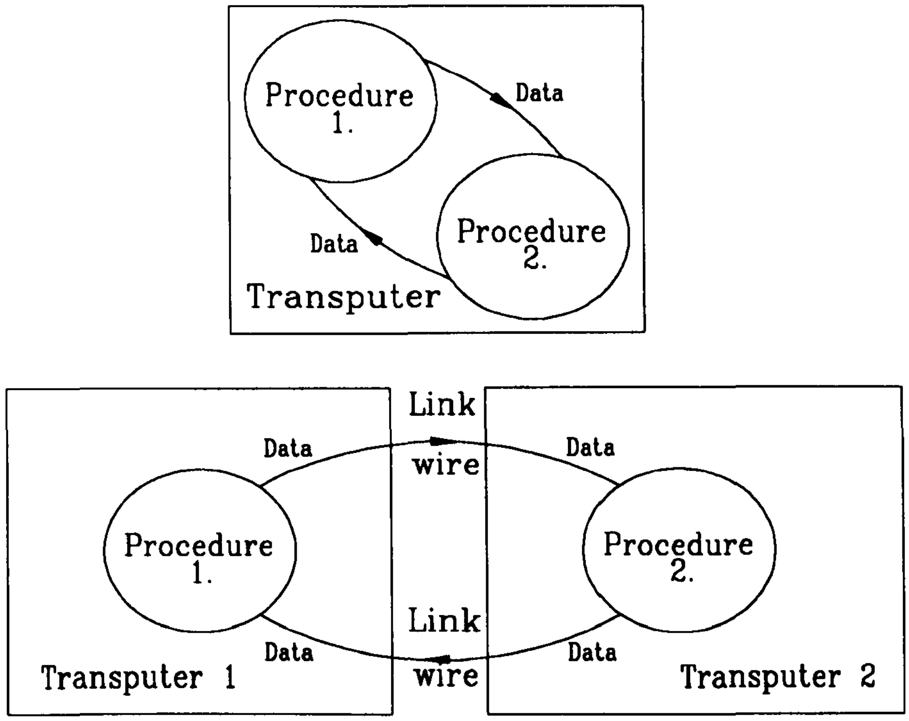
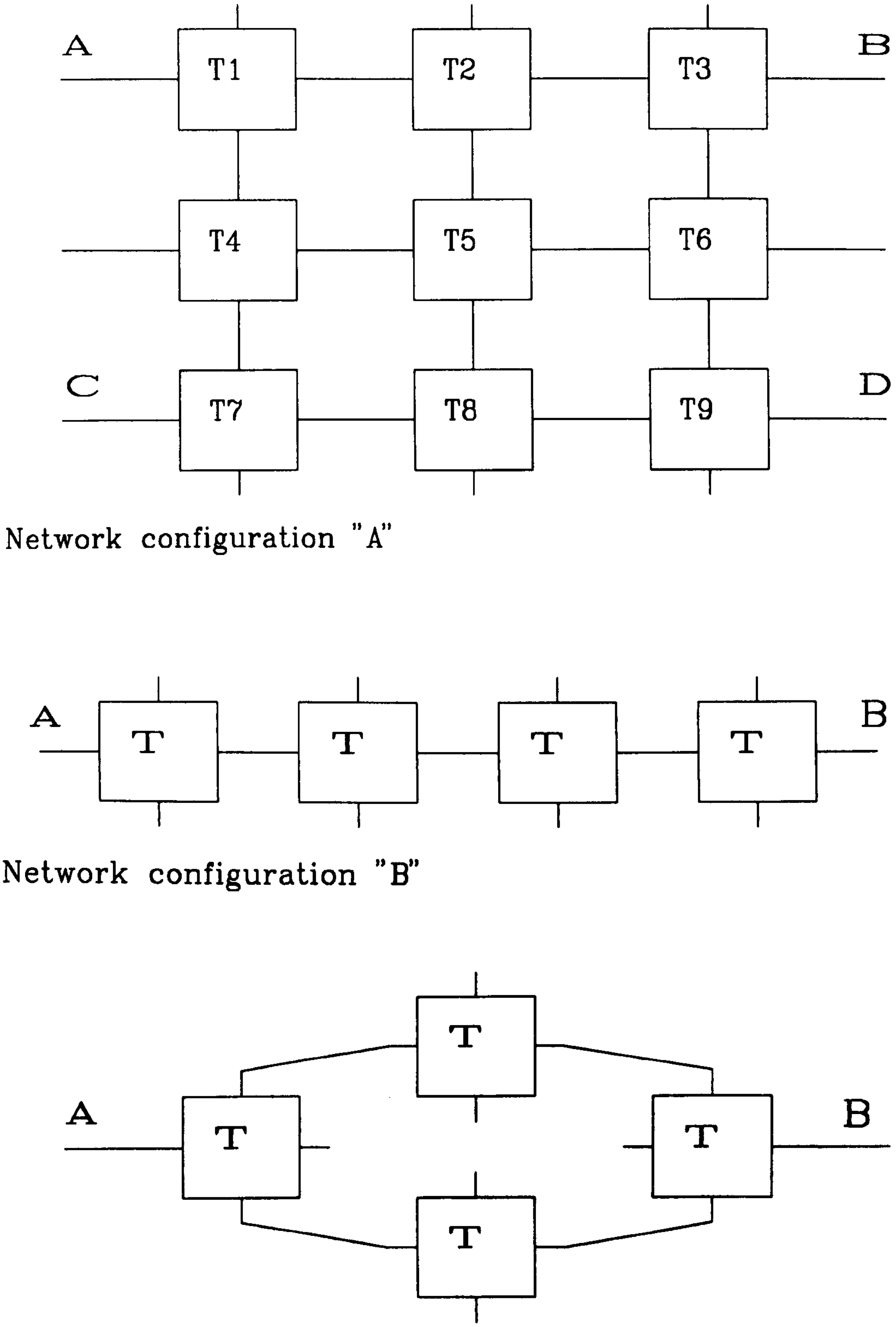


Figure 2.5 Two concurrent procedures passing data, on one transputer, and on two transputers connected by links



Network configuration "A"

Network configuration "B"

Network configuration"C"

Figure 2.6 Alternative transputer architectures

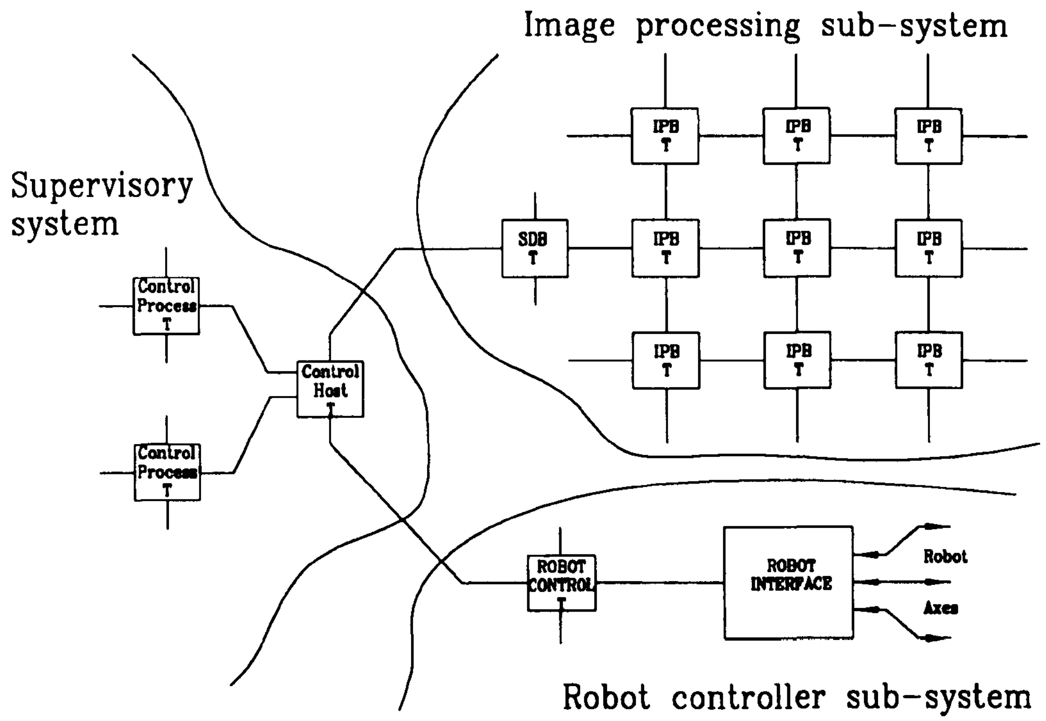


Figure 2.7 A possible configuration for a network of transputers for industrial image processing and control

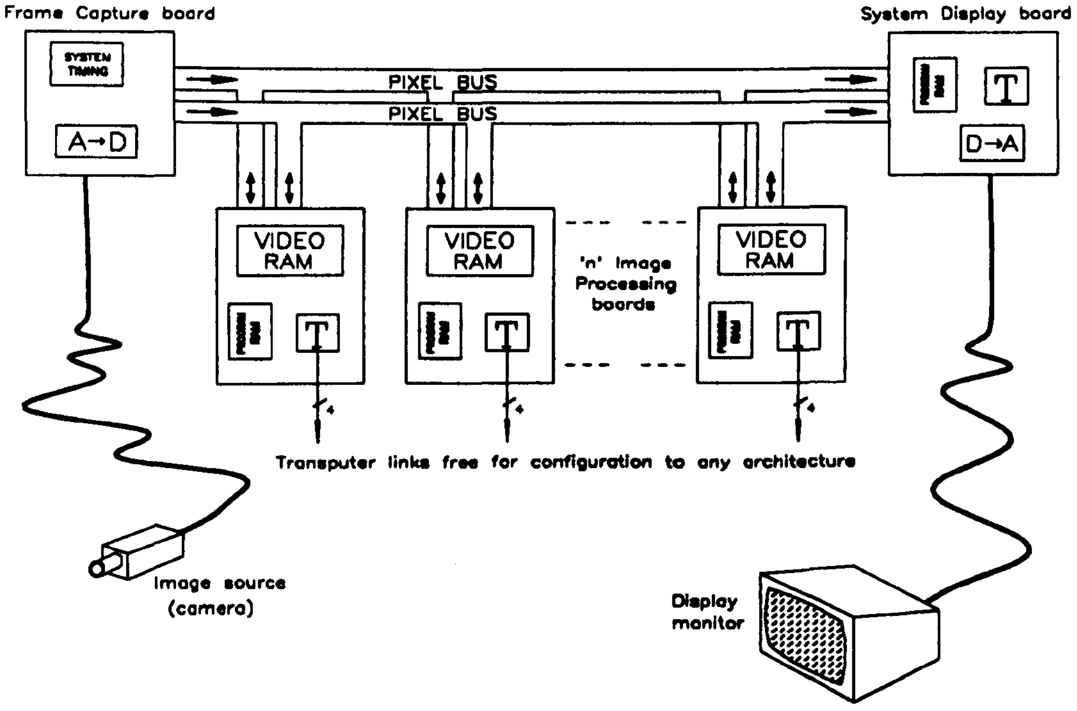


Figure 2.8 Transputer Image Processing System (TIPS)

3. Chapter 3 - Image processing

This chapter describes the software techniques of image processing used to produce results for industrial applications. In such applications the overall speed of producing the results is of importance as the image processing software must work as fast as the production line demands. Parallel programming with multiple processors provides the potential of dividing the processing between a number of processors to reduce the time taken to produce the desired results. Parallel versus sequential programming is discussed with a description of the Occam programming language on the Inmos transputer. This is intended to provide a basis of understanding for the description of the algorithms and software described in the case studies in chapters four and five.

3.1 Techniques for solving image processing problems

Once the image has been specified and the desired features identified, there are many well established techniques available to enable a computer to achieve image enhancement and feature extraction and these are discussed here. But often, in order to achieve a result within any given time constraints, these methods are applied selectively and locally to small areas of the image. It is often more reliable and effective, and requires less processing if unique engineering solutions, tailored to the specific image and its particular features, are developed.

3.1.1 General algorithms

Much of the research work has been involved with developing algorithms which are applied to the whole image. These algorithms are largely designed to perform the functions below, and are geared to scenes of unspecified picture. Typically these algorithms are intended to perform one of the following:

- Improve the display properties
- Provide information about the image for other algorithms
- Segment the image into regions
- Remove noise
- Remove blur
- Enhance features such as edges
- Perform geometrical operations like magnification or rotation
- Code the image to reduce its size for storage or transmission.

These algorithms may be used singly or in combination to achieve the desired result. They can be applied to any image and will produce a result which is mathematically defined.

3.1.1.1 Image histogram

The image histogram provides a measure of the distribution of pixel intensities in an image, and for each level of grey from black to white (0 to 255) the number of pixels holding that value is counted. For an image of 256 grey levels the result is an array of

256 numbers representing the number of pixels with each shade. This algorithm may be applied globally to the whole image or to any select group of pixels.

Mathematically the histogram is defined by:

$$\sum_{g=0}^{255} h(g) = n$$

where h(g) is the histogram of n pixels in the image (or part image) of intensities 'g' from 0 to 255 shades of grey. This information can be used directly for feature extraction or further image enhancement.

(1) Contrast enhancing

O'Gorman [29] defines a method of contrast enhancement which ensures that the image exhibits the optimum range of intensities, such that the full grey scale is utilised without saturating beyond black or white. This technique is suitable for most images, but when working with specific images, perhaps those which have a limited range of hues of a small band of shades, prior knowledge may lead to the histogram information being used to extend the contrast over the narrower range or compress it in another area where the range of shades is greater.

(2) Threshold values

Often the histogram information proves useful when choosing values for thresholding the image. In the survey of thresholding techniques by Sahoo *et al* [30] a number of useful methods for the application of histogram information to the selection of threshold values are discussed. Beattie [31] extends the problem of thresholding an image to consider the problems of noise superimposed on the image.

(3) Feature rejection and extraction

A knowledge of the histogram of a specified image can lead to the use of this information with subsequent images for the selection of areas which are background and not needed for processing. Removal of such background or fixed features, prior to the application of other algorithms, such as for edge detection, reduces the number of features which need be considered. For example, in the case study of the measurement of Ferroxcube cores in chapter five, it is known that a dark object appears on a light background and thus the background can be blanked to black or white, that is assigned to a known value, which enables easier identification of the object.

3.1.1.2 Thresholding

Thresholding is a technique which divides the image into regions of similar shade in order to simplify the image and make processing easier and quicker. Data is lost following a thresholding process, but when the result is used with reference to the original image it can produce a faster solution to the processing problems with the detail restored. The thresholding technique can use a fixed threshold value, above which all grey levels are changed to white and below which to black and this threshold level may be predetermined or calculated by a histogram method over the whole screen, an area of it, or a line. There may be more than one fixed threshold levels assigned to show several bands of shade, commonly, background and different image features.

Variable thresholding is a technique which allows the threshold values to change over different parts of the image and these can be calculated dynamically as the threshold algorithm is performed. For example, if object pixels have a brighter shade than the background a value would be selected to distinguish between the two. However, if one side of the image was brighter than the other, possibly owing to uneven illumination of the scene, a fixed threshold may select the required features in the middle, but also selects background on one side while choosing nothing on the other. See figure 3.1 .

Thresholding is a useful method for the identification and location of features, but it can help in the detailed assessment of the true shape of an object when used in conjunction with the original image. The thresholding technique loses data which can disturb the true shape of feature and it must be used with care. The paper "How grey scale processing improves vision system performance", James [32] discusses the loss of accuracy which is incurred during thresholding.

3.1.1.3 Smoothing or averaging filters

Software filters of this kind are useful for the reduction of noise or unwanted detail in an image and are low pass filters. The filtering is performed by calculating the new value for a given pixel from the current values of its neighbours. It is common to calculate the new value from the eight pixels surrounding the one under test and its own value. These nine pixels are said to be in a 3 x 3 window. The table below shows how the average could be performed.

A	B	C
D	E	F
G	H	I

If the pixel locations (A, B, ..., I) have intensities (I_A, I_B, ... I_I) then the new value I_{EN} for pixel E would be given by:

$$I_{EN} = \frac{(I_A + I_B + I_C + I_D + I_E + I_F + I_G + I_H + I_I)}{9}$$

This type of notation for image processing filters is more often shown with the weighting factor affecting each pixel. For an even, or unweighted, average filter this would be shown as:

$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$

The general form of this is given by:

$$f_n(x,y) = \sum_{i-1}^{i+1} \sum_{j-1}^{j+1} f(i,j).w(i,j)$$

Where "f_n" is the new image, formed from the present image "f()" multiplied by the weighting factors "w()" and averaged over a window i x j.

The danger when using this type of filter is that the image can be blurred to the extent that the feature which is being sought can no longer be found. In order to overcome this problem many software smoothing filters have been considered which will preserve edges, but smooth out single pixels and contours. The simplest smoothing filter involves replacing the value of the pixel in question with the average of itself with its near neighbours. It is advisable to create a new smoothed image from the results rather than to replace the pixel value on the original image as this will affect the calculation for the next pixel.

A weighted average gives extra value to the current pixel, thus reducing the degree of smoothing by increasing the "weight" of its value in the average calculation. The principle can be further extended to weight each of the pixels in the filter.

Filters which preserve edges operate by replacing the current intensity value of the pixel being modified, by the median value (or the average of the values around the median) of the intensities surrounding this pixel. The simplest method is to choose a 3 x 3 window centred on the pixel under revision, find the median of their nine intensities and replace this value as the new one. A more complex version of this involves taking the average of the values of a number of neighbouring pixels whose values lie near to the median of those of interest.

Looking at a 3 x 3 window centred on the pixel with intensity 14 next to the boundary between the light and dark areas, the new value would be calculated as follows:

8	8	240
12	14	251
9	7	247

In order of magnitude the intensities are:

7 8 8 9 <u>12</u> 14 240 247 251

giving a median of 12

For an averaged median the three values close to the median may be used giving the new resultant intensity as 11:

$$(9 + 12 + 14) / 3 = 11 \text{ (when truncated)}$$

3.1.1.4 Edge detection and enhancement filters

Edge detection and enhancement filters perform an opposite function to the smoothing filters, described above, operating as high pass software filters and have the effect of highlighting edges or boundaries between light and dark areas. The simplest edge detection filter is to equate the gradient of shade in two dimensions about the pixel, by finding the difference between the neighbouring pixels and ignoring the direction information. This would appear as:

	p(i,j+1)	
p(i-1,j)	p(i,j)	p(i+1,j)
	p(i,j-1)	

where the new pixel $p_n(i,j)$ is calculate from the gradients in the x and y directions from:

$$\begin{aligned} p_{nx}(i,j) &= p(i+1,j) - p(i-1,j) \\ p_{ny}(i,j) &= p(i,j+1) - p(i,j-1) \end{aligned}$$

These equations are more generally written as:

$$\begin{aligned} p_x(i,j) &= win_x \ast p_x(i,j) \\ p_y(i,j) &= win_y \ast p_y(i,j) \end{aligned}$$

where win is a window describing the multiplication factors to produce the right shape window or any weighting desired and $\ast$ represents the sum of products of the terms. The window (win) for the above equations is:

$$win_x = -1\ 0\ 1 \quad \text{and} \quad win_y = \begin{array}{c} -1 \\ 0 \\ 1 \end{array}$$

It is common to describe the operation of finding the difference between pixels as a summation with the window providing the sign for the subtraction to find the difference. Moving across an edge the gradient will start at zero, increase to a maximum and then return to zero. This technique produces broad edges which usually need to be further thinned down. A better method, producing thinner edges, is to keep only the pixels which are at the maximum gradient values, in their direction, and dynamically thresholding (See section 3.1.1.1.(2)) the resultant pixel values to create the new image. This however, is more involved and slower as it necessitates looking at the pixel values after the initial edge enhancement to remove those which are not a maximum.

The magnitude of the gradient at the pixel in question, $p_n(x,y)$ is then calculated by:

$$p_n(i,j) = \sqrt{(p_x(i,j)^2 + p_y(i,j)^2)}$$

The direction given by:

$$p_{nd}(i,j) = \tan^{-1}(p_x(i,j) / p_y(i,j))$$

Two other commonly used edge detection filters are the Sobel [33] and Laplacian [16].

The Sobel filter is a more complex gradient operator of the type described above, having windows of the form:

$$\text{win}_x = \begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix} \quad \text{and} \quad \text{win}_y = \begin{matrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{matrix}$$

The Laplacian when applied to an image is defined by:

$$\nabla^2 f \equiv \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

when considering the image as a two dimensional function f(x,y); Niblack [16] shows that when applied to a 3 x 3 window the centre position is calculated from the window:

$$\text{win}_x = \begin{matrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{matrix} \quad \text{or} \quad \text{win}_y = \begin{matrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{matrix}$$

3.1.1.5 Motion compensation

Applying vision systems to problems where the object under test is moving presents a number of difficulties. There are essentially four types of problem which can be categorised as:

- ❑ Blur from camera's latency
- ❑ Blur from camera's horizontal pixel scan
- ❑ Bending from vertical line scan
- ❑ Double image from interlaced image capture

(1) Blur from camera's latency

There is a description of how vidicon and ccd cameras function in section 2.2.3. This explains that it takes a finite time, depending on the type and make of camera, for each pixel, or camera cell, to respond to the amount of light shining on it. Consequently, when a boundary between light and dark areas moves across the field of view it gives an intensity between the light and dark values and appears blurred. The width of the blur is dependent on the speed of the boundary and the latency of the camera.

(2) Blur from cameras horizontal pixel scan

Pixel scan blur occurs in two ways, depending on whether the direction of pixel scan is in the same direction as the moving object, so that one overtakes the other, or that they are moving in opposite directions.

The object moves across the camera at a rate which may, or may not, be constant over the whole field of view. This means that the camera scans forward by discrete, finite steps with pauses between. Thus the same point may appear in more than one pixel as the pixel scan tries to overtake its motion, and may be missed completely if they are moving in opposite directions. This may be termed "pixel scan blurring" where both motions are in the same directions and the same point in the object may be read several times, and "lost detail blurring" where points are missed.

When a camera scans an image it will pause to look at a pixel for a finite period and measure the average shade value during that period before moving to the next. Since a single point in the image may be read by the camera several times in succession, and the pixel being read will change during the period that it is "exposed" then the average value will be affected.

Sharp edges will show blurring and the measured object will appear wider, on the image, than its true stationary width if the object movement and camera scan direction are the same, and will be thinner if the directions are opposite.

There are several cases to be considered:

- ❑ the speed of motion of the image is much greater than the horizontal scan speed of the camera and the moving object could be missed completely and would only affect pixels in a single line,
- ❑ the object's speed is similar to the horizontal scan speed, again only a single line would capture the object,
- ❑ the scan speed is greater than the image speed, where the differential speed is sufficiently great to produce a usable image.

Neither of the first two situations need further consideration since no useful image is obtained. The final case is considered further, but for this effect to produce more than a one pixel error on each line the object must move more than 512 pixels in the field time of 20 ms. Practice shows that movement of this degree does not produce a usable image and the amount of movement tolerable in an image is in the order of a few pixels from the first to last line.

(3) Bending from vertical line scan

Bending of the moving image, which can be called "line scan blurring" is due to the transverse component of the relative velocities of scan and object motion.

The problems of blurring and bend are well known to photographers of moving objects and are most simply seen when using a still camera with a focal plane shutter which moves vertically when capturing a horizontally moving straight edged object. The photographic result is a curved image because one end of it is "exposed" at a later time than the other. The effect is often to be seen in the photographs of sportsmen, such as the swing of a golf club, and rotating aircraft propellers. The degree of "bend" depends on the relative velocities of the shutter and moving object. The relative displacement affects all points in the image, but is most noticeable at the leading and trailing edges, and of course where the object is moving fastest. Thus rotating objects, like a propeller or a golf club bend more markedly at the outside end, and in those positions where it is moving at right angles to the direction of shutter movement.

In the case of video cameras bending is a consequence of the line scan time of the camera as well as the speed of the object and is, therefore, a fundamental problem. Here the problem is considered only in the special case of constant speed linear motions which are mutually at right angles.

(4) Double image from interlaced image capture

In the case of interlaced video cameras, which scan alternate even lines and then intermediate odd lines, there is a separation of the images in the two sets of lines. The degree of separation will depend on the relative velocities of the object and the time interval between the two sets of lines. It will occur at all points on a moving object and is seen in addition to the curve most noticeable on vertical edges. Here again it is discussed only in the specially simple case of constant speeds of line scan and moving object which are assumed to be mutually at right angles, but of course if the picture included two, or more, objects moving at different velocities, or rotational or reciprocating motion the algorithms needed to cope with it would be that much more difficult and complex.

(5) Solutions to motion problems

The problems outlined above do not appear alone, but as a combination; however, some of them may be ignored since they become inconsequential compared with the more pronounced affects of others. An awareness of the problems is important so as to know the limits of the system and what degree of motion can be tolerated.

The most visible feature in an image of a moving object from an interlaced camera is a double image. This exhibits itself by the image appearing to move as the odd and even fields are alternately displayed. This can be compensated for in software by regarding the two fields separately and moving them back together by the difference, in pixels, which they are apart. If this offset is constant it need not be calculated each frame by software, but can be found by looking for the difference at a known edge. As the image becomes more complex, or the object is moving over a fixed, but variable background, the difficulty becomes greater.

The bending of objects from vertical line scanning can be compensated for in software, either with prior knowledge of how much to shift each line, or by knowing the shape of the object and shifting each line in a linear or geometrical manner to fit the shape.

Blur on an image can be improved by application of an edge enhancing filter of the type discussed in section 3.1.1.4.

(6) High speed shuttered ccd camera

A more reliable and possibly cheaper solution to these problems is to specify the use of a high speed shuttered camera such as the Pulnix TM-765 [14]. Cameras of this type can operate with exposures as short as $^1/_{30000}$ second, but cost well over twice as much as an unshuttered camera. When the cost of software and additional processing is added to the unshuttered camera this is likely to prove more than for a shuttered camera. The use of shuttered cameras is therefore the realistic solution to motion problems, where there is adequate lighting.

(7) Example

Figure 3.2 shows the vision system arrangement and the images produced by a straight bar passing under interlaced and non-interlaced cameras which are viewing items on a moving conveyor. If the conveyor is moving at constant speed, then knowledge of this could be used to shift the lines back into position. Further correction of the lines can then be performed by applying some prior knowledge of the shape of the object.

The arrangement of the camera and bar as described highlights the problem, but if the bar passed in a different orientation or the camera was rotated, the bar may simply appear shorter, longer, thinner or fatter. For a counting operation this is not important, but if the inspection required a measurement of its dimensions, this would be significant.

3.1.1.6 Constant background feature removal

Given suitable lighting and constant gain, background features, which are fixed and always appear in the same place with the same intensity, can be easily removed by image subtraction. Figure 3.3. An image which only has the background is stored and subtracted, pixel by pixel, from the image with the feature.

A better implementation of this principle involves a comparison of each pixel to determine whether the pixel is showing something in front of the background (ie the two values are significantly different) and no subtraction is required. Only the differences greater than a threshold set to eliminate noise are displayed.

3.1.2 Engineering algorithms to suit the need

When considering a solution to an industrial image processing problem reliability is the paramount consideration. If the software will not always find the right result from the specified image, the system does not inspire faith and will not be used with confidence. The lesser considerations are those specified by design requirements such as speed, image specification, hardware and cost. In order to find a solution that works "on site", engineering compromises are made and cut-down versions of the described algorithms are developed and combined with hardware and optical aids.

When choosing a solution the first consideration should be to establish hardware and optical arrangements which will simplify the software. The choice of lighting used in the scene, described in section 3.1.3.1, and from which angle it is to be viewed both influence the software. Setting the object against a plain background will increase the chance of finding the feature correctly. If this cannot be done, perhaps the use of a monochromatic light source to illuminate the object and a matched filter to remove the background may be more reliable than subtracting a 'busy' background by the software. This method of using a monochromatic light source with an optical matched bandpass filter to highlight the feature from the background is used in chapter four to provide cross-sectional information about a weld's profile while removing the background.

An important first question when inspecting components on conveyor lines is, "Are the components stationary at any time?" If a new line is being installed the answer may be "Yes, they can be." Stationary objects are much easier to measure as no compensation is required for blur of the image and movements between frames. Thus hardware can reduce the amount of processing involved. But if the line will not permit the objects to be stationary then the use of fast shuttered CCD cameras may reduce any movement to an insignificant level. Knowledge of the speed of the line is essential when specifying either hardware or software, but the amount of blur of a fast moving conveyor line can be reduced by using a shuttered camera and the residual blur further compensated by the software so that an accurate and reliable solution is obtained. It is likely that the speed of fast shuttered cameras, such as the Pulnix TM-760 [14], with speeds as fast as $^1/_{30000}$ sec will produce an image with imperceptible movement.

In order to improve the speed performance of the software after any hardware solutions have been agreed, most of the general methods (discussed in section 3.1.1 and detailed in Castleman [34]) for global application to the whole image can be implemented either to known areas of the image or single lines or pixels as required. A complete understanding of both how each particular algorithm works and the principles behind the general behaviour of each type of algorithm is necessary, as it is often better to develop new algorithms peculiar to the problem being tackled.

3.1.3 A knowledge of the hardware limitations

In order to write fast and reliable image processing software one must have a thorough appreciation of the working of the hardware from the optics, through the camera and digitising circuitry to processing and display. The software engineer writing image processing algorithms who is armed with this knowledge is able to specify aspects of the hardware design to ensure that the software can assume certain characteristics and accommodate problems which cannot be avoided.

Most of the problems encountered in image processing can be solved in software alone, but usually at the expense of speed. The wider the scope of variation in the image with which the software must cope, the more checking and processing of the image is to be done and the more time required. There are many things which can be specified in the hardware design to aid the software and reduce the processing. The cost of a hardware solution, or hardware assistance, is often far less than the cost of the increased computing power needed to meet the demands of the more complex software, as well as the cost of writing it.

The fundamental constraint of image processing is said to be "if you cannot see it, you cannot process it", but there is a range of degrees of difficulty between seeing an object and not seeing it. The more distinct or consistent the feature is, the quicker it can be found and processed. It is common to use combinations of both hardware and software techniques which are described below to make the features very clear and consistent.

3.1.3.1 Lighting

In the same way the lighting of scenes and objects for photography is critical to the desired result, it is a factor of paramount importance in capturing good images for processing. Some scenes or objects, may require completely even lighting, others illumination from one side to cause reflection or shadows to highlight the features, or use of monochromatic light with an enhanced colour balance.

The use of lighting powered by direct current which gives a constant illumination output is preferred, especially when using CCD cameras. Vidicon cameras, such as the Hitachi [35], have a high image retention and can smooth out the fluctuations of the light intensity produced by a.c. powered lights. The problem is most apparent using CCD cameras, such as the Pulnix [14], in which the lighting of the whole image can be seen to "beat" when viewed frame by frame on a video recorder. The "beating" image occurs because the camera sample rate and mains frequency are both about 50 Hz. The interlaced camera samples all the odd lines followed by the even lines and thus the base intensity of each line varies over the two fields (as described in Section 2.2.2.3.(4)). Using this knowledge the problem can be alleviated in software; however, the use of d.c. lighting is a cheaper and more reliable solution. Figure 3.4 illustrates the effects of a.c. lighting.

The use of laser light is often advantageous. Laser diodes produce a divergent beam of coherent light of a single wave length and are very small and inexpensive. Consequently, it is easy to remove much of the unwanted background detail by use of matched band pass optical filters while concentrating on the areas illuminated by the laser. The use of lasers has proved most successful in the solution of the weld measurement problem discussed in chapter four, as well as many applications of seam tracking where the laser light is focussed into a line perpendicularly across a weld seam to highlight the gap between the plates prior to welding.
Figure 3.5 shows the recorded intensity profiles across a seam with both even illumination and laser stripe illumination together with the viewed images. The laser method has proved to be more reliable in operation as areas of lower reflectivity on the metal surface do not affect the gap in the laser line, whereas, under natural light, such patches can cause a lower intensity than the seam and, therefore, confuse the program.

Laser light illumination in the above example has been used simply to amplify a feature, but laser stripes have a further application which is to allow the cross-section profile to be measured and the information stored. Viewing a surface illuminated by natural light gives no measurable information about the surface, but shining a fine laser line across it and removing unwanted background, highlights the surface profile detail along that line. In chapter four, the case study of weld measurement discusses the extraction of numerical weld profile information.

The techniques of using laser light or natural light need not be used in isolation, but by careful arrangement, two independent algorithms working on the same image can aid each other to produce better results.

3.1.3.2 Gain control

Varying the gain of the image adjusts its brightness. Thus, when viewing a bright image it may be beneficial to reduce the gain so that the areas which are white have more grey scales to differentiate between the various shades of white. This adjustment may prevent the image from saturating. Of course, the converse is true when viewing a dark image. Most cameras and some frame capturing boards incorporate automatic gain control which electronically adjusts the gain to make the average intensity of the whole image appear as mid-grey (128 on a scale from 0 to 255), and this can cause a problem when the image is to be processed.

A standard method of feature extraction is to remove the constant background from the image being processed (see section 3.1.1.6), but this only works if the stored "background only" image had the same illumination level, gain and contrast as the image from which it is being removed. That is, the background levels of both images must be the same. If automatic gain control circuitry was active and the background was dull with the feature having a lighter shade, the average of the picture would have changed and the background levels would have been assigned different values. For example see figure 3.3 .

3.1.3.3 Image noise

The appearance of noise on an image originates from many different sources, however, it is not consistent from image to image. If the scene is constant, several frames may be taken, compared or averaged to reduce the effect of the noise or remove it, Castleman [34].

3.2 Sequential and parallel processing

Most processors and programming languages operate by taking one operation, processing it, and then moving on to the next. Thus, in order to achieve any processing goal, it is necessary to decide in what order operations must be done, "this" first, "that" second and so on. In parallel programming, as with all natural events, several, or even many, operations may be happening at once. For example, if a program requires three main operations to function, one of which relies on results of the other two, but they need no further information than they have at the start, the first two operations may be executed in parallel, simultaneously, and consequently achieve an increase in system speed. See figure 3.6 .

Since the third operation requires the results of the previous two operations (1 and 2), it cannot start until they both have finished. If the program was running continuously and each operation takes the same time to execute, the results would be delivered after every three operational periods for the sequential program, and after only two operational periods in the parallel programmed case. While the delay period cannot be further improved with the use of more processors, the sample rate could be increased to give a result every operation period by using the three processors so that the operation is "pipe-lined". Now, while operation 3 is executed, operations 1 and 2 can be calculating results for the next cycle. The timing of this is shown in figure 3.7 .

Multiple processing of the same image in parallel presents the programmer with some special considerations. It is essential that one algorithm does not change the image so that another operates on incorrect or modified data. For example, if one algorithm is drawing its results back in place of the original image, to superimpose the two, and the second algorithm operates on this combined image, then the second is likely to produce inaccurate, if not wrong results.

Parallel processing provides a more natural solution with the ability to speed up delivery of the result by using more processors. A good example of this is the algorithm to find a laser line across an image. Sequentially this would be done by starting at column 0 and finding each point across to column 511 (for a 512 x 512 resolution image). To speed up the process the program could be put into parallel with each point of the line found simultaneously on an increased number of processors.

Example of a sequential algorithm to find a laser stripe:

```
SEQ x = 0 FOR 512
    ... find laser line for this column.
```

To speed this up using more processors can only be done using a parallel construct with each part in sequence.

```
PAR
    SEQ x = 0 FOR 256              Each sequential part would
        ... find laser line            be placed on its own
    SEQ x = 256 FOR 256                processor with access to
        ... find laser line            the image.
```

By using only a parallel construct, the split is more natural.

```
PAR processor = 0 FOR number of processors
    ... PAR x = processor * (512 / Number of processors) FOR (512 / Number of processors)
        ... find laser line
```

This can now be changed for more processors by changing the variable number of processors.

3.2.1 Sequential programming

Popular languages, such as C, Fortran, BASIC, and Pascal allow the programmer to develop algorithms which execute one instruction at a time. It is not possible to dictate that whilst one operation is being executed another can also take place. For a given processor, the program is developed to run on it and the program can only be made to run faster by changing it. Algorithms can be made to run faster in a number of ways; by careful control and knowledge of the amount of data available, the programmer can choose faster addressing techniques, and also write code which either checks fewer parameters or is less reliable to operate in all circumstances. A knowledge of the processor's instructions may permit the programmer to write programs using a special, quick feature, but programs become over clever and difficult for others to comprehend and, ultimately, when the program will not produce the results fast enough, a faster single processor has to be sought.

3.2.2 Parallel programming

The parallel programmer may develop programs more logically with thought spent on functionality and reliability rather than clever techniques to use a quirk of the selected processor to achieve the desired speed. The transputer [27], when programmed in Occam, offers the full potential of developing parallel algorithms at all levels. This allows the programmer to use parallel, or concurrent, constructs and algorithms on one processor, by dividing the program up for final implementation on several processors to achieve the required speed.

3.3 The Occam programming language

The Occam programming language is a high level language for which the Inmos transputer has been designed. Occam was conceived in 1982 by David May and Tony Hoare and is now supported by Inmos Ltd as the primary language for the transputer. Occam is designed to express concurrent algorithms or processes at all levels of a program and to enable the compilation of these processes on one or more processors. Concurrent processes may not share common data or variables, but pass information between them by channels.

3.3.1 Occam Channels

Channels provide a point to point communications method between two processes. Process 1 will transmit a piece or pieces of data down a channel and will wait until the receiving process takes it. The data must be received in the same order as it is transmitted. These channels can exist between two processes on the same transputer, or on two processes on two separate transputers by mapping the channel to one of the transputer's links. (See section 2.4.2.1 The Inmos Transputer.)

3.3.2 Occam Constructs

Occam offers one of the simplest programming languages to learn as there are only six constructs.

SEQ	Sequence
PAR	Parallel
IF	Conditional
CASE	Selection
ALT	Alternation
WHILE	Loop

3.3.2.1 SEQ

Code following a SEQ will be processed sequentially. For example, in the program below four numbers must be added to produce two results C and F as follows:

```
A + B = C
D + E = F
```

This could be written in Occam to process sequentially as:

```
SEQ
    C := A + B
    F := D + E
```

Assuming A, B, D and E were defined, the results F and C would be produced at the end of the program by equating C first then F.

3.3.2.2 PAR

Code following a PAR will be processed in parallel. For example in the above program where four numbers are added this could be executed as:

```
PAR
    C := A + B
    F := D + E
```

However, in this case, the results are produced simultaneously. The process of adding A to B to produce C and adding D to E to produce F are running concurrently. It is now that the power of the transputer to speed up processing can be seen. In the first example, using the SEQ construct the process cannot be speeded up as one sum is calculated before the next, but in the second example the two processes could be placed on separate processors and then the time taken to calculate the results of both equations would be halved.

3.3.2.3 IF

The IF construct behaves in the same way as in most computer languages, offering the ability to choose which process to perform dependent on some programmed condition. Using the above example either of the sums can be performed dependent on another variable, which may, or may not, be used in the sum, "CALCULATE_C".

```
IF
    CALCULATE_C = FALSE
        F := D + E
    CALCULATE_C = TRUE
        C := A + B
```

At the end, the result for either F or C is calculated dependent on whether "CALCULATE_C" is TRUE or FALSE.

3.3.2.4 CASE

The CASE construct is similar to the IF construct, but it offers a number of options from which to select the action to be performed. An input variable to the CASE construct is compared with those given in order to choose which set of operations is to be performed. In the example below, objects are to be sorted by their colour. The previous part of the program has already determined the colour of the object.

```
CASE colour
    red
        ... put object in red bin
    green
        ... put object in green bin
    blue
        ... put object in blue bin
    black
        ... put object in black bin
    white
        ... put object in white bin
    TRUE
        ... failed to meet colour test
```

3.3.2.5 ALT

The ALT construct allows the program to perform which ever operation is triggered first from a channel. For example, three systems spotting different items on a conveyor could advise another piece of software every time each one sees one of their designated items in order that it may be counted. The ALT construct would allow these to be randomly detected.

```
ALT
    system_1 ? another_item
        ... increment the number of item 1 counted
    system_2 ? another_item
        ... increment the number of item 2 counted
    system_3 ? another_item
        ... increment the number of item 3 counted
```

3.3.2.6 WHILE

A loop can be set up using the WHILE construct so that a piece of software will keep being run repeatedly, until a certain parameter is satisfied. A system, which is making some component, may be required to stop production after 100 have been made; this could be implemented by:

```
WHILE (total_components < 100)
    ... make another component
```

3.4 Parallelising image processing algorithms

When re-writing an existing sequential algorithm, in order to parallelise it, or writing a new one for parallel operation, some results are required from one process before the next process can continue. (See description in section 3.2). Bearing this principle in mind, most image processing operations can be implemented in a parallel fashion although care is needed to ensure that the parallel version produces identical results to that of the sequential one.

3.4.1 Thresholding

A threshold can be easily implemented in parallel by replacing the sequential code directly with parallel code; this can then be transported to as many processors as required.

For example, the code in a sequential program might be:

```
SEQ x = 0 FOR 512
    SEQ y = 0 FOR 512
        ... threshold this pixel at location (x,y)
```

In parallel Occam this becomes:

```
PAR x = 0 FOR 512
    PAR y = 0 FOR 512
        ... threshold this pixel at location (x,y)
```

In the parallel algorithm every 512 x 512 (= 262 144) pixels are thresholded simultaneously. This could be executed on one transputer, but if so, it would take longer than the sequential version of the same algorithm, since the extra time is accumulated by the time taken to switch between all the parallel tasks. However, as soon as the parallel algorithm is implemented on more than one processor, an increase in speed is achieved. In order to make the program run on "n" processors the above algorithm would be modified as below and defined as a procedure.

The Occam code could be:

```
PLACED PAR this_processor = 0 FOR n
    PROCESSOR this_processor
        threshold_column( (this_processor * (512 / n)), (512 / n) )
:
```

where the procedure "threshold_column()" is:

```
PROC threshold_column(start_x, column_width)
    PAR x = start_x FOR column_width
        PAR y = 0 FOR 512
            ... threshold this pixel at location (x,y)
:
```

And "this_processor" is 0, 1, 2, ... (n-1), given n processors up to 512. This will result in each processor thresholding a vertical column of the image which is (512 / n) pixels wide.

3.5 Processing after enhancement and feature extraction

In engineering, image enhancement leading to the extraction of a feature is not the final goal of an image processing system, some other result is required. Acquiring the raw information about the feature is only a stepping stone leading to the computation of the results. The image processing may highlight an object in the picture and produce its coordinates in terms of the image, but it may also have to be measured, identified and sorted. Information may then be passed on to another system, such as an industrial robot, to form the basis of the task with the object being viewed.

In an automated industrial system which uses digital image processing to provide a vision sensor, the software must produce a result, which may be a command to "pass" or "fail" an item, which could be indicated by a logic signal sent to another system. In this case the "robot" then either does nothing or takes an appropriate pre-planned action such as removal of the component from the conveyor line. Alternatively, the result might be a direction given as "up" or "down", "left" or "right", rotation about alpha, beta or gamma axes with its corresponding magnitude. Whatever the result, it must be correct, accurate, of the specified precision and at the right time.

3.5.1 Measurement

In its most basic form, the measurement of a feature using image processing involves counting the number of pixels comprising the feature and multiplying by a calibration factor to convert the result into a known usable scale. Castleman [34] discusses measurement and assessment of shape in more detail.

Pixel counting provides the simplest type of measurement, but the calibration of pixels must be determined at an early stage. Consider the feature A (in figure 3.8) where the distance between the dots in millimetres is the dimension required. It is also known that they are each always only a single pixel in size and are horizontally orientated on the image. Their distance apart would be calculated by summing the number of pixels between them plus one pixel and multiplying the result by the horizontal size of one pixel. If feature A was rotated such that the pixels did not lie on the same horizontal or vertical lines, it is necessary to measure the horizontal and vertical distances between them (x,y). The figures for x and y are converted to millimetres before the single angled distance is calculated by application of Pythagorus' theorem.

$$\text{Distance apart} = \sqrt{(\ (x * \text{calib\_factor\_x})^2 + (y * \text{calib\_factor\_y})^2\ )}$$

Feature A demonstrates the technique of single pixel measurement in x and y coordinates, but it is unlikely to be encountered in practice as the two dots will be made up of many pixels, each with increasing density towards the middle. (See feature B, figure 3.8) In this case, before measurement a choice has to be made from which point in each dot the measurement is to be taken. Perhaps, the pixel with maximum intensity might be chosen, but this would be liable to noise distorting error, alternatively, the image could be thresholded to give an area of significance of which the centre is found. A technique may be to calculate the centre point from the "centre of mass" of the significant shape by considering the grey scale intensity as density. Constraints of cost, speed and accuracy clearly play a major part in selecting from the choices. Once the two points are found the calculation of the dimension can be made as in feature A.

By far the most common measurements are those associated with object lines or their sides. The distance between them and their relative angles are commonly found. Assuming that the lines have been thinned down to a single pixel by a method detailed in section 4.5.1.3(3), the equation of this line can be calculated. For straight lines a successful method is to perform a least square fit, Thomas & Finney [36], of the constituent pixels, which leads to an equation of the standard form:

$$y = mx + c$$

Once these equations are calculated the measurements are easily made using standard trigonometric methods, remembering to calibrate x and y before they are combined.

3.5.2 Robot guidance

The resultant information from a vision system designed for robot guidance is likely to be an instruction to move or rotate either the robotic machine or the camera in any axis, with the extent of travel required. Again a calibration is needed. The simplest, single axis guidance, will be designed to bring the feature in the image to a pre-determined position which may be fixed and often will be the centre of the screen.

Robot guidance by a vision sensor employs most aspects of image processing and involves consideration of many difficult problems. As described in section 3.1.1.5, the elimination of motion from the image is most desirable and it is recommended that a shuttered camera is used in order to reduce the problem to an insignificant level. Speed is another consideration and the maximum and minimum times that the software may take to execute each loop must be tested in order to guarantee that the system will meet the response time required by the robot's control loop.

3.5.3 Inspection for Quality Assurance

The result from a quality assurance test may be as simple as pass or fail, but in order to achieve this result, measurement, shape analysis and tone assessment may all be required. Visual inspection performs an important part of quality inspection since it allows quick analysis of shade, colour, texture, shape and size.

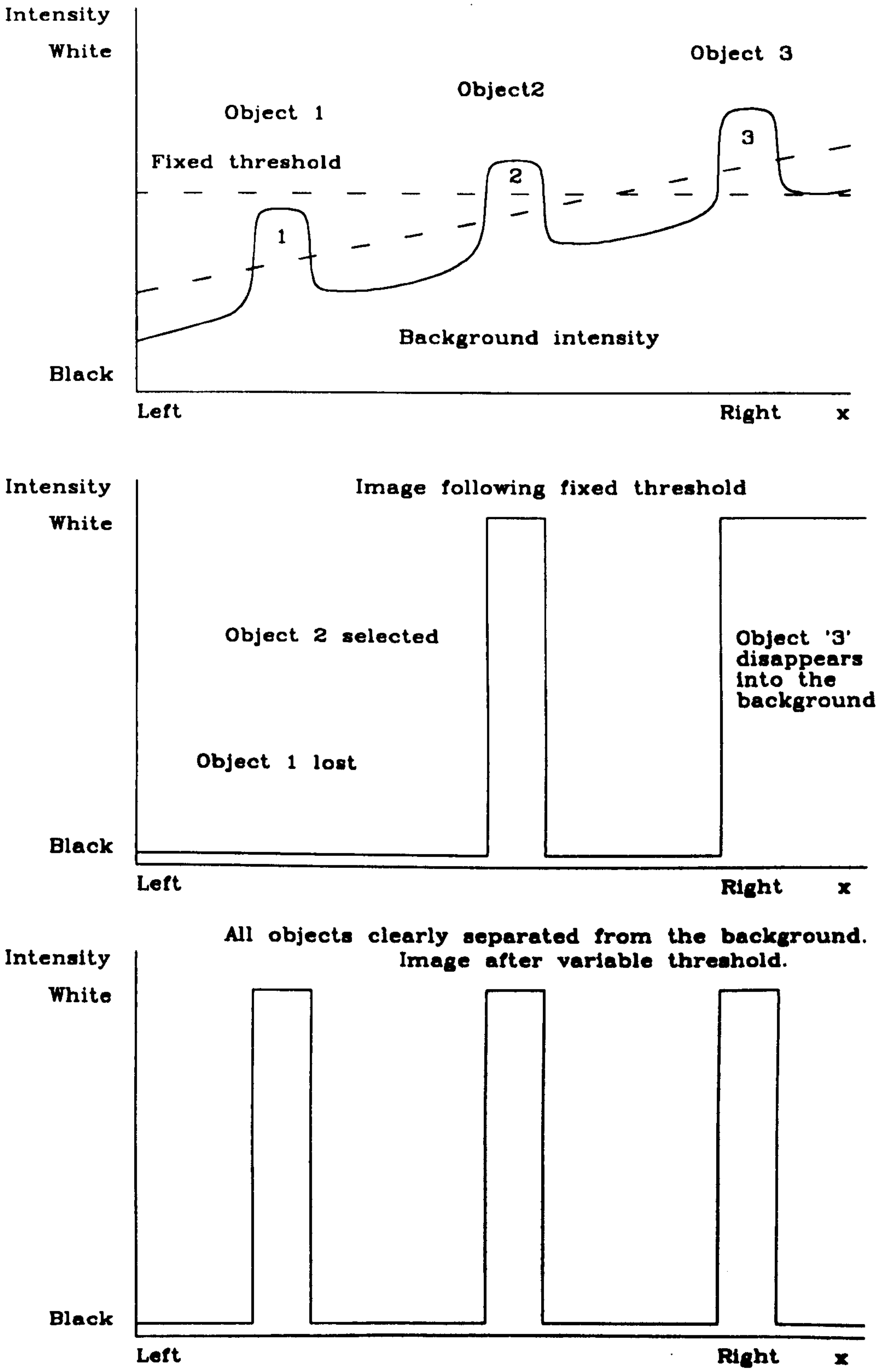
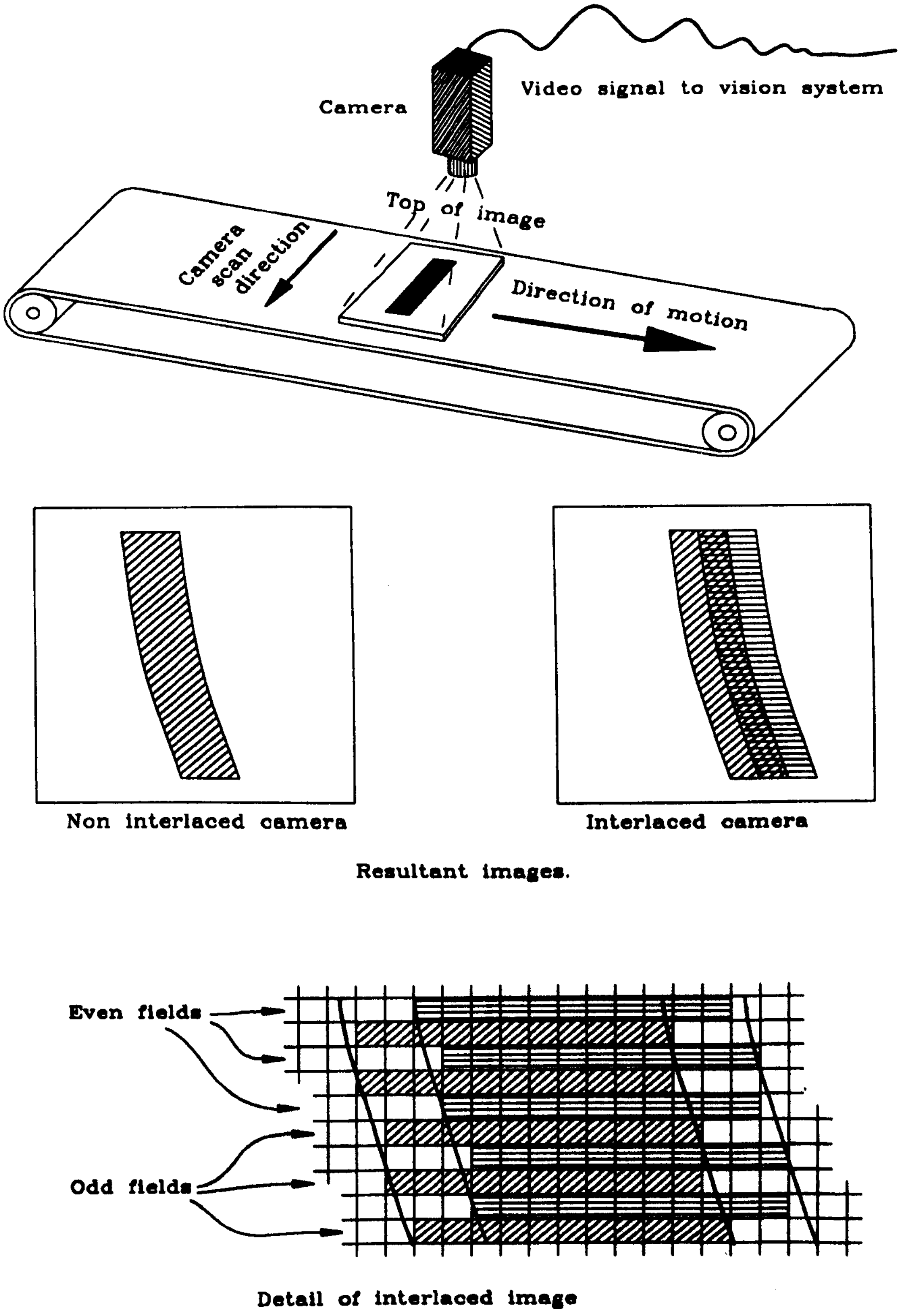


Figure 3.1 Example of fixed and variable thresholding



Non interlaced camera

Interlaced camera

Resultant images.

Detail of interlaced image

Figure 3.2 The problem of interlaced video signals with moving objects

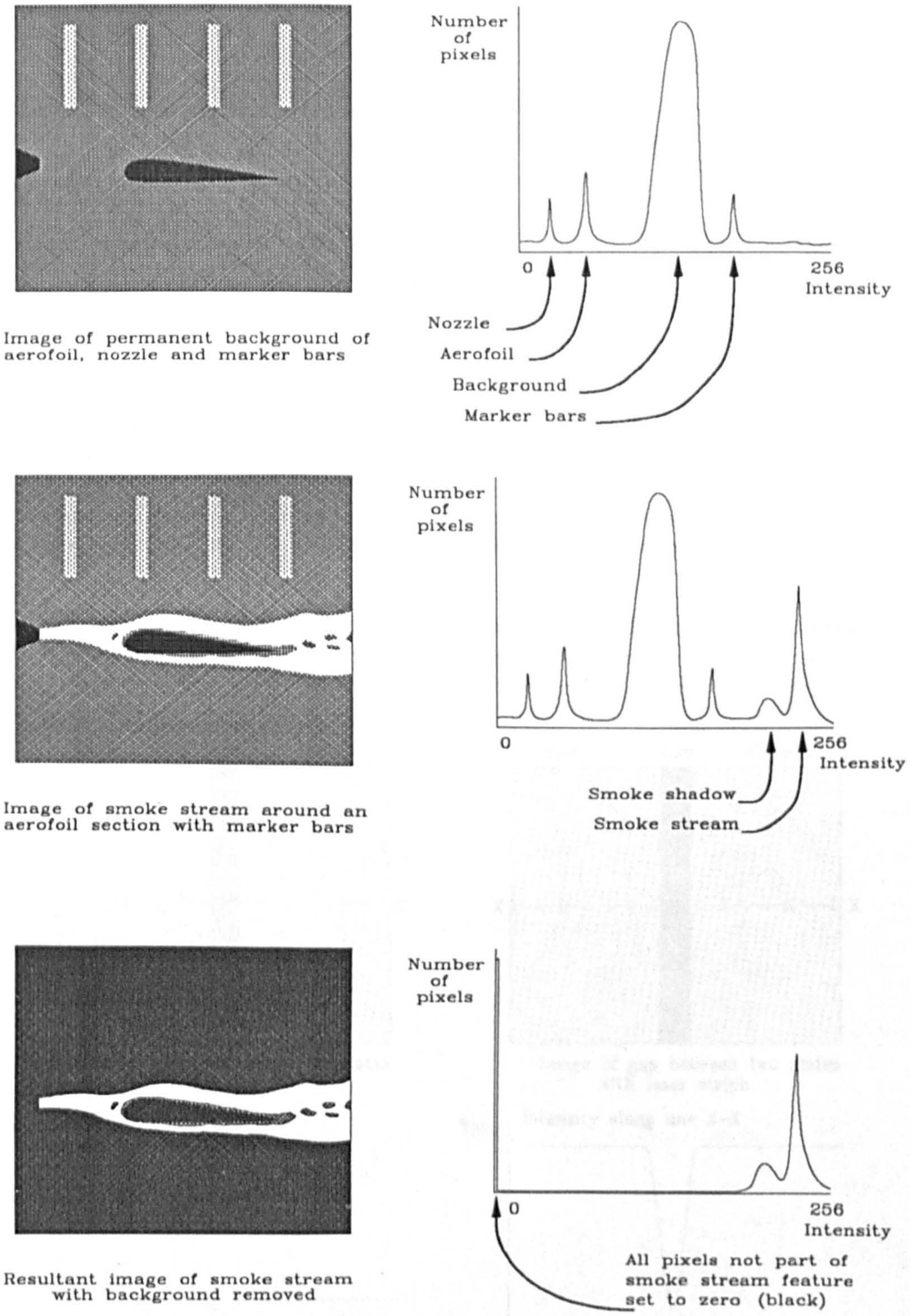
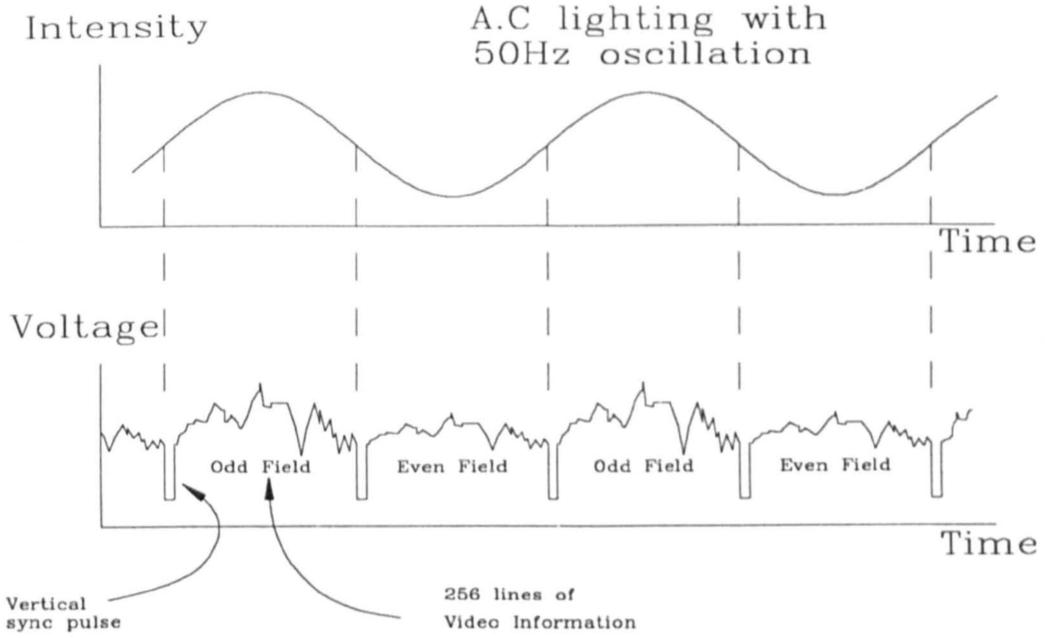


Image of permanent background of aerofoil, nozzle and marker bars

Image of smoke stream around an aerofoil section with marker bars

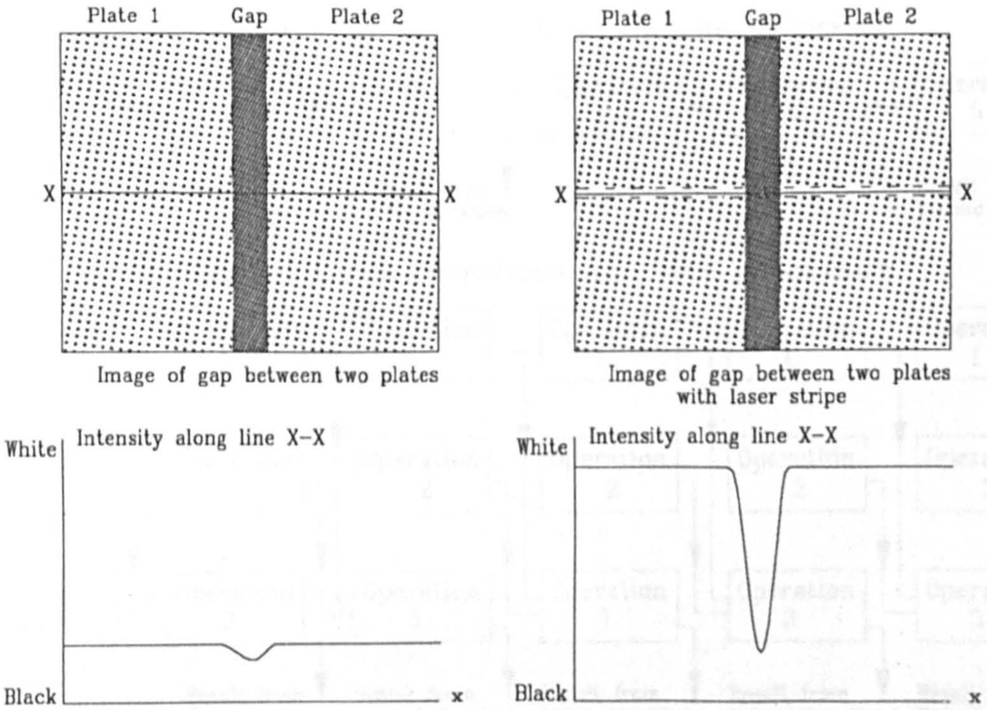
Resultant image of smoke stream with background removed

Figure 3.3 Images and histograms showing background removal



In the example above all the odd fields will be much brighter than the even fields.
The two frequencies are not locked together and will therefore drift relative to each other

Figure 3.4 The effect of a.c. lighting on video signal and image intensity



Intensity profiles across the two images at the lines marked

Figure 3.5 Laser stripe across a seam between two metal plates making the feature more pronounced for image processing

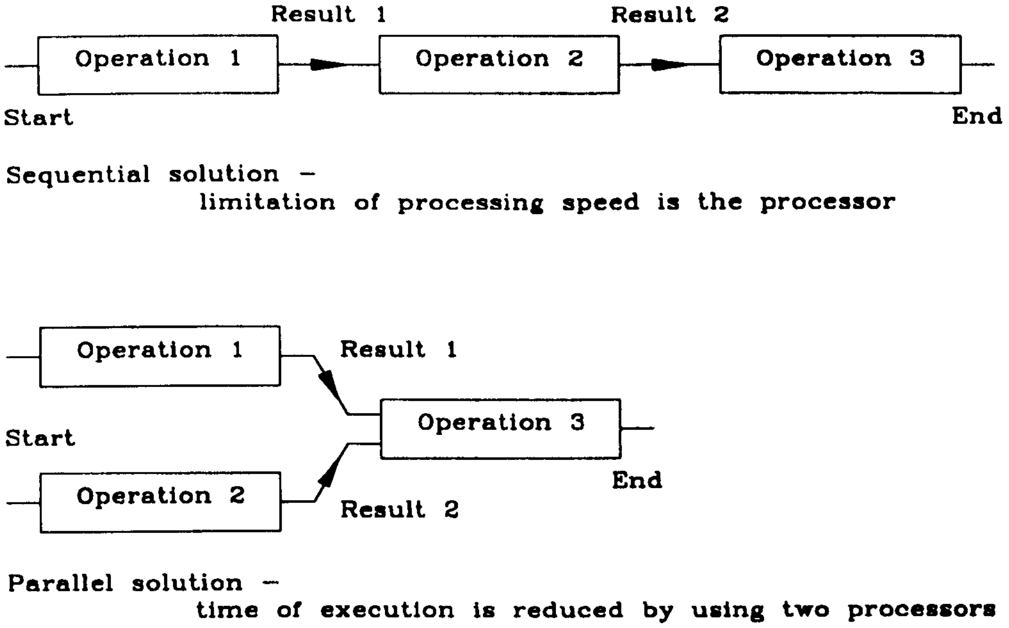


Figure 3.6 A three operation process speeded up by using parallel processing to perform the first two simultaneously

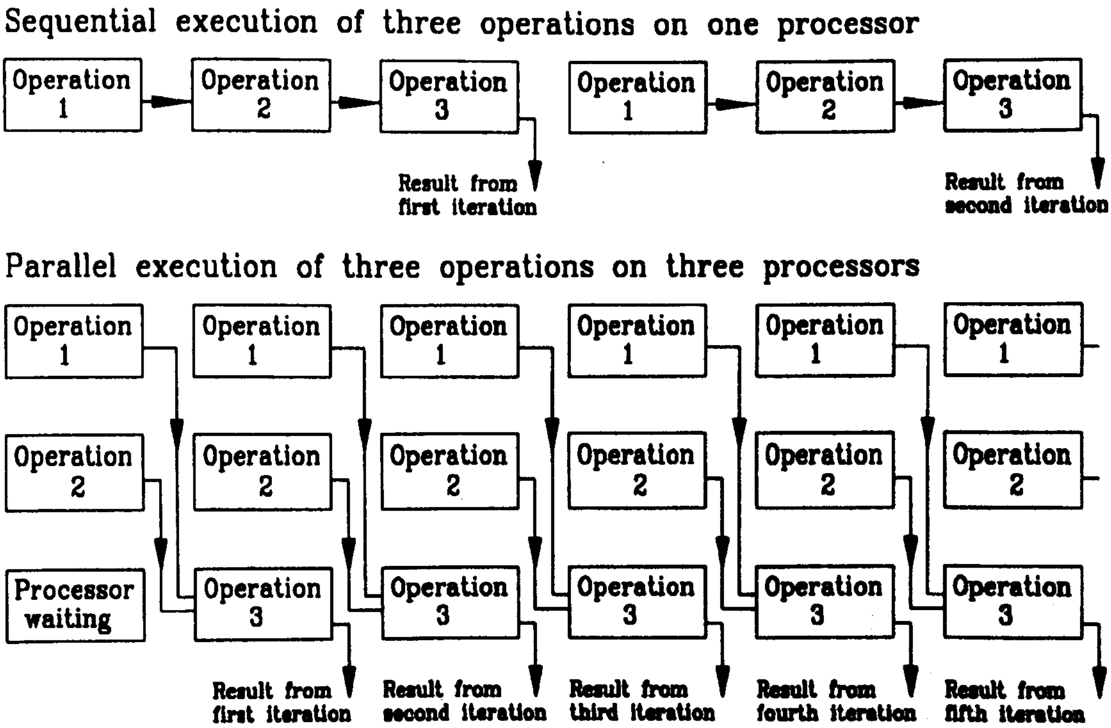

Sequential execution of three operations on one processor
Operation 1
Operation 2
Operation 3
Operation 1
Operation 2
Operation 3
Result from first iteration
Result from second iteration
Parallel execution of three operations on three processors
Operation 1
Operation 2
Processor waiting
Operation 3
Result from first iteration
Result from second iteration
Result from third iteration
Result from fourth iteration
Result from fifth iteration

Figure 3.7 Three processors increasing the sample rate by three and reducing the response time by a third

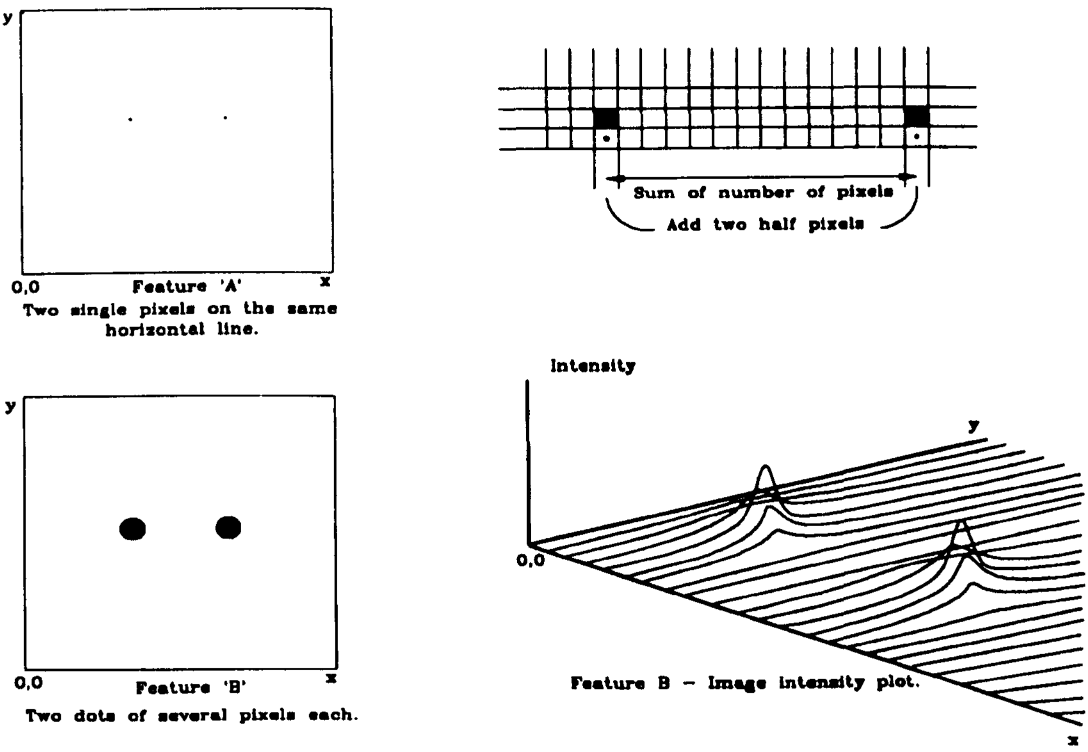

y
0,0
Feature 'A'
x
Two single pixels on the same horizontal line.
Sum of number of pixels
Add two half pixels
y
0,0
Feature 'B'
x
Two dots of several pixels each.
Intensity
y
0,0
Feature B - Image intensity plot.
x

Figure 3.8 Measurement between image processing features

4. Chapter 4 - Portable weld measurement gauge

Chapters four and five describe the research work which has been undertaken together with the corresponding results from the applications of industrial image processing. These studies are intended to demonstrate the principles outlined in the preceding chapters.

Chapter four describes the conception, design and development of an automatic, portable gauge to measure the profile of welded joints using image processing techniques. Firstly, current methods used for weld measurement are discussed and recent developments in automated welding are reviewed to show how an automated measurement tool may be applied as a sensor to welding processes. The design of a portable system to measure weld profiles is described and results from the portable gauge are shown and discussed. In addition, the same algorithms, when used on-line to ensure weld quality in a cladding process are discussed, with a view to their application in an automated sensor. Finally, future development of the gauge and other areas of applications are considered with a view to marketing the idea as a range of profile measurement sensors, from an off-line model to a high speed on-line model which inspects in real time, as well as a portable gauge for post welding measurement.

4.1 Design criteria for a new gauge

The initial design phase of a gauge to measure weld profiles was to establish the requirements of an ideal gauge, in terms of the measurements it must make, what types of welds it should measure, and how it should function to suit the user best. Consultation was not only limited to the sponsoring company, The Welding Institute, but many other weld engineers, inspectors and welders from several companies in production and research were interviewed for their professional opinions and recommendations.

4.1.1 The Welding Institute's requirements

The Welding Institute believe that there is a need in weld inspection for a gauge which can measure the profile of a welded joint and produce a paper print-out of the results. After discussion, it was felt that the following weld parameters should be measured.

- ❑ Reinforcement
- ❑ Width
- ❑ Undercut on both sides of the weld
- ❑ Plate to plate displacement
- ❑ Angles between plates
- ❑ Toe angle between plate and the weld
- ❑ Leg length of fillet welds

The definitions of these features are given in AWS D1.2 [37] and BS 499 [38] with the dimensions for each measured by the gauge given section 4.1.3.2 and shown in software in section 4.4.1.4(3). These measurements were required to have equivalent values on fillet welds and these are also described in section 4.1.3.2 . Figures 4.1 and 4.2 show the measurements diagrammatically. It was decided that the gauge should offer the capability of measuring welds joining plates of a range of angles to allow both fillet and butt welds to be measured.

At a meeting at The Welding Institute in December 1987 it was agreed that it would be wise to design a system which offered the features required by the weld inspectors and engineers, and to produce an electronic portable weld measurement gauge, but to design it with a view to its inclusion as an integral part of a range of weld measurement systems. The implementation of the algorithms on faster and more powerful processors would share the development costs between a high speed system for on-line inspection, as well as off-line using less powerful processing.

4.1.1.1 Comments of the Welding Institute's engineers

Departments at the Welding Institute were asked to comment on the design of the present Welding Institute gauge and to provide some ideals on how it could be improved. Six engineers from the various departments provided reports. All of the reports ,were critical of the present gauge for its accuracy and ease of use. Norman [39] reported that "for accurate measurements or for use by fabrication inspectors it is not adequate. The possible implications of product liability legislation and the phrasing of existing codes of practice necessitate either a change in marketing or a change in design." he continues later to quantify the accuracy to which the measurements can be made; "it is unlikely that accuracy better than ± 0.5 mm could be achieved in any reading - the significance of these inaccuracies clearly depends on the size of the dimension being measured and its importance."

The Institute's engineers had in mind the differing needs of the welders in industry and those of the welding inspectors; several suggested that after some improvements to the existing gauge , it would still be adequate for the welders. However their comments showed a clear dissatisfaction with the present gauge as a precision instrument. The inspectors required an instrument which had a higher order of accuracy and made several suggestions of the kind of device which could achieve this.

It was suggested by Dunkerton [40] the there is a need for a "complete new design of instrument", for use by inspectors, but he felt that two markets existed where the present gauge was adequate for use by welders. Blakeley [41] reported "The accuracy of the welding gauge must be greater than the accuracy with which welds can be made, and the accuracy required by welding inspectors". He proposed a new concept to produce "a solid state welding gauge". This gauge is a microprocessor based tool using optical reflective methods to measure the weld features with a digital numeric display.

Crawford [42] suggested, "An electronic instrument employing an optical system with light sensitive diode detector banks and digital read-out would have the advantages of high accuracy and ease of use". He recognised that "It is possible that such a novel device may be treated with some scepticism by welders, and not receive general acceptance."

4.1.2 The welding industry's requirements

In order to get an appreciation of the requirements which the welding industry would have for an automatic weld measurement gauge a number of companies were contacted and they made inspectors and welders available for interview. The companies which made personnel available were:

- ❑ The Welding Institute, Cambridge
- ❑ C.E.G.B. Marchwood Research Laboratories, Southampton
- ❑ Air Products Ltd, Acrefair, Wrexham
- ❑ N.E.I. Parsons, Newcastle
- ❑ Babcock Power, Glasgow
- ❑ E.S.A.B., Stevenage
- ❑ Harwell Atomic Research Centre, Oxford

The measurements which were universally considered to be important were: reinforcement, width, undercut, toe angle, displacement between plates and the angle between the plates. However, most of the professional comment related to the operation of the gauge and the features which would prove useful.

4.1.2.1 Comments of weld inspectors

Industrial weld inspectors showed great reluctance towards an electronic gauge and, initially, appeared sceptical that it would be possible to measure weld profiles automatically. Mostly, their resistance soon changed to encouragement once it was established that the gauge would not perform a complete visual inspection, and consequently make their job redundant. The design of an electronic tool which measures weld profiles was considered useful, as all the inspectors felt that the present gauges, which are designed to do this, fall so far short of the requirements as to be generally useless. Most inspectors rely on the acuity of their eyes and a six inch steel rule to ensure that the weld is not outside the specification, rather than to measure the weld at regular intervals.

The inspectors at Air Products Ltd. were particularly enthusiastic, since all specifications for the manufacture of pressure vessels require visual inspection and measurement of the entire lengths of all welds as well as X-ray tests to reveal internal defects. The inspection operation in the manufacture of pressure vessels can take several days for a large vessel, and a tool which automatically measures a weld's profile, would both speed up the inspection and increase the number of points at which measurements could be made.

The weld inspectors felt that it was important that the gauge should be easy and quick to use. Consequentially a full "QWERTY" keyboard was designed into the gauge so that information could be typed in as easily as possible. The numeric results from the gauge should be shown at the same time following each measurement. A graphical representation of the weld profile was also considered to be important in order to show that the measurement has been made in the correct place and that the weld is correctly sited under the measuring aperture.

The gauge should be able to be placed over the weld and measure at a specific point, that is a position about which the inspector is concerned. In addition it was also suggested that it would be advantageous to be able to set the measurement going continuously while the weld passed beneath the gauge.

Offering independence from the mains power supply by incorporating a battery into the gauge was not seen as a major advantage since most places where the gauge would be used are likely to have a mains power supply available.

All inspectors felt that the gauges which are currently available are inadequate and rarely used, because they are inaccurate and can only measure a limited number of weld types.

4.1.2.2 Comments of welders

Welders showed approval of the idea of an automated measurement gauge if it would remove the variation in the standard that constitutes a passable weld between inspectors. It was suggested by a couple of welders, who were questioned, that the precision to which the gauge could measure, should not be greater than that to which a welder could weld. However, it was generally accepted that this precision varied according to the welding process.

4.1.3 Defining a weld and its parameters

Current measurements are directed to either fillet or butt welds of which their measurable dimensions are shown in figures 4.1 and 4.2. If welds between plates at any angle are to be measured, it is preferable for the dimensions to be the same throughout the whole range of plate angles. Since the electronic gauge can measure welds between plates of various angles, a set of weld dimensions which are compatible with existing measurements, had to be defined for use in the gauge.

4.1.3.1 Existing qualitive specifications

The current measurements which are accepted are geared to two types of weld, fillet and butt welds, leaving a 'grey' area between. The measurements are defined in figures 4.1 and 4.2. The two main sets of standards used in defining these measurements in the UK are The American Welding Society (AWS) and British Standards. The AWS standards which defines the measurements of a weld are AWS D1.1 [43], for steel, and AWS D1.2 [37], for aluminium with the comparable British Standard BS 4870 [44]. These standards define the dimensions and what measurement values constitute a passable weld. Further standards set out the methods and requirements of complete visual inspection, of which weld profile measurements are included, BS 5289 [45] and ANSI/AWS B1.10-86 [46], ANSI/AWS B1.11-88 [47]. Some companies making specific high precision or critical welds define additional standards, for example BNFL have such a standard for visual inspection of stainless steel [48].

4.1.3.2 Defining the weld features

Each feature or dimension of a weld which is to be measured must be defined quantitatively and rigorously for all weld types. Investigation into the measurements which are currently made, show different names for the same dimension on different weld types (butt and fillet). For an automatic measurement system it is necessary to define quantitatively each measurement so that the same feature is always measured; also, the measurements must conform to the commonly accepted measurements which are presently recorded by inspectors. Most welding reference to these measurements is in terms of when they are too big or too small or exhibit a defect. This may be, for example, excess reinforcement or any undercut; good descriptions of these profile defects are given in Weld-link [49] and IIW [50].

The diagrams showing the measurements corresponding to the weld features discussed below are shown in figure 4.3.

(1) Reinforcement

The reinforcement of a weld is the maximum height to which the weld metal has formed above the level of the parent metal. This quantity is defined only for butt welds, where the parent metal plates are parallel (plate angle of 180°) and have no misalignment (displacement of 0 mm). In designing an automatic gauge to measure weld reinforcement, it became necessary to specify a more defined dimension for reinforcement so that welds between plates at varying angles and displacements can be recorded. Since the project's aim required the software to measure welds between plates at various angles, it was important to consider the measurements made for fillet welds (plate angle of 90°). In this case the measurement of reinforcement could be translated into the measurement made of the throat, shown in figure 4.1. Thus the measurement of reinforcement shown in figure 4.3 encompasses both the throat and traditional measurement for reinforcement.

The measurement of the throat of a fillet weld, which does not appear on the gauge, is given by reinforcement. The definition of reinforcement as defined in figure 4.3 directly corresponds to the throat as measured by other gauges.

(2) Width

The width of a weld is defined as the distance between the points where the weld metal meets the parent metal on either side of the weld. These points are not necessarily clear from a weld's profile, but given that the corners of the weld are found correctly, the width is measured between them.

(3) Undercut

Undercut may occur at either side of the weld, at the point close to where the weld metal meets the parent metal. It is defined as a recess in the parent metal before the weld and is included in the measurement of the weld width. Undercut is measured with respect to the adjacent parent plate as a dip below its level.

(4) Toe angle

The toe angles of a weld profile are the angles at either side of the weld between the parent plate and the weld at the weld corners. Each toe angle may be greater or less than 180° depending on whether or not any undercut is present.

(5) Plate angle

The angle between the two parent metal plates is taken as the plate angle where 180° indicates a butt weld and 90° a fillet weld.

(6) Plate displacement

Plate displacement applies most significantly for butt or lap welds where the measurement is also described as plate "fit-up". Assuming the plates are parallel and horizontal, the displacement is the vertical displacement between the upper surfaces of the plates. This measurement is also known as misalignment.

(7) Leg length

The measurement of leg length applies only to fillet welds, and is shown in BS4870 [44] to be the distances to either side of the weld from a point where the lines of parent metal cross, at the centre of the weld.

4.2 A review of tools currently used in weld measurement

All of the methods currently used for weld profile measurement employ simple manual mechanical gauges; the main ones are described in the following sub-sections with their merits and draw-backs. All these gauges require contact with the weld and cannot be used on inaccessible welds. Plate 4.1 shows the gauges described in the following sections. The use of these gauges as a part of visual inspection is discussed in Wehmuller [51] and Timberlake [52].

4.2.1 The Welding Institute gauge

The Welding Institute gauge is mechanical and solidly constructed from stainless steel, plate 4.1 (top right). There are two moving slides attached to the main plate of the gauge which move to contact the weld to so that measurements can be made. The measurements which can be taken by the rotating pointer are: weld reinforcement, undercut and, for fillet welds only, plate angle. The sliding pointer allows measurement of fillet weld throat thickness. Weld width is measured using the rule on the main plate.

The gauge cannot measure all desirable parameters for all weld types and it requires 140 mm of flat plate on either side of the weld. The measurement scales are marked in both imperial and metric, but the gauge can measure only to the nearest 0.5 mm. This gauge is awkward to use and a great deal of care has to be applied in order to produce an accurate result.

Engineers at The Welding Institute were asked to produce reports on the gauge. These were critical of the present gauge as an inspection aid. Typical of their comments were:- Watson [53] "The gauge in its present form has been criticised on several counts:

1 It is difficult to use
2 There is a parallax error when the gauge is viewed at an angle
3 There are large zero and scale errors (often as much as 10%)",

and Slater [54] "This gauge is inappropriately described as a precision instrument."

4.2.2 The Welding Institute weld inspection replicas

The weld inspection replicas (plate 4.1 top left), produced by The Welding Institute, claim "to enable rapid visual identification and accurate estimation of surface shape imperfections on both butt and fillet welds" [55]. The replicas are a set of two mouldings. Examples of butt and fillet welds with undercut from 0 mm to 2 mm in 0.5 mm steps are provided on one piece. The other replica piece shows examples of toe angles of 157°, 140°, 125°, and 107°, with root penetrations of 1.1 mm, 2.4 mm, 3.2 mm as well as concavity. The weld features which can be measured are made by comparing the weld, under inspection, with the samples on the replica. This, so called, gauge is of limited use since only small weld examples are provided and any measurements which may be discerned are almost unquantifiable.

4.2.3 Contour gauge

The contour gauge consists of a line of steel rods, held in a straight line, which when pushed against a weld take on its profile, plate 4.1 (middle right). This gauge allows the profile to be viewed away from the weld. The precision depends on the rod size, which on the gauge shown is 0.81 mm, giving a resolution of ± 1.6 mm on width or height. The measurements are distorted when the rods are larger than any hollow or pit being measured. For example, a weld with undercut which is less than the width of the rods would not show up, and, consequently, would not be measured. Further inaccuracy can occur if the gauge is not set perpendicular to the weld. An example of a profile gauge is shown in plate 4.1 and described in detail in BS 5289 [45]. Gauges of this type can be applied to both fillet and butt welds; a manufacturer of these is Biltse Instr. Fab. b.v., in the Netherlands, who make "WEVO weld gauge apparatus" [56].

4.2.4 Fillet weld gauge

Most suppliers of welding equipment produce a simple gauge punched out from aluminium plate, plate 4.1 (middle left). They consists of a number of arcs around the edges of the gauge with the height and width marked, and these are arranged to allow assessment of plate angle. It is less provided as a gauge rather as an aid to inspection, but they are commonly used by inspectors as a quick check to a weld. This type of gauge is detailed in BS 5289 [45].

4.2.5 Six inch steel rule

A six inch long steel rule is the most commonly used inspection tool to assess a weld profile measurement, plate 4.1 (bottom). The ruler is placed across the weld on its edge to measure the width and rocked to assess its reinforcement. Inspectors generally use their fingernails to check for undercut and toe angle, but these are, more often than not, assessed rather than measured absolutely. The accuracy of measurements made are down to the skill and experience of the inspector.

4.2.6 Automated weld inspection

There has been much development in the control and guidance of welding in the past ten years, however far less concentration has been placed into the inspection of welds. MVS [57] is a Canadian company which sells systems primarily for weld guidance using a similar laser stripe technique as employed in the gauge. MVS also suggest that post weld inspection is performed with their MVS-10 system. However when describing the post-weld inspection their sales literature states "The post-weld inspection measures the material shift (mismatch) that occurs during welding". Thus only the plate displacement (fit-up) is measured and not the other weld parameters.

4.3 Design of the hardware for the portable gauge

The design of the portable welding gauge was affected by a number of influences which have determined its looks, size, features, function and the electronics. These influences varied in their importance and did not always reinforce each other to provide a positive factor in a design decision. An engineering balance was struck between the ideal design of a portable gauge suggested by one section of people and the ideal gauge as described by another group or individual. The significance of the ideals of each contributing party were assessed and considered with due regard to the constraints which were placed by people or technology. The influential parties are listed below with the main constraining factors.

- Weld inspection research engineers at The Welding Institute
- Weld inspectors in industry
- Welding research engineers at The Welding Institute
- Welders in industry
- Functional requirements
- Other electronic weld equipment
- Envisaged product sale price
- Envisaged sales volume
- Component cost for development

The positive influences have already been discussed in section 4.1 . The design criteria of the gauge are now re-assessed with regard to the constraints listed above. In practical terms the constraints were as follows:

- ❑ The gauge must be portable
- ❑ The component cost of the gauge must be less than £1500

These constraints allowed much greater flexibility in the design of the hardware to fulfil as many of the features and functions which the influential parties desired. Section 4.1.1 adds one other desire, namely, that the research towards designing a portable gauge should be capable of implementation as an on-line measurement system.

4.3.1 The design decisions

Once the requirements and constraints for a gauge had been established, the next stage was to consider how best they could be achieved. A non-contact inspection would be necessary for real-time, on-line inspection, if it was not to interfere with the welding process. Portability necessitated that it must be small and light enough to carry, but rugged enough to withstand knocks and bumps.

An optical measurement sensor was considered to be the most flexible and could build on previous research into weld seam-tracking and monitoring, Terry [1]. A vision based gauge using a laser stripe technique was chosen in order to extract the information of the weld profile.

4.3.1.1 Deciding on what processor system to base the gauge

Discussion of whether to use "off-the-shelf" components or to design special hardware reduced to a choice between basing the system on established IBM standard computers or the Inmos transputer. The IBM PC standard computers were the only system to offer portability with adequate processing power. No advantage would be gained by designing specific hardware around the Intel processors, used in the IBM computers, or using the similar Motorola processor family, since all the limitations of the IBM computers would be imposed with the additional problems associated with designing specific hardware. The Inmos transputer, on the other hand, gave the potential of high processing power and the capability of increasing the power by adding more processors. This, coupled with the development of an image processing system which would offer the capability of processing images in real-time, placed the transputer system as the only competitor to the IBM based solution.

The use of a PC-bus frame store, such as the Matrox PIP 1024 [2], see chapter two, plugged into a portable PC would fulfil the portability need. However, for the system to run on-line, plugging the framestore into a higher power PC, for example with an 80386 processor, might have allowed measurement fast enough for on-line inspection. This solution was not pursued for a number of reasons.

- ❑ A costing at the time showed that a portable PC based measurement system would have cost about £6400 to prototype whereas a transputer based system would only cost about £1500 to build.
- ❑ The transputer provides an extremely fast single processor, but unlike the PC it can be connected together with many transputers to share the processing and produce the results as fast an necessary; more details of this are given in chapter two.
- ❑ At the same time as the choice of the computer on which to base the measurement system was to be made, a general purpose vision system was being developed at Liverpool University for performing image processing in real-time. This provided a cost effective computer on which to base the portable gauge and could itself be used for real time weld measurement, Chambers [24].

The Transputer Image Processing System (TIPS) was used as the development system for the weld measurement software with a special system built for the portable gauge which is compatible with TIPS.

4.3.2 User interface hardware

Electronic circuits to interface the user hardware were designed to allow textual and graphical information to be presented on screen and be printed out on to paper. Entry of alpha and numeric information was considered important and to speed up the entry a full "QWERTY" keypad and number pad was chosen. These features can be seen in plate 4.2 which shows the completed gauge. Towards the end of the project it was requested that it would be advantageous to offer the capability of down-loading the results on to another computer for post analysis. This was achieved by developing a serial board compatible with the RS232 transmission standard [58], which connected to the gauge externally, but could easily be incorporated inside the gauge. The interfacing circuits of these peripheral components to the transputer are described in the following sections. The electronic circuit boards in the portable gauge are shown in plate 4.3 and 4.4 .

4.3.2.1 Small text and graphics printer

The printer which was chosen for incorporation into the gauge is a Datac 164SB [59] and comprises a printer mechanism with drive circuit. It operates from a single 5 Volt supply capable of delivering 1.5 Amps. The interface to the drive circuit utilised the parallel input which simply connected to one of the Inmos C011 communication chips. Since the printer requires data in only one direction the C011 was shared with the keypad. The interface circuit for the C011 is shown in figure 4.4 and also shows the reset circuit onto this board. The C011 in figure 4.4 connects to the printer drive circuit board by the output data lines, "Q0-7" of the C011 with "QVALID" and "QACK" providing the handshaking signals for "STROBE" and "BUSY". The hand shaking followed the same timing protocol, but required "QVALID" to be inverted since "STROBE" is active low. the printer is reset with the rest of the circuit, at power up or manually, but must be driven through a 100 Ω resistor and is active low. The parallel interface is selected by tying the selecting pin to ground. The connection to the printer is shown in figure 4.5 . A paper feed feature is offered, and designed to be controlled directly from the keypad from the "LNFD" key indicating line feed. The connection to the printer is shown in figure 4.6 .

The printer offers 40 characters per line, but can generate double width and/or double height characters in text mode. These are shown in plate 4.5 showing examples of print outs from the gauge during trials. In graphics mode, also shown in plate 4.5, resolutions of 96 dots per line are available.

