CONTROL OF A PCB DRILLING MACHINE BY VISUAL FEEDBACK

J.A.G. Hale and P. Saraga
Mullard Research Laboratories,
Redhill, Surrey, RH1 5HA, England

Abstract

A visual input is likely to play an important role in many future mechanical handling and inspection systems. This paper will describe one experimental implementation of a machine of this type; an automatic printed circuit board drilling machine in which a TV camera is used to view the board and a small computer is used to process the TV image. The computer controls the machine using only the information extracted from the picture.

Introduction

It is becoming increasingly difficult for industry to find people who are willing to do dangerous or monotonous jobs. Those who are performing boring or repetitive tasks are liable to become frustrated and error prone. Problems of this type together with the increasing cost of manpower provide strong incentives for industry to automate their produetion lines.

With the advent of computers, traditional fixed automation is being selectively replaced by programmable numerically controlled machines. These machines nevertheless follow a fixed set of instructions, and in general they cannot cope with a change in their environment. As computers have become cheaper and more powerful it has become possible to introduce into factories more intelligent machines (1) which are aware of their environment and which are capable of reacting to changes in that environment. Examples of this type of machine include Hitachi's visually controlled bolt tightener, and General Motor's system for placing wheels on hubs (2,3). In this paper we are concerned with machines which examine their environment with visual sensors and which we call visually controlled machines. Much of the discussion however would apply to machines equipped with other types of sensor.

Justification for Visually Controlled Automation

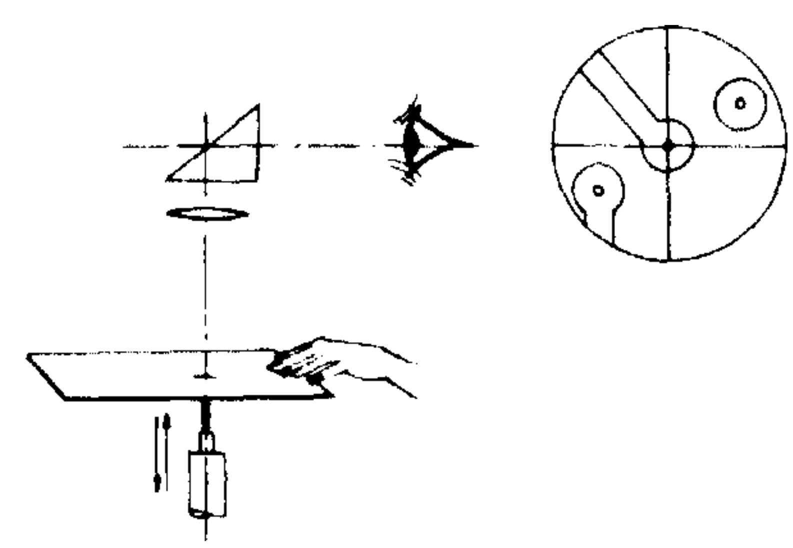
It can be argued that there should be no need for visually controlled machines in the ideal automatic factory. Consider for example, the problem of automatic assembly. If the orientation and position of all components were preserved from the point of initial fabrication, where they are well known, to the final assembly into the completed product, then 'blind automation' would be quite satisfactory. In practice however there are many reasons why this desirable goal cannot be achieved. It may often be necessary to Met go' of parts, (for example when small sheet metal pressings are plated or deburred), and to store them between manufacture and assembly. If the components are stored in a 'loose' state, then either a person or a machine is required to feed the component to the automatic assembly machine. In many cases, this problem is solved by ingenious mechanical designs such as bowl feeders. Some components however provide harder problems for the mechanical designer and it is in these circumstances that a visually controlled machine is useful.

Thus visually controlled machines may be cost effertive when it is uneconomic to keep components in jigs, when frequent product changes make complete programming expensive, or when it is important to detect drifts in the product or machine (2-5).

The Application of Visual Control to Drilling Printed Circuit Boards

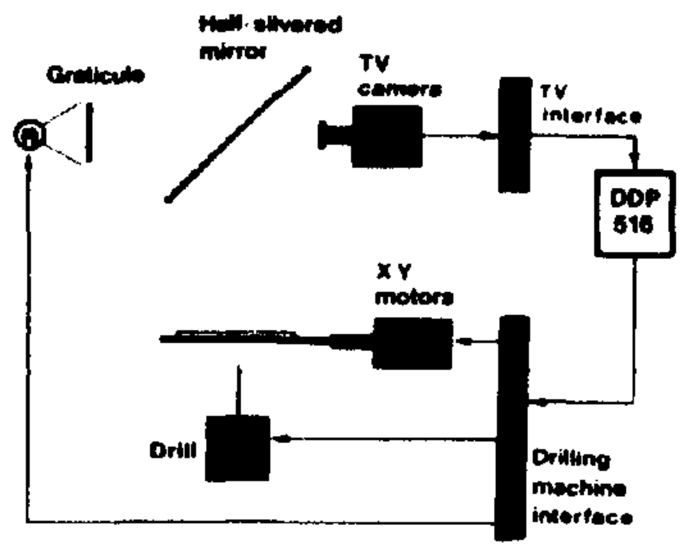
The drilling of Printed Circuit Boards (PCBs) under visual control is a problem in which both the mechanics and the scene analysis are relatively simple. The solution to the problem could however have genuine practical application.

In research laboratories and other establishments in which PCBs are made in very short runs of small numbers of boards, the boards are often drilled by hand using a single spindle drilling machine such as that illustrated diagrammatically in Fig.la.