4.3.2.2 Full QWERTY keyboard

The keypad shares the Inmos C011 communications interface chip to the transputer link with the printer. The support circuit for the C011 is shown in figure 4.4 and detailed in the Transputer Data Book [27]. The keypad uses the input side of the C011 connecting to data lines "I0-7" and achieving data handshaking with "IVALID" and " IACK". The keypad is a sealed matrix type and is polled using a "9600-PRO" keyboard encoder. The "9600-PRO" contains the logic circuits necessary to debounce and decode a keyboard of matrix arrangement. It produces a nine bit data output which is converted into the ASCII character by the "2716" EPROM, see figure 4.6. The keyboard encoder offers decoding of a full "QWERTY" and numeric keyboard with "SHIFT" and "CONTROL" functions. The EPROM provides a lookup table which decodes the number codes for each key press into the standard character codes.

A key press causes the "DATA READY" line of the "9600-PRO" to become active signalling to the C011 that valid data is waiting. When the data is read by the transputer, at the other end of the link, the "IACK" line is asserted by the C011 and the next key press is ready to be taken. A selectable click, or bleep, sounds when the key press is ready by the transputer to acknowledge the key press to the user. The circuit which produces the sound is shown in figure 4.7. A 555 timer chip generates the oscillations to drive a audio warning device and is held normally in "RESET" to prevent it oscillating. When a sound is required "RESET" is released under the control of a "74123" pulse generator. The selection of a click or bleep is achieved by varying the length of the pulse from the 74123, by changing the capacitance across its inputs.

4.3.2.3 Text and graphics display

A liquid crystal display, made by Hitachi [60], was selected for the display function of the gauge. The LM313XBN offers text and graphic display, though is used on the gauge mainly in graphic mode. Interfacing the display to the transputer was achieved in a similar way as the keyboard and printer, except the timings are generated by a state machine programmed into the PAL 22V10 shown in figure 4.8. The communication with the display is bi-directional, hence requiring two registers to latch the data in either direction to meet the timing requirements of the interface. Back lighting of the LCD panel is provided with an electro luminescent panel mounted behind the display area. This is driven using a "P4" dc-ac inverter chip matched to the size of the panel.

4.3.2.4 Save to pc

A useful feature was considered to be able to use the gauge for data logging and sending the results from any measurements to another computer for storage or later analysis. Since the cost of incorporating a floppy disk drive was considered to be too high a cheaper solution was adopted and an RS232 standard serial communication was added. The circuit diagram for this is shown in figure 4.9 which shows a C011 providing communication with the transputer interfaced to a 6402 UART and a MAX 232 generating the RS232 standard communication with the other computer. A 9 pin "D" connector provides compatibility with the IBM AT serial communications adapter. An assortment of baud rates and word lengths are configured by selection of the DIL switches on the board.

4.3.3 Image processing hardware

The gauge is based on the Transputer Image Processing System design to maintain compatibility with TIPS for any future "on-line" applications of the software. However, modifications and additions to the design of the TIPS specification have been incorporated. The gauge uses only two of the three TIPS boards; a standard Image Processing Board and a modified and extended Frame Capture Board.

4.3.3.1 Image Processing Board

The Image Processing Board contains a single transputer and is used to run all the software required by the gauge. Thus, unlike TIPS, the image processing board controls the image capture board, determining when a new image is required and in which frame buffer it is to be placed. Since the design brief was to produce a low cost gauge, the IPB has only one frame buffer of 512 x 512 pixels. This allows full image processing to be performed, but the processing of one image must be completed before the next can be captured. Only having one frame buffer does not present a significant problem since sending the image or results to the liquid crystal display, takes much of the total time for each measurement iteration.

The image processing board is connected to the user interface board by two bi-directional Inmos links to provide the keypad, screen and printer. The remaining two links are used for control of the image capture board, receiving the program and providing communication with any other computer via the RS232 communications board. All of the interfaces to these peripherals are described in section 4.3.2 .

The TIPS Image processing board is fully detailed in Chambers [24] with the circuit schematic diagrams shown in figures 4.10, 4.11 and 4.12 .

4.3.3.2 Frame Capture Board (FCB)

A special board was produced for the image capture to allow it to be controlled by the IPB and to provide a display function for recording a measurement session onto video tape or to a raster scan monitor for detailed inspection and calibration. The design of this board followed that for the TIPS FCB, but had extra circuits to allow these features and to permit the program to be down-loaded from an EPROM to the IPB by the control link. The display function was incorporated into the TIPS FCB design and is detailed in Chambers [24] along with the rest of the FCB design. The modified circuit schematics of the FCB are shown in figures 4.13 and 4.14 with the extra circuits and modifications detailed below.

(1) Down load the program code for running

The program, software, was compiled using the Transputer Development System (TDS) Occam2 compiler for loading into an EPROM from which it is sent to the transputer in the gauge at run time via one of the links. The C011 used on the image capture board to program the timing registers is used only in the out going direction. Thus the circuit to down-load the program, in the EPROM, was designed to use the inward side of this C011, see figure 4.15 .

A 27512 EPROM was used with the program occupying about 42 kBytes of the 64 kBytes available. The program is transmitted to the transputer, byte at a time, by a 16 bit counter providing the addressing for the data to the C011 via a 74374 latch. The program is sent to the transputer after a "RESET" is activated at power on or manually. The EPROM is located on the rear of the frame capture board to make changing it easier; this is shown in plate 4.4 .

(2) Save to video

The feature allowing a measurement session to be recorded onto video also allows the gauge to be connected to any 15 kHz interlaced monitor so that the image which the camera is viewing can be seen in detail. The feature was added to the basic design of the TIPS Frame Capture Board originally for debugging the software, but was subsequently thought to be of use during inspection where ultimate recording of the weld was desirable. The addition to the TIPS FCB involved adding display circuitry to the existing design.

The program can be requested, by pressing "CONTROL and D", to produce the graphical measurement data on to the image superimposed over the original picture. This allows the operator to tune the gauge, by altering the set up parameters, to measure different sizes and types of weld and also provides a much more accurate display for calibration.

Since the ability of being able to save the measured results onto video tape was not considered of great importance this was not included.

4.3.4 The measurement head

The head comprises a low cost CCD camera fitted with a lens and optical band pass filter matched to a 5 mWatt laser diode and a button switch with indicator LEDs which connect to the gauge main box with a single, multicore cable. The connections to the circuit boards are shown in figure 4.16, with the laser switched on and off through a reed relay switched by a bit from a memory mapped register of the transputer on the image processing board. The switch also connects to another register on the image processing. The video cable bringing the image from the camera to the gauge is housed in this multicore cable, but connects directly to the image capture board. A diagram showing the arrangement of the camera and laser *in situ* in the head is shown in figure 4.17 and shown in plate 4.6.

4.3.4.1 The camera

A low cost, Philips CCD camera [61] was selected for design into the gauge. It requires a 12 Volt supply and produces RS170 standard video which is connected to the frame capture board in the gauge main box. The camera has a 'C' mount connector to attach the lens which, being a common lens fitting allows a wide variety of lenses to be fitted. The automatic gain control and lens aperture control are switched off fixing the image produced to a constant level.

4.3.4.2 Optical bandpass filter

An optical band pass filter of 780 nm is fitted to the front of the camera's lens which matches the frequency of the laser to permit maximum transmission of the laser light to the camera with minimum background light. This filter is used to reduce variation of image intensity due to sunlight and to eliminate welding light disurbing measurement.

4.3.4.3 The laser diode

The laser used in the measurement head has a power of 5 mW with a wavelength of about 780 nm and, therefore, falls into the class 3b category of lasers. The laser is focused into a thin stripe, or line, by a cylindrical lens which can be seen in the measurement head in plate 4.6. The laser used is a Sharp LT026MD0 [62] infra red laser diode.

4.3.4.4 Safety considerations of the laser diode

Although the laser which is used in the head falls into the emission class 3b, for laser radiation, unlike gas lasers (for which the class was originally derived) the light beam emitted from the laser diode scatters; consequently it is not as hazardous. However caution must still be applied when using the head and it is advised not to look into, or at, the laser diode when it is on. Since the diode which is used emits invisible light no blink reflex or visible indication is apparent that the diode is active. The head is fitted with an ultra bright Light Emitting Diode (LED) as a warning, but during maintenance special care is needed.

Further information on the safe use of lasers can be found in British Standards B.S. 4803, [63].

4.4 The software design for the portable gauge

The software requirement for the portable gauge divides into two sections: that part which performs the weld measurement and that which is required for the user interface. Each part of the program is as important as the other; if the software which handles the user interface is poor and presents the results in the wrong way for the inspector to utilise, then the results from the rest of the system will not be taken with confidence by the welding engineer, even if these are completely accurate and reliable. It is easy to concentrate on the academically more taxing parts of programming, in terms of the mathematics and the algorithm, but equal attention must be devoted to considering what the weld inspector requires and how best to present this information to him.

Both parts are discussed in detail in this section considering the weld measurement software first.

4.4.1 Weld measurement

Consistent and accurate results are essential and much care has been taken in the design of the algorithms to ensure their security. Provision for as many error situations as possible has involved a great deal of testing of the software to minimise bugs. Testing the software has involved checking the inputs to and outputs from each algorithm, and their internal function, as well as the measurement software as a whole.

Versatility to measure so many different types of weld profile adds more checking and, consequently, more processing, all of which is performed on a single processor in the portable gauge.

4.4.1.1 Development

The gauge was initially modelled using an IBM-AT as a host for a general purpose transputer board, B004, made by Inmos [64], with images captured and displayed on a Matrox "PIP-1024" framestore [2]. This allowed the image processing software for the weld measurement to be developed before construction of the gauge was started. This development system is shown diagrammatically in figure 4.18.

The software was written using the Transputer Development System (TDS) [65], which is supplied with the B004 board. Hence, programs were compiled ready for running, testing and debugging.

(1) The Inmos B004 transputer board

The B004 board [64] comprises a transputer (see chapter two), 2 Mbytes of memory, mapped into its address space, and also an interface to an IBM-PC compatible computer. The transputers links are available for configuration as required, but, if used as the only B004 board in an IBM computer, one of these is connected to the interface part of the board to enable communication between the transputer and the IBM computer.

Programs are run on the transputer by sending them to it via one of the links shortly after the transputer has been reset. The TDS software runs on the B004 which limits the size of user programs that can be developed if they are also to be run on this board.

(2) The Matrox PIP framestore

The PIP framestore offers four 512 x 512 resolution frame buffers of 256 grey levels, and it interfaces to the IBM computer by a number of registers mapped into the IBM's I-O map. Thus, any pixel may be accessed (read or written) by writing the x and y locations of the pixel into two address registers and reading from or writing to a third data register. A feature designed to speed up pixel access is also incorporated in the design in which the address registers are automatically incremented after each access. Thus, once the line start address has been given, the rest of the pixels in the line can be read or written consecutively by reading from or writing to the data register. Other registers are also provided to set up hardware timings and to configure the board.

The resolution of 512 x 512 with 256 grey levels is the same as in the TIPS. Unlike TIPS though, the pixels are not square, but have an aspect ratio of 4 : 3 to match standard television monitors. This complication is particularly obvious when performing image processing on circular objects, such as the ferroxcube core central pillar. The equation of the circle must be modified to account for the unequal aspect ratio. By using compensating factors, this could be changed for implementation on the TIPS with its square pixel images.

(3) Modelling the gauge

Initially, the software was developed on the hardware, as shown in figure 4.18, with the program run on the B004 and the images captured and displayed using the PIP framestore. The sequence of events of an image processing iteration in the program was as follows:

i Ask the PIP framestore to capture a new image and transfer it through the IBM to the B004
ii Process this image
iii Draw results and report information on to the image
iv Transfer the modified image back through the IBM computer to the PIP framestore for display analysis

For the weld measurement software, the model was sufficiently similar to the gauge to allow the majority of the algorithms to be written. The image appeared in the memory of the transputer on the B004 as an array of 512 x 512 pixels ready for the program to perform the weld profile extraction and then the weld measurement. The only changes which were required were to replace the subroutine which captured the image by a new one and to remove the function which displayed the processed image at the end of the sequence.

The iteration time of the model system was very slow, (it took over six seconds to transfer the image to or from the PIP framestore) and only a single transputer was available to run the program. Once the hardware for the gauge was completed this allowed much faster development of the measurement software.

4.4.1.2 The measurement software

The measurement software was formed into a separately compiled procedure which allowed it to be easily incorporated into the user interface of the portable gauge, or into a harness for implementation in on-line applications using TIPS. The procedure is written using a fold editor and is described in this fashion to convey the functionality of the program. The software described is the version running on the portable gauge using predominantly sequential constructs, but some parts of the program could be written in parallel if implementation on several processors was desired. The parallelisation of the software is discussed in section 4.8.2.2, "On-line implementation".

The function is called with the results returned in three arrays. A channel "fcb" is provided for control of the capture circuitry.

```
PROC pms(CHAN OF ANY fcb, [512]INT line, [5]INT weld.results,
         [11]REAL32 real.weld.results)
                                                            {{{  libs
  ...  libs   -    -    -    -    -    -    -              #use pms_vals
  ...  results vars                                         #use port_pms
  ...  ipb vars                                             #use t4math
  ...  vars for drawing                                     }}}
  SEQ
    ...  init vars from call
    ...  get picture in and find laser line
    ...  process line
    ...  draw on image
:
```

The library functions, definitions and initialisation of variables and values can be seen where they are called by the program in the following sections. The program then divides into three parts; the first part "getpicturein and find laser line" requiring access to the image and is discussed in section 4.4.1.3, "The image processing software". The second stage extracts the weld profile and measures the weld features, this is discussed in section 4.4.1.4 "The data extraction". These parts cannot be run in parallel to achieve a speed increase on more processors since the line of the laser stripe across the weld profile must first be found before it can be used to give the data about the weld's profile. A more detailed discussion of parallelisation of the software is considered in section 4.8.2.2, "On-line implementation of weld measurement". The last part "draw on image" is not used on the portable gauge unless a monitor is connected and the function enabled. This is a time consuming operation, but is required for debugging the system and could be useful in on-line implementation.

4.4.1.3 The image processing software

There are many approaches to solving image processing problems, as have been described in chapter three. Since response time is of importance in the software, as the target system will be run on only one processor, an engineering approach to the solution was adopted and no global mathematic algorithms or filters were applied to the image, for enhancement, prior to feature extraction. Specific algorithms have been developed to find the profile from the raw image data, and the image is enhanced by hardware methods such as the laser stripe and narrow band-pass optical filter which are described earlier in this chapter.

The algorithm for finding the weld profile comprised the following elements:

- i Capture an image of the weld profile highlighted by the laser stripe
- ii Find the points on edges of the image where the laser stripe starts
- iii Search across the screen from one side to the other to find each vertical, y, position of the line for each horizontal, x, position.
- iv Find the limits of the weld

These are now discussed in the following sections with the Occam code at this level shown below.

```
...  Vals and vars
SEQ
  ...  set up fcb
  ...  capture into next frame if not positioning
  ...  find end points of line (X,Y)
  ...  find line
```

The capture board is set up with a function in the library, "port_pms" and is described in the next section.

(1) Capture the image

The routine to capture an image involves the program switching registers in the capture circuitry on the capture board to turn the capturing process on, then off again a frame later. The transputer's "event" pin is used to synchronise the switching of the capturing process and must be cleared if it is set by a previous frame. Once capture has been started, the system waits for one frame, to ensure that the whole image will be read into memory and waits for the end of the frame signified by the next "event" signal. This is illustrated in the following software.

```
CHAN OF ANY event:
PLACE event AT 8:
BYTE junk:
SEQ

  ...  flush event  -     -     -     -     -

  fcb ! 24(BYTE) -- set off capture
  fcb ! 3(BYTE)

  ...  delay 1 frame-     -     -     -     -

  fcb ! 24(BYTE) -- finish capture
  fcb ! 0(BYTE)
  event ? junk   --  wait for end of capture
```

```
{{{  flush event
PRI ALT
  event ? junk
    SKIP
  TRUE & SKIP
    SKIP
}}}
```

```
{{{  delay 1 frame
TIMER clk:
INT time:
VAL delay IS 15625/25:
SEQ
  clk ? time
  clk ? AFTER time PLUS delay
}}}
```

(2) Find the ends of the laser stripe across the weld profile

The ends of the laser line in the screen are found in order to give a start location for the next piece of software which finds the profile of the weld from this line. It also serves the purpose of checking that a line is present on the screen. The two ends are found independently and can be performed in parallel. Only one end is described since the logic for the other end is the same. These end points can be seen drawn onto the processed images and are marked as lines "3" and "4".

The result of this piece of code is an (x,y) location of the point at which the laser line appears nearest to that edge of the image. Normally, the line extends beyond the image and this point is found on the first line searched. The algorithm operates by reading each pixel intensity in turn, along a vertical column from one edge towards the other, until a value is found above a predetermined intensity or threshold. This threshold is a value for which the laser line always is greater and the background is usually below. Once this first point is encountered, the pixel with the highest intensity along the column is sought and then this is used to calculate a threshold for the window used in the averaging filter later in the program which finds the weld profile. If no point is found above the pre-set threshold, the image is deemed to be blank and the measurement aborted with a corresponding error code.

```
...  Vals and vars
SEQ
  ...  find first point above the threshold
  IF
    end.found = FALSE
      SEQ
        l.end.y := 0
        l.end.x := 0
        pms.error := 1
    TRUE
      SEQ
        ...  find y position of highest intensity for current x value
        ...  calc window threshold
        l.end.y := biggest.window.intensity.y
        l.end.x := x
```

The above algorithm which is used to find the first point greater than the threshold value is shown coded into occam below. The program starts searching the image vertically for a pixel above the threshold value, and moves to the next column if one is not encountered. Once a point has been found, the search along this column continues to find the next pixel which is below the threshold value to establish the approximate of the thickness of the line. This is important since the same laser line will appear wider or narrower according to the surface of the metal plate.

```
SEQ
  x := 10
  y := 0
  {{{  look at image for point above the threshold
  WHILE (image[proc.frame][y][x] < threshold) AND (x < border)
    IF
      y < (resolution-1)
        y := y + 1                                        {{{  increment x and reset y
      TRUE                                                SEQ
        ...  increment x and reset y  -     -               x := x + 1
  }}}                                                       y := 0
  IF                                                      }}}
    (x >= border)
      end.found := FALSE
    TRUE
      end.found := TRUE
  ...  find the width of the line for window.width
```

(3) Find the weld profile

This next part of the program starts at the point found on the line at the left side of the image and produces an array containing 512 points, the y value for each x value in the array. The image is searched in a similar way to that used to find the start point but a one dimensional filter is used to average the pixel intensity values in order to remove noise. Instead of searching the whole column to find the best point, a limited search is performed which is centred on the last point, as the line should be continuous and not suddenly jump. Further refinement is incorporated to account for a laser line which does not have the expected Gaussian distribution of intensities about its centre (see figure 4.19). If the intensity across the line is flat (having several pixel locations across it with the same averaged intensity) and sloping away on either side, the mid value of the laser line is found. This point is added to the array and is checked to ensure that the line has not stopped or has no significant gap where data is missing when an accurate measurement of the weld cannot be made. A small gap in the line is permitted, provided the laser stripe does not disappear altogether, since dirt on the plate can cause a lower transmission of light to the camera and causes the line to appear weaker.

```
...  vals and vars
SEQ
  ...  init vars
  x := l.end.x
  last.y := l.end.y
  WHILE (x <= r.end.x) AND (line.found = TRUE)
    ...  vars
    SEQ
      ...  find y position of highest intensity for current x value
      ...  reposition point if Gaussian is flat
      ...  store line
      ...  check if laser line is present and increment x
```

The resultant line produced from this algorithm is drawn onto the processed image as a black line along the line of the laser stripe. This algorithm can be seen as a method of thinning the laser stripe, on the image, from about 10 pixels to a single one for each x location.

4.4.1.4 The data extraction

Once the array containing the y values for each x is available the information about the weld profile can be extracted. As can be seen in the occam code below, the main steps leading to measurement are finding the equations of the flat parent plate either side of the weld, which then leads to finding the joins (or corners) between the weld and the plates on either side of the weld. Once located, the weld corners combined with the weld profile line provide all the necessary information for the weld feature to be measured in accordance with the definitions given in section 4.1.3.2 .

```
...  vars
SEQ
  IF
    pms.error = 0  -- ie no error
      SEQ
        ...  init arrays
        ...  find eqns for left & right and hence base lines
        ...  find weld start points
        ...  calc weld params for lcd
        ...  put results ready for return
    TRUE
      ...  put zero results ready for return
```

If an error has been found while finding the weld profile line in the previous stages of the software all the return values are set to zero.

(1) Find the mathematical equation of the parent metal

The accuracy to which a straight line can be fitted to a set of data points is dependent on the number of points given. The default value, "sample.limit", in the software requires that the parent plate should provide a datum, which provides a good set of points to fit the line and is 100 pixels for each side (about two fifths of the screen must be parent plate). Vertical lines, "1" and "2" are draw on to the processed image to show the extent of the parent plate which is required. A straight line is then fitted to this flat part of the weld profile with a least squares fit algorithm. This can be performed in parallel but both the equations of the fitted lines are required before the equation of the base line can be calculated.

```
SEQ
  least.square.fit(line, l.end.x, (l.end.x+sample.limit), m.left, c.left)
  least.square.fit(line, (r.end.x-sample.limit), r.end.x, m.right, c.right)
  {{{  find equation of base line
  INT x,y:
  SEQ
    m.base := ((m.left + m.right )/2)
    IF
      m.left = m.right
        c.base := ((c.left + c.right) /2)
      TRUE
        SEQ
          x := ((c.right - c.left) * m.scaler) / (m.left - m.right)
          y := ( ((m.left * x) / m.scaler ) + c.left)
          c.base := (y - ((m.base * x)/m.scaler) )
  }}}
```

The algorithm for least squares fit of a straight line to the line of parent metal is shown in occam code below and is developed from Thomas and Finney [36]. The lines "5" and "6" on the processed images at the end of this chapter indicate the lines found from the least squares fit.

```
PROC least.square.fit([]INT line,VAL INT left.x, right.x, INT m, c)

  #USE pms_vals
  INT n:                                                ┌ {{{  init vars
  INT64 x.sum,y.sum,x.squared.sum,xy.sum:               │ SEQ
  SEQ                                                   │   x.sum := 0(INT64)
    ... init vars   -     -     -     -     -          ┤   y.sum := 0(INT64)
    n := right.x - left.x                               │   x.squared.sum := 0(INT64)
    SEQ x = left.x FOR n                                │   xy.sum := 0(INT64)
      SEQ                                               └ }}}
        x.sum := x.sum + (INT64(x))
        y.sum := y.sum + (INT64(line[x]))
        x.squared.sum := x.squared.sum + (INT64(x * x))
        xy.sum := xy.sum + (INT64(line[x] * x) )
    {{{  calc c and m
    SEQ
      c := INT( ( (x.sum * xy.sum) - (x.squared.sum * y.sum) ) /
                ( (x.sum * x.sum) - (x.squared.sum * (INT64(n))) ) )
      m := (((INT(y.sum)) - (n * c)) * m.scaler) / (INT(x.sum))
    }}}
:
```

This procedure is a separately compiled module and resides in the library "port_pms".

(2) Find where the weld starts and finishes

The start of the weld is defined as the point at which the weld filler metal starts. This is not visible from the profile and thus the gauge looks for a point along the profile where the line deviates significantly from the line of parent metal. This means that a weld between two plates which has a flat profile cannot be located. However, the samples provided by the Welding Institute and all other welds offered for inspection had distinct cross sectional profiles and consequently were measurable.

The software considers the left and right corners of the weld separately searching from the middle of the image. The weld must lie to both sides of the centre of the image. The distances from the line representing the parent plates, found from the least squares fit procedure, to the profile line are calculated. The perpendicular distance is calculated, since the weld samples may be of fillet type and not lie on the axis of the screen. The centre line "0" and corner points, "8" and "9", are drawn onto the processed images in the results sections at the end of this chapter.

Noise on the profile line is accounted for by calculating the maximum and minimum differences between the two lines over a section which is known to be flat parent plate. This aids versatility in finding the weld corners of both welds on smooth as well as roughly finished plates. The point of maximum difference along the line is used as the starting point from which to find the weld corners by searching outwards from this point, since this is likely to be within the weld metal. The software then looks for a point on the line which falls between the lines of maximum and minimum differences found on the parent metal, as shown in figure 4.20. This point is taken to be a close approximation to the corner, if the profile line then remains predominantly within these bounds for the next significant length. If the profile line deviates beyond the upper and lower deviation lines within a significant length, the next location beyond this point is sought and the test starts again.

A secondary algorithm is used to locate the corner more accurately from the first point and is performed during the measurement of toe angle and is discussed in the corresponding section.

```
SEQ
  ...  calc perpendicular distances from the line equations
  ...  initialise max and min diffs
  ...  find max and min difference on left line for sample.limit
  ...  find max and min difference on right line for sample.limit
  ...  modify diffs
  ...  find weld start points looking from the maximum difference out
```

(3) Measure the weld features

The corner points represent the limits of the weld and combined with the profile line between them all measurements can be calculated. Provided that no error has been set at the previous stage of the program and that the left corner has not been found to the right of the right corner, the measurements are made.

```
IF
  pms.error = 0
    ...  Vars
    SEQ
      IF
        (left.start.x < right.start.x)
          SEQ
            ...  plate angle
            ...  calc toe.angles
            ...  calc width
            ...  calc undercuts and positive weld points
            ...  calc height
            ...  plate displacement
            ...  leg length
        TRUE
          pms.error := 6
  TRUE
    SKIP
```

The six individual types of measurement are now discussed.

(a) Angle between the plates

The angle between the plates is calculated from the calculated lines of parent metal such that 180° signifies that the two plates are parallel and form a butt weld.

```
REAL32 real.m.left,real.m.right:
SEQ
  real.m.left  := ( (REAL32 ROUND(m.left)) / real.m.scaler )
  real.m.right := ( (REAL32 ROUND(m.right)) / real.m.scaler )
  IF
    real.m.left = 0.0(REAL32)
      SKIP
    TRUE
      real.m.left  := real.m.left * (calib.factor.x / calib.factor.y)
  IF
    real.m.right = 0.0(REAL32)
      SKIP
    TRUE
      real.m.right := real.m.right * (calib.factor.x / calib.factor.y)
  plate.angle := (PI - (ATAN(real.m.left)-ATAN(real.m.right)))
  plate.angle := ( (plate.angle * 180.0(REAL32)) / PI)
```

(b) Toe angles

Two toe angles are measured, one at the left corner of the weld and one at the right corner of the weld. The angle between two lines, equated by least squares fit from the weld profile line, on either side of the corner point is calculated. A series of toe angles are calculated on either side of the approximate corner point, which has been found above, are evaluated to find the sharpest angle. This serves two purposes, firstly to locate the corners more accurately for the other measurements and also to allow a dual system of corner finding to cope more accurately with welds of both shallow toe angle profile as well as sharp angle profile. Further, as the toe angle may be a point from which cracks start, it is important that the most accute angle is equated.

```
INT m1, c1, m2, c2, new.start.x:
REAL32 max.toe.angle, toe.angle:
SEQ
  new.start.x := left.start.x
  max.toe.angle := 0.0(REAL32)
  SEQ x = (left.start.x - toe.search) FOR (toe.search *2)
    SEQ
      least.square.fit(line, (x-toe.length), x, m1, c1)
      least.square.fit(line, x, (x+toe.length), m2, c2)
      {{{  calc toe angle
      REAL32 real.toe.angle, real.m:             {{{  calc diff in m of lines
      INT m:                                     IF
      SEQ                                          ((m1 < 0) AND (m2 < 0)) OR
        ...  calc diff in m of lines  -                  ((m1>0) AND (m2>0))
        IF                                           m := m1 - m2
          m = 0                                    TRUE
            toe.angle := 0.0(REAL32)                 m := m1 + m2
          TRUE                                   }}}
            SEQ
              real.m := (REAL32 ROUND(m)) / real.m.scaler
              real.m := real.m * (calib.factor.x / calib.factor.y)
              toe.angle := (ATAN(real.m)) * (180.0(REAL32) / PI)
        IF
          toe.angle >= 0.0(REAL32)               {{{  remember max toe angle
            SKIP                                 IF
          TRUE                                     ((toe.angle>max.toe.angle)AND
            toe.angle := (-toe.angle)               ((perp.diff.left[x]<max.diff)AND
      }}}                                             (perp.diff.left[x]>min.diff)))
      ...  remember max toe angle    -               SEQ
                                                       max.toe.angle:=toe.angle
  left.start.x := new.start.x                          new.start.x := x
  left.toe.angle := max.toe.angle                  TRUE
                                                     SKIP
                                                 }}}
```

Variation of the variables "toe.length", "toe.search" (above) and "corner.adjust" (found in the initial corner finding algorithm) allow the gauge to be tuned between an algorithm aimed at placing the corner at the sharpest angle and one aimed at placing it at a point of significant deviation from the parent metal line.

(c) Width

The width of the weld is the distance between the left and right corner points, measured directly between them using both the x and y directions and is shown in figure 4.3. The line "10" on the processed images discussed in the results section shows the line along which the width is measured.

```
INT lx,rx,ly,ry:
INT square:
REAL32 r.diff.x, r.diff.y, real.square:
SEQ
  lx := left.start.x                                 {{{  calibrate r.diff.x
  rx := right.start.x                                IF
  ly := line[lx]                                       r.diff.x = 0.0(REAL32)
  ry := line[rx]                                         SKIP
  r.diff.x := (REAL32 ROUND(rx - lx))                  TRUE
  ...  calibrate r.diff.x -    -     -     -             r.diff.x:=r.diff.x /
  r.diff.y := (REAL32 ROUND(ry - ly))                             calib.factor.x
  ...  calibrate r.diff.y                            }}}
  real.square := ((r.diff.y * r.diff.y) + (r.diff.x * r.diff.x))
  width := POWER( real.square, 0.5(REAL32))
```

(d) Undercut

The undercut of the weld is calculated for each side of the weld and is measured from the calculated parent metal line, where, and if, the weld profile line falls below the line, between the corners of the weld. The results images show any areas of undercut shaded in dark grey.

```
PROC calc.left.undercut(VAL [512]INT line, VAL[512]REAL32 perp.diff.left,
                        VAL INT left.start.x, right.start.x, REAL32 left.undercut,
                        INT left.undercut.x, left.weld.x)

  #USE pms_vals                                            {{{  Vars
  ...  Vars   -     -     -     -     -     -     -        BOOL in.undercut:
  SEQ                                                      REAL32 undercut:
    ...  init vars                                         }}}
    SEQ x = left.start.x FOR ((right.start.x - left.start.x)/2)
      SEQ
        IF
          (perp.diff.left[x] < 0.0(REAL32))
            {{{  undercut present - remember it if the biggest so far
            SEQ
              in.undercut := TRUE
              undercut := perp.diff.left[x]
              IF
                (undercut < left.undercut)
                  SEQ
                    left.undercut := undercut
                    left.undercut.x := x                   {{{  no undercut
                TRUE                                       IF
                  SKIP                                       (in.undercut = TRUE)
            }}}                                                SEQ
          TRUE                                                   left.weld.x:=x
            ... no undercut      -     -     -     -             in.undercut:=FALSE
    {{{  reject result if at middle of the weld                TRUE
    IF                                                           SKIP
      left.undercut.x >= ((left.start.x +                  }}}
                ((right.start.x - left.start.x)/2)) - 5)
        SEQ
          undercut := 0.0(REAL32)
          left.undercut := 0.0(REAL32)
          left.undercut.x := 0
          left.weld.x := 0
      TRUE
        IF
          left.undercut < 0.0(REAL32)
            left.undercut := (-left.undercut)
          TRUE
            SKIP
    }}}
:
```

(e) Reinforcement

The reinforcement, shown in figure 4.3, is essentially the maximum perpendicular distance of the profile line, above the calculated weld base line, between the left and right corner points of the weld. It also forms the measurement of throat thickness of a fillet weld. The base line is number "7" on the results images.

```
PROC calc.height(VAL [512]INT line, VAL [512]REAL32 perp.diff.base,
                  VAL INT left.start.x, right.start.x,REAL32 height)

  ...  vals and vars
  SEQ
    ...  init vars
    SEQ x = left.start.x FOR (right.start.x - left.start.x)
      SEQ
        this.height := perp.diff.base[x]                 {{{  find max and min heights
        ...  find max and min heights -      -           IF
    {{{  assign value to returned height                   (this.height > max.height)
    IF                                                       SEQ
      max.height >= (-(min.height))                            max.height:=this.height
        SEQ                                                (this.height < min.height)
          height    := max.height                            SEQ
      TRUE                                                     min.height:=this.height
        SEQ                                                TRUE
          height    := min.height                            SKIP
    }}}                                                  }}}
:
```

(f) Displacement between the plates

The plate displacement is shown in figure 4.3 and is measured to be the distance between the two lines of parent plate along a line perpendicular to the weld base line which bisects the cord between the left and right corners of the weld. The line along which it is measured on each image in the results is shown and labelled "11".

```
...  Vars
SEQ
  x.mid := right.start.x - left.start.x
  y.mid := line[right.start.x] - line[left.start.x]
  IF
    (m.base > (-m.nearly.horiz)) AND (m.base < m.nearly.horiz)
      {{{  calc plate displcement
      SEQ
        x.left  := x.mid
        y.left  := ((m.left * x.left) + (c.left * m.scaler)) / m.scaler
        x.right := x.mid
        y.right := ((m.right * x.right) + (c.right * m.scaler)) / m.scaler
        plate.displacement := (REAL32 ROUND(y.left-y.right))
        ...  calibrate plate.displacement
      }}}
    TRUE
      {{{  calc plate displacement
      REAL32 diff.x, diff.y, square:
      SEQ
        c.perp := y.mid + ((x.mid * m.scaler) / m.base)
        {{{  find x.left and y.left
        IF
          (m.left > (-m.nearly.horiz)) AND (m.left < m.nearly.horiz)
            SEQ
              y.left := c.left
              x.left := x.mid
          TRUE
            SEQ
              y.left := INT( ( ( (INT64(m.left*m.base))*(INT64(c.perp)) ) +
                                  (INT64(c.left*m.scaler.sqrd)) ) /
                                (INT64(m.scaler.sqrd+(m.left*m.base) ) ))
              x.left := ((y.left - c.left) * m.scaler) / m.left
        }}}
        ...  find x.right and y.right                          {{{  calibrate diff.x
        diff.x := (REAL32 ROUND(x.left-x.right))               IF
        ...  calibrate diff.x   -    -    -    -                 diff.x = 0.0(REAL32)
        diff.y := (REAL32 ROUND(y.left-y.right))                   SKIP
        ...  calibrate diff.y                                    TRUE
        square := (diff.x*diff.x)+(diff.y*diff.y)                  diff.x := diff.x /
        plate.displacement:=POWER(square,0.5(REAL32))                    calib.factor.x
      }}}                                                      }}}
```

(g) Leg length

The leg lengths of a fillet weld are the two distances measured from a point within the weld where the lines of parent metal cross to the edges, or corners of the weld. This measurement is made on the gauge provided that the intersection point of the two parent lines occurs between the corners of the weld in the x direction. If this condition is not met, the weld is concluded not to be a fillet weld and the leg lengths are returned as zero.

```
                                          {{{  vars
...  vars     -     -     -     -         INT lx,rx,ly,ry,cx,cy:
SEQ                                       INT square:
  lx := left.start.x                      REAL32 real.diff.x, real.diff.y, real.square:
  rx := right.start.x                     }}}
  ly := line[lx]
  ry := line[rx]                          {{{  set leg lengths = 0
  IF                                      SEQ
    m.left = m.right                        left.leg  := 0.0(REAL32)
      ...  set leg lengths = 0              right.leg := 0.0(REAL32)
    TRUE                                  }}}
      SEQ
        cx := ( (c.right - c.left) * m.scaler) / (m.left - m.right)
        cy := ((m.left * cx) / m.scaler) + c.left
        IF
          (cx < lx) OR (cx > rx)
            ...  set leg lengths = 0
          TRUE
            SEQ
              {{{  calc left.leg
              SEQ
                real.diff.x := (REAL32 ROUND(cx - lx))
                {{{  calibrate real.diff.x
                IF
                  real.diff.x = 0.0(REAL32)
                    SKIP
                  TRUE
                    real.diff.x := real.diff.x / calib.factor.x
                }}}
                real.diff.y := (REAL32 ROUND(cy - ly))
                ...  calibrate real.diff.y
                real.square := ((real.diff.y * real.diff.y)
                                   + (real.diff.x * real.diff.x))
                left.leg := POWER( real.square, 0.5(REAL32))
              }}}
              ...  calc right.leg
```

4.4.2 Calibrating the gauge

The calibration of the gauge is applied to the measurement head, and, therefore, would change for different heads. Measurement is made by counting pixels on the image and then scaling them in horizontal and vertical directions using predetermined factors. The calibration is described in the following sections with the software used.

4.4.2.1 Calibration

A calibration facility is offered on the gauge which requires a calibration sample to set the calibration factors in the gauge, shown in plate 4.7. This allows different lenses to be fitted to the head or for different heads to be used. The gauge has standard calibration factors set for the head as it has been assembled, thus no calibration is needed unless the head is changed.

The calibration sample consists of a brass block machined to have a 5 mm step of 30° or 45°. The calibration algorithm then looks for the two corners of the block from which the number of pixels in the x and y directions give the calibration factor in each direction. These are 23.0 for x and 19.0 for y giving the resolution of the system horizontally to be 0.043 mm / pixel and 0.051 mm / pixel vertically for the head shown with the gauge in plate 4.2. Images 4.5 and 4.6 shows raw and processed images of the 30° calibration block giving the pixel calibration factors of 27.71 mm in the x direction and 22.40 mm in the y direction.

```
SEQ
  IF
    calibrating = TRUE -- ie calibration
      {{{  calc x & y calibration factors with 45 deg block 5mm high & wide
                                          {{{  Vals
      ...  Vals     -     -     -       { VAL REAL32 real.block.width IS 5.0(REAL32):
      SEQ                                 VAL REAL32 real.block.height IS 5.0(REAL32):
        ...  zero results                 INT block.width,block.height:
        IF                                }}}
          pms.error = 0
            {{{  do calibration
            SEQ
              block.width:=(right.corner-left.corner)
              ...  find mod of block width
              block.height := weld.line[right.corner]-weld.line[left.corner]
              ...  find mod of block height
              calib.factor.x := (REAL32 ROUND(block.width))  / real.block.width
              calib.factor.y := (REAL32 ROUND(block.height)) / real.block.height
            }}}
          pms.error = 7
            ...  reset.calibration factors to standard
          TRUE
            samples := samples - 1
      }}}
    TRUE
      SKIP
```

```
{{{  find mod of block width                 {{{  find mod of block height
IF                                           IF
  block.width < 0                              block.height < 0
    block.width := (-block.width)                block.height := (-block.height)
  TRUE                                         TRUE
    SKIP                                         SKIP
}}}                                          }}}
```

It is likely that on a production version of the gauge interchangeable measurement heads would be calibrated during manufacture and distributed with the factors. If these were programmed into each head and read by the gauge there would be no need for the user to perform calibration or to type in the value.

4.4.3 The user interface

The user interface provides the software to read the keyboard and measure switch, on the head, update the screen, transmit results to a serial port and run the image processing software. Low level routines have been written to interface all the user interface hardware to the processor in order to make the measurement as easy to perform as possible. All the software for the user interface, as well as for the measurement with the structure and most important parts described in the following sections.

4.4.3.1 Main program loop

The procedure below shows the top level of the program running on the portable gauge. After the system is initialised and tested the main program loop then waits for a key to be pressed while flashing a cursor. Once a key has been pressed the main part of the program is initiated.

```
PROC gauge(CHAN OF ANY to.null,from.null)
  ...  vals, vars, libs, chans, procs etc
  SEQ
    ...  switch off all leds and laser
    ...  set display screens, fcb etc.   and wait for key
    ...  init vars
    WHILE TRUE
      SEQ
        ALT
          ...  get char -- THIS IS THE MAIN BIT !!!
          ...  flash cursor
        ...  do display
:
```

4.4.3.2 The main part of the program

When a key is pressed, it is received by the program along the channel "from.kpad", as can be seen below, and a series of actions is taken according to which key is pressed, and the function selected. Viewing plate 4.2, which shows the completed portable gauge, the special function keys can be seen with their corresponding function on the key cap. By pressing the key marked "RUN" the fold "... measure weld" is run causing measurement as described in section 4.4.1.3(3) . Likewise, on pressing the key marked "PRN" the code in "... info to printer" is run and results in the input data being printed.

If the key which has been pressed is not a special function key the letter to which it corresponds is put in the current field at the position of the cursor. This action is performed in the fold "... text in field arrays".

```
{{{  get char -- THIS IS THE MAIN BIT !!!
from.kpad ? junk
  SEQ
    ...  get rid of cursor
    char := INT(junk)
    IF
      ...  measure weld
      ...  set welding code data
      ...  display logged data
      ...  display parameters
      ...  info to printer
      ...  change field
      ...  text into field arrays
      ...  switch display.mode
      ...  skip
    ...  up date mode field 6 on to display
    ...  up date mode field 7 on to display
    ...  up date mode field 8 on to display
    ...  set y cursor position
    ...  set x cursor position
}}}
```

4.4.3.3 Weld measurement

When key "RUN" is pressed the hexadecimal character number "85" ("86" if "CAPS" and "RUN" are pressed together) is transmitted from the keyboard interface hardware to the transputer. The software, which is shown below, is activated and following some initialisation software, the laser diode in the measurement head is switched on. This is performed by writing to a register in the transputer's memory, connected to the hardware controlling the laser. This is achieved in software by writing to the variable "led" which is 32 bits wide and is placed in the memory.

The variable "do.measure" has been set to "TRUE" in the initialisation sequence causing the measurement to be continuous, unless it is set to "FALSE" in the "mode control" fold or by the "RUN" key being pressed again to end measurement.

```
{{{  measure weld
(char = #84) OR (char = #85) -- F2/shiftF2
  ...  Vals and vars
  SEQ
    ...  init vars
    ...  set up pms.display
    ...  print header
    ...  switch on laser
    WHILE do.measure = TRUE
      SEQ
        ...  mode control
        ALT
          ...  get char
          ...  do measurement
    ...  switch off laser and single leds
    ...  set logging
    ...  put on number of samples
    ...  up date mode field 11 on to display
}}}
```

4.4.3.4 Mode control

Four modes of operation are available for measurement, namely: positioning, single shot, continuous and calibration. The mode is selected on the main screen prior to measurement. The modes are described below.

```
IF
  mode = 0 -- Position head with no measurement
    ...  position head
  mode = 1 -- Single measurement mode
    ...  set single shot mode
  mode = 2 -- Continuous measurement mode
    pms.error := 0
  mode = 3 -- Calibration measurement mode
    ..  set calibration vars
  TRUE
    SKIP
```

(1) Position mode

The positioning mode is intended for use prior to the continuous mode and allows the user to align the measurement head over the weld without measurement occurring. The advantage of the positioning mode is the fast update of the display, as no time is taken in finding the line and calculating the measurements. The display shows a thresholded picture of the image from the camera, as can be seen in plate 4.8.

```
{{{  vals and vars
INT x,y,proc.frame,lcd.x,lcd.y:
}}}
SEQ
  ...  capture image into the next frame
  {{{  display thresholded image
  SEQ
    y := 0
    lcd.y := 0
    WHILE y < resolution
      SEQ
        lcd.y := y / 8
        x := 0
        lcd.x := 0
        ...  horz line at this y-
        y := y + 8
  }}}
  pms.error := 9
```

```
 {{{  horz line at this y
WHILE x < resolution
  SEQ
    lcd.x := x / 3
    IF
      image[proc.frame][y][x] <
                      (BYTE(pos.threshold))
        pms.display[lcd.y][lcd.x]:=1(BYTE)
      TRUE
        pms.display[lcd.y][lcd.x]:=0(BYTE)
    x := x + 3
}}}
```

(2) Single shot mode

The single shot mode has two states of operation; firstly, the gauge continuously captures images and calls the measurement software with an error code of "9", which signifies that no measurement is required. This produces a single pixel line on the display which allows the user to position the measurement head over the weld at the desired point. The positioning stage continues until the red button on the measurement head is pressed, at which time the gauge captures an image and measures the profile. The gauge now pauses, displaying the results of the measurements until either the red button is pressed again or the "RUN" key is pressed to end the measurement.

The red button on the measurement head has two modes of operation, if it is pressed and held on, the gauge measures the profile which it captures, but if the button is depressed and released quickly the gauge returns to the positioning state.

```
...  vals and vars
SEQ
  ...  capture image into the next frame
  {{{  turn on ready led and check if key is pressed
  SEQ
    IF
      pms.error = 0
        pms.error := 9
      TRUE
        SEQ
          IF
            (switch /\ #01) = #01 -- not pressed
              SEQ
                safe.led := safe.led /\ (~single.off)
                led := safe.led << 8
                pms.error := 9
            TRUE
              SEQ
                safe.led := safe.led \/ single.off
                led := (safe.led << 8)
                pms.error := 0
  }}}
```

(3) Continuous mode

The software extract in section 4.4.3.4 "Mode control" shows that no action has to be taken to result in continuous measurement mode as this is the default mode. The error code is set to "0".

(4) Calibration mode

In order to force the system to re-calibrate from the profile of the calibration sample, a number of software switches are set. The Boolean variable "calibrating" is set to "TRUE", but the continuous action caused by the variable "do.measurement" is switched off by setting it to false. The error code passed to the measurement software is set to "0".

```
SEQ
    calibrating := TRUE
    do.measurement := FALSE
    pms.error := 0
```

4.4.3.5 "Get char"

Once the mode configuration has been set, the program either runs the measurement software or responds to a key press. If a key is pressed one of the following actions may be taken: the options either allow measurement to finish and produce a graphic print of the screen, or varies the measurement parameters shown in the following software extract.

```
from.kpad ? junk
  SEQ
    set.up.display(to.display, from.display)
    char := INT(junk)
    IF
      ...  finish monitoring
      ...  print screen
      ...  increase threshold
      ...  decrease threshold
      ...  increase search.length
      ...  decrease search.length
      ...  increase window.width
      ...  decrease window.width
      ...  increase corner.adjust
      ...  decrease corner.adjust
      ...  increase sample.limit
      ...  decrease sample.limit
      ...  switch display.mode
      ...  pause
      ...  skip
```

The measurement parameters are altered by increasing, or decreasing, their value according to the key pressed. An example of this is shown in the fold of "`...  increase search length`" which follows.

```
{{{  increase search.length
(char = (INT('w'))) OR (char = (INT('W'))) -- UP/shiftUP
  IF
    search.length < 50
      search.length := search.length + 1
    TRUE
      SKIP
}}}
```

4.4.3.6 "Do measurement"

The code just prior to the measurement software being called switches on the LED which signifies that measurement is taking place. This is performed in the same way as the laser is switched on and off, but using a different bit in the same register. Once the measurement has finished, this LED is switched off. If calibration is to be performed the calibration factors are set otherwise this fold is ignored. The text and graphics parts of the display come before the results and measurement information is put on the screen.

```
{{{  do measurement
TRUE & SKIP
  ...  vals and vars
  SEQ
    ...  switch on measure led
    ...  call pms program
    ...  switch off measure led
    ...  do calibration
    ...  clear pms graphics display and put on least squares lines
    ...  reset samples if > 10000
    ...  put on number of error and adjust if flat weld
    ...  do results
    ...  finish if data logging full
    ...  display screen
    ...  wait for key if single shot
    ...  if calibrating, ask if this calibration is ok
}}}
```

The software then continues to cycle if in continuous mode, or returns to the main display if not.

4.5 Tests on, and results from the gauge

The next section describes the tests which were used to validate the measurements produced by the gauge. These were devised to test the accuracy to which the measurements can be taken, and are provided to two decimal places so that fluctuations can be seen. Test blocks as well as real welds were used. The results are shown in section 4.10, where images of each weld are available for comparison with the numerical results. Additionally graphs of some of the specimen welds showing the fluctuation during the recording of 100 samples at the same location on the weld are provided. These results are then discussed in section 4.5.2 .

4.5.1 The test measurements

The three tests are considered below and firstly test the gauge against a standard block, of weld type profile, which has been machined to known dimensions. Secondly, the gauge measured some specimen weld samples, which had previously been measured, for comparison with existing manual gauges. Finally each of the specimen welds, and the standard block sample, were each measured 100 times, at the same location on the weld, recording consecutive results.

4.5.1.1 Measurement of standard block

A set of standard blocks which provide a defined profile have been made from brass on a precision machine tool. These blocks each have a uniform sloped step profile of a range of angles at 10°, 20°, 30°, 45° and 60° and are shown in plate 4.7 . The blocks served two purposes: to calibrate the measurement head and gauge, and to provide a means of validation of the results gained. When the gauge is calibrated using one sample then it produced the correct results for itself and the others. Measurements of consistency using one of the standard blocks is discussed in section 4.5.2.1 with data in section 4.10.

4.5.1.2 Validating the measurements against other gauges

The gauges described in section 4.2 were all less precise than the portable gauge under test, however the use of two six inch steel rules and a protractor, used with great care, provided a means by which the values produced by the gauge could be compared. These measurements, with the exception of toe angle, were taken in each case before those made by the automatic gauge. Both the manual and gauge measurements are recorded with their corresponding processed images in section 4.10.

The manual measurements were taken from a point on the weld as close as possible to that of the gauge. The position was drawn onto the weld with a spirit marker and fine pencil line, but contained an assessed tolerance of 0.5 mm in the two positions.

(1) Technique of manual measurement

The combined use of two thin steel rules, a protractor, and pin was found to provide the most accurate manual measurement of the weld features in most cases, but the welding institute and profile gauges were also used.

(a) Width

Measurement of width was the simplest and recorded by laying the rule across the top of the weld at right angles and viewing it from vertically above taking care to reduce any error of parallax. This measurement was not made for fillet welds.

(b) Reinforcement

On the larger weld samples the Welding Institute gauge was used according to its instructions, but for the smaller welds by careful placement of a steel rule on edge across the weld, which was held in a vice, the reinforcement could be gauged by comparing the scale on the rule with the gap under it.

(c) Undercut

The measurement of clearly visible undercut was made using a pin placed into the undercut to mark a point judged to be its maximum depth, and a rule placed on edge across the parent plate, butting up to the weld. The depth of the undercut was thus highlighted and could be judged by comparing the length of the pin exposed beneath the rule with its scale. This method was compared with results gained by the Welding Institute gauge and assessed to be more accurate, but much slower. Imperceptibly small undercut, of less than about $^1/_4$ mm could only be checked by scratching a finger nail over the weld toe.

(d) Plate displacement

This could not be measured manually on all samples except those which were a lap type weld. Distinction between plate displacement and plate angle could not be made with the manual tools as the measurement is made within the weld itself.

(e) Toe angle

This proved to be a controversial measurement and most difficult to access. The accuracy to which it could be made was +/- 10°. The method used relied on fitting the corner of a rule into the toe of the weld and positioning it to sit on the weld. Measurement was then taken with a 180° protractor resting normally across the weld such that the rule edge passed through the centre of the protractor. The measurement of toe angle could not be made for fillet welds.

(f) Leg lengths

The leg lengths of the 90° fillet welds were most efficiently measured by the Welding Institute gauge, but on the sample of plates at about 130° an error would be introduced by using this gauge.

4.5.1.3 Testing the results for consistency

For each of the seventeen weld specimens, 100 samples were taken at the same single location as the manual measurements. These samples were consecutively measured to highlight any variation in results and thus to give a value for the consistency of the gauge's measurements. The set of 100 samples for each weld was processed to give the distribution of values, and the mean, mode and maximum deviations from the mode calculated. These data sets are tabulated in section 4.10, and shown graphically for selected specimens to demonstrate the fluctuation over the 100 samples. Additionally graphs of the frequency of occurrence of the values of weld toe locations are provided to show the consistency.

4.5.2 Discussion of results

The gauge has been tested on many different types of weld samples, including butt, fillet, lap and pipe welds. A selection of these is shown in plate 4.9 . Detailed results of some of these is discussed in this section. Reference should be made to the images, specimen sample data and corresponding graphs in section 4.10. The images show both processing information leading to the weld features and measurement locations. In association with the images data for comparison of the manual and gauge measurements is logged together with the set up and adjustment factors set for each weld.

Image 4.6 shows an example of the profile of the 30° standard block. The graph 4.1 shows that at the location "left.start.x" 145 pixels from the left side of the image, the variation of the location of this feature (marked on the image as "8") is random over the sample and between +2 and -3 of the mean value.

Graph 4.2 shows the corresponding measurements at each sample calculated from the start locations in graph 4.1 . It can be seen that fluctuations in the start locations directly affect the calculated measurements.

The tabular results show in the first four columns the pixel locations with frequency of occurrence of the block start points. The other columns show the calculated values and frequencies which show their consistency. Below each column the mean, mode, maximum and minimum values and maximum deviations from the mode.

4.5.2.1 Standard block

The start locations are all within 5 pixels of the other values. The calibration factors, image 4.6, show that each pixel is $^{1}/_{27.71}$ mm wide and $^{1}/_{22.40}$ mm high giving a maximum variation on any point as 0.18 mm. However, only the left.start.x location has this variation and the other locations have a variation of one pixel, giving in the x direction 0.036 mm. Consequently a maximum variation in measurement is less than 0.216 mm. The variation in the left.start.x location above that of the rest may be attributed to the block; the inside corner which it is finding is less sharp a feature than the upper outside corner.

The 30° block has a step of 5 mm, giving a width of 10 mm. Comparing this with the measured results from the gauge shows a width of 9.8 mm, varying by 0.2 mm, and displacement (step) of 5.1 mm with a variation of 0.2 mm.

On this performance the gauge has a similar accuracy when making linear measurements to a vernier calliper calibrated to $^1/_{10}$ mm.

The plate angle of the block, between the upper and lower "parent" surfaces is 180°. The gauge measured this to between 179.3° and 180.6°, giving an accuracy of 0.01%.

The lines of fit (toe.length, see section 4.4.1.4(3)) for the toe angles, for the block, were lengthened to 40 pixels, which, as described, increase the accuracy of the angle measurement beyond that which could be achieved on a curved weld toe.

The standard blocks were no more accurate than the tools could be set during their manufacture.

4.5.2.2 Butt welds

The maximum deviation in finding the x start points for all ten butt welds (1000 samples) is shown in graph 4.15 to be 21 pixels. The worst case is the Welding Institute sample weld 3, "Root concavity". This sample has a very shallow transition between the parent plate and weld metal. Only a small amount of noise in the image or laser stripe will cause the least squares fit of the parent plate to produce slightly different fit. A small change over the length of fit will result in a much greater variation further from this section. Since the adjustment factors determining the start finding algorithm are tuned for a shallow weld this variation will cause a corresponding variation in the start location. Plate 4.10 shows the Welding Institute specimen plate 4 "Undercut".

4.5.2.3 Fillet welds

The measurement of toe angle on the fillet weld specimens is significantly less good than the results gained for butt welds. In both cases the calculated value is dependent upon the position at which the corner is found. The variation in the toe angle is dependant on where the start points of the weld are found from the first start point algorithm (deviation from the parent metal). The metal surface of the plate in this sample is not smooth and causes a wider and more noisy laser line. This has the affect of increasing the likelihood of the causing sight variation in the equations found for the parent metal between samples. The join between the weld and parent metal, in this specimen is fairly shallow, thus relying on the deviation from the parent plate to indicate an approximate start location. The second algorithm (to find the point of sharpest corner and relocate the start point to this position) is affected by the noisy line and will both vary the position of the corner and find different toe angles. Plates 4.11 and 4.12 show the specimen fillet weld with plate angle of 130° and for the multipass weld.

4.5.3 Summing up

The results show a worst case fluctuation in width readings, for the butt weld specimen, of 0.3 mm in a reading of 9.1 mm with the maximum deviation of 0.2 mm from the mode of 9.2 mm. This is an accuracy of 2.2%. However this represents only two samples in 1000 (there are 10 butt weld specimens). The measurement of weld profiles is performed to a higher degree of accuracy than by the manual gauges which, as has been stated in section 4.2.1, has an accuracy of only 10%.

The measurements for the fillet welds did not prove so consistant producing on the width measurements variations of 0.7 mm in 5.8 mm for sample "FIL1" and 1 mm in 8.7 mm in sample "FIL2". This produced a fluctuations of 12% and 11.5% in width measurement.

Toe angle showed the greatest variation in results, but it is felt that this is also the least reliable manual measurement. In the majority of specimens the fluctuation over the 100 samples was less than 10°, but varied on a couple of the specimens by as much as 18° from lowest to highest reading. This is still as good an accuracy as it is felt can be made manually.

The most consistant results were produced for undercut where all the measurements were within 0.1 mm of each other.

These figures are taken as if the user simply takes the results as the measurements, however the gauge displays a picture of the line of weld profile and where the weld start points have been found, see plate 4.3. The user then can discard readings in which the weld start points have been found wrongly.

4.5.4 The Welding Institute's evaluation of the gauge

The Welding Institute have had the portable gauge for evaluation for over six months and are using it to perform measurement more accurately and faster than previous measurement methods. In their report Stephens and Bourton [66] conclude:

> "The gauge has performed to a high enough standard in field tests to suggest that is capable of being manufactured and marketed commercially. However, it must be remembered that the gauge is a prototype and a number of essential alterations will have to be carried out. These concern the safety of the device and its flexibility in use (i.e. ability to cope with many types of joints.) A company will be sought to develop the gauge further and carry out many of the recommendations stated in this report. It is envisaged that TWI will receive a fixed percentage of the royalties from the product."

The changes which they specify in their report are divided into two categories, essential and recommended. The essential changes are:

1 Laser classification must be reduced to class 3A status
2 Measurement heads for various joint types and sizes are required
3 Software amended to cope with multi-pass welds
4 Software algorithm for measuring toe angle needs modifying
5 Battery power supply is needed for on-site use
6 A disk drive should be fitted
7 A glass cover fitted to the measurement head

The laser classification can be reduced from the current 3B in two ways, either the laser diode can be replaced by one which produces visible rather than infra-red light, or it can be replaced by one of less than 1 mWatt in power. There should be no effect on the operation of the gauge by doing this, since the software has been designed to find a very faint line. The gauge currently has only one head to demonstrate the principle, however this thesis shows that welds of a wide variety of sizes and types, including multi-pass, can be successfully measured. The variation of toe angle results from the short length over which it is measured and has now been rectified by increasing this length.

The recommended changes are:

1 Reduce the size and weight of the gauge
2 Larger font and black ink printouts
3 Rollers fitted to the head for ease of movement
4 Addition of a linear scale in the head and LCD screen
5 The head should be able to measure distance travelled
6 A handle should be fitted to the head for ease of use

All of these are refinements to the design and do not detract from the performance offered by the gauge.

In the report, comparison with existing gauges Stephens and Bourton agree with the view in this thesis that a full comparison cannot be made since several of the dimensions measured by the portable electronic gauge cannot be measured by existing methods.

4.6 Timing the gauge

The main parts of the software have been timed to give a break down of the measurement software processing requirement and to show the overall speed of the gauge. Timings have also been made showing the affect of the start point tuning factors. The timings are given in section 4.12, and include results for both the T425, in the gauge and for the same software running on TIPS with a T800.

4.7 The cost of the portable gauge

The cost sheet in section 4.11.1 gives a breakdown of the main items in the prototype gauge which totalled £1544. A production version is likely to have a component cost similar to the estimate of about £1800 shown in section 4.11.2 . However this is based on one off costs and if the gauge was to be produced in volume this figure would be reduced.

Certain design changes could be implemented to reduce the cost of components and have been included in the costing in section 4.11.2 . A new cheaper version of the Inmos T425 transputer is now available at under half the price. The T400 is compatible with the T425, but only offers two Inmos links. A second cost reduction can be implemented by not using the links for screen, keyboard, printer and data logging by memory mapping all of these peripherals into the address space of the transputer. An LCD panel is now available which would increase the resolution of the display from the current 256 x 64 to 480 x 128 pixels for less than the cost of the current display, at the time of development.

The production gauge would have all the necessary circuitry all on one custom printed circuit board, limiting the size of the gauge by only the size of the keyboard and ergonomic constraints. The measurement head could also now be made smaller for the same cost since miniature cameras, such as the pulnix TM 526, have become available.

4.8 Demonstration of on-line implementation

The opportunity to apply the system to measurement during welding became available as part of a proposal to guide automatic M.I.G. welding of cladding. It was proposed to mount a guidance head in front of the welding torch, in a similar way as Clark [4] had shown in T.I.G. welding guidance. In addition to the guidance, weld measurement was desired, on-line, while welding, to warn of the weld metal reinforcement above the level of the upper lap plate.

4.8.1 Description of the welding process

Very large plates of mild steel were to be completely clad on one side with strips of hasteloy. The production of this is a two part process where multiple rolls of hasteloy are unreeled across, and welded to, the flat steel plate, shown in figure 4.21. At the end of the first stage the steel plate has strips of hasteloy, welded along both edges, evenly spaced over its entire area. The second stage involves unreeling and welding more hasteloy to bridge the gaps where the mild steel is still exposed, see figure 4.22. This now leaves the steel completely clad with overlapping plates of hasteloy as can be seen in figure 4.23 . The steel base plate was about 12 metres wide and will be clad with about twenty-five hasteloy plates each about 0.5 metres wide. Plate 4.12 shows an example the completed lap weld with hasteloy strips on a mild steel base.

The discussion of the project is limited in this thesis to the proposal of on-line measurement.

4.8.2 The measurement system

The system to be used for the measurement was software similar to that described in section 4.4.1.3, used in the portable gauge, but running on a TIPS. The timings of each iteration are shown in section 4.10 using a T800 transputer. The T800 incorporates a floating point maths unit which speeded up the iteration time from the same parts of the software running on a T425 in the portable gauge.

4.8.2.1 Camera and optics

A measurement head similar in design to that described in section 4.3.4 was used, but a Pulnix miniature CCD camera replaced the cheaper Philips device. A 30 mW laser diode was used in place of the 5 mW model in the portable gauge to compensate for any intense arc light which may be present. The measurement head was mounted behind the torch with the camera facing away from the arc, positioning the laser about 180 mm behind the arc.

4.8.2.2 The software

The software was similar to that running on the portable gauge to find the weld start points and hence the measurements. The user interface software was not required. The same algorithms to find the start of the laser line, the weld profile and then the corners were employed. The measurement of the excess reinforcement was implemented by modifying the algorithm which calculates undercut. Instead of searching only for areas which dip below the parent metal line any area above the line was measured and renamed overfill. Once the overfill became greater than 1 mm a warning was indicated by TIPS.

4.9 Conclusion

The stated aim of the gauge was to measure fillet, butt and external fillet welds of sizes ranging from about 5 mm to 20 mm. The gauge should present the results in a clear and useful way to the weld inspector and produce a paper print out of these results for documentation. All of these aims have been achieved and it has been shown that the gauge can measure all the weld types required to an accuracy of better than 2.5 % for butt welds. Further to these basic aims the gauge has been used to log weld measurements and transfer them to another computer system for storage and post processing using its RS232 communications adapter.

Discussions are taking place to determine the feasibility of taking the prototype gauge described in this thesis to being a production model. In this case it could be considerably reduced in size by designing circuit boards with components mounted on both sides. The main limiting factor on the size reduction which could be achieved is that of the human interface; the screen must be large enough to see and of high enough resolution to be useful, and the keyboard must be big enough to use.

The accuracy of the gauge could be improved further by statistically analysing several samples for each measurement. The results shown in section 4.10 are for measurements calculated from the analysis of single images. If for each measurement a number of images were used and results which significantly deviated from the majority were disregarded, then using the rest to calculate the mean or mode to produce the measurements then the variation apparent in the single image results would be diminished. The timings shown in section 4.12 can be speeded up by using the parallel capability of the transputer. A combination of parallel and pipelined techniques with several transputers would reduce the time taken to find the results from several images to that similar to calculating the results from a single image.

It is known from the timings that the software requiring access to the image takes a maximum time of 0.034 + 0.034 + 0.101 = 0.169 seconds; this is derived from the maximum time to find the left and right ends of the line and then the time to find each pixel along the line with a window.width of 10 and search.length of 5. Dividing this processing on to two processors would reduce it to about 0.034 + 0.05 = 0.084 seconds; each processor finding one end and then half the line. This is now half the maximum speed at which images can be received. However since the image should be similar the end points need only be re-established that they are in the same place as the first image an the time to this would be much less. Thus with two transputers running at 20 MHz half real time is achievable by the image processing software.

The processing of the line to produce the measurements shows the advantage of the T800 transputer with its floating point processing. Using a T800-20 this part of the software takes about 0.154 seconds which when divided on to two processor could produce the results in 0.08 seconds, in line with the image processing discussed previously. If the measurements must be produced in less than a second then 12 images could be independently processed, and used to produce a more accurate result. The use of more processor can only increase the iteration time for each image to 40 ms (ie the time for each new image to be captured). Thus in one second the maximum number of independent images which could be analysed can be 25. This would be a significant increase in accuracy, but whether the measurement of welds requires this increased accuracy was doubted by the weld inspectors and welders.

Complete Occam source listing of the software is available on request to The University of Liverpool.

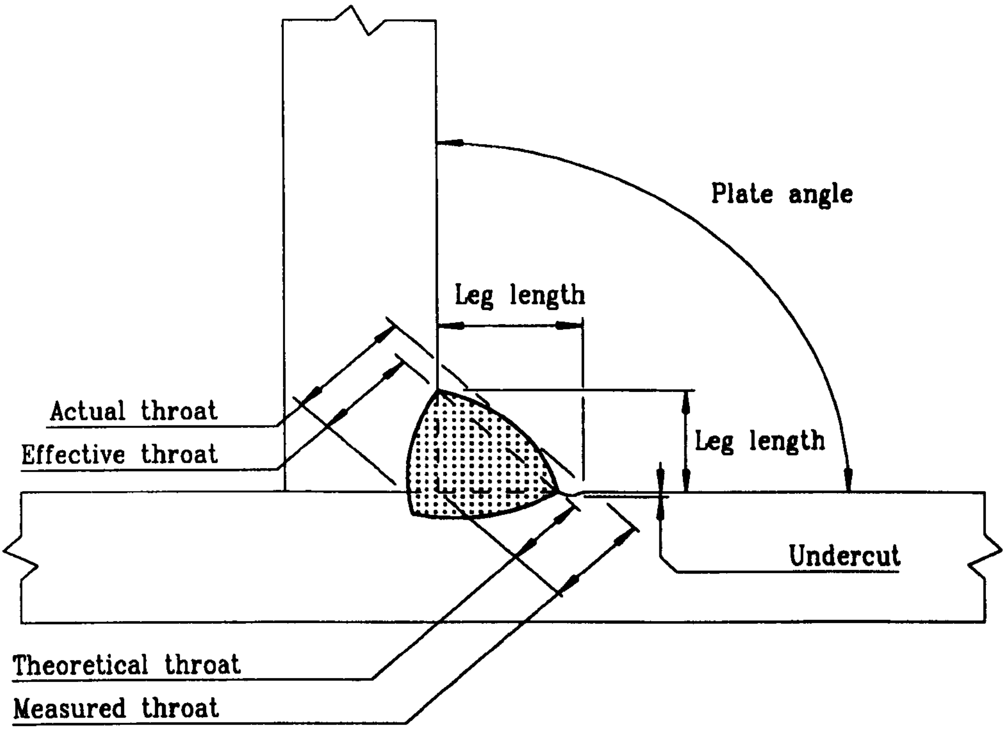


Figure 4.1 Fillet weld dimensions

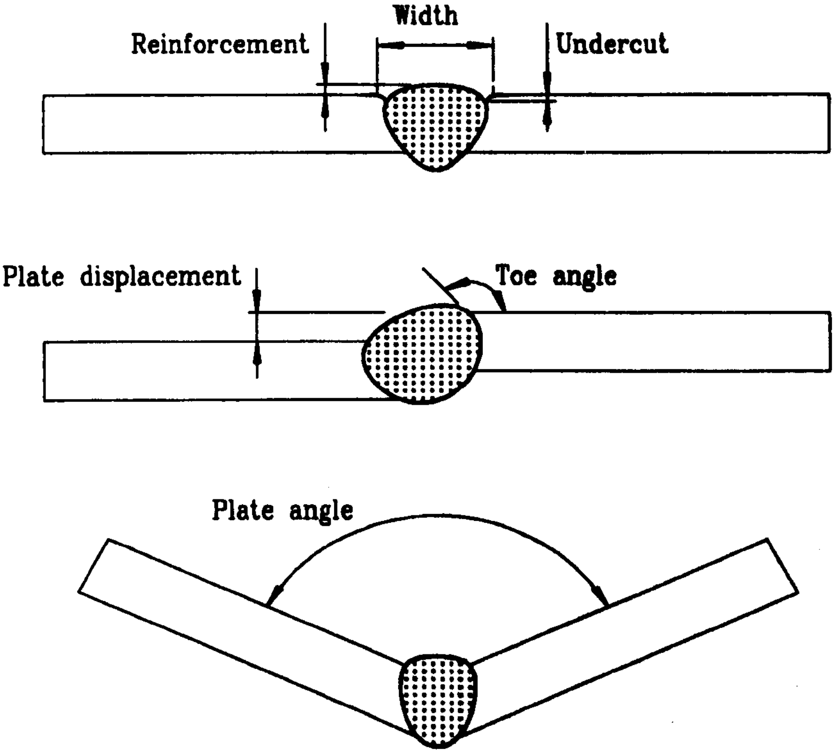


Figure 4.2 Butt weld dimensions

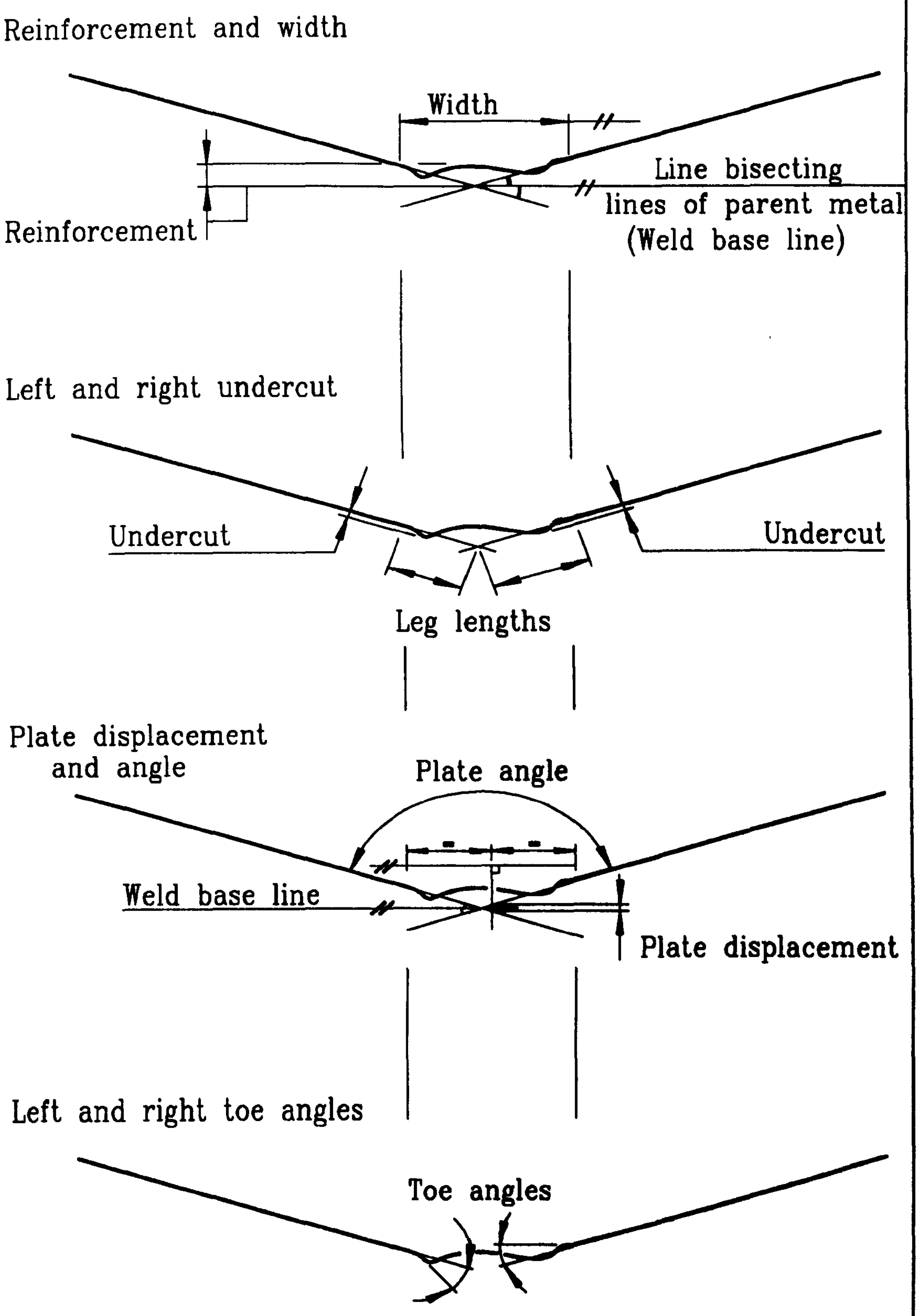


Figure 4.3 Measurements as defined in gauge

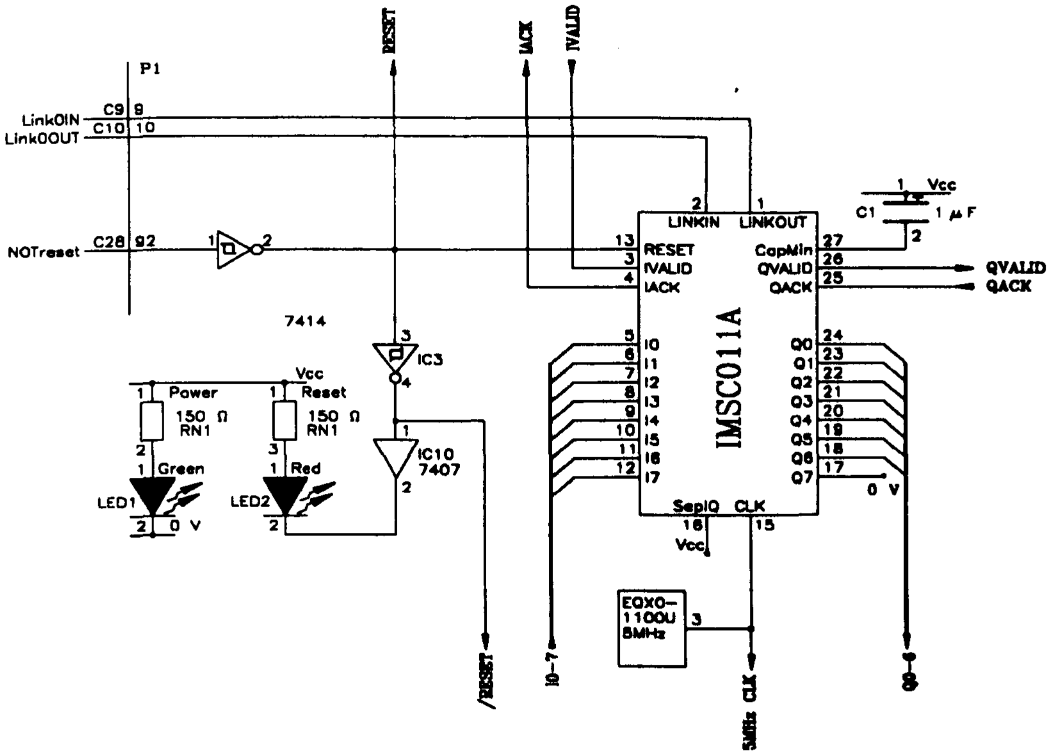


Figure 4.4 Link interface for printer and keypad

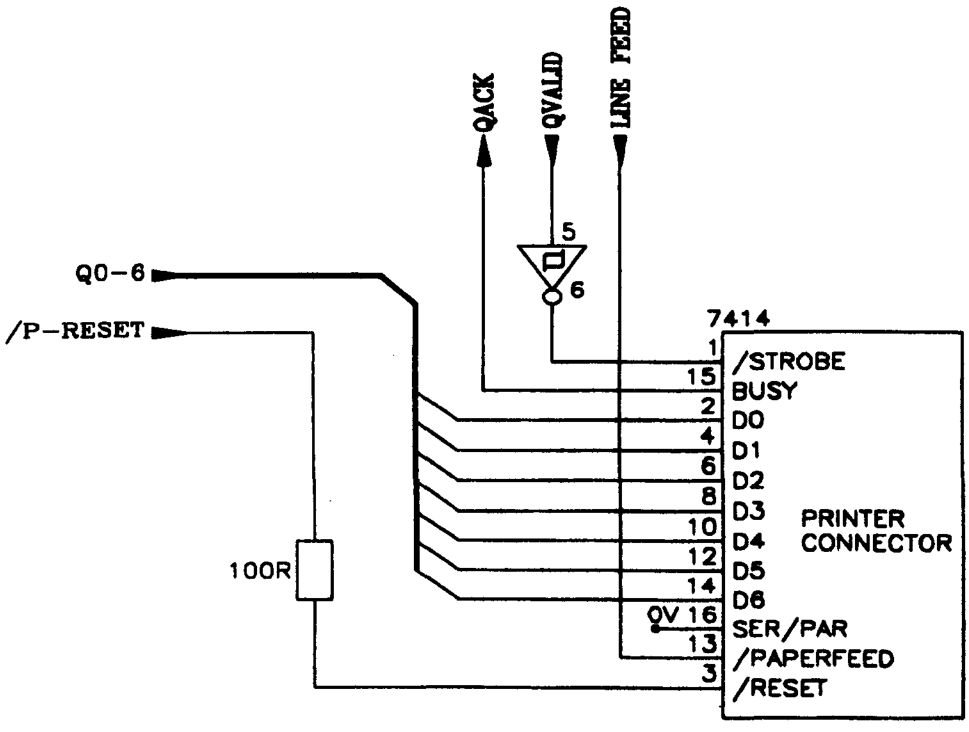


Figure 4.5 Printer interface

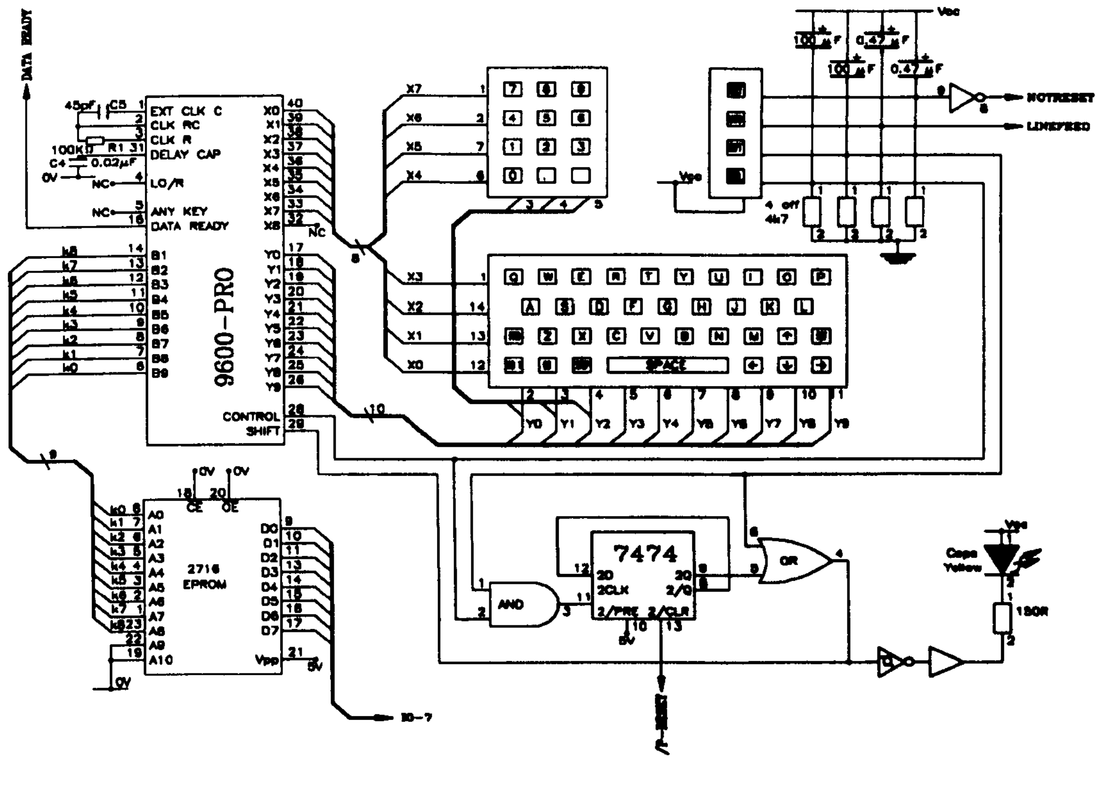


Figure 4.6 Keypad interface

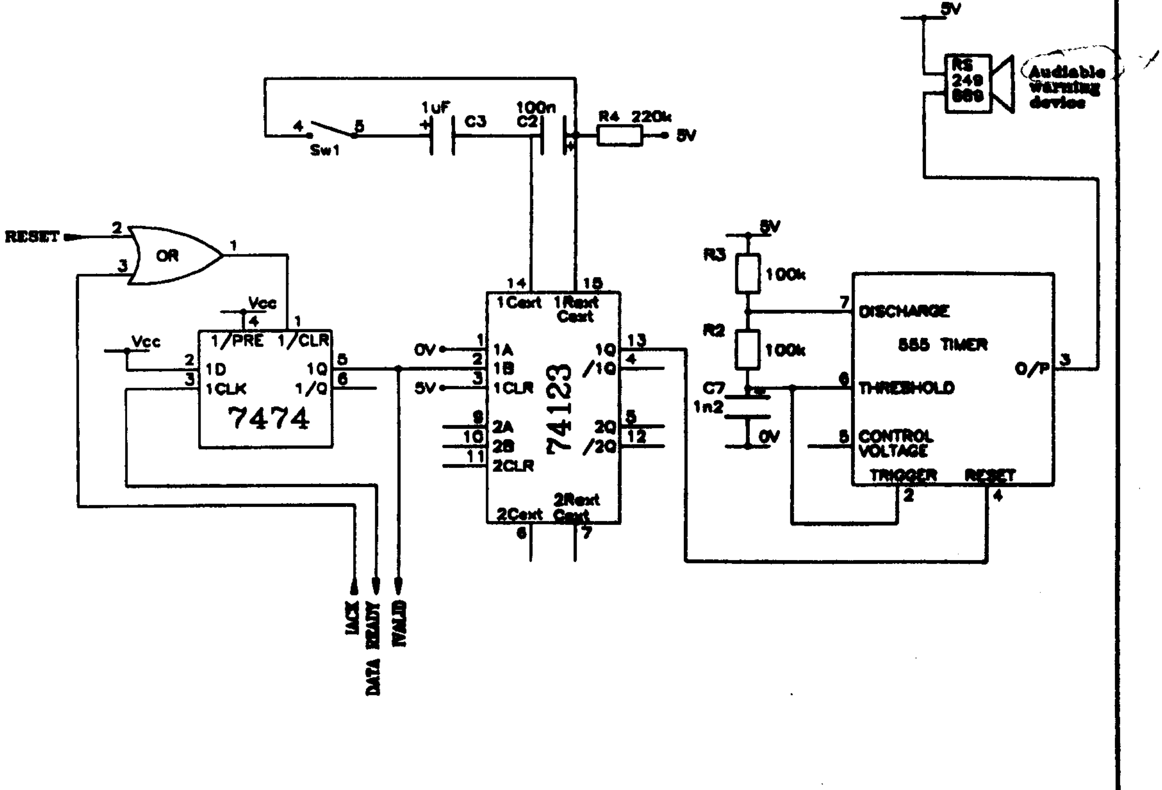


Figure 4.7 Keypad click circuit

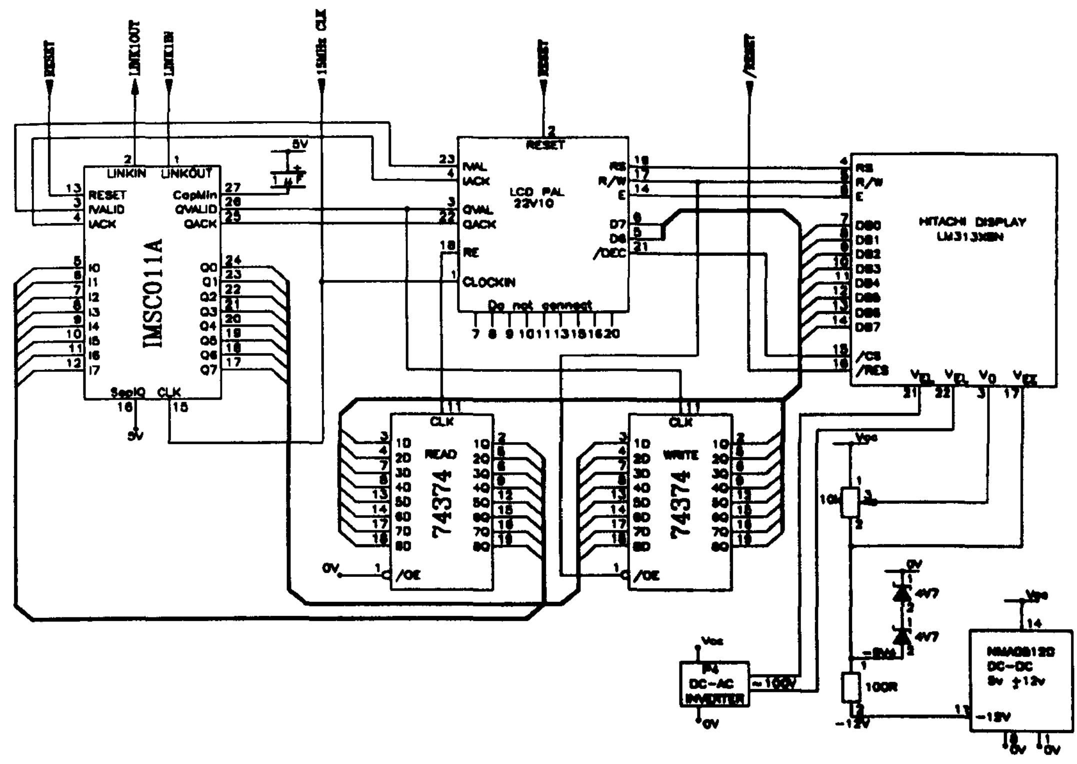


Figure 4.8 Link to liquid crystal display interface

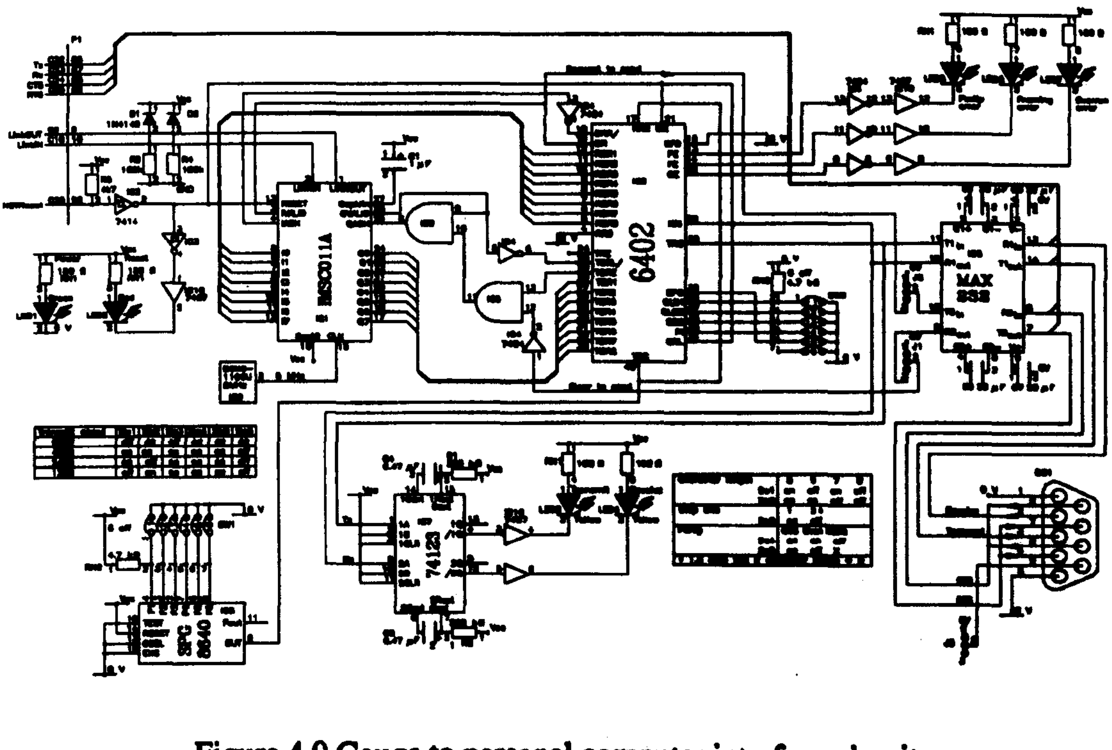
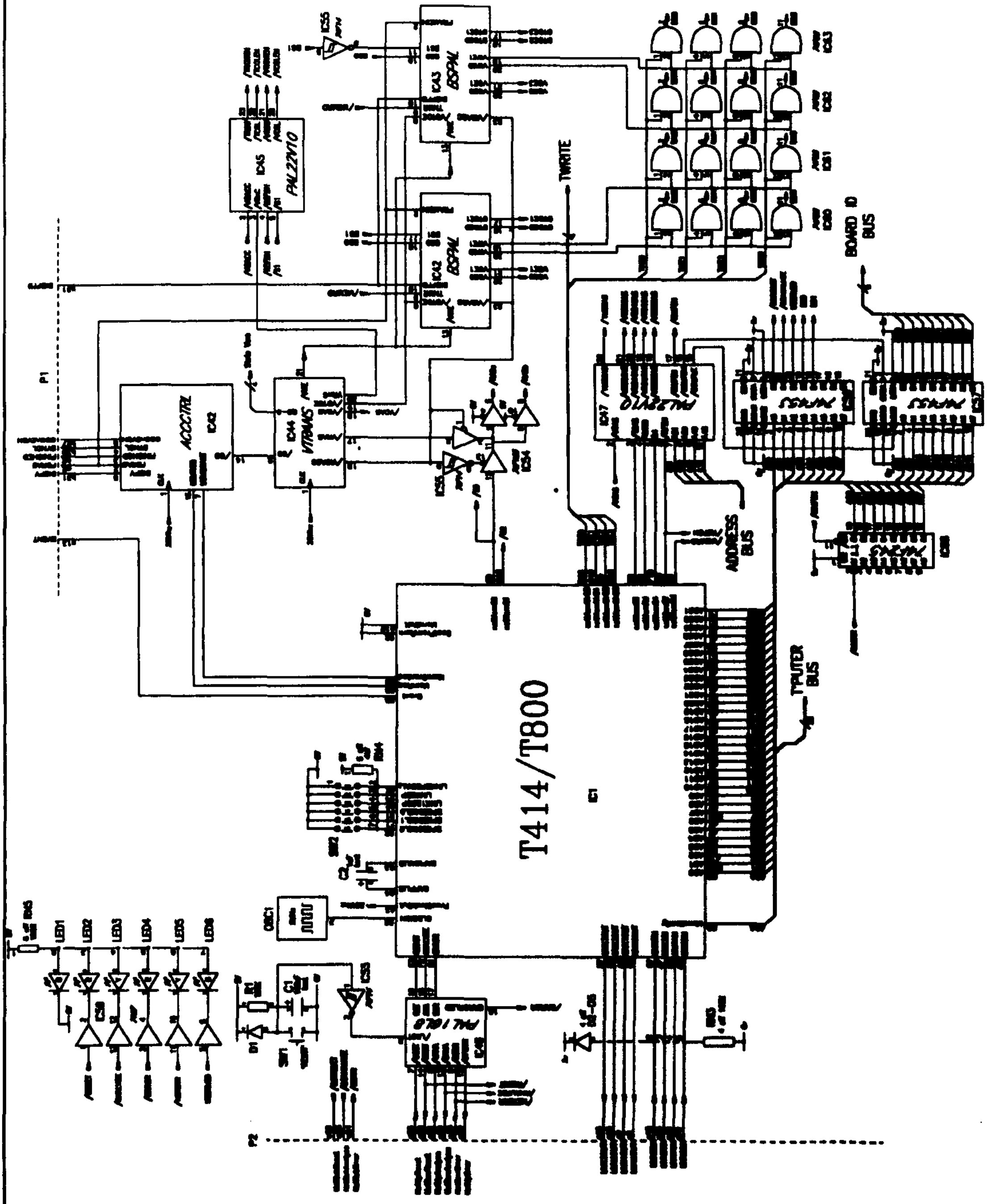
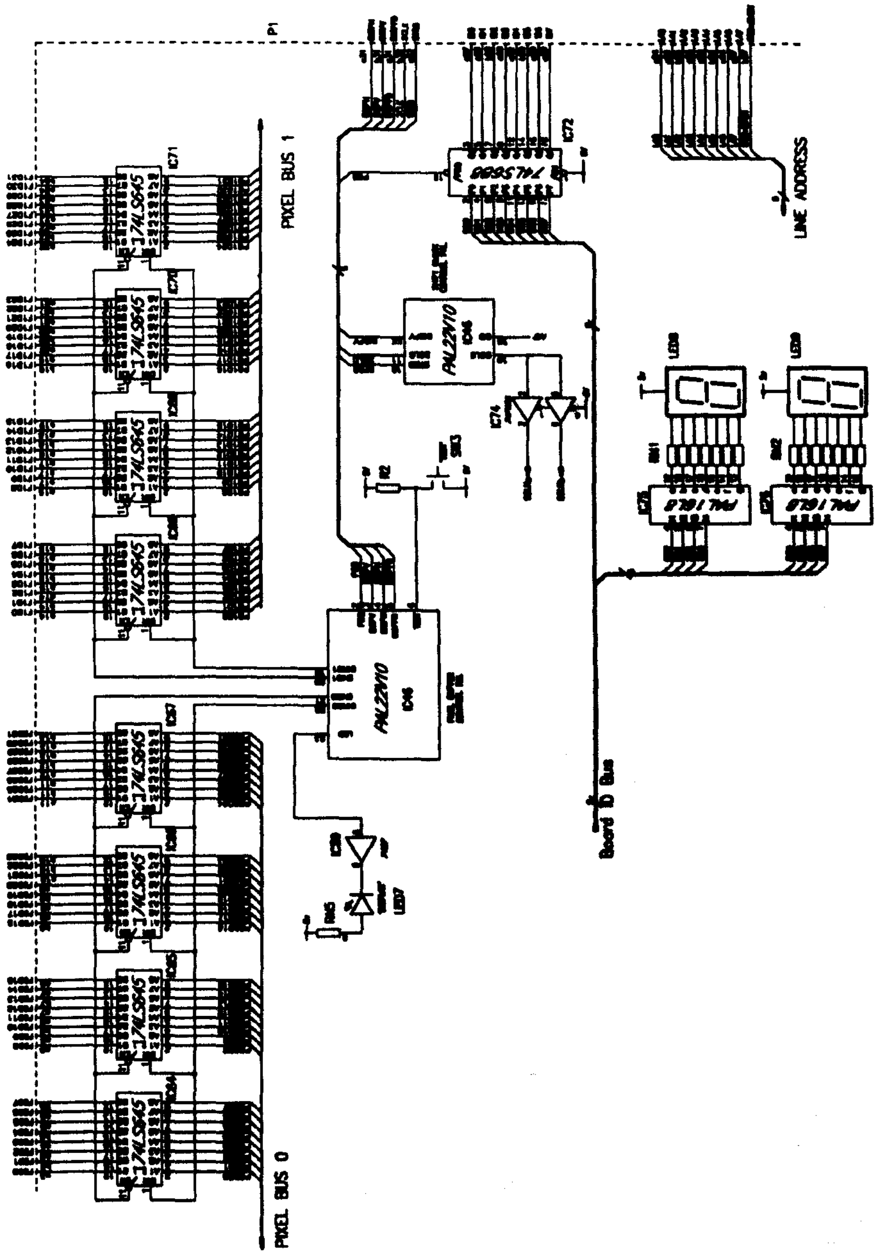


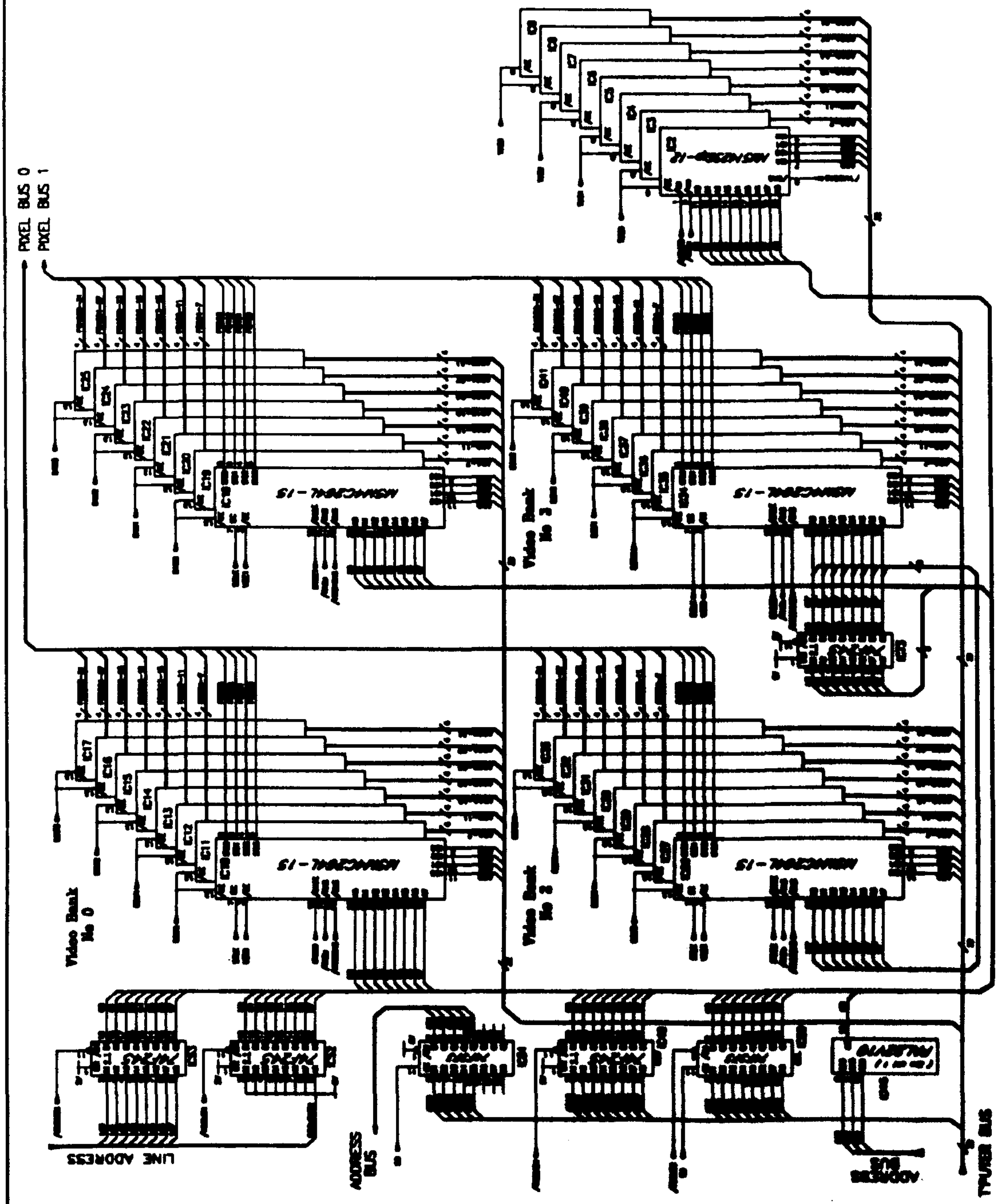
Figure 4.9 Gauge to personal computer interface circuit
(RS232 serial interface)



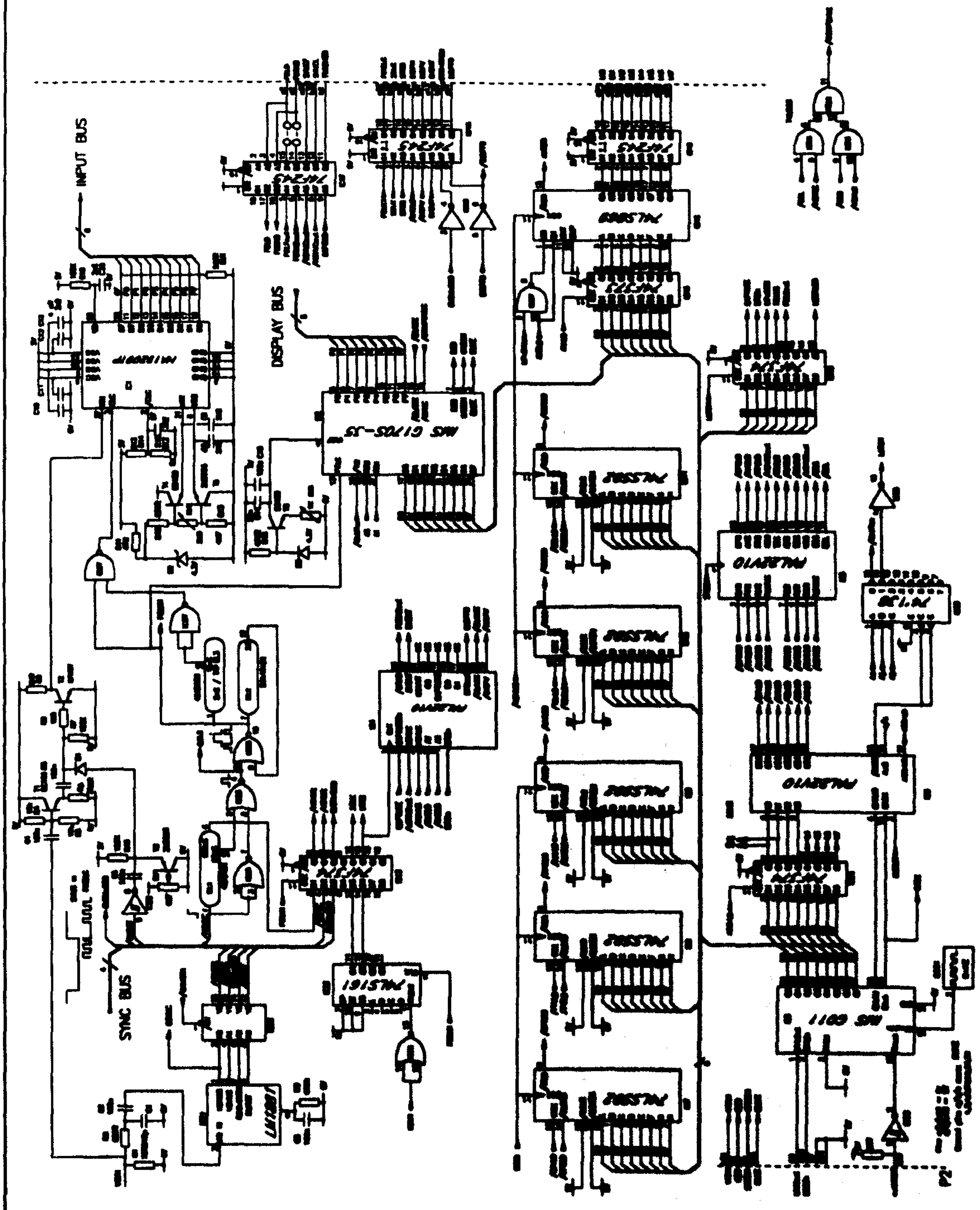
Reproduced from [24]
Figure 4.10 Schematic diagram of Image Processing Board



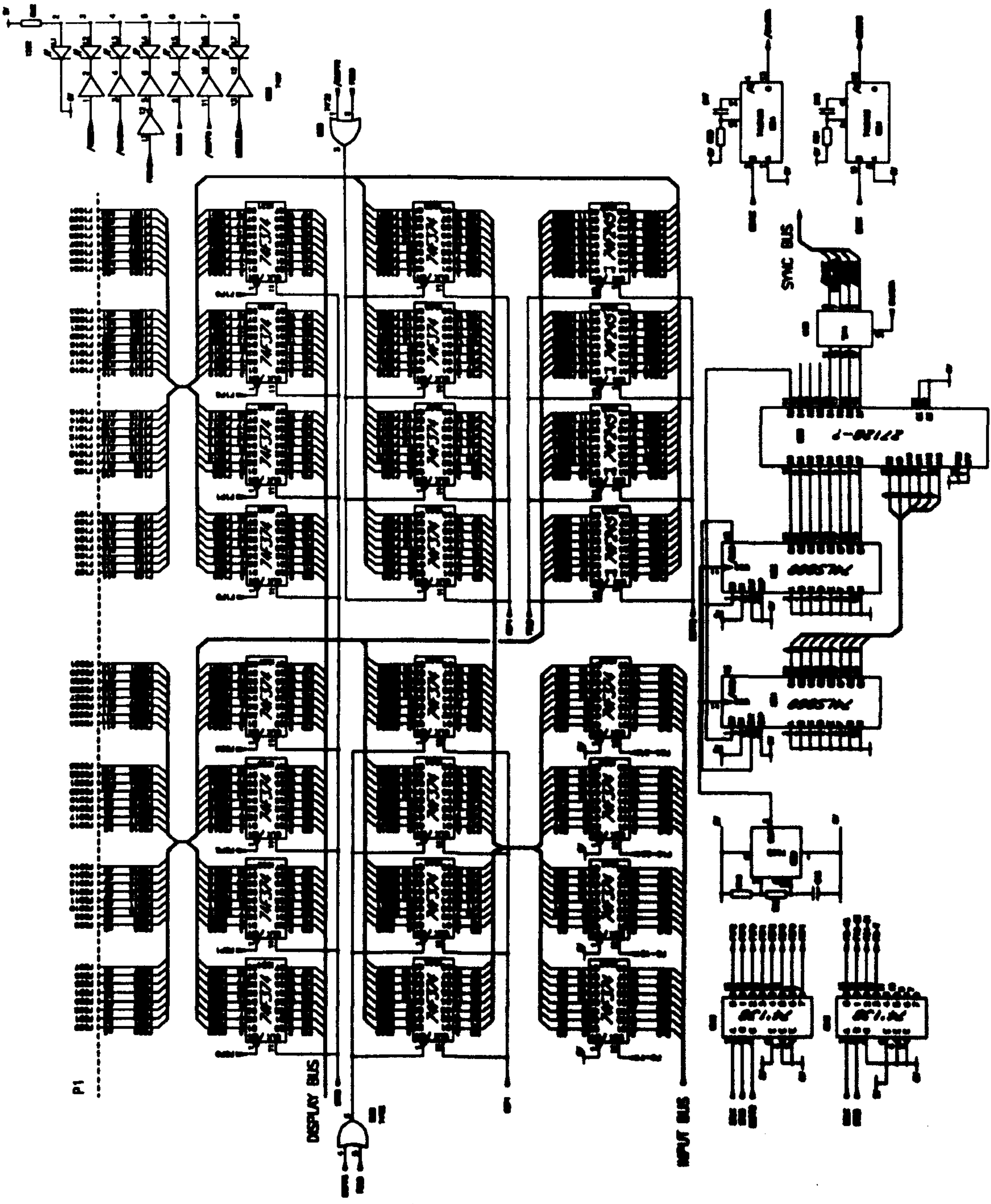
Reproduced from [24]
Figure 4.11 Schematic diagram of Image Processing Board



Reproduced from [24]
Figure 4.12 Schematic diagram of Image Processing Board



Reproduced from [24]
Figure 4.13 Schematic diagram of Frame Capture Board



Reproduced from [24]
Figure 4.14 Schematic diagram of Frame Capture Board

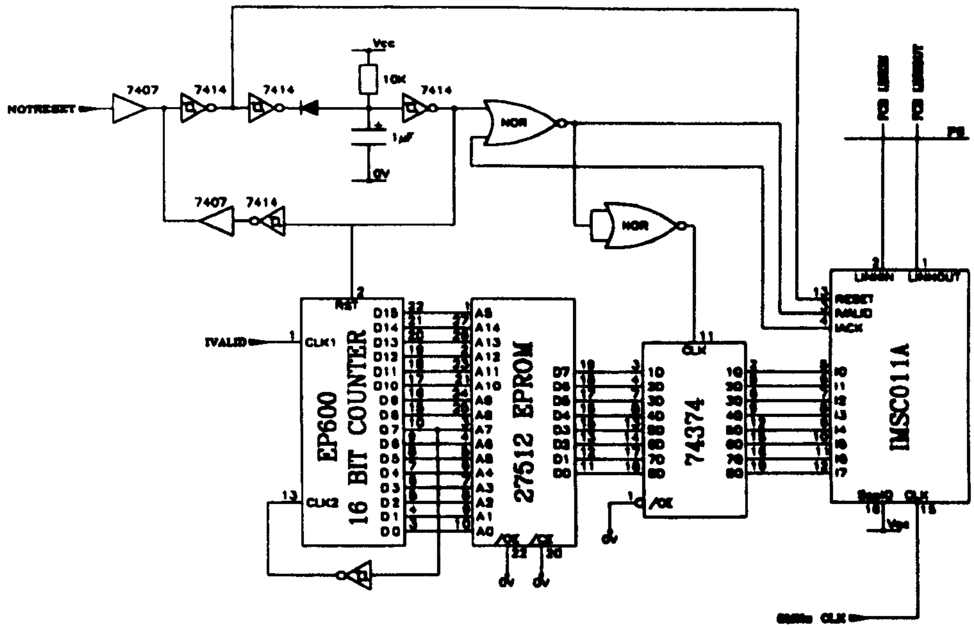


Figure 4.15 Software on EPROM down-loading circuit

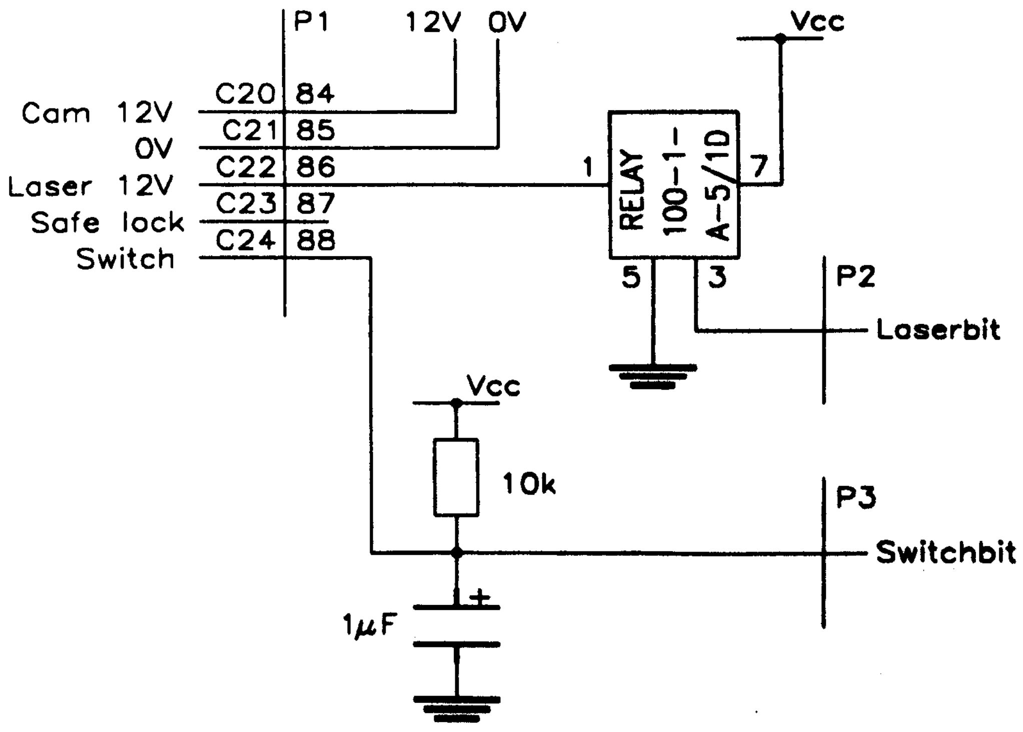


Figure 4.16 Measurement head interface circuit

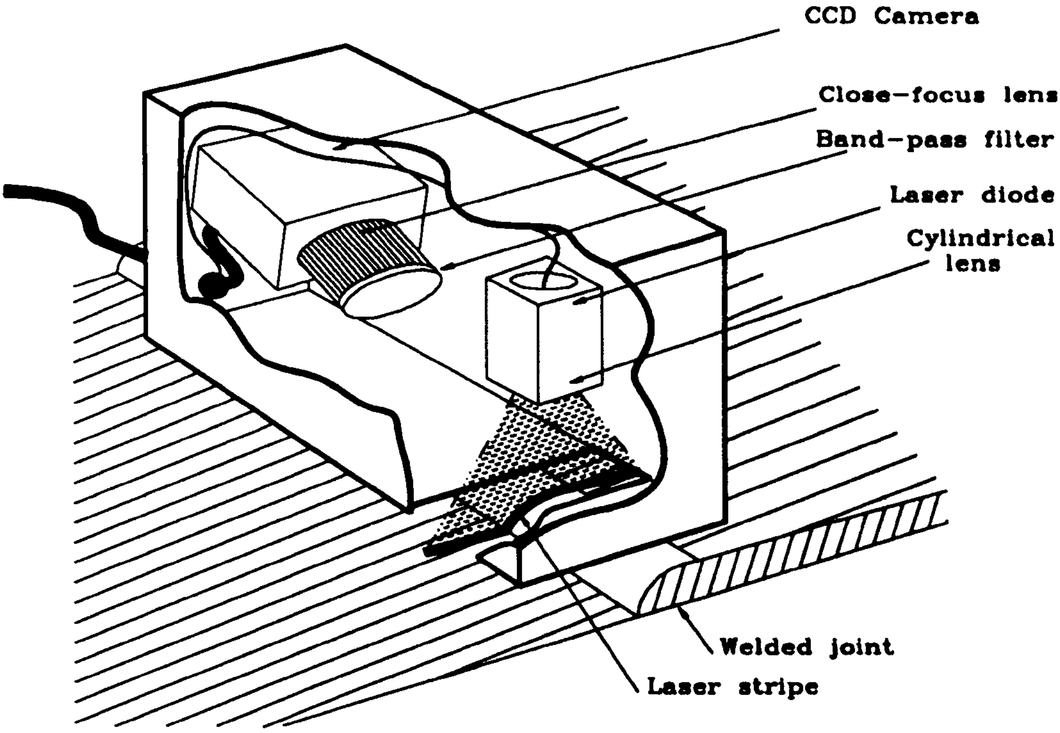


Figure 4.17 Weld measurement head

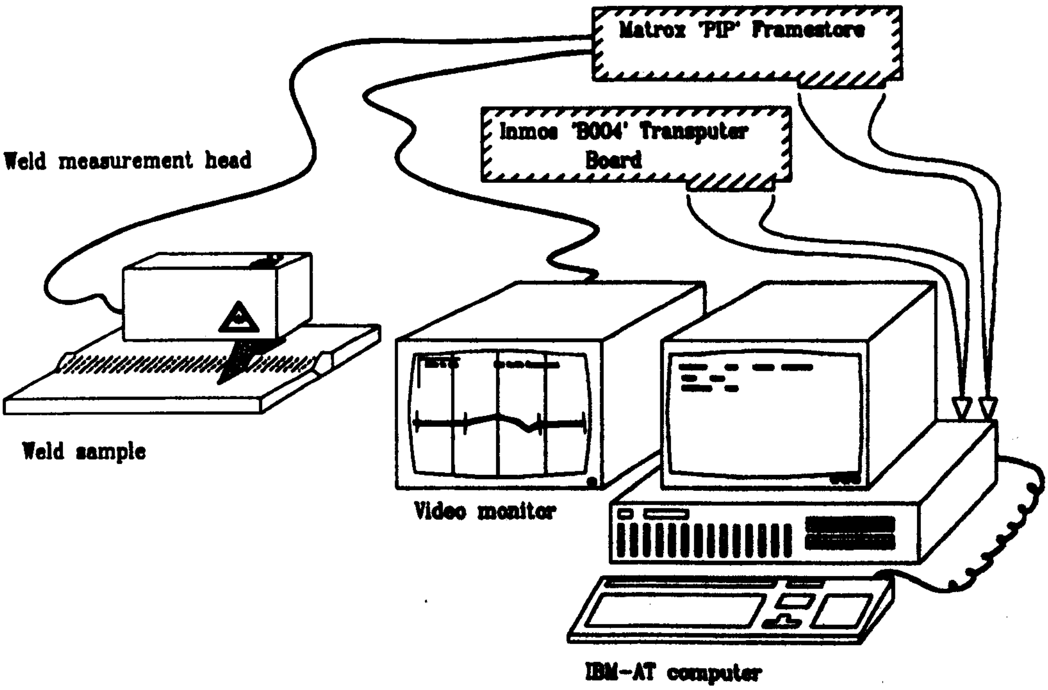


Figure 4.18 Weld measurement development system

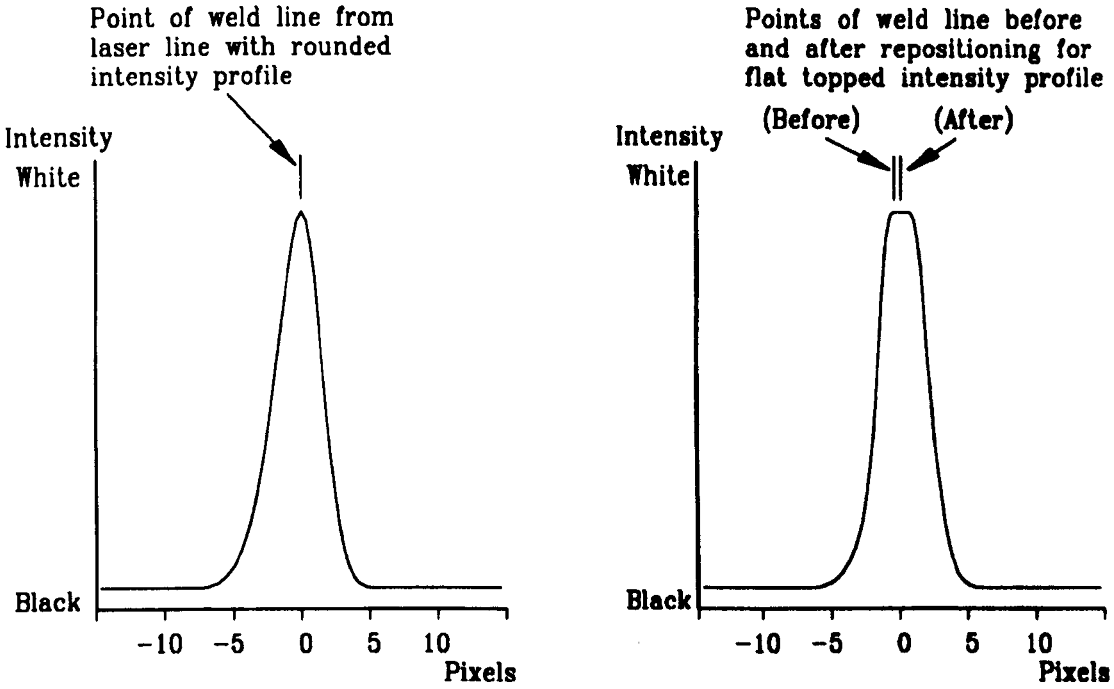
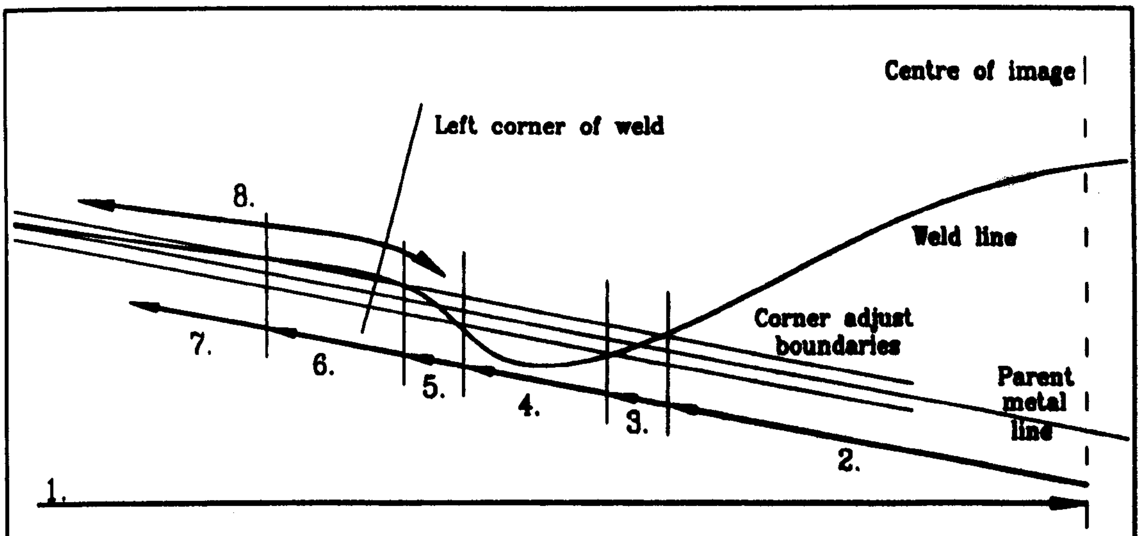


Figure 4.19 Line finding algorithm working on laser lines with rounded, and flat topped, intensity profiles



1. Search from the edge to the centre of the image to find the maximum deviation between the weld line and parent metal line
2. Search from point of maximum deviation to point where weld line falls within corner adjust factor from parent metal line
3. Check that the weld line remains within corner adjust limit for pre-set length
4. Continue searching as in 2.
5. Check as 3.
6. Continue searching as 2.
7. Check as 3.
8. Once pre-set, flat metal length is found search for maximum toe angle

Figure 4.20 Operation of corner finding algorithm on weld profile with undercut

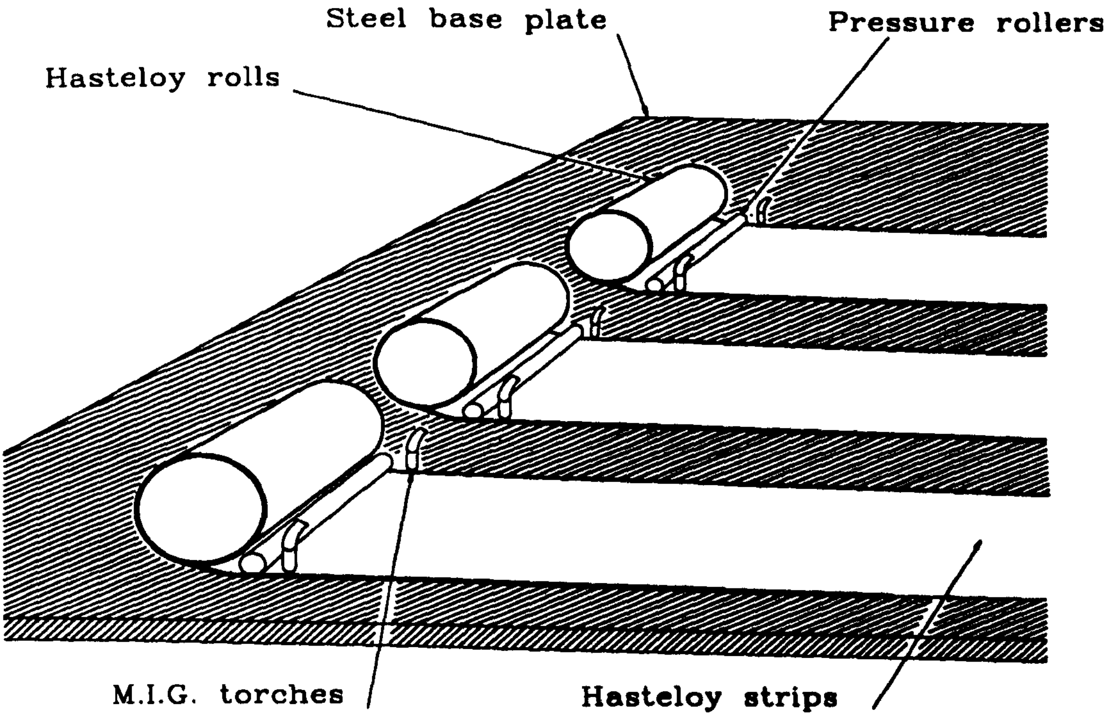


Figure 4.21 First stage of laying hasteloy cladding on steel plate

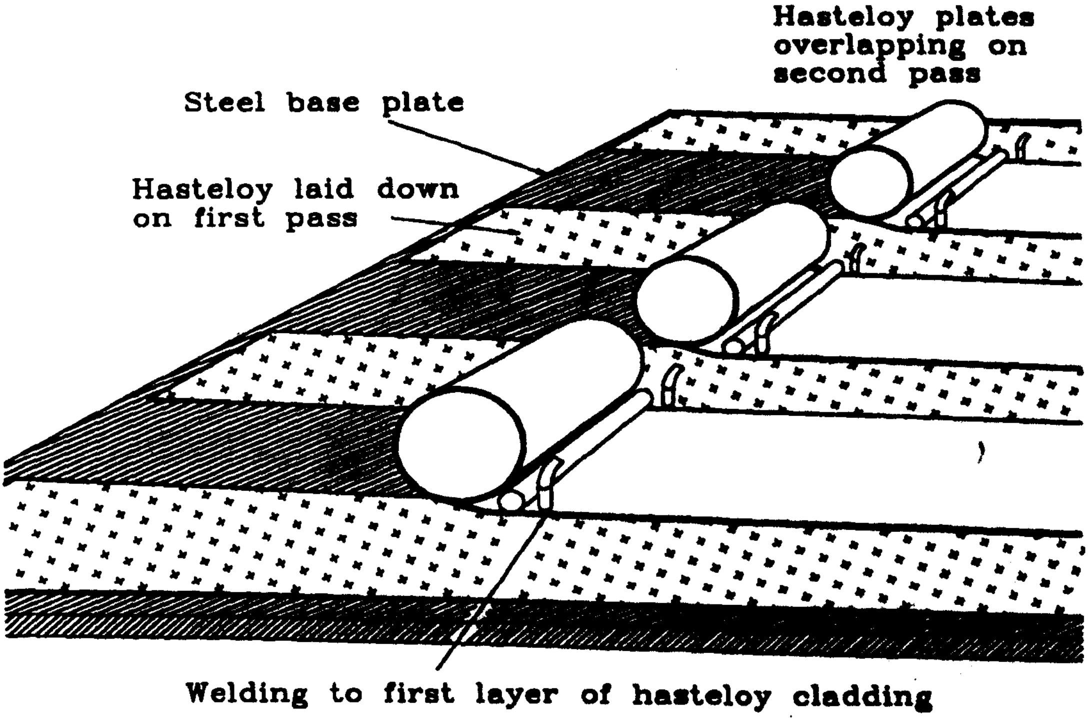

Hasteloy plates overlapping on second pass
Steel base plate
Hasteloy laid down on first pass
Welding to first layer of hasteloy cladding

Figure 4.22 Welding of second layer of cladding

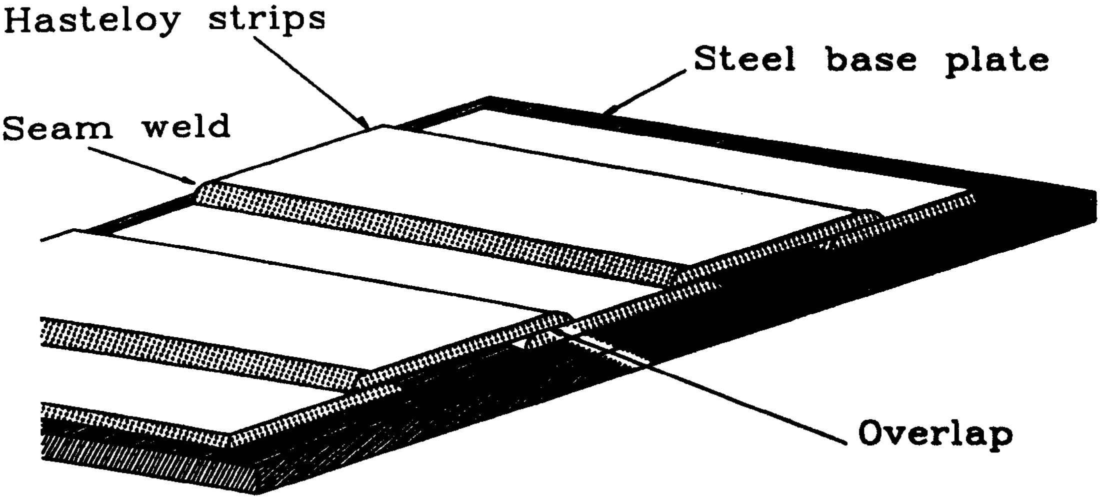


Figure 4.23 Steel plate with both layers of cladding

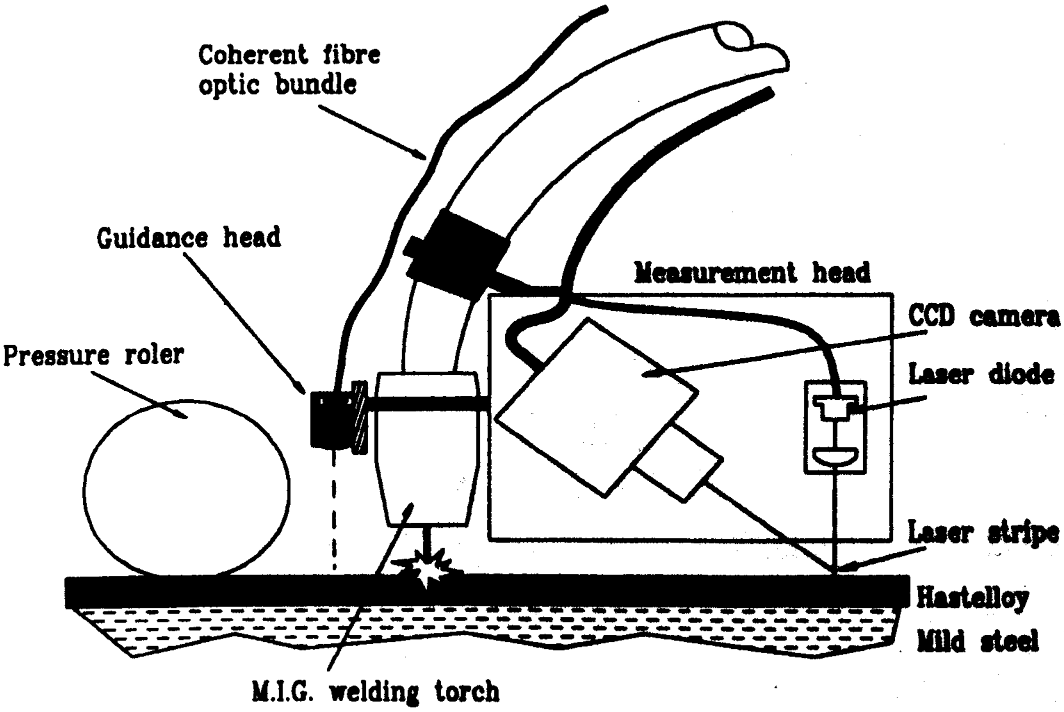


Figure 4.24 Position of tracking and measurement heads on torch

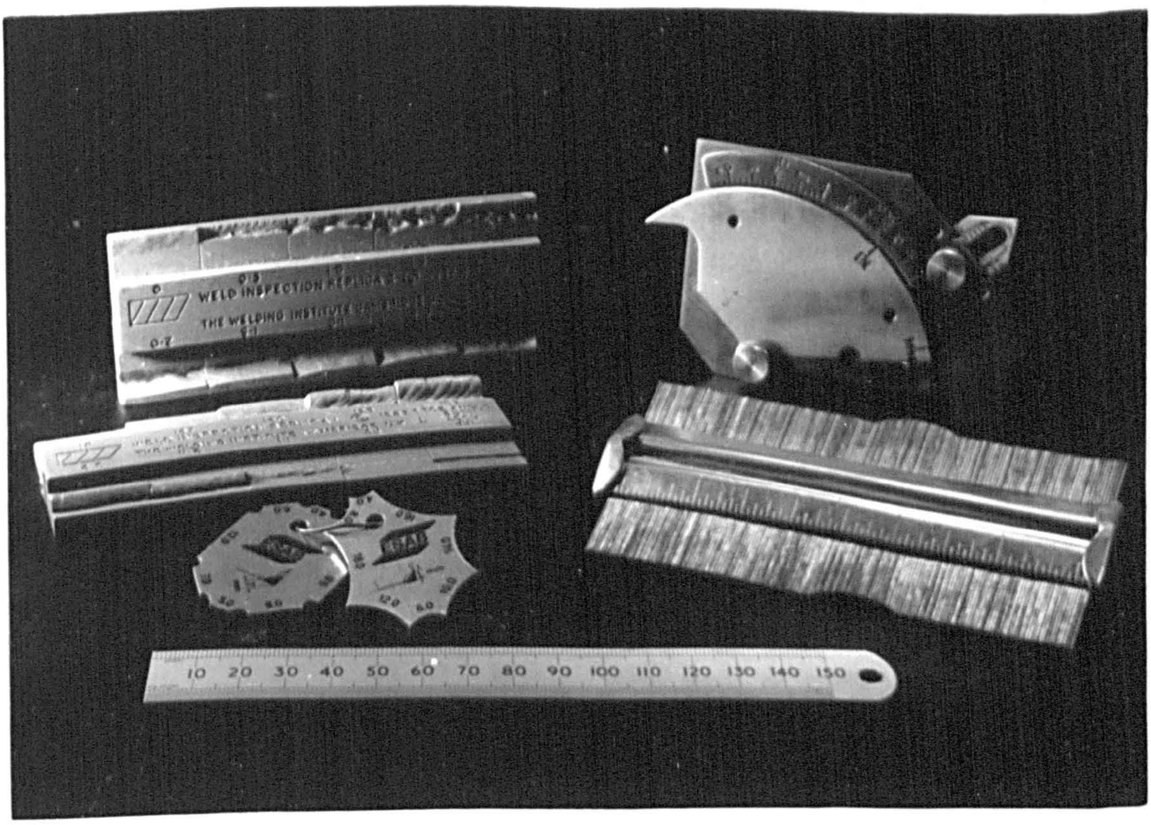


Plate 4.1 Weld gauges: Weld replicas (top left), Welding Institute gauge (top right), Fillet weld gauge (middle left), Six inch rule (bottom), Contour gauge (middle right)

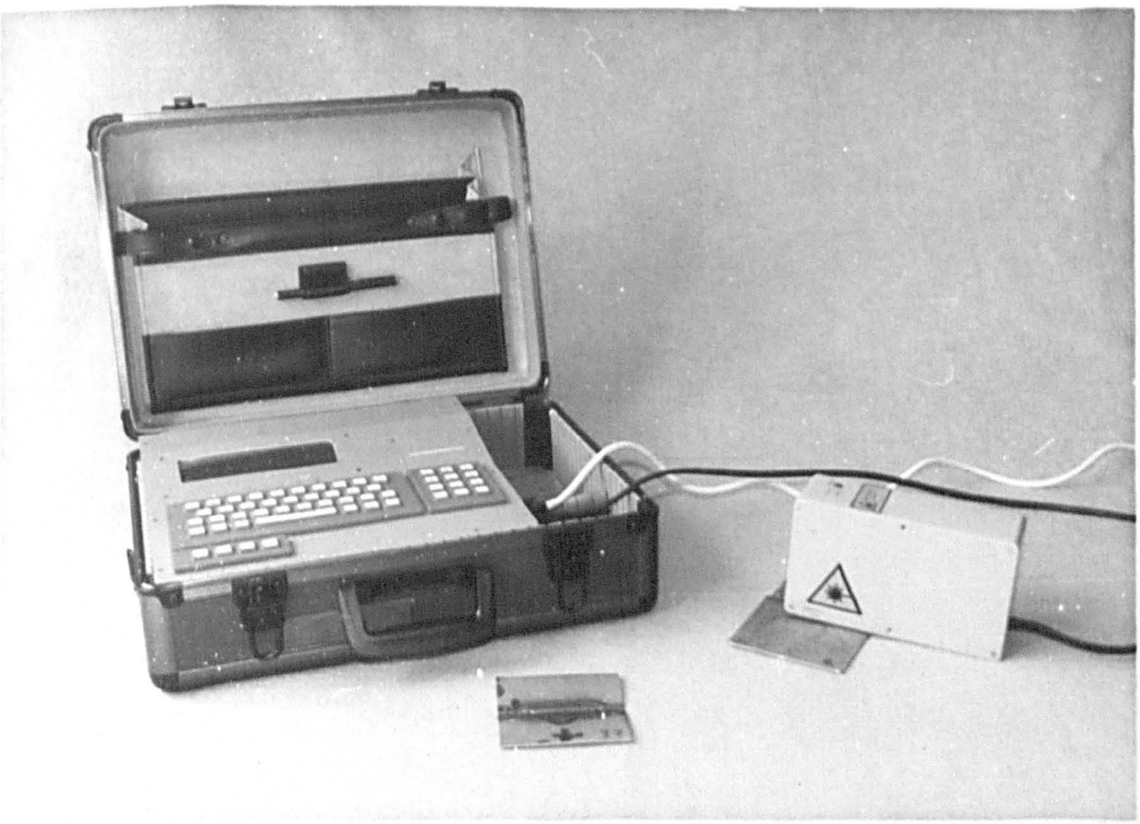


Plate 4.2 The portable, electronic weld gauge

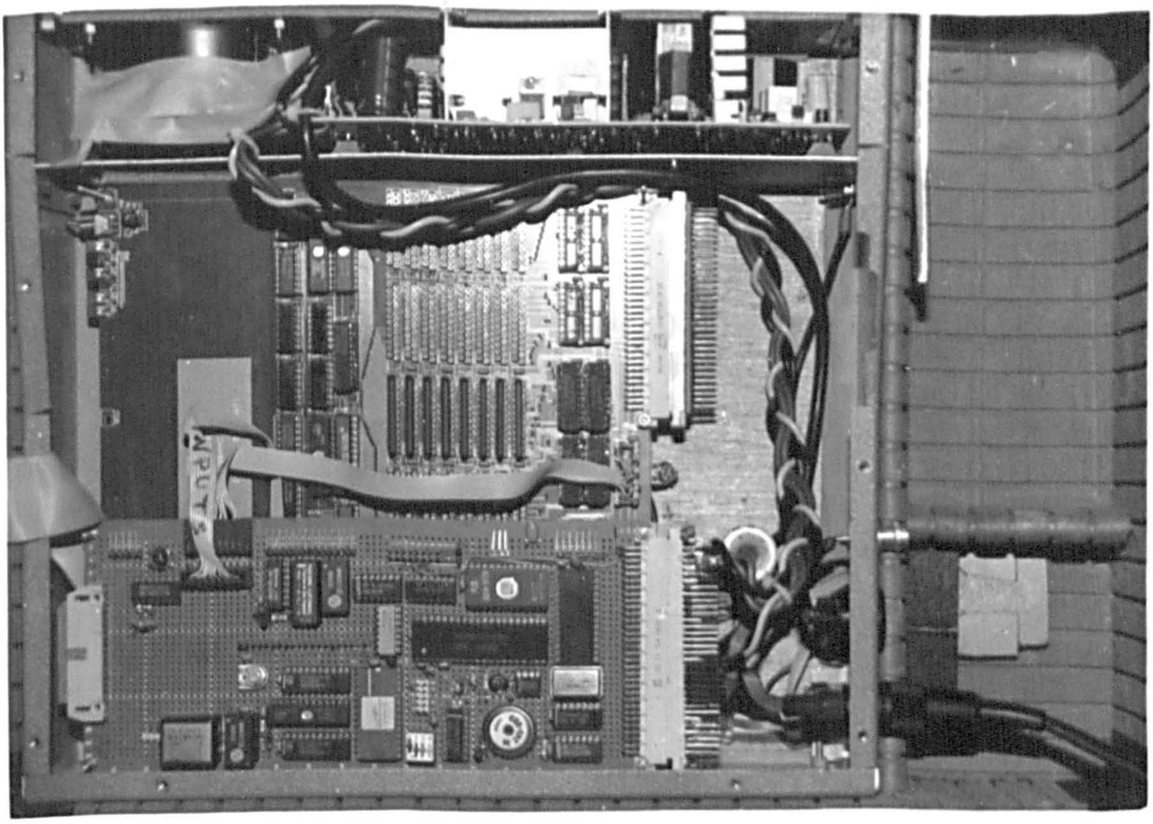


Plate 4.3 Portable electronic weld gauge circuit boards viewed from above

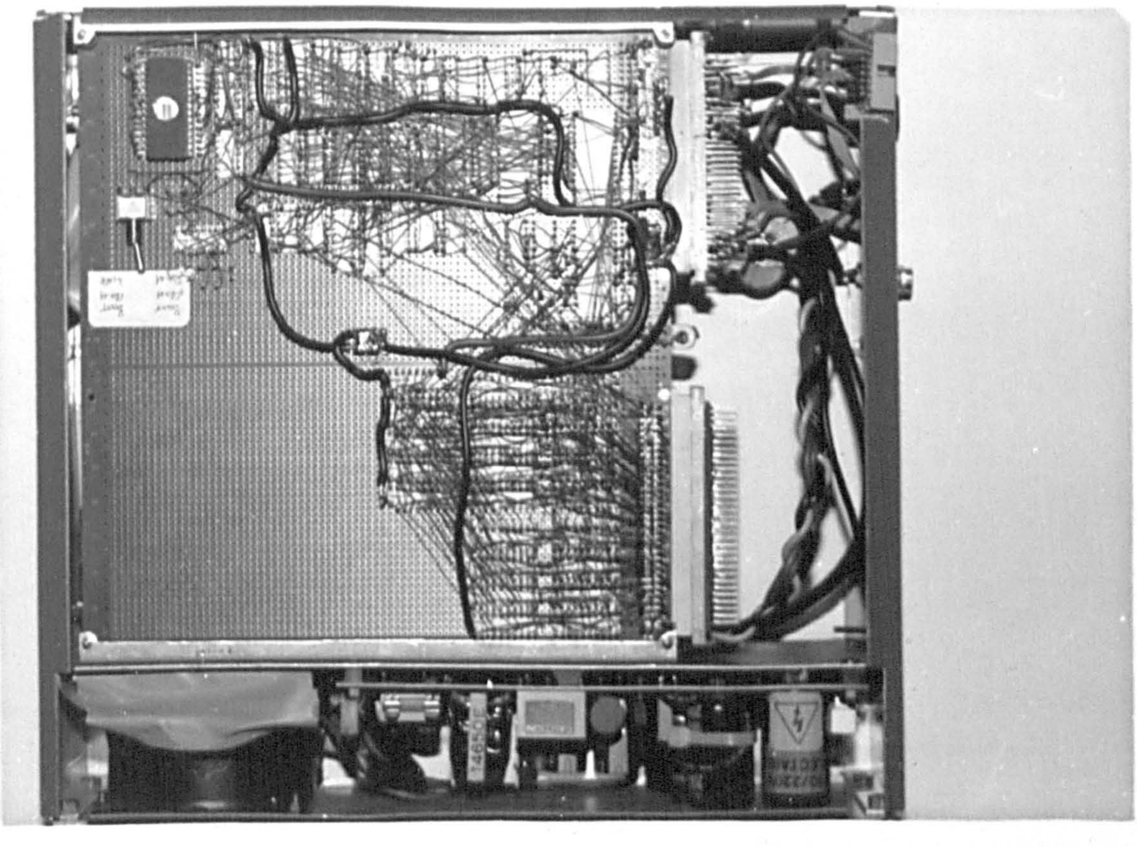
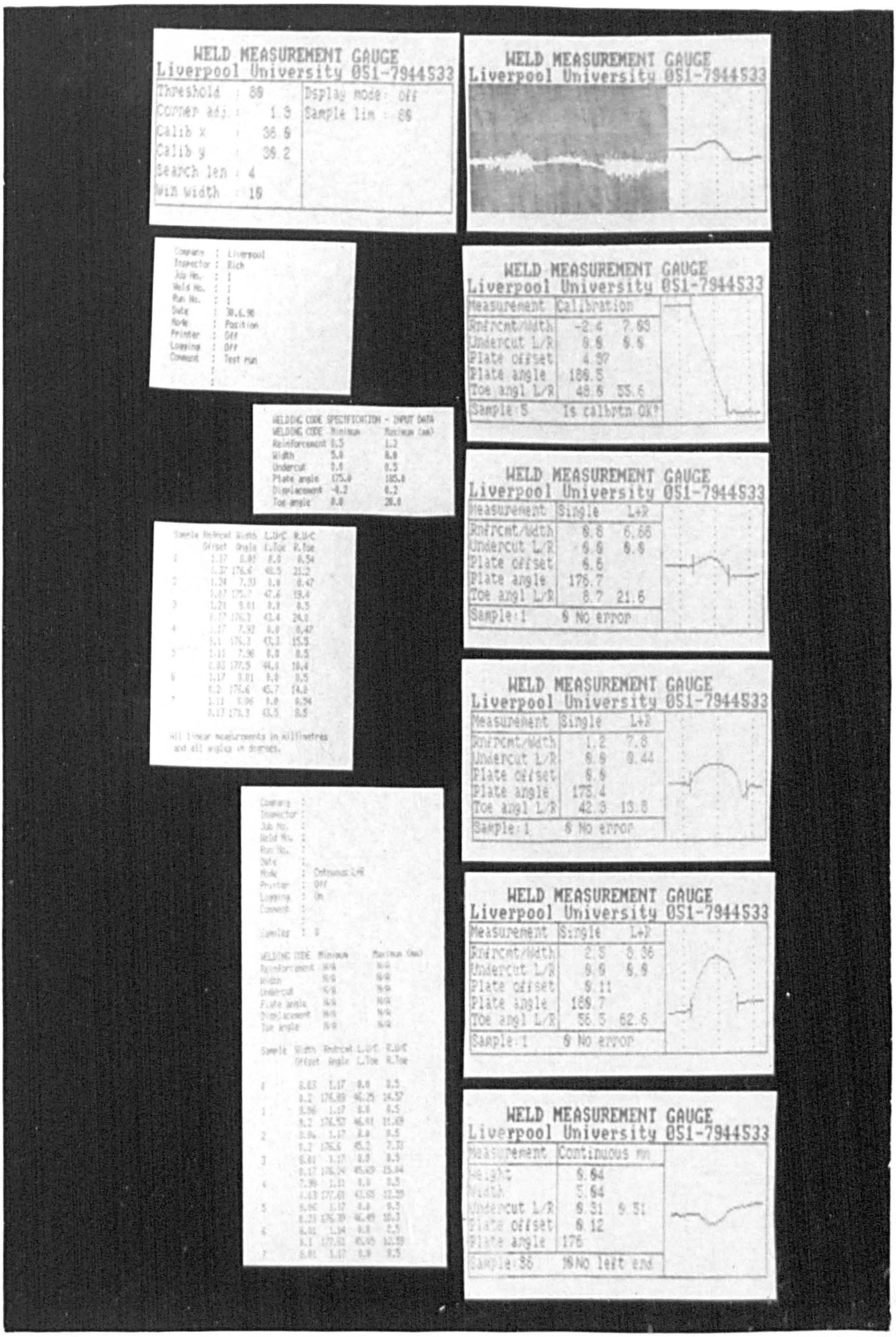


Plate 4.4 The portable, electronic weld gauge circuit boards viewed from below



```
WELD MEASUREMENT GAUGE
Liverpool University 051-7944533
Threshold    : 80         Dsplay mode : Off
Corner adj.  :   1.5      Sample lim  : 80
Calib x      :  36.0
Calib y      :  39.2
Search len   : 4
Win width    : 10
```

```
Company   : Liverpool
Inspector : Rich
Job No.   : 1
Weld No.  : 1
Run No.   : 1
Date      : 30.6.90
Mode      : Position
Printer   : Off
Logging   : Off
Comment   : Test run
```

```
WELDING CODE SPECIFICATION - INPUT DATA
WELDING CODE  Minimum    Maximum (mm)
Reinforcement 0.5        1.2
Width         5.0        8.0
Undercut      0.0        0.5
Plate angle   175.0      185.0
Displacement  -0.2       0.2
Toe angle     0.0        20.0
```

```
Sample Rnfrcmt Width  L.U/C R.U/C
       Offset  Angle  L.Toe R.Toe
1       1.17   8.05   0.0   0.54
        0.37 176.6   48.5  21.2
2       1.24   7.33   0.0   0.47
        0.07 175.7   47.6  19.4
3       1.21   9.01   0.0   0.5
        0.17 176.3   43.4  24.0
4       1.17   7.92   0.0   0.47
        0.1  175.3   43.3  15.5
5       1.11   7.90   0.0   0.5
        0.03 177.5   44.0  10.4
6       1.17   8.01   0.0   0.5
        0.2  175.6   45.7  14.0
7       1.11   5.06   0.0   0.54
        0.17 179.3   43.5   0.5

All linear measurements in millimetres
and all angles in degrees.
```

```
Company   :
Inspector :
Job No.   :
Weld No.  :
Run No.   :
Date      :
Mode      : Cntnuous L+R
Printer   : Off
Logging   : On
Comment   :

Samples   : 8

WELDING CODE  Minimum    Maximum (mm)
Reinforcement N/A        N/A
Width         N/A        N/A
Undercut      N/A        N/A
Plate angle   N/A        N/A
Displacement  N/A        N/A
Toe angle     N/A        N/A

Sample  Width  Rnfrcmt L.U/C R.U/C
        Offset Angle   L.Toe R.Toe

0       8.03   1.17    0.0    0.5
        0.2  175.89   46.25  14.57
1       8.06   1.17    0.0    0.5
        0.2  176.57   46.41  11.69
2       8.04   1.17    0.0    0.5
        0.2  176.6    45.2    7.33
3       8.01   1.17    0.0    0.5
        0.17 176.34   45.69  15.04
4       7.99   1.11    0.0    0.5
        0.03 177.61   43.05  12.39
5       8.06   1.17    0.0    0.5
        0.21 176.39   46.49  10.3
6       8.01   1.14    0.0    0.5
        0.1  177.01   45.05  12.59
7       8.01   1.17    0.0    0.5
```

```
WELD MEASUREMENT GAUGE
Liverpool University 051-7944533
```

```
WELD MEASUREMENT GAUGE
Liverpool University 051-7944533
Measurement   | Calibration
Rnfrcmt/Wdth  |  -2.4   7.69
Undercut L/R  |   0.0   0.0
Plate offset  |   4.97
Plate angle   | 180.5
Toe angl L/R  |  48.0  55.6
Sample:5        Is calbrtn OK?
```

```
WELD MEASUREMENT GAUGE
Liverpool University 051-7944533
Measurement   | Single   L+R
Rnfrcmt/Wdth  |   0.6   6.66
Undercut L/R  |   0.0   0.0
Plate offset  |   0.0
Plate angle   | 176.7
Toe angl L/R  |   8.7  21.6
Sample:1        0 No error
```

```
WELD MEASUREMENT GAUGE
Liverpool University 051-7944533
Measurement   | Single   L+R
Rnfrcmt/Wdth  |   1.2   7.8
Undercut L/R  |   0.0   0.44
Plate offset  |   0.0
Plate angle   | 175.4
Toe angl L/R  |  42.3  13.8
Sample:1        0 No error
```

```
WELD MEASUREMENT GAUGE
Liverpool University 051-7944533
Measurement   | Single   L+R
Rnfrcmt/Wdth  |   2.5   8.36
Undercut L/R  |   0.0   0.0
Plate offset  |   0.11
Plate angle   | 169.7
Toe angl L/R  |  56.5  62.6
Sample:1        0 No error
```

```
WELD MEASUREMENT GAUGE
Liverpool University 051-7944533
Measurement   | Continuous mm
Height        |   0.64
Width         |   5.64
Undercut L/R  |   0.31  0.51
Plate offset  |   0.12
Plate angle   | 176
Sample:36       10 No left end
```

Plate 4.5 Example prints of measurements from the gauge

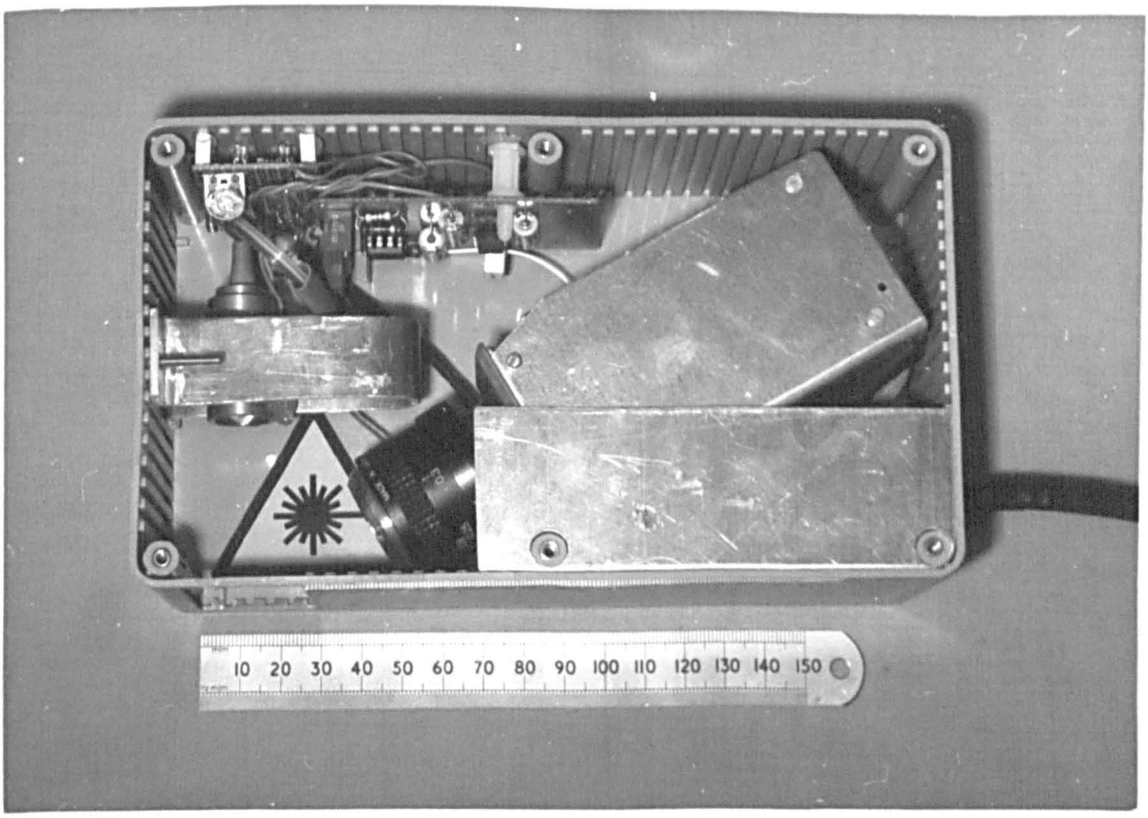


Plate 4.6 The measurement head

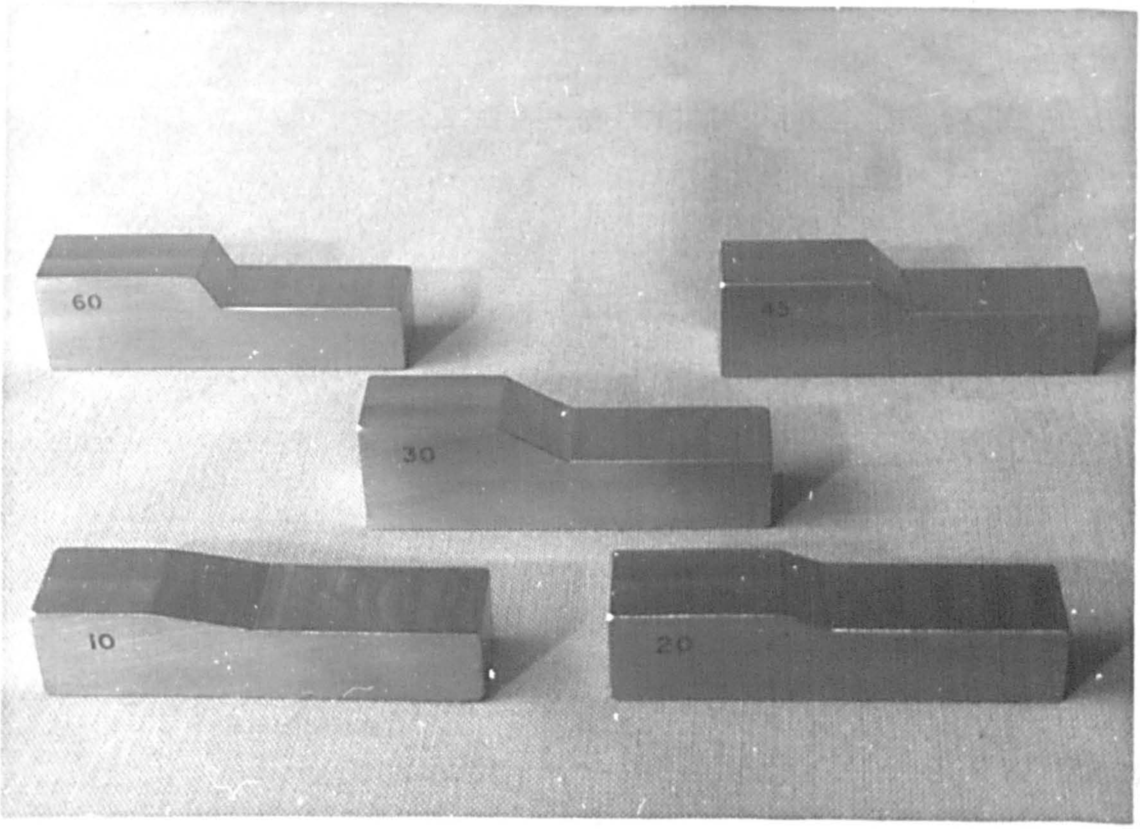
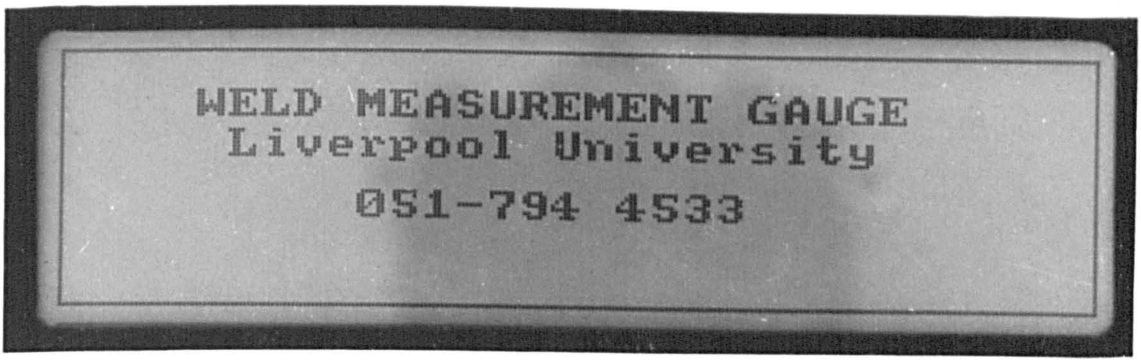


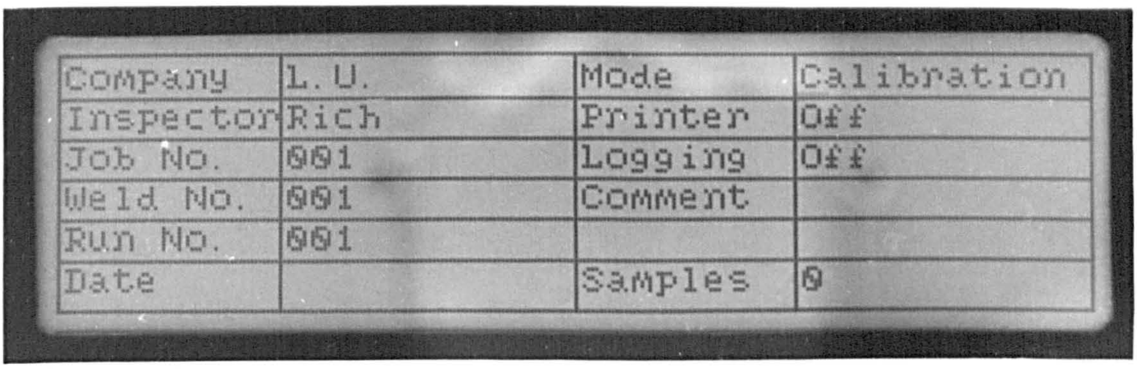
Plate 4.7 Calibration blocks



WELD MEASUREMENT GAUGE
Liverpool University

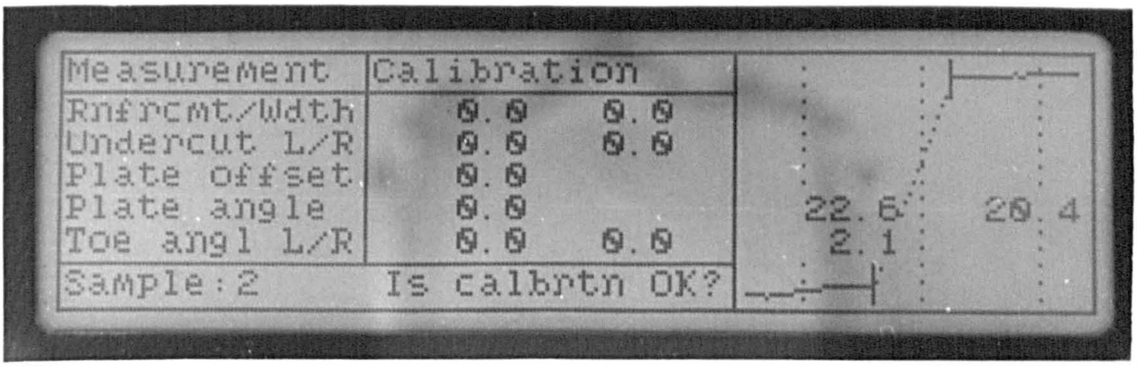
051-794 4533

Plate 4.8a LCD screen - Title



Company	L. U.	Mode	Calibration
Inspector	Rich	Printer	Off
Job No.	001	Logging	Off
Weld No.	001	Comment	
Run No.	001		
Date		Samples	0

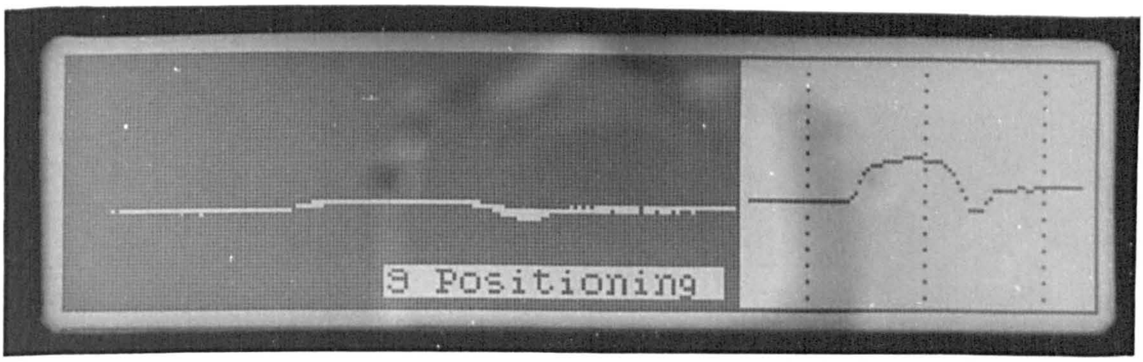
Plate 4.8b LCD screen - Main display



Measurement	Calibration	
Rnfrcmt/Wdth	0.0	0.0
Undercut L/R	0.0	0.0
Plate offset	0.0	
Plate angle	0.0	
Toe angl L/R	0.0	0.0

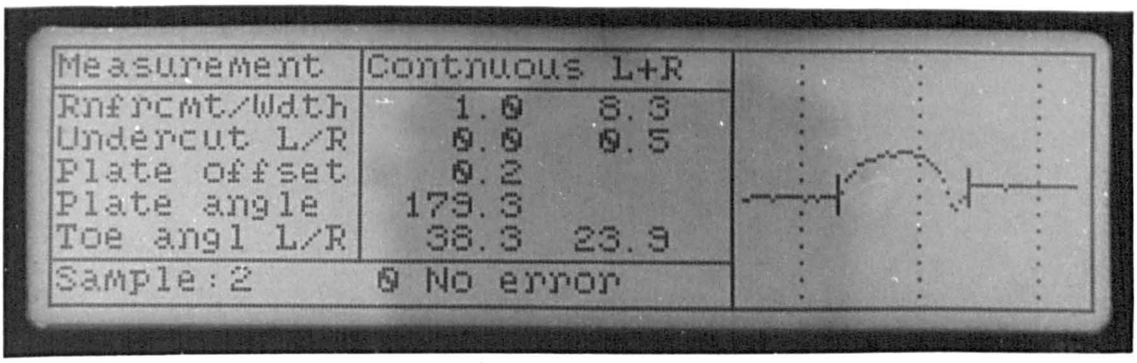
Sample:2 Is calbrtn OK?

Plate 4.8c LCD screen - Measurement display during calibration



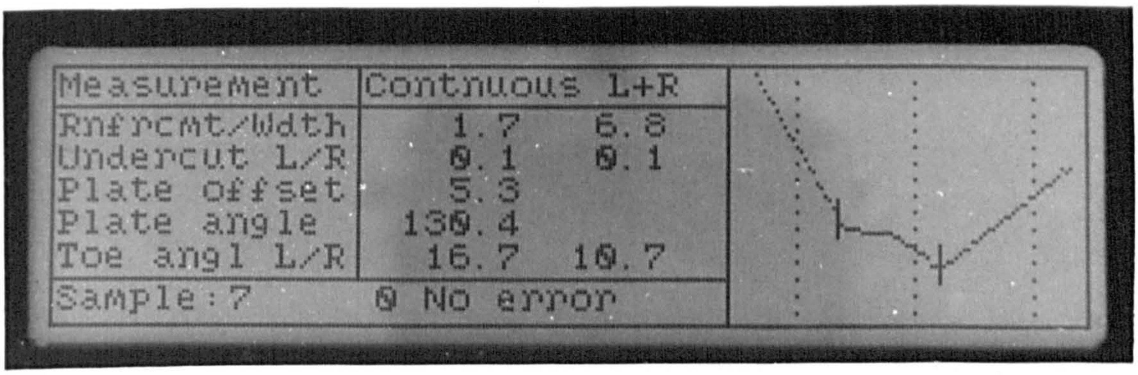
```
9 Positioning
```

Plate 4.8d LCD screen - Positioning display



```
Measurement      Contnuous L+R
Rnfrcmt/Wdth        1.0    8.3
Undercut L/R        0.0    0.5
Plate offset        0.2
Plate angle       179.3
Toe angl L/R       38.3   23.9
Sample:2          0 No error
```

Plate 4.8e LCD screen - Measurement display with butt weld with undercut



```
Measurement      Contnuous L+R
Rnfrcmt/Wdth        1.7    6.8
Undercut L/R        0.1    0.1
Plate offset        5.3
Plate angle       130.4
Toe angl L/R       16.7   10.7
Sample:7          0 No error
```

Plate 4.8f LCD screen - Measurement display with fillet weld

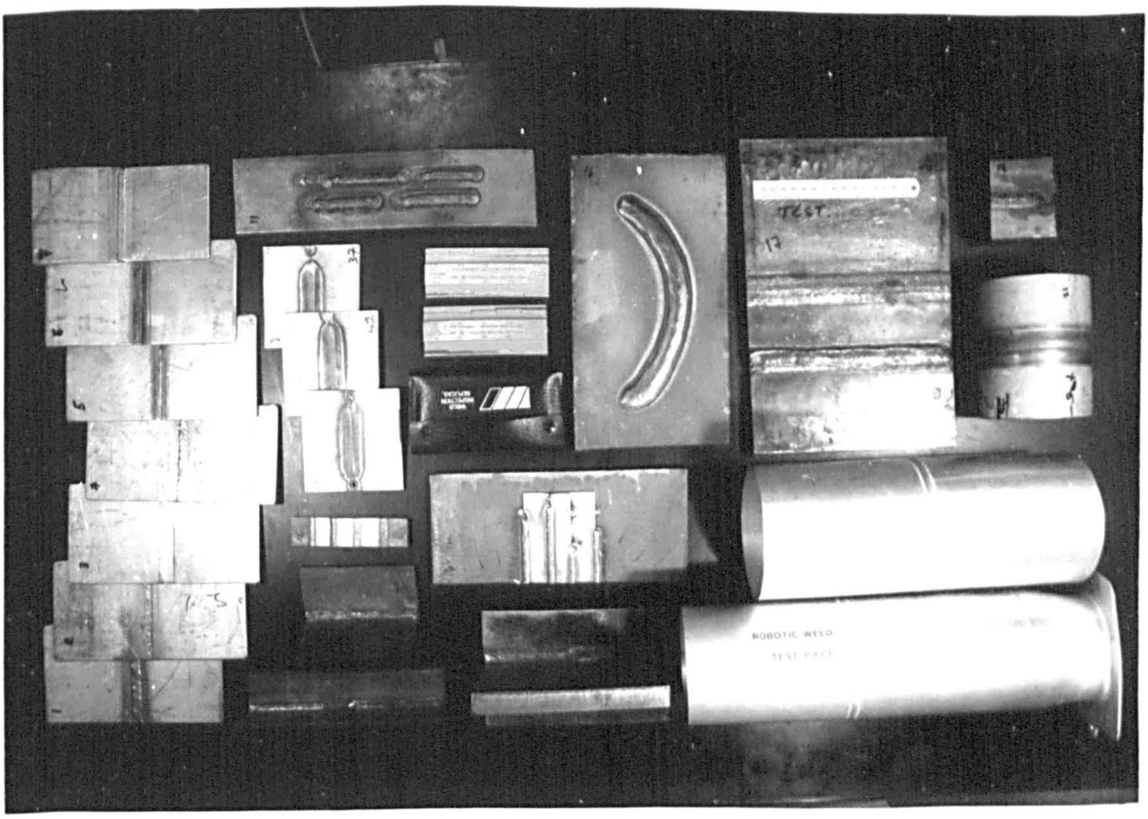


Plate 4.9 A selection of weld samples used to test the gauged

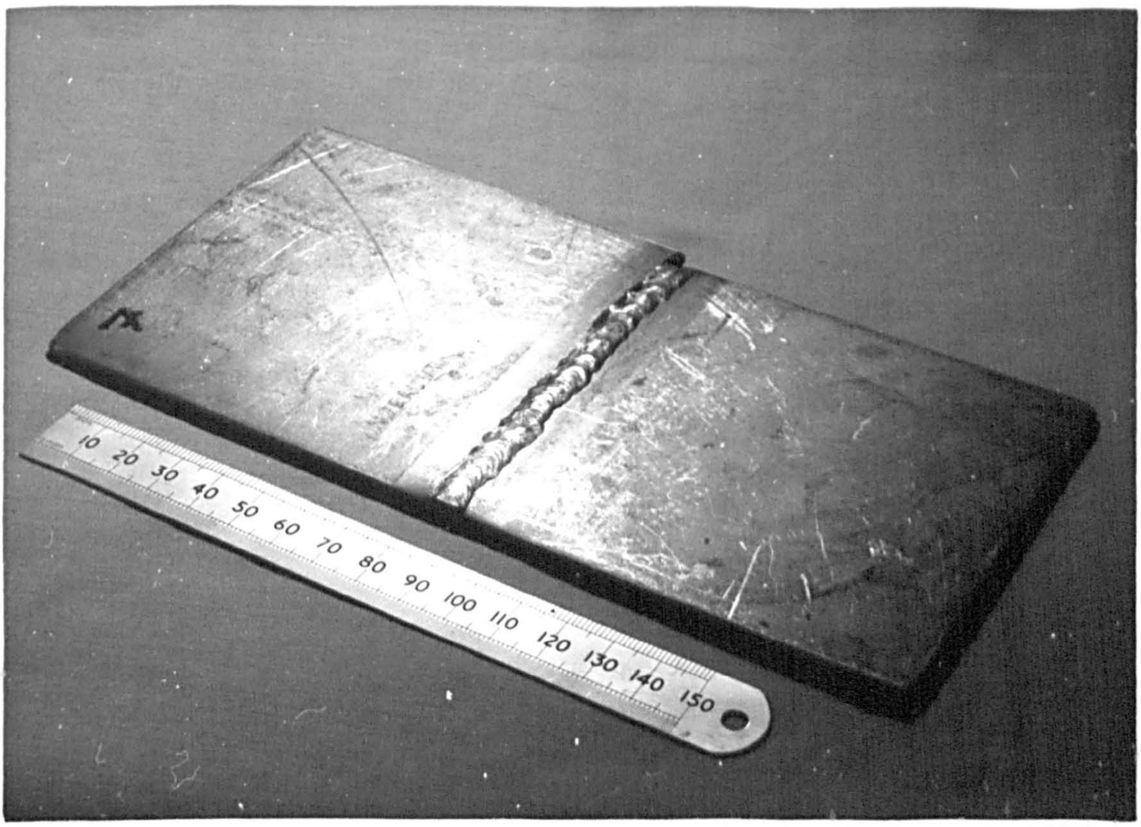


Plate 4.10 Sample butt weld provided by The Welding Institute

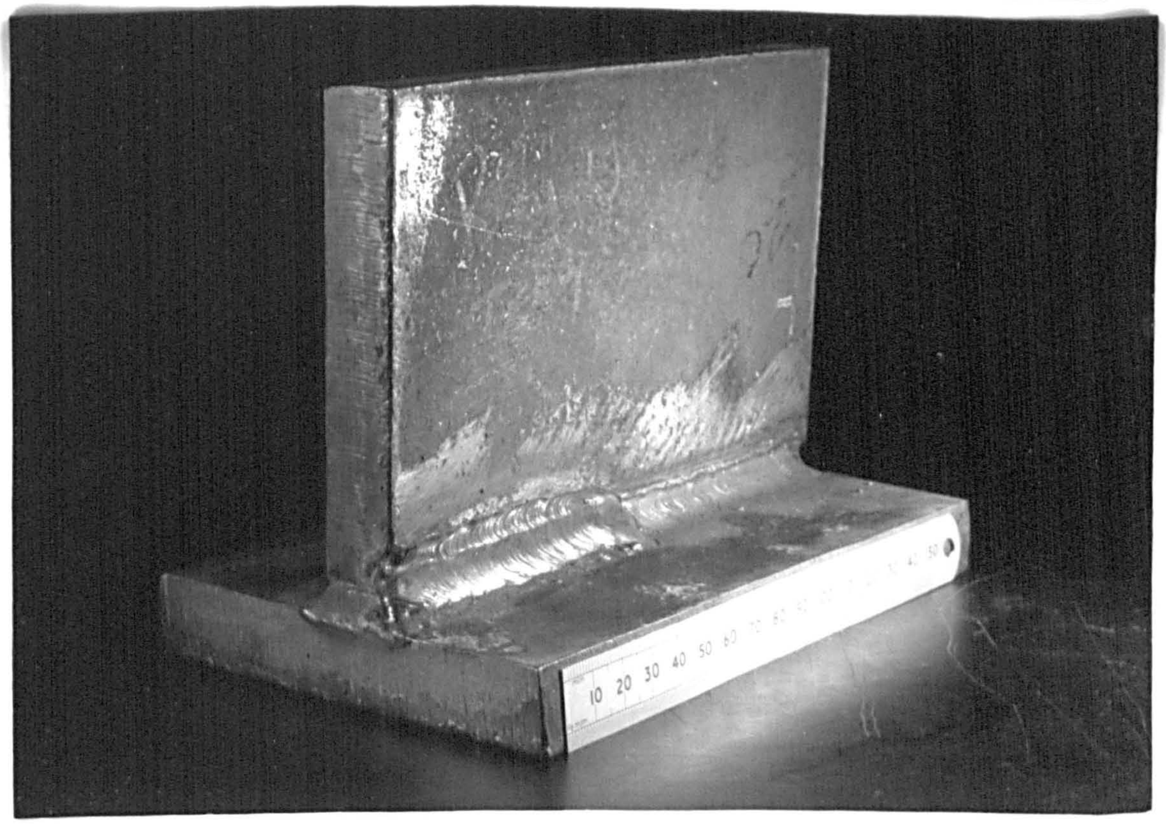


Plate 4.11 Sample multipass fillet weld

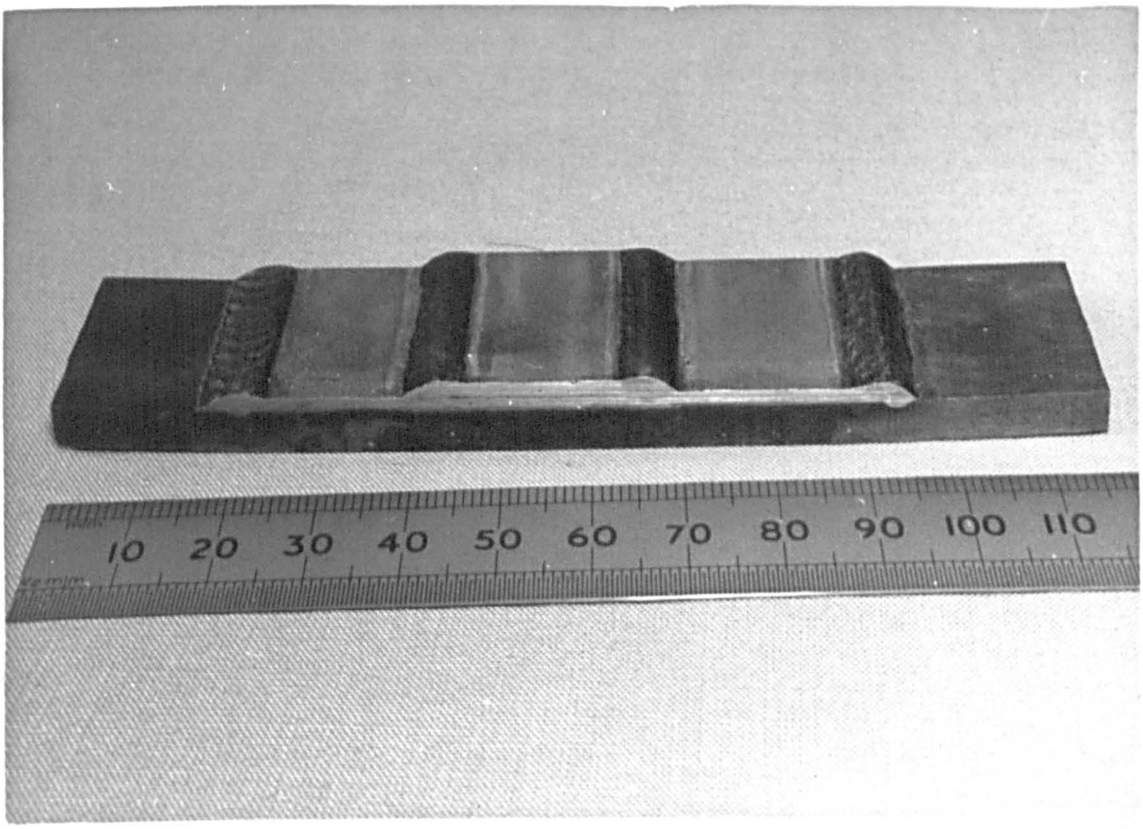


Plate 4.12 Sample lap weld

4.10 Results from weld profile measurements

The results which follow are divided into four sections, 4.10.1 images of weld samples, 4.10.2 shows their corresponding data and graphs of the data are shown in 4.10.3 . Section 4.10.4 then shows images and data and graphs from the on-line trials performed with the measurement software. To achieve the measurement of the wide variety of weld sizes and types shown, the gauge's measurement head was adjusted for each sample. This would be achieved commercially by selling heads of different types for the various weld sizes.

4.10.1 Images from test weld samples

Images 4.1 and 4.2 are included to show the view which the camera sees without a band-pass filter mounted in front of the lens for butt and fillet weld samples. These images show how the laser stripe provides a cross-section view of the weld and permits the weld information to be extracted. Images 4.3 and 4.4 then show the laser stripe across the same butt and fillet welds with the band-pass filter fitted. Image 4.5 shows the unprocessed image of the calibration block. The images 4.6 onward then show the processed scenes and correspond to the data in section 4.10.2 .

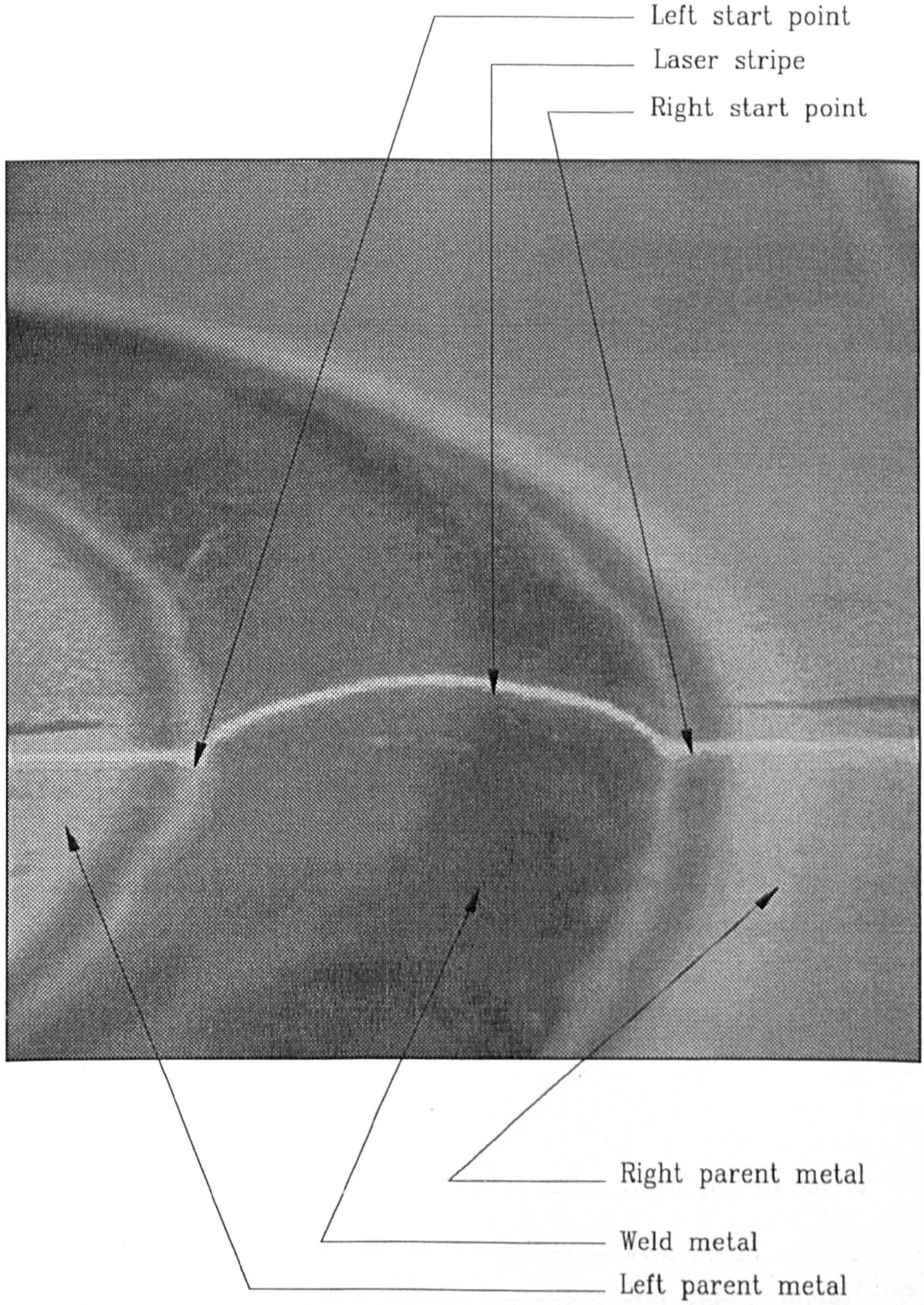


Image 4.1 A butt weld seam with laser stripe viewed without an optical band pass filter fitted to camera lens

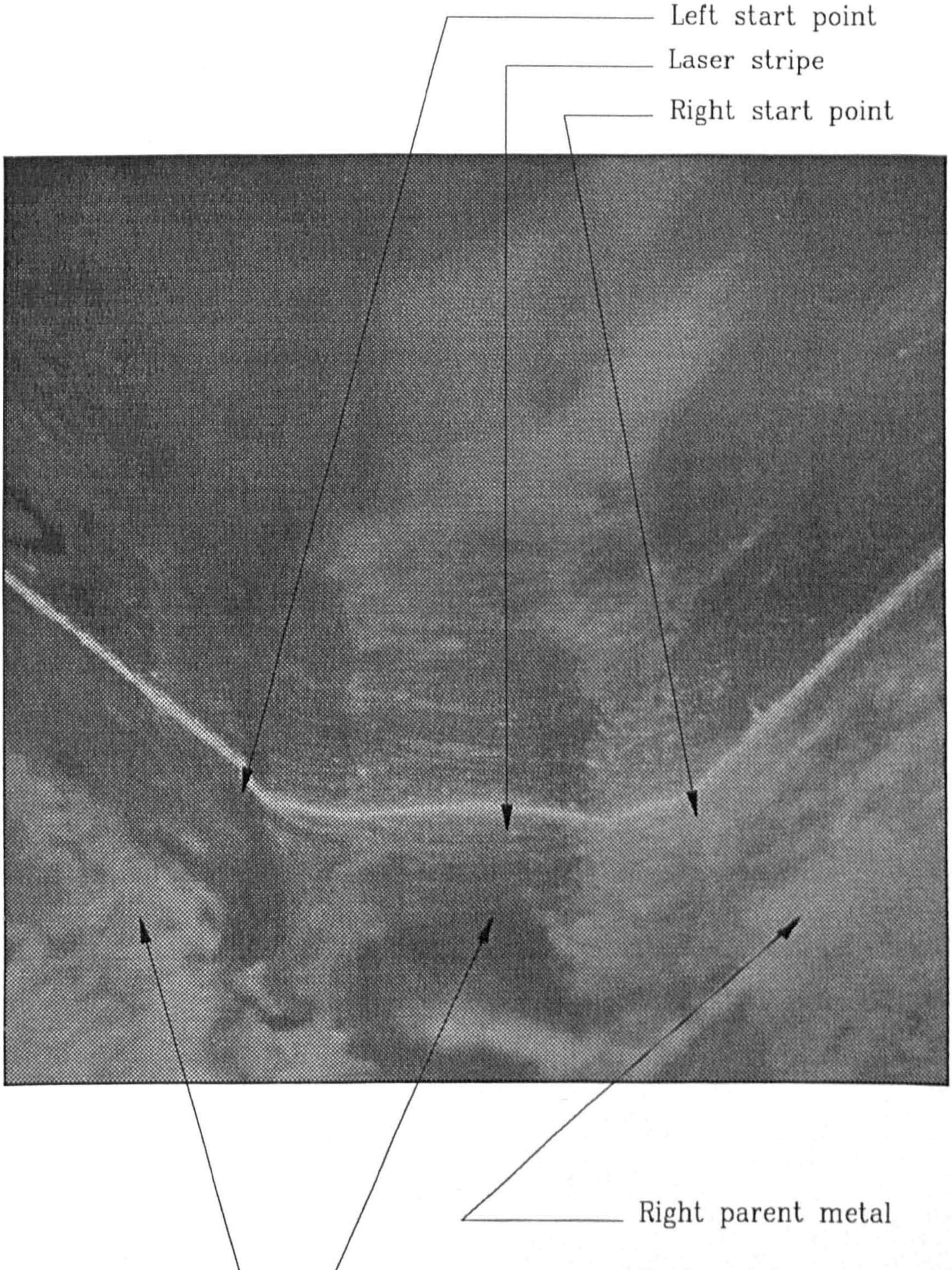


Image 4.2 An fillet weld seam with laser stripe viewed without an optical band pass filter fitted to camera lens

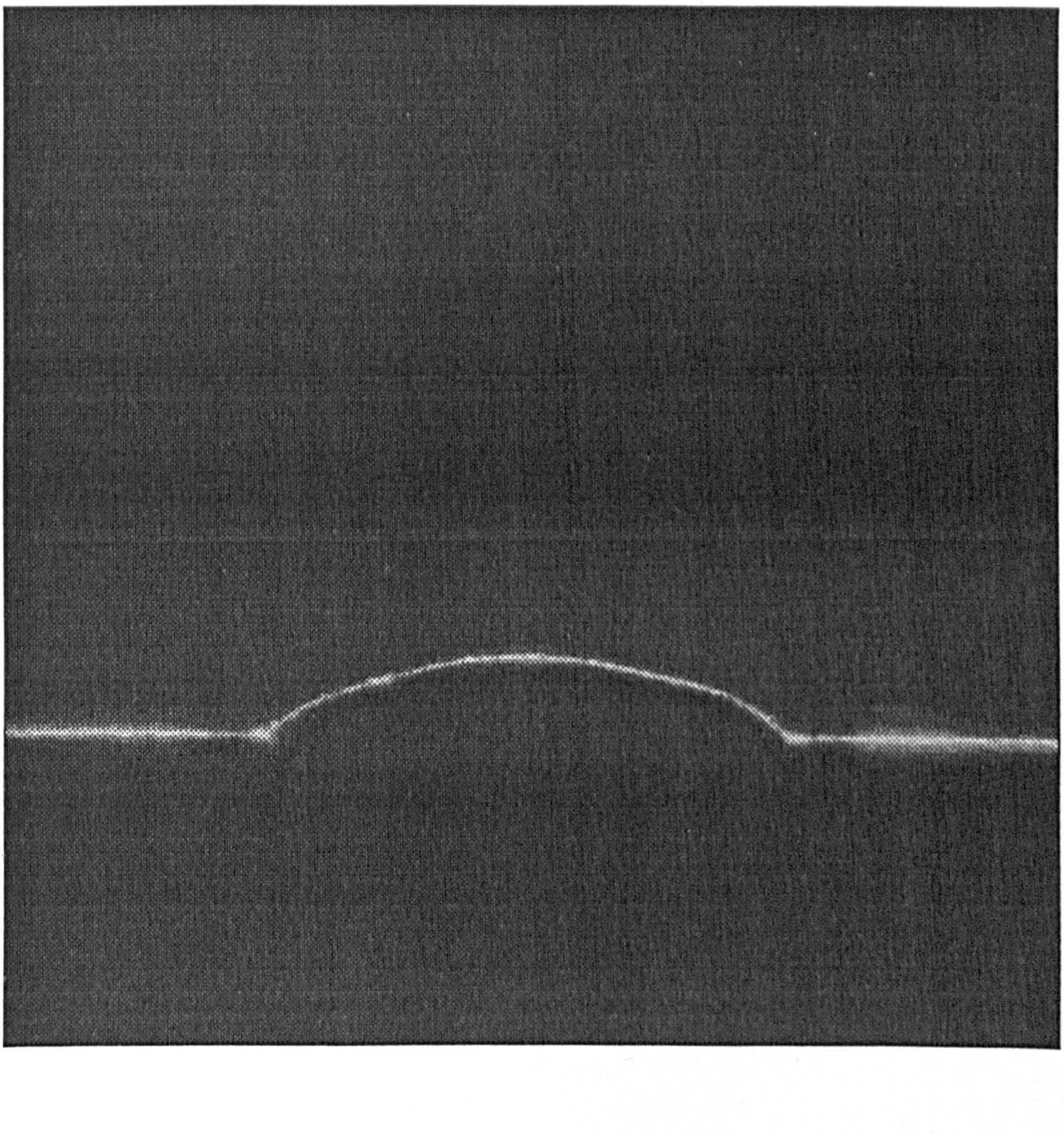


Image 4.3 A butt weld seam with laser stripe viewed with an optical band pass filter fitted to camera lens

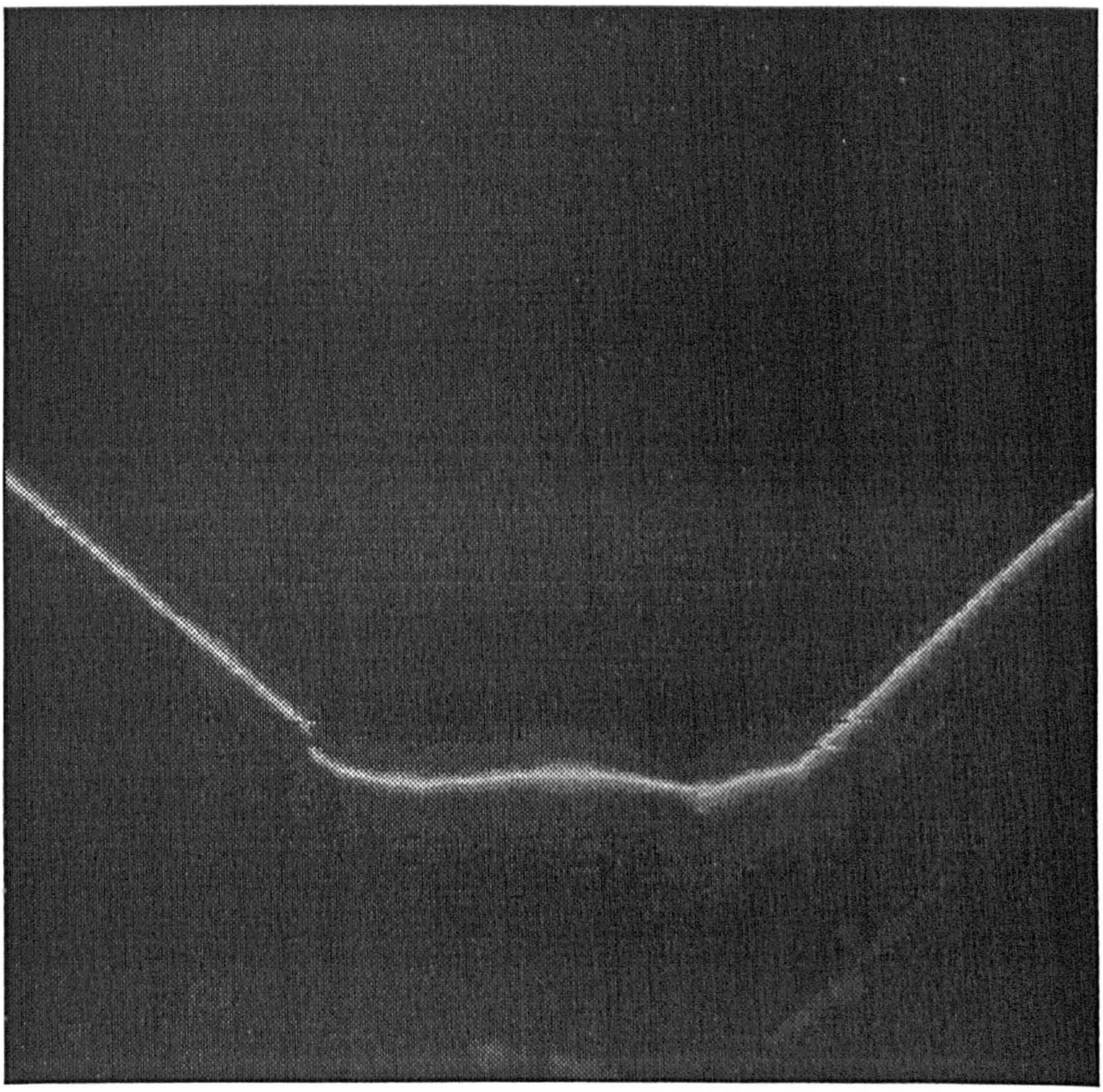


Image 4.4 A fillet weld seam with laser stripe viewed with an optical band pass filter fitted to camera lens

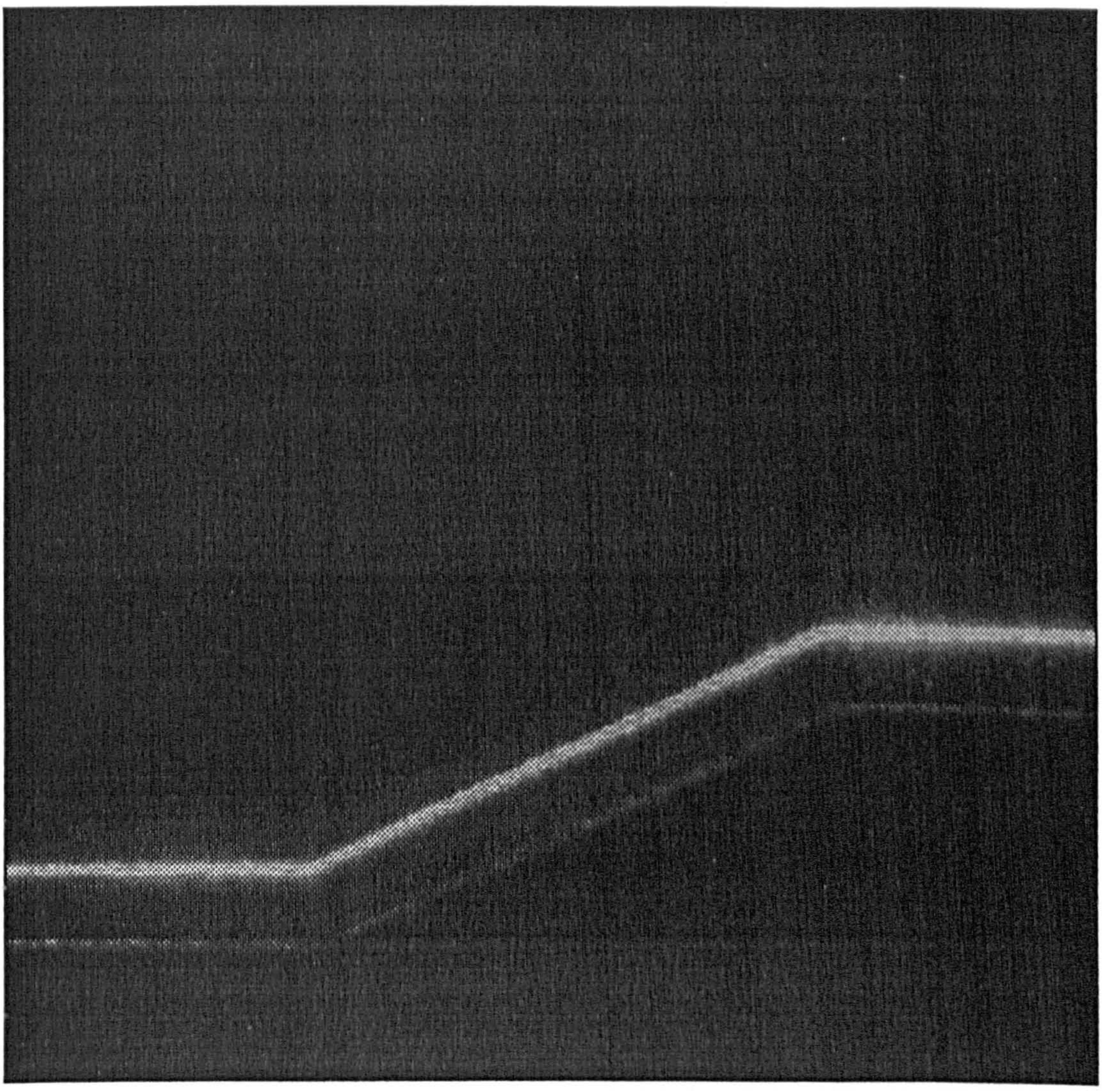


Image 4.5 Calibration sample with laser stripe before calibration

Line definitions

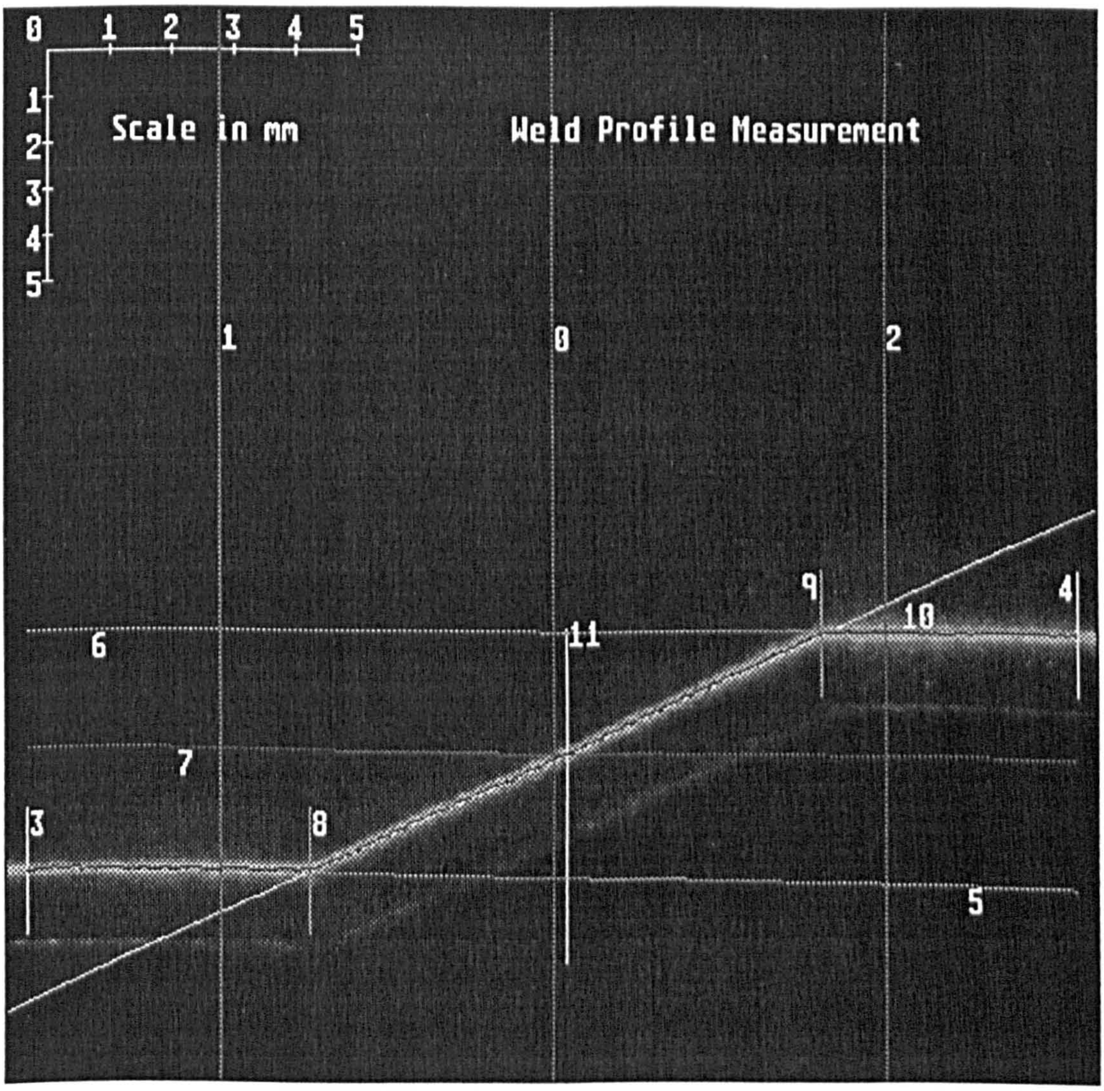
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	5	pixels
Window width	10	pixels
Calibration.factor.x	27.71	pixels/mm
Calibration.factor.y	22.40	pixels/mm
Corner.adjust	1.3	mm

Corner locations

Left.start.x	142	pixels
Right.start.x	382	pixels
Left.start.y	411	pixels
Right.start.y	299	pixels



Manual measurements

Width	10	mm
Reinforcement	-	mm
Left undercut	0	mm
Right undercut	0	mm
Plate displacement	5	mm
Plate angle	180	°
Left toe angle	30	°
Right toe angle	30	°
Left leg	-	mm
Right leg	-	mm

+/- 0.25mm or 10°

Gauge measurements

Width	10.09	mm
Reinforcement	-2.59	mm
Left undercut	0.0	mm
Right undercut	0.0	mm
Plate displacement	5.00	mm
Plate angle	178.67	°
Left toe angle	30.62	°
Right toe angle	31.59	°
Left leg	0.0	mm
Right leg	0.0	mm

Image 4.6 Calibration sample after calibration

Line definitions

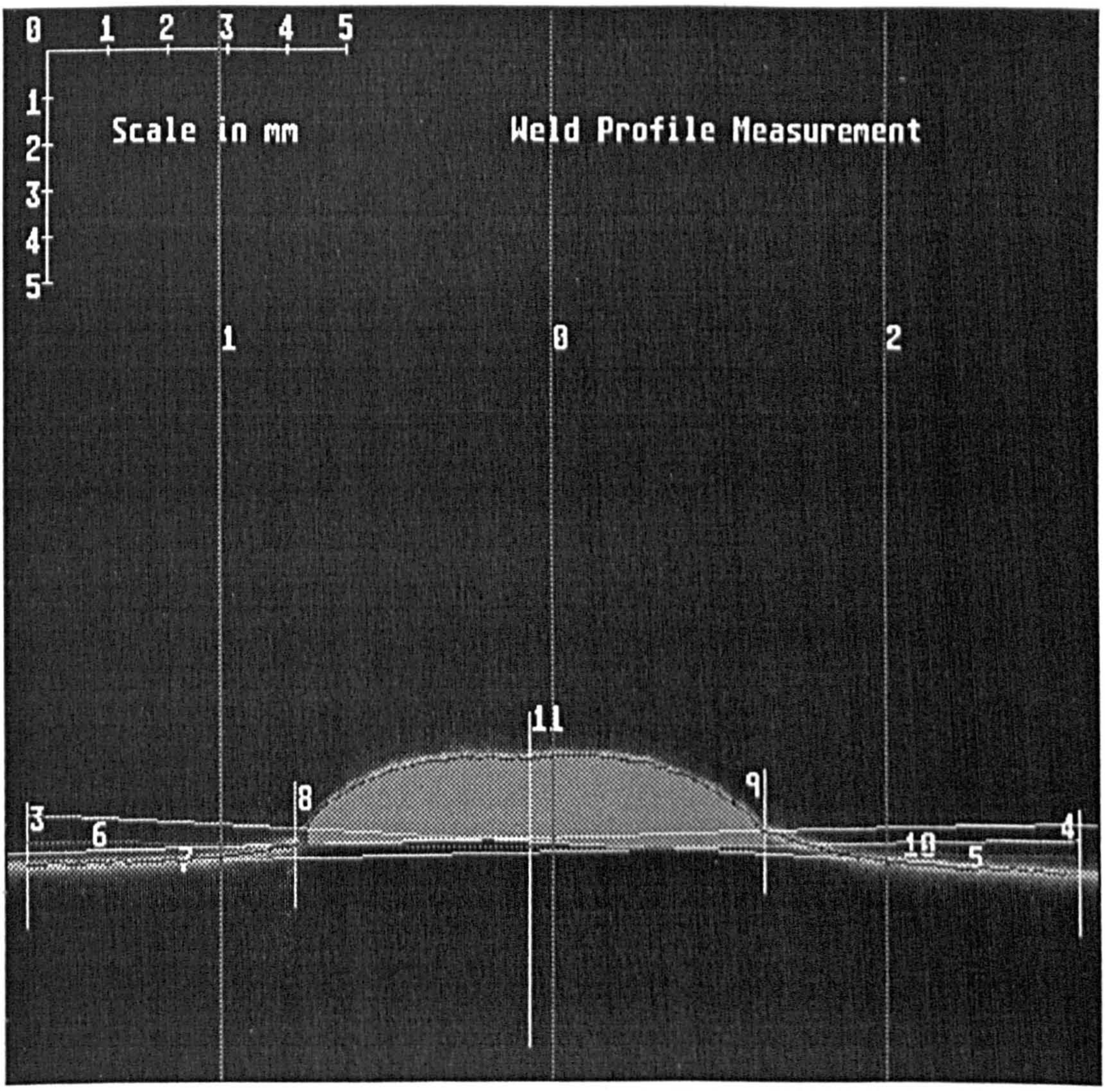
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	5	pixels
Window width	10	pixels
Calibration.factor.x	27.71	pixels/mm
Calibration.factor.y	22.40	pixels/mm
Corner.adjust	1.4	mm

Corner locations

Left.start.x	135	pixels
Right.start.x	355	pixels
Left.start.y	399	pixels
Right.start.y	393	pixels



Manual measurements

Width	7.8	mm
Reinforcement	2	mm
Left undercut	0	mm
Right undercut	0	mm
Plate displacement	-	mm
Plate angle	180	°
Left toe angle	60	°
Right toe angle	50	°
Left leg	-	mm
Right leg	-	mm

+/- 0.25mm or 10°

Gauge measurements

Width	7.8	mm
Reinforcement	1.9	mm
Left undercut	0.0	mm
Right undercut	0.0	mm
Plate displacement	0.2	mm
Plate angle	187.5	°
Left toe angle	63.0	°
Right toe angle	46.0	°
Left leg	5.7	mm
Right leg	2.1	mm

Image 4.7 Measured sample of Welding Institute sample butt plate 1 - "Excess penetration"

Line definitions

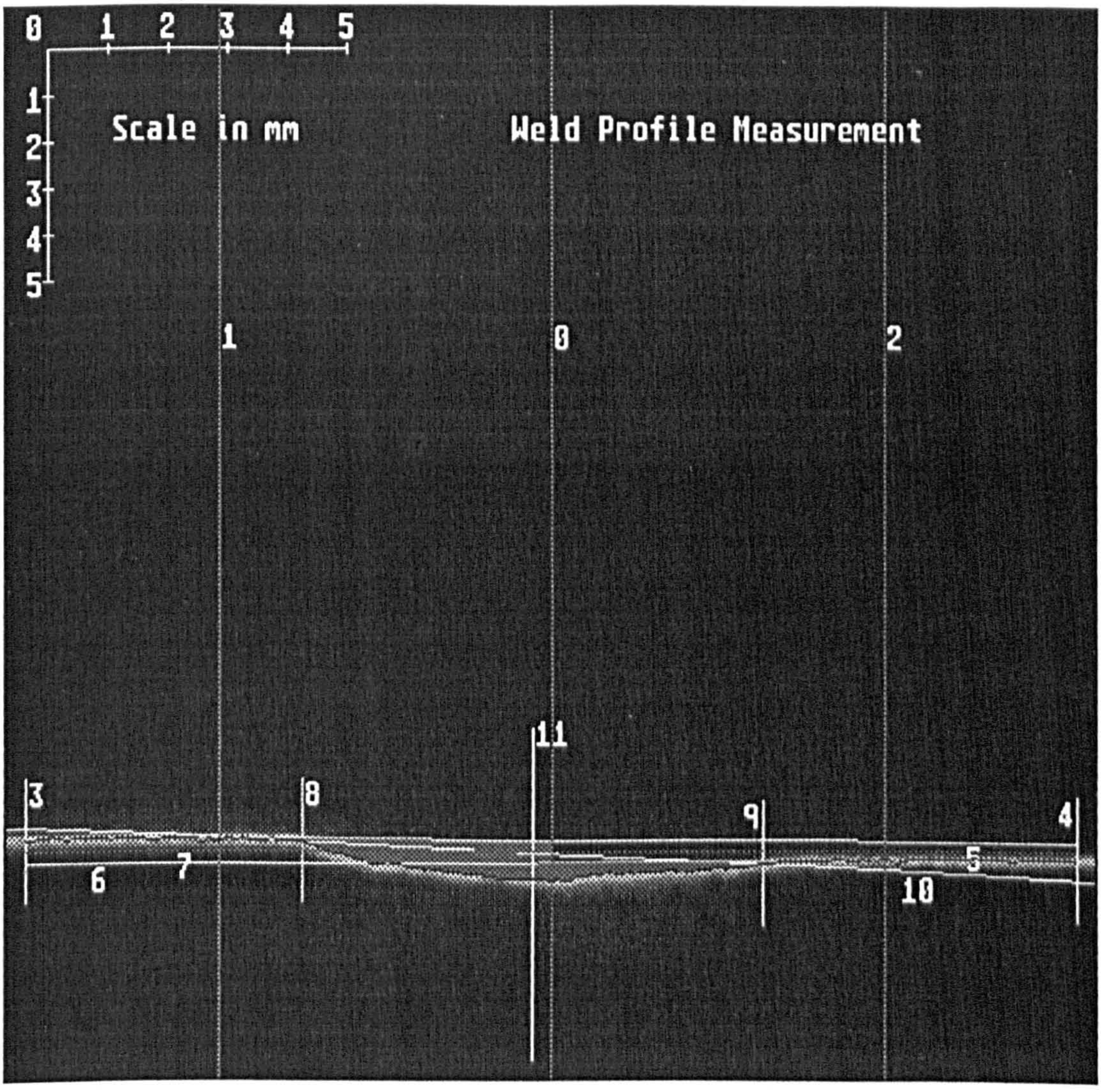
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	5	pixels
Window width	10	pixels
Calibration.factor.x	27.71	pixels/mm
Calibration.factor.y	22.40	pixels/mm
Corner.adjust	0.95	mm

Corner locations

Left.start.x	139	pixels
Right.start.x	355	pixels
Left.start.y	397	pixels
Right.start.y	408	pixels



Manual measurements

Width	8	mm
Reinforcement	-0.5	mm
Left undercut	-	mm
Right undercut	-	mm
Plate displacement	-	mm
Plate angle	180	°
Left toe angle	20	°
Right toe angle	10	°
Left leg	-	mm
Right leg	-	mm

+/- 0.25mm or 10°

Gauge measurements

Width	7.81	mm
Reinforcement	-0.63	mm
Left undercut	0.8	mm
Right undercut	0.4	mm
Plate displacement	0.45	mm
Plate angle	179.36	°
Left toe angle	24.54	°
Right toe angle	7.96	°
Left leg	0.0	mm
Right leg	0.0	mm

Image 4.8 Measured sample of Welding Institute sample butt plate 2 - "Surface concavity"

Line definitions

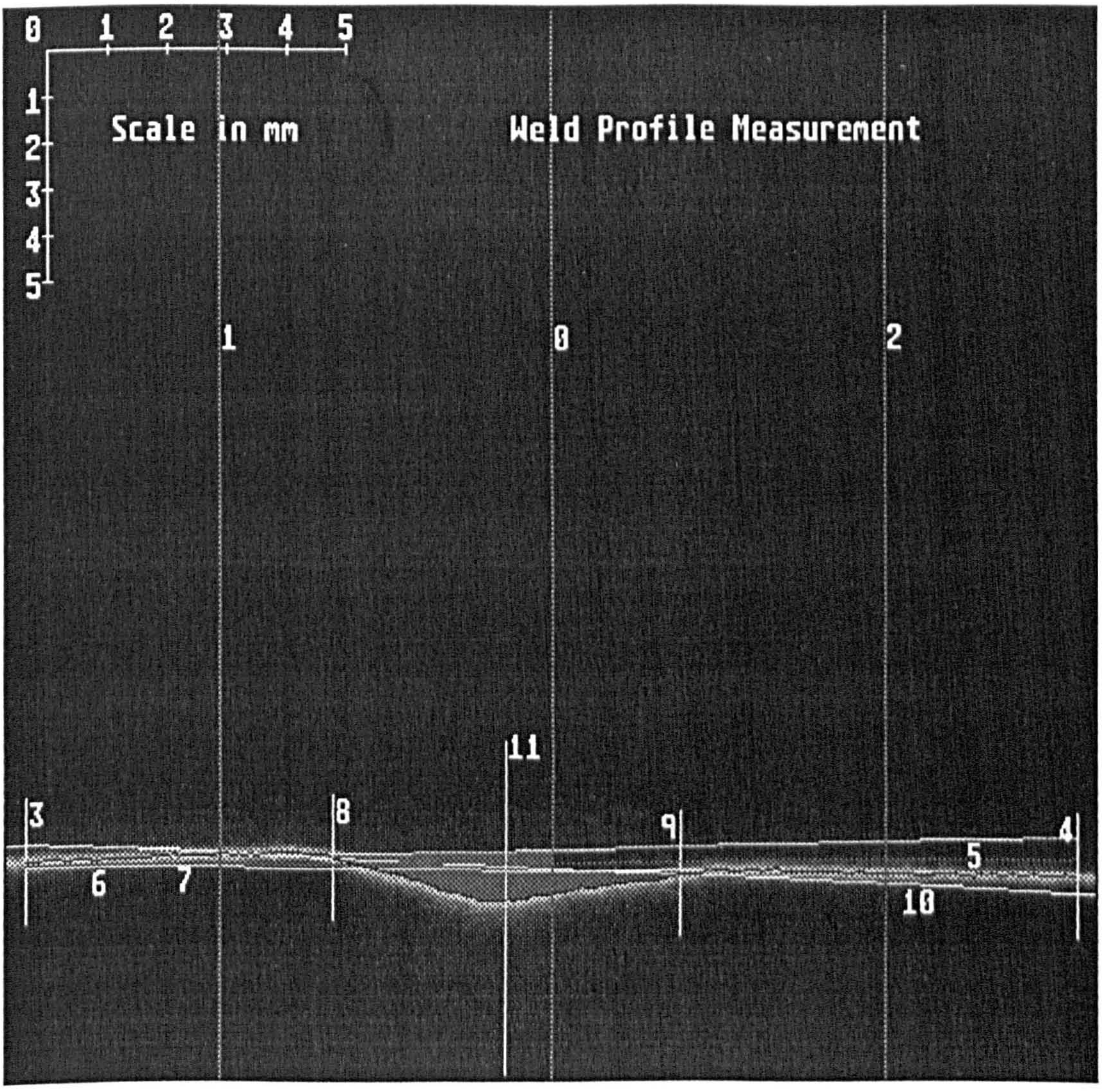
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	5	pixels
Window width	10	pixels
Calibration.factor.x	27.71	pixels/mm
Calibration.factor.y	22.40	pixels/mm
Corner.adjust	1.65	mm

Corner locations

Left.start.x	153	pixels
Right.start.x	316	pixels
Left.start.y	405	pixels
Right.start.y	413	pixels



Manual measurements

Width	6.5	mm
Reinforcement	-0.8	mm
Left undercut	-	mm
Right undercut	-	mm
Plate displacement	-	mm
Plate angle	180	°
Left toe angle	25	°
Right toe angle	20	°
Left leg	-	mm
Right leg	-	mm

+/- 0.25mm or 10°

Gauge measurements

Width	5.89	mm
Reinforcement	-0.8	mm
Left undercut	1.03	mm
Right undercut	0.0	mm
Plate displacement	0.36	mm
Plate angle	182.13	°
Left toe angle	12.22	°
Right toe angle	17.77	°
Left leg	0.0	mm
Right leg	0.0	mm

Image 4.9 Measured sample of Welding Institute sample butt plate 3 - "Root concavity"

Line definitions

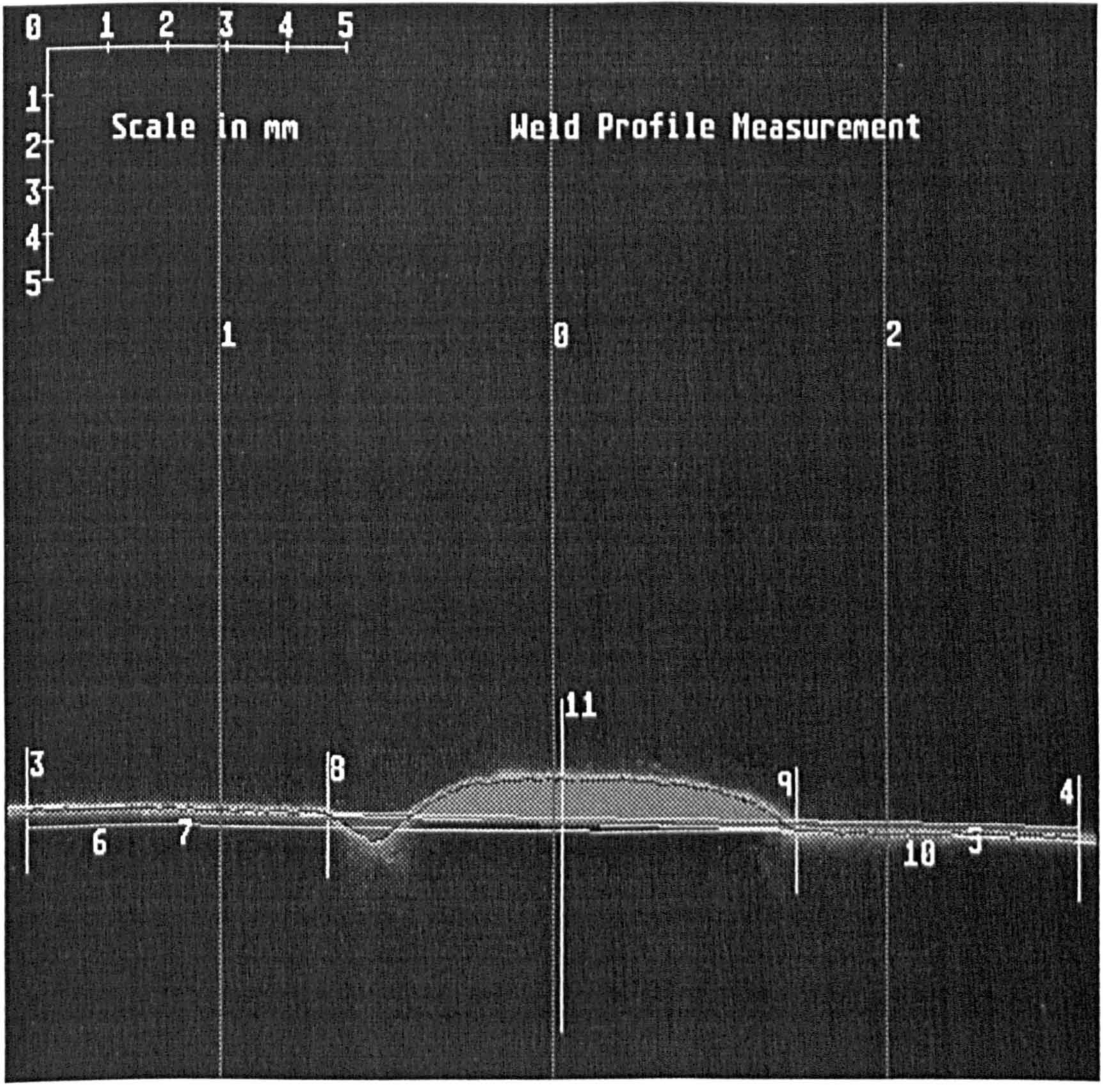
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	5	pixels
Window width	10	pixels
Calibration.factor.x	27.71	pixels/mm
Calibration.factor.y	22.40	pixels/mm
Corner.adjust	1.4	mm

Corner locations

Left.start.x	150	pixels
Right.start.x	370	pixels
Left.start.y	386	pixels
Right.start.y	394	pixels



Manual measurements

Width	8	mm
Reinforcement	1	mm
Left undercut	0.5	mm
Right undercut	0	mm
Plate displacement	-	mm
Plate angle	180	°
Left toe angle	30	°
Right toe angle	40	°
Left leg		mm
Right leg		mm

+/- 0.25mm or 10°

Gauge measurements

Width	7.95	mm
Reinforcement	0.98	mm
Left undercut	0.58	mm
Right undercut	0.0	mm
Plate displacement	0.31	mm
Plate angle	179.29	°
Left toe angle	29.32	°
Right toe angle	42.97	°
Left leg	0.0	mm
Right leg	0.0	mm

Image 4.10 Measured sample of Welding Institute sample butt plate 4 - "Undercut"

Line definitions

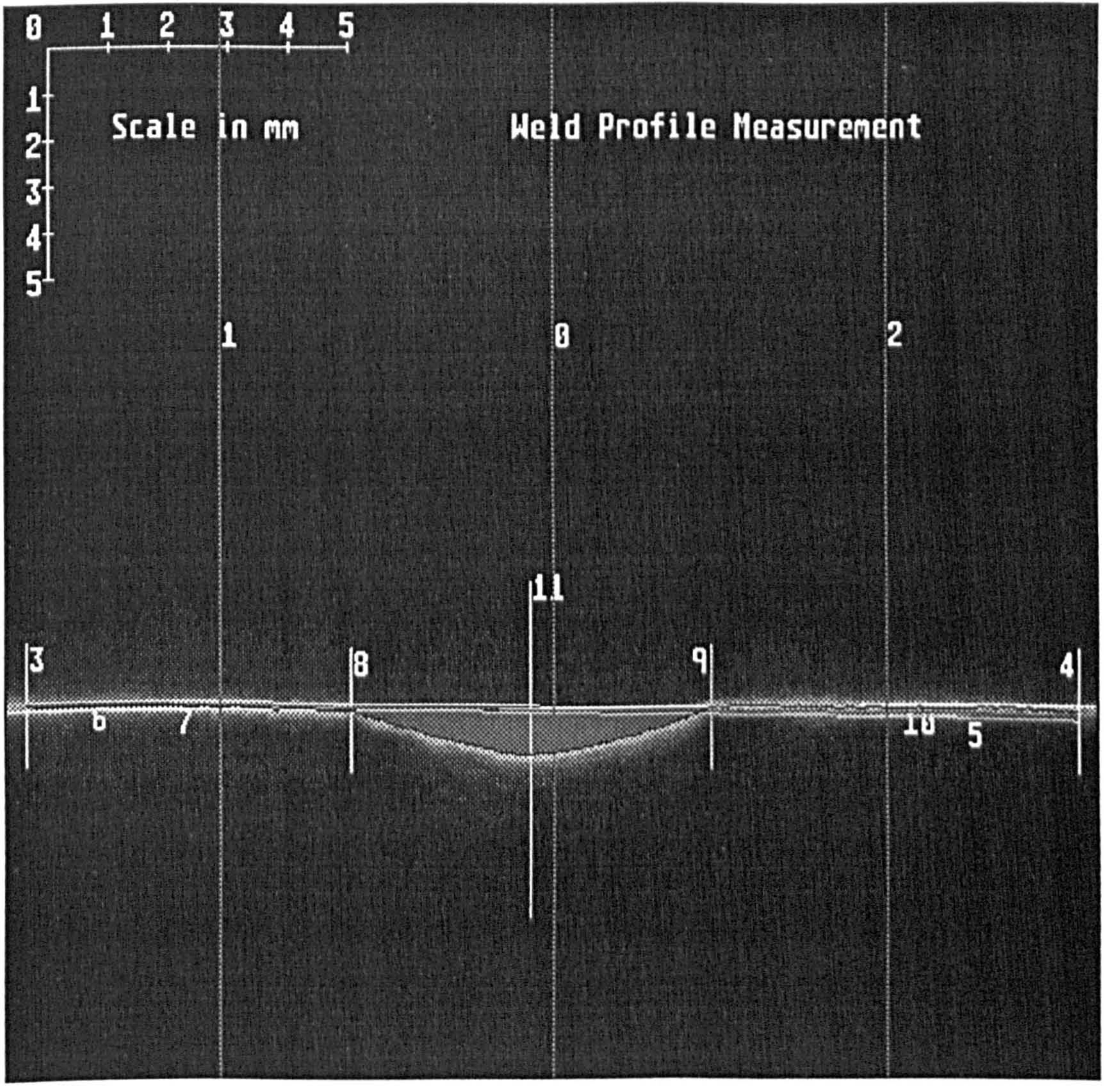
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	5	pixels
Window width	10	pixels
Calibration.factor.x	27.71	pixels/mm
Calibration.factor.y	22.40	pixels/mm
Corner.adjust	1.4	mm

Corner locations

Left.start.x	161	pixels
Right.start.x	330	pixels
Left.start.y	336	pixels
Right.start.y	335	pixels



Manual measurements

Width	6	mm
Reinforcement	-0.9	mm
Left undercut	0.9	mm
Right undercut	0	mm
Plate displacement	-	mm
Plate angle	180	°
Left toe angle	30	°
Right toe angle	30	°
Left leg	-	mm
Right leg	-	mm

+/- 0.25mm or 10°

Gauge measurements

Width	6.10	mm
Reinforcement	-0.94	mm
Left undercut	0.89	mm
Right undercut	0.0	mm
Plate displacement	0.13	mm
Plate angle	179.29	°
Left toe angle	26.73	°
Right toe angle	27.07	°
Left leg	0.0	mm
Right leg	0.0	mm

Image 4.11 Measured sample of Welding Institute sample butt plate 5 - "Incomplete filled groove"

Line definitions

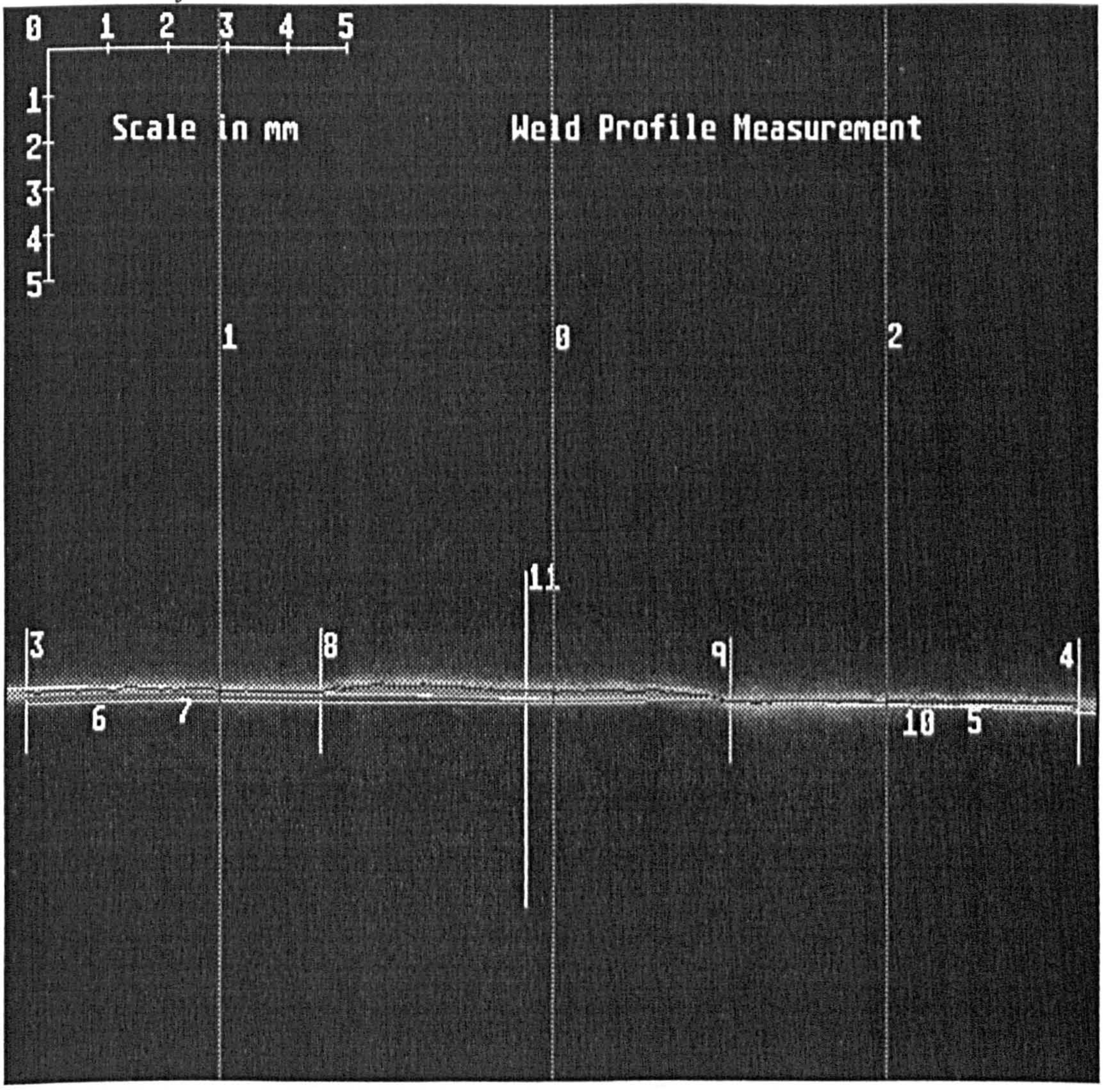
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	5	pixels
Window width	10	pixels
Calibration.factor.x	27.71	pixels/mm
Calibration.factor.y	22.40	pixels/mm
Corner.adjust	0.75	mm

Corner locations

Left.start.x	147	pixels
Right.start.x	339	pixels
Left.start.y	326	pixels
Right.start.y	331	pixels



Manual measurements

Width	7	mm
Reinforcement	0.5	mm
Left undercut	0	mm
Right undercut	0	mm
Plate displacement	-	mm
Plate angle	180	°
Left toe angle	20	°
Right toe angle	20	°
Left leg	-	mm
Right leg	-	mm

+/- 0.25mm or 10°

Gauge measurements

Width	6.93	mm
Reinforcement	0.36	mm
Left undercut	0.0	mm
Right undercut	0.0	mm
Plate displacement	0.18	mm
Plate angle	178.58	°
Left toe angle	22.75	°
Right toe angle	20.61	°
Left leg	0.0	mm
Right leg	0.0	mm

Image 4.12 Measured sample of Welding Institute sample butt plate 6 - "Acceptable root profile"

Line definitions

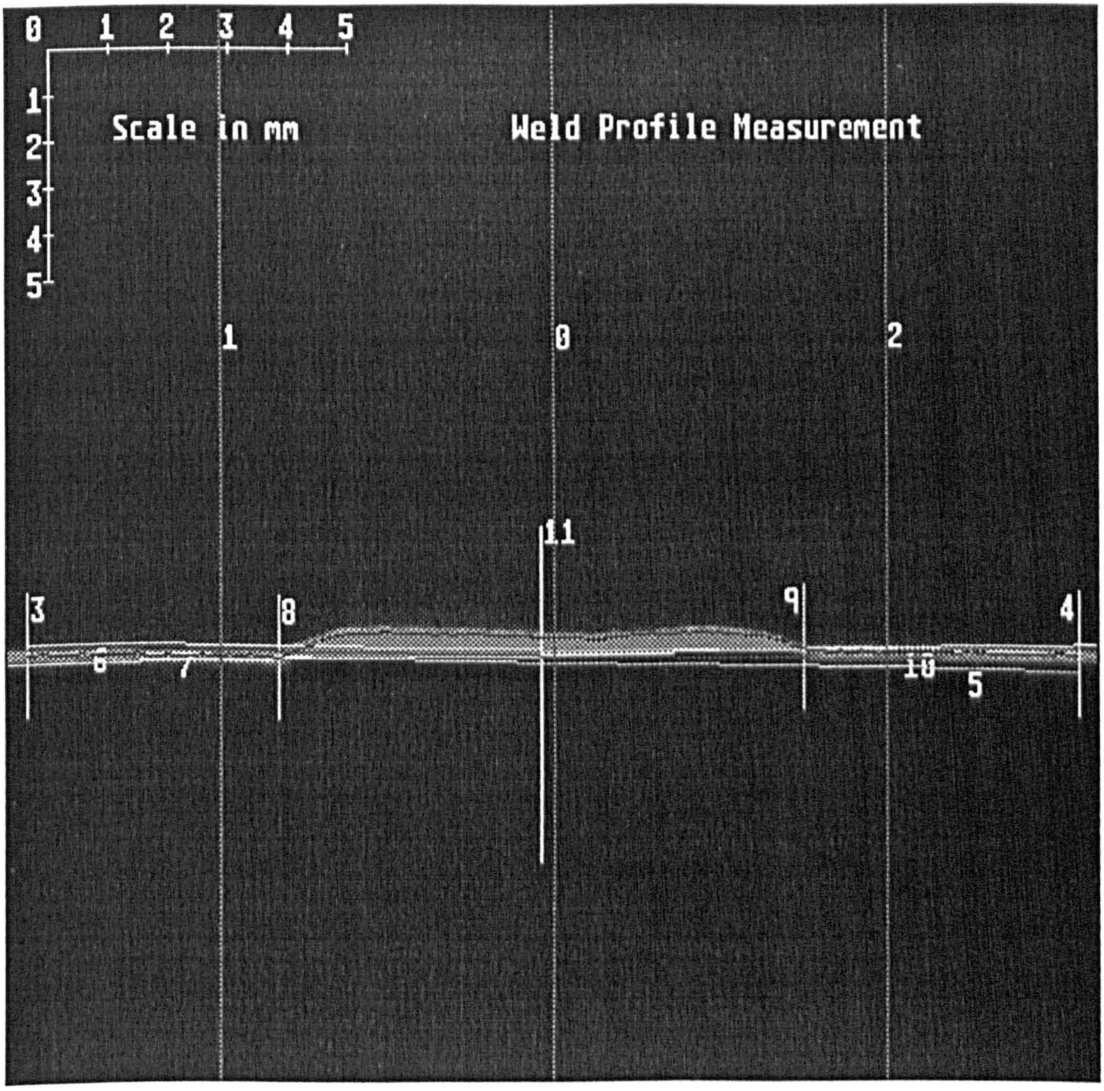
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	5	pixels
Window width	10	pixels
Calibration.factor.x	27.71	pixels/mm
Calibration.factor.y	22.40	pixels/mm
Corner.adjust	1.4	mm

Corner locations

Left.start.x	127	pixels
Right.start.x	373	pixels
Left.start.y	310	pixels
Right.start.y	306	pixels



Manual measurements

Width	9	mm
Reinforcement	0.5	mm
Left undercut	0	mm
Right undercut	0	mm
Plate displacement	-	mm
Plate angle	180	°
Left toe angle	30	°
Right toe angle	20	°
Left leg	-	mm
Right leg	-	mm

+/- 0.25mm or 10°

Gauge measurements

Width	8.88	mm
Reinforcement	0.54	mm
Left undercut	0.04	mm
Right undercut	0.0	mm
Plate displacement	0.31	mm
Plate angle	179.36	°
Left toe angle	27.85	°
Right toe angle	18.85	°
Left leg	0.0	mm
Right leg	0.0	mm

Image 4.13 Measured sample of Welding Institute sample butt plate 7 - "Acceptable surface profile"

Line definitions

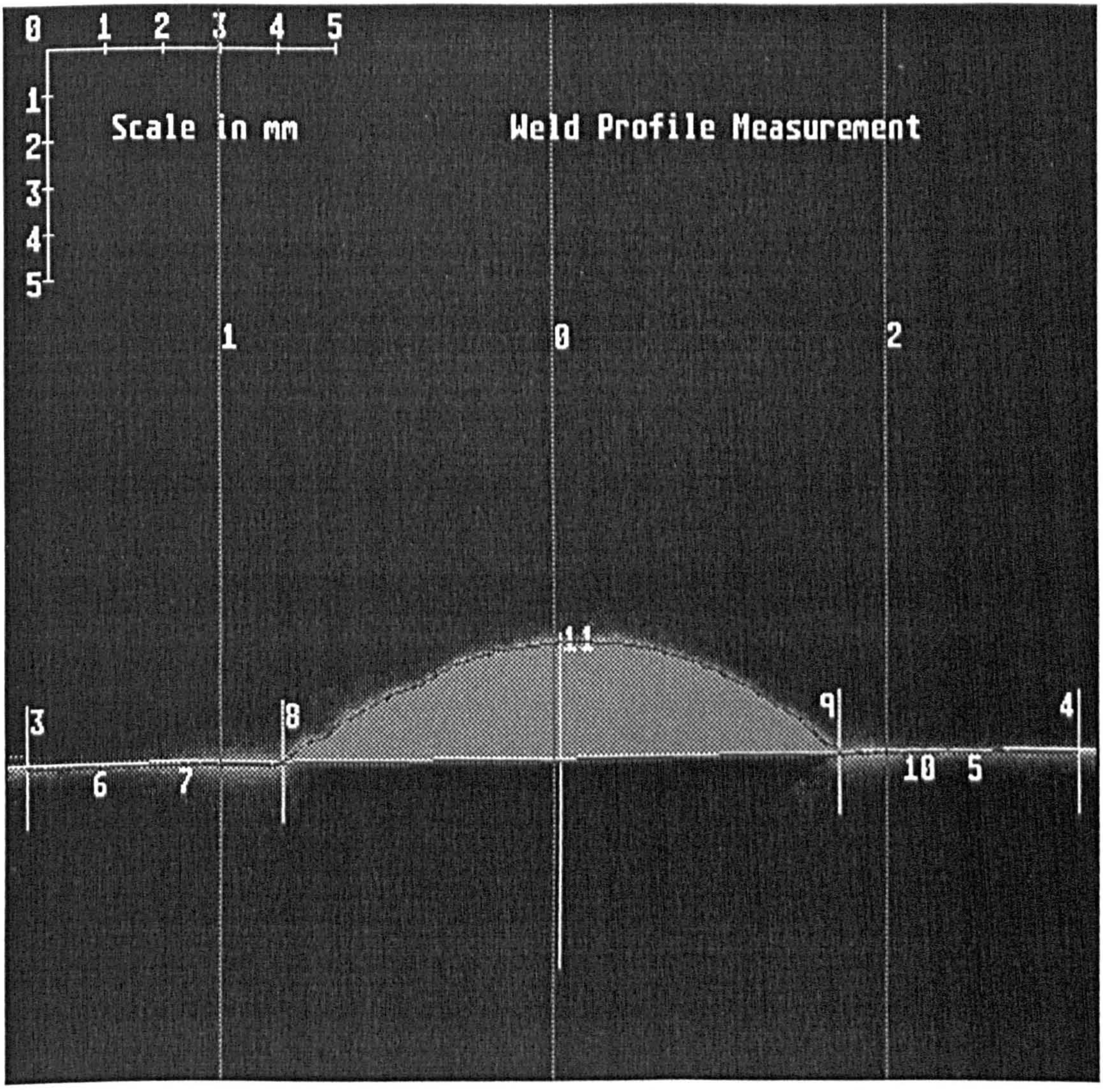
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	5	pixels
Window width	6	pixels
Calibration.factor.x	27.25	pixels/mm
Calibration.factor.y	22.4	pixels/mm
Corner.adjust	1.1	mm

Corner locations

Left.start.x	129	pixels
Right.start.x	390	pixels
Left.start.y	360	pixels
Right.start.y	356	pixels



Manual measurements

Width	9.5	mm
Reinforcement	2.5	mm
Left undercut	0	mm
Right undercut	0	mm
Plate displacement	0	mm
Plate angle	180	°
Left toe angle	40	°
Right toe angle	45	°
Left leg	-	mm
Right leg	-	mm

+/- 0.25mm or 10°

Gauge measurements

Width	9.58	mm
Reinforcement	2.54	mm
Left undercut	0.0	mm
Right undercut	0.0	mm
Plate displacement	0.0	mm
Plate angle	180.0	°
Left toe angle	40.7	°
Right toe angle	42.19	°
Left leg	0.0	mm
Right leg	0.0	mm

Image 4.15 Measured sample of weld bead on plate - M.I.G. welded

Line definitions

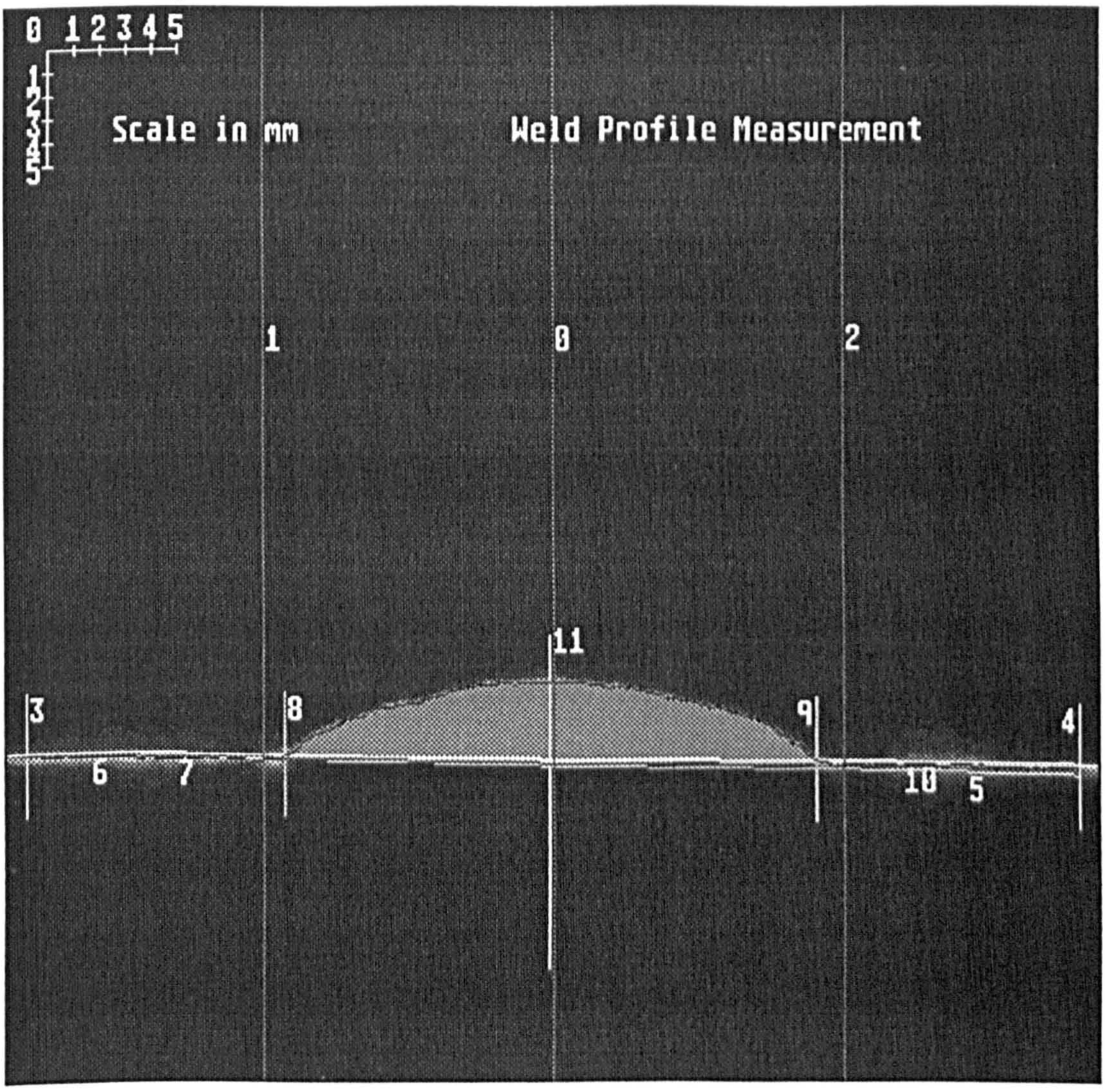
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	5	pixels
Window width	10	pixels
Calibration.factor.x	11.55	pixels/mm
Calibration.factor.y	10.60	pixels/mm
Corner.adjust	1.4	mm

Corner locations

Left.start.x	130	pixels
Right.start.x	379	pixels
Left.start.y	356	pixels
Right.start.y	359	pixels



Manual measurements

Width	21.5	mm
Reinforcement	3.5	mm
Left undercut	0	mm
Right undercut	0	mm
Plate displacement	0	mm
Plate angle	180	°
Left toe angle	45	°
Right toe angle	40	°
Left leg	-	mm
Right leg	-	mm

+/- 0.25mm or 10°

Gauge measurements

Width	21.57	mm
Reinforcement	3.77	mm
Left undercut	0.0	mm
Right undercut	0.0	mm
Plate displacement	0.19	mm
Plate angle	179.5	°
Left toe angle	39.25	°
Right toe angle	34.25	°
Left leg	0.0	mm
Right leg	0.0	mm

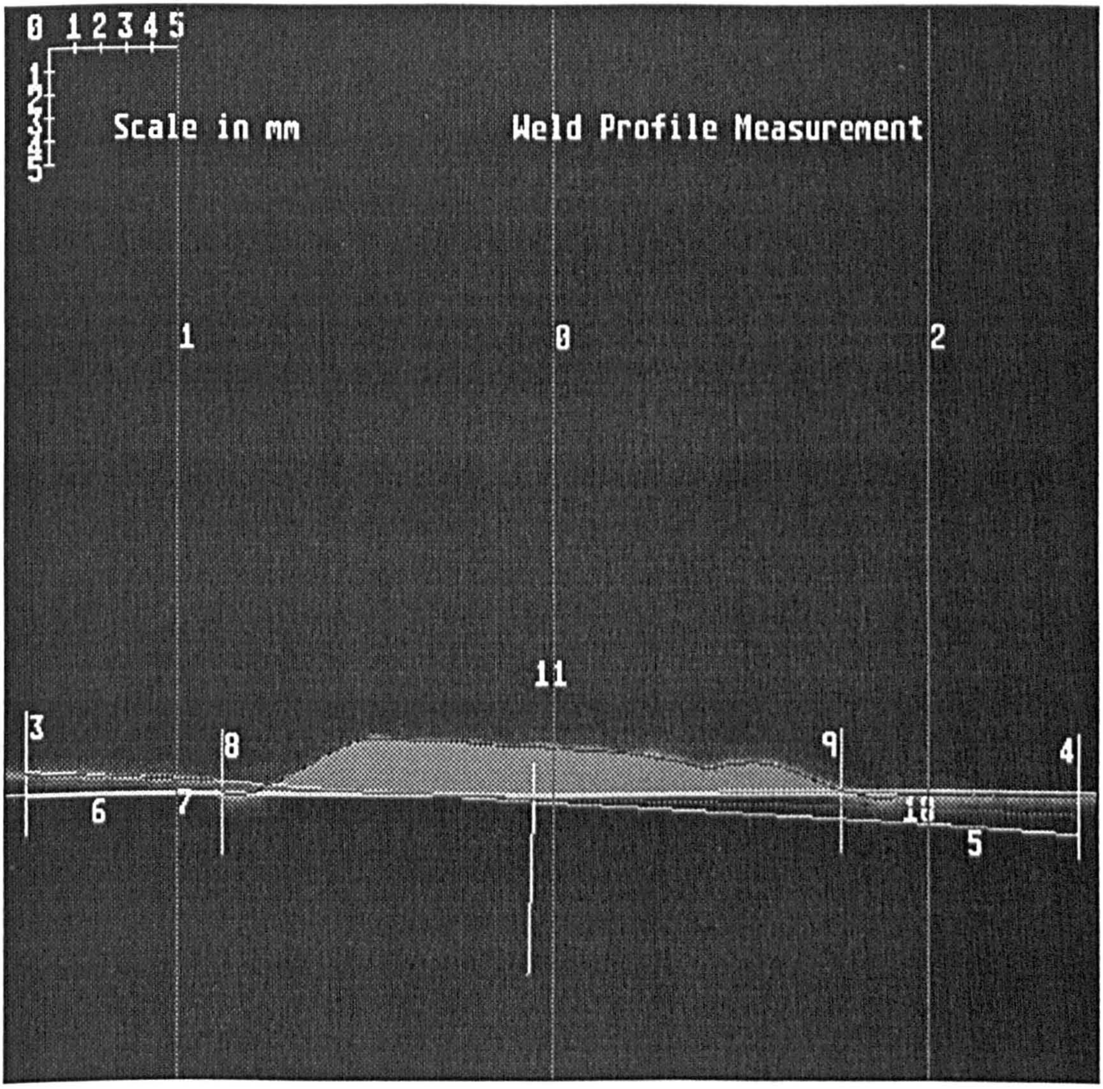
Image 4.16 Measured sample of large bead on plate - a submerged arc welded

Line definitions

0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values		
Search length	5	pixels
Window width	3	pixels
Calibration.factor.x	11.55	pixels/mm
Calibration.factor.y	10.6	pixels/mm
Corner.adjust	1.0	mm

Corner locations		
Left.start.x	101	pixels
Right.start.x	391	pixels
Left.start.y	375	pixels
Right.start.y	374	pixels



Manual measurements			**Gauge measurements**		
Width	25	mm	Width	25.12	mm
Reinforcement	2.5	mm	Reinforcement	2.45	mm
Left undercut	1	mm	Left undercut	1.46	mm
Right undercut	0	mm	Right undercut	0.0	mm
Plate displacement	-	mm	Plate displacement	0.47	mm
Plate angle	<180	°	Plate angle	176.2	°
Left toe angle	50	°	Left toe angle	65.35	°
Right toe angle	20	°	Right toe angle	47.45	°
Left leg	-	mm	Left leg	8.23	mm
Right leg	-	mm	Right leg	16.89	mm

+/- 0.25mm or 10°

Image 4.17 Measured sample of a multipass weld - Manual metal arc welded

Line definitions

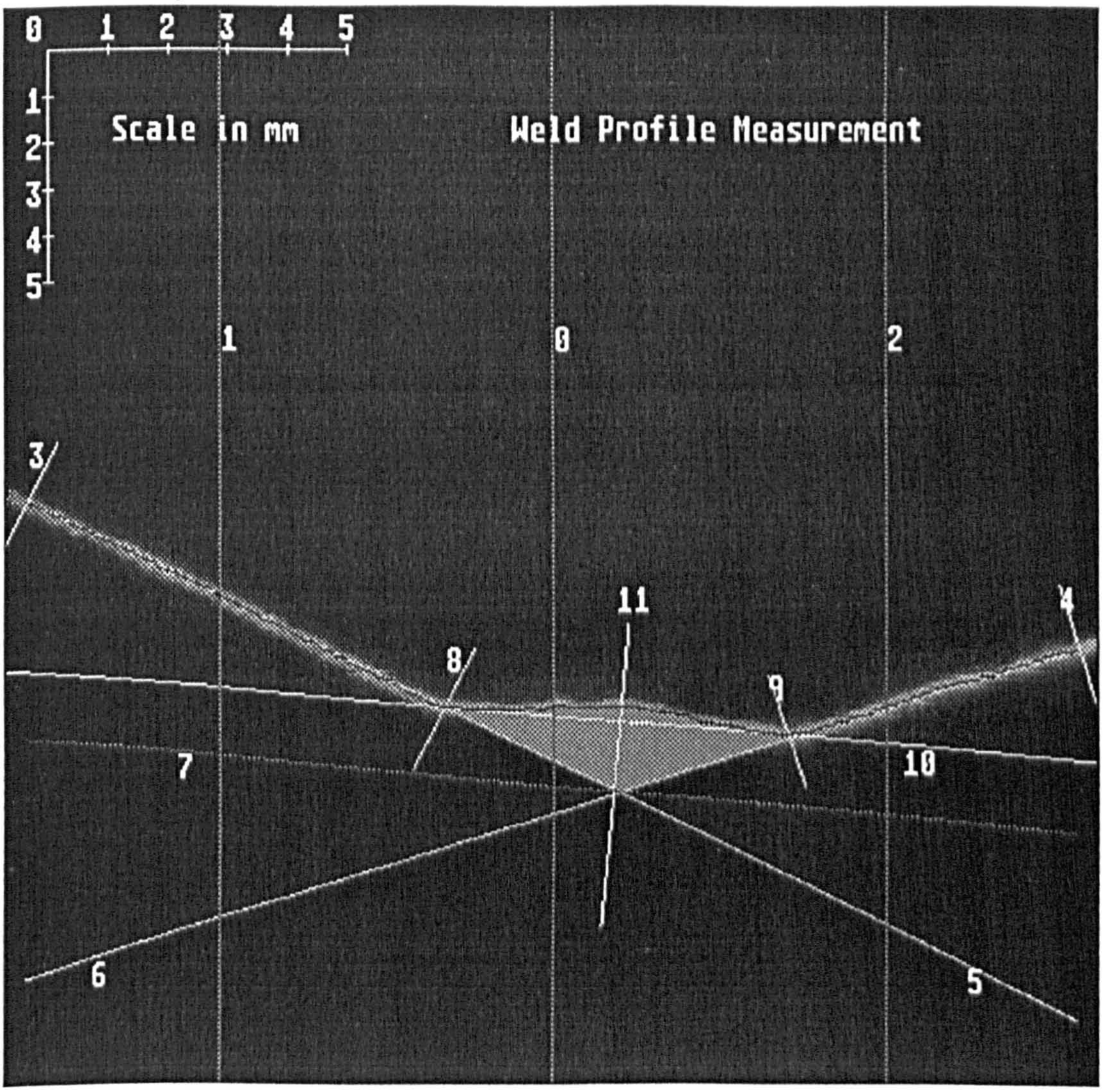
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	5	pixels
Window width	10	pixels
Calibration.factor.x	27.71	pixels/mm
Calibration.factor.y	22.40	pixels/mm
Corner.adjust	1.3	mm

Corner locations

Left.start.x	205	pixels
Right.start.x	366	pixels
Left.start.y	332	pixels
Right.start.y	346	pixels



Manual measurements

Width	6	mm
Reinforcement	-	mm
Left undercut	0	mm
Right undercut	0	mm
Plate displacement	-	mm
Plate angle	130	°
Left toe angle	-	°
Right toe angle	-	°
Left leg	-	mm
Right leg	-	mm

+/- 0.25mm or 10°

Gauge measurements

Width	5.84	mm
Reinforcement	1.59	mm
Left undercut	0.07	mm
Right undercut	0.0	mm
Plate displacement	5.68	mm
Plate angle	126.78	°
Left toe angle	22.27	°
Right toe angle	2.34	°
Left leg	3.45	mm
Right leg	3.13	mm

Image 4.18 Measured sample of weld between plates an angle - manual metal arc welded

Line definitions

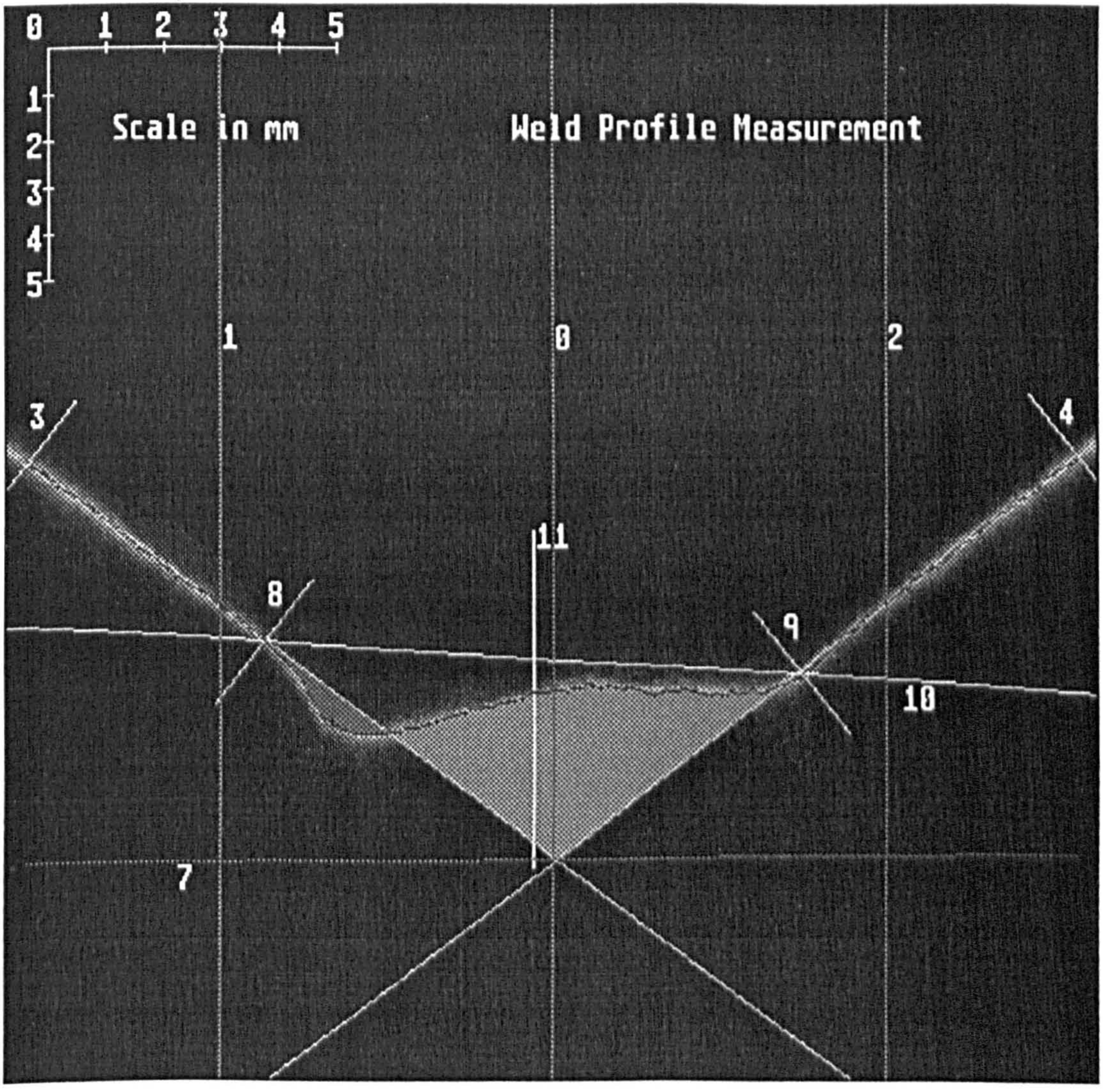
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	5	pixels
Window width	10	pixels
Calibration.factor.x	27.25	pixels/mm
Calibration.factor.y	22.40	pixels/mm
Corner.adjust	1.0	mm

Corner locations

Left.start.x	121	pixels
Right.start.x	373	pixels
Left.start.y	302	pixels
Right.start.y	318	pixels



Manual measurements

Width	9	mm
Reinforcement	4	mm
Left undercut	1	mm
Right undercut	<0.5	mm
Plate displacement	0	mm
Plate angle	90	°
Left toe angle	-	°
Right toe angle	-	°
Left leg	5.5	mm
Right leg	5	mm

+/- 0.25mm or 10°

Gauge measurements

Width	9.27	mm
Reinforcement	4.64	mm
Left undercut	0.54	mm
Right undercut	0.15	mm
Plate displacement	0.36	mm
Plate angle	93.74	°
Left toe angle	16.66	°
Right toe angle	29.55	°
Left leg	6.79	mm
Right leg	5.76	mm

Image 4.19 Measured sample of a small fillet bead on plate - manual metal arc welded

Line definitions

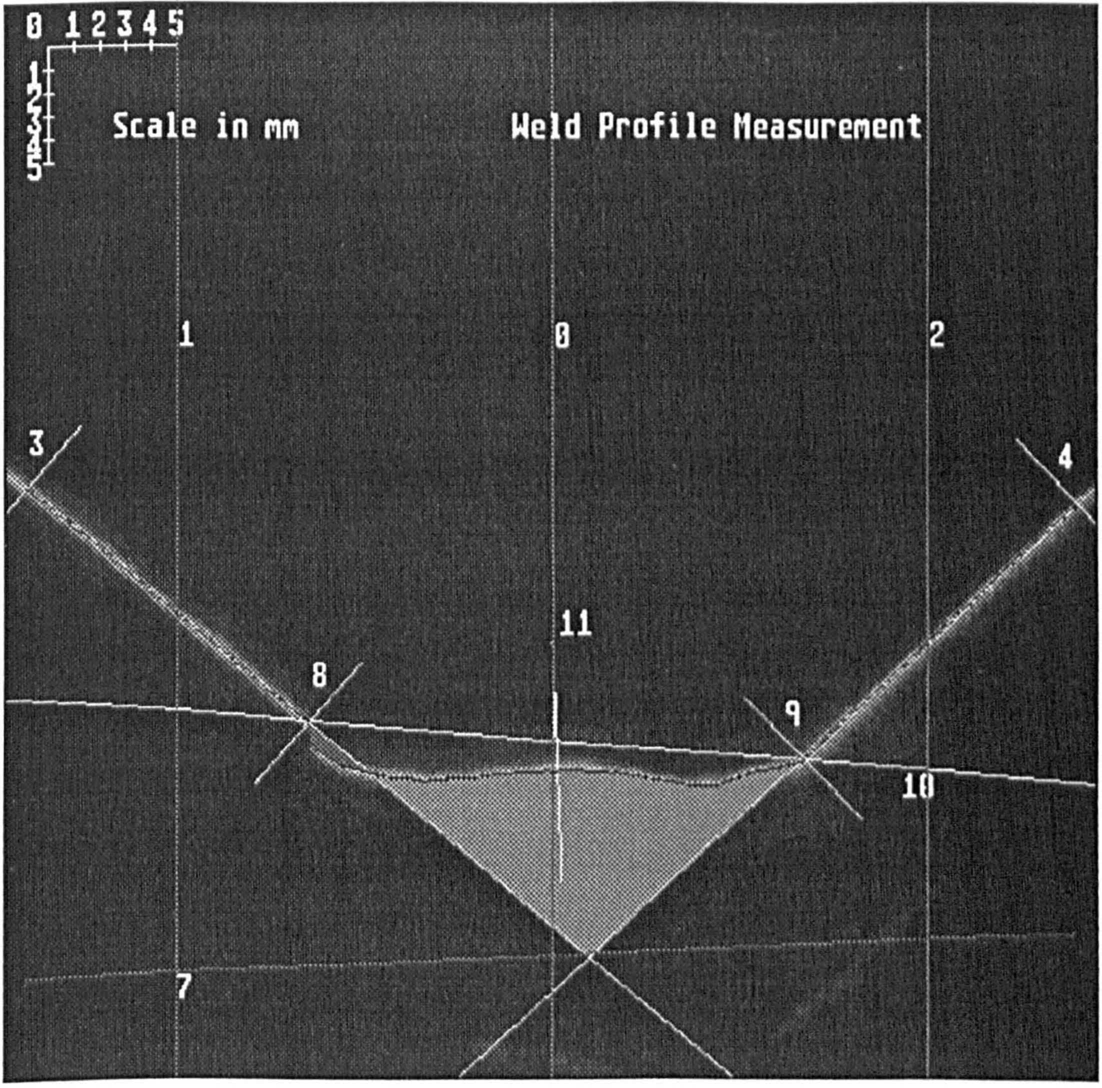
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	22	pixels
Window width	3	pixels
Calibration.factor.x	11.55	pixels/mm
Calibration.factor.y	10.60	pixels/mm
Corner.adjust	1.0	mm

Corner locations

Left.start.x	142	pixels
Right.start.x	374	pixels
Left.start.y	341	pixels
Right.start.y	360	pixels



Manual measurements

Width	20	mm
Reinforcement	7.5	mm
Left undercut	1	mm
Right undercut	0	mm
Plate displacement	-	mm
Plate angle	90	°
Left toe angle	-	°
Right toe angle	-	°
Left leg	15	mm
Right leg	12	mm

+/- 0.25mm or 10°

Gauge measurements

Width	20.17	mm
Reinforcement	10.94	mm
Left undercut	0.71	mm
Right undercut	0.13	mm
Plate displacement	6.89	mm
Plate angle	91.68	°
Left toe angle	84.76	°
Right toe angle	65.35	°
Left leg	15.38	mm
Right leg	12.38	mm

Image 4.20 Measured example of multipass fillet weld - manual metal arc welded

Line definitions

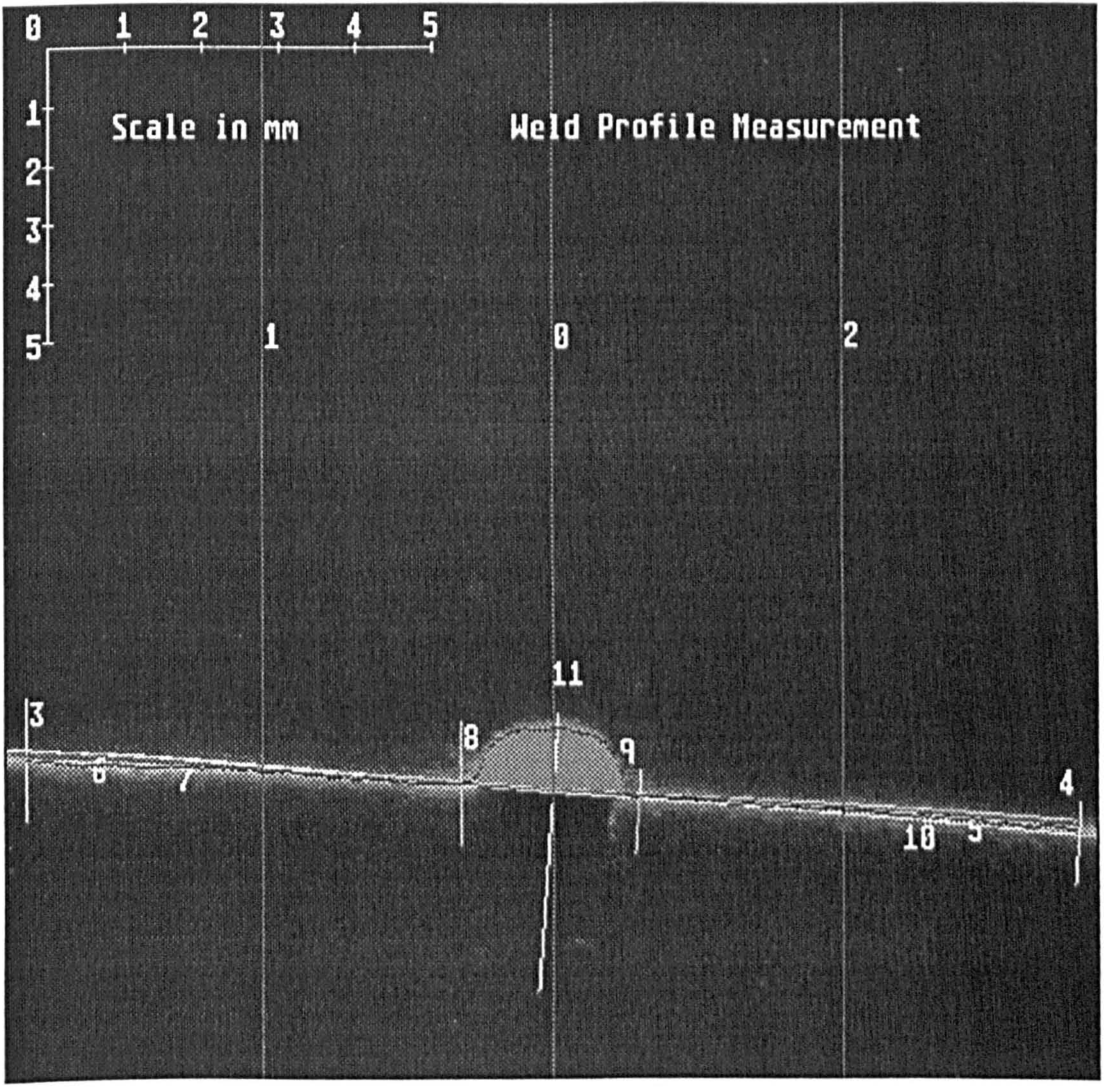
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	9	pixels
Window width	10	pixels
Calibration.factor.x	35.91	pixels/mm
Calibration.factor.y	27.6	pixels/mm
Corner.adjust	1.35	mm

Corner locations

Left.start.x	213	pixels
Right.start.x	296	pixels
Left.start.y	371	pixels
Right.start.y	378	pixels



Manual measurements

Width	2	mm
Reinforcement	1	mm
Left undercut	0	mm
Right undercut	0	mm
Plate displacement	-	mm
Plate angle	180	°
Left toe angle	60	°
Right toe angle	70	°
Left leg	-	mm
Right leg	-	mm

+/- 0.25mm or 10°

Gauge measurements

Width	2.33	mm
Reinforcement	0.99	mm
Left undercut	0.36	mm
Right undercut	0.26	mm
Plate displacement	0.15	mm
Plate angle	181.26	°
Left toe angle	60.24	°
Right toe angle	65.22	°
Left leg	2.26	mm
Right leg	0.09	mm

Image 4.21 Measured sample of a solder bead laid down in fuse manufacturing process

4.10.2 Data from test samples

The following data is generated for each of the sample images in the previous section and shows the pixel locations of the weld start points and the corresponding measurements for 100 samples at the same position on each weld. The variation is then used to deturmine the accuracy of the gauge. The results for the 100 samples are shown in the form of the location or measurement with the number of times which it occured. Below these the mean, mode and maximum and minimum values and the maximum variation are given.

The measurement values are as follows:

l_s_x	left.start.x
l_s_y	left.start.y
r_s_x	right.start.x
r_s_y	right.start.y

width	width
reinf	reinforcement
l_uc	left undercut
r_uc	right undercut
disp	plate displacement (fit-up)
angle	plate angle
l_toe	left toe angle
r_toe	right toe angle
l_leg	left leg
r_leg	right leg

Graphs of the locations and measurements of the 100 samples are given in the following section.

Specimen sample - CALIB			search.length		5		w.width	10	cal.x	27.72	cal.y	22.40	cnr_adj	1.4	
	l_s_x	r_s_x	l_s_y	r_s_y		width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence															
Num	146.00	382.00	409.00	299.00		9.80	2.80	0.00	0.00	5.10	180.50	30.00	30.00	0.00	0.00
Freq	35	78	74	66		55	32	101	101	76	32	79	101	101	101
Num	144.00	383.00	410.00	298.00		9.90	2.60			5.20	179.30	29.00			
Freq	21	23	27	35		43	12			20	12	17			
Num	147.00					10.00	-2.40			5.30	180.40	31.00			
Freq	37					3	32			5	25	5			
Num	145.00						3.00				179.40				
Freq	5						2				13				
Num	148.00						2.50				180.60				
Freq	2						5				18				
Num	143.00						-2.50				180.30				
Freq	1						1				1				
Num							2.70								
Freq							10								
Num							-2.70								
Freq							7								
Mean															
	145.91	382.23	409.27	298.65		9.85	0.67	0.00	0.00	5.13	180.21	29.88	30.00	0.00	0.00
Mode															
	147.00	382.00	409.00	299.00		9.80	2.80	0.00	0.00	5.10	180.50	30.00	30.00	0.00	0.00
Max															
	148.00	383.00	410.00	299.00		10.00	3.00	0.00	0.00	5.30	180.60	31.00	30.00	0.00	0.00
Min															
	143.00	382.00	409.00	298.00		9.80	-2.70	0.00	0.00	5.10	179.30	29.00	30.00	0.00	0.00
Maximum deviation															
	5.00	1.00	1.00	1.00		0.20	5.70	0.00	0.00	0.20	1.30	2.00	0.00	0.00	0.00

Specimen sample - PL1			search.length		5		w.width	10	cal.x	27.71	cal.y	22.40	cnr_adj	1.9	
	l_s_x	r_s_x	l_s_y	r_s_y		width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence															
Num	121.00	336.00	396.00	393.00		7.80	1.90	0.00	0.00	0.20	187.50	64.00	46.00	2.10	5.70
Freq	96	88	99	89		97	99	99	99	96	58	42	28	61	64
Num	122.00	337.00		394.00		7.70				0.10	187.40	63.00	48.00	1.90	5.60
Freq	3	11		10		2				3	34	54	18	19	3
Num											187.60	62.00	47.00	2.00	5.90
Freq											6	3	12	6	19
Num											187.20		45.00	1.80	6.00
Freq											1		24	7	7
Num													50.00	1.60	6.20
Freq													4	1	1
Num													49.00	1.50	6.30
Freq													9	5	5
Num													51.00		
Freq													2		
Num													44.00		
Freq													2		
Mean															
	121.03	336.11	396.00	393.10		7.80	1.90	0.00	0.00	0.20	187.47	63.39	46.74	2.00	5.79
Mode															
	121.00	336.00	396.00	393.00		7.80	1.90	0.00	0.00	0.20	187.50	63.00	46.00	2.10	5.70
Max															
	122.00	337.00	396.00	394.00		7.80	1.90	0.00	0.00	0.20	187.60	64.00	51.00	2.10	6.30
Min															
	121.00	336.00	396.00	393.00		7.70	1.90	0.00	0.00	0.10	187.20	62.00	44.00	1.50	5.60
Maximum deviation															
	1.00	1.00	0.00	1.00		0.10	0.00	0.00	0.00	0.10	0.40	2.00	7.00	0.60	0.70

Specimen sample - PL2			search.length		5		w.width	10	cal.x	27.71	cal.y	22.40	cnr_adj	0.95	
	l_s_x	r_s_x	l_s_y	r_s_y		width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence															
Num	139.00	363.00	397.00	407.00		8.10	-0.50	0.70	0.40	0.40	178.20	25.00	8.00	0.00	0.00
Freq	85	74	46	79		71	85	95	99	80	62	33	39	99	99
Num	141.00	364.00	398.00	408.00		8.00	-0.40	0.80		0.30	178.40	21.00	13.00		
Freq	4	4	53	20		3	13	4		19	16	44	2		
Num	140.00	369.00				8.30	-0.60				178.80	22.00	11.00		
Freq	1	1				1	1				1	12	41		
Num	137.00	353.00				7.70					178.10	27.00	9.00		
Freq	5	3				5					8	1	4		
Num	138.00	356.00				7.80					179.20	19.00	10.00		
Freq	4	4				15					1	2	1		
Num		352.00				8.20					178.30	28.00	6.00		
Freq		1				4					8	2	4		
Num		354.00									178.50	20.00	5.00		
Freq		5									1	1	1		
Num		355.00									179.40	23.00	7.00		
Freq		7									1	3	4		
Num											179.50	18.00	14.00		
Freq											1	1	1		
Num													15.00		
Freq													1		
Num													12.00		
Freq													1		
Mean															
	138.95	361.38	397.54	407.20		8.04	-0.49	0.70	0.40	0.38	178.28	22.64	9.42	0.00	0.00
Mode															
	139.00	363.00	398.00	407.00		8.10	-0.50	0.70	0.40	0.40	178.20	21.00	11.00	0.00	0.00
Max															
	141.00	369.00	398.00	408.00		8.30	-0.40	0.80	0.40	0.40	179.50	28.00	15.00	0.00	0.00
Min															
	137.00	352.00	397.00	407.00		7.70	-0.60	0.70	0.40	0.30	178.10	18.00	5.00	0.00	0.00
Maximum deviation															
	4.00	17.00	1.00	1.00		0.60	0.20	0.10	0.00	0.10	1.40	10.00	10.00	0.00	0.00

Specimen sample - PL3.TXT search.length					5	w.width	10	cal.x	27.71	cal.y	22.40	cnr_adj	1.65		
	l_s_x	r_s_x	l_s_y	r_s_y		width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence															
Num	150.00	332.00	403.00	411.00		6.60	-0.70	1.00	0.00	0.40	182.20	22.00	17.00	0.00	0.00
Freq	12	22	98	28		15	99	100	100	92	33	28	34	100	100
Num	149.00	317.00	404.00	413.00		6.10	-0.80			0.30	182.10	23.00	20.00		
Freq	47	19	1	47		16	1			8	27	24	9		
Num	148.00	324.00	402.00	414.00		6.20					182.30	13.00	19.00		
Freq	25	27	1	21		1					27	1	36		
Num	147.00	323.00		412.00		6.40					182.00	15.00	16.00		
Freq	13	20		4		14					11	3	3		
Num	151.00	333.00				6.30					182.40	24.00	14.00		
Freq	2	5				35					1	7	2		
Num	145.00	334.00				6.70					181.90	26.00	15.00		
Freq	1	4				16					1	2	3		
Num		325.00				6.00						20.00	18.00		
Freq		1				2						14	13		
Num		331.00				6.80						18.00			
Freq		1				1						3			
Num		322.00										21.00			
Freq		1										5			
Num												17.00			
Freq												5			
Num												16.00			
Freq												1			
Num												19.00			
Freq												2			
Num												25.00			
Freq												4			
Num												27.00			
Freq												1			
Mean	148.61	325.14	403.00	412.61		6.39	-0.70	1.00	0.00	0.39	182.18	21.51	17.97	0.00	0.00
Mode	149.00	324.00	403.00	413.00		6.30	-0.70	1.00	0.00	0.40	182.20	22.00	19.00	0.00	0.00
Max	151.00	334.00	404.00	414.00		6.80	-0.70	1.00	0.00	0.40	182.40	27.00	20.00	0.00	0.00
Min	145.00	317.00	402.00	411.00		6.00	-0.80	1.00	0.00	0.30	181.90	13.00	14.00	0.00	0.00
Maximum deviation	6.00	17.00	2.00	3.00		0.80	0.10	0.00	0.00	0.10	0.50	14.00	6.00	0.00	0.00

Specimen sample - PL4.TXT search.length					5		w.width	10	cal.x	27.71	cal.y	22.40	cnr_adj	1.4	
	l_s_x	r_s_x	l_s_y	r_s_y		width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence															
Num	150.00	369.00	390.00	398.00		7.90	1.00	0.50	0.00	0.30	179.10	37.00	41.00	0.00	0.00
Freq	24	32	25	41		80	89	97	99	11	15	7	28	99	99
Num	149.00	368.00	391.00	397.00		8.00	1.10	0.60		0.20	179.00	35.00	44.00		
Freq	46	61	69	58		12	10	2		87	7	23	1		
Num	148.00	370.00	392.00			7.80				0.10	178.90	36.00	40.00		
Freq	16	6	5			7				1	29	11	27		
Num	151.00										178.80	34.00	39.00		
Freq	10										11	32	35		
Num	152.00										178.70	32.00	43.00		
Freq	3										6	4	1		
Num											179.90	33.00	45.00		
Freq											1	17	1		
Num											178.60	31.00	38.00		
Freq											12	1	1		
Num											178.40	29.00	36.00		
Freq											11	2	3		
Num											179.80	30.00	37.00		
Freq											1	1	1		
Num											178.50	27.00	42.00		
Freq											6	1	1		
Mean	149.37	368.44	390.80	397.41		7.91	1.01	0.50	0.00	0.21	178.82	34.17	39.94	0.00	0.00
Mode	149.00	368.00	391.00	397.00		7.90	1.00	0.50	0.00	0.20	178.90	34.00	39.00	0.00	0.00
Max	152.00	370.00	392.00	398.00		8.00	1.10	0.60	0.00	0.30	179.90	37.00	45.00	0.00	0.00
Min	148.00	368.00	390.00	397.00		7.80	1.00	0.50	0.00	0.10	178.40	27.00	36.00	0.00	0.00
Maximum deviation	4.00	2.00	2.00	1.00		0.20	0.10	0.10	0.00	0.20	1.50	10.00	9.00	0.00	0.00

Specimen sample - PL5.TXT search.length					5		w.width	10	cal.x	27.71	cal.y	22.40	cnr_adj	1.2	
	l_s_x	r_s_x	l_s_y	r_s_y		width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence															
Num	163.00	328.00	336.00	336.00		5.90	-0.90	0.90	0.00	0.10	179.10	24.00	31.00	0.00	0.00
Freq	50	22	98	35		32	26	98	98	72	26	13	21	98	98
Num	164.00	330.00		335.00		6.00	-0.80			0.00	179.00	26.00	29.00		
Freq	48	61		63		66	72			26	28	17	2		
Num		329.00									179.20	28.00	26.00		
Freq		13									1	30	63		
Num		327.00									178.90	27.00	23.00		
Freq		2									44	30	8		
Num											178.80	25.00	33.00		
Freq											1	8	2		
Num													35.00		
Freq													1		
Num													30.00		
Freq													1		

PL5.TXT contd.

Mean	163.49	329.36	336.00	335.36	5.97	-0.83	0.90	0.00	0.07	178.98	26.57	27.16	0.00	0.00
Mode	163.00	330.00	336.00	335.00	6.00	-0.80	0.90	0.00	0.10	178.90	28.00	26.00	0.00	0.00
Max	164.00	330.00	336.00	336.00	6.00	-0.80	0.90	0.00	0.10	179.20	28.00	35.00	0.00	0.00
Min	163.00	327.00	336.00	335.00	5.90	-0.90	0.90	0.00	0.00	178.80	24.00	23.00	0.00	0.00
Maximum deviation	1.00	3.00	0.00	1.00	0.10	0.10	0.00	0.00	0.10	0.40	4.00	12.00	0.00	0.00

Specimen	sample -	PL6.TXT	search.length		5	w.width	10	cal.x	27.71	cal.y	22.40	cnr_adj	1.05	
	l_s_x	r_s_x	l_s_y	r_s_y	width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence														
Num	154.00	352.00	324.00	330.00	7.20	0.30	0.00	0.10	0.10	179.40	9.00	15.00	6.50	0.70
Freq	95	68	100	100	63	100	100	100	88	25	26	89	43	47
Num	155.00	351.00			7.10				0.00	179.20	8.00	16.00	6.40	0.00
Freq	5	32			37				12	47	74	11	4	47
Num										179.10			0.00	1.90
Freq										21			47	5
Num										179.30			5.30	1.80
Freq										5			5	1
Num										178.90			5.50	
Freq										2			1	
Mean	154.05	351.68	324.00	330.00	7.16	0.30	0.00	0.10	0.09	179.23	8.26	15.11	3.37	0.44
Mode	154.00	352.00	324.00	330.00	7.20	0.30	0.00	0.10	0.10	179.20	8.00	15.00	0.00	0.70
Max	155.00	352.00	324.00	330.00	7.20	0.30	0.00	0.10	0.10	179.40	9.00	16.00	6.50	1.90
Min	154.00	351.00	324.00	330.00	7.10	0.30	0.00	0.10	0.00	178.90	8.00	15.00	0.00	0.00
Maximum deviation	1.00	1.00	0.00	0.00	0.10	0.00	0.00	0.00	0.10	0.50	1.00	1.00	6.50	1.90

Specimen	sample -	PL7.TXT	search.length		5	w.width	10	cal.x	27.71	cal.y	22.40	cnr_adj	1.3	
	l_s_x	r_s_x	l_s_y	r_s_y	width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence														
Num	110.00	363.00	308.00	307.00	9.10	0.50	0.00	0.00	0.30	179.20	30.00	23.00	0.00	0.00
Freq	92	51	93	30	92	96	97	94	96	23	75	13	97	97
Num	111.00	362.00	309.00	308.00	9.20	0.40		0.10	0.20	179.10	28.00	20.00		
Freq	1	43	4	67	4	1		3	1	57	17	40		
Num	109.00	364.00			9.00					179.30	29.00	17.00		
Freq	4	2			1					3	3	31		
Num		359.00								178.90	26.00	22.00		
Freq		1								3	2	8		
Num										179.00		19.00		
Freq										9		3		
Num										179.40		25.00		
Freq										2		2		
Mean	109.97	362.54	308.04	307.69	9.10	0.50	0.00	0.00	0.30	179.12	29.54	19.88	0.00	0.00
Mode	110.00	363.00	308.00	308.00	9.10	0.50	0.00	0.00	0.30	179.10	30.00	20.00	0.00	0.00
Max	111.00	364.00	309.00	308.00	9.20	0.50	0.00	0.10	0.30	179.40	30.00	25.00	0.00	0.00
Min	109.00	359.00	308.00	307.00	9.00	0.40	0.00	0.00	0.20	178.90	26.00	17.00	0.00	0.00
Maximum deviation	2.00	5.00	1.00	1.00	0.20	0.10	0.00	0.10	0.10	0.50	4.00	8.00	0.00	0.00

Specimen	sample -	CEGB	search.length		5	w.width	6	cal.x	27.26	cal.y	22.40	cnr_adj	1.1	
	l_s_x	r_s_x	l_s_y	r_s_y	width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence														
Num	127.00	390.00	361.00	356.00	9.60	2.50	0.00	0.00	0.00	180.00	43.00	43.00	0.00	0.00
Freq	85	84	89	88	85	9	101	101	100	8	8	37	84	84
Num	126.00	391.00	360.00	357.00	9.70	2.60			0.10	179.70	42.00	46.00	4.50	5.10
Freq	3	4	10	11	13	92			1	7	21	24	2	6
Num	129.00	389.00	362.00	355.00	9.50					179.90	39.00	45.00	7.80	2.10
Freq	10	4	2	2	1					70	4	5	6	10
Num	125.00	392.00			9.80					179.80	40.00	48.00	4.40	6.80
Freq	1	9			2					14	24	1	4	1
Num	124.00									179.60	41.00	44.00	7.50	
Freq	1									2	44	4	4	
Num	128.00											47.00	2.80	
Freq	1											24	1	
Num												42.00		
Freq												4		
Num												40.00		
Freq												1		
Num												41.00		
Freq												1		
Mean	127.13	390.18	360.92	356.09	9.62	2.59	0.00	0.00	0.00	179.87	41.05	44.78	1.04	0.58
Mode	127.00	390.00	361.00	356.00	9.60	2.60	0.00	0.00	0.00	179.90	41.00	43.00	0.00	0.00
Max	129.00	392.00	362.00	357.00	9.80	2.60	0.00	0.00	0.10	180.00	43.00	48.00	7.80	6.80
Min	124.00	389.00	360.00	355.00	9.50	2.50	0.00	0.00	0.00	179.60	39.00	40.00	0.00	0.00
Maximum deviation	5.00	3.00	2.00	2.00	0.30	0.10	0.00	0.00	0.10	0.40	4.00	8.00	7.80	6.80

Specimen sample	- VICK		search.length		5		w.width	10	cal.x	11.55	cal.y	10.60	cnr_adj	1.4	
	l_s_x	r_s_x	l_s_y	r_s_y		width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence															
Num	130.00	380.00	357.00	360.00		21.60	3.80	0.00	0.00	0.20	179.70	37.00	39.00	0.00	0.00
Freq	53	47	93	53		73	73	100	100	59	1	11	21	100	100
Num	131.00	379.00	358.00	359.00		21.50	3.70			0.10	179.50	40.00	36.00		
Freq	42	38	2	47		15	23			13	28	6	14		
Num	132.00	381.00	356.00			21.70	3.90			0.30	179.60	42.00	38.00		
Freq	1	14	5			9	4			28	12	3	15		
Num	129.00	378.00				21.40					179.40	34.00	32.00		
Freq	2	1				1					39	19	4		
Num	128.00					21.90					179.30	38.00	34.00		
Freq	1					1					20	10	13		
Num	127.00					21.80						35.00	30.00		
Freq	1					1						8	1		
Num												33.00	37.00		
Freq												3	15		
Num												43.00	41.00		
Freq												1	5		
Num												36.00	33.00		
Freq												5	2		
Num												28.00	44.00		
Freq												1	1		
Num												39.00	43.00		
Freq												2	2		
Num												32.00	40.00		
Freq												20	3		
Num												31.00	35.00		
Freq												4	4		
Num												30.00			
Freq												7			
Mean	130.37	379.74	356.97	359.53		21.60	3.78	0.00	0.00	0.22	179.44	34.81	37.09	0.00	0.00
Mode	130.00	380.00	357.00	360.00		21.60	3.80	0.00	0.00	0.20	179.40	32.00	39.00	0.00	0.00
Max	132.00	381.00	358.00	360.00		21.90	3.90	0.00	0.00	0.30	179.70	43.00	44.00	0.00	0.00
Min	127.00	378.00	356.00	359.00		21.40	3.70	0.00	0.00	0.10	179.30	28.00	30.00	0.00	0.00
Maximum deviation	5.00	3.00	2.00	1.00		0.50	0.20	0.00	0.00	0.20	0.40	15.00	14.00	0.00	0.00

Specimen sample	- MULT		search.length		5		w.width	3	cal.x	11.55	cal.y	10.60	cnr_adj	1.0	
	l_s_x	r_s_x	l_s_y	r_s_y		width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence															
Num	101.00	391.00	375.00	374.00		25.10	2.50	1.50	0.00	0.50	176.30	65.00	47.00	8.60	16.50
Freq	100	52	100	86		52	100	100	100	100	96	100	96	96	48
Num		392.00		375.00		25.20					176.20		65.00	8.20	16.60
Freq		34		10		34					4		4	4	34
Num		395.00		376.00		25.50									16.90
Freq		14		4		14									18
Mean	101.00	391.90	375.00	374.18		25.19	2.50	1.50	0.00	0.50	176.30	65.00	47.72	8.58	16.61
Mode	101.00	391.00	375.00	374.00		25.10	2.50	1.50	0.00	0.50	176.30	65.00	47.00	8.60	16.50
Max	101.00	395.00	375.00	376.00		25.50	2.50	1.50	0.00	0.50	176.30	65.00	65.00	8.60	16.90
Min	101.00	391.00	375.00	374.00		25.10	2.50	1.50	0.00	0.50	176.20	65.00	47.00	8.20	16.50
Maximum deviation	0.00	4.00	0.00	2.00		0.40	0.00	0.00	0.00	0.00	0.10	0.00	18.00	0.40	0.40

Specimen sample	- FIL1.TXT		search.length		5		w.width	10	cal.x	27.71	cal.y	22.40	cnr_adj	1.5	
	l_s_x	r_s_x	l_s_y	r_s_y		width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence															
Num	211.00	365.00	332.00	346.00		5.60	1.60	0.00	0.00	6.10	128.70	26.00	10.00	3.30	3.10
Freq	22	10	50	67		13	100	99	101	11	10	5	11	26	31
Num	208.00	364.00	331.00	347.00		5.70	1.70	0.10		5.90	128.80	20.00	14.00	3.20	3.00
Freq	7	16	26	34		10	1	2		28	17	5	7	29	17
Num	210.00	366.00	330.00			5.40				6.20	127.00	17.00	9.00	3.40	2.90
Freq	10	8	23			14				13	2	6	7	24	5
Num	204.00	361.00	329.00			5.50				5.60	128.90	28.00	11.00	3.50	3.20
Freq	20	7	2			14				3	7	2	2	20	30
Num	207.00	362.00				5.80				5.70	128.40	16.00	16.00	3.60	3.30
Freq	5	9				22				8	7	8	5	2	18
Num	213.00	370.00				6.00				5.80	128.00	25.00	23.00		
Freq	10	8				19				15	5	8	1		
Num	212.00	369.00				5.90				6.00	128.20	22.00	17.00		
Freq	6	16				7				10	17	2	4		
Num	214.00	368.00				6.10				6.30	128.30	15.00	6.00		
Freq	2	12				2				15	16	7	19		
Num	200.00	363.00									128.10	19.00	4.00		
Freq	1	9									6	13	6		
Num	202.00	360.00									128.50	18.00	5.00		
Freq	1	2									2	18	5		
Num	203.00	367.00									128.60	24.00	19.00		
Freq	3	3									8	6	4		
Num	206.00	371.00									125.90	11.00	8.00		
Freq	3	1									3	2	6		
Num	209.00										125.80	14.00	7.00		
Freq	1										1	2	5		
Num	205.00											23.00	2.00		
Freq	10											9	4		
Num												21.00	3.00		
Freq												5	4		

Num											27.00	12.00		
Freq											2	4		
Num											30.00	15.00		
Freq											1	3		
Num												13.00		
Freq												4		
Mean	208.23	365.60	331.23	346.34	5.72	1.60	0.00	0.00	5.99	126.45	20.06	9.42	3.34	3.14
Mode	211.00	364.00	332.00	346.00	5.80	1.60	0.00	0.00	5.90	126.80	18.00	6.00	3.20	3.10
Max	214.00	371.00	332.00	347.00	6.10	1.70	0.10	0.00	6.30	127.00	30.00	23.00	3.60	3.30
Min	200.00	360.00	329.00	346.00	5.40	1.60	0.00	0.00	5.60	125.80	11.00	2.00	3.20	2.90
Maximum deviation	14.00	11.00	3.00	1.00	0.70	0.10	0.10	0.00	0.70	1.20	19.00	21.00	0.40	0.40

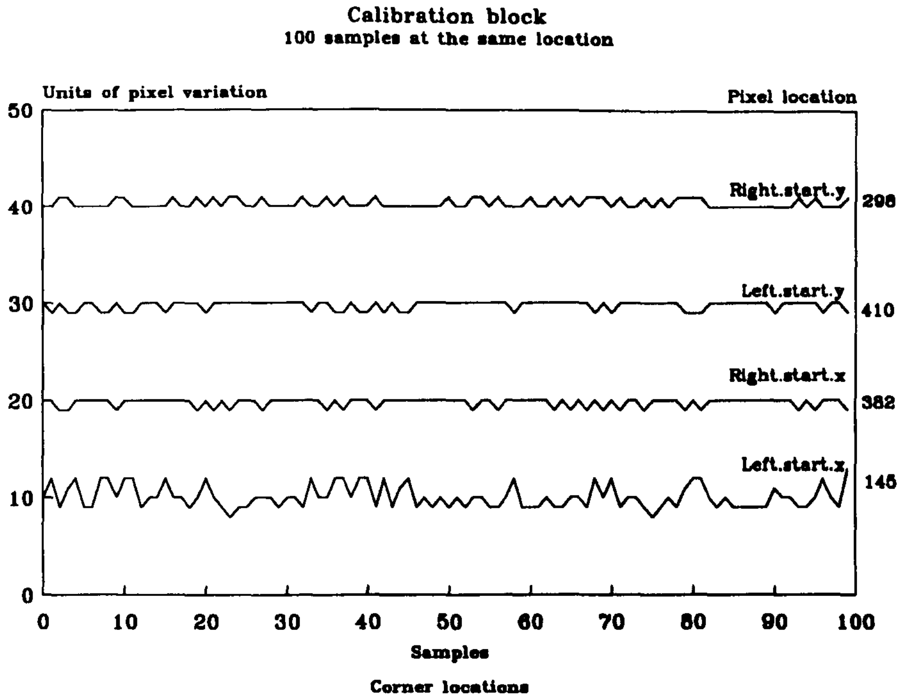
Specimen sample - FIL2.TXTsearch.length					5	w.width	10	cal.x	27.25	cal.y	22.40	cnr_adj	1.1	
	l_s_x	r_s_x	l_s_y	r_s_y	width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence														
Num	120.00	365.00	302.00	322.00	9.00	4.70	0.50	0.00	0.80	93.60	21.00	35.00	6.80	5.50
Freq	25	16	24	53	21	12	92	82	11	5	11	7	31	60
Num	131.00	364.00	312.00	323.00	8.60	4.20	0.60	0.10	1.60	93.70	31.00	33.00	6.20	5.40
Freq	6	41	1	32	13	15	8	18	36	1	1	32	5	17
Num	133.00	366.00	315.00	320.00	9.10	4.10			0.60	93.40	40.00	30.00	6.10	5.70
Freq	42	17	42	3	11	46			11	8	1	9	25	6
Num	119.00	369.00	300.00	324.00	8.70	4.50			1.50	93.50	26.00	34.00	6.90	5.60
Freq	7	3	3	8	4	3			8	3	3	11	1	14
Num	132.00	367.00	314.00	319.00	8.90	4.60			1.20	94.10	32.00	27.00	6.60	5.30
Freq	11	2	11	2	4	21			4	1	6	1	4	3
Num	124.00	363.00	305.00	321.00	8.80	4.40			1.30	93.30	33.00	32.00	6.00	
Freq	3	9	4	2	3	1			3	14	4	22	32	
Num	123.00	362.00	313.00		8.40	4.00			0.90	93.90	35.00	36.00	6.50	
Freq	1	6	5		8	1			3	3	10	1	1	
Num	125.00	370.00	301.00		8.50	5.00			1.00	93.80	38.00	31.00	7.20	
Freq	1	2	5		34	1			1	5	3	12	1	
Num	134.00	371.00	307.00		8.30				0.70	93.20	22.00	29.00		
Freq	3	2	1		1				10	45	9	3		
Num	113.00	361.00	317.00		9.30				1.80	93.10	19.00	28.00		
Freq	1	2	3		1				3	7	2	2		
Num			296.00						1.70	93.00	20.00			
Freq			1						6	8	3			
Num									1.10		18.00			
Freq									1		1			
Num									2.00		25.00			
Freq									1		2			
Num									2.10		16.00			
Freq									1		1			
Num									0.20		17.00			
Freq									1		1			
Num											39.00			
Freq											2			
Num											37.00			
Freq											6			
Num											23.00			
Freq											2			
Num											30.00			
Freq											2			
Num											34.00			
Freq											20			
Num											24.00			
Freq											3			
Num											36.00			
Freq											7			
Mean	127.92	364.70	309.88	322.34	8.72	4.32	0.51	0.02	1.25	93.30	29.75	32.27	6.33	5.50
Mode	133.00	364.00	315.00	322.00	8.50	4.10	0.50	0.00	1.60	93.20	34.00	33.00	6.00	5.50
Max	134.00	371.00	317.00	324.00	9.30	5.00	0.60	0.10	2.10	94.10	40.00	36.00	7.20	5.70
Min	113.00	361.00	296.00	319.00	8.30	4.00	0.50	0.00	0.20	93.00	16.00	27.00	6.00	5.30
Maximum deviation	21.00	10.00	21.00	5.00	1.00	1.00	0.10	0.10	1.90	1.10	24.00	9.00	1.20	0.40

Specimen	sample - FIL3.TXTsearch.length				23	w.width	2	cal.x	11.56	cal.y	10.60	cnr_adj	4.1		
	l_s_x	r_s_x	l_s_y	r_s_y		width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence															
Num	143.00	371.00	365.00	369.00		19.80	9.00	0.60	0.10	3.20	91.40	72.00	43.00	14.30	12.40
Freq	91	68	101	81		80	52	68	84	43	76	81	31	91	79
Num	142.00	370.00		370.00		19.70	8.90	0.70	0.00	3.10	91.60	71.00	42.00	14.40	12.30
Freq	10	14		12		13	27	33	10	16	2	20	22	10	14
Num		372.00		367.00		19.90	9.10		0.20	3.40	91.50		40.00	12.60	
Freq		12		7		8	22		7	7	23		23	7	
Num		373.00		368.00						3.00			45.00	12.50	
Freq		7		1						19			7	1	
Num										2.70			37.00		
Freq										5			9		
Num										2.80			47.00		
Freq										8			4		
Num										2.90			44.00		
Freq										2			2		
Num										3.30			34.00		
Freq										1			2		
Num													31.00		
Freq													1		
Mean	142.90	371.12	365.00	368.97		19.80	9.00	0.63	0.10	3.10	91.43	71.80	41.58	14.31	12.40
Mode	143.00	371.00	365.00	369.00		19.80	9.00	0.60	0.10	3.20	91.40	72.00	43.00	14.30	12.40
Max	143.00	373.00	365.00	370.00		19.90	9.10	0.70	0.20	3.40	91.60	72.00	47.00	14.40	12.60
Min	142.00	370.00	365.00	367.00		19.70	8.90	0.60	0.00	2.70	91.40	71.00	31.00	14.30	12.30
Maximum deviation	1.00	3.00	0.00	3.00		0.20	0.20	0.10	0.20	0.70	0.20	1.00	16.00	0.10	0.30

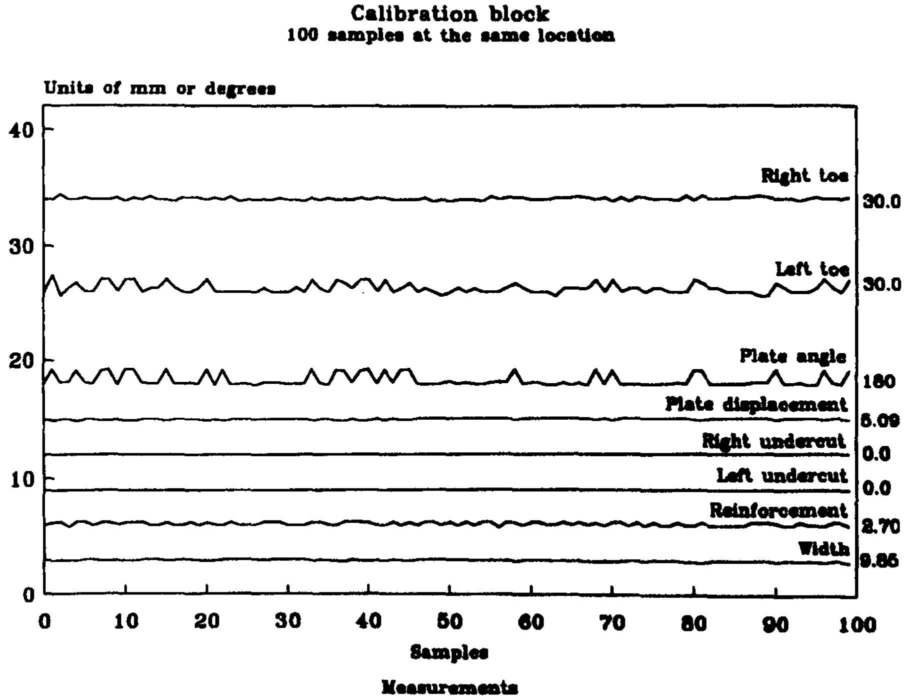
Specimen	sample - sold.txt		search.length		9		w.width	10	cal.x	36.91	cal.y	27.60	cnr_adj	1.36	
	l_s_x	r_s_x	l_s_y	r_s_y		width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence															
Num	213.00	297.00	372.00	378.00	1.40	2.30	1.00	0.00	0.40	0.20	180.70	60.00	64.00	0.30	2.10
Freq	34	38	9	11	100	87	99	69	37	26	7	30	18	1	2
Num	214.00	298.00	370.00	379.00		1.80	1.10	0.30	0.00	0.30	180.80	62.00	63.00	0.20	0.00
Freq	44	1	63	1		1	1	7	1	16	3	1	2	1	86
Num	234.00	296.00	371.00	377.00		2.20		0.10	0.20	0.10	180.90	61.00	65.00	0.00	0.90
Freq	1	60	20	88		7		16	38	55	2	1	68	86	3
Num	216.00	295.00	369.00			2.00		0.40	0.10	0.40	181.00	58.00	66.00	1.50	0.10
Freq	2	1	2			1		3	8	2	6	1	12	3	2
Num	226.00		368.00			2.40		0.20	0.30	0.80	181.20	59.00		2.30	0.30
Freq	1		5			4		5	16	1	2	65		2	4
Num	212.00		366.00								181.40	57.00		2.00	1.90
Freq	7		1								31	1		4	1
Num	220.00										181.50	56.00		0.50	1.70
Freq	4										2	1		1	1
Num	215.00										181.60			0.60	2.30
Freq	5										9			1	1
Num	219.00										181.70			1.90	0.70
Freq	2										4			1	1
Num											181.90				
Freq											9				
Num											182.00				
Freq											1				
Num											181.10				
Freq											5				
Num											181.30				
Freq											3				
Num											182.10				
Freq											1				
Num											180.60				
Freq											10				
Num											180.40				
Freq											2				
Num											180.50				
Freq											3				
Mean	214.27	296.39	370.22	377.13		2.29	1.00	0.06	0.28	0.17	181.24	58.29	64.90	0.20	0.15
Mode	214.00	296.00	370.00	377.00		2.30	1.00	0.00	0.20	0.10	181.40	59.00	65.00	0.00	0.00
Max	234.00	298.00	372.00	379.00		2.40	1.10	0.40	0.40	0.80	182.10	62.00	66.00	2.30	2.30
Min	212.00	295.00	366.00	377.00		1.80	1.00	0.00	0.00	0.10	180.40	56.00	63.00	0.00	0.00
Maximum deviation	22.00	3.00	6.00	2.00		0.60	0.10	0.40	0.40	0.70	1.70	6.00	3.00	2.30	2.30

4.10.3 Graphs from test samples

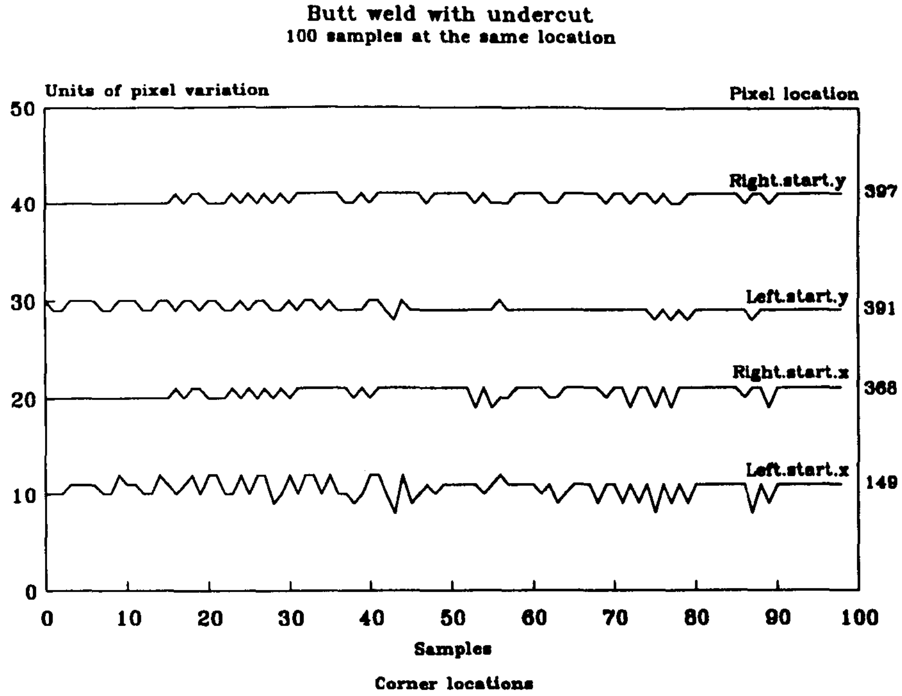
The following graphs show the variation of the weld start locations and measurements as they were taken for each sample. Finally graphs 4.15 to 4.18 show the distributions of weld, start point x and y values for all butt welds and all fillet welds.



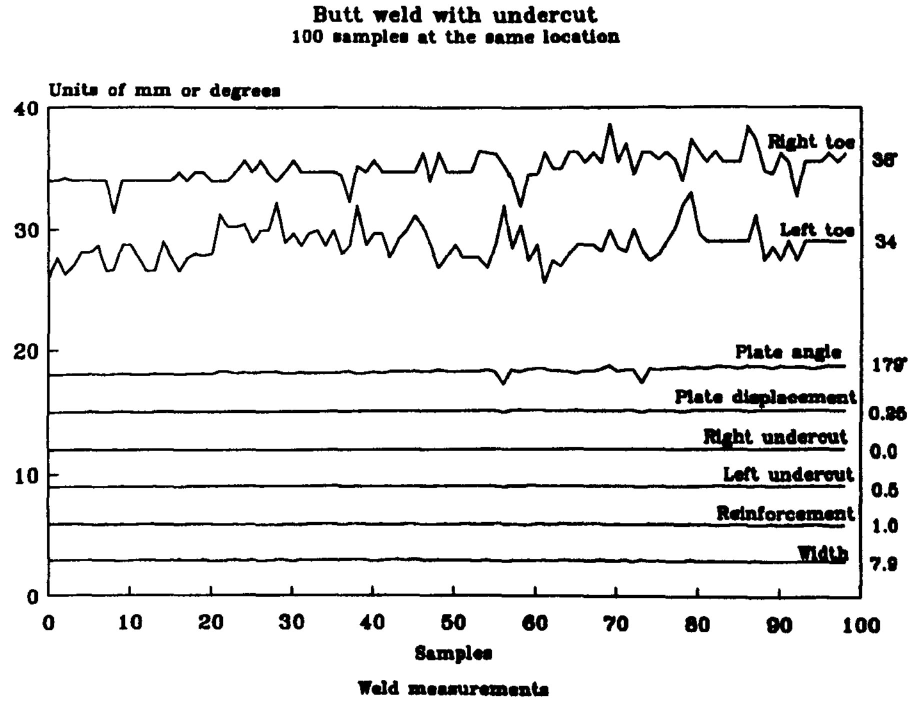
Graph 4.1 Pixel variation over 100 samples at the same location on calibration block



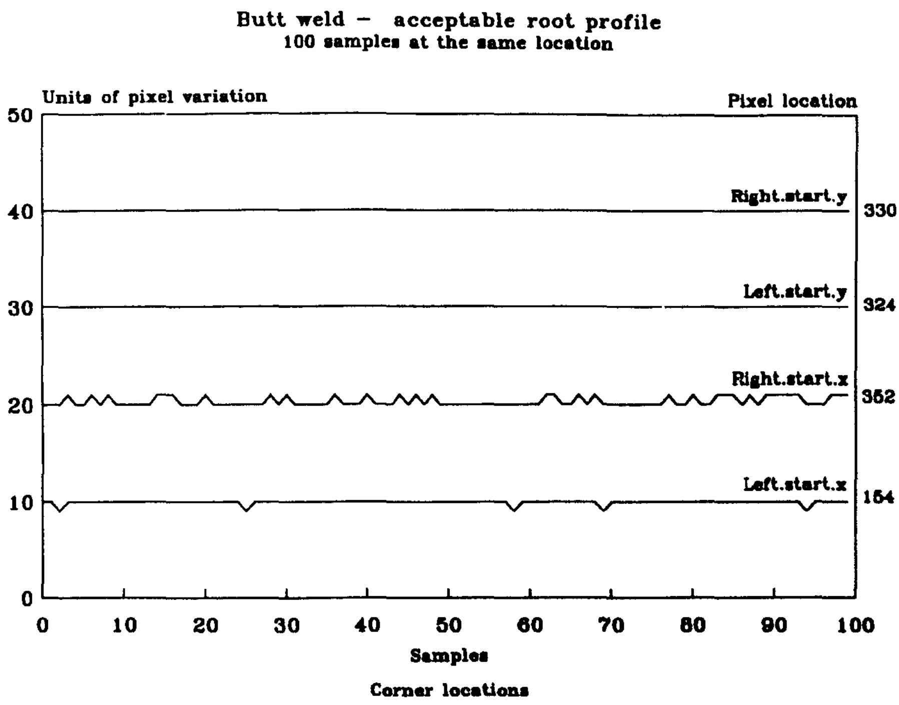
Graph 4.2 Measurement variation over 100 samples at the same location on calibration block



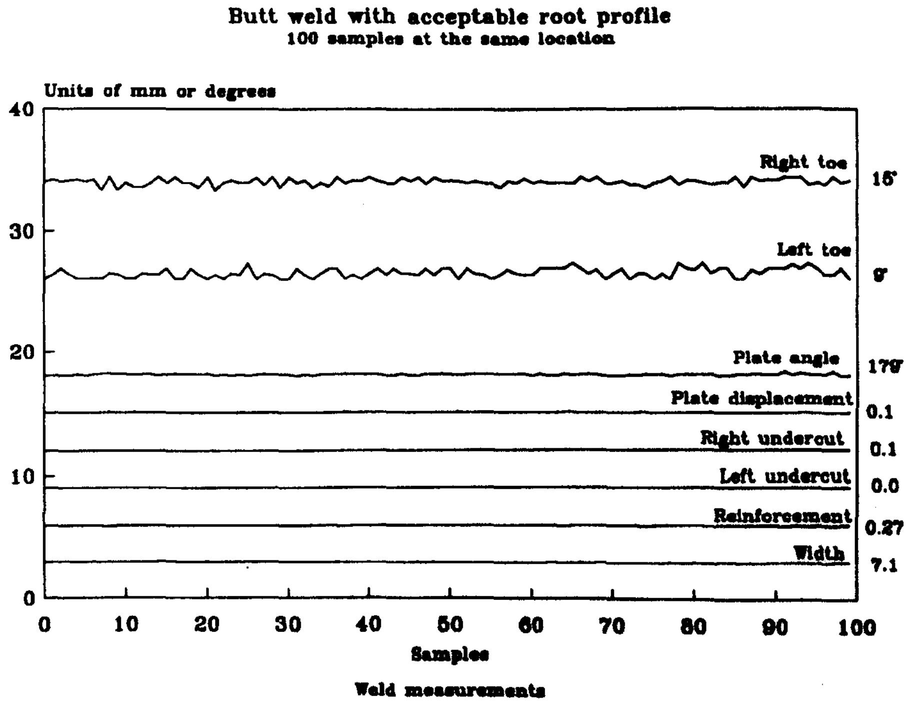
Graph 4.3 Pixel variation over 100 samples at the same location on sample butt weld with undercut



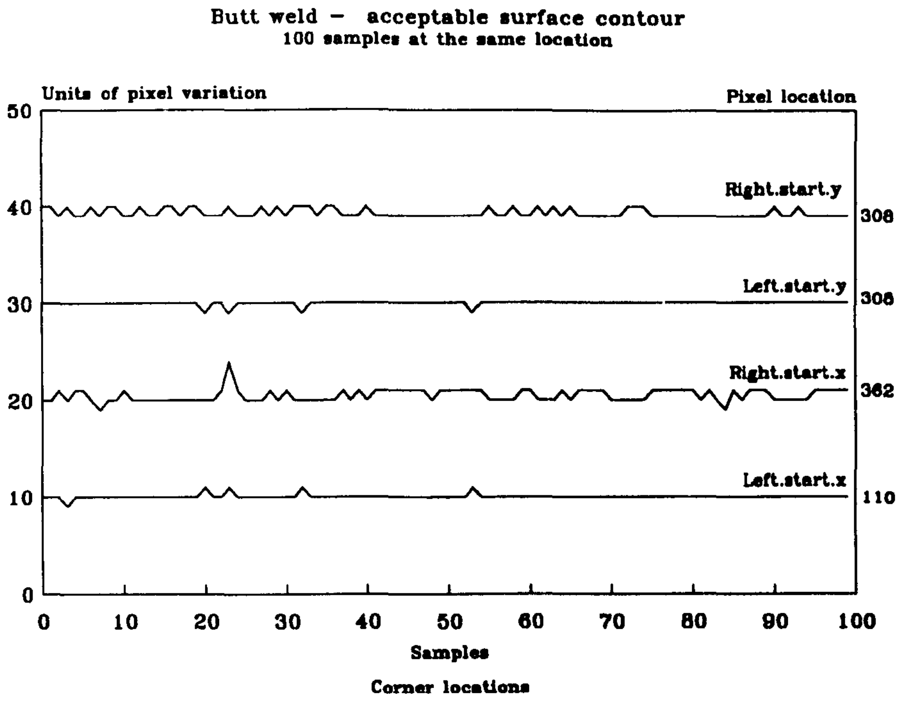
Graph 4.4 Measurement variation over 100 samples at the same location on sample butt weld with undercut



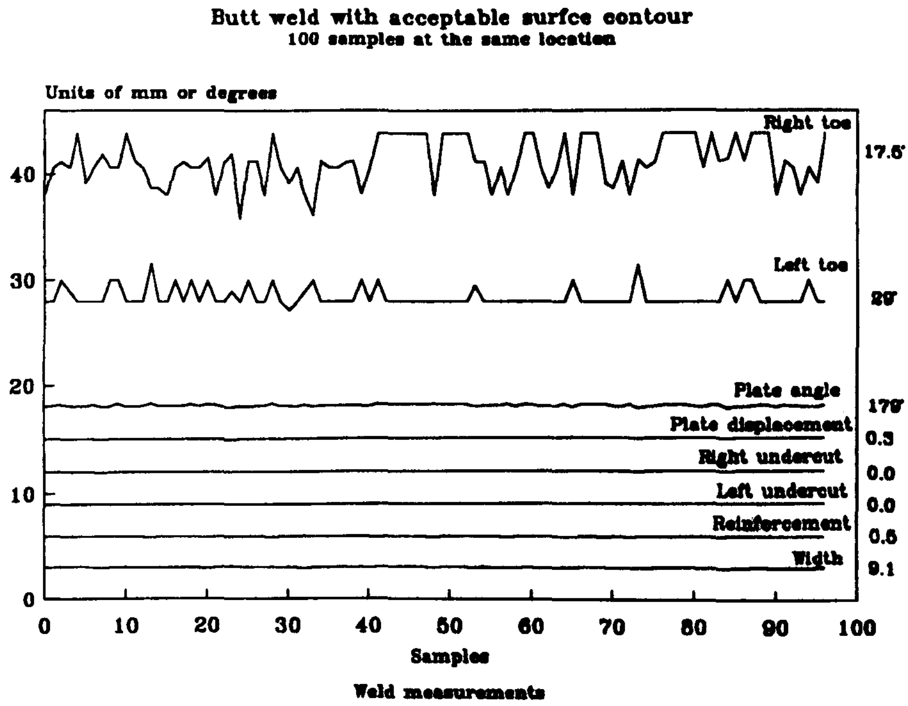
Graph 4.5 Pixel variation over 100 samples at the same location on sample butt weld with acceptable root profile



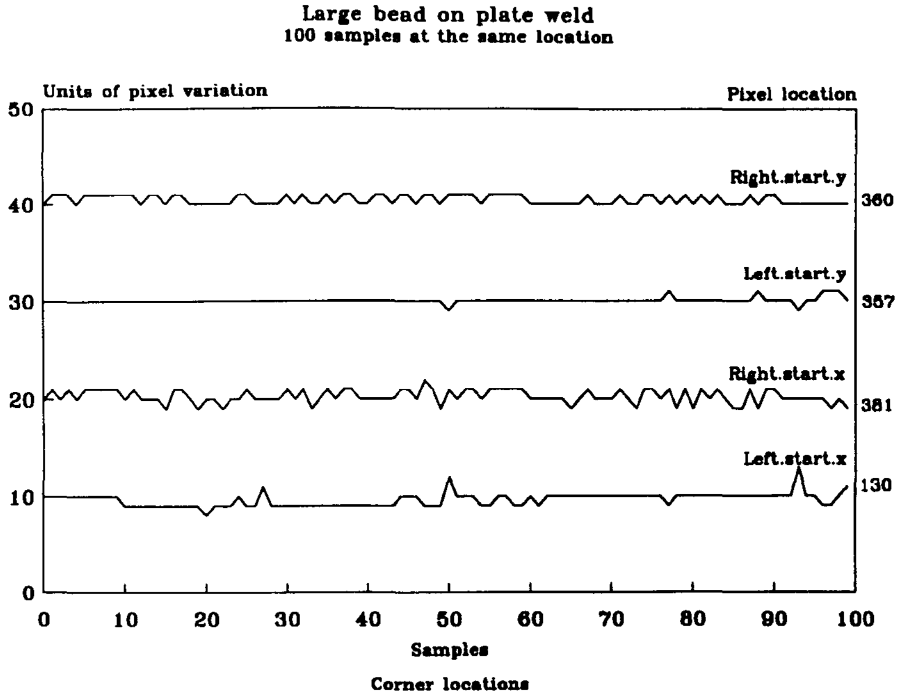
Graph 4.6 Measurement variation over 100 samples at the same location on sample butt weld with acceptable root profile



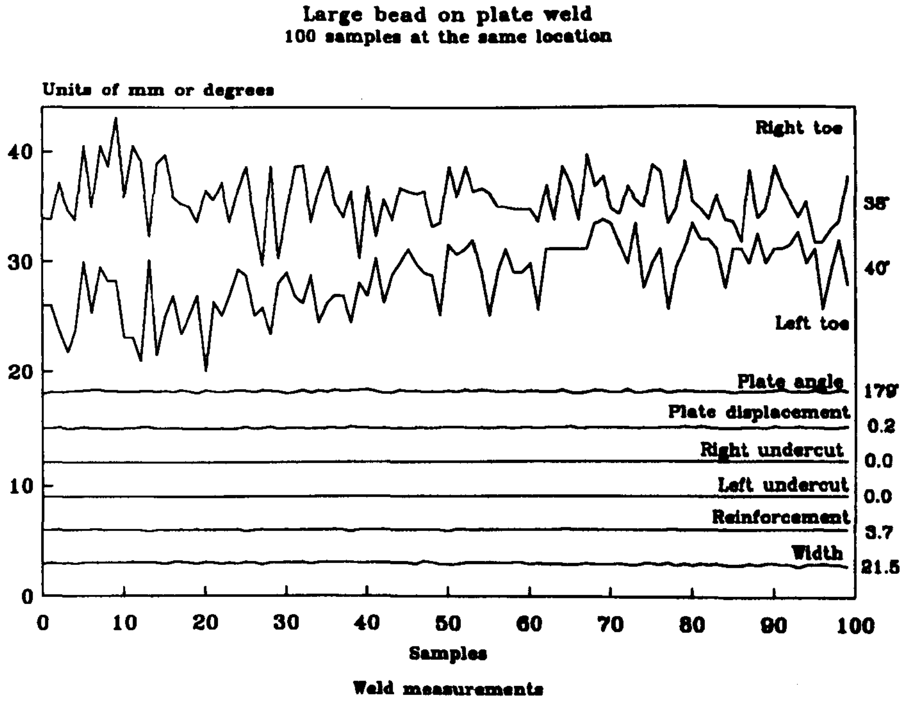
Graph 4.7 Pixel variation over 100 samples at the same location on sample butt weld with acceptable surface profile



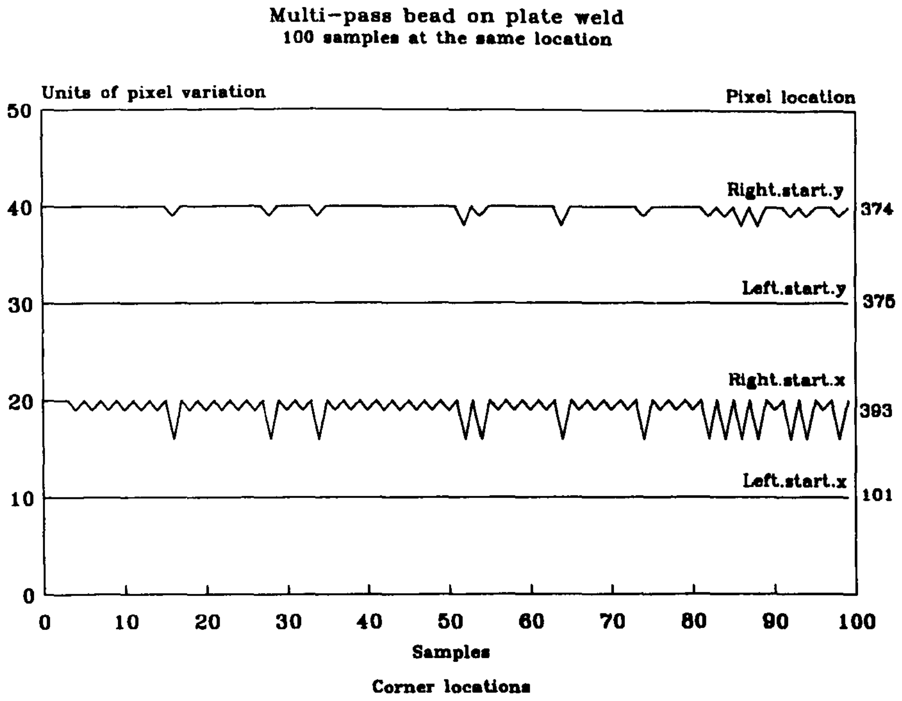
Graph 4.8 Measurement variation over 100 samples at the same location on sample butt weld with acceptable surface profile



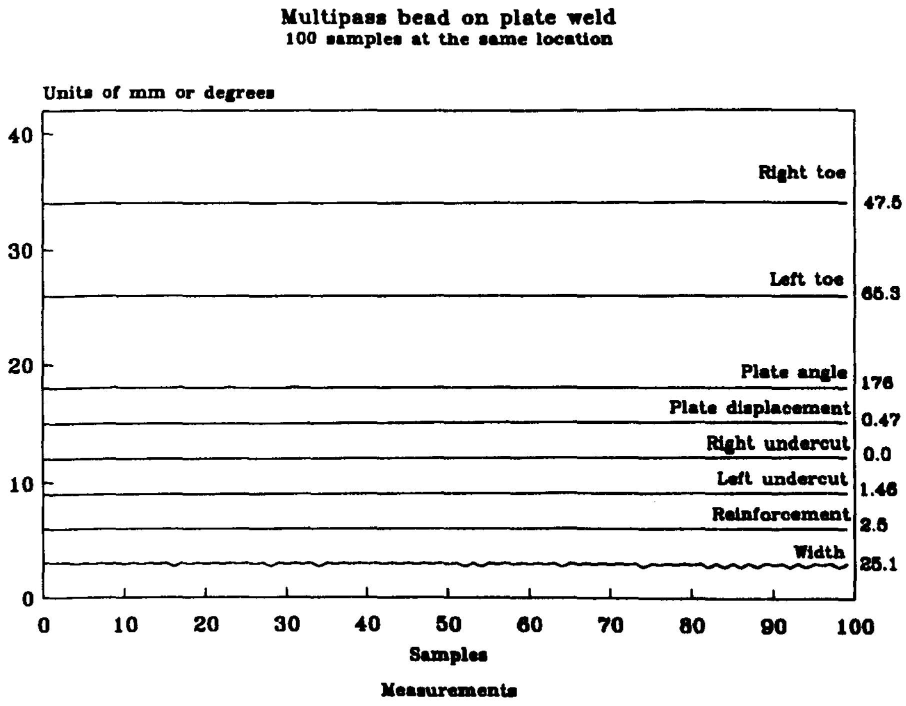
Graph 4.9 Pixel variation over 100 samples at the same location on large single pass, submerged arc butt weld



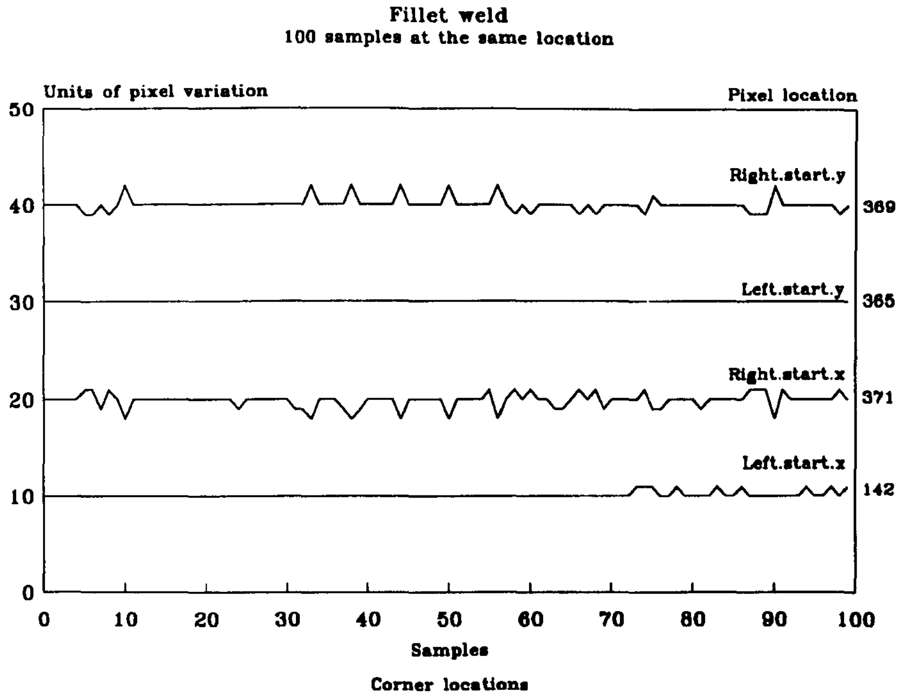
Graph 4.10 Measurement variation over 100 samples at the same location on large single pass, submerged arc butt weld



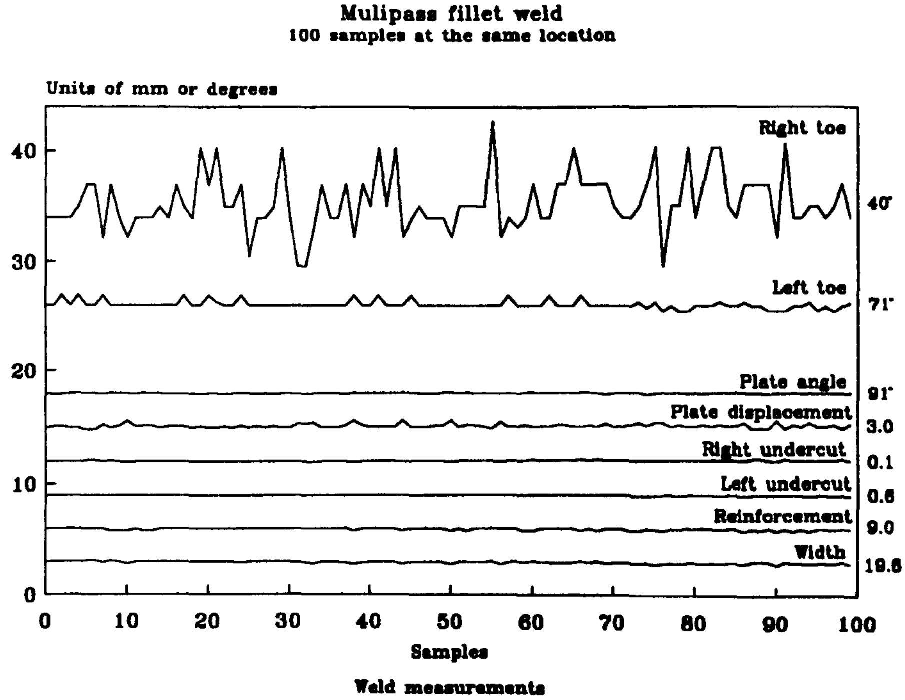
Graph 4.11 Pixel variation over 100 samples at the same location on manual metal arc multipass butt weld



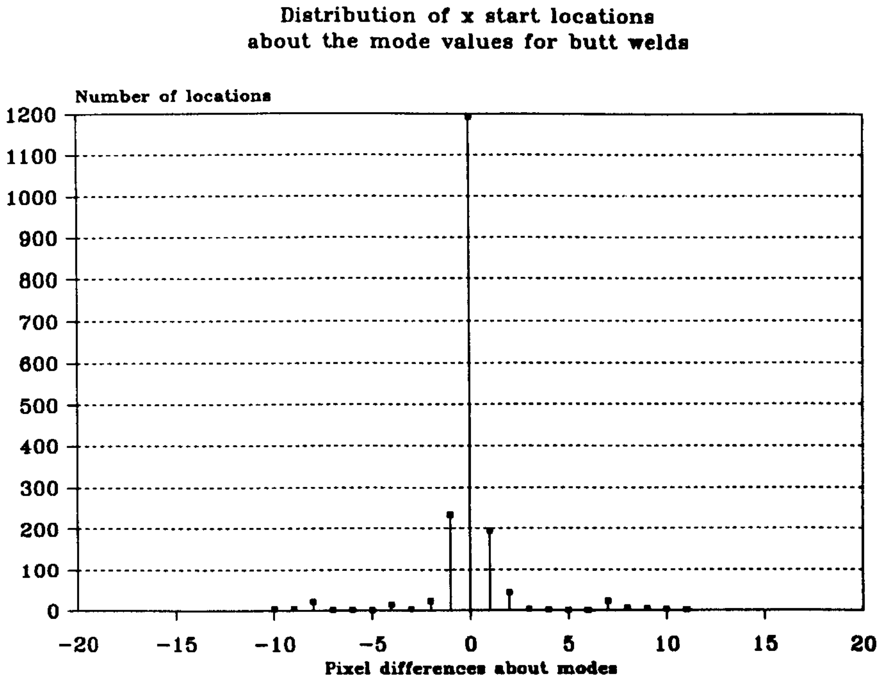
Graph 4.12 Measurement variation over 100 samples at the same location on manual metal arc multipass butt weld



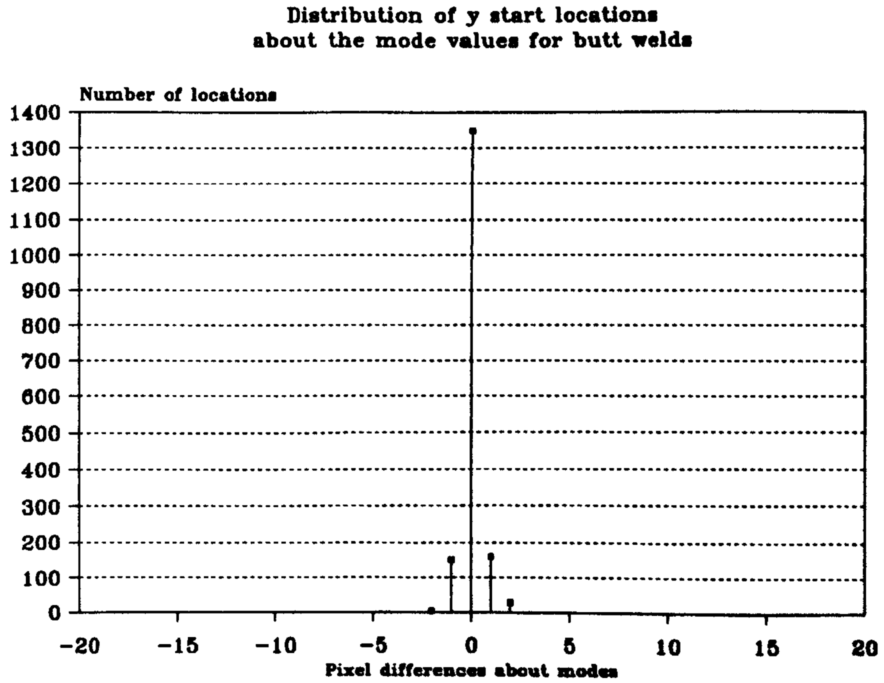
Graph 4.13 Pixel variation over 100 samples at the same location on multipass fillet weld with undercut



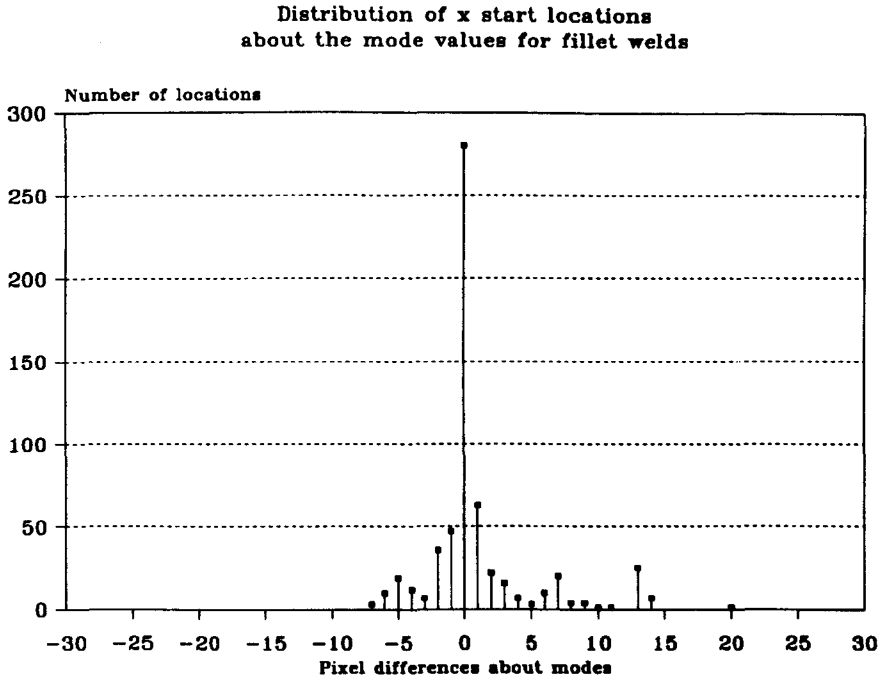
Graph 4.14 Measurement variation over 100 samples at the same location on multipass fillet weld with undercut



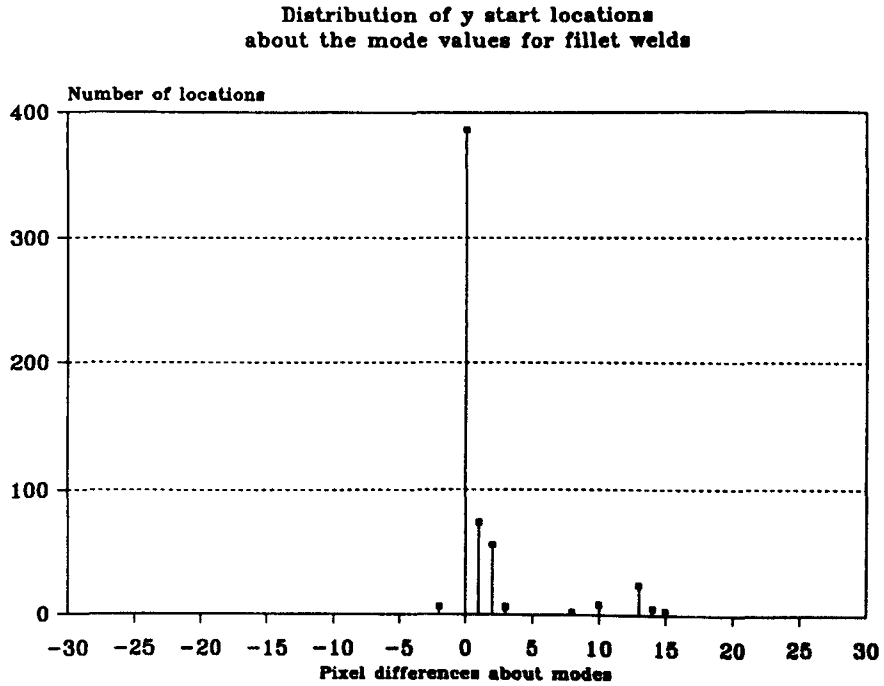
Graph 4.15 Combined distribution of x corner locations about the mode values for all the butt weld samples



Graph 4.16 Combined distribution of y corner locations about the mode values for all the butt weld samples



Graph 4.17 Combined distribution of x corner locations about the mode values for all the fillet weld samples



Graph 4.18 Combined distribution of y corner locations about the mode values for all the fillet weld samples

4.10.4 Results from on-line trials

The results from the trials of on-line weld measurement of lap welded joints follows the same pattern as those for the gauge, with images followed by their corresponding data, of start point locations and measurements, and graphs.

Line definitions

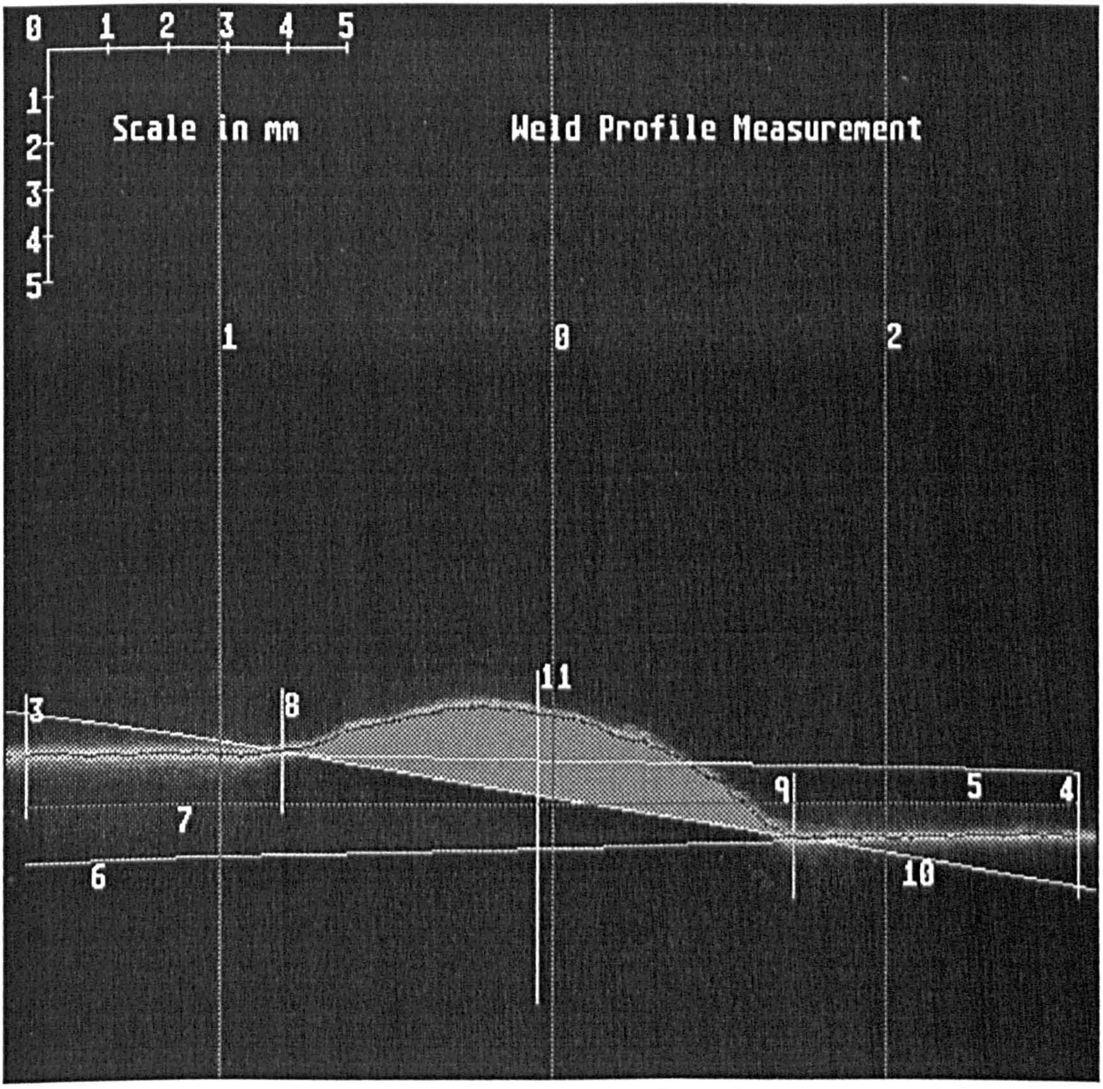
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	5	pixels
Window width	10	pixels
Calibration.factor.x	27.71	pixels/mm
Calibration.factor.y	22.4	pixels/mm
Corner.adjust	1.3	mm

Corner locations

Left.start.x	129	pixels
Right.start.x	369	pixels
Left.start.y	355	pixels
Right.start.y	396	pixels



Manual measurements

Width		mm
Reinforcement		mm
Left undercut		mm
Right undercut		mm
Plate displacement		mm
Plate angle		°
Left toe angle		°
Right toe angle		°
Left leg		mm
Right leg		mm

Gauge measurements

Width	8.85	mm
Reinforcement	2.1	mm
Left undercut	0.0	mm
Right undercut	0.0	mm
Plate displacement	1.88	mm
Plate angle	176.81	°
Left toe angle	10.79	°
Right toe angle	36.27	°
Left leg	0.0	mm
Right leg	0.0	mm

Image 4.22 Measured example of lap weld inspected 'on-line' - automatic M.I.G. welded

Line definitions

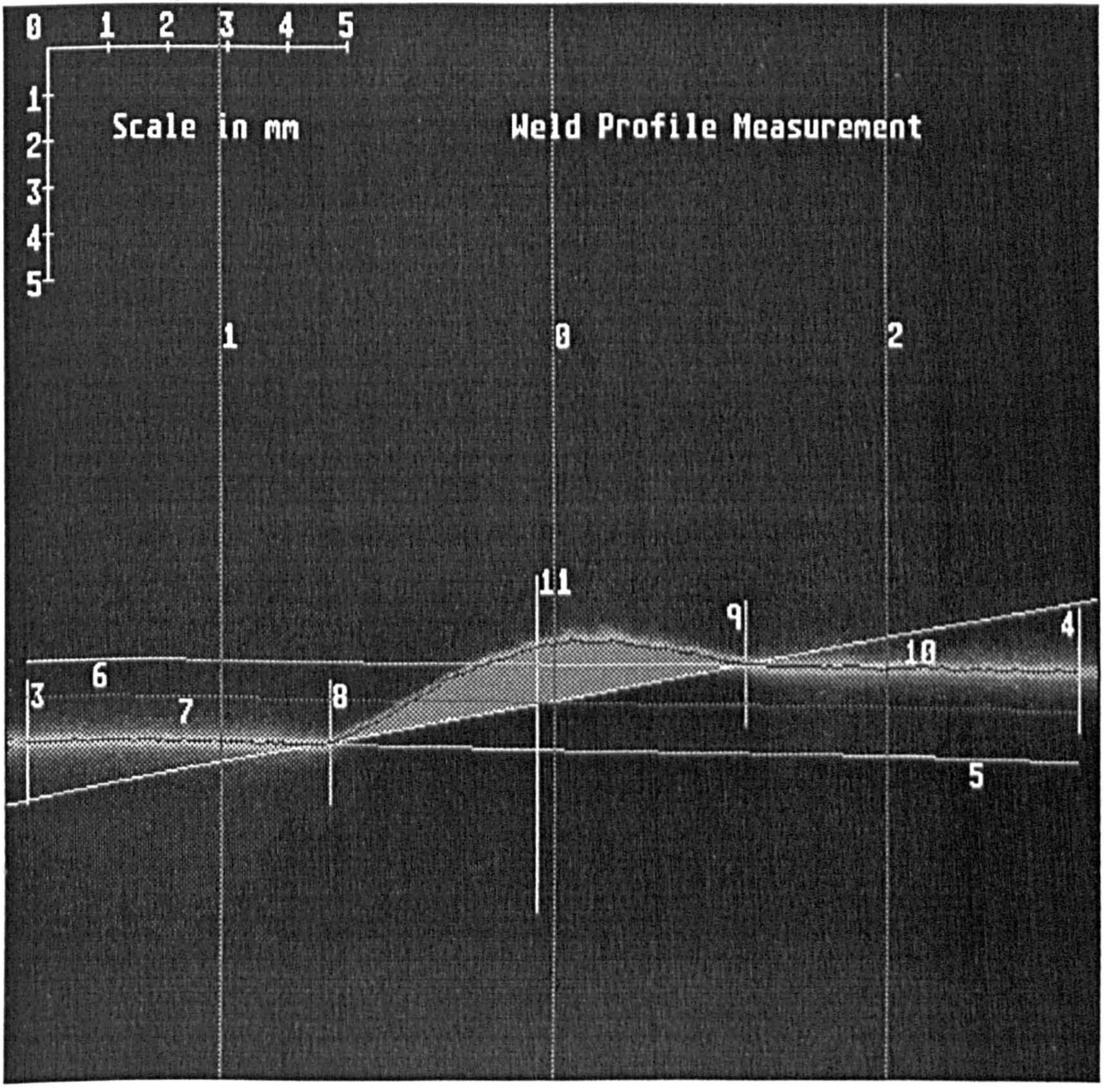
0	Image centre	1	Left least line	2	Right least line	3	Left line start
4	Right start line	5	Left parent line	6	Right parent line	7	Weld base line
8	Left start point	9	Right start point	10	Width line	11	Plate offset line

Adjustment values

Search length	5	pixels
Window width	10	pixels
Calibration.factor.x	27.71	pixels/mm
Calibration.factor.y	22.40	pixels/mm
Corner.adjust	1.0	mm

Corner locations

Left.start.x	151	pixels
Right.start.x	346	pixels
Left.start.y	351	pixels
Right.start.y	314	pixels

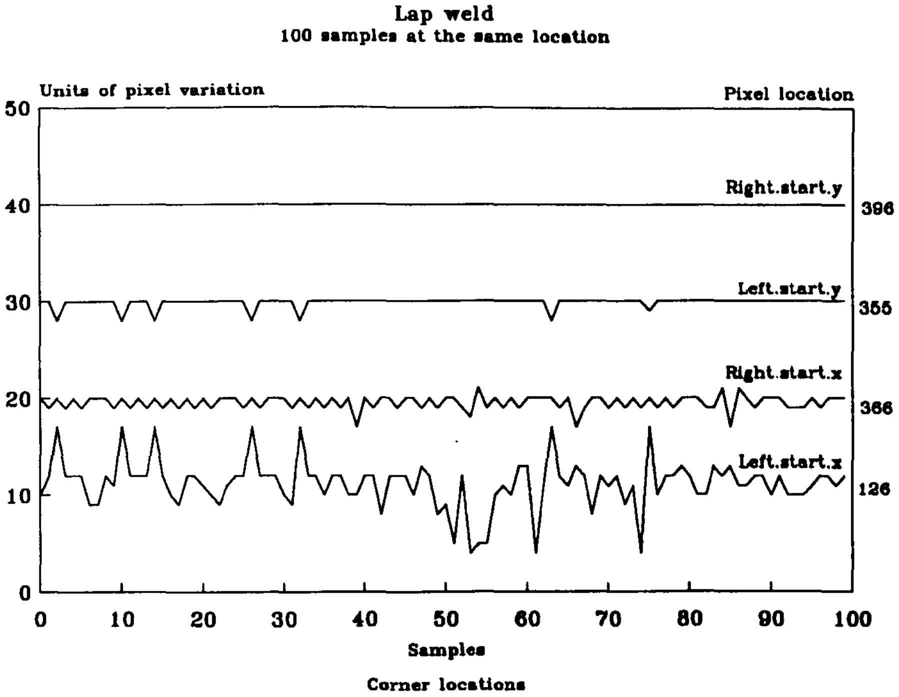


Manual measurements			**Gauge measurements**		
Width		mm	Width	7.23	mm
Reinforcement		mm	Reinforcement	1.38	mm
Left undercut		mm	Left undercut	0.0	mm
Right undercut		mm	Right undercut	0.0	mm
Plate displacement		mm	Plate displacement	1.79	mm
Plate angle		°	Plate angle	179.22.	°
Left toe angle		°	Left toe angle	27.01	°
Right toe angle		°	Right toe angle	10.72	°
Left leg		mm	Left leg	0.0	mm
Right leg		mm	Right leg	0.0	mm

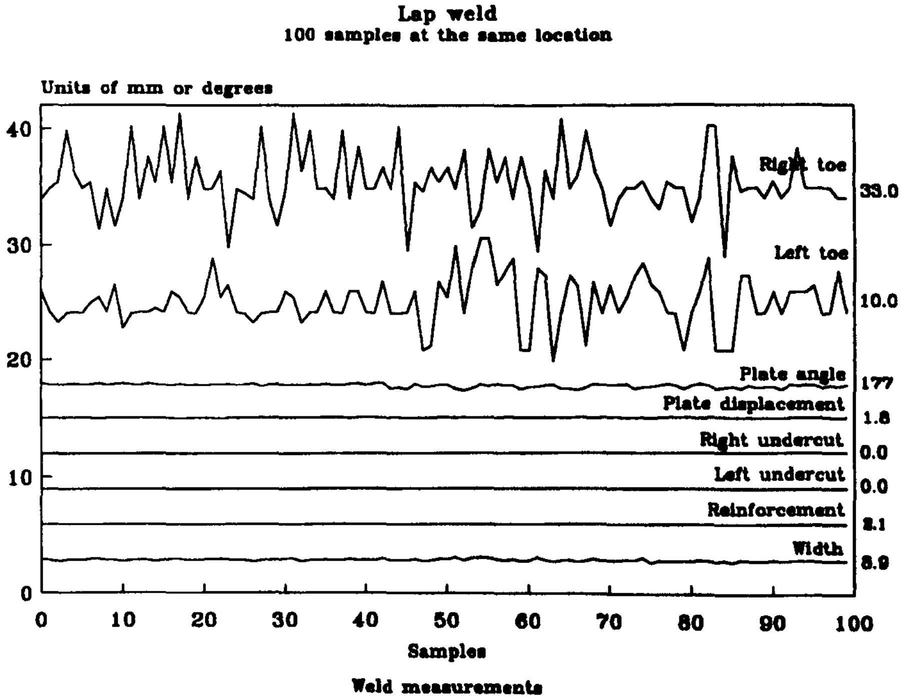
Image 4.23 Measured sample of lap weld inspected 'on-line' automatic M.I.G. welded

Specimen sample - BCK1			search.length	5		w.width		cal.x	27.71	cal.y	22.40	cnr_adj	1.3	
	l_s_x	r_s_x	l_s_y	r_s_y	width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence														
Num	126.00	366.00	355.00	396.00	8.90	2.10	0.00	0.00	1.80	176.70	9.00	35.00	0.00	0.00
Freq	17	55	93	100	53	70	100	100	95	23	20	19	100	100
Num	124.00	367.00	357.00		9.00	2.00			1.90	176.90	11.00	34.00		
Freq	41	38	6		25	30			3	40	41	29		
Num	119.00	369.00	356.00		9.10				1.70	176.80	10.00	33.00		
Freq	7	3	1		9				2	15	2	11		
Num	127.00	368.00			8.80					177.20	8.00	29.00		
Freq	7	1			7					12	10	4		
Num	125.00	365.00			8.70					177.10	12.00	32.00		
Freq	12	3			4					3	1	8		
Num	128.00				8.60					177.00	6.00	37.00		
Freq	3				2					5	5	5		
Num	123.00									177.40	14.00	28.00		
Freq	7									1	9	8		
Num	131.00									177.30	5.00	31.00		
Freq	3									1	1	5		
Num	132.00										3.00	27.00		
Freq	3										2	3		
Num											7.00	41.00		
Freq											7	1		
Num											17.00	30.00		
Freq											1	2		
Num											2.00	42.00		
Freq											1	1		
Num												39.00		
Freq												1		
Num												40.00		
Freq												1		
Num												43.00		
Freq												1		
Num												44.00		
Freq												1		
Mean	124.82	366.46	355.13	396.00	8.92	2.07	0.00	0.00	1.80	176.90	9.78	33.40	0.00	0.00
Mode	124.00	366.00	355.00	396.00	8.90	2.10	0.00	0.00	1.80	176.90	11.00	34.00	0.00	0.00
Max	132.00	369.00	357.00	396.00	9.10	2.10	0.00	0.00	1.90	177.40	17.00	44.00	0.00	0.00
Min	119.00	365.00	355.00	396.00	8.60	2.00	0.00	0.00	1.70	176.70	2.00	27.00	0.00	0.00
Maximum deviation	13.00	4.00	2.00	0.00	0.50	0.10	0.00	0.00	0.20	0.70	15.00	17.00	0.00	0.00

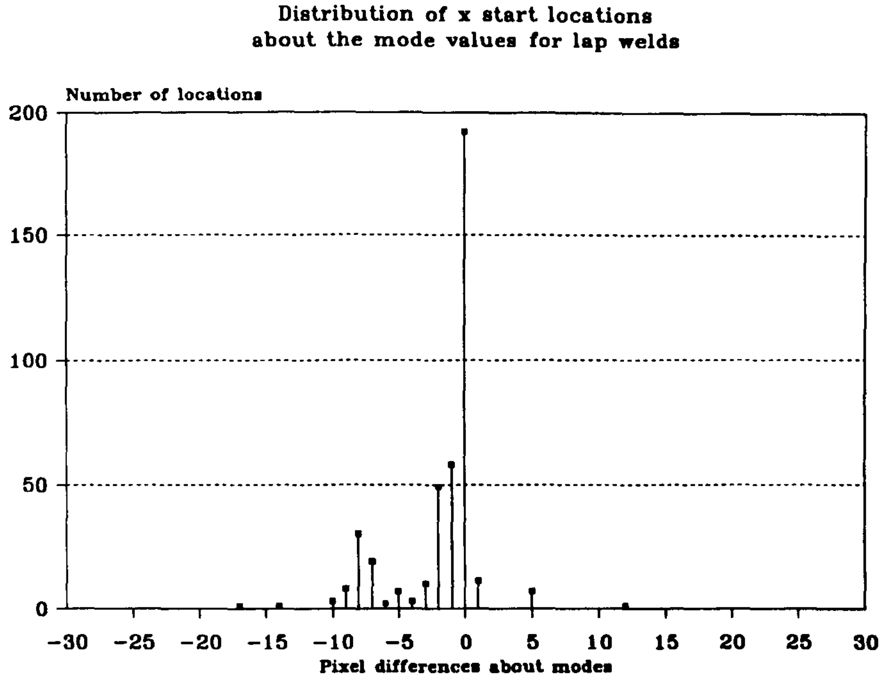
Specimen sample - BCK2			search.length	5		w.width	10	cal.x	27.71	cal.y	22.40	cnr_adj	1.6	
	l_s_x	r_s_x	l_s_y	r_s_y	width	reinf	l_uc	r_uc	disp	angle	l_toe	r_toe	l_leg	r_leg
Frequency of occurrence														
Num	153.00	345.00	351.00	314.00	7.10	1.50	0.00	0.00	1.80	179.30	31.00	8.00	0.00	0.00
Freq	30	27	101	57	19	37	101	101	97	34	4	27	101	101
Num	151.00	344.00		313.00	7.20	1.30			1.90	179.40	27.00	11.00		
Freq	67	16		42	39	20			4	26	62	15		
Num	152.00	337.00		315.00	6.80	1.40				179.20	28.00	14.00		
Freq	4	29		1	10	44				29	23	22		
Num		342.00		310.00	7.00					179.50	29.00	12.00		
Freq		7		1	8					7	8	5		
Num		338.00			6.90					179.10	30.00	22.00		
Freq		4			20					5	4	16		
Num		346.00			7.30							16.00		
Freq		8			3							3		
Num		343.00			7.40							9.00		
Freq		2			1							2		
Num		347.00			6.50							7.00		
Freq		3			1							2		
Num		336.00										4.00		
Freq		1										7		
Num		354.00										10.00		
Freq		1										1		
Num		339.00										17.00		
Freq		1										1		
Num		351.00												
Freq		1												
Num		325.00												
Freq		1												
Mean	151.63	341.96	351.00	313.55	7.06	1.42	0.00	0.00	1.80	179.30	27.86	12.24	0.00	0.00
Mode	151.00	337.00	351.00	314.00	7.20	1.40	0.00	0.00	1.80	179.30	27.00	8.00	0.00	0.00
Max	153.00	354.00	351.00	315.00	7.40	1.50	0.00	0.00	1.90	179.50	31.00	22.00	0.00	0.00
Min	151.00	325.00	351.00	310.00	6.50	1.30	0.00	0.00	1.80	179.10	27.00	4.00	0.00	0.00
Maximum deviation	2.00	29.00	0.00	5.00	0.90	0.20	0.00	0.00	0.10	0.40	4.00	18.00	0.00	0.00



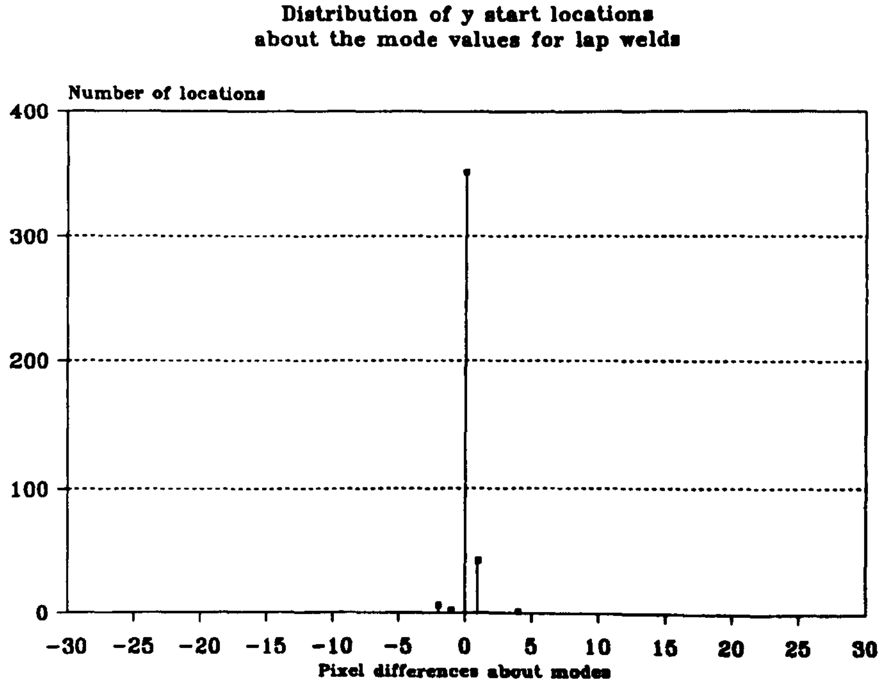
Graph 4.19 Pixel variation of weld start points over 100 samples



Graph 4.20 Pixel variation of weld start points over 100 samples



Graph 4.21 Combined distribution of y corner locations about the mode values for all the butt weld samples



Graph 4.22 Combined distribution of y corner locations about the mode values for all the butt weld samples

4.11 Costs of the gauge

The cost of the portable weld measurement gauge is broken down into the major items and given as a one off pricing. Firstly the cost of the prototype gauge which was built is given and then followed by a costing of the components for a commercial gauge. The main differences between commercial and prototype gauges are the processor, an Inmos T400 transputer replaces the T425, the display resolution is increased, and the 10 off cost for printed circuit boards is included.

4.11.1 Cost breakdown of the prototype portable gauge

No.	Item	Type	Cost	Total
1 off	Transputer	IMST425-G20S	£118.47	£118.47
4 off	Link adapter	IMSC011-P20S	£6.16	£24.64
1 off	C.L.U.T.	IMSG178J-50	£13.44	£13.44
1 off	Miniature printer	PM164	£107.25	£107.25
1 off	LCD display	LM313XBN	£167.50	£167.50
1 off	Keyboard(QWERTY)	177-679	£34.30	£34.30
1 off	Keyboard(0-9)	177-671	£14.07	£14.07
1 off	Keyboard encoder	KR9600P	£5.82	£5.82
8 off	PAL's	TIBPAL22V10ANCT	£7.50	£60.00
8 off	PAL's	PALC22V10H-25CQS	£18.30	£146.40
1 off	ADC	HA19209TP	£37.70	£37.70
1 off	Case	LC6D	£42.99	£42.99
1 off	CCD Camera	Philips MCD100-8	£329.00	£329.00
1 off	Laser	LT026MD	£38.00	£38.00
2 off	Prototyping board	041684X	£8.63	£17.26
8 off	Program Memory	TC514256AZ-10	£18.00	£144.00
1 off	Power supply	BVM60A	58.59	£58.59
8 off	Video Memory	M5M4C264L-15	£10.40	£84.80
Miscelaneous components (logic, LEDs, sockets, etc.)				£100.00
Total				£1544.23

4.11.2 Estimated component cost of a production portable gauge

No.	Item	Type	Cost	Total
1 off	Transputer	IMST400	£60.00	£60.00
1 off	Link adapter	IMSC011-P20S	£6.16	£6.16
1 off	C.L.U.T.	IMSG178J-50	£13.44	£13.44
1 off	Miniature printer	PM164	£38.40	£38.40
1 off	LCD display	LMG6250ULGR	£136.56	£136.56
1 off	Keyboard	IP-67	£50.00	£50.00
8 off	PAL's	TIBPAL22V10ANCT	£7.50	£60.00
8 off	PAL's	PALC22V10H-25CQS	£18.30	£146.40
1 off	ADC	HA19209TP	£37.70	£37.70
1 off	Case	LC6D	£42.99	£42.99
1 off	CCD Camera	PULNIX TM526	£263.00	£263.00
1 off	Laser	LT026MD	£32.00	£32.00
1 off	PCB @ 10 off price - each		£500.00	£500.00
8 off	Program Memory	TC514256AZ-10	£14.00	£112.00
1 off	Power supply	BVM60A	58.59	£58.59
8 off	Video Memory	M5M4C264L-15	£10.40	£84.80
Miscelaneous components (logic, LEDs, sockets, etc.)				£100.00
Total				£1742.04

4.12 Timings of system for on-line application

The main parts of the software have been timed to show where much of the time is spent and how the image and adjustment parameters affect the software performance.

Fold name	Laser line location	Software factors	T800-20	T425-20
find lind			0.128	0.128
process line			0.154	0.620
find end points	Line at y=10		0.022	0.022
	Line at y=500		0.030	0.030
	Blank image		0.034	0.034
find line	Line at all angles	s.l = 5 w.w = 5	0.086	0.086
	Line at all angles	s.l = 5 w.w = 10	0.101	0.101
	Line at all angles	s.l = 5 w.w = 15	0.116	0.116
	Line at all angles	s.l = 5 w.w = 20	0.131	0.131
	Line at all angles	w.w = 10 s.l = 3	0.073	0.073
	Line at all angles	w.w = 10 s.l = 5	0.101	0.101
	Line at all angles	w.w = 10 s.l = 7	0.129	0.129
	Line at all angles	w.w = 10 s.l = 10	0.184	0.184
	Line at all angles	w.w = 10 s.l = 15	0.240	0.240
	Line at all angles	w.w = 10 s.l = 20	0.323	0.323
	Blank image		0.0	0.0
least squares fit	Line across all weld types		0.002	0.002
calc.perp.dist	Line across all weld types		0.0002	0.0005
find max diff	Line across all weld types		0.001	0.23
check flat weld	Line across all weld types		0.002	0.002
calc results	Line across all weld types		0.024	0.034

All timings are in seconds

5. Chapter 5 - On-line metrology and flaw inspection

In this chapter, image processing is employed to provide a quality assurance inspection of components on a production line. The components are inductor cores, made of Ferroxcube, Mullard [67], by Philips Components Ltd, which are produced for assembly into miniature transformers for printed circuit board mounting. Plate 5.1 shows these cores, with figure 5.1 defining their dimensions and tolerances. Currently the inspection is performed manually; a number of people are employed to inspect the cores by picking them up from one bin, placing the good ones in another and discarding the sub-standard ones. This manual inspection checks the whole core qualitatively for damage or deformity and then the hole in the central pillar is checked quantitatively by comparing it with two rods, only one of which should pass through, forming a 'go / no go' test (see figure 5.2). Manual testing and inspection of 100% of cores limits the production capacity, as the number of people required to perform this level of testing becomes uneconomic. An automated test is therefore desirable.

Philips not only wanted the automatic inspection to perform the 'go-no go' test, but additionally to accurately measure other component dimensions and check for chipped edges. Chapter five firstly introduces the ferroxcube inductor core and then discusses the requirements which the inspection should achieve. The chapter then shows how the best optical arrangement was chosen to give a reliable image for processing. The operation of the software is then described to show how careful design achieves both speed and robustness. In order to prove the reliability and accuracy of the system a number of specimen cores with known defects were exhaustively tested, and the results from these tests are then discussed and presented at the end of the chapter.

5.1 The requirements of the automatic inspection

A detailed definition of the required measurements, with their corresponding tolerances, is described in this section. Prior to this, a description of the ferroxcube inductor core is given with an outline of the considerations for their manufacture.

5.1.1 The ferroxcube inductor core

Ferroxcube inductor cores are made from a ferromagnetic material which has the properties of high magnetic flux with low hysteresis loss and are suitable for pulse and wide band transformer applications for printed circuit board mounting.

5.1.1.1 Ferroxcube

Ferroxcube is a hard, black substance, chemically inert and made from a mix of two or more single ferrites, Watt [68]. These are commonly manganese with zinc, nickel with zinc or manganese with magnesium and respectively these form the basis of the three grades which are available, grades A, B and D. Ferroxcube is made by mixing and blending the component ferrites in the form of powders which are held in suspension in a slurry (or slip), for pouring into moulds, drying and firing. The inductor cores shrink by about 18% during the firing process.

5.1.1.2 Inductor cores

The inductor cores to be inspected by this project are of type RM6-S, made from grade A10 ferroxcube, manganese and zinc, and are made as above with pressure during firing. The shrinkage is about 18%, but varies both between different batches and, to a lesser extent, within the same batch. These cores can be purchased from most electronic component distributors such as Farnell Components Ltd [69].

The shape of the sample cores under inspection in this project are defined in figure 5.1. The algorithms required for the image processing can be considered for each of a number of key features (illustrated in figure 5.3) and listed below.

- ❑ Core centre position
- ❑ Central pillar inner diameters
- ❑ Central pillar outer diameters
- ❑ Left flange arc
- ❑ Right flange arc
- ❑ Left flange end
- ❑ Right flange end

Each of these identifiable features must be found and form the basis of the measurements and inspection defined in the following sections.

5.1.2 The manufacturing considerations

The cores are cast from a slurry of ferrite dust suspended in liquid and compressed into a mould before it is heated to form the desired shape. The moulds absorb the liquid, to leave a hard solid material in the mould. During the manufacturing process the cores shrink, but the shrinkage is not constant and depends on the supplied raw material. Pressure is be applied to the drying slurry to reduce the amount of shrinkage which occurs during firing compared to purely slip casting, Farag [70].

The raw material is not of constant composition and therefore it is difficult to maintain consistency in production. Compensation for the variation in the supplied raw material is achieved by adjustment of the proportions of water and powder combined to produce the slurry and the temperature and pressure applied in the manufacturing process. The inconsistency of the raw material and variation of the process affect the cores by making them more or less likely to crack, and to have chipped edges and be over or under size on any of their dimensions so that their shape is not consistent.

5.1.3 Measurement of the ferroxcube inductor cores

In any measurement project it is necessary to establish which measurements are to be made and to what tolerance each of them must conform. Consequently, before any software is developed, it is important to ensure that the system is capable of measuring to the required precision. The precision of a vision system is dependant on both the proportion of the field of view which is occupied by the core and the resolution of the system; see chapters two and three.

5.1.3.1 The dimensions and tolerances specified for the core

The dimensions with their tolerances are listed below in tabular form and correspond to the drawing in figure 5.1 .

Dimension	Measurement upper and lower limits	
Inside diameter of pillar	3.0 mm	3.1 mm
Outside diameter of pillar	6.2 mm	6.4 mm
Overall length	17.3 mm	17.9 mm
Radii of inner flange walls	12.4 mm	12.9 mm

5.1.3.2 The measurements required of the vision system

The dimensions considered important are:

- ❑ The inside diameter of the central pillar
- ❑ The outside diameter of the central pillar
- ❑ The overall length
- ❑ The radii of the inner facing walls of the outer flanges

These dimensions are shown in figure 5.4 .

5.1.3.3 The predicted precision of the system

There are three main limitations to the precision of the measurement system, these are the resolution of the frame buffers, the resolution of the camera and the resolution to which the software works. The pixel resolution of each of the frame buffers is 512 x 512. The camera, which was used for development (Pulnix TM560) has a resolution of 500 x 582 but would be replaced by one of higher resolution when finally installed.

(1) The resolution of the hardware

The inductor core measures almost 18 mm in its longest dimension. The camera and lenses are arranged such that the inductor core occupies as much of the field of view as possible, while allowing some variation of position from sample to sample; about $^4/_5$ ths was found to be optimum. The resolution can therefore be calculated by knowing that four fifths of the image is approximately 430 pixels.

$$\text{pixel size} = \frac{18\text{ mm}}{430\text{ pixels}} = 0.042\text{ mm}$$

In this case the tolerance of a measurement between two points is to 0.084 mm (± $^1/_2$ pixel at each point), which is within the desired precision of 0.1 mm. However, this assumes that the frame captured into the frame buffer matches the aspect, size and resolution of the camera's image. TIPS captures a square image with square pixels, and to achieve this only uses part of the cameras 4:3 aspect ratio image. Thus with a camera with horizontal resolution of 500 pixels (Pulnix TM560) this is about 375 pixels. Consequently the resolution of the hardware using TIPS is:

$$\text{pixel size} = \frac{18\text{ mm}}{300\text{ pixels}} = 0.06\text{ mm}$$

This has a tolerance of 0.12mm, which is below the minimum resolution of 0.1 mm. It may therefore be necessary to use a camera which gives a resolution in the image of 512 pixels or has a resolution in the x direction of greater than 680 pixels.

(2) The resolution of the software

The accuracy of the measurement of the dimensions hinges on the accuracy to which the edges can be found. The algorithm to find the edge, described in section 5.3 and in software in section 5.4.4.3(2)(b), but in simplified terms compares the intensity gradient at each pixel with the next to find the pixel with steepest slope. Once the gradient is found all calculations are performed using floating point arithmetic. By using floating point arithmetic the software adds no significant errors after the edge has been found. Therefore it is only the initial edge searching software with its tolerance of one pixel that need be considered. Thus the theoretical resolution of the system hardware is still the overriding factor.

5.1.4 Check component for chips and damage

Some edges and corners are more prone to damage than others and are also more critical to the operational performance of the core. Consequently, while some edges need to be inspected, others may be ignored. The central pillar is considered the most sensitive part if affected by chips, since any reduction in cross sectional area may take the magnetic flux to saturation and increase the overall magnetic resistance. The outer corners of the flanges are considered the most vulnerable to damage.

The inspection concentrates on the dimensions and parts below:

- ❑ The inside circumference of the central pillar
- ❑ The outside circumference of the central pillar
- ❑ The inner edges of the flanges
- ❑ The outer edges of the flanges

A close-up photograph of a chipped corner is shown in plate 5.2.

5.1.5 Compare with specification for pass or fail

When all the measurements have been completed, they must be compared with the dimensions of the specification in order to ensure that the core lies within the defined tolerances. It is easier to establish whether a core should pass or fail the inspection on the basis of the measurements. However, the presently qualitative inspection for damage proves to be more difficult since no specifications define how big a chip can be before it is sufficiently significant for the core to fail. The inspection progressed, using the tolerances for dimensions given in the specification, by assuming that all possible diameters and lengths, within the definition of the dimension, must lie within the prescribed limits. For example, all diameters of a circle must lie within the tolerance for that dimension. Questions such as "How round does the circle have to be?" were considered in order to establish a basis for measuring a chipped circumference.

5.2 The computing hardware and optical arrangement

The development of the algorithms and software, necessary for solving the problems presented by this application, have been developed side by side with the image processing system on which it has been implemented. As a consequence the Transputer Image Processing System, TIPS, was modelled using an existing IBM based image processing board with the programs developed on a transputer board in the IBM-AT, described in this section.

5.2.1 The vision system

The application was developed to be performed on the Transputer Image Processing System (TIPS), Chambers [24] (refer to section 2.4) which provides the capability of performing the image processing within the required period. While this system was being developed, the software was written on a general purpose transputer board, "B004", made by Inmos [64], with images captured and displayed on a Matrox "PIP-1024" framestore [2]. These boards transfer the raw and processed images through the IBM-AT which acts as their host.

5.2.1.1 Modelled on B004 and PIP

The development system described in this section is shown diagrammatically in figure 5.5. It is based on an IBM-AT personal computer into which are plugged an Inmos B004 transputer board and a Matrox PIP framestore. The software was written using the Transputer Development System (TDS) [65], which is supplied with the B004 board. Hence, programs were compiled ready for running, testing and debugging.

(1) The Inmos B004 transputer board

The B004 board comprises a transputer, 2 Mbytes of memory, mapped into its address space, and also an interface to an IBM-PC compatible computer. The transputers links are available for configuration as required, but, if used as the only B004 board in an IBM computer, one of these is connected to the interface part of the board to enable communication between the transputer and the IBM computer.

Programs are run on the transputer by sending them to it via one of the links shortly after the transputer has been reset. The TDS software runs on the B004 which limits the size of user programs that can be developed if they are also to be run on this board.

(2) The Matrox PIP framestore

The PIP framestore offers four 512 x 512 resolution frame buffers of 256 grey levels, and it interfaces to the IBM computer by a number of registers mapped into the IBM's I-O map. Thus, any pixel may be accessed (read or written) by writing the x and y locations of the pixel into two address registers and reading from or writing to a third data register. A feature designed to speed up pixel access is also incorporated in the design, in which the address registers are automatically incremented after each access. Thus, once the line start address has been given, the rest of the pixels in the line can be read or written consecutively by reading from or writing to the data register. Other registers are also provided to set up hardware timings and to configure the board.

The resolution of 512 x 512 with 256 grey levels is the same as in the TIPS. Unlike TIPS though, the pixels are not square, but have an aspect ratio of 4 : 3 to match standard television monitors. This complication is particularly obvious when performing image processing on circular objects, such as the ferroxcube core central pillar. The equation of the circle must be modified to account for the unequal aspect ratio. By using compensating factors, this could be changed for implementation on the TIPS with its square pixel images.

(3) TIPS modelled

Initially, the software was developed on the hardware, as shown in figure 5.5, with the program run on the B004 and the images captured and displayed using the PIP framestore. The sequence of events of an image processing iteration in the program was as follows.

i Ask the PIP framestore to capture a new image and transfer it through the IBM to the B004
ii Process this image
iii Draw results and report information on to the image
iv Transfer the modified image back through the IBM computer to the PIP framestore for display analysis

The model was similar to the TIPS, with the image appearing in memory ready for the image processing program to perform its functions; all that needed changing was the subroutine which captured the image and also the removal of the one which displayed the processed image at the end of the sequence. The iteration time was extremely slow, as it took over six seconds to transfer the image to or from the PIP framestore and only a single transputer was available to run the program.

A transputer board has been developed at the University of Liverpool, named the "2MBTB", standing for "Two Mega Byte Transputer Board", Chambers [24], and this was used to run the inspection software as a target system representing a TIPS image processing board. Hence with the 2MBTB, the model of TIPS more closely represented the hardware configuration on which the software would ultimately run.

The programs were compiled on the B004 board and sent to the 2MBTB to run in a similar way as they now do on TIPS. The image transfer time was still the major performance limitation, but the whole program was developed using this modelled system.

5.2.1.2 Implementation on TIPS

The realisation of TIPS removed the time "overhead" in transferring the image from, and back to, the PIP frame store. The configuration of the system on which the trials in this thesis were performed is shown in figure 5.6. This shows the complete implementation used to gain results and to test the software with TIPS configured with one image processing board, a frame capture board and a system display board. Two 2MBTBs were used, one handling the control of the software and terminal communication and the other used to run TDS [65]. The 2MBTB running TDS, would not be required in systems for use on the shop floor, but during development allowed storage of results and associated images on to the hard disk in the IBM computer. A data terminal, connected to the TIPS system display board via a Transputer Serial Adapter Card (TSCA), which is used for keyboard input by the operator and status information.

The video input from the camera is connected to the frame capture board of TIPS which sends the digitised image to the image processing board under the control of the system display board. The image processing board then processes the image to measure the core and find any chipped edges.

5.2.2 The optical arrangement

Various arrangements of the camera, lens, lighting and method were pursued to ease the image processing software task, before the most effective one was chosen. The final configuration is shown in figure 5.7 and in plate 5.3. The camera is mounted vertically above the inductor core with illumination provided by two lamps either side of the lens. The camera is fitted with a standard 16 mm lens with extension tubes between the lens and the camera so that the ferroxcube core fills the frame.

5.2.2.1 Illumination

The arrangement used to light the core for inspection proved to require some experimentation. The difficulty was to distinguish between the top of a flange and the base of the core. Since the base and the flange are of identical material they show similar shades on the image and determination of the edge between them is difficult. The use of two lamps, either side of the camera lens, shining down onto the core created shadows on the base but illuminated the flanges and pillar top surfaces directly. This illumination with the expected intensity profile is shown in figure 5.8 along with the profile produced by a uniform light source. Image 5.2 shows a typical image with the actual intensity profile along the horizontal central section line. Once this profile had been established and had proved reliable, the algorithms used to find the required dimensions could be developed. Lamps with a source area of 25 mm were found to produce the best image, offering enough shadow on the core base to make it distinguishable from the flange and pillar tops, yet diffuse enough to allow detail to remain along both inner and outer edges. The intensity from each source on the background (and across the core) diminishes with the inverse square of the distance from each lamp. This produces a background intensity at either end of a core which is higher than the flange top and negates the shadow from the bulb at the opposite end of the core.

5.2.2.2 Camera settings

A Pulnix TM560 [71] CCD camera is used with the Automatic Gain Control, AGC, turned off in order to ensure images of constant brightness level (see section 3.1.3.2). The camera is positioned about 140 mm vertically above the core with a standard 16 mm lens set with its aperture at F4 to produce the optimum image quality. A combination of extension tubes measuring 12 mm in total are placed between the camera and the lens to produce an image of a core at maximum size.

5.2.2.3 The background

A plain, flat, neutral background of mid grey is used to make it distinguishable from the dark grey core, without causing a large contrast difference between the background and the core. A white background is not used as this causes blooming at the edges between the core and the background and black is not used as the contrast between the background and the core is insufficient.

5.2.2.4 Other arrangements which were rejected

Single sided illumination did not prove successful since the shadows, which were created, occurred on one side of the feature making some edges difficult, if not impossible, to detect. Dual, low side illumination using point light sources was also unsuccessful as, although the contrast between the shadow regions and the illuminated top surfaces was excellent, the contrast between the background and core at the outer flange edges was inadequate to determine the edge.

Laser stripe illumination (see chapter four) with the camera mounted at 45° to the vertical was considered to provide the cross-sectional information of the core. There were three reasons for not using this method.

- i The curvature of the flanges could obscure the laser line
- ii The positioning and orientation of the core under the line needed to be far more precise than when viewing the whole image
- iii The measurement and inspection is only along the cross section of the laser line.

5.3 Discussion of the analysis to find the core's features

The software, which is written to achieve the tasks of inspection and measurement, can be considered as a number of small operations or parts.

- ❑ Wait for input advising that a core is awaiting testing
- ❑ Find the core's precise location in the image
- ❑ Find edge features
- ❑ Measure lengths / diameters / radii
- ❑ Check for chipped edges and corners
- ❑ Output results to screen / robotic system
- ❑ Log data for later analysis or filing

Some of these parts can be performed simultaneously, while others require data from one part before another can continue. As edge feature locations become available, the measurement and checking for chips can start and be performed concurrently, outputting the results to the screen and to any robotic software which needs the data for its operation. These results may also be saved if data logging is required. An overview of the software structure can be seen in the image processing software top level fold shown in section 5.4.4.3(2) while the above parts are discusssed in this section.

5.3.1 Wait for input advising that a core is awaiting testing

The program cannot perform the other functions until an inductor core is ready and an image of it is available. Thus, while the software is waiting for a core no other processing is performed. Once the image is available, the algorithm to find the core's precise location on the image can begin, hence providing this information for the rest of the program. The instruction advising the system that a core is ready for inspection is given via one of the transputer links, and the software waiting for this instruction uses up no processing time while waiting for the channel communication from a transputer link. This is implemented in Occam as follows:

```
INT go:
CHAN OF INT next.core:
SEQ
    next.core  ?  go
```

Occam is described in chapter three and detailed in The Occam Programming Guide, Hoare [28], but the program will not pass the "next.core ? go" line until some data is received along the channel called "next.core" and assigned to the integer variable "go".

5.3.2 Location of the core

The location of the core in the program, as shown in figure 5.9 is given on the image by the centre of the central pillar. In order to find the location, two intensity profiles, lying across the core's central pillar, are examined, one horizontal and one vertical, both passing through the centre of the image. These profile lines intersect with the pillar's edges to reveal four points on the outer circumference of the pillar. Stipulating that the hole in the pillar must lie somewhere over the centre of the image, the core's location is found with the following algorithm.

Given four coordinates lying on the circumference of a circle with radius "r", on which any point is described by x and y in the equation:

$$(x - x_c)^2 + (y - y_c)^2 = r^2$$

the centre of the circle (x_c , y_c) can be found from the combination of the equations of the circle and the straight lines running horizontally and vertically through this circle.

The centre point of a circle can be found from the intersection of two perpendicular bisectors of any two different chords of the circle. Using horizontal and vertical chords, detailed in figure 5.9, produces the following equations for the centre of the core.

$$x_c = x_{h1} + ((x_{h2} - x_{h1}) / 2)$$

$$y_c = y_{v1} + ((y_{v2} - y_{v1}) / 2)$$

In order to compensate for chipped edges on the pillar, the centre is also found from the points at which the intersecting lines cross the hole at the centre of the pillar. The arithmetic mean of these two results gives the location of the centre of the ferroxcube core. This method imposes a limitation on the variation in position of the ferroxcube core within the image, from test to test.

5.3.3 Find the required features of the core

Once the centre of the core has been determined, two more intersecting lines are examined which pass through the centre point of the core, and from analysis of the intensity profile, the features are then extracted between which the dimensions are measured.

5.3.3.1 Outside edges of the flanges

Intensity lines are analysed from the edges of the image towards the centre in order to find the change in intensity profile which denotes the outside edge of the flanges between the core and the background. Since there is a limit to the distance that a core may move (its central hole must lie over the centre of the image), the search for the edge of the core along the new intensity profiles are also limited.

5.3.3.2 Inside and outside edges of the central pillar

The positions of inside and outside edges of the central pillar are found from the intensity profile of the plan view figure 5.8, and are compared with the values of the diameters calculated when finding the centre point of the core (section 5.3.2.1). Both the inside and outside pillar edges are found in four places from the vertical and horizontal intensity profile plots. This process enables the rejection of values resulting from chipped or damaged edges. The program then calculates the average of the good values in order to find the best fitting equations for their circles. Thus, having calculated the radii of the pillar, the equations of the inner and outer circles are defined which allows the checking for damage to proceed; see section 5.3.5.2 .

5.3.3.3 Inside curved edge of the flanges

The radius of the inside curved edge of each flange is found in the same way as the diameters of the central pillar edges, except that in this case only the horizontal profile is used. The distances from the centre of the core to their respective edges defines the required radii. Now with these radii known, the equation for each of the two edges are defined.

5.3.4 Measurement of the dimensions

The system is calibrated in both the x and y directions to establish the size, in millimetres, which each pixel represents in each direction. The dimensions which lie only in the x or y directions can be measured by counting or calculating the number of pixels representing that dimension on the image and multiplying this value by the calibration scaling factor; see section 5.5 on calibration scaling. When calibrating dimensions or equations which have a direction involving both an x and y part, the calibration factors must be used separately on both the x and y components before they are combined.

5.3.5 Inspection for damage

The inspection of the inductor cores for deformity and damage is applied only to the plan view. The flange edges and the central pillar are checked by comparing the point found on the edge with the next point closest to it, calculated from the equation of the template line running along that edge.

5.3.5.1 Flange edges

The point obtained from the intensity profile which indicates the outside edge of a flange, is used as the start point from which to find the equation of the edge's template line. The point of maximum slope between dark and light is taken as the interface point between the core and the background. This point is found for each value of y, looking at the intensity profile in the x direction, over a limited range. The approximate slope is found by subtracting two values either side of the one under test. The largest difference gives the point of maximum slope. Since the values along the edge should change slowly, any single pixel which is significantly out of line is placed back between its neighbours to reject noise. The algorithm which is used to find the points is shown with Occam style constructs below.

```
PAR
    ... look in an upwards direction to find the edge.
    ... look in a downwards direction to find the edge.
```

The two parts can be performed concurrently one looking at the image above the start coordinate and one looking at the image below it. Considering one of these, the upward direction, (the downward is the same algorithm but in the opposite direction):-

```
BOOL good.edge:
SEQ
    good.edge := TRUE
    WHILE good.edge
        ... decrement y          -- remember the image origin is at the top left corner
        SEQ this.x = x.start FOR 20
            ... calculate the difference in intensities either side of this point
            ... remember this difference and x value if it is the biggest so far
        ... check that the line has a significant edge, set "good.edge" false if not
```

Once the edge points have been found they are compared with the line equation on which they are expected to lie. If they are too far from the expected line, the edge is deemed to have been chipped. This inspection technique works in any orientation, with the line equation for each edge being calculated from the orientation of the core. For purposes of demonstration the cores were placed horizontally in the frame.

The flanges' edges are compared with these line equations, including that for the vertical edge and those for the two 45° edges. The edge equations are of the form

$$y = m x + c$$

where the values of m and c are derived for each core's location.

5.3.5.2 Flange arcs

The flange arcs are found in the same way as the edges, except that the comparison is made with the equation of the corresponding circle:

$$(x - x_c)^2 + (y - y_c)^2 = r^2$$

where r is the radius of the arc, obtained from the location of the edges. The program then looks upwards and downwards moving successively along the line until the top or bottom corner of the flange is found. Beyond this corner only background appears on the intensity profile line.

5.3.5.3 Pillar circumferences

The central pillar is divided into four sectors to check for damage (top, bottom, left and right). The sectors are processed in the same way as the flange arcs, with left and right sectors decrementing and incrementing y and finding the corresponding value of x. The top and bottom arcs are processed by decrementing and incrementing x and finding the corresponding value of y. The comparison with the template circles and the found edges enables chips to be detected in the same four sectors.

5.3.6 Comparing the results with the specification tolerances

If the measured dimensions fall between the predetermined values which are stored in the program for that dimension and the core is not chipped, it passes the test. The comparison of the measured result for each dimension was applied as follows, with the values stored in the program are shown in the table in section 5.1.3.1 .

```
IF
        (this.result > min.dimension) OR (this.result < max.dimension)
                componentpasses := TRUE
        TRUE
                componentpasses := FALSE
```

After all the dimensions have been checked, and if the variable "component passes" is TRUE, then the core is deemed to have passed the inspection and the results are issued. Conversely, if it fails, the "fail" signal is given.

5.3.7 Output results to a screen or robotic system

The output device for the results may be either a data terminal, the image processing display, a printer, a logging system, a robotic system, or a combination of these. Communication with these types of devices is often slower than the rate at which the results are being produced, therefore either the process slows down or some of the results are lost. It was decided to put the pass or fail message onto the image processing display, and also to offer the capability of transmitting the result along an RS232 standard serial line, so that any compatible device could receive the result.

Since a robot would have its own specific interface requirements, the pass or fail signal would be implemented along a transputer link in a similar way to the serial communications adaptor TSCA board, Chambers [24].

5.3.8 Logging the data for later analysis or filing

To evaluate the accuracy of the measurements made by the inspection, the results presented later in this chapter needed the development of data logging procedures to save the measurements for post analysis. This involved outputting edge locations, measurements, and sample images to an IBM-AT computer.

During discussion of the system with the Philips' production engineers, it became clear that the measurement data could be used to adjust the manufacturing process. The data, or results, are used to see whether, and how the manufacturing process is going wrong, by analysing the numbers of cores which are passing and failing and whether the components going over or under size. Building up a greater knowledge of the process behaviour may allow prediction and better control of the process by compensating for variation in raw material composition.

5.4 Discussion of the software

The image processing software as described in this section is configured to run on a single image processing board. A discussion is given of on-line implementation which describes how the programs and algorithms could be divided to run on several boards. Implementation on one image processing board still requires the software to run on the system display board and also on a root processor, see figure 5.6. All software is written in Occam with a fold editor and is described in this context.

5.4.1 System configuration software

The top or configuration level software defines what software is running on which processor, how the processors are connected together and, therefore, the communication between processors and the type of processor on each board.

```
...  protocols
...  SC root software
...  SC sdb software
...  SC ipb software
...  channel addresses
...  chans
PLACED PAR
  PROCESSOR 0 T4            -- ROOT
    ...  root software
  PROCESSOR 1 T4            -- SDB
    ...  sdb software
  PROCESSOR 2 T4            -- IPB
    ...  ipb software       -- Ferroxcube measurement and inspection
```

Each line starting "..." indicates that more source code lies within this line and is called a fold. Thus the line "... protocols" reveals the code which follows.

```
PROTOCOL commands
  CASE
    capture;   INT
    display;   INT
    process;   INT
    i.request; INT
    r.request; INT
    b.request; INT
    specimen;  INT
:
```

```
PROTOCOL transmit
  CASE
    paused;        BOOL
    finished.processing; INT
    i.transmit;  [512][512]BYTE
    r.transmit;  [22]INT
    rr.transmit; [22]INT; [10]REAL32
    r.transmit;  [22]INT
    rb.transmit; [512][512]BYTE; [512][512]BYTE;
                   [22]INT; [10]REAL32
    b.transmit; [512][512]BYTE;[512][512]BYTE;[22]INT
:
```

Two protocols are defined which state the order of transmission of data along channels in the software and are used for communication between the root processing board, system display board, image processing board and the board running in the host IBM-AT. The names preceding the data type information are sent first as a "tag byte" to indicate what information follows, thus if the protocol "capture" is transmitted, this will always be followed by an integer. This is shown in use in section 5.4.2, where data is received from a channel with the protocol "commands".

The lines starting "... sc" house the programs which are to run on the processors defined below the line "PLACED PAR". In this software the three programs are those for the "root", "SDB" and "IPB" processors. Within the "PLACED PAR" the processor types are defined, for the inspection system these are all T414 transputers signified "T4" and given a number. Here the root is numbered 0, SDB 1 and IPB 2.

5.4.2 Root processor software

The program which runs on the root processor is designed to allow communication between the host IBM-AT computer and TIPS. The program waits for communication from the channel "from.sdb" and receives the associated data, as defined by the protocols in 5.4.1. The program then re-transmits the data in the same format to a program on the host transputer in the IBM-AT. The "WHILE TRUE" loop commits the program to loop, for ever waiting for more data once the last has been successfully transmitted.

```
... protocols
PROC root(CHAN OF transmit to.root, from.root,
          CHAN OF commands to.sdb,  CHAN OF transmit from.sdb)
  INT frame.no, junk:
  [512][512] BYTE image1, image2:
  [22] INT results:
  [10] REAL32 real.measurements:
  BOOL pause:
  SEQ
    WHILE TRUE
      SEQ
        from.sdb ? CASE
          i.transmit; image1
            from.root ! i.transmit; image1
          rr.transmit; results; real.measurements
            from.root ! rr.transmit; results; real.measurements
          rb.transmit; image1; image2; results; real.measurements
            from.root ! rb.transmit; image1; image2; results; real.measurements
:
```

The protocols are defined and the function call "root(   )", gives the order of the channels and names which they are called in this program.

5.4.3 System Display Board software

The program which runs on the System Display Board has the following functions:

- □ Maintain continuous running of the system by issuing commands
- □ Service the frame capture board for capture sequences
- □ Respond to user input from the terminal
- □ Update terminal screen
- □ Collate data, calculate measurements and transmit for logging

In order to execute this in the software, there are two main parts to the program: that which talks to the terminal, "terminal handling" and the main program denoted "run image processing".

```
... protocols
PROC sdb(CHAN OF commands to.sdb, CHAN OF transmit from.sdb,
         CHAN OF commands to.ipb, CHAN OF transmit from.ipb)

  ...  libs
  ...  channel addresses
  ...  terminal stuff
  ...  sdb vals
  ...  sdb vars and placements
  ...  vars
  PAR
    SEQ
      ...  setup
      ...  run image processing
    ...  terminal handling
:
```

The terminal interface software, "terminal handling", runs in parallel with the main program and takes the data from the terminal and passes the command to the main program. This software also supports "xon - xoff" handshaking.

5.4.3.1 Compiler set-up information

The folds ("..."), above the "PAR" statement, house the set-up information for compilation. The fold "libs" tells the compiler which libraries are used for functions such as for reading from and writing to the terminal, and include "write.full.string()", as well as the routines for type conversion such as "REAL32TOINT()" and "INTTOSTRING()". Full descriptions of these and the other functions used within the software are given in the TDS manual [65]. The definitions of channels for the terminal and their placement on links is defined in the fold "terminal stuff" and is given below, along side the corresponding fold. The variables used to control the various registers on the system display board hardware are given in the following fold, "... vars". This fold also defines the mapping of the frame buffers in the transputer memory at hex address #30 000 000 and the frame capture board hardware registers addresses.

```
...  libs
...  channel addresses
...  terminal stuff -     -     -     -     -
{{{  sdb vals, vars and placements
VAL INT o.port.val IS #D7: --6,7 not used = 1,
--5 fold = 1(!fold image),4 fgrab = 1(!capture),
--3 disp.frame.0 = 0(frames 0 & 2),
--2 user.led = 1(on),1 ss.analyse = 1(inactive),
--0 ss.reset = 1 (inactive)
VAL INT d.reg.0 IS #37:--display active horiz
VAL INT d.reg.1 IS #E8:--display front porch
INT   frame.no, fcb.disp.reg, o.port, i.port:
PLACE  o.port AT #20000000:
PLACE  i.port AT #20020000:
[2]INT d.reg:
PLACE  d.reg  AT #20030000:
[512][128] INT id:
PLACE  id     AT #30020000:
[4]INT clut:
PLACE  clut   AT #20010000:
CHAN OF INT   event:
PLACE  event  AT 8:
[512][1024]BYTE display.image:
PLACE display.image AT #30000000:
PROTOCOL fcb.pro IS BYTE; BYTE:
CHAN OF  fcb.pro to.fcb, from.fcb:
PLACE    to.fcb   AT link3.out:
PLACE    from.fcb AT link3.in:
}}}
```

```
{{{  terminal stuff
CHAN OF ANY screen:
CHAN OF INT keyboard:
CHAN OF BYTE to.term:
CHAN OF BYTE from.term:
PLACE to.term AT link2.out:
PLACE from.term AT link2.in:
...  scr.key.ANSI term driver
}}}
```

5.4.3.2 Run-time set up of the System Display Board

Before the main part of the program can run and for the board to work properly, interface software must set up the hardware on both the frame capture board as well as the system display board. This is contained within the "setup" fold and is as follows.

```
SEQ
  ...  set up fcb
  ...  set up sdb
  ...  put up background and border
  ...  capture into all buffers
  ...  Screen display
```

(1) Setting up the Frame Capture Board

Firstly, the static values used to set up the registers on the Frame Capture Board (FCB) are defined. These configure the timings of the system and have the effect of setting the size of the display, its position within the frame, and the position of the image within the frame. These standard values are sent to the registers on the FCB with the "SEQ - FOR" loop and put it into the initial state defined in the comments in the software below.

```
{{{  fcb vals
VAL [] BYTE fcb.reg.val IS [ #00(BYTE),    --  Back porch
                             #C8(BYTE),    --  Front porch
                             #80(BYTE),    --  Active horizontal
                             #DC(BYTE),    --  Top border          (#DC = 34 lines)
                             #00(BYTE),    --  Bottom border
                             #00(BYTE) ]:  --  Active vertical     (#00 = 256 lines)
VAL BYTE disp.reg.val IS     #03(BYTE):    -- 7 user.led = 0 (on),
--6,5 not used = 0,4 fold = 0(!fold image),
--3 transparent = 0(!transparent),2 up.not.down = 0((0,0) at top left),
--1 disp.frame.0 = 1(frames 0 & 2),0 capture = 0(!capture)
}}}
SEQ
  SEQ reg = 0 FOR 6
    to.fcb ! BYTE(reg); fcb.reg.val[reg]
  to.fcb ! 24(BYTE); disp.reg.val
  fcb.disp.reg := ((INT(disp.reg.val)) /\ #FF)
```

(2) Setting up the System Display Board

The software uses the values given in the compiler set-up above. The position of the display is defined by the values for the front and back porch set into the registers "d.reg[0]" and "d.reg[1]". The System Display Board (SDB) must be instructed which pixels to display from each Image Processing Board (IPB). This is performed with one IPB as follows by setting the "ID memory" as below.

```
SEQ
  o.port    := o.port.val                          {{{  set id memory
  d.reg[0]  := d.reg.0    -- front porch           SEQ id.y = 0 FOR 512
  d.reg[1]  := d.reg.1    -- back porch              SEQ id.x = 0 FOR 128
  ...  set id memory-    -    -    -    -             id[id.y][id.x] := #00
  {{{  set clut                                    }}}
  SEQ
    {{{  wait for syn
    WHILE ( i.port /\ #04)=4                       {{{  wait 2 lines
      ...  wait 2 lines   -    -    -    -         TIMER clock:
    }}}                                            INT   time:
    clut[0] := #00                                 VAL   delay IS 2:
    SEQ shade = 0 FOR 248 -- CLUT grey scale       SEQ
      SEQ                                            clock ? time
        clut[1] := shade -- red                      clock ? AFTER time PLUS delay
        clut[1] := shade -- green                  }}}
        clut[1] := shade -- blue
    ...  set colours
    clut[2] := #FF
  }}}
```

Plate 5.3 shows an image of a core under inspection; either side of the image is an area of the screen not displaying the image, but a border. This can be used for graphical display and is under the control of the system display board. It has been set to display a blank blue background with a red border. The code to do this is contained in the fold "... put up background and border". This code operates as follows with the border width of two pixels and x limit giving a display area of 795 pixels.

```
SEQ
  {{{  put on background
  SEQ x = 0 FOR 795
    SEQ y = 0 FOR 512
      display.image[y][x] := BLUE
  }}}
  {{{  put up border
  SEQ y = 0 FOR 512
    PAR
      display.image[y][0]   := RED
      display.image[y][1]   := RED
      display.image[y][795] := RED
      display.image[y][795] := RED
  SEQ x = 0 FOR 795
    PAR
      display.image[0][x]   := RED
      display.image[1][x]   := RED
      display.image[511][x] := RED
      display.image[510][x] := RED
  }}}
```

The four frame buffers on the image processing board are initialised next by issuing a capture command to the IPB and executing the capture sequence on the SDB for each buffer. This is performed in the loop below, with a full explanation of the capture procedure ("... capture") given later in this chapter in section 5.3.3.3(2)(a).

```
SEQ frame.no = 0 FOR 4
  ...  capture
```

(3) Setting up the terminal

The final part of the set-up operation of the system is to set the display screen of the terminal. Communication with the terminal is via an RS232 serial connection and the terminal responds to hexadecimal instructions and escape sequences defined by the VT220 terminal manual [72]. The software written to achieve this is listed below where the terminal screen is first cleared, then the title text is put on in double height and the instructions in normal height letters.

```
...  vals
SEQ
  ...  clear screen -    -    -    -    -    -
  {{{  title screen
  SEQ
    ...  Liverpool university - double height
    ...  Ferroxcube core inspection - double
    ...  Instruction information to screen
  }}}
```

```
{{{  clear screen
BYTE char:
SEQ
  char := 27(BYTE)
  write.char(screen, char)
  char := '['(BYTE)
  write.char(screen, char)
  char := '2'(BYTE)
  write.char(screen, char)
  char := 'J'(BYTE)
  write.char(screen, char)
  }}}
```

The escape sequences to clear the screen are sent as shown in the "clear screen" fold listed along side the program above.

5.4.3.3 Main program on SDB - "Run Image Processing"

The main loop of the SDB software either responds to an input from the channel from the terminal software, running in parallel on the SDB, or initiates and coordinates the running of the image processing software performing the inspection on the IPB.

```
...  vars, tolerances
SEQ
  get.results := FALSE
  scroll := FALSE
  pause := FALSE
  WHILE TRUE
    ALT
      keyboard ? key
        ...  do terminal
      TRUE & SKIP
        ...  do processing
```

The fold "vars, tolerances" contain the definition of the variables and static values used in the program and define the maximum and minimum measurement dimensions for inspection. The calibration factors are also defined here which give the horizontal and vertical size of each pixel in millimetres.

```
{{{  vars, tolerances
BOOL scroll,  pause,  get.results:
INT junk, key, error, specimen.no, results.count:
INT inner.circle.left.x,       inner.circle.right.x,          outer.circle.left.x:
INT outer.circle.right.x,      inner.circle.upper.y,          inner.circle.lower.y:
INT outer.circle.upper.y,      outer.circle.lower.y,          inner.flange.left.x:
INT inner.flange.right.x,      outer.flange.left.x,           outer.flange.right.x:
INT left.flange.x.radius,      inner.x.radius,   centre.x,    outer.x.radius:
INT right.flange.x.radius,     inner.y.radius,   centre.y,    outer.y.radius:
REAL32 real.inner.x.diameter,  real.left.flange.x.diameter:
REAL32 real.inner.y.diameter,  real.right.flange.x.diameter:
REAL32  real.outer.x.diameter, real.outer.y.diameter,         real.core.length:
[10]REAL32 real.measurements:
VAL REAL32 real.expected.core.length.max      IS     17.9(REAL32):
VAL REAL32 real.expected.core.length.min      IS     17.3(REAL32):
VAL REAL32 real.expected.flange.diameter.max  IS     12.9(REAL32):
VAL REAL32 real.expected.flange.diameter.min  IS     12.4(REAL32):
VAL REAL32 real.expected.inner.diameter.max   IS      6.4(REAL32):
VAL REAL32 real.expected.inner.diameter.min   IS      6.2(REAL32):
VAL REAL32 real.expected.outer.diameter.max   IS      3.1(REAL32):
VAL REAL32 real.expected.outer.diameter.min   IS      3.0(REAL32):
VAL REAL32 cal.factor.x                       IS   24.675(REAL32):
VAL REAL32 cal.factor.y                       IS   22.749(REAL32):
}}}
```

(1) The terminal program

The terminal program offers the opportunity of entering four input characters during the normal running of the program to perform three tasks. These are " ", "t", "s" and "p" and can be seen from the program below.

```
SEQ
  IF
    ...  key = ' ' - pause
    ...  key = 't' - transmit mode
    ...  key = 's' - scroll results to terminal
    ...  key = 'p' - set specimen number
    ...  TRUE      - ( SKIP )
```

If the key which has been pressed is not one of the four special characters "SPACE", "t", "s" or "p", it is ignored.

(a) Pause

When the space bar is pressed on the terminal the program continues to run without issuing any further capture and process sequences until it is pressed again, thus continuing to service the terminal input and output. The outline Occam code to perform this follows.

```
key = (INT(' '))
  {{{  pause
  SEQ
    IF
      pause = FALSE
        SEQ
          pause := TRUE
          ...  tell screen - PAUSED
      TRUE
        SEQ
          pause := FALSE
          ...  tell screen - RUNNING
  }}}
```

The variable pause is used as a software switch and is set to TRUE or FALSE and used in the program's main loop.

(b) Transmit mode

The transmit mode operation is designed to save data about a specimen and / or its associated raw and processed images. The user requests the transmit mode by typing the letter "t" on the terminal followed by the type of transfer which is required. This may be just the currently displayed image, the results from the last inspection or both. These operations are initiated by typing "i", "r" or "b" and, as before, if any other key is pressed it is assumed to indicate that no transmission (or action) is required and the program continues normal operation.

If one of the keys "i", "r" or "b" is pressed, the software tells the image processing board that the transmission of data and / or images is required, by using the "CASE" with the respective protocol, and then it waits for the data to be sent in the fold labelled "get, calc and send data". If pixel locations are returned, the program calculates the dimensions from them and adds them to the data transmitted to the "Root processor board".

```
key = (INT('t'))
  {{{  transmit mode
  ...  vals
  SEQ
    ...  do terminal display
    {{{  wait for key and action
    ...  vars
    SEQ
      no.action := FALSE
      keyboard ? key
      IF
        key = (INT('i'))
          ...  request image transfer
        key = (INT('r'))
          ...  request results transfer
        key = (INT('b'))
          ...  request i & r transfer
        TRUE
          SEQ
            no.action := TRUE
            ...  do terminal display
      IF
        no.action
          SKIP
        TRUE
          SEQ
            ...  get, calc and send data
    ...  do terminal display
    }}}
  }}}
```

```
{{{  request i & r transfer
SEQ
  write.char(screen,(BYTE(key)))/
  to.ipb ! b.request; 0
}}}
```

```
{{{  get, calc and send data
from.ipb ? CASE
  i.transmit;image1
    ...  send
  r.transmit;results
    ...  calc real.meas & send
  b.transmit;image1;image2;results
    ...  calc real.meas & send
}}}
```

The calculation of the dimensions from the pixel locations is discussed later in this chapter under the main processing loop.

(c) Scroll data mode

The scroll data mode has the effect of toggling a software switch "scroll" to tell another part of the program to display all the results of every inspection on the screen of the terminal.

```
key = (INT('s'))
  {{{  scroll mode
  IF
    scroll
      scroll := FALSE
    TRUE
      scroll := TRUE
  }}}
```

(d) Specimen number

If inspection of specific inductor cores is being performed, for example when taking the results at the end of this chapter, the specimen number function allows the user to enter a number corresponding to the specimen, which is then given to the program running on the IPB for display on the processed image.

```
key = (INT('p'))
  {{{  specimen no
  ...  vals                                                 {{{  do screen
  SEQ                                                       SEQ
    ...  do screen    -    -    -    -    -    -           -{  ...  clear screen
    {{{  wait for key and action                              ...  title
    SEQ                                                       ...  put up display
      keyboard ? key                                        }}}
      IF
        ( (key >= (INT('0'))) AND (key <= (INT('9')) ) )
          ...  set specimen.no and echo to terminal
        TRUE
          SKIP
    }}}
    ...  Screen display
  }}}
```

(2) The main processing loop

The main program loop performs three operations: to set the new frame number to be used on the IPB, capture an image and process it provided that the system has not been put into paused mode by the user as described in section 5.4.3.3(1)(a).

```
IF                                                          {{{  set next buffer
  pause = FALSE                                             IF
    SEQ                                                       frame.no = 0
      ...  set next buffer-    -    -    -    -    -       -{      frame.no := 2
      ...  capture                                            TRUE
      ...  process                                              frame.no := 0
  TRUE                                                      }}}
    SKIP
```

The method of swapping the frame buffers is shown along side the fold above and shows that only buffers 0 and 2 are used.

(a) The capture sequence

The capture sequence controls a hardware interface and involves switching register values on the FCB and SDB and telling the IPB to load its register values at the correct time. Firstly, the program checks that the event channel has not been set by a previous frame and then it waits for a new event to signal the start of the next frame. The FCB register is set to capture this frame and the banks set by the "o.port" register on the SDB. The IPB is then told to prepare to capture this frame by selecting the correct frame buffer. The software then waits for the end of the frame, signified by the event given at the start of the following frame, and stops capturing.

```
SEQ
  ...  flush event
  ...  wait for event
  ...  wait a field
  SEQ
    IF
      (frame.no /\ #02) <> 0                    -- ie frames 2 and 3
        SEQ
          to.fcb ! 24(BYTE); BYTE((fcb.disp.reg \/ #02) /\ #7F) -- led on
          ...  wait for event
          o.port := (o.port /\ #F3) /\ #EF  -- led on, f.grab on
      TRUE                                      -- ie frames 0 and 1
        SEQ
          to.fcb ! 24(BYTE); BYTE((fcb.disp.reg /\ #FD) \/ #80) -- led off
          ...  wait for event
          o.port := (o.port \/ #0C) /\ #EF  -- led off, f.grab on
    to.ipb ! capture; frame.no
    ...  wait for event at end of capture
```

(i) Flush event, Wait for event, Wait a field

```
{{{  flush event       {{{  wait for event       {{{  wait a field
INT junk:              INT junk:                 TIMER clock:
PRI ALT                SEQ                       INT time:
  event ? junk           event ? junk            VAL delay IS 15625/30:
    SKIP               }}}                       SEQ
  TRUE & SKIP                                      clock ? time
    SKIP                                           clock ? AFTER time PLUS delay
}}}                                              }}}
```

(b) Process

The IPB is told to process the frame it has just captured by a channel communication from the SDB. The SDB then waits for the result to be returned from the IPB before calculating the measurements and, if in scroll mode sending them to the terminal, and to the IBM-AT if they have been requested.

```
VAL proc.frame.no IS [ 2, 3, 0, 1 ]:
SEQ
  to.ipb ! process; proc.frame.no[frame.no]
  ...  results from ipb
  ...  results to terminal if scroll mode on
  ...  request results transfer
```

5.4.4 Image Processing Board software

The IPB software includes code for compiler use as well as run-time set up of the hardware before image processing can start, in the same way as the SDB software. The protocols are again defined before the function call.

```
{{{F ipb software
...  protocols
PROC ipb(CHAN OF commands to.ipb, CHAN OF transmit from.ipb,
         CHAN OF ANY to.ipb.1, from.ipb.1)
  ...  setup
  ...  vars
  SEQ
    ...  set up ipb
    ...  main program loop
:
}}}
```

5.4.4.1 Compiler set-up information

The "setup" fold, below, states the use of the TDS library "ioconv" [65] and defines the values to initialise the hardware register "o.port". The frame buffers for the images are arranged as four 512 x 512 byte images with starting address #30 000 000. Colours are defined to be the values between 248 and 255. Values used in the image processing software are defined before the fold "char stuff" which holds the character set for writing text on the images (as used for "Sample number") and the software associated with this.

```
{{{  libs
#USE ioconv
}}}
{{{  ipb vals, vars and placements
VAL INT o.port.val IS #07: -- 7, 6, 5 not used = 0;
-- 4 bank.select = 0(frames 0 / 1);  3 = 0; 2 user.led = 1 (inactive);
-- 1 ss.analyse = 1 (inactive); 0 ss.reset = 1 (inactive)
[4][512][512]BYTE image:
PLACE image AT #30000000:
INT o.port, i.port, frame.no, frame.count:
PLACE o.port AT #20000000:
PLACE i.port AT #28000000:
...  set colours    -     -     -     -     -
}}}
...  char stuff
{{{  vals
VAL INT resolution IS 512:
VAL INT centre     IS 256:
VAL m.scaler       IS 1000:
VAL min.chip.size  IS 4:
VAL min.chip.lines IS 3:
}}}
...  least squares fit
}}}
```

```
{{{  set colours
VAL BYTE WHITE  IS 255(BYTE):
VAL BYTE RED    IS 254(BYTE):
VAL BYTE BROWN  IS 253(BYTE):
VAL BYTE GREEN  IS 252(BYTE):
VAL BYTE CYAN   IS 251(BYTE):
VAL BYTE BLUE   IS 250(BYTE):
VAL BYTE PURPLE IS 249(BYTE):
VAL BYTE BLACK  IS 248(BYTE):
}}}
```

(1) The least squares fit algorithm

A function to least squares fit points to a straight line and give its slope and offset is shown below. The method of least squares fit is detailed in Thomas and Finney [36] and is coded into Occam below. The function requires: an array of points, a start point and an end point, between which the fit is made. It returns the slope, m, and offset, c, of the equation of the straight line which best fits the points in the form of $y = m x + c$.

```
PROC least.square.fit([]INT line,VAL INT left.x, right.x, INT m, c)
  INT n:
  INT64 x.sum,y.sum,x.squared.sum,xy.sum:
  SEQ
    ...  init vars  -     -     -     -     -
    n := right.x - left.x
    SEQ x = left.x FOR n
      SEQ
        x.sum := x.sum + (INT64(x))
        y.sum := y.sum + (INT64(line[x]))
        x.squared.sum := x.squared.sum + (INT64(x * x))
        xy.sum := xy.sum + (INT64(line[x] * x) )
    {{{  calc c and m
    SEQ
      c := INT( ( (x.sum * xy.sum) - (x.squared.sum * y.sum) ) /
                ( (x.sum * x.sum) - (x.squared.sum * (INT64(n))) ) )
      m := (((INT(y.sum)) - (n * c)) * m.scaler) / (INT(x.sum))
    }}}
:
```

```
{{{  init vars
SEQ
  x.sum := 0(INT64)
  y.sum := 0(INT64)
  x.squared.sum := 0(INT64)
  xy.sum := 0(INT64)
}}}
```

5.4.4.2 Run-time set-up of the IPB

Variables are initialised and the register "o.port" is set before writing to each pixel to clear the images.

```
SEQ
  frame.count := 0
  specimen.no := 0
  error := 0
  o.port := o.port.val
  o.port := (0<<8)\/( o.port/\#FF )
  ...  clear all images
```

5.4.4.3 Main program

A continuous loop waits for communication on the channel "`to.ipb`" for instruction of what action to perform. During normal operation the IPB will be instructed to capture a frame and then process it. The "`*.request`" actions allow the current data and images to be saved to the host IBM-AT for logging and post analysis.

```
{{{  main program loop
WHILE TRUE
  SEQ
    error := 0
    to.ipb ? CASE
      capture; frame.no
        ...  capture
      display; frame.no
        ...  display
      process; frame.no
        ...  image processing
      i.request; junk
        ...  send image in current frame
      r.request; junk
        ...  send results
      b.request; junk
        ...  send results and images
      specimen; specimen.no
        ...  get specimen.no
}}}
```

(1) Capture

The capture sequence is mainly controlled by the SDB, but the IPB must select the correct buffer into which the image is to be captured. This is achieved by programming the "o.port" register.

```
{{{  capture
SEQ
  {{{  swap banks
  IF
    (frame.no /\ #01) = 0          -- ie frame 0 or 1
      o.port := (o.port /\ #FFEF) \/ #0004  -- 4led on
    TRUE                           -- ie frame 2 or 3
      o.port := (o.port \/ #0010) /\ #FFFB  -- Bled off
  }}}
}}}
```

(2) Image Processing

The image processing software can be considered as two parts: firstly finding the location points and feature points of the core to perform the inspection, and secondly drawing the lines and text back on to the image for display. For an on-line system in a factory, the display would not normally be required and not performed except when requested during setting up or checking the system.

```
{{{  image processing
...  Vals and vars
SEQ
  ...  set up
  ...  find centre points
  ...  find edge features
  ...  calc inductor core radii                    } Inspection
  ...  search edges
  ...  find equations of the flange outer edges
  ...  compare with edges to find chips
  ...  write specimen number
  ...  draw on results                             } Display
  ...  write on error
  ...  send results to sdb
}}}
```

(a) Finding the centre point

The algorithm to find the centre point of the inductor core uses the knowledge that the core's central pillar will lie over the centre of the image. The intensity histograms, see image5.2, of x = 256 and y = 256 are viewed and the points of steepest gradient with the expected sign of slope are taken to be the edges of the circle. Finding an edge is shown in the opened fold "`... look for horizontal circle edges`", where finding the edge "`inner.circle.left.x`" is demonstrated. This algorithm is applied similarly for each edge point. Provided the edges have a slope which is greater than that expected, the centre is found by applying the knowledge that the bisecting point of the cord between the two points on the edges gives the centre location in that given direction. This is shown in the opened fold "`... calc centre points`".

```
{{{  find centre points
SEQ
  {{{  look for horizontal circle edges
  SEQ
    {{{  find inner.circle.left.x
    INT x,intensity.diff,max.intensity.diff:
    SEQ
      max.intensity.diff := 0
      x := centre - 6
      WHILE x > (centre - 80)
        SEQ
          intensity.diff:=(INT(image[frame.no][centre][(x-2)]))-
                          (INT(image[frame.no][centre][(x + 2)]))
          IF
            intensity.diff > max.intensity.diff
              SEQ
                max.intensity.diff := intensity.diff
                inner.circle.left.x := x
            TRUE
              SKIP
          x := x - 1
      ...  is this a good edge  -    -    -    -
    }}}
    ...  find outer.circle.left.x
    ...  find inner.circle.right.x
    ...  find outer.circle.right.x
  }}}

  ...  look for vertical circle edges
  {{{  calc centre points
  SEQ
    inner.centre.x := i.c.l.x + ((i.c.r.x - i.c.l.x) / 2)
    inner.centre.y := i.c.u.y + ((i.c.l.y - i.c.u.y) / 2)
    outer.centre.x := o.c.l.x + ((o.c.r.x - o.c.l.x) / 2)
    outer.centre.y := o.c.u.y + ((o.c.l.y - o.c.u.y) / 2)
    centre.x := inner.centre.x - ((inner.centre.x - outer.centre.x) / 2)
    centre.y := inner.centre.y - ((inner.centre.y - outer.centre.y) / 2)
  }}}

}}}
```

```
{{{  is this a good edge
IF
  max.intensity.diff < 20
    error := 1
  TRUE
    SKIP
}}}
```

(b) Find edge features

The algorithm to find the centre of the circle is re-applied to the image, this time looking at the intensity histograms for the lines passing through the centre of the core. These histograms give the points on the edges for each core feature edge.

```
{{{  find edge features
SEQ
  {{{  look for horizontal circle edges
  SEQ
    {{{  find inner.circle.left.x
    INT x,intensity.diff,max.intensity.diff:
    SEQ
      max.intensity.diff := 0
      x := centre - 6
      WHILE (x > (centre.x - 50)) AND ( x > 10)
        SEQ
          intensity.diff := (INT(image[frame.no][centre.y][(x - 2)]))-
                            (INT(image[frame.no][centre.y][(x + 2)]))
          IF
            intensity.diff > max.intensity.diff
              SEQ
                max.intensity.diff := intensity.diff
                inner.circle.left.x := x
            TRUE
              SKIP
          x := x - 1
      ...  is this a good edge  -     -     -     -
    }}}
    ...  find outer.circle.left.x
    ...  find inner.flange.left.x
    ...  find outer.flange.left.x
    ...  find inner.circle.right.x
    ...  find outer.circle.right.x
    ...  find inner.flange.right.x
    ...  find outer.flange.right.x
  }}}
  ...  look for vertical circle edges
}}}
```

```
{{{  is this a good edge
IF
  max.intensity.diff<20
    error := 1
  TRUE
    SKIP
}}}
```

(c) Calculate inductor core radii

The radii for both inner and outer edges (circles) of the central pillar are found in the x and y directions independently from the diameters. This allows the roundness of the central pillar to be checked and shows whether the shrinkage of the core is uniform. The radii of the flanges are calculated from the centre of the core.

```
SEQ
  inner.x.radius          := ((inner.circle.right.x - inner.circle.left.x)  / 2)
  inner.y.radius          := ((inner.circle.lower.y - inner.circle.upper.y) / 2)
  outer.x.radius          := ((outer.circle.right.x - outer.circle.left.x)  / 2)
  outer.y.radius          := ((outer.circle.lower.y - outer.circle.upper.y) / 2)
  left.flange.x.radius    :=  centre.x              - inner.flange.left.x
  right.flange.x.radius  := inner.flange.right.x   - centre.x
```

(d) Search edges

The edge feature points, found in (b) find edge features, are used as the starting locations to search for the complete edges of the features. A limited search centred on the current edge location is performed on the next line to find the position of highest gradient which is taken to be the edge location of the feature at this line. For the circles the edges are searched in x or y directions for the upper and lower, or left and right features respectively.

```
SEQ
  {{{  search flange edges
  SEQ
    {{{  search left flange lower outer edge
    ...  vars -     -     -     -     -     -     -
    SEQ
      no.edge := FALSE
      last.x := outer.flange.left.x
      y := centre.y
      WHILE ((y < (centre.y + 110)) AND (no.edge = FALSE))
        SEQ
```

```
{{{  vars
INT max.intensity.diff:
INT max.x,last.x, y:
BOOL no.edge:
}}}
```

continued over page ...

```
        {{{  find steepest gradient at this point
        INT x,intensity.diff:
        SEQ
          max.intensity.diff := 0
          SEQ x = (last.x - 4) FOR 12
            SEQ
              intensity.diff := (INT(image[frame.no][y][(x - 2)])) -
                                     (INT(image[frame.no][y][(x + 2)]))
              IF
                intensity.diff > max.intensity.diff
                  SEQ
                    max.intensity.diff := intensity.diff
                    max.x := x
                TRUE
                  SKIP
        }}}
        ...  check max.x-    -    -    -    -
        left.flange.outer.edge[y] := max.x
        ...  is this a good edge
        y := y + 1
  }}}
  ...  search left flange upper outer edge
  ...  search right flange lower outer edge
  ...  search right flange upper outer edge
  ...  search left flange lower inner edge
  ...  search left flange upper inner edge
  ...  search right flange lower inner edge
  ...  search right flange upper inner edge
}}}
...  search circle edges
```

```
{{{  check max.x
IF
  max.x < 10
    SEQ
      last.x := 10
      max.x  := 10
  max.x > 500
    SEQ
      last.x := 500
      max.x  := 500
  TRUE
    last.x := max.x
}}}
```

(e) Find the equations of the flange outer edges

An algorithm which applies the least squares fit principle, Thomas and Finney [36], detailed in section5.4.4.1(1), is used over the outer edges of the flanges to give the equations of their edges.

```
SEQ
  least.square.fit(left.flange.outer.edge,(centre.y+50), (centre.y+100),
                         left.flange.lower.m, left.flange.lower.c)

  least.square.fit(right.flange.outer.edge, (centre.y+50), (centre.y+100),
                         right.flange.lower.m, right.flange.lower.c)

  least.square.fit(left.flange.outer.edge,(centre.y-100), (centre.y-50),
                         left.flange.upper.m, left.flange.upper.c)

  least.square.fit(right.flange.outer.edge, (centre.y-100), (centre.y-50),
                         right.flange.upper.m, right.flange.upper.c)

  least.square.fit(left.flange.outer.edge,(centre.y-10), (centre.y+10),
                         left.flange.middle.m, left.flange.middle.c)

  least.square.fit(right.flange.outer.edge, (centre.y-10), (centre.y+10),
                         right.flange.middle.m, right.flange.middle.c)
```

(f) Find edge chips

The edge chips are found by comparing the searched edges, found in section (d), with those expected from the radii calculations and the equations of the circles. Only if the searched edge falls inside the expected feature's edge does it constitute a possible problem. Chips are only deemed to be significant if they are greater than a prescribed length and width.

```
SEQ
  {{{  inner circle
  ...  vars    -    -    -    -    -
  SEQ
    x.centre := centre.x
    y.centre := centre.y
    radius := ( (inner.x.radius * cal.x) + (inner.y.radius * cal.y) ) / 2
    x.radius.cos.45 := (((inner.x.radius*cal.x) * 70711) / 100000)
    y.radius.cos.45 := (((inner.y.radius*cal.y) * 70711) / 100000)
    radius.sqrd := radius * radius
```

```
{{{  vars
INT x.radius.cos.45,y.radius.cos.45
INT radius.sqrd:
INT radius,x.centre,y.centre:
}}}
```

```
    {{{  draw top and bottom quarter
    ...  vars -    -    -    -    -    -
    SEQ
      y := radius
      last.y := radius
      left.upper.chip.count  := 0
      left.lower.chip.count  := 0
      right.upper.chip.count := 0
      right.lower.chip.count := 0
      SEQ x = 0 FOR x.radius.cos.45
        SEQ
          y.sqrd := (radius.sqrd - ( x  * x )  )
          ...  check sign
          ...  calc square root of y
          ...  calc points on circle
          ...  check points
          ...  draw circle points on image
          ...  draw upper chips on image
          ...  draw lower chips on image
    }}}
    ...  draw left and right quarter
  }}}
  ...  outer circle
  ...  left flange arc
  ...  right flange arc
  ...  flange outer edges
```

```
{{{  vars
INT ,y.circle.1,y.circle.2:
INT last.y,y,y.sqrd:
INT x.circle.1,x.circle.2:
INT l.up.chip.cnt:
INT l.low.chip.cnt:
INT r.up.chip.cnt:
INT r.low.chip.cnt:
}}}
```

If the chip is deeper than this value a counter records how long this significant chip is in the fold "... does the chip fail". If this counter is longer than the permitted maximum, the core fails. The length counter is reset to zero if the edge falls below the maximum width.

```
{{{  draw on chips on right
SEQ
  y1 := inner.circle.upper.right.edge[(x.circle.1 - x.centre)]
  IF
    y1 < (y.circle.2 - min.chip.size)
      SEQ
        SEQ y = y1 FOR ((y.circle.2 - y1) + 1)
          image[frame.no][y][x.circle.1] := BLUE
        r.up.chip.cnt := r.up.chip.cnt + 1
    TRUE
      r.up.chip.cnt := 0
  ...  does chip fail    -    -
}}}
```

```
{{{  does chip fail
IF
  r.up.chip.cnt>min.chip.lines
    chipped := TRUE
  TRUE
    SKIP
}}}
```

(g) Draw results onto the image

A point or line may be drawn onto the image by setting the pixel to a determined value. This is achieved in the software by setting the location in the array "image[frame.no][y][x]" to the prescribed value. This would not be necessary for fully automated running, but in order to demonstrate what the software is doing at any time, drawing the results on to the image clearly shows where the edges have been found and where any chips exist. Two examples of the software which draw results on to the image are shown below. These examples are: the intensity profile (or histogram) which fills up the bottom of the image to the value of the intensity, as shown in image5.2, and the drawing of the left and right edges of the central pillar in which the circular edges are drawn.

```
SEQ
  ...  draw horz intensity profile at middle of the image
  ...  draw on circle location lines
  ...  draw on centres
  ...  draw on borders
  ...  draw on edges
  ...  draw on inner circle edges
  ...  draw on outer circle edges
  ...  draw on outer.flange template lines
```

(i) "draw horz intensity profile at middle of the image"

```
INT this.y,y:
SEQ x = 0 FOR 512
  SEQ
    this.y := 511 - (INT(image[frame.no][centre][x]))
    y := 511
    WHILE y > this.y
      SEQ
        IF
          (y \ 32) = 0
            image[frame.no][(y)][x]  := 100(BYTE)
          TRUE
            image[frame.no][(y)][x]  := 0(BYTE)
        y := y - 1
```

(ii) "draw left and right edges"

```
SEQ count = 0 FOR y.radius.cos.45
  SEQ
    x := inner.circle.left.lower.edge[count]
    image[frame.no][centre.y + count][x] := GREEN
    x := inner.circle.left.upper.edge[count]
    image[frame.no][centre.y - count][x] := GREEN
    x := inner.circle.right.lower.edge[count]
    image[frame.no][centre.y + count][x] := GREEN
    x := inner.circle.right.upper.edge[count]
    image[frame.no][centre.y - count][x] := GREEN
```

(h) Prepare results

Transmission of the results from each core inspection to the SDB is performed via the channel "from.ipb". Since a limit of 20 values is imposed on the channel protocol, by the Occam compiler, the variables are packed into a single array as follows.

```
SEQ
  results[0]  := inner.circle.left.x
  results[1]  := inner.circle.right.x
      •
      •
      •
  results[20] := error
  results[21] := specimen.no
  from.ipb ! r.transmit; results
```

5.5 Calibration of the system

Both the x and y directions of the image were calibrated separately on the development system (using the PIP framestore), as well as on TIPS. The development system had a 4 : 3 aspect ratio for each pixel and demanded separate calibration scaling factors for each direction, but it was decided to keep the two dimensional calibration in order to check that the square pixels on TIPS are actually square.

The calibration method chosen, involves a similar process to that used for measurement discussed above. It adopts the same optical arrangement, measuring a core of known size which then calculates the calibration factors as below.

$$\text{Calibration factor} = \frac{\text{Number of pixels between edges}}{\text{Measured length}}$$

Since discrete integer mathematics is employed and numbers are truncated, a scaling factor of 1000 is incorporated so that a loss in resolution is not incurred.

$$\text{Calibration factor} = \frac{(\text{Number of pixels between edges} \quad \text{x} \quad 1000)}{\text{Measured length}}$$

Although this method proves adequate, it requires careful positioning to locate the standard core over the centre of the image and in line with the pixel orientation. An omni-directional calibration sample would be better, as it permits any orientation to be used. A section of circular bar is used which is placed in the image such that it lies approximately over the centre of the image. The algorithm to find the centre of the inductor core is used to find the centre of the bar from which the diameters are then measured and the system calibrated.

i Find edges of the bar
ii Find centre of the circle
iii Find edges vertically and horizontally from centre
iv Calculate calibration factors

The equation for the calibration factors is the same as that used for the standard core, with a scaling factor of 1000.

5.6 Results from trials

Nine ferroxcube inductor core samples were supplied by Philips, selected from a typical production batch, and used to test the software and hardware of the system. These included specimen cores which were broken and chipped to varying degrees as well as a passable core. All of the dimensions on all unbroken samples supplied were within the set tolerances for measurement.

5.6.1 The tests performed

A number of tests were devised to evaluate the hardware and software to ensure several operational factors.

- ❑ Reliability
- ❑ Accuracy
- ❑ Consistency
- ❑ Durability
- ❑ Absence of core
- ❑ Illumination failure

These factors were chosen to ensure that the combination of hardware and software ran without failure, measured and inspected the inductor cores with sufficient accuracy over long periods and would not produce incorrect results if parts of the system failed or unexpected images were captured.

The reliability of the system against failure through hardware or software design fault was performed by "soak testing" the system to see if it stopped. Inspection was at full rate, daily, over a number of weeks. Since each inspection cycle takes about 0.5 seconds, in any eight hour day the system was tested over 56 000 times and after a fortnight, over $^1/_2$ million times. During these trial periods the system was not touched or operated and the specimen core under inspection was not moved.

Accuracy and consistency were tested together by sampling each specimen core over 100 times at various locations in the field of view, within the specified area defined for accurate measurement to take place. The results of these trials were logged and shown in section 5.9.

The durability of the software was assured by continuously inspecting a core while moving the core around, and in and out of, the field of view. This test was performed by hand ensuring that the positions where the core lay on the limits of the inspection area, or where it was partially out of the field of view, were thoroughly examined. This was a qualitive test and provided the system did not fail, or produce a pass for a substandard core the system would pass this test.

The system had to withstand a blank field of view and not register this as a passable core. In addition to the blank field, illumination failure causes incorrect lighting, either from one side, or no lighting at all. The system had to ensure that cores were not passed if any of these happened.

5.6.2 Manual measurement to validate the results from the system

An accurate method of measuring the cores was required to validate the results of the inspection by the system. Initially, each core was measured using a micrometer screw gauge and vernier callipers. However, some dimensions were unmeasurable by this means or could not measured to the desired accuracy. Consequently, a vernier travelling microscope was used to measure all of the dimensions to an accuracy of 0.1 mm. Normal precautions were observed when recording the measurements in order to account for any slack in the travel of the microscope and care was taken to measure to as near to the same position on each edge as possible. The edges were uneven and varied by about 0.05 mm along their length. Consequently the measurements were not made to a higher precision and the dimensions were measured to the average position as judged by the eye. An example of the fluctuations along a typical edge is shown in plate 5.2 which depicts the "outer.circle.right.edge" of specimen "5". This plate shows a field of view of 8 mm by 6 mm.

5.6.3 The automatic measurement and inspection by the system

Results were taken for each specimen showing the discrete pixel locations of each feature point as well as the measurements, in millimetres, of the desired dimensions. Each was then repeated one hundred times for each specimen while it was held stationary and in the same position in the camera's field of view.

The results of pixel location were recorded to show the variation in pixels where the software found each edge, with the dimension measurements given to indicate the accuracy to which the dimensions are made, and the affect on them by the pixel location fluctuations. The measurements are given one order of magnitude greater than the system can measure to show the degree of precision and the fluctuations of the values. The results were sampled one hundred times with the core in the same position to highlight any system inconsistencies. In addition to these tests, two of the specimens, 1 and 9, were sampled at the limits of the position space in the image to show whether the location significantly affected the results.

5.6.4 Accuracy of the measurements (cf. travelling microscope)

The measurements are calculated as described in section 5.1.3 and shown in the description of the software in section 5.4.3.3(2)(b) . The fluctuation in measurements shown in section 5.9 is caused by the variations in finding the pixel location.

5.6.4.1 Core specimens whose measurement results are disregarded

The results for specimens "4" and "8" show an error code of "2", indicating that these cores are broken. It is clear from the images that the values given for some of the pixel locations and the measurements are invalid. Since it is likely for some of the samples the centre locations and edge locations are not correct, the measurements have not been included in the graphs of the data at the end of this chapter.

The measurements for specimen "6" has also been disregarded since the chipped flange is so large as to affect the lighting of the central pillar in many of the samples. Image 5.19 shows a sample which has found the location of the point "`outer.circle.upper.y`" above the actual location and on the edge of the core base. This occurs because this has the sharpest intensity gradient when the flange is not present to shadow this feature.

5.6.4.2 Analysis of valid measurements

Most pixel location points show a variation of two or three locations centred around a point which is preferred. If the steepest edge falls between two pixels, both these values have similar frequencies of occurrence and the slight variations between captured frames can cause the system to find the feature location in a range of pixel points.

(1) Variation in pixel point location

The results, given in section 5.9 at the end of this chapter, show the frequency of occurrence of each pixel location recorded for each edge feature. These data results correspond to the digital picture of the core, in location on the image, with the feature and edge lines drawn on them. The raw, or unprocessed, image corresponding to this is also shown for comparison.

Analysis of the results reveals that the measurements in the "y" direction show a greater pixel point variation with a consequential variation in measurements. This larger variation can be attributed to the shape of the core requiring a much more precise positioning in the frame to allow the lighting to illuminate the core as required. If the core is not correctly aligned the lighting does not differentiate the central pillar from the core base as well as it does the core base edge with the background. The result is that the software finds the "`outer.circle.upper.edge`" or "`outer.circle.lower.edge`" on the edge of the core base and not on the central pillar.

This causes the centre location of the core in the y direction to be displaced and affects the results accordingly. This has no effect on the x direction measurements.

Viewing the graphs of x direction results shows a resolution of ± 0.18 mm if the worst case results are included. However, analysis of the results for specimen "9", the un-chipped core, shows that the measurements generally vary by up to one pixel in the x direction giving an error of less than 0.1 mm.

5.6.5 The inspection for chipped edges

The definition for a chipped edge is given in sections 5.3.5 and 5.4.4.3(2)(f). The results were taken with the chip length along the edge set to 3 pixels (0.13 mm) and the width set to 4 pixels (0.16 mm), which can be varied by the operator to meet the specification required.

Variation of the image had a lesser effect with this part of the inspection and cores which were chipped were failed most of the time. Sample 9, which had no significant chips, passed the inspection on nearly 100% of the samples, except on one of the tests where it was near to the limit of the position in the frame, the pass rate dropped to 97%.

All of the specimens and every test had the same values for chip size rejection.

5.7 Conclusions and future developments

The system (hardware and software) performs the tasks of measurement and inspection to accuracy of better than 0.1 mm on un-chipped samples in the x direction. The system offers a more consistent success rate for passing cores which should be passed and failing those which should be failed than that which a human inspection can achieve and avoids the quality assurance problems of "five o'clock" or "Friday afternoon" component inspection. In Lilley *et al* [73] when discussing the inspection of solder joints by human inspection, declare a less than 20% agreement between inspectors viewing the same joint. In addition to improved inspection and consistency, the system offers inspection at a rate of 7200 samples per hour, which is a major improvement on the human inspection performance.

The system could be extended to measure cores lying at any angle in the image. Removal of the need to guarantee the orientation of the cores as they pass beneath the camera would allow the manufacturing system more flexibility and reduce the cost of the overall inspection process. Better resolution of the camera and of the image processing system would enable much of the variation in results to fall well below the resolution required by the inspection. The results presented in this chapter show that the measurements are performed to an accuracy of 0.1 mm, over 99.9 % of the time with only a single pixel difference each side of the correct result being the most common error. Consequently a resolution increase of the system, by doubling it to 1024 by 1024, would eliminate nearly all erroneous results. Processing an image of this size would, of course, take either four times as long or require four times as much processing to complete the same task in the same time as has been achieved for the 512 by 512 image.

Since a "single shot" approach is required, where the position is found from a single image, the method of multiple sampling to find an average of the variations over any number of samples and calculating the best location (which may be between two pixels) is not discussed further in this thesis.

The pixel location points were taken from a single image. However, if all the points along an edge were averaged to give a value, this variation could be reduced. Combination of the measurement algorithms with those searching for chipped edges could allow the "bad" values in chips to be removed from the calculation. But, since a chipped sample fails the test, it need not be measured further and time can be saved by rejecting the core if it fails the chip inspection before it is measured.

Image processing is set for great advance in the future and will certainly become widespread throughout industry as an automatic measurement and inspection tool. The combination of TIPS with this software demonstrates that the potential of performing visual inspection and measurement using image processing in on-line applications is now entirely possible.

Complete Occam source listing of the software is available on request to The University of Liverpool.

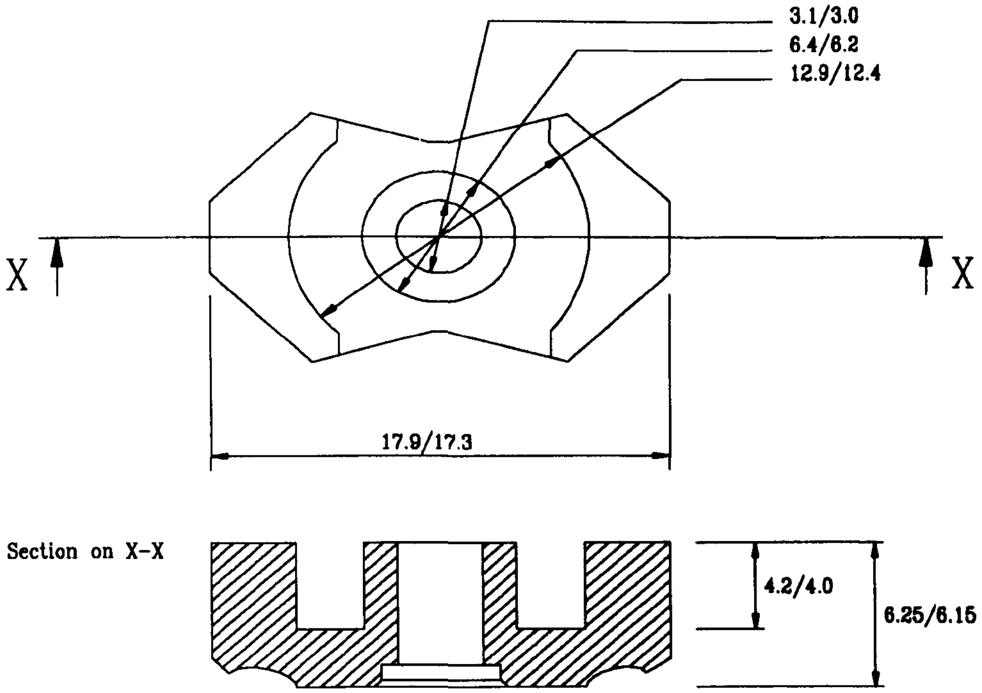


Figure 5.1 Upper and lower dimensions of an RM6-S Ferroxcube inductor core

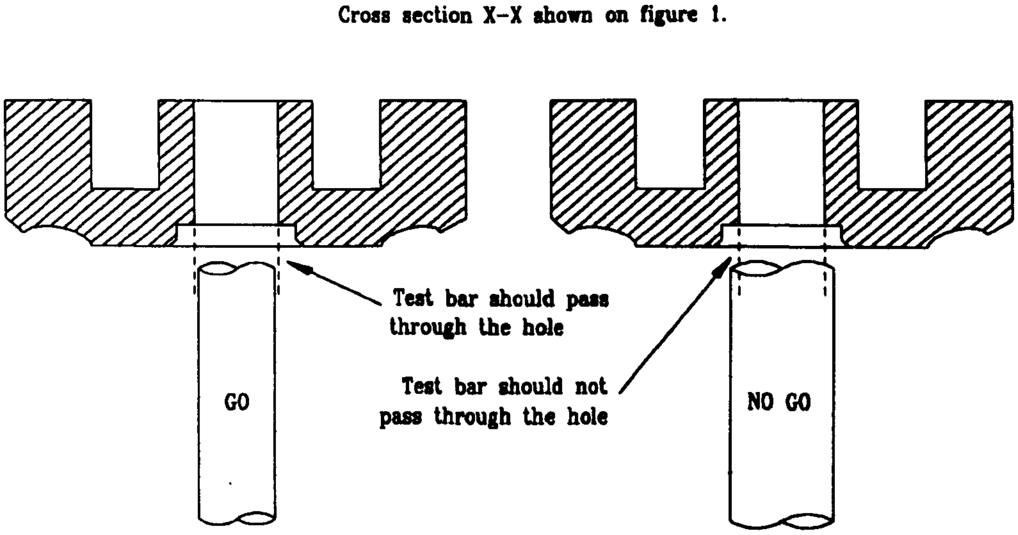

Cross section X-X shown on figure 1.
Test bar should pass through the hole
Test bar should not pass through the hole
GO
NO GO

Figure 5.2 "Go / no-go" test of the hole in the central pillar

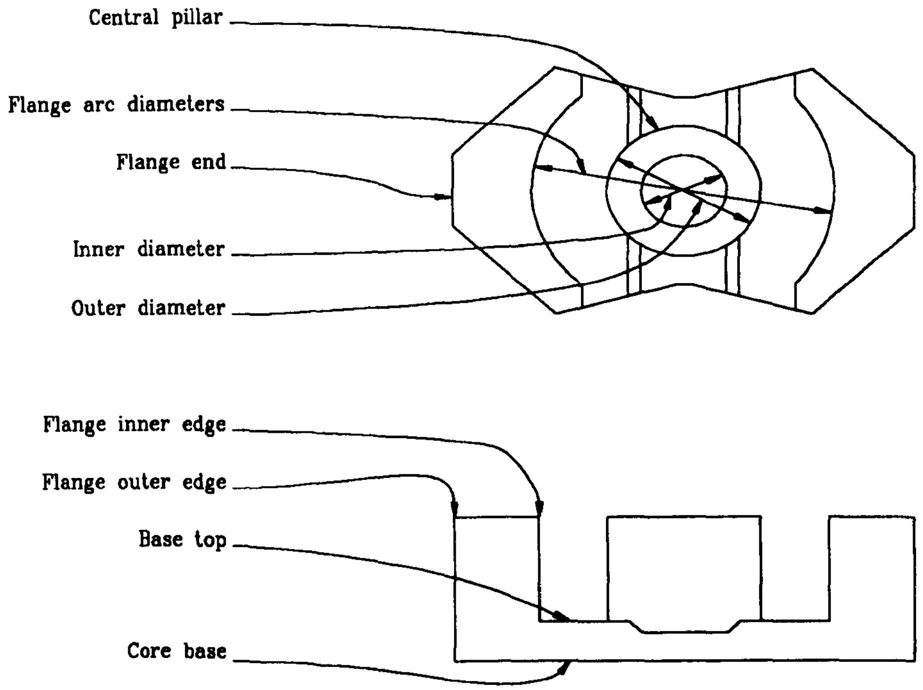


Figure 5.3 Inspection features of an inductor core

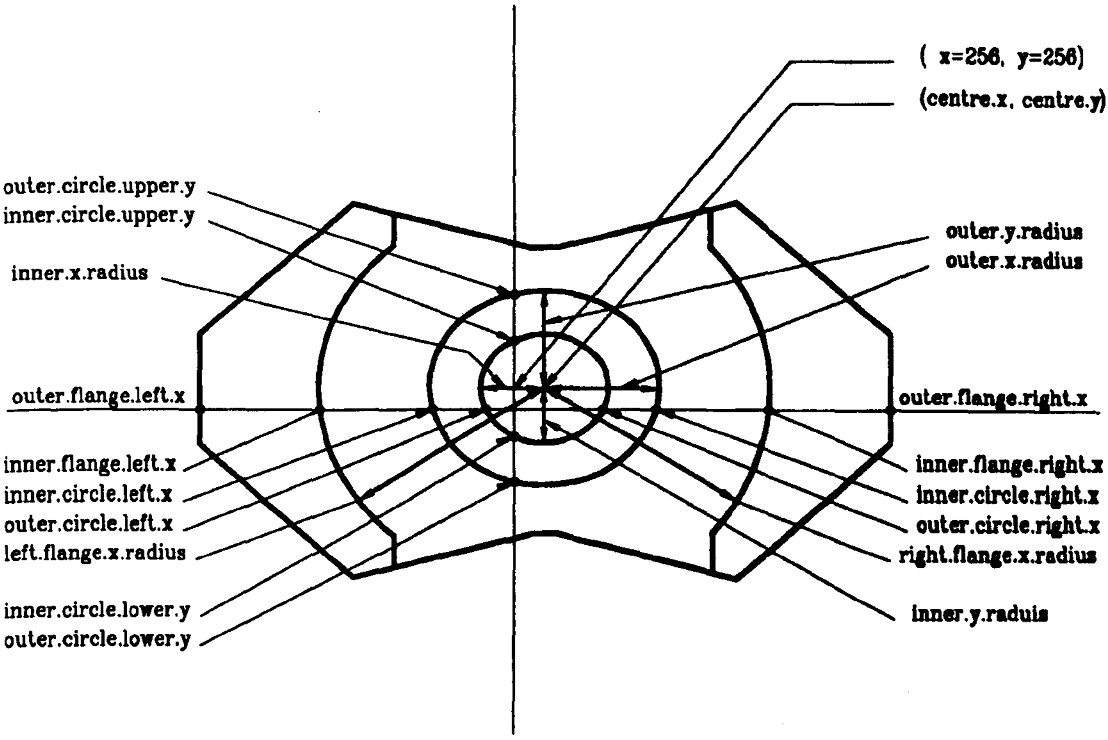


Figure 5.4 Location point and feature names as used in the software

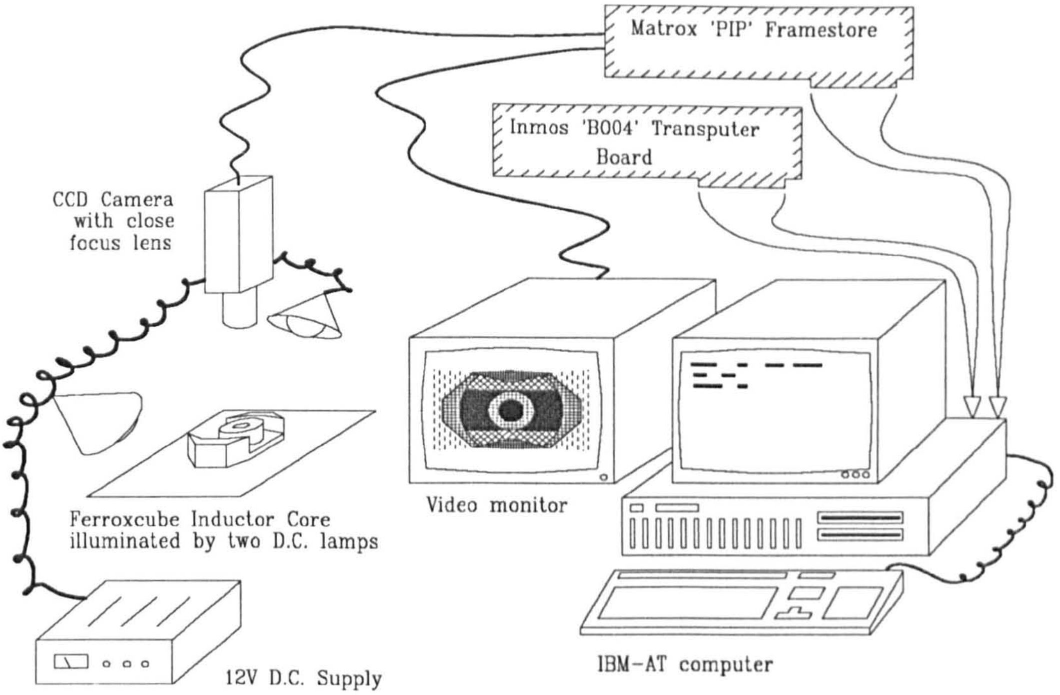

Matrox 'PIP' Framestore
Inmos 'B004' Transputer Board
CCD Camera with close focus lens
Ferroxcube Inductor Core illuminated by two D.C. lamps
Video monitor
IBM-AT computer
12V D.C. Supply

Figure 5.5 Development system used to model The Transputer Image Processing System (TIPS)

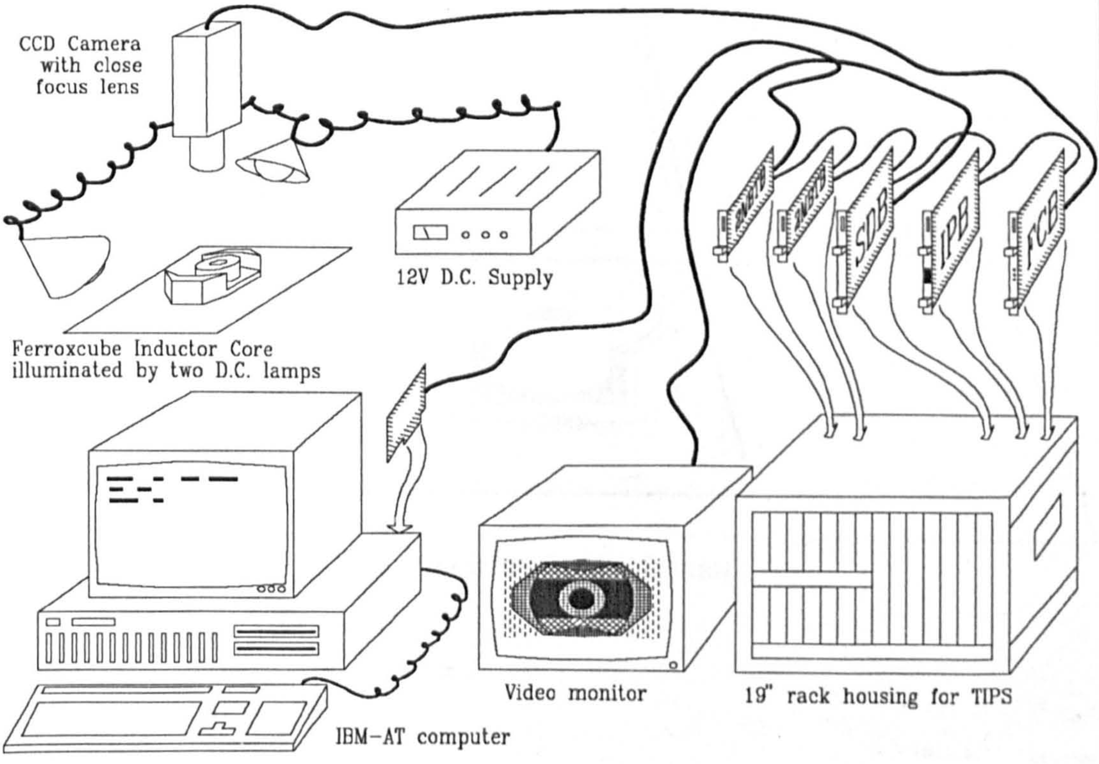

CCD Camera with close focus lens
12V D.C. Supply
Ferroxcube Inductor Core illuminated by two D.C. lamps
SDB
IPB
FCB
Video monitor
19" rack housing for TIPS
IBM-AT computer

Figure 5.6 Implementation on TIPS

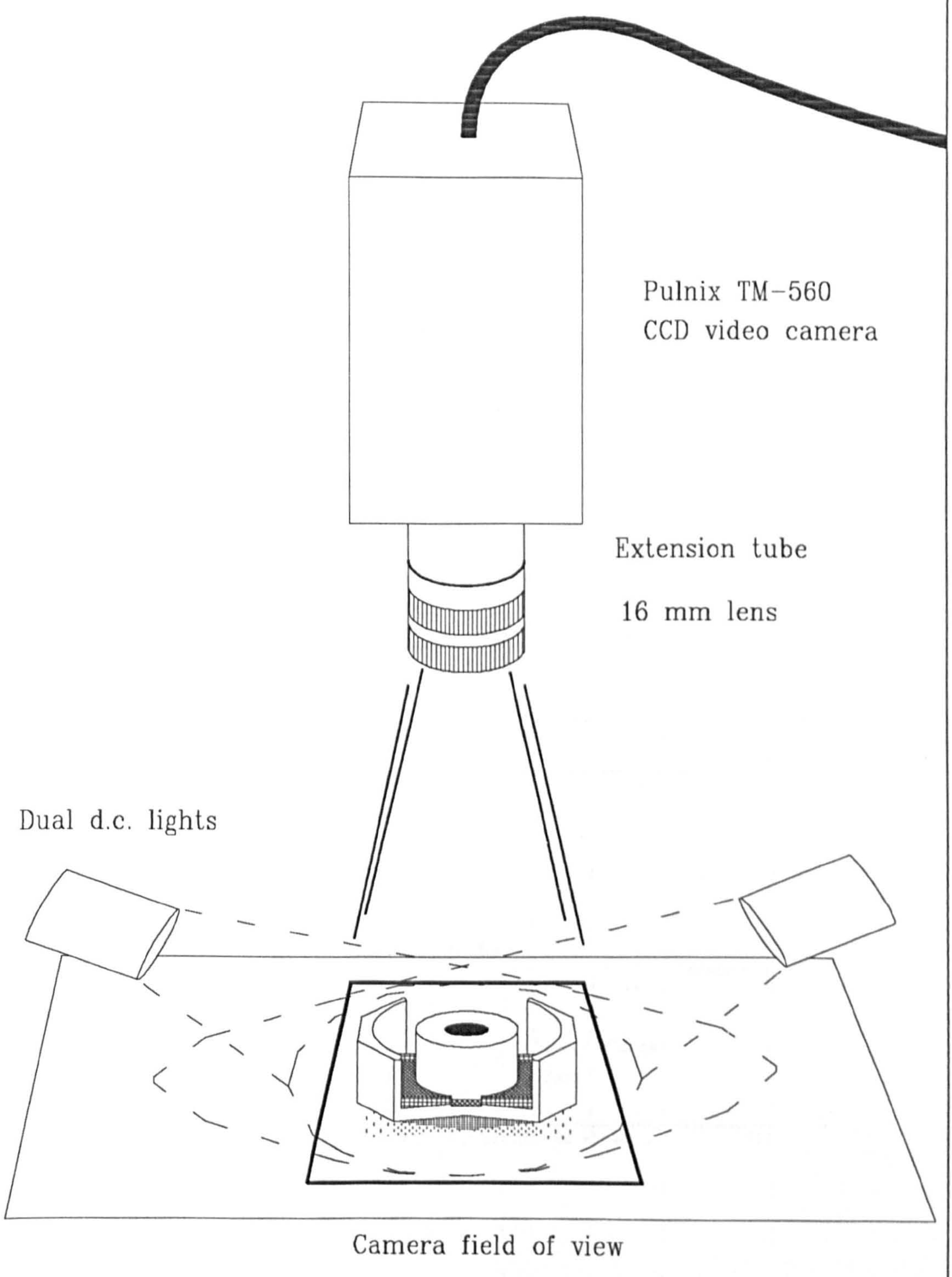


Figure 5.7 The illumination and optical arrangement for inspection with two lamps and the camera mounted vertically above the core

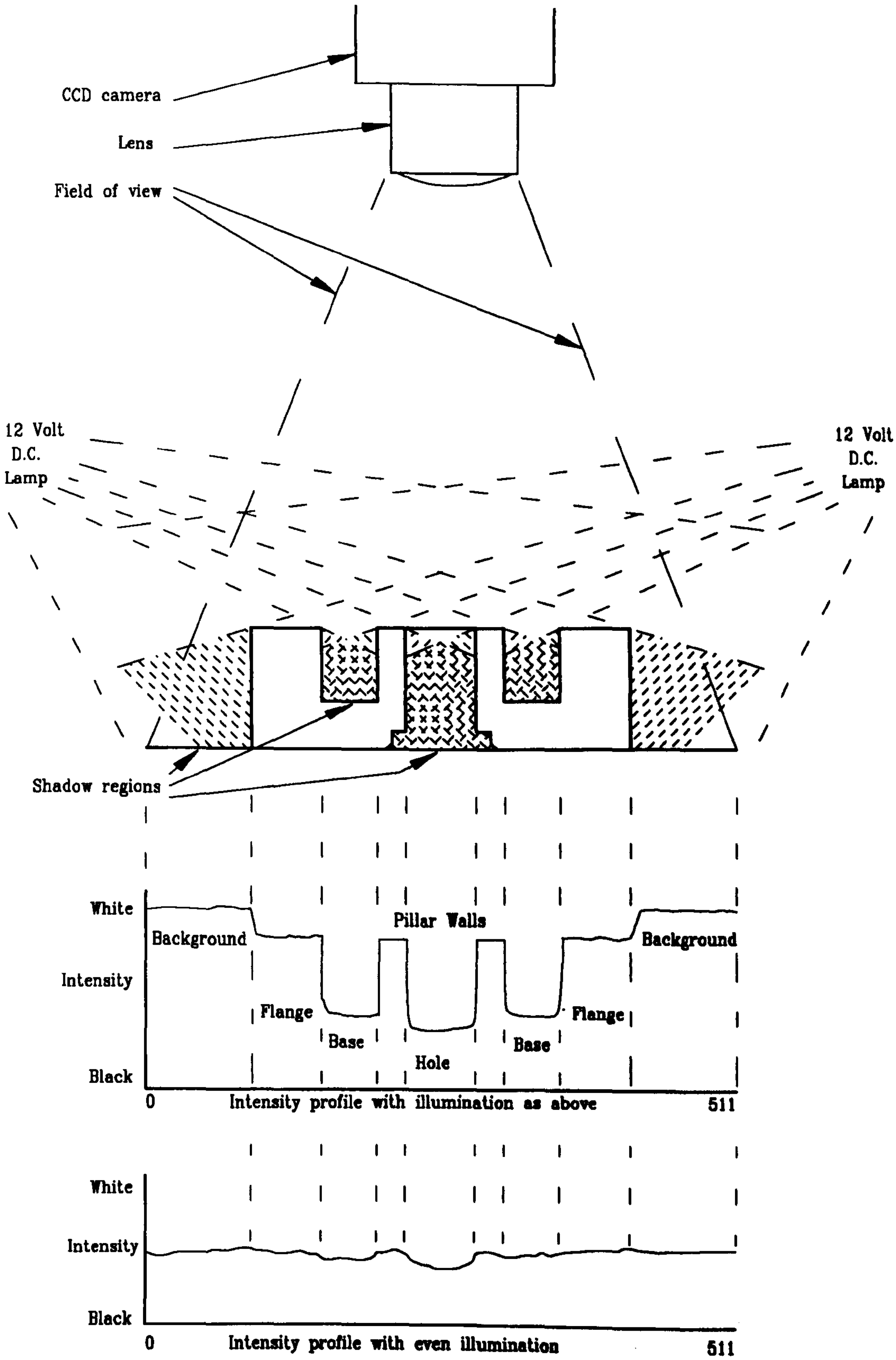


Figure 5.8 Intensity histograms across the centre of a core with even and controlled illumination

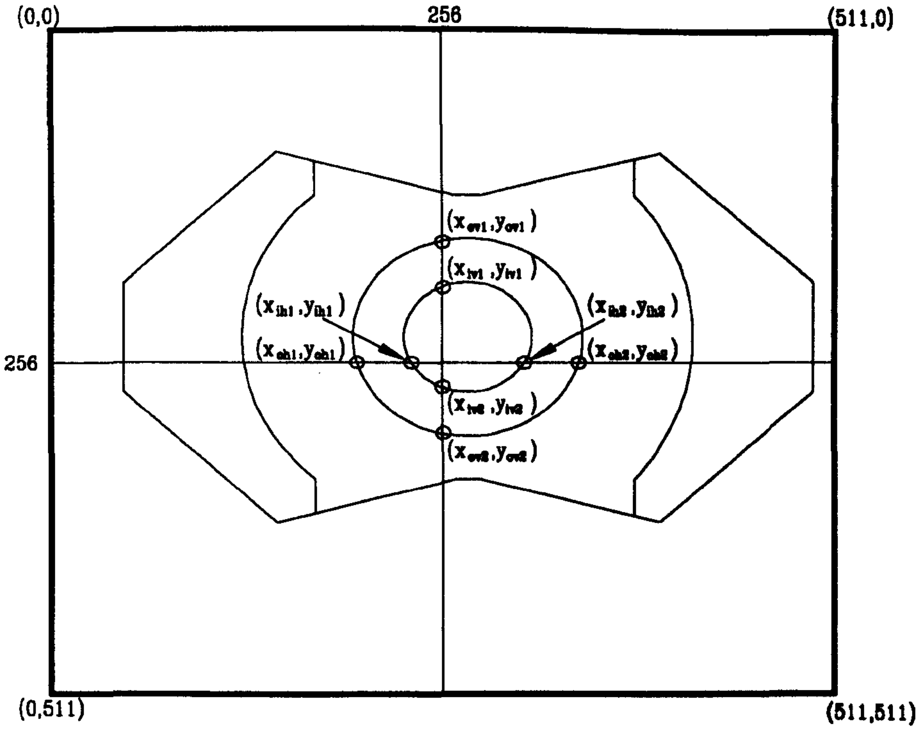


Figure 5.9 Coordinates used to calculate the position of the core in the image

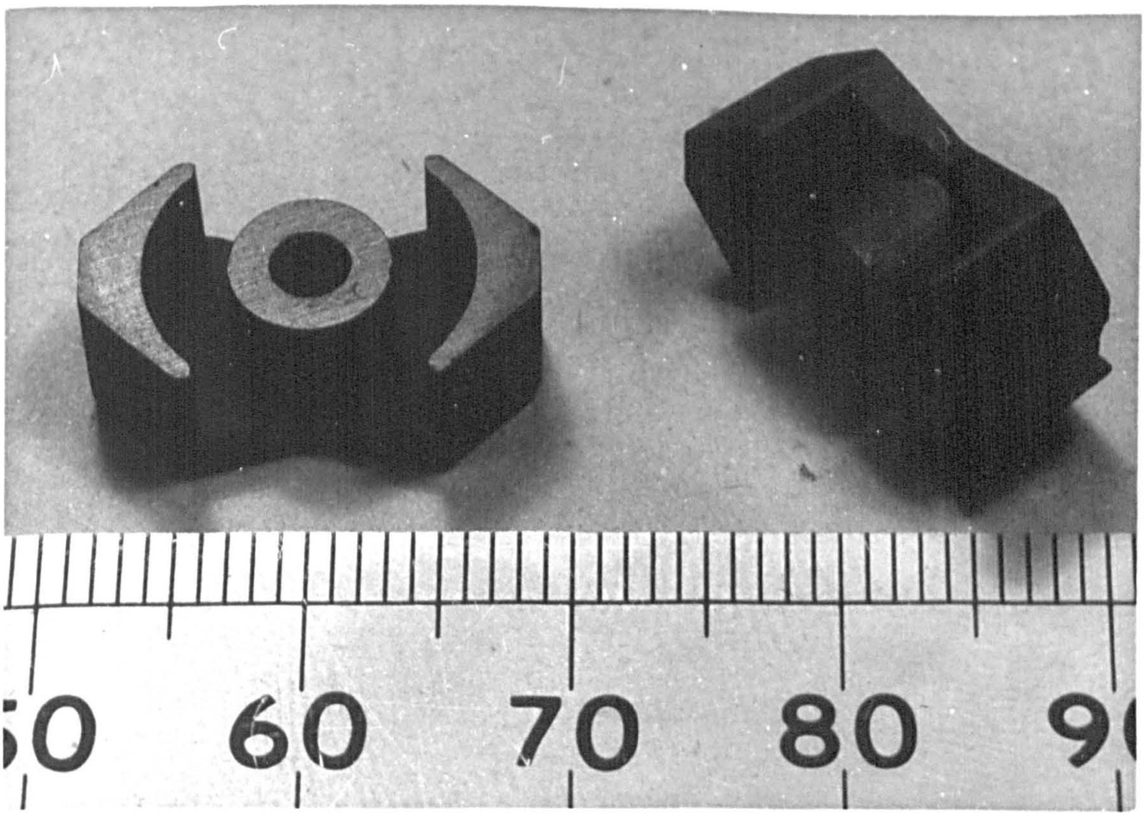


Plate 5.1 Ferroxcube inductor core with steel rule giving scale in mm

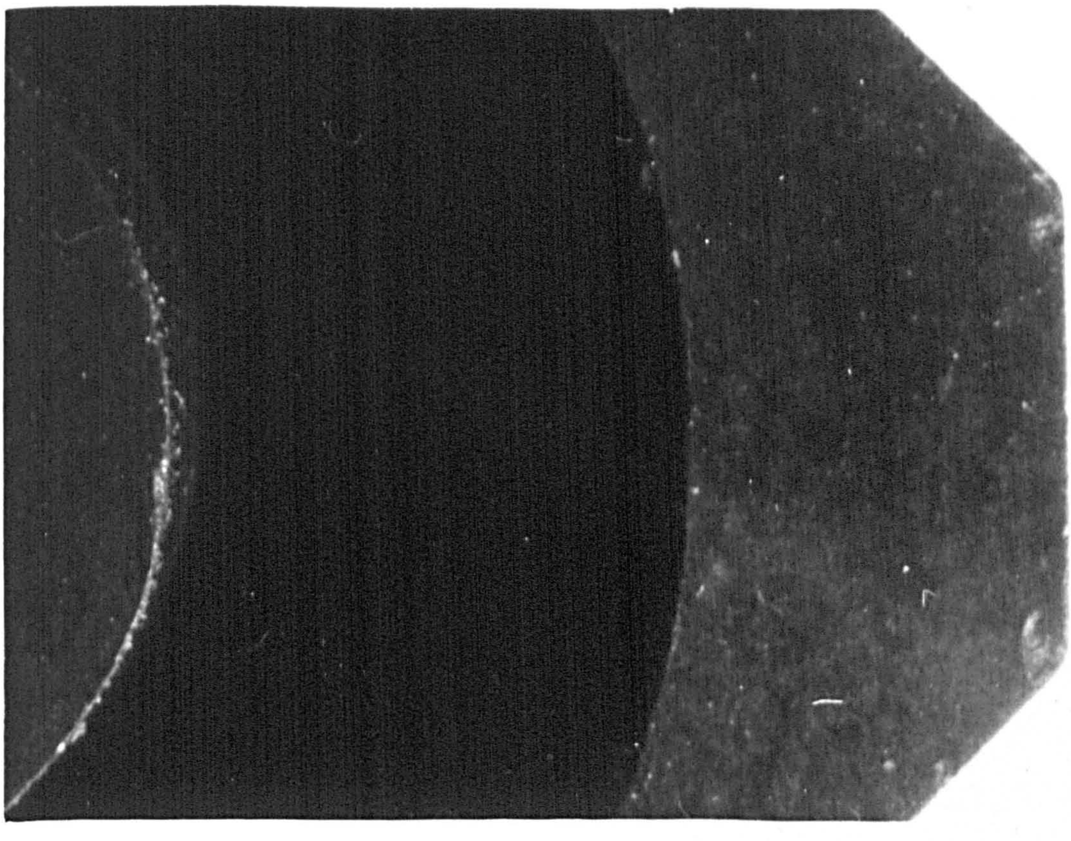


Plate 5.2 Close up of inductor core showing chipped edges and variation along the edges

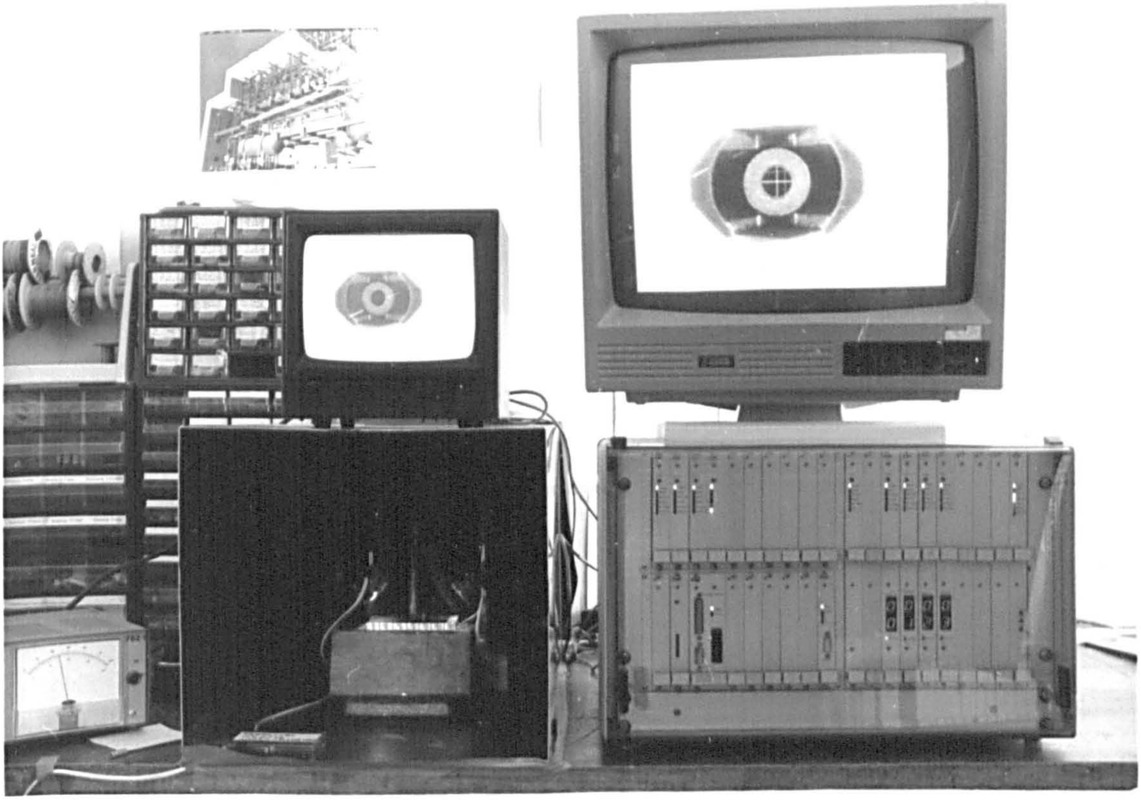


Plate 5.3 Ferroxcube inductor core measurement and inspection running on TIPS

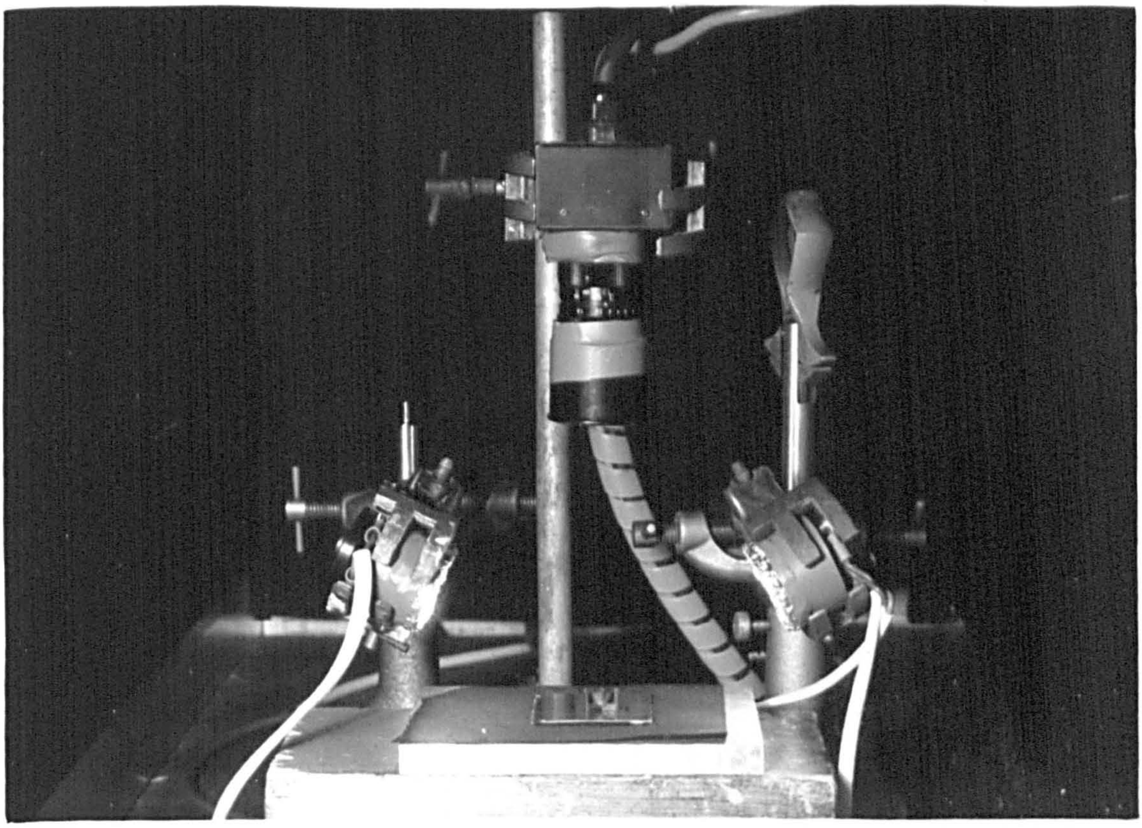


Plate 5.4 Optical arrangement with camera and lights above inductor core

5.8 Results from chapter five

The results which follow are divided into three sections, 5.8.1 showing images of the ferroxcube inductor core specimens, 5.8.2 shows their corresponding data and section 5.8.3 graphs from the data.

5.8.1 Images from the ferroxcube specimens

Images 5.1 to 5.7 show examples with the various information individually, which is drawn onto the fully processed images. Images 5.8 onwards then show unprocessed core specimen images follows by processed images for all specimens 1 to 9.

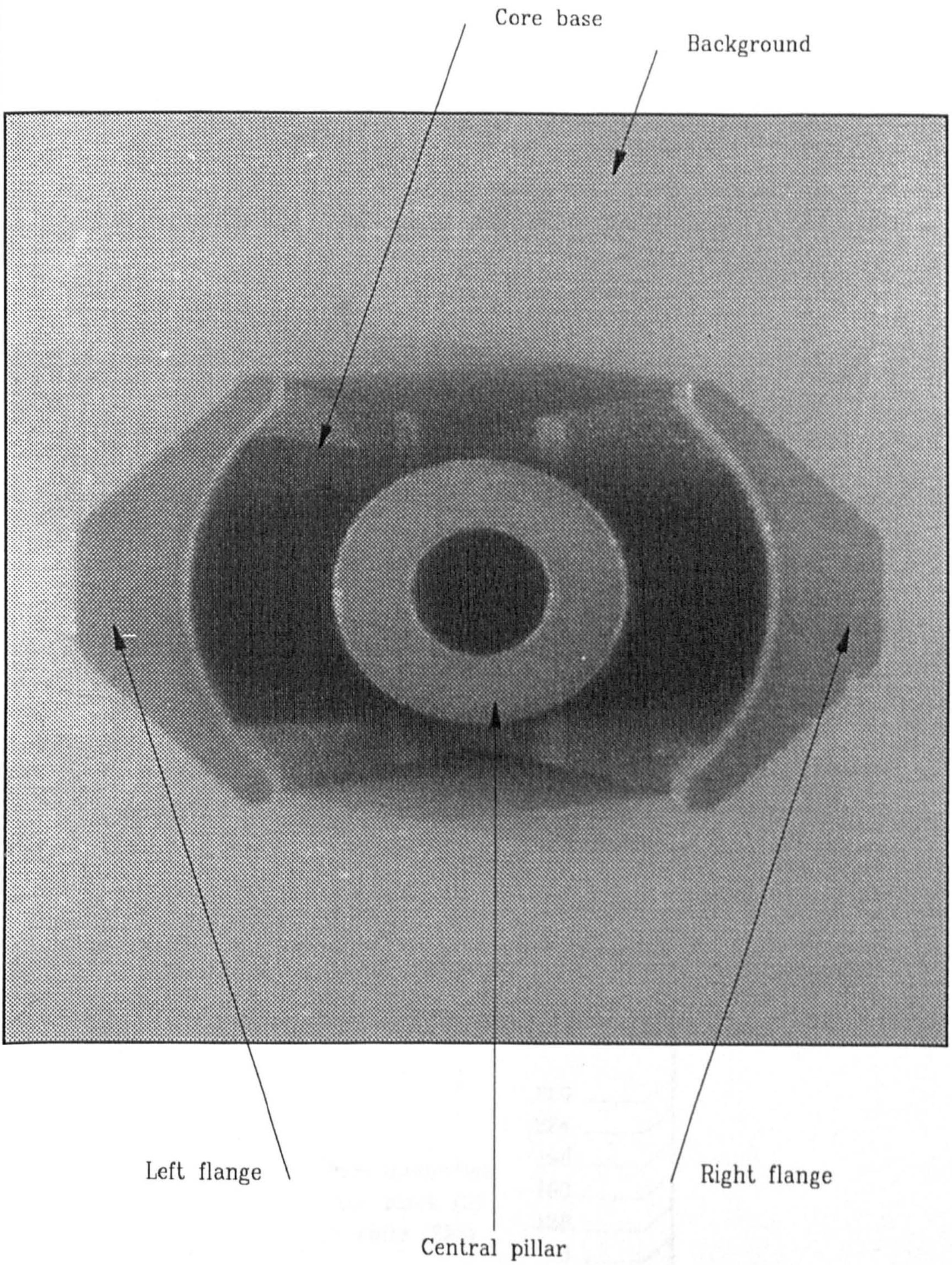


Image 5.1 Typical unprocessed image of a ferroxcube inductor core

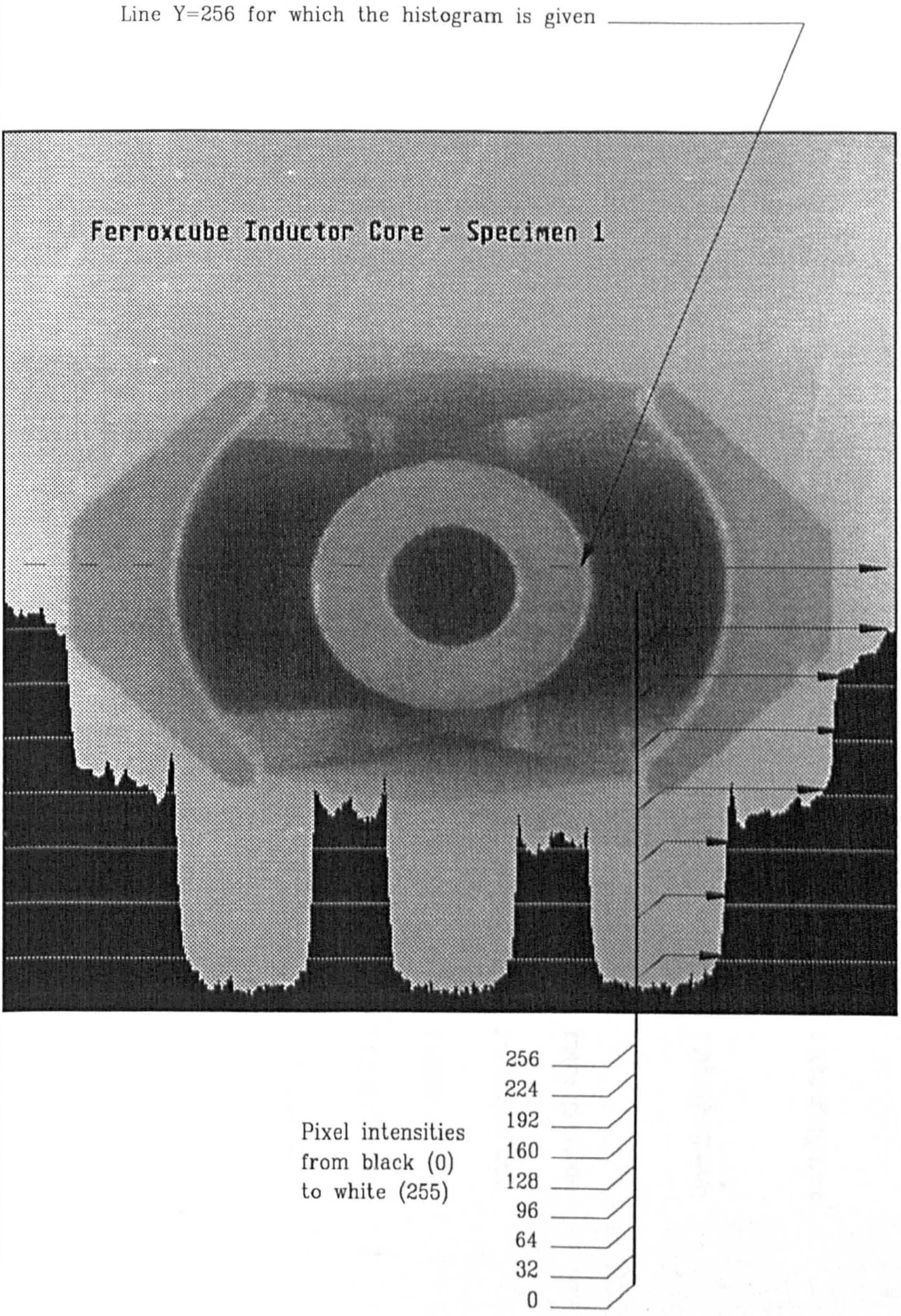


Image 5.2 Intensity histogram along horzontal line y = 256

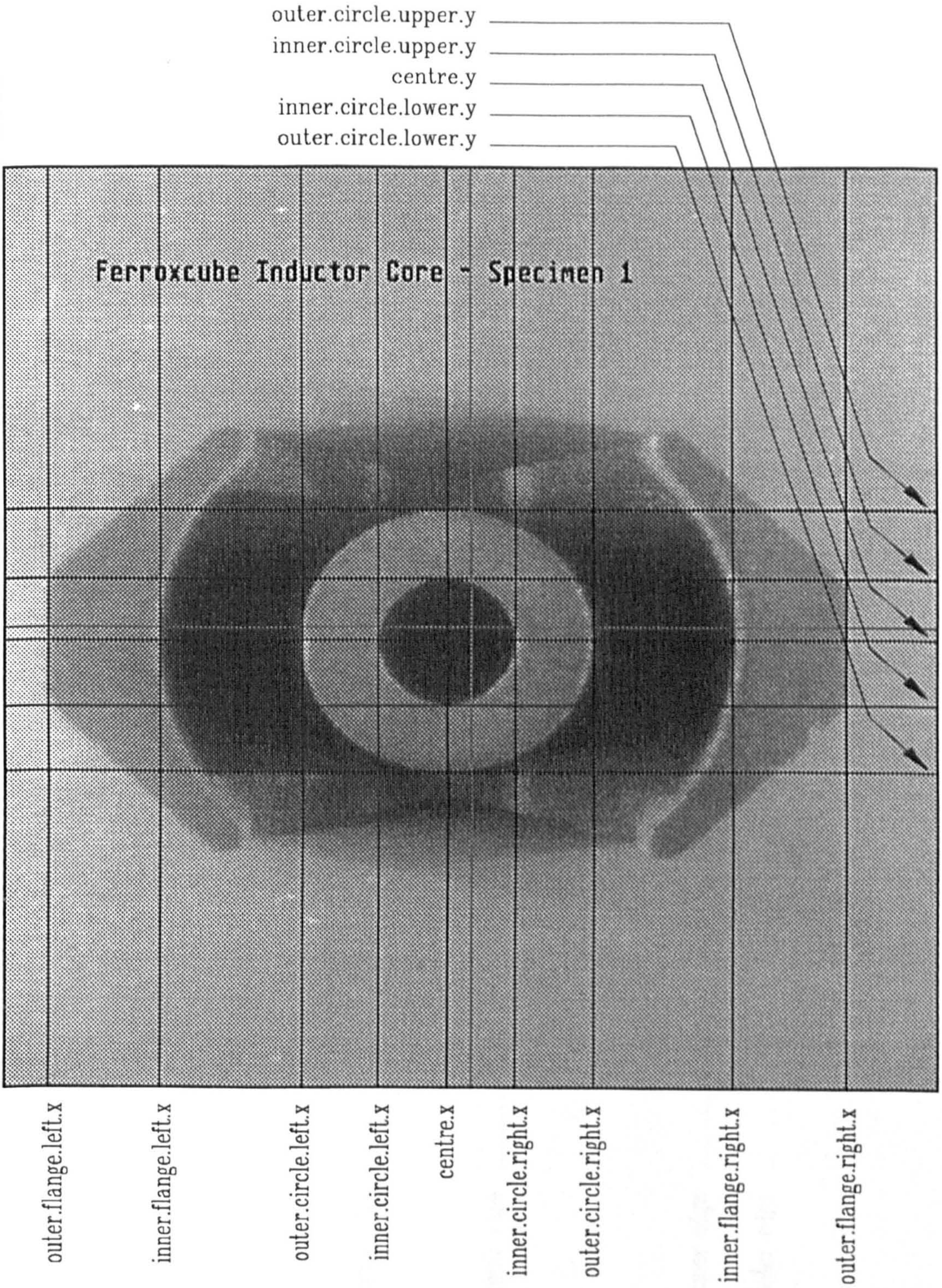


Image 5.3 Location lines with corresponding software names

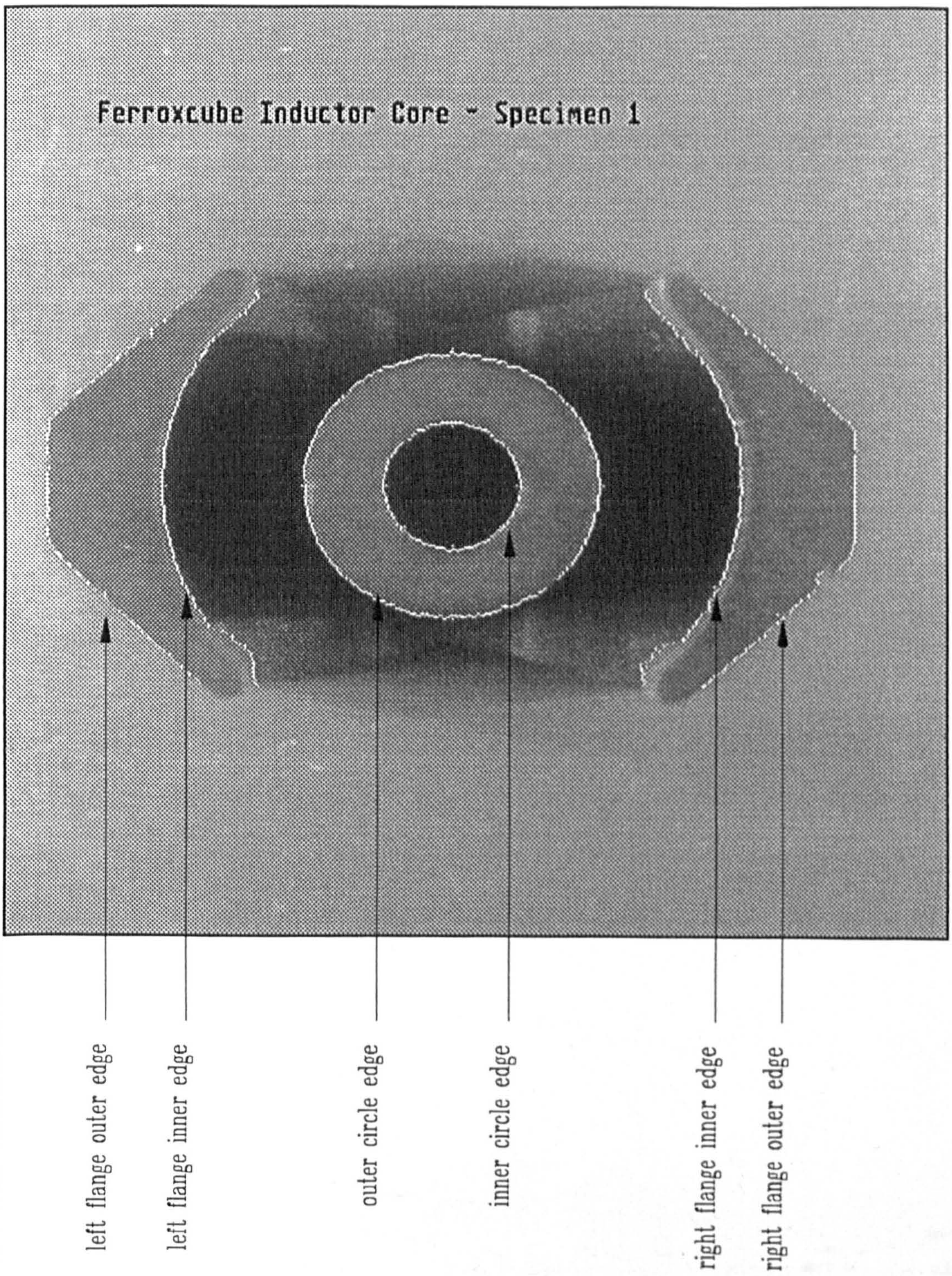
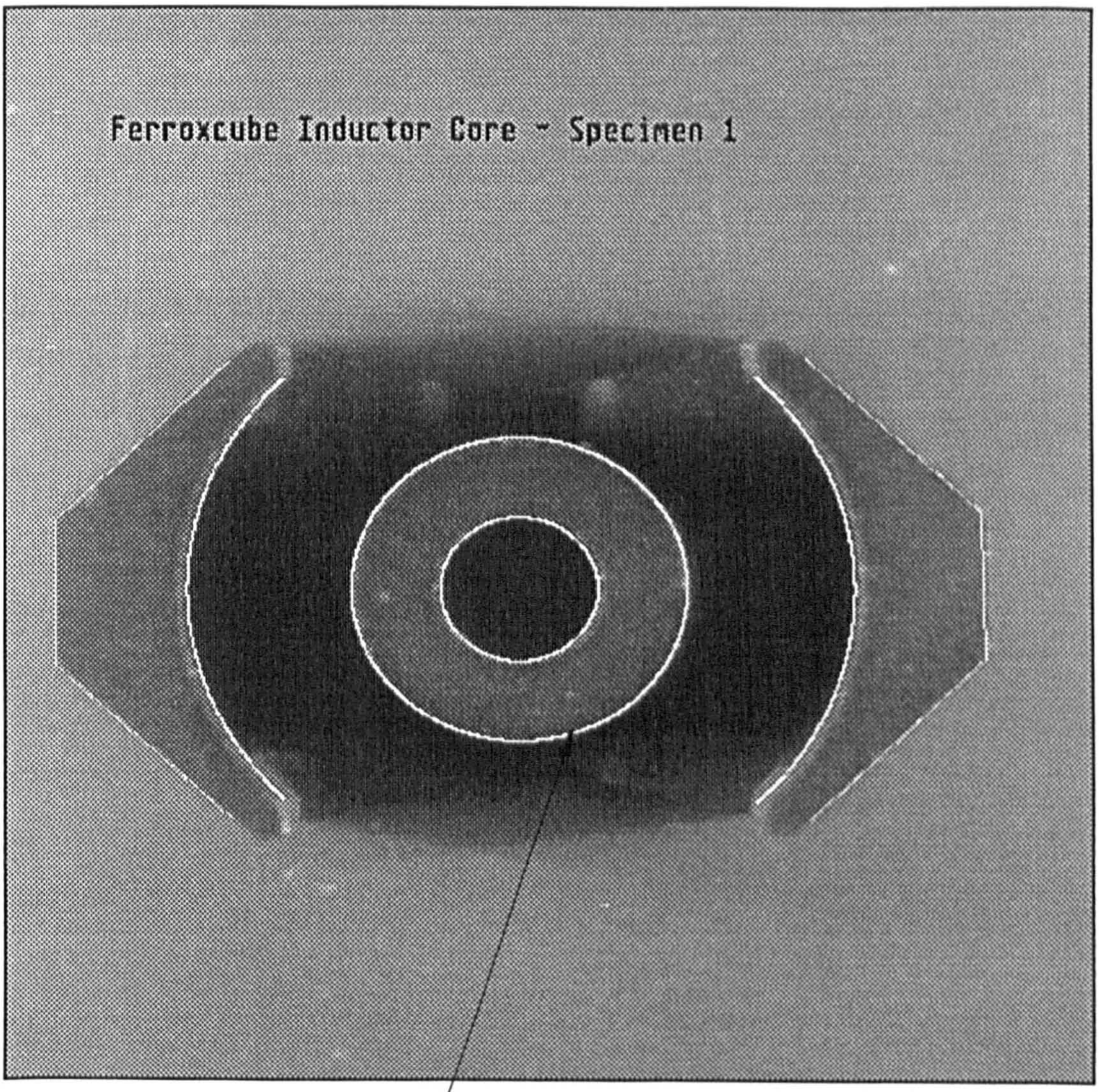


Image 5.4 Edges of the core found by inspection software



Expected or template
lines in white

Image 5.5 Template lines drawn on core

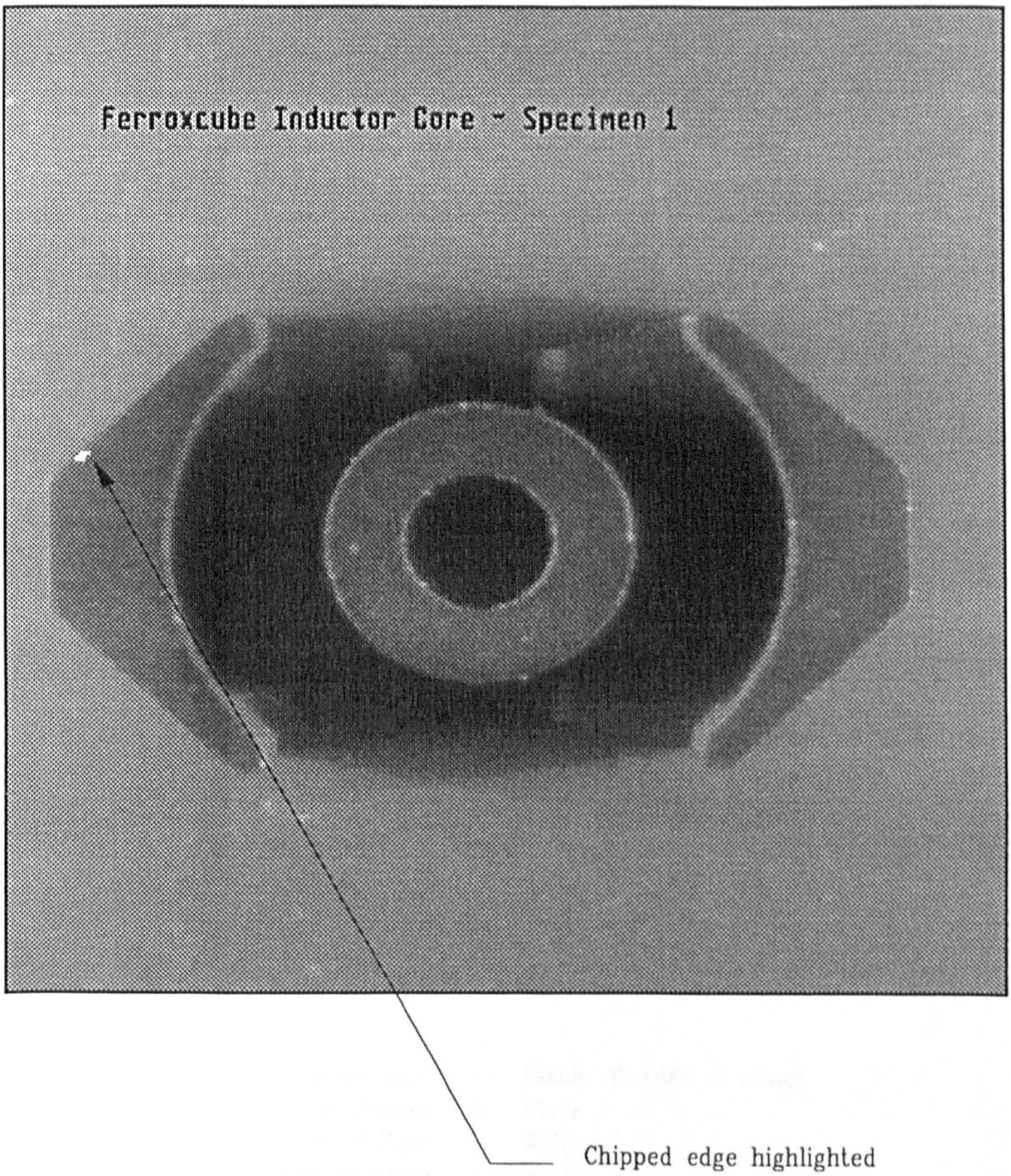
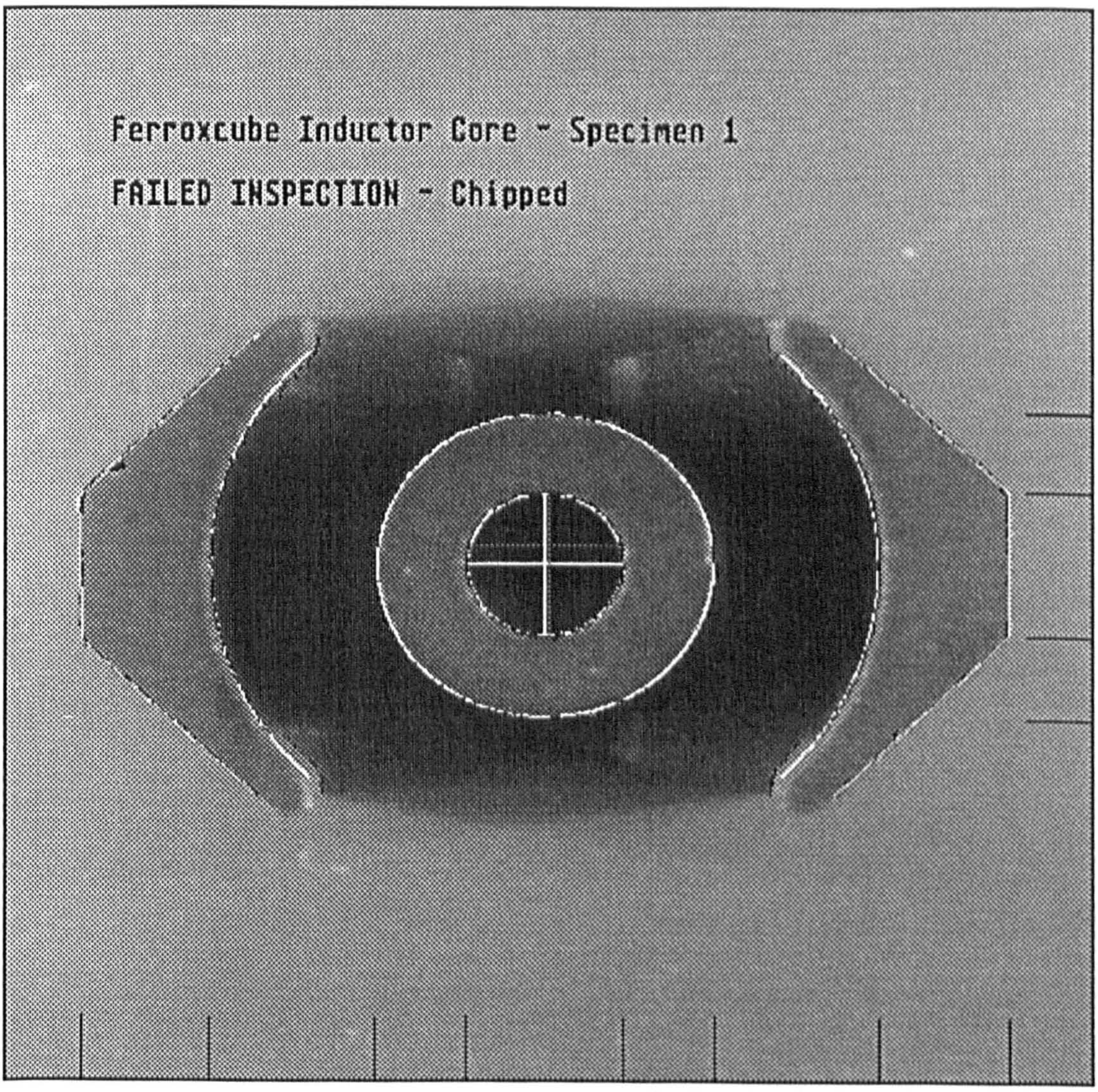


Image 5.6 Highlighted chipped areas of the core

Processed image showing all features and lines


Ferroxcube Inductor Core - Specimen 1
FAILED INSPECTION - Chipped

Feature location lines	–	Black, at edge of image
Template edge lines	–	Black
Searched edge lines	–	White
Chipped edges	–	Black
Core centre	–	White
Image centre (256,256)	–	Grey

Image 5.7 Core with location lines, edges, template lines and chipped edges shown

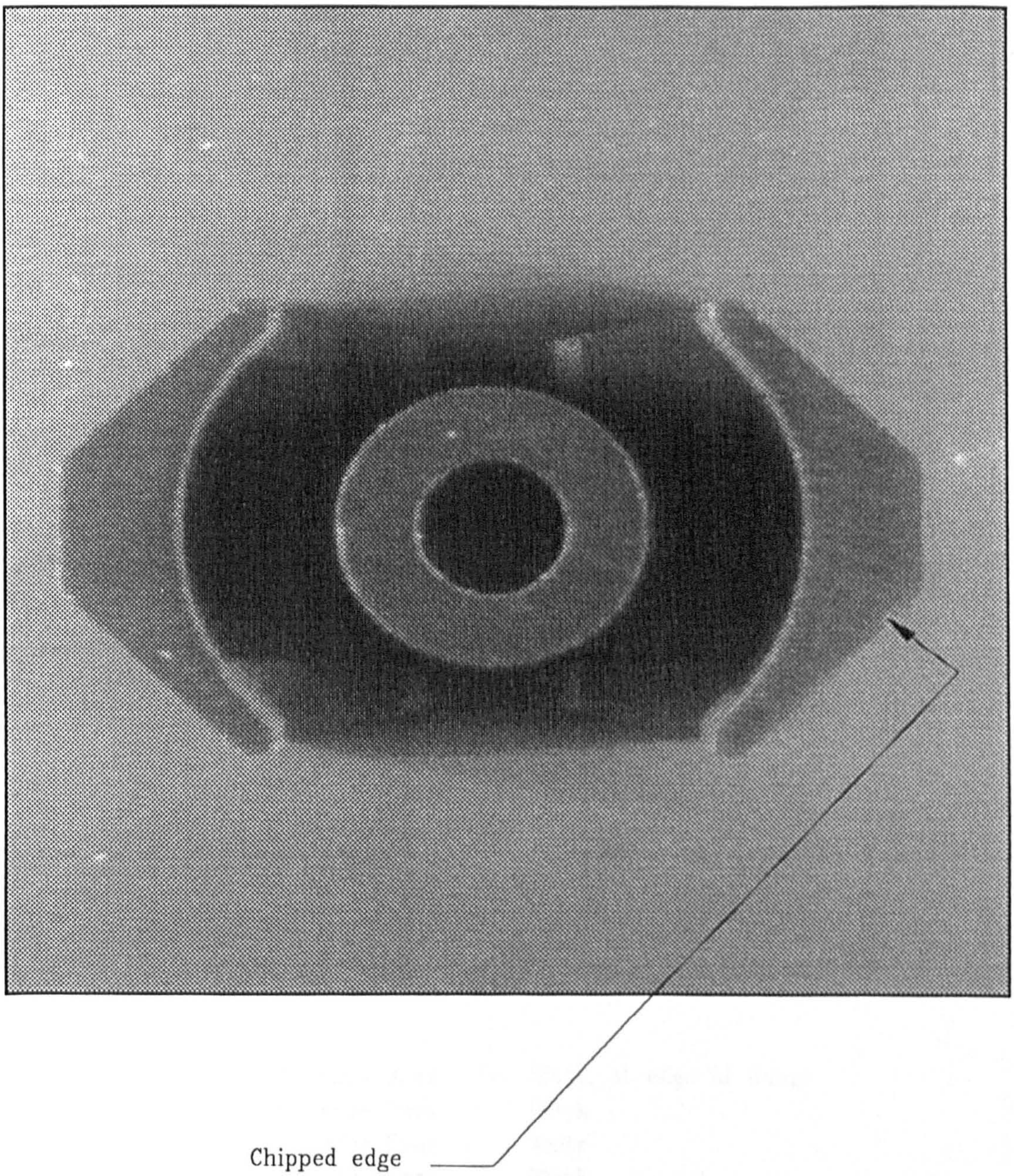
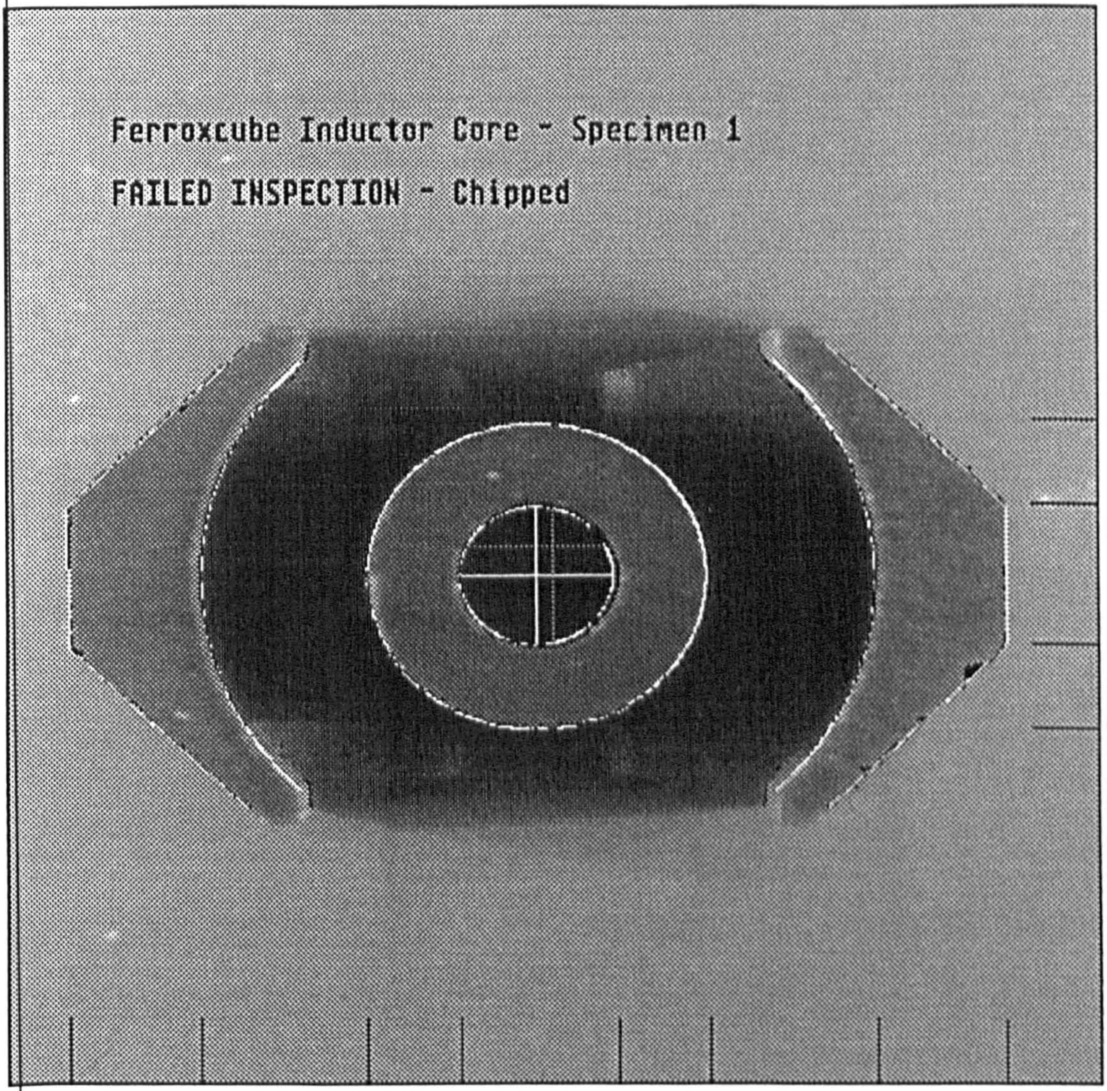


Image 5.8 Test sample 1, unprocessed image

Processed image showing all features and lines


Ferroxcube Inductor Core - Specimen 1
FAILED INSPECTION - Chipped

Feature location lines	–	Black, at edge of image
Template edge lines	–	Black
Searched edge lines	–	White
Chipped edges	–	Black
Core centre	–	White
Image centre (256,256)	–	Grey

Image 5.9 Test sample 1, after processing (See Section 5.9.3)

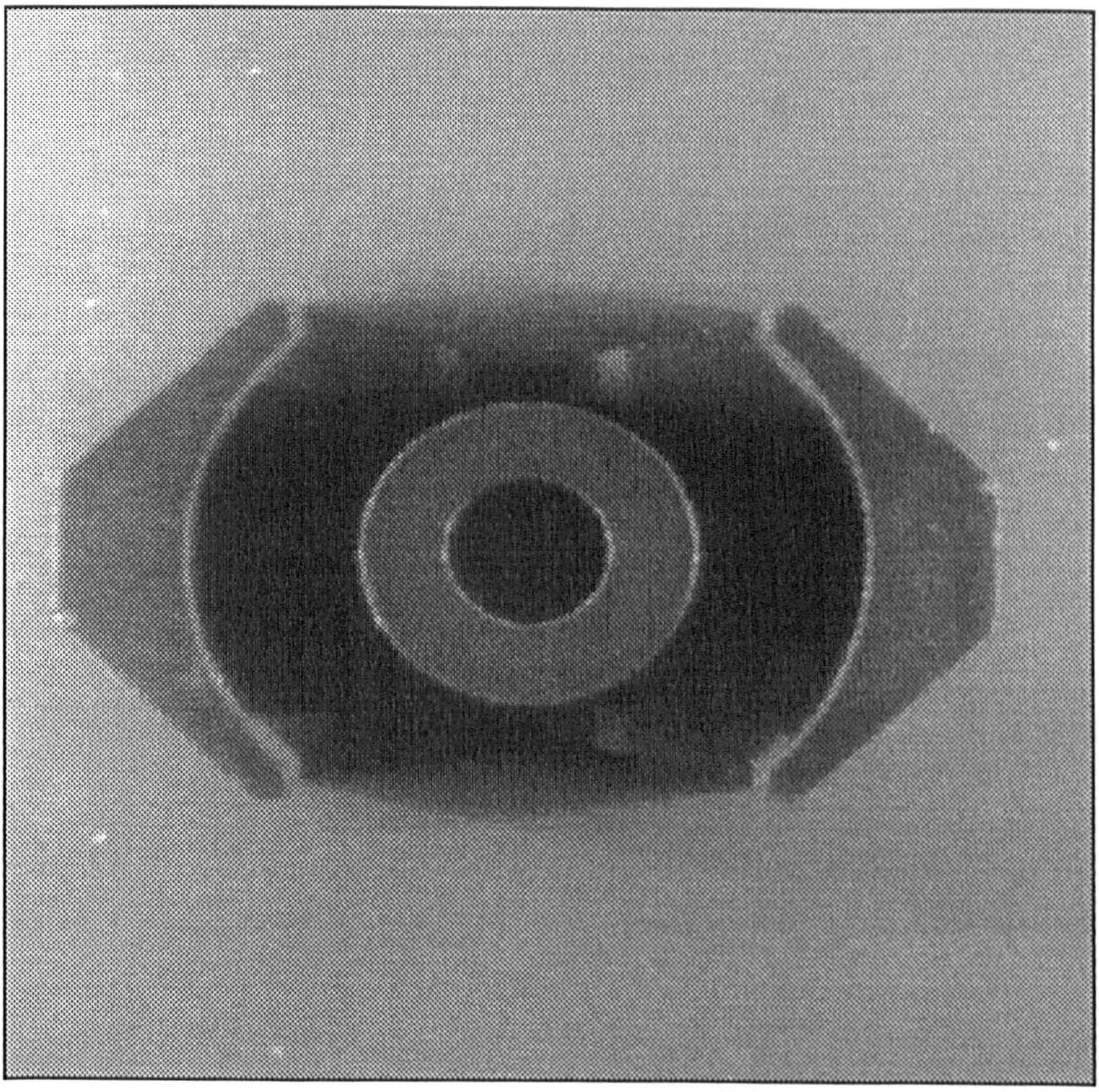
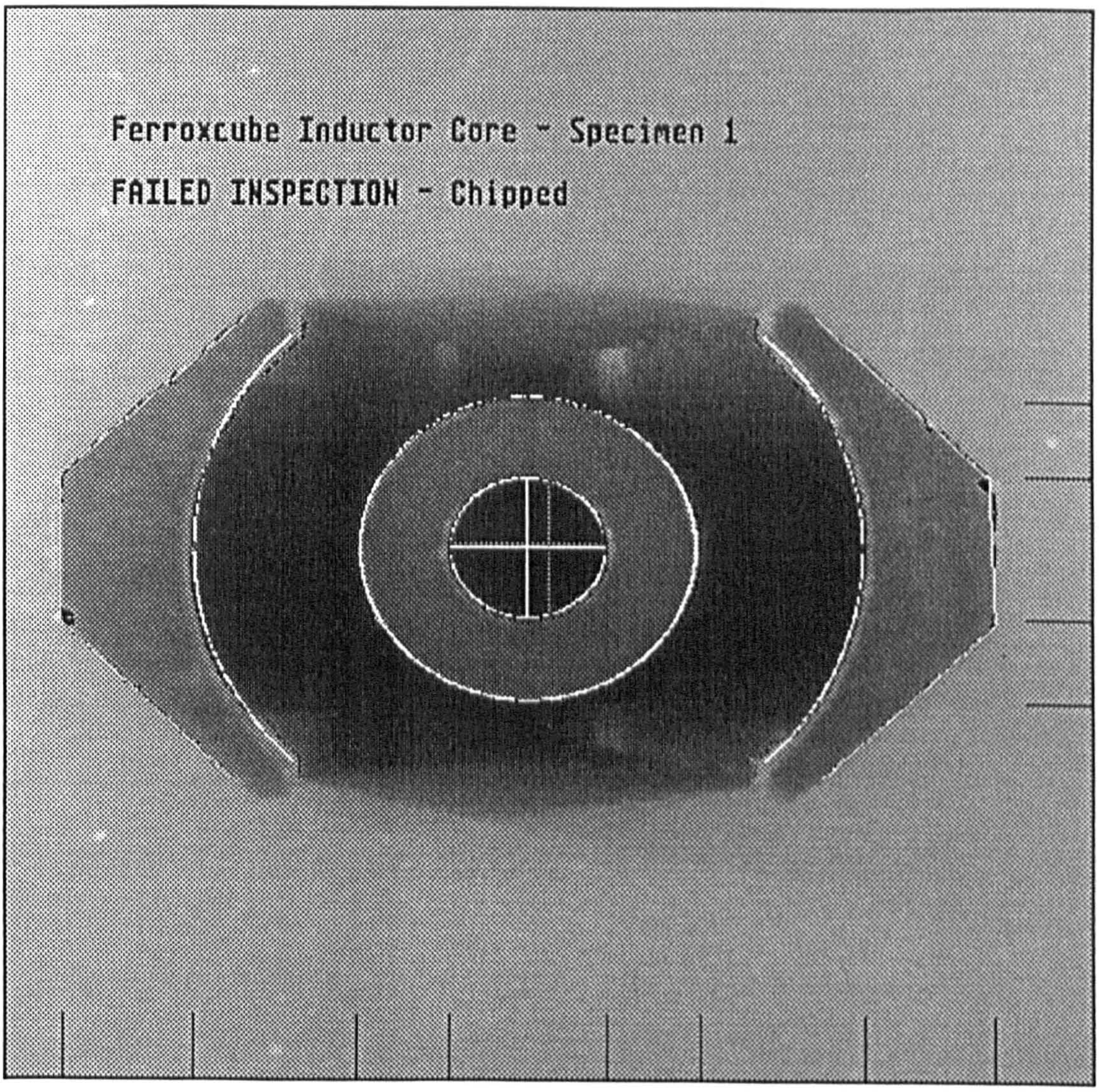


Image 5.10 Test sample 2, unprocessed image

Processed image showing all features and lines


Ferroxcube Inductor Core - Specimen 1
FAILED INSPECTION - Chipped

Feature location lines	–	Black, at edge of image
Template edge lines	–	Black
Searched edge lines	–	White
Chipped edges	–	Black
Core centre	–	White
Image centre (256,256)	–	Grey

Image 5.11 Test sample 2, after processing (See Section 5.9.3)

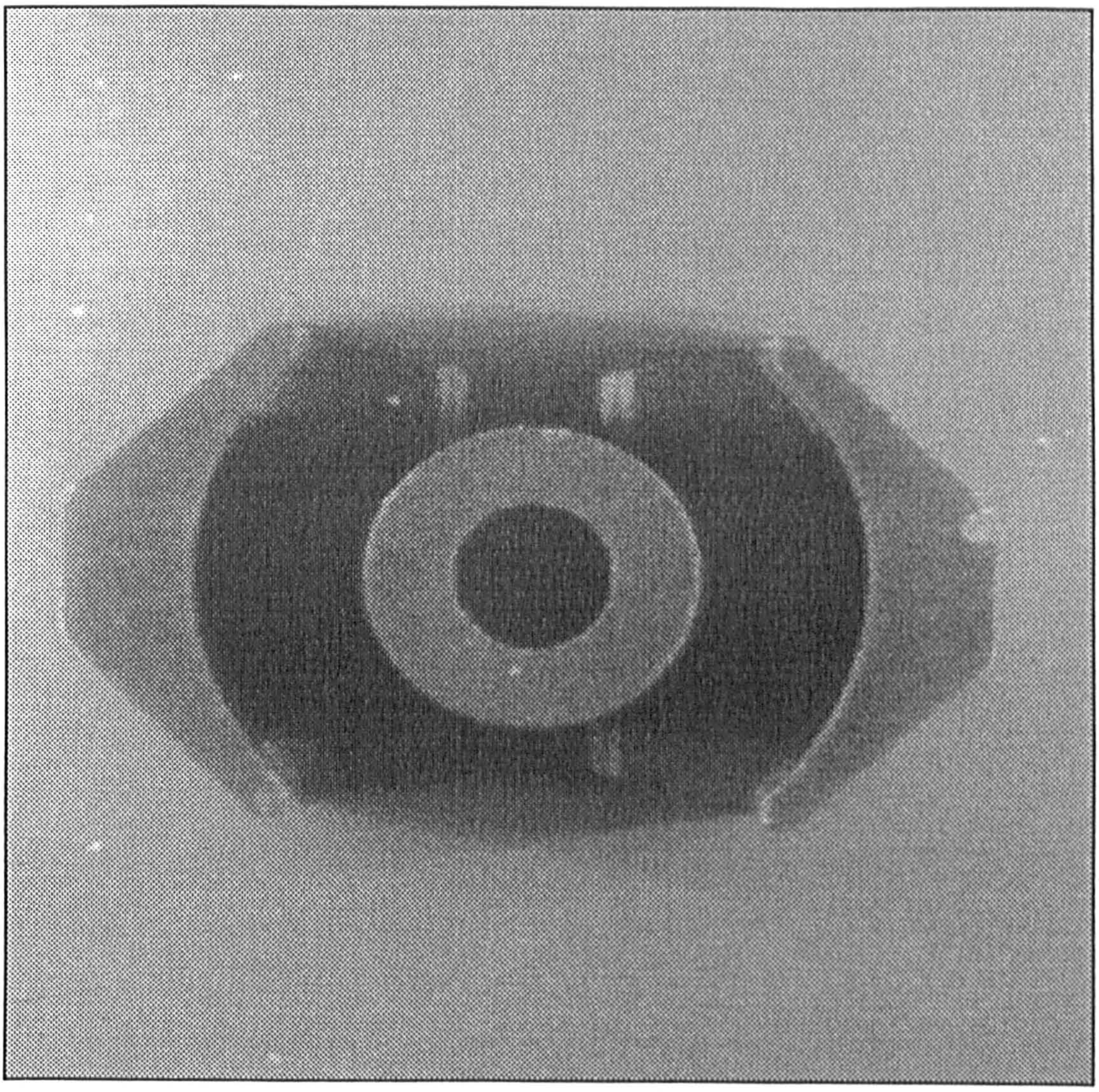
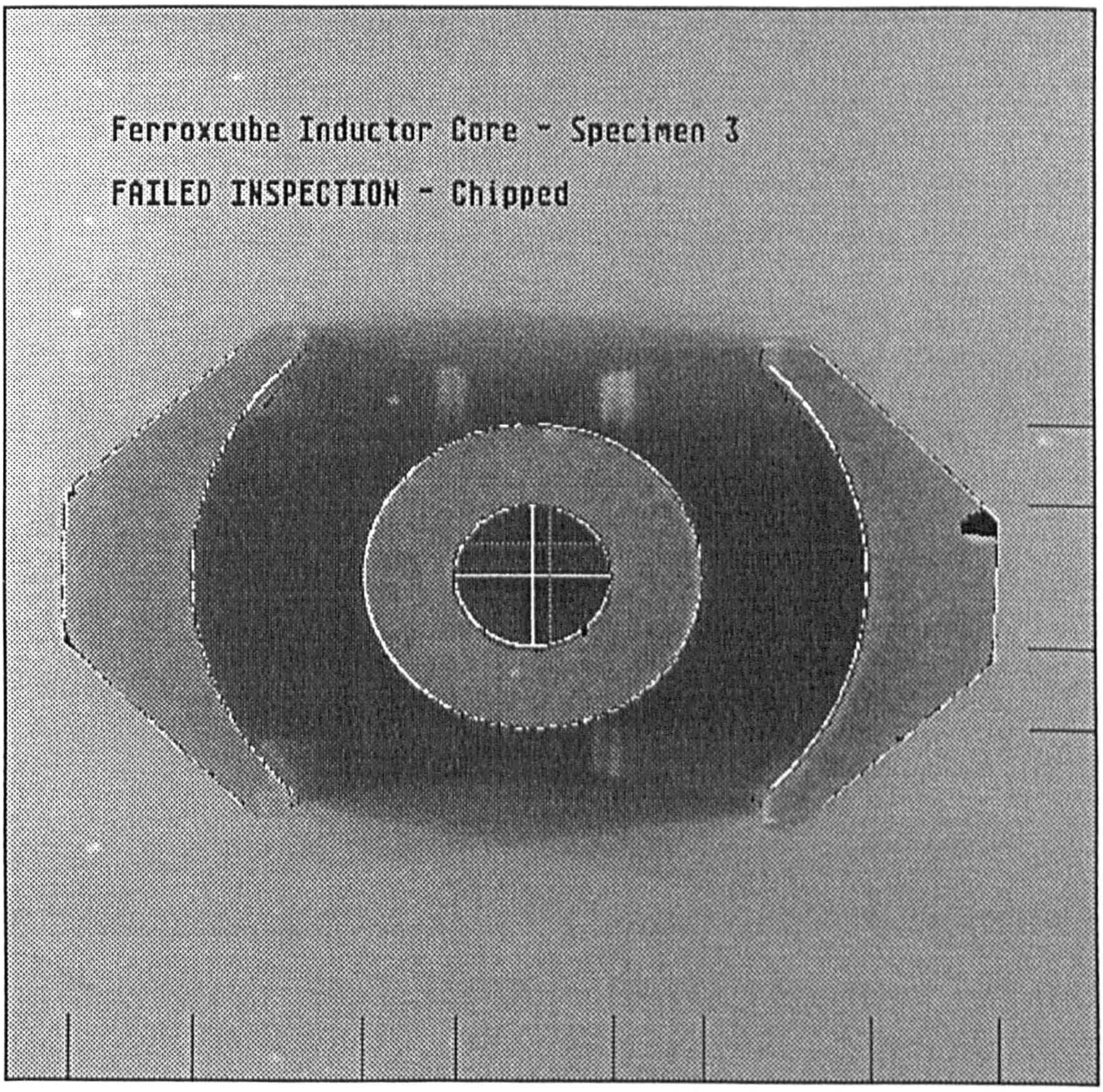


Image 5.12 Test sample 3, unprocessed image

Processed image showing all features and lines


Ferroxcube Inductor Core - Specimen 3
FAILED INSPECTION - Chipped

Feature location lines	–	Black, at edge of image
Template edge lines	–	Black
Searched edge lines	–	White
Chipped edges	–	Black
Core centre	–	White
Image centre (256,256)	–	Grey

Image 5.13 Test sample 3, after processing (See Section 5.9.3)

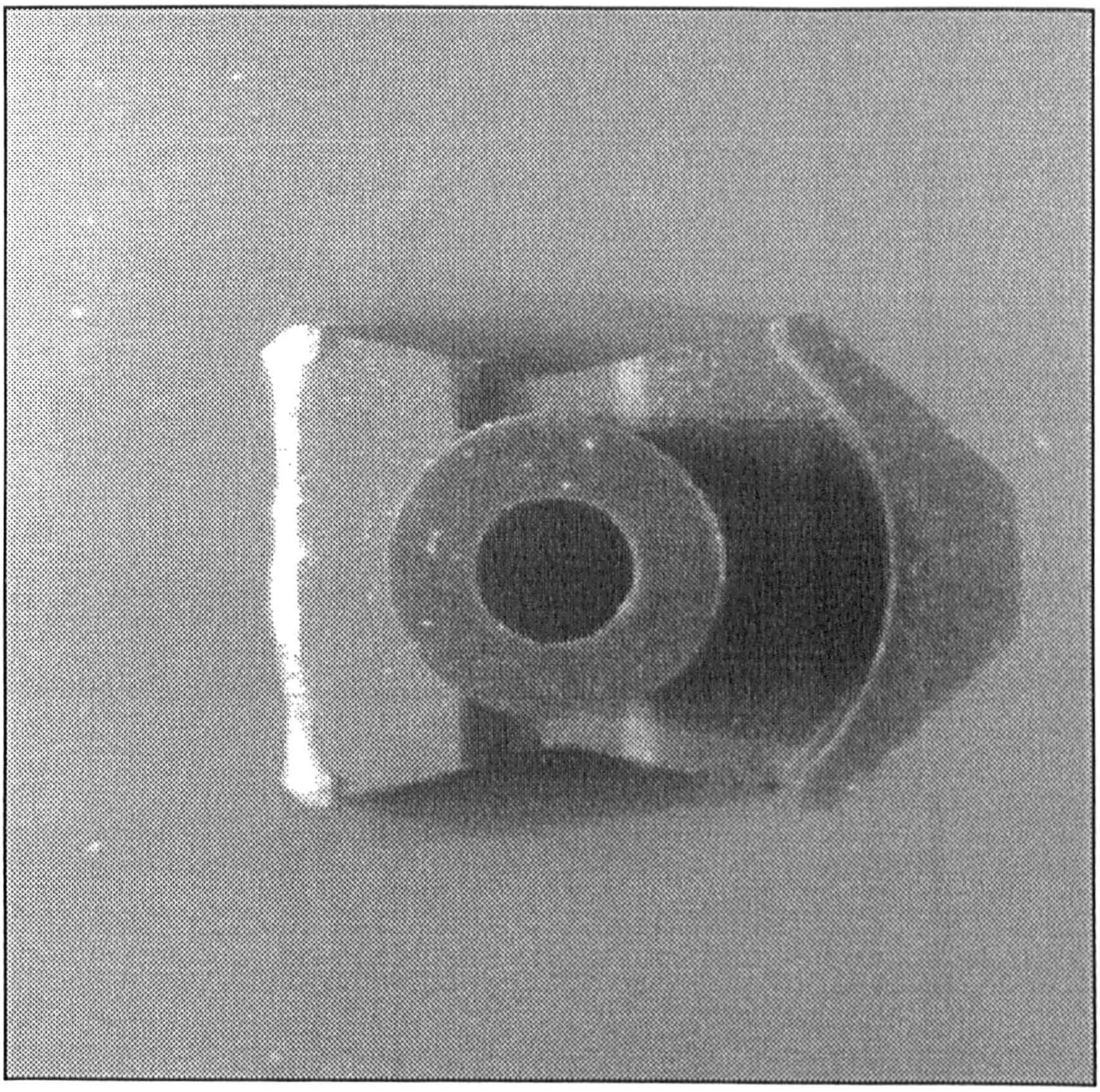
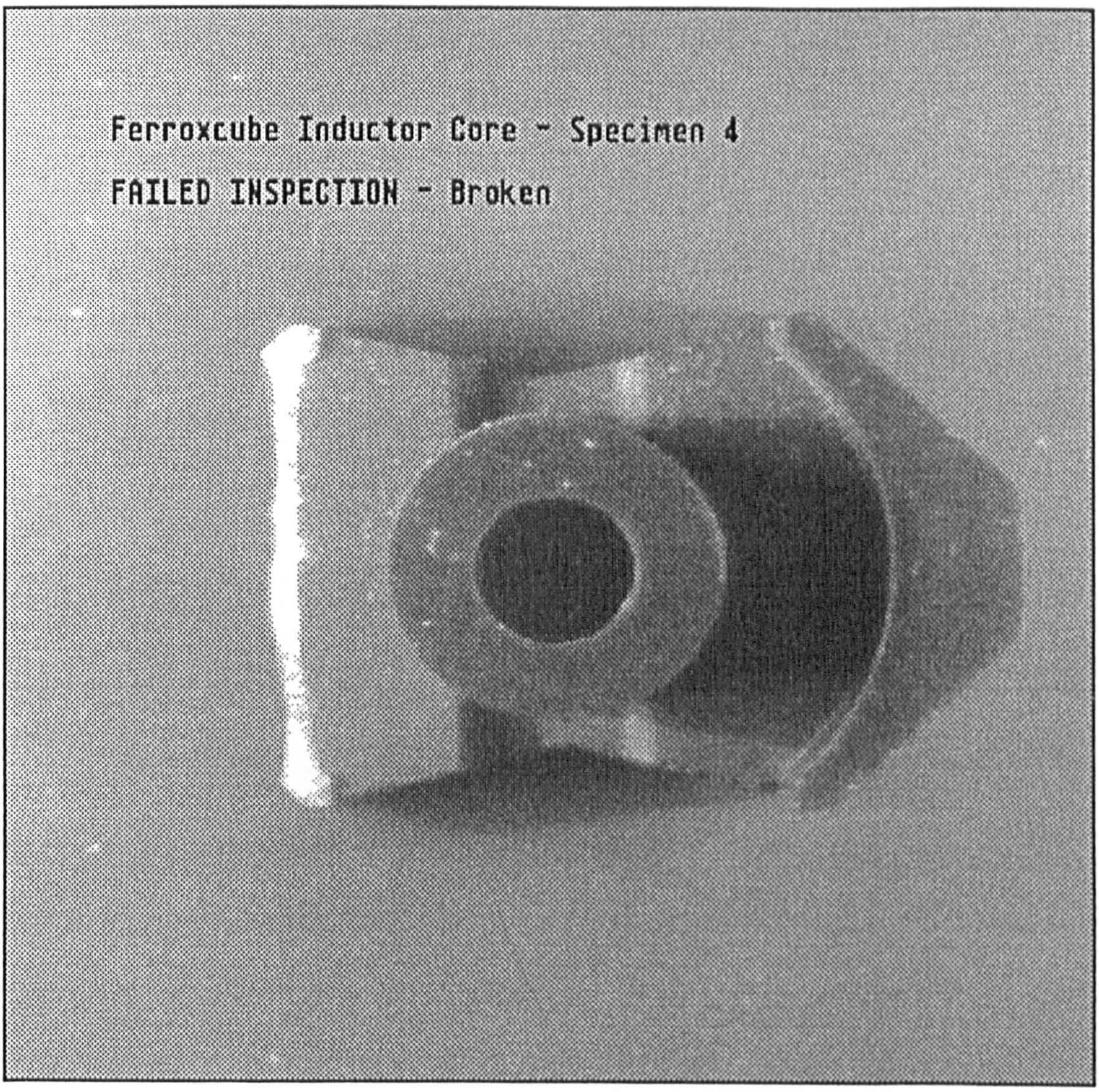


Image 5.14 Test sample 4, unprocessed image

Processed image showing all features and lines


Ferroxcube Inductor Core - Specimen 4
FAILED INSPECTION - Broken

Feature location lines	–	Black, at edge of image
Template edge lines	–	Black
Searched edge lines	–	White
Chipped edges	–	Black
Core centre	–	White
Image centre (256,256)	–	Grey

Image 5.15 Test sample 4, after processing (See Section 5.9.3)

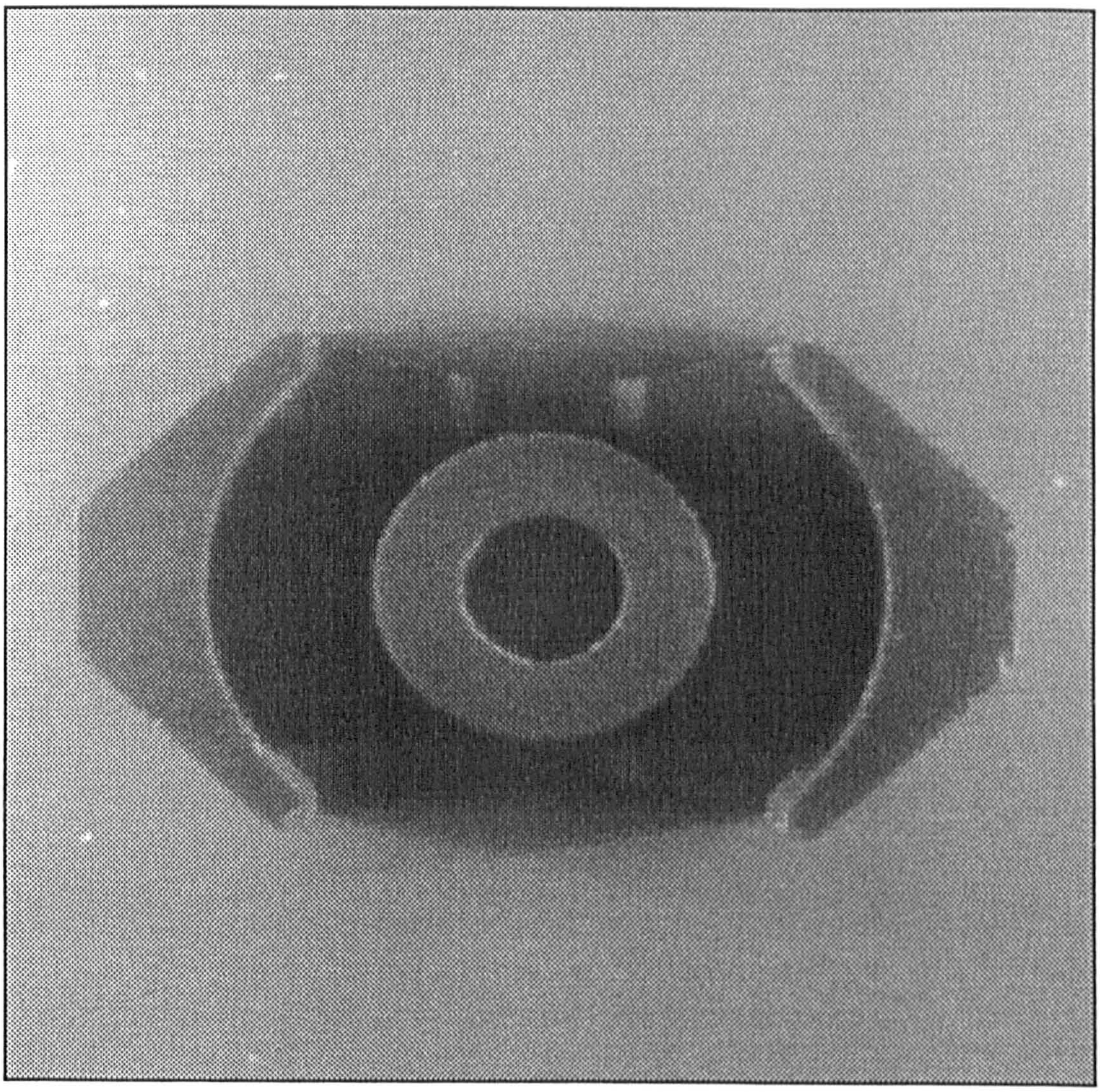
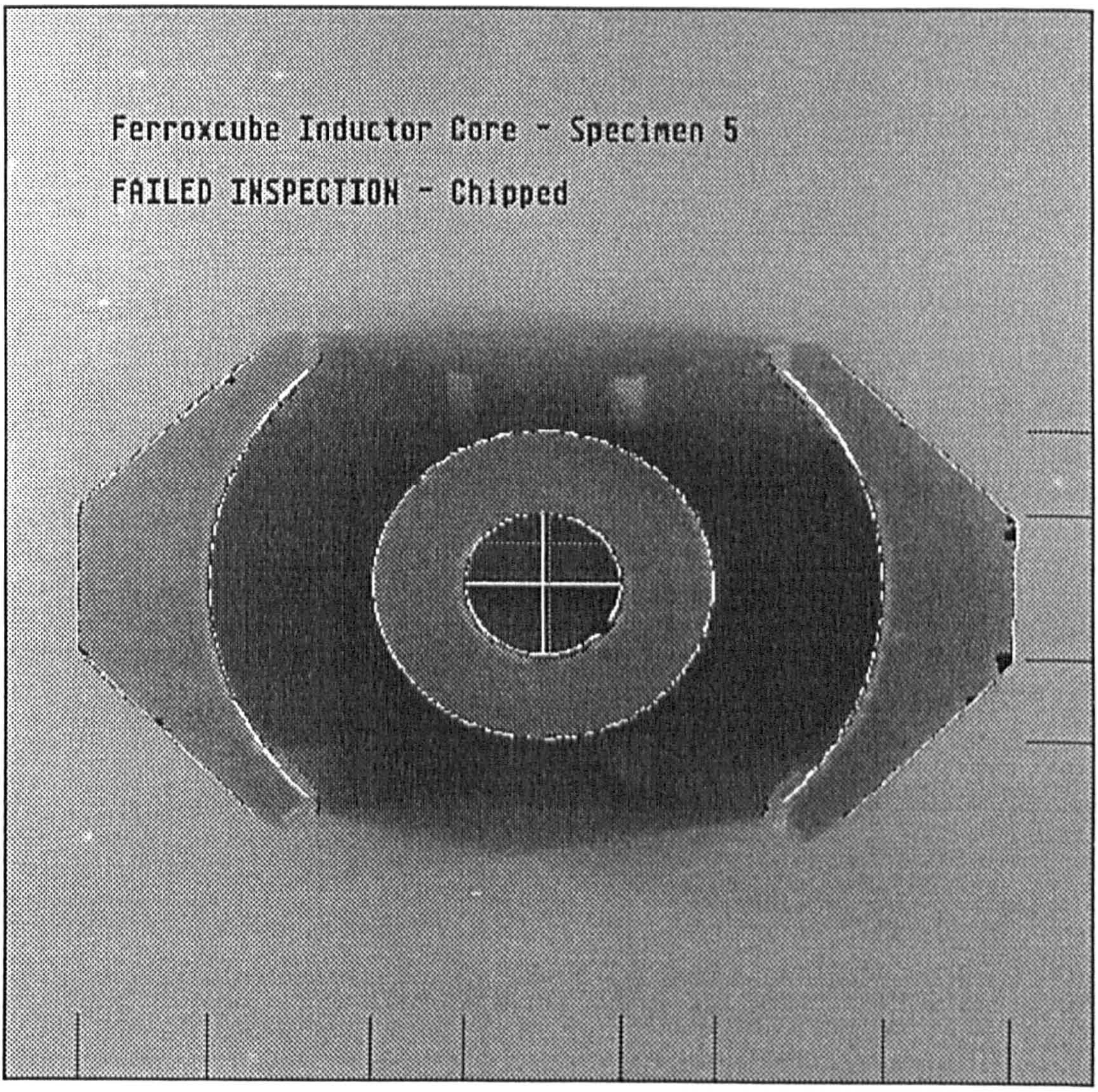


Image 5.16 Test sample 5, unprocessed image

Processed image showing all features and lines


Ferroxcube Inductor Core - Specimen 5
FAILED INSPECTION - Chipped

Feature location lines	–	Black, at edge of image
Template edge lines	–	Black
Searched edge lines	–	White
Chipped edges	–	Black
Core centre	–	White
Image centre (256,256)	–	Grey

Image 5.17 Test sample 5, after processing (See Section 5.9.3)

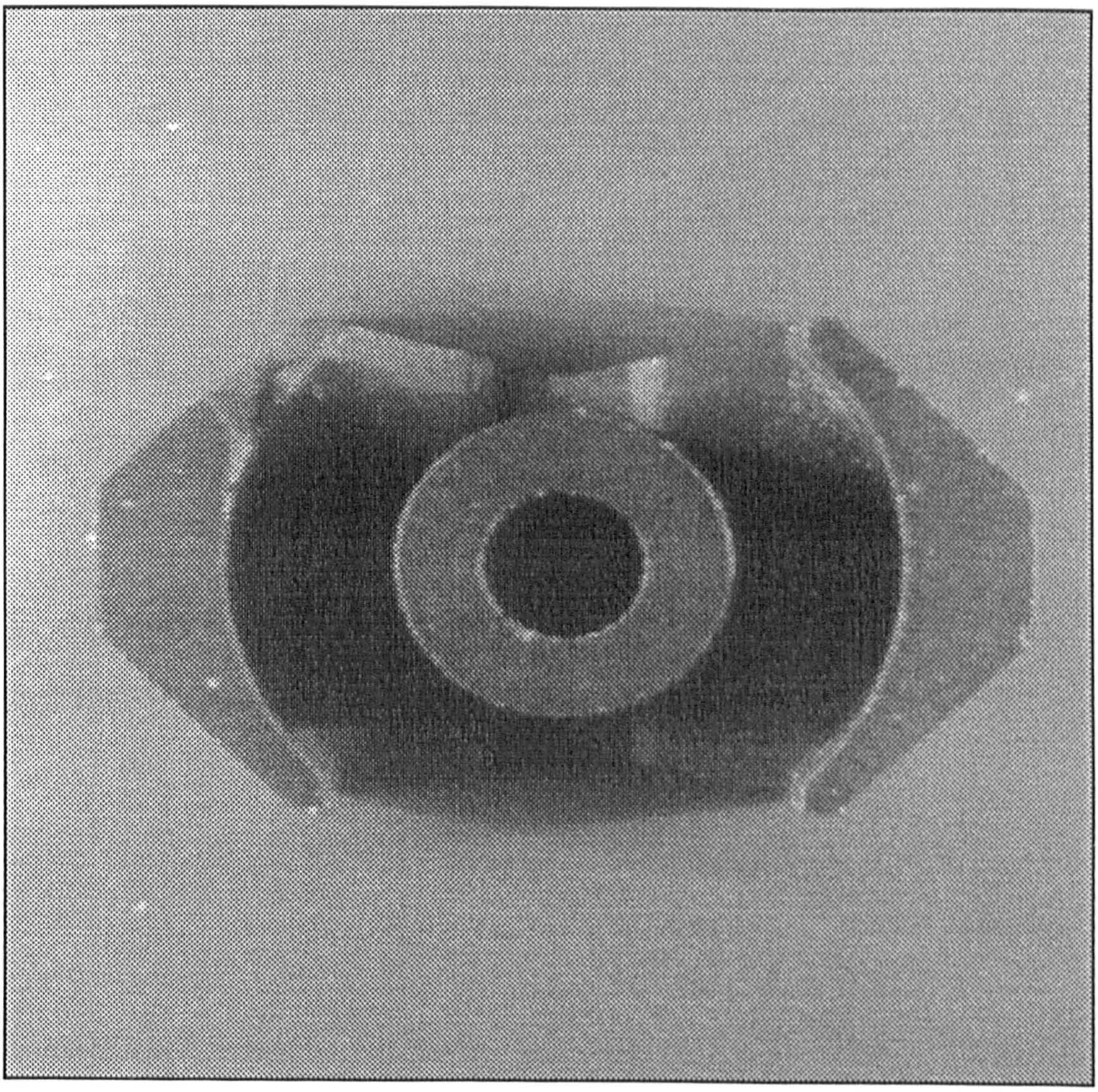
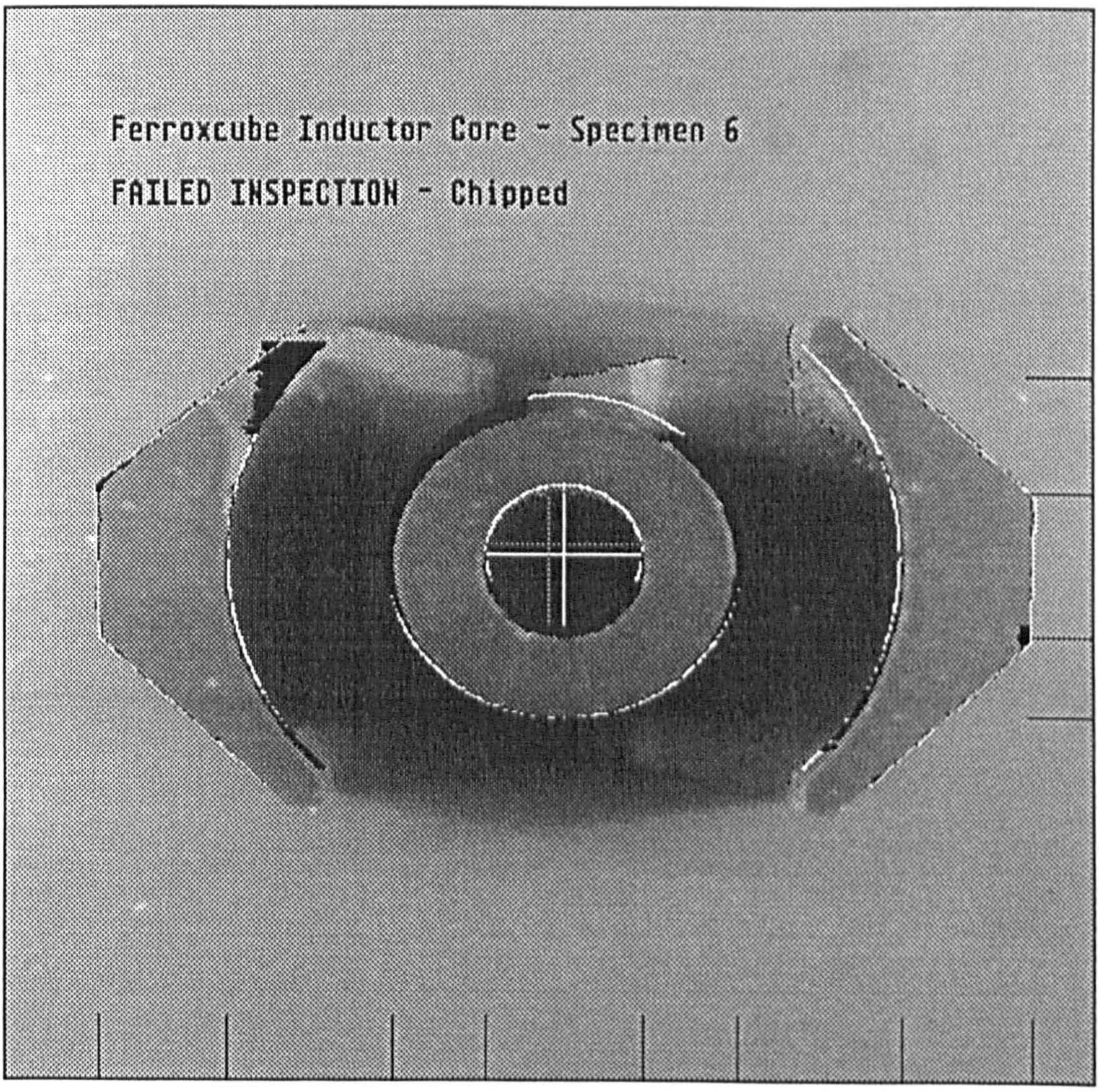


Image 5.18 Test sample 6, unprocessed image

Processed image showing all features and lines


Ferroxcube Inductor Core - Specimen 6
FAILED INSPECTION - Chipped

Feature location lines	–	Black, at edge of image
Template edge lines	–	Black
Searched edge lines	–	White
Chipped edges	–	Black
Core centre	–	White
Image centre (256,256)	–	Grey

Image 5.19 Test sample 6, after processing (See Section 5.9.3)

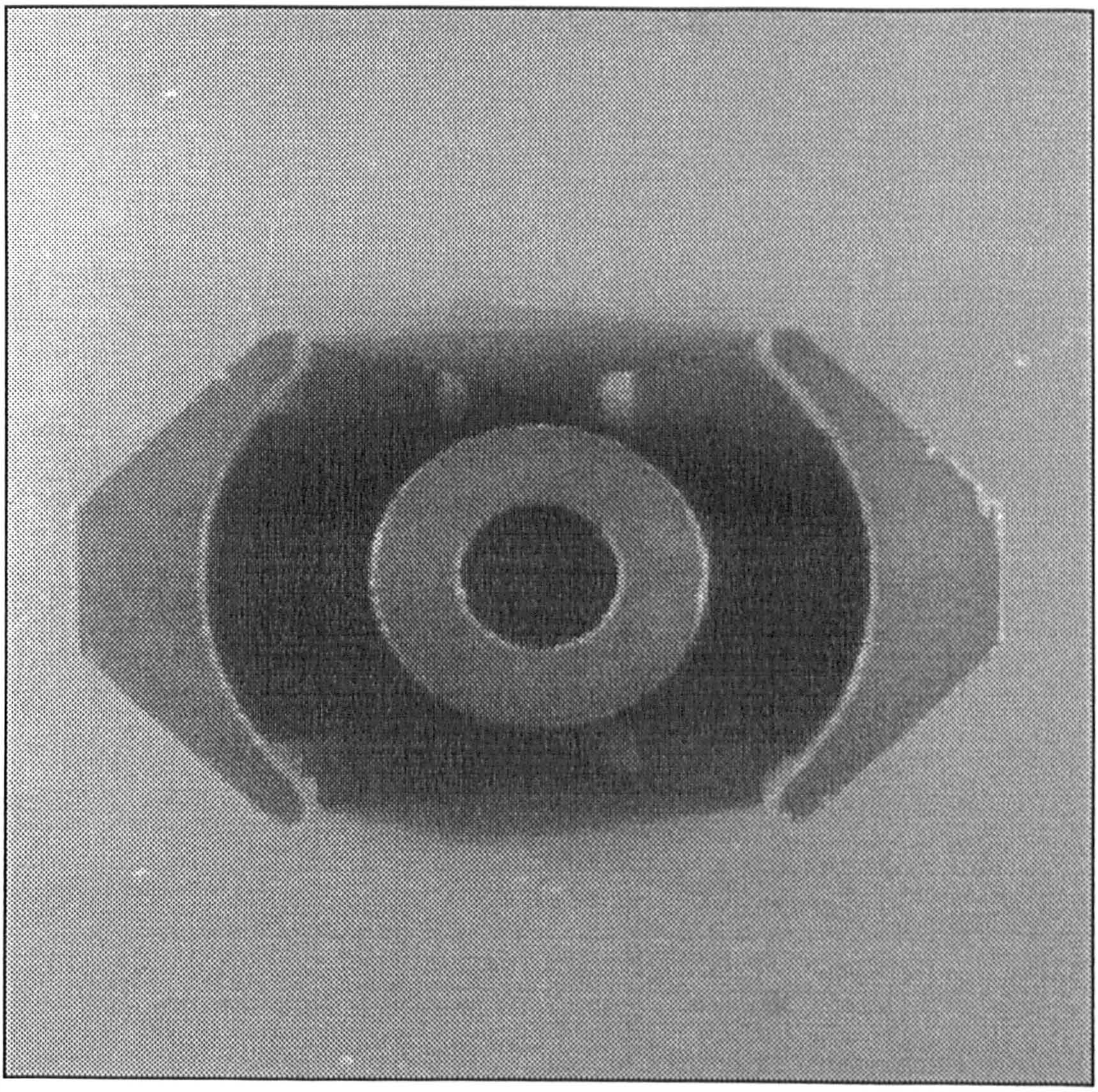
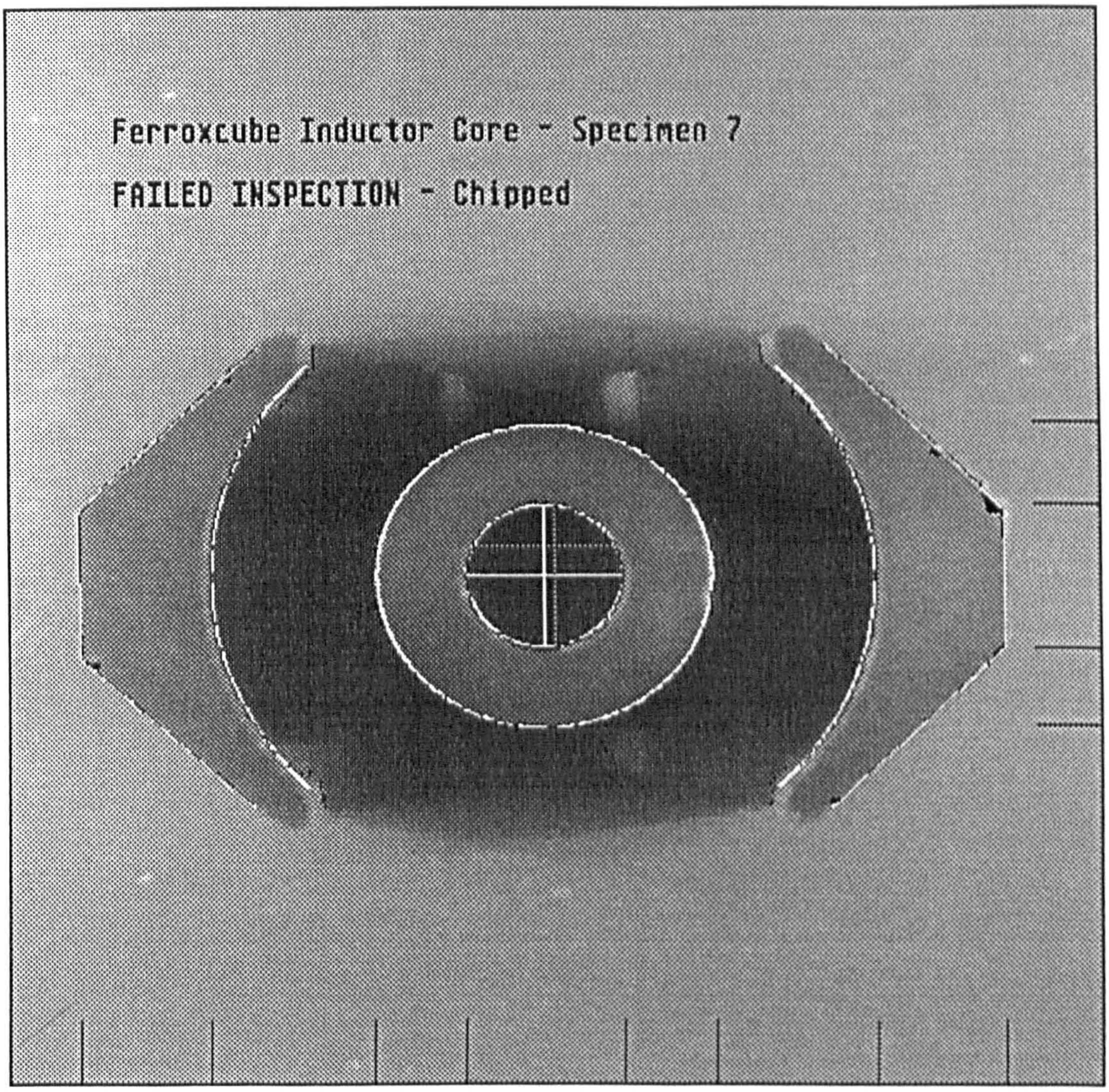


Image 5.20 Test sample 7, unprocessed image

Processed image showing all features and lines


Ferroxcube Inductor Core - Specimen 7
FAILED INSPECTION - Chipped

Feature location lines	–	Black, at edge of image
Template edge lines	–	Black
Searched edge lines	–	White
Chipped edges	–	Black
Core centre	–	White
Image centre (256,256)	–	Grey

Image 5.21 Test sample 7, after processing (See Section 5.9.3)

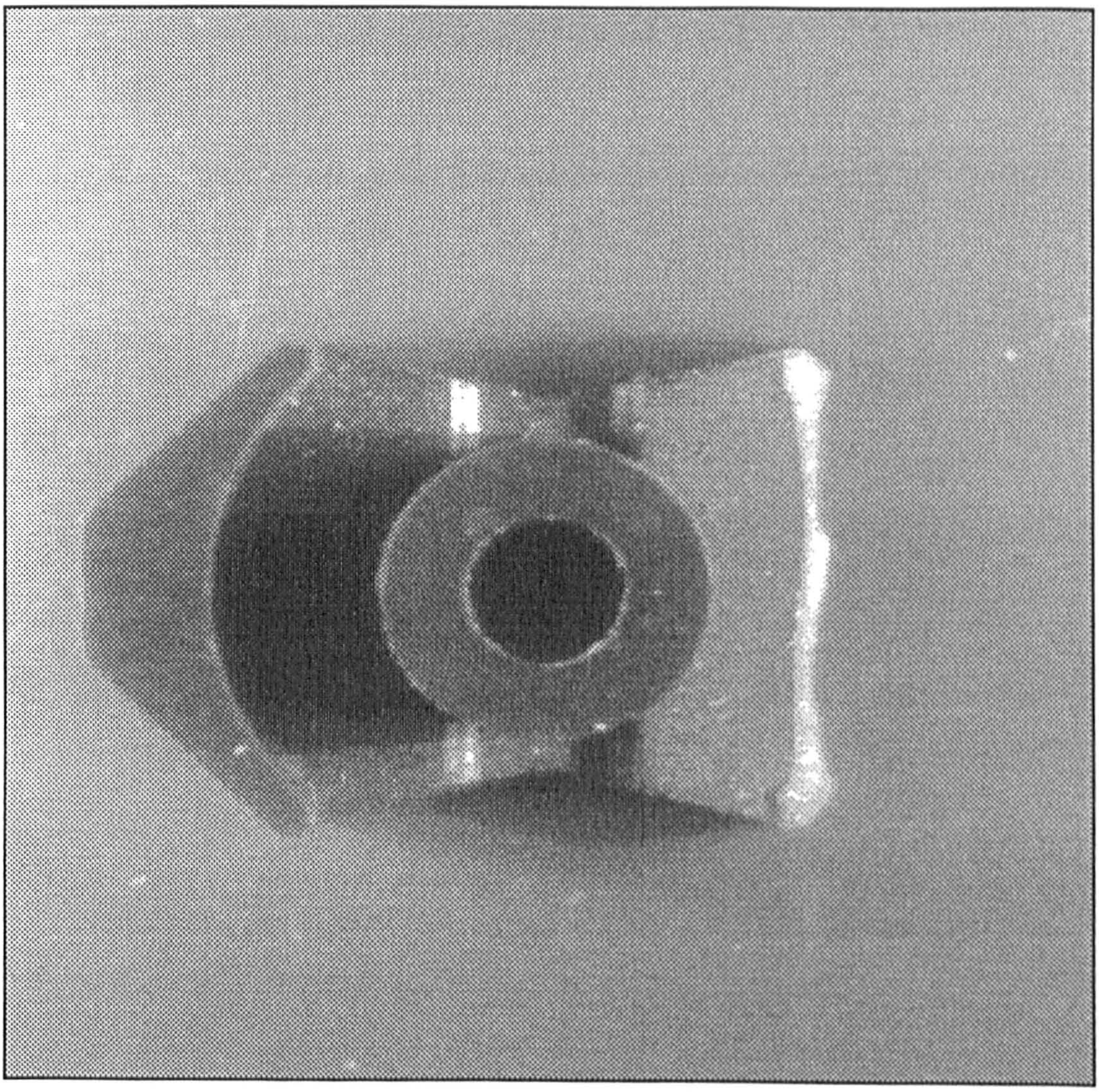
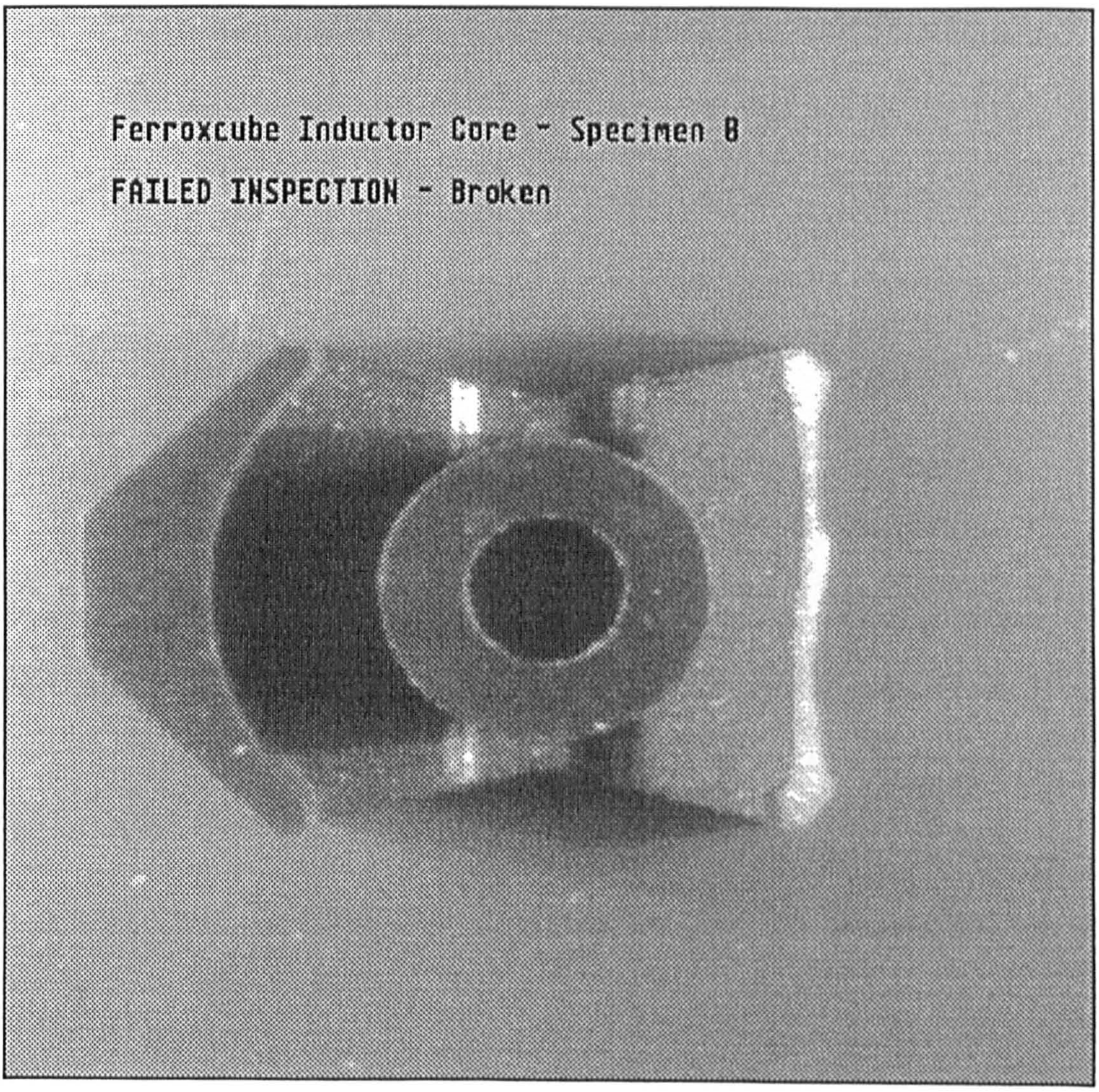


Image 5.22 Test sample 7, unprocessed image

Processed image showing all features and lines


Ferroxcube Inductor Core - Specimen 8
FAILED INSPECTION - Broken

Feature location lines	–	Black, at edge of image
Template edge lines	–	Black
Searched edge lines	–	White
Chipped edges	–	Black
Core centre	–	White
Image centre (256,256)	–	Grey

Image 5.23 Test sample 8, after processing (See Section 5.9.3)

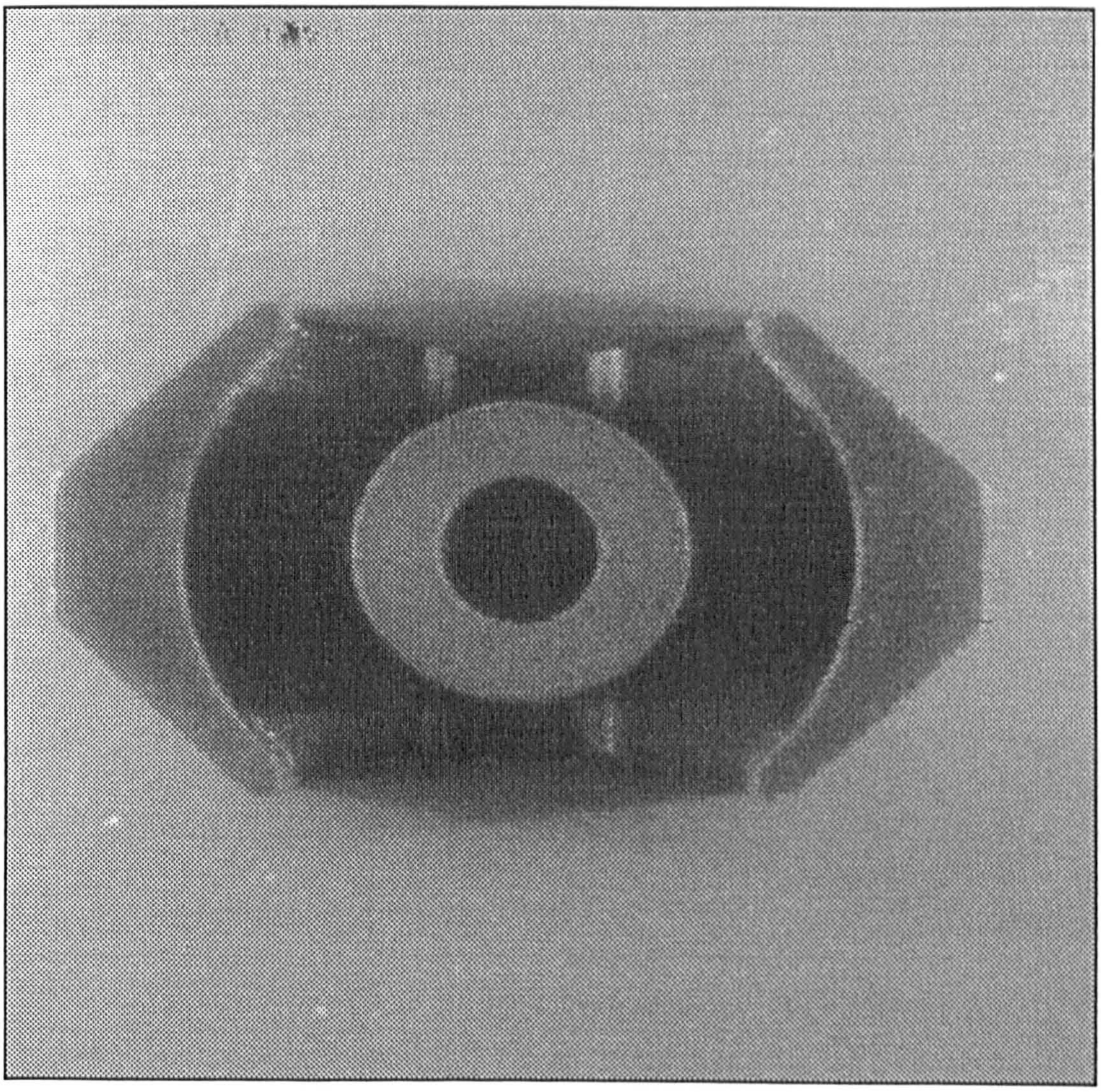
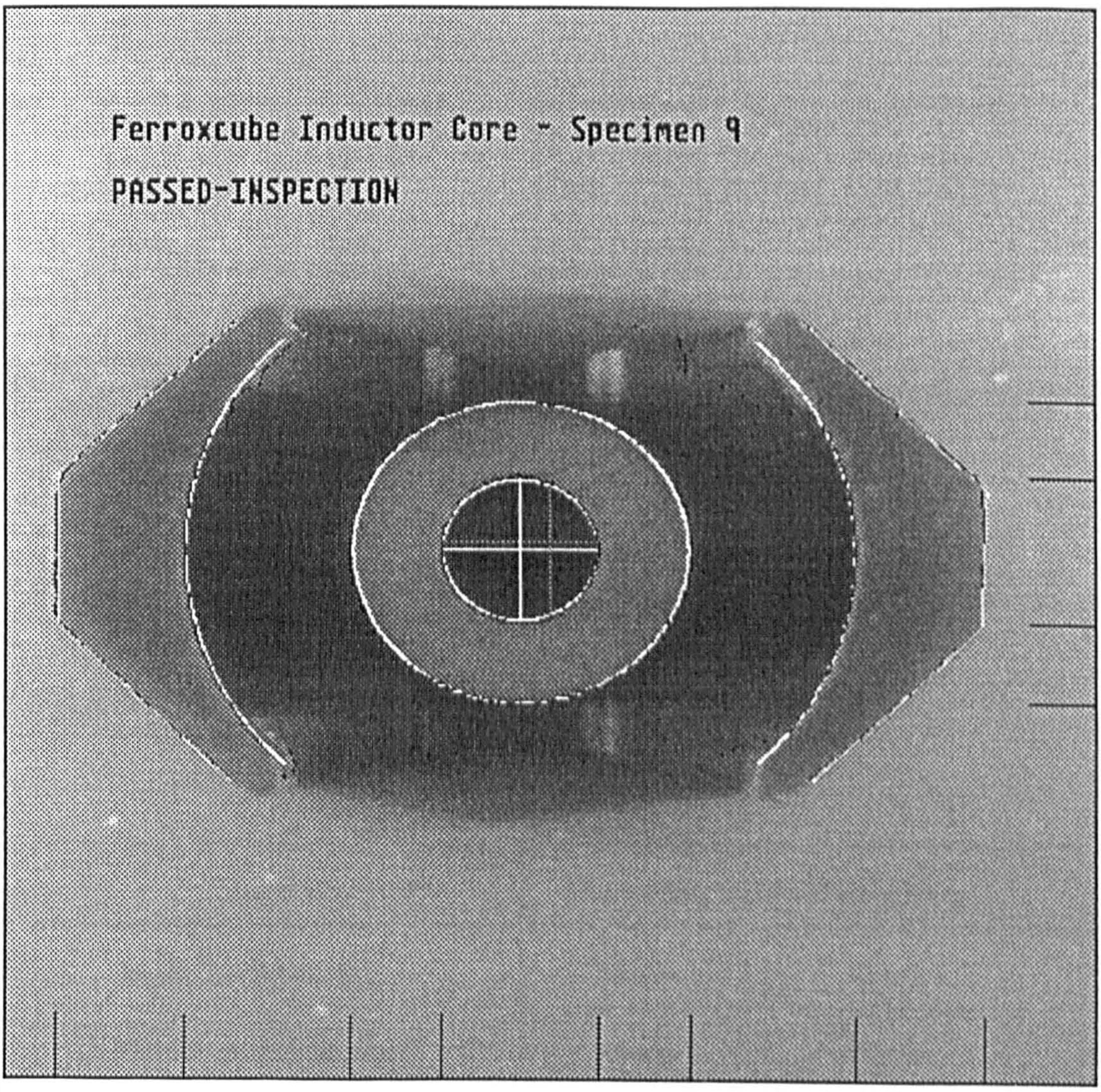


Image 5.24 Test sample 9, unprocessed image

Processed image showing all features and lines



Feature location lines	–	Black, at edge of image
Template edge lines	–	Black
Searched edge lines	–	White
Chipped edges	–	Black
Core centre	–	White
Image centre (256,256)	–	Grey

Image 5.25 Test sample 9, after processing (See Section 5.9.3)

5.8.2 Data

The following pages of data shows the pixel locations and calculated measurements for all the core specimens and corresponds to the images in the previous sections. These results are each for 100 samples with the core held in the same position in the image and show the variation and hence the accuracy of the system. The presented data titles are described below:

Measurements given in pixels

iclx	inner.circle.left.x
icrx	inner.circle.right.x
oclx	outer.circle.left.x
ocrx	outer.circle.right.x
icpy	inner.circle.upper.y
icwy	inner.circle.lower.y
ocpy	outer.circle.upper.y
ocwy	outer.circle.lower.y
iflx	inner.flange.left.x
ifrx	inner.flange.right.x
oflx	outer.flange.left.x
ofrx	outer.flange.right.x
ixa	inner.x.radius
iya	inner.y.radius
oxa	outer.x.radius
oya	outer.y.radius
lfxa	left.flange.x.radius
rfx	right.flange.x.radius
cx	centre.x
cy	centre.y
e	error
n	specimen.number

Measurements given in millimetres

cl	core.length
lfxd	left.flange.x.diameter
rfxd	right.flange.x.diameter
ixd	inner.x.diameter
iyd	inner.y.diameter
oxd	outer.x.diameter
oyd	outer.y.diameter

Error numbers

0	Passed inspection
1	Core is chipped
2	Core is broken

Dimension	Measurement upper and lower limits	
Inside diameter of pillar	3.0 mm	3.1 mm
Outside diameter of pillar	6.2 mm	6.4 mm
Overall length	17.3 mm	17.9 mm
Radii of inner flange walls	12.4 mm	12.9 mm

Table 5.1 Ferroxcube inductor core maximum and minimum dimension

Sample	cl	i.x.d	i.y.d	o.x.d	o.y.d	l.f.d	r.f.d
1	17.8	3.1	3.0	6.3	6.2	12.8	12.8
2	17.8	3.1	3.1	6.4	6.4	12.8	12.8
3	17.6	3.1	3.0	6.3	6.3	13.0	12.6
4	N / A	3.1	3.0	6.3	6.3	13.0	N / A
5	17.7	3.1	3.1	6.4	6.4	13.0	12.8
6	17.7	3.1	3.1	6.4	6.4	12.8	12.8
7	17.5	3.2	3.0	6.3	6.3	12.6	12.8
8	N / A	3.1	3.1	6.3	6.3	12.8	N / A
9	17.6	3.1	3.1	6.4	6.3	12.7	12.6

Table 5.2 Ferroxcube inductor core maximum and minimum dimension

```
Specimen sample - TDATA1.TXT, Images FERROX1R.PIC & FERROX1P.PIC 100 samples
Frequency of occurrence - pixel locations
    iclxicrxoclxocrxicpyicwyocpyocwyiflxifrxoflxofrxixa iya oxa oya lfxarfxacx   cy  e   n
Num 211 286 169 329 234 301 194 339 90  407 30  467 37  33  80  72  159 158 249  267 2   0
Freq29  100 41  99  35  86  78  13  81  100 100 82  100 32  100 80  81  100 100  71  98  100
Num 212     168 328 232 303 196 338 91          468     34      73  158      268 0
Freq71      59  1   41  14  4   77  19          18      59      4   19        26  2
Num                 233     195 341                     35      71            269
Freq                20      16  3                       8       15            2
Num                 235     197 340                     32      70            270
Freq                3       2   7                       1       1             1
Num                 236
Freq                1
Mode
    212 286 168 329 232 301 194 338 90  407 30  467 37  34  80  72  159 158 249  267 98  0
Mean
    212 286 168 329 233 301 194 338 90  407 30  467 37  34  80  72  159 158 249  267
Frequency of occurrence - measurements (mm)
        cl      lfxd    rfxd    ixd     iyd     oxd     oyd
Num     17.71   12.89   12.81   3.00    2.90    6.48    6.33
Freq    82      81      100     100     32      100     80
Num     17.75   12.81                   2.99            6.42
Freq    18      19                      59              4
Num                                     3.08            6.24
Freq                                    8               15
Num                                     2.81            6.15
Freq                                    1               1
Mode
        17.71   12.89   12.81   3.00    2.99    6.48    6.33
Mean
        17.72   12.87   12.81   3.00    2.97    6.48    6.32
```

```
Specimen sample - TDATA2.TXT, Images FERROX2R.PIC & FERROX2P.PIC 100 samples
Frequency of occurrence - pixel locations
    iclxicrxoclxocrxicpyicwyocpyocwyiflxifrxoflxofrxixa iya oxa oya lfxarfxacx   cy  e   n
Num 209 283 165 327 224 293 186 331 88  404 27  465 37  34  81  72  158 158 246  258 2   2
Freq100 100 82  100 78  13  35  46  82  97  100 99  100 90  82  60  83  96  99   86  100 100
Num         166     225 292 187 330 87  405     464     33  80  71  159 159 245  259
Freq        18      8   84  49  39  18  3       1       6   18  35  17  4   1    6
Num                 226 294 188 332                     35      73          260
Freq                1   1   11  11                      4       4           7
Num                 223 295 189 333                             70          257
Freq                13  2   5   4                               1           1
Mode
    209 283 165 327 224 292 187 331 88  404 27  465 37  34  81  72  158 158 246  258 2   2
Mean
    209 283 165 327 224 292 187 331 88  404 27  465 37  34  81  72  158 158 246  258
Frequency of occurrence - measurements (mm)
        cl      lfxd    rfxd    ixd     iyd     oxd     oyd
Num     17.75   12.81   12.81   3.00    2.99    6.57    6.33
Freq    99      83      96      100     90      82      60
Num     17.71   12.89   12.89           2.90    6.48    6.24
Freq    1       17      4               6       18      35
Num                                     3.08            6.42
Freq                                    4               4
Num                                                     6.15
Freq                                                    1
Mode
        17.75   12.81   12.81   3.00    2.99    6.57    6.33
Mean
        17.75   12.82   12.81   3.00    2.99    6.55    6.30
```

```
Specimen sample - TDATA3.TXT, Images FERROX3R.PIC & FERROX3P.PIC 100 samples
Frequency of occurrence - pixel locations
     iclx icrx oclx ocrx icpy icwy ocpy ocwy iflx ifrx oflx ofrx ixa  iya  oxa  oya  lfxa rfxa cx   cy   e    n
Num  211  285  167  327  238  306  200  344  88   405  29   465  37   34   80   72   160  157  248  272  2    3
Freq 97   100  100  100  37   31   75   45   28   100  100  87   99   71   99   52   29   100  100  9    100  100
Num  210                 237  305  199  343  87             464  36   33   79   71   161       271
Freq 3                   52   63   18   39   72             13   1    29   1    47   71        90
Num                      239  307  201  342                                     70                  273
Freq                     9    6    7    14                                      1                   1
Num                      236            345
Freq                     2              2
Mode
     211  285  167  327  237  305  200  344  87   405  29   465  37   34   80   72   161  157  248  271  2    3
Mean
     211  285  167  327  238  305  200  343  87   405  29   465  37   34   80   72   161  157  248  271
Frequency of occurrence - measurements (mm)
          cl        lfxd      rfxd      ixd       iyd       oxd       oyd
Num       17.67     12.97     12.73     3.00      2.99      6.48      6.33
Freq      87        29        100       100       71        100       53
Num       17.63     13.05                         2.90                6.24
Freq      13        71                            29                  47
Num
Freq
Mode
          17.67     13.05     12.93     3.00      2.99      6.48      6.33
Mean
          17.67     13.03     12.73     3.00      2.96      6.48      6.29
```

```
Specimen sample - TDATA4.TXT, Images FERROX4R.PIC & FERROX4P.PIC 100 samples
Frequency of occurrence - pixel locations
     iclx icrx oclx ocrx icpy icwy ocpy ocwy iflx ifrx oflx ofrx ixa  iya  oxa  oya  lfxa rfxa cx   cy   e    n
Num  222  296  201  339  230  301  175  351  181  415  139  477  37   33   82   65   120  149  263  265  1    4
Freq 57   55   9    100  15   93   18   32   64   98   61   59   100  100  100  100  100  100  99   61   100  100
Num  221  295  207       234  303  226  350  139  416  90   476                                     262  266
Freq 43   45   22        79   7    75   65   36   2    11   41                                      1    37
Num            202       233       177  352                111                                      267
Freq           7         1         1    2                  2                                        2
Num            165       236       225  353                83
Freq           35        2         4    1                  23
Num            174       229       228                     142
Freq           1         3         1                       2
Num            200                 227                     138
Freq           9                   1                       1
Num            208
Freq           5
Num            216
Freq           8
Num            193
Freq           1
Num            210
Freq           2
Num            192
Freq           1
Mode
     222  296  165  339  334  301  226  350  181  415  139  477  37   33   82   65   120  149  263  265  1    4
Mean
     222  296  191  339  233  301  216  350  166  415  120  477  37   33   82   65   120  149  263  265
Frequency of occurrence - measurements (mm)
          cl        lfxd      rfxd      ixd       iyd       oxd       oyd
Num       13.70     9.73      12.08     3.00      2.90      6.65      5.71
Freq      29        100       100       100       100       100       100
Num       13.66
Freq      33
Num       15.64
Freq      2
Num       14.79
Freq      1
Num       15.93
Freq      4
Num       15.97
Num       19
Freq      15.68
Num       9
Freq      13.58
Num       2
Freq      14.83
Num       1
Mode
          13.66     9.73      13.08     3.00      2.90      6.65      5.71
Mean
          14.44     9.73      12.08     3.00      2.90      6.65      5.71
```

Specimen sample - TDATA5.TXT, Images FERROX5R.PIC & FERROX5P.PIC 100 samples

```
Frequency of occurrence - pixel locations
     iclx icrx oclx ocrx icpy icwy ocpy ocwy iflx ifrx oflx ofrx ixa  iya  oxa  oya  lfxa rfxa cx   cy   e    n
Num  216  291  172  334  241  310  202  346  95   413  35   473  37   34   81   72   159  159  254  275  2    5
Freq 100  49   100  100  29   99   29   9    96   100  33   33   100  96   100  29   33   33   33   91   100  100
Num       290            242  309  201  347  96        34   472       33        73   158  160  253  276
Freq      51             68   1    70   65   3         67   67        4         71   65   67   67   9
Num                      243       203  348  94                                       157
Freq                     3         1    23   1                                        2
Num                                     349
Freq                                    3
Mode
     216  290  172  334  242  310  201  347  95   413  34   472  37   34   81   73   158  160  253  275  2    5
Mean
     216  290  172  334  242  310  201  347  95   413  34   472  37   34   81   73   158  160  253  275
Frequency of occurrence - measurements (mm)
          cl        lfxd      rfxd      ixd       iyd       oxd       oyd
Num       17.75     12.89     12.89     3.00      2.99      6.57      6.33
Freq      100       33        33        100       96        100       29
Num                 12.81     12.97               2.90                6.42
Freq                65        67                  4                   71
Num                 12.73
Freq                2
Mode
          17.75     12.81     12.97     3         2.99      6.57      6.42
Mean
          17.75     12.83     12.94     3.00      2.99      6.57      6.39
```

Specimen sample - TDATA6.TXT, Images FERROX6R.PIC & FERROX6P.PIC 100 samples

```
Frequency of occurrence - pixel locations
     iclx icrx oclx ocrx icpy icwy ocpy ocwy iflx ifrx oflx ofrx ixa  iya  oxa  oya  lfxa rfxa cx   cy   e    n
Num  226  300  182  344  232  300  174  338  104  422  44   482  37   34   81   82   159  159  263  261  2    6
Freq 99   100  92   100  69   92   24   61   99   23   100  100  99   84   92   16   99   23   100  82   100  100
Num  227       183       231  299  176  337  105  421            36   35   80   81   158  158  260
Freq 1         8         21   8    17   37   1    77             1    7    8    74   1    77   4
Num                      230       175  339                           33        80             263
Freq                     7         56   2                             9         10             1
Num                      233       177                                                         262
Freq                     3         3                                                           13
Mode
     226  300  182  344  232  300  175  338  104  421  44   482  37   34   81   81   159  158  263  261  2    6
Mean
     226  300  182  344  232  300  175  338  104  421  44   482  37   34   81   81   159  158  263  261
Frequency of occurrence - measurements (mm)
          cl        lfxd      rfxd      ixd       iyd       oxd       oyd
Num       17.75     12.89     12.89     3.00      2.99      6.57      7.21
Freq      100       99        23        99        84        92        16
Num                 12.81     12.81     2.92      3.08      6.48      7.12
Freq                1         77        1         7         8         74
Num                                               2.90                7.03
Freq                                              9                   10
Mode
          17.75     12.89     12.81     3.00      2.99      6.57      7.12
Mean
          17.75     12.89     12.83     3.00      2.99      6.56      7.13
```

Specimen sample - TDATA7.TXT, Images FERROX7R.PIC & FERROX7P.PIC 100 samples

```
Frequency of occurrence - pixel locations
     iclx icrx oclx ocrx icpy icwy ocpy ocwy iflx ifrx oflx ofrx ixa  iya  oxa  oya  lfxa rfxa cx   cy   e    n
Num  214  288  170  331  235  304  197  342  94   406  33   466  37   34   80   72   157  155  251  270  0    7
Freq 97   97   10   100  31   77   98   30   100  100  100  99   95   84   100  78   100  100  100  75   19   100
Num  213  287  171       236  306  199  341                 467  36   35        71             269  2
Freq 1    3    90        60   1    2    45                  1    5    9         21             25   80
Num  215                 237  305       340                                33        73                  1
Freq 2                   9    21        22                                 7         1                   1
Num                           307       343
Freq                          1         3
Mode
     214  288  171  331  236  304  197  341  94   406  33   466  37   34   80   72   157  155  251  270  2    7
Mean
     214  288  171  331  236  304  197  341  94   406  33   466  37   34   80   72   157  155  251  270
Frequency of occurrence - measurements (mm)
   cl lfxd  rfxd  ixdiydoxdoyd
Num       17.55     12.73     12.56     3.00      2.99      6.48      6.33
Freq      99        100       100       95        84        100       78
Num       17.59                         2.92      3.08                6.24
Freq      1                             5         9                   21
Num                                               2.90                6.42
Freq                                              7                   1
Mode
          17.55     12.73     12.56     3.00      2.99      6.48      6.33
Mean
          17.55     12.73     12.56     3.00      2.99      6.48      6.31
```

```
Specimen sample - TDATA8.TXT, Images FERROX8R.PIC & FERROX8P.PIC 100 samples
Frequency of occurrence - pixel locations
     iclx icrx oclx ocrx icpy icwy ocpy ocwy iflx ifrx oflx ofrx ixa  iya  oxa  oya  lfxa rfxa cx   cy   e    n
Num  228  327  179  353  245  311  197  362  100  333  79   376  33   30   67   71   173  60   272  278  1    8
Freq 83   98   100  96   98   98   22   24   100  100  100  100  100  100  100  100  100  100  96   7    100  100
Num  227  326       333  247  313  188  363                                                    267  277
Freq 15   2         1    2    2    77   56                                                     1    76
Num  229            341            187  364                                                    269  279
Freq 1              1              1    19                                                     1    17
Num  230 ·          361                 366                                                    274
Freq 1              1                   1                                                      1
Num                 350                                                                        271
Freq                1                                                                          1
Mode
     228  327  179  353  245  311  188  363  100  333  79   376  33   30   67   71   173  60   272  277  1    8
Mean
     228  327  179  353  245  311  190  363  100  333  79   376  33   30   67   71   173  60   272  277
Frequency of occurrence - measurements (mm)
         cl         lfxd       rfxd       ixd        iyd        oxd        oyd
Num      12.04      14.02      4.86       2.67       2.64       5.43       6.24
Freq     100        100        100        100        100        100        100
Mode
         12.04      14.02      4.86       2.67       2.64       5.43       6.24
Mean
         12.04      14.02      4.86       2.67       2.64       5.43       6.24
```

```
Specimen sample - TDATA9.TXT, Images FERROX9R.PIC & FERROX9P.PIC 100 samples
Frequency of occurrence - pixel locations
     iclx icrx oclx ocrx icpy icwy ocpy ocwy iflx ifrx oflx ofrx ixa  iya  oxa  oya  lfxa rfxa cx   cy   e    n
Num  205  279  162  322  224  294  189  330  84   399  24   459  37   35   80   70   158  157  242  260  0    9
Freq 100  100  100  100  21   68   9    50   99   93   100  92   100  35   100  5    62   60   61   44   97   100
Num                      226  295  188  331  83   398       458       34        71   157  158  241  259  2
Freq                     39   30   90   43   1    7         8         65        87   38   36   39   56   3
Num                      225  297  187  333                                     72        156
Freq                     39   1    1    2                                       8         4
Num                      227  293       332
Freq                     1    1         5
Mode
     205  279  162  322  225  294  188  330  884  339  24   459  37   34   80   71   158  157  242  259  0    9
Mean
     205  279  162  322  225  294  188  331  84   399  24   459  37   34   80   71   158  157  242  259
Frequency of occurrence - measurements (mm)
         cl         lfxd       rfxd       ixd        iyd        oxd        oyd
Num      17.63      12.81      12.73      3.00       3.08       6.48       6.15
Freq     92         62         64         100        35         100        5
Num      17.59      12.73      12.64                 2.99                  6.24
Freq     8          38         36                    65                    87
Num                                                                        6.33
Freq                                                                       8
Mode
         17.63      12.81      12.73      3.00       2.99       6.48       6.24
Mean
         17.63      12.78      12.70      3.00       3.02       6.48       6.24
```

```
Specimen sample - TDATA1M.TXT, Image FERROX1M.PIC 100 samples
Frequency of occurrence - pixel locations
     iclx icrx oclx ocrx icpy icwy ocpy ocwy iflx ifrx oflx ofrx ixa  iya  oxa  oya  lfxa rfxa cx   cy   e    n
Num  217  291  173  334  225  293  186  329  96   412  35   472  37   34   80   71   158  158  254  258  2    1
Freq 54   100  80   100  64   89   95   45   73   92   100  83   100  81   100  44   73   92   100  64   97   100
Num  216       174       227  295  188  330  95   411       473       33        72   159  157   259 0
Freq 46        20        6    11   3    49   27   8         17        16        53   27   8     33 3
Num                      224       187  331                           35        70              260
Freq                     12        2    6                             3         3               3
Num                      226
Freq                     18
Mode
     217  291  173  334  225  293  186  330  96   412  35   472  37   34   80   72   158  158  254  258  2    1
Mean
     217  291  173  334  225  293  186  330  96   412  35   472  37   34   80   72   158  158  254  258
Frequency of occurrence - measurements (mm)
         cl         lfxd       rfxd       ixd        iyd        oxd        oyd
Num      17.71      12.81      12.81      3.00       2.99       6.48       6.24
Freq     83         73         92         100        81         100        44
Num      17.75      12.89      12.73                 2.90                  6.33
Freq     17         27         8                     16                    53
Num                                                  3.08                  6.15
Freq                                                 3                     3
Mode
         17.71      12.89      12.81      3.00       2.99       6.48       6.33
Mean
         17.72      12.83      12.80      3.00       2.98       6.48       6.28
```

```
Specimen sample - TDATA1I.TXT, Image FERROX1I.PIC 100 samples
Frequency of occurrence - pixel locations
    iclxicrxoclxocrxicpyicwyocpyocwyiflxifrxoflxofrxixa iya oxa oya lfxarfxacx   cy  e   n
Num 227 301 183 344 196 264 157 302 106 422 44  482 37  34  80  72  157 159 263  228 2   1
Freq77  100 100 100 100 95  93  45  33  73  100 100 100 95  100 92  33  73  100  31  100 100
Num 226                 263 158 301 105 421             33      73  158 158  229
Freq23                  5   1   52  67  27              5       4   67  27   45
Num                         159 303                             71           230
Freq                        4   2                               4            22
Num                         156 305                                          232
Freq                        2   1                                            1
Mode
    227 301 183 344 196 264 157 302 105 422 44  482 37  34  80  92  67  159 263  229 2   1
Mean
    227 301 183 344 196 264 157 302 105 422 44  482 37  34  80  72  158 159 263  229
Frequency of occurrence - measurements (mm)
        cl      lfxd    rfxd    ixd     iyd     oxd     oyd
Num     17.75   12.73   12.89   3.00    2.99    6.48    6.33
Freq    100     33      73      100     95      100     92
Num             12.81   12.81           2.90            6.42
Freq            67      27              5               4
Num                                                     6.24
Freq                                                    4
Mode
        17.75   12.73   12.89   3.00    2.99    6.48    6.33
Mean
        17.75   12.78   12.87   3.00    2.99    6.48    6.33
```

```
Specimen sample - TDATA10.TXT, Image FERROX10.PIC 100 samples
Frequency of occurrence - pixel locations
    iclxicrxoclxocrxicpyicwyocpyocwyiflxifrxoflxofrxixa iya oxa oya lfxarfxacx   cy  e   n
Num 200 274 157 317 244 311 204 348 79  395 18  455 37  33  80  72  158 158 237  277 2   1
Freq98  100 100 83  73  97  95  99  100 100 100 43  100 74  100 96  100 100 100  74  93  100
Num 199         318 243 313 206 350             456     34      71           279 0
Freq2           17  23  3   1   1               57      26      4            1   7
Num                 246     205                                              278
Freq                1       4                                                25
Num                 242
Freq                1
Num                 245
Freq                2
Mode
    200 274 157 317 244 311 204 348 79  395 18  456 37  33  80  72  158 158 237  277 2   1
Mean
    200 274 157 317 244 311 204 348 79  395 18  456 37  33  80  72  158 158 237  277
Frequency of occurrence - measurements (mm)
        cl      lfxd    rfxd    ixd     iyd     oxd     oyd
Num     17.71   12.81   12.81   3.00    2.90    6.48    6.33
Freq    43      100     100     100     74      100     96
Num     17.75                           2.99            6.24
Freq    57                              26              4
Mode
        17.75   12.81   12.81   3.00    2.90    6.48    6.33
Mean
        17.73   12.81   12.81   3.00    2.92    6.48    6.33
```

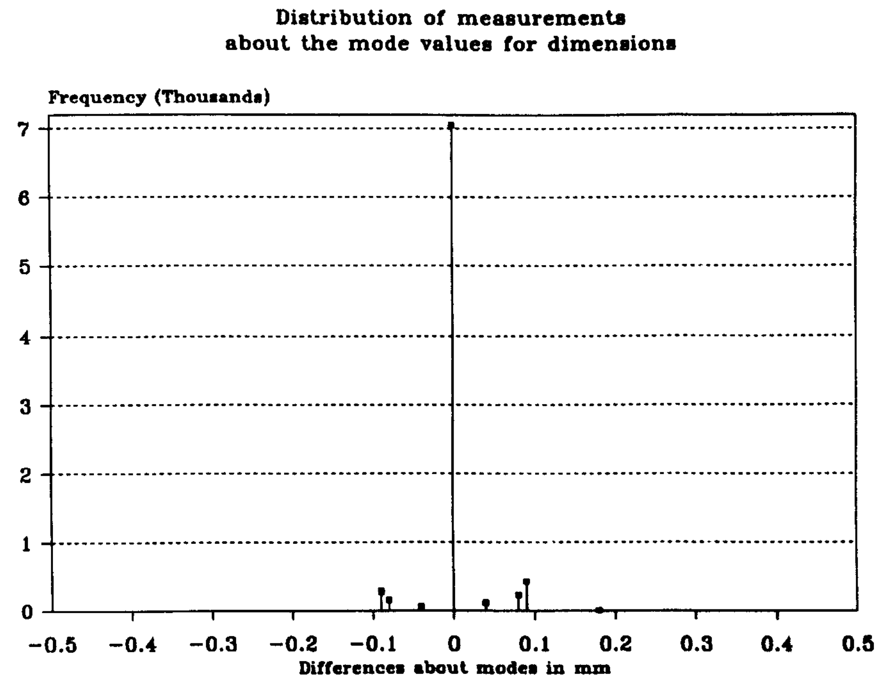
```
Specimen sample - TDATA9M.TXT, Image FERROX9M.PIC 100 samples
Frequency of occurrence - pixel locations
    iclxicrxoclxocrxicpyicwyocpyocwyiflxifrxoflxofrxixa iya oxa oya lfxarfxacx   cy  e   n
Num 219 292 176 336 222 291 185 328 98  412 38  472 37  34  80  71  157 157 255  256 0   9
Freq89  100 98  99  59  19  38  18  71  39  100 81  100 78  100 75  29  48  59   77  87  100
Num 218     175 337 221 290 184 327 97  413     473     35      70  158 158 256  255 2
Freq11      2   1   38  68  55  61  29  61      19      21      13  71  41  41   16  13
Num                 223 292 186 326                     33      72      156  257
Freq                3   11  4   16                      1       12      11   7
Num                     293 187 329
Freq                    2   3   5
Mode
    219 292 176 336 222 290 184 327 97  413 38  472 37  34  80  71  158 157 255  256 0   9
Mean
    219 292 176 336 222 290 185 327 97  413 38  472 37  34  80  71  158 157 255  256
Frequency of occurrence - measurements (mm)
        cl      lfxd    rfxd    ixd     iyd     oxd     oyd
Num     17.59   12.73   12.73   3.00    2.99    6.48    6.24
Freq    81      29      59      100     78      100     75
Num     17.63   12.81   12.64           3.08            6.15
Freq    19      71      41              22              13
Num                                                     6.33
Freq                                                    12
Mode
        17.57   12.01   12.73   2.92    2.99    6.48    6.24
Mean
        17.60   12.80   12.75   2.96    3.01    6.48    6.24
```

```
Specimen sample - TDATA9I.TXT, Image FERROX9I.PIC 100 samples
Frequency of occurrence - pixel locations
     iclx icrx oclx ocrx icpy icwy ocpy ocwy iflx ifrx oflx ofrx ixa  iya  oxa  oya  lfxa rfxa cx   cy   e    n
Num  236  310  193  353  198  268  161  303  115  429  55   490  37   35   80   71   157  157  272  232  0    9
Freq 100  100  100  97   48   38   54   68   98   100  88   99   100  38   100  67   75   77   77   82   89   100
Num                 354  197  267  162  304  114       56   489       34        70   158  156  273  233  2
Freq                3    25   54   39   29   2         12   1         62        30   25   23   23   18   11
Num                      199  266  160  305                                     72
Freq                     26   8    5    3                                       3
Num                      201       163
Freq                     1         2
Mode
     236  310  193  351  198  267  161  303  115  430  55   490  37   34   80   71   157  158  272  232  0    9
Mean
     236  310  193  353  198  267  161  303  115  430  55   490  37   34   80   71   157  158  272  232
Frequency of occurrence - measurements (mm)
          cl       lfxd     rfxd     ixd      iyd      oxd      oyd
Num       17.63    12.73    12.64    3.00     3.08     6.48     6.24
Freq      87       75       23       100      38       100      67
Num       17.59    12.81    12.73             2.99              6.15
Freq      13       25       77                62                30
Num                                                             6.33
Freq                                                            3
Mode
          17.63    12.73    12.81    3.00     2.99     6.48     6.24
Mean
          17.62    12.75    12.79    3.00     3.02     6.48     6.22
```

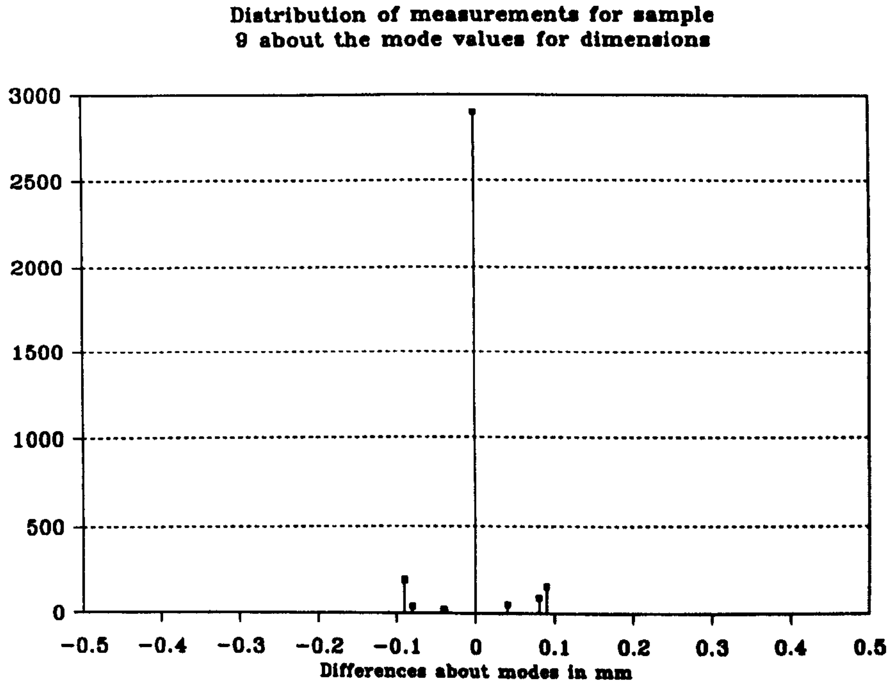
```
Specimen sample - TDATA90.TXT, Image FERROX90.PIC 100 samples
Frequency of occurrence - pixel locations
     iclx icrx oclx ocrx icpy icwy ocpy ocwy iflx ifrx oflx ofrx ixa  iya  oxa  oya  lfxa rfxa cx   cy   e    n
Num  198  272  155  315  240  310  203  345  76   391  17   451  36   35   80   71   159  156  235  274  0    9
Freq 100  100  100  100  78   25   72   55   70   67   98   33   58   24   100  80   15   17   19   46   96   100
Num                      242  309  204  346  77   392  16   452  37   34        72   158  157  234  275  1
Freq                     1    70   15   41   30   33   2    67   42   76        9    59   83   81   50   1
Num                      241  308  202  347                                     70   157        276 2
Freq                     15   3    9    4                                       11   26          4  3
Num                      243  311  205
Freq                     2    2    4
Num                      239
Freq                     4
Mode
     198  271  155  315  240  309  203  345  76   391  17   452  36   34   80   71   158  157  234  275  0    9
Mean
     198  271  155  315  240  309  203  345  76   391  17   452  36   34   80   71   158  157  234  275
Frequency of occurrence - measurements (mm)
          cl       lfxd     rfxd     ixd      iyd      oxd      oyd
Num       17.59    12.89    12.64    3.00     3.08     6.48     6.24
Freq      32       15       17       100      24       100      80
Num       17.63    12.81    12.73             2.99              6.33
Freq      68       59       83                76                9
Num                12.73                                        6.15
Freq               26                                           11
Mode
          13.63    12.81    12.73    2.92     2.99     6.48     6.24
Mean
          17.61    12.80    12.74    2.95     3.01     6.48     6.24
```

5.8.3 Graphs from the test specimens

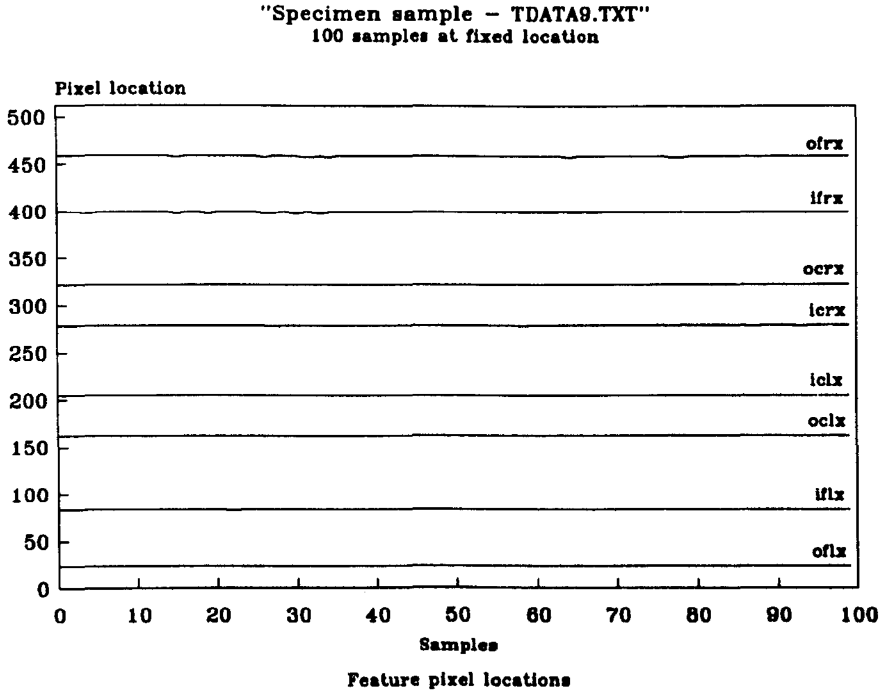
Graphs 5.1 and 5.2 show the distributions of feature pixel locations and measurements for all non-broken specimens. Graphs 5.3 and 5.4 show the variation of the feature pixel locations and measurements over the 100 samples which were taken for each specimen.



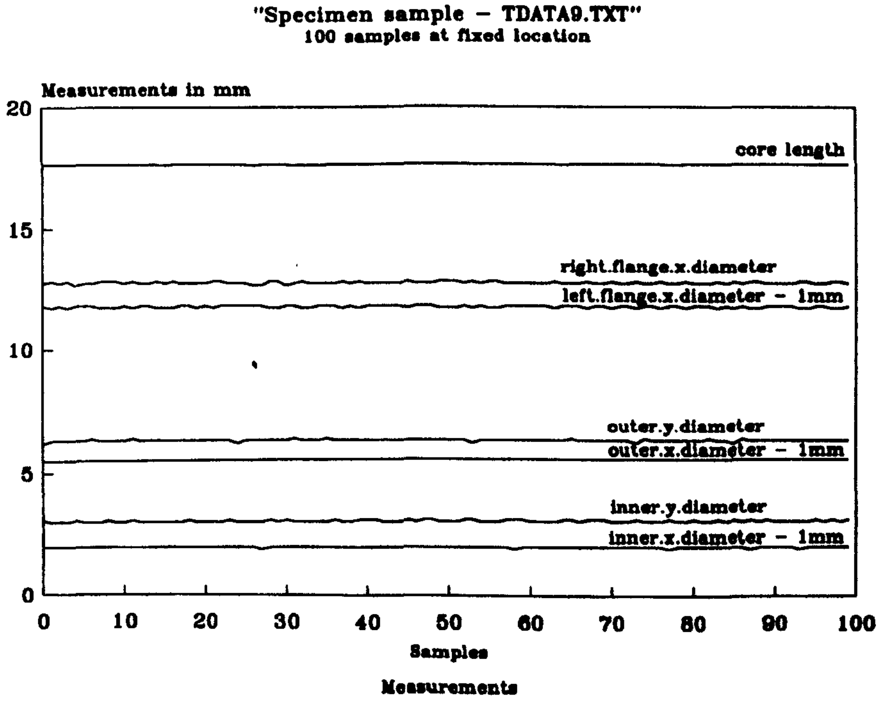
Graph 5.1 Frequency distributions of differences of all measurements and dimensions (excluding broken and badly damaged specimens 4, 6 & 8)



Graph 5.2 Frequency distributions of differences of measurements and dimensions of all samples of specimen 9



Graph 5.3 Pixel variation over 100 samples with inductor core in the same location



Graph 5.4 Measurement variation over 100 samples with inductor core in the same location

5.9 Timings of software

The main parts of the software have been timed to show where much of the time is spent. Some of these main parts have had their sub-parts timed.

Fold name	T425-20 (secs)
find centre points	0.0076
look for horizontal edges	0.0038
find inner.circle.left.x	0.0009
find inner.circle.right.x	0.0010
look for vertical edges	0.0038
find inner.circle.upper.y	0.0009
calc centre points	0.00006
find edge features	0.01
look for horizontal circle edges	0.0071
find inner.circle.left.x	0.0005
find outer.circle.left.x	0.0008
find inner.flange.left.x	0.0013
find inner.flange.left.x	0.0008
find inner.circle.right.x	0.0007
find outer.circle.right.x	0.0008
find inner.flange.right.x	0.0013
find inner.flange.right.x	0.0008
look for vertical circle edges	0.0029
find inner.circle.upper.y	0.0006
find outer.circle.upper.y	0.0008
find inner.circle.lower.y	0.0006
find outer.circle.lower.y	0.0009
calc inductor core radii	0.00006
search edges	0.225
search flange edges	0.134
search left flange lower outer edge	0.0167
search left flange upper outer edge	0.0171
search right flange lower outer edge	0.0166
search right flange upper outer edge	0.0167
search circle edges	0.091
search left circle lower outer edge	0.0072
search left circle lower inner edge	0.0034
search left circle right outer edge	0.0084
search left circle right inner edge	0.0038
find equations of the flange outer edges	0.0052
left.flange.lower.edge	0.0011
left.flange.middle.edge	0.0005
compare with edges to find chips	0.25
inner circle	0.0351
outer edge	0.0857
left flange arc	0.0568
right flange arc	0.0571
flange outer edges	0.0112

6. Chapter 6 - Conclusions and future work

This thesis has shown that by using carefully designed algorithms on fast processors, image processing is now usable by industry for automatic measurement and inspection. The advantages of using visual techniques are many fold, not least that it is often the most natural method, since man uses his eyes as his primary sense when inspecting his surroundings. The use of computerised vision for inspection, improves on man's qualitative sight by allowing an immediate quantitative visual inspection. This thesis demonstrates that a weld's profile may be completely measured in about half a second where it would take a man to measure the same dimensions more than 100 times longer. Furthermore, a complex component such as the ferroxcube inductor core, can be both reliably measured and inspected for chipped edges in a fraction of the time of the manual inspection. Some of the dimensions of this component and of the weld's profile are now measurable, but with the manual gauges they prove very difficult, if not impossible, to measure. The computerised vision system also permits inspection and measurement in hostile environments, where man is reluctant, or at risk, to venture.

6.1 The portable weld measurement gauge

Weld measurement has previously relied on the judgement of the inspector, this was due to four main reasons. Firstly, it was difficult to get the current tools into the weld joint to measure at the desired point. Secondly, they only produced a point measurement and not a complete profile and therefore relied on their skilled use. Thirdly, they can only be used on fillet or butt welds, since these have defined reference datums from which the measurements can be made. The previous gauges are not accurate enough to measure small features such as undercut. An automatic gauge eliminates the inconsistency in inspector's judgement because all measurements taken are based on quantitative techniques.

The Welding Institute was interested in investigating whether a gauge could be designed which would measure both a weld's profile automatically and to provide a paper print out of the results as a permanent record. The important measurements of a welded joint's profile are reinforcement, width, undercut, toe angles of the weld, displacement and angle between the plates. Discussions with representatives of the welding industry refined and added to these requirements, in particular the device should be able to store its data for later recovery and it should be able to monitor continuously. It should be light-weight and robust, non-contacting and non-destructive to the inspected work, as easy and safe to operate as possible, and reliable and consistent in its measurements. All of these features and requirements have been achieved in the portable gauge.

This research has formally defined the weld features in a quantitative way by establishing a datum for all weld types. This datum is universal, and overcomes the problems associated with the measurement of welds made between plates at any angle other than fillet or butt welds. Generic algorithms have been developed to find and measure a weld's features. These algorithms while shown here in a portable gauge, could be implemented for on-line weld control systems.

Continuous operation of the gauge required that it should both capture and process each inspection fast enough for on-line use. This has led to the choice of both the laser -stripe technique and transputer technology. Satisfactory operation of this arrangement led naturally to consideration of using the device to monitor and possibly control the weld as it was being made. The system has been demonstrated performing this function and an automatic continuous welding machine can be further developed using this approach.

The gauge allows remote weld measurement and inspection in areas where people are unable to go. The monitoring head can be redesigned to make it smaller, cameras are now available at a reasonable cost which are less than $^1/_4$ of the size of that used presently. Furthermore, by use of coherent fibre optic bundles a tiny head could be produced which would be able to fit into very confined spaces, such as pipes or welds between plates at acute angles. The gauge could become the source for a range of weld measuring instruments and automatic welding tools.

For this research to be of value to industry, the measurements must be proved to be accurate. The gauge has therefore been thoroughly tested with not only the Welding Institute's test samples, but with many different weld specimens. The Welding Institute has had opportunity to test and assess the gauge over a period of six months. Stephens and Bourton [66] conclude their report on their evaluation of portable gauge, for The Welding Institute, by saying "The gauge has performed to a high enough standard in field tests to suggest that it is capable of being manufactured and marketed commercially." In trials they found that one project's measurements, which would manually have taken over two weeks, were measured in less than an afternoon, and at greater accuracy.

The natural progression arising from the research into measurement of weld profiles, is to use the techniques in this thesis to control the welding process as welding is performed. By using many processors to perform the image processing tasks the speed can be increased to real-time, and the algorithms used in this research provide a reliable and tested foundation on which to investigate controlling the welding process.

6.2 On-line metrology and flaw inspection

The application of image processing to the inspection of components on a production line has been considered using the Philips Ferroxcube inductor cores as a case study example. Previously samples were inspected manually, but automated inspection was required to provide 100% testing while increasing the production rate by 600%. This research has resulted in a process which will not only measure all the parameters currently inspected manually, but is also capable of measuring many more parameters recording the results for post-analysis. This information could be used in the future to correlate with the batch and material source information enabling Philips to introduce production control in the manufacturing process.

The system which has been devised can locate positions, define edge positions, measure dimensions to the required degree of accuracy, calculate area and determine shape.. The system makes an overall pass/fail decision which can be output to a screen, passed to other devices such as a robot, or logged for further analysis. Development of the system in the future to include an interface to the robotic machine is required to provide a complete on-line quality control system. The system could be further extended to measure cores lying at any angle in the image. At present the orientation of the cores as they pass beneath the camera must be constant and preset. Such flexibility would greatly aid the manufacturing process and eliminate the cost of arranging for the cores to be similarly orientated.

6.3 Accuracy

The measurement of dimensions in any stationary object, when viewed by a camera, will have an accuracy which is dependent upon the proportion of the screen which that dimension fills. For a 512 x 512 pixel screen, the object can be made to fill 500 pixels and therefore, the least error possible from this image for its largest dimension would be 1 part in 250, or 0.4% (1 pixel at each end). A typical dimension may however occupy only 200 pixels, and therefore have an error of 1%. The error is independent of the actual size of the object since the image will have been magnified to fill the major part of the screen and then calibrated using a standard sample. Clearly the accuracy will diminish for features which are small compared with the size of the whole image. A vision system can offer better accuracy than is obtained with normal measuring instruments used in industrial inspection.

There are cameras which have up to 1024 x 1024 pixels which are capable of resolving images with greater acuity and enabling measurement with higher accuracy than standard cameras. Cameras with a faster frame scan rate may also become available, which will provide images for the vision system more rapidly; in this case the actual real-time limit is reduced. The need for higher frame scan cameras has not previously been sufficient for them to have been developed, but the advent of industrial image processing may create that need.

6.4 Cost considerations

Cost is always a relevant consideration in engineering. In the preliminary "First Year Report" (1988) it was shown that the cost of components to produce the necessary hardware for the parallel processing transputer based weld gauge was considerably less at that time (£1570) than if the system had been devised to run on either a portable IBM-AT clone using a PIP frame store (£6400), or a bespoke system based on the 80286 processor (£3370). While the cost of some of these components have fallen during the past two or more years, not least the IBM-AT clone, there is no reason to doubt that the cost of producing the hardware for the transputer based vision system is significantly cheaper than the other methods considered.

The greater cost in any commercial system would be for developing the necessary software and in proving the system. There is the prospect that faster and more powerful processors will become available in the near future which will permit the reduction of the number which are needed to work in parallel to achieve the same task. These new devices will permit more complex analysis of the image to be achieved by fewer processors. Therefore even if the cost of this new generation of processors is higher, cost savings in hardware can be expected by the need for fewer processors.

6.5 Future work

The case studies have shown that image processing for industrial inspection and measurement is set for great advances in the future, and its use for on-line and real-time operation is likely. The work undertaken and described in this thesis shows that the combination of parallel processing with specific software has the realistic potential for development into a wide range of automatic measurement instruments and industrial processes.

Research into the application of image processing to more complex tasks is to be pursued. The greatly increased processing power offered by TIPS in combination with specific image processing techniques will allow inspection from images of poor quality. In this direction the research into weld inspection is to be extended, through the use of x-rays, to check welds for sub-surface defects such as cracks.

The sub-sea environment offers many opportunities for vision systems to aid or replace unpleasant and dangerous manual tasks through coupling with intelligent underwater robots. In this case the raw images are of poor quality, with features being temporarily obscured by cloudy water, bubbles and drifting debris. Processing is needed to improve each image with knowledge acquired from previous images, as well as to extract the required features and information. Research is currently starting to tackle some of the problems associated with sub-sea robotic vision and will be looking into interpreting multiple images to build a three dimensional model of the scene in view.

References

[1] P.B.Terry, 'An optical automated weld profile monitor', Ph.D. Thesis, University of Liverpool, 1987

[2] PIP-1024, Matrox Electronic Systems, 1055 St. Regis Blvd., Dorval, Canada

[3] Hedengren K.H.,'Advances in machine vision: Methodology for automatic, image-based inspection of industrial objects', ch. 4, pp. 160-191, Springer-Verlag, 1989, ISBN 0-387-96822-9

[4] Clark S., Lucas J., Parker A.B., 'Seam tracker for TIG welding' IEE Proceedings, Vol. 132, No. 4, July 1985, pp.164-167

[5] Morgan A.D. *et al*, 'Third conference on image processing and its applications: Image processing for robot road following', pp. 421-425, IEE, July 1989, ISBN 0-85296382-3

[6] Stout K., 'Quality control in automation', ch. 1, pp. 11-13, Kogan page, 1985, ISBN 0-85038-936-4

[7] BS5750 'Quality systems', Parts 0-4, British Standards Institution, 1987 (Part 4, 1990)

[8] Gonzalez R.C. & Wintz P., 'Digital image processing' pp. 1-10, Addision Wesley, 1987, ISBN 0-201-11026-1

[9] Cole H.A., 'Basic colour television', Oxford Technical Press, 1983, ISBN 0-29139641-0

[10] Norton P., 'Programmers guide to the IBM PC', ch. 8, pp. 159-240, Microsoft Press, 1985, ISBN 0-941845-46-2

[11] 'IC Memory Data Book', Hitachi Ltd, 1987, Book No. PCBICMEM/8709

[12] 'The VMEbus specification', Rev C.1, Motorola, 1985

[13] 'MC68020, Enhanced 32 bit microprocessor user manual', Motorola, 1987

[14] 'TM-765 High resolution CCD shutter camera', Pulnix America Inc., Unit 5, Intec 2, Wade Rd., Basingstoke, UK, Doc.No. VSI/D/989/745

[15] 'PIP-640', Matrox Electronic Systems, 1055 St. Regis Blvd., Dorval, Canada

[16] Niblack W., 'An introduction to digital image processing', ch. 2, pp. 23-68, Prentice Hall International, 1986, ISBN 0-13-480674-3

[17] King G.J., 'Beginners guide to colour television', pp. 24-36, Newnes-Butterworths, 1973, ISBN 0-408-00101-1

[18] Stafford R.H., 'Digital Television: Bandwidth reduction and communication aspects', pp. 175-178, John Wiley & Sons, 1980, ISBN 0-471-078-57-3

[19] 'Technical reference', IBM personal computer reference library, IBM, 1984

[20] 'Multibus I, Architecture reference book', Intel International (UK) Ltd, Swindon, Wilts, Doc.No. 210883-002

[21] 'Micro computer products for OEM's & system integrators', Data Translation Ltd., Molly Millar's Lane, Wokingham, Berks, UK

[22] 'Micro computer vision systems', Digithurst Ltd., 7 Church Lane, Royston, Herts, UK

[23] 'PIPE: Parallel Image Processing Engine', Akebia Ltd., Lever House, 3 St. Jame's Road, Kingston upon Thames, Surrey, UK

[24] Chambers S.P., 'TIPS: A transputer based real-time vision system', Ph.D. Thesis, University of Liverpool, 1990

[25] '19" Systems for electronic packaging: 19" systems specification', Schroff UK Ltd., Maylands Avenue, Hemel Hempstead, Herts, UK, 1989, Doc.No. GBIA-G2/88 2/3.5

[26] EIA-STD-RS170, E.I.A. Standard

[27] 'The Transputer Data Book', (Second Edition), Inmos Ltd., 100 Aztec West, Almondsbury, Bristol, UK, 1989, Doc.No. 72 TRN 203 01

[28] Hoare C.A.R. (Ed.), 'Occam 2 Reference manual', Prentice Hall, 1988, ISBN 0-13-629-312-3

[29] O'Gorman C., 'A note for histogram equalisation for optimal intensity range utilisation', Computer Vision, Graphics, and Image Processing, 41, 2, February 1988, pp. 229-232

[30] Sahoo *et al*, 'A survey of thresholding techniques', Computer Vision, Graphics, and Image Processing, 41, 1988, pp. 233-260

[31] Beattie R.J., 'Threshold selection in the presence of noise', IEE, 2nd. International Conference on Image Processing and its Applications, No. 265, 1986, pp. 112-116

[32] James V., 'How grey scale imaging improves vision system performance', SPIE, VOL. 654, Automatic optical inspection 1986, pp. 123-124

[33] Kitler J., 'On the accuracy of the Sobel edge detector', Image and Vision Computing, Vol. 1 No. 1, February 1983, pp. 37-42

[34] Castleman K.R., Digital Image Processing, Prentice Hall, 1982

[35] Hitachi, Model HV-730, CCTV camera operation manual, Hitachi Denshi (UK) Ltd.

[36] Thomas.B.T. & Finney R.L., 'Calculus and analytic geometry' (Sixth edition), pp. 884-887, Addison-Wesley, 1984, ISBN 0-201-16309-8

[37] AWS D1.2, 'Structural welding code for aluminium', Section 6: Inspection, pp. 13-14, Appendix C: Terms and definitions, pp. 33-36, American Welding Society

[38] BS499, 'Welding terms and symbols', British Standards Institution, Part 1, Section 6, Terms relating to weld imperfections, 1983

[39] Norman E.W.L., "The Welding Institute gauge", Internal report to the Welding Institute, March 1980

[40] Dunkerton S.B., "Welding gauge - a contract proposal", Internal report to the Welding Institute, 1980

[41] Blakeley P., "The Welding Institute welding gauge", Internal report to the Welding Institute, 1980

[42] Crawford D.G., "Welding Institute gauge - proposals", Internal report to the Welding Institute, 1980

[43] AWS D1.1, 'Structural welding code for steel', Section 6: Inspection, pp. 89-91, Appendices A and B: Effective throats for fillet welds, American Welding Society

[44] BS4870:1981, 'Approval testing of welding procedures', British Standards Institution, Part 1: Fusion welding of steel, Part 2: TIG or MIG welding of aluminium and its alloys

[45] BS5289:1986, 'Visual inspection of fusion welded joints', British Standards Institution, Code of practice, 1986

[46] AWS B1.10-86, 'Guide for the non-destructive inspection of welds', American Welding Society

[47] AWS B1.11-88, 'Guide for the visual inspection of welds', American Welding Society

[48] BNFL, 'Fabrication of stainless steel', Section 5.3, Visual inspection, Doc.No. NF0081/1, Issue 3, October 1985

[49] Weld-link, 'Defects in fusion welds' pp.47-58, 63-64, 75-76, Pub. Crown, 1987

[50] IIW, 'Workmanship tolerances and permissible weld discontinuities', Welding in the World, Vol 21, No. 5/6 pp. 90-110, 1983

[51] Weymueller C R, 'Know your welding NDT - Visual inspection', Welding Design and Fabrication, March 1984, pp. 93-101

[52] Timberlake D R, 'Revised visual inspection criteria require sound justification', Welding Journal, February 1988, pp. 38-41

[53] Watson M.N., "Design of a replacement welding gauge", Internal report to the Welding Institute, February 1980

[54] Slater G., "WI welding gauge", Internal report to the Welding Institute, January 1980

[55] Weld inspection replicas, The Welding Institute, Abington, Cambridge

[56] 'WEVO weld gauge apparatus', Biltse Instr Fabr b.v., RODEM b.v., Utrechtseweg 364, 3731 GE DEBILT, The Netherlands.

[57] MVS, Modular Vision Systems Inc., 3195 De Miniac, Montreal, Canada

[58] RS232, EIA standard RS232

[59] Datac plc., 'Datac 164SB Printer Controller manual', pp. 1-8, Atlantic Street, Altringham, Cheshire, UK, July 1989, Doc.No. DS8017

[60] Hitachi, 'LCD controller/Driver LSI', pp. 450-502, Hitachi Ltd., 1984, Doc.No. DBLCD/88/12

[61] Norbain Security, 'Philips MDC 100-8', Norbain House, Boulton Road, Reading, UK

[62] LT026MD0,Sharp Laser Diode, Sharp Corporation, Japan

[63] BS 4803, 'Guide on protection of personnel against hazards of laser radiation'

[64] 'IMS-B004 IBM PC add-in board', Inmos Ltd., 1000 Aztec West, Almondsbury, Bristol, UK, 1987, Doc.No. 72 TCH 011

[65] Inmos Ltd, 'Transputer Development System Manual', Prentice Hall International, 1988, Doc. No. 72 TRN 011 00

[66] Stephens M., Bourton M.A., "Evaluation of portable weld profile monitor", The Welding Institute, Abington, Cambridge, UK, September 1990

[67] 'Mullard Ferroxcube', Mullard Ltd., May 1955, Doc.No. MC.8155

[68] Watt O.H., 'Metals, ceramics and polymers', pp. 615-617, Cambridge University Press, 1974, ISBN 0-521-08238-2

[69] Components catalogue, pp 1080, Farnell Electronic Components Ltd., Canal Rd, Leeds, England, October 1990

[70] Farag M.M., 'Material and process selection in engineering', pp. 83-88 Applied Science Pub. Ltd., 1979, ISBN 0-85334-824-3

[71] 'TM-560 High resolution CCD camera', Pulnix America Inc., Unit 5, Intec 2, Wade Rd., Basingstoke, UK

[72] VT220 Owners manual, Digital Equipment Corporation, Doc.No.EK-VT220-UG-002

[73] Lilley F., Hobsen C.A., Koukash M, 'Visual inspection of surface mounted solder joints', Coherent and Electro-Optics Research Group, Liverpool Polytechnic, U.K.

Glossary

2MBTB - Two MegaByte Transputer Board

ADC - Analogue to Digital Converter

AGC - Automatic Gain Control

Algorithm A rule or method for solving a mathematical problem in a finite number of steps: a step by step method of solving a problem.

Ambient light Light which is present in the environment around a machine vision system and generated from outside sources.

Analogue signal The representation of signal by a smooth, continuous function curve of voltage vs time.

Back lighting The use of a light source placed behind an object so that a clear silhouette of an object is formed. This is used when surface features on an object are not important.

Bandpass Optical Filters Devices which are transparent to light of only a narrow range of frequencies.

Bandwidth The information carrying capacity of a circuit often expressed in frequency (MHz). As the bandwidth increases, the information capacity increases.

BBC - British Broadcasting Corporation

Binary image A black and white digitised image represented as zeros and ones.

Binary system A vision system which creates a digitised image of an object in which each pixel can have only two values such as black or white, one or zero.

BIOS - Basic Input Output System

Boundary The line formed by the joining of two image regions, each region having a different light intensity or pixel value.

C-mount A type of lens mount conventionally found on many machine vision cameras. Some cameras have U-mount or bayonet mounts and use an adapter for C-mount lenses.

CCD Charge Coupled Device
A type of semiconductor material.

CCIR The European monochrome broadcasting system standard which requires 625 lines at 50 Hz frame rate.

Chrominance The numerical definition of colour by its position in the chromacity diagram

Contrast The difference in light intensities between two regions in an image. This term is generally used to measure the difference between the lightest and darkest portions of an image.

Convolution Superimposing an MxN operator over an MxN pixel area (window) in the image, multiplying corresponding points together and summing the result. A normalization factor is frequently included in the process.

Correlation A correspondence between attributes in an image and a reference image.

DAC - Digital to Analogue Converter
An electronic hardware device which converts a digital signal into a continuous voltage or current proportional to the digital input.

Detector	The light sensing portion of an electro-optical system. The detector translates light energy into an electrical signal.
Digital image	A representation of a visual image by an array of brightness values.
DIL	- Dual-In-Line A particular pattern of electronic components designed to fit printed circuits (switches, sockets etc)
DOS	- Disk Operating System
DSP	- Digital Signal Processing.
Edge	A change in pixel values (exceeding some threshold) between two regions of relatively uniform values. Edges correspond to a discontinuity in surface orientation, surface reflectance or illumination.
Edge detection	The ability to determine the true edge of an object.
EPROM	- Erasable Programmable Read Only Memory The devices include a transparent quartz widow allowing the user to erase the programmed bit pattern by exposing it to ultra-violet light. It can then be re-programmed.
Eurocard	A standard size for circuit boards measuring 160 mm by 100 mm
FCB	- Frame Capture Board The TIPS circuit board which digitises an incoming video signal.
Features	Simple image data attributes such as pixel amplitudes, edge or point locations and textural descriptors or somewhat more elaborate image patterns such as boundaries and regions.
Fibre optic	A thin transparent fibre along which light can be passed but contained within its walls.
Frame	Digital data representing a picture.
Frame buffer	An area of memory used to store digital images.
Frame grabbing	Taking a set of data from a camera output representing a scanned frame and storing it in temporary memory for further analysis.
Front lighting	The use of illumination in front of an object by observing the pattern made by an object intersecting the structured light.
Grey level	A quantised measurement of image brightness or other pixel property usually given in integer values.
Grey scale image	An image consisting of an array of pixels where each pixel has a value representing the average light intensity on an area. Typically, 16, 64 or 256 levels are possible for each pixel, depending on the number of bits available to process and store data.
Handshaking	A method of passing data between two devices to ensure it is received correctly.
Histogram	Frequency counts of the occurrence of each intensity (grey level) in an image usually plotted as a number of pixels with a given grey value vs grey level.
Hz	- Hertz The standard unit of frequency (cycles per second).
IBM, IBM-PC, IBM-AT,	- International Business Machines Inc The company's machines have become an industry standard.
ID memory	An area of memory on the TIPS SDB which determines which pixels are displayed from which TIPS IPB.

Image	A projection of a scene into a plane. Usually represented as an array of brightness values.
Image enhancement	The use of processing techniques to accentuate certain properties to improve information received from an image.
Image processing	Transformation of an initial image into a second image with more desirable properties, such as increased sharpness, less noise and reduced geometric distortion.
Inmos CO11	An integrated circuit to interface an Inmos link with 8 bit data.
Intensity	The relative brightness of an image or portion of an image.
IO map	- Input Output map.
IPB	- Image Processing Board The TIPS circuit board which stores and processes digitised images.
Iteration time	The time taken for each repetition of a process.
Laplacian operator	The sum of the second derivatives of the image intensity in the x and y directions is called the Laplacian. The Laplacian operator is used to enhance the value of pixel values adjacent to edge elements.
Laser	- Light Amplification by Stimulated Emission of Radiation A device which emits mono-chromatic coherent light.
Latency	Delay
LED	- Light Emitting Diode A semi-conductor light source that emits visible light or infra-red radiation.
Linear array	A solid state video detector consisting of a single row of light sensitive semi-conductor devices.
Luminance	The description of brightness of an image
Machine vision	The ability of an automated system to perform certain tasks normally associated with human vision, including sensing, image formation, image analysis and image interpretation or decision making.
MFLOPS	- Million Floating Point Operations Per Second.
MIPS	- Million Instructions Per Second.
NTSC	A standard for colour images in America which augments the RS170 standard by utilizing a 3.58 Mhz colour sub-carrier on the video signal.
Occam	A high level computer programming language developed for use with the Inmos Transputer and facilitating parallel programming of the transputer.
PAL	- Phase Alternating Line Standard for colour images in the European broadcast system which is equivalent to NTSC in America.
PC-Bus	Connections for interface to an IBM compatible personal computer.
PIP frame store	A frame store produced by Matrox Ltd.
Pixel	The smallest element of a scene, a picture element, over which an average brightness value is determined and used to represent that portion of the scene. Pixels are arranged in a rectangular array to form a complete image of the scene.
Pixel value	Average brightness value over the pixel area, usually rounded off to an integer value.
Processing speed	The time required for a vision system to analyse and interpret an image.

QWERTY A standard layout typewriter keyboard arrangement.

Reflectance The ratio of total reflected to total incident illumination at each point.

Registration Processing images to correct geometrical and intensity distortions, relative translational shifts and magnification differences between one image and another or between an image and a reference map. When registered, there is a one to one correspondence between a set of points in the image and in the reference.

Resolution The smallest feature of an image which can be sensed by a vision system. Resolution is generally a function of the number of pixels in the image with a greater number of pixels giving better resolution.

RGB - Red, Green and Blue.
The three primary colours of light.

RS170 Standard TV monitor interface; 625 scans, interlaced, 1/40 second, active time 64 microseconds, retrace time 11.6 microseconds.

RS232 Standard computer interface protocol for transmitting serial data between computers.

SDB - System Display Board
The circuit board which controls the timings and operation of TIPS.

Segmentation The process of dividing a scene into a number of individually defined regions or segments.

SNR - Signal to Noise Ratio
The ratio between the usable signal and any extraneous noise signal present. this ratio is expressed in dB. If the SNR is exceeded, the transmitted signal quality will be unacceptable.

Structured lighting Sheets of light and other projective light patterns used to determine shape and/or dimensions of an object by observing the pattern made by an object intersecting the structured light.

Template An object outline to be matched with an observed image field; usually performed at pixel level.

Texture A local variation in pixel values that repeats in a regular or random way across a portion of an image or object.

Thresholding Separating elements or regions of an image for processing based on pixel values above or below a chosen (threshold) value or grey level.

TIG - Tungsten Inert Gas welding
Method of electrical welding using a non consumable conductor shielded by inert gas.

TIPS - Transputer Image Processing System

TSCA - Transputer Serial Communications Adapter

VLSI - Very Large Scale Integration

VME - Versa Memory Europa

Windowing A technique for reducing data processing requirements by electronically defining only a small portion of the image to be analysed. All other parts of the image are ignored.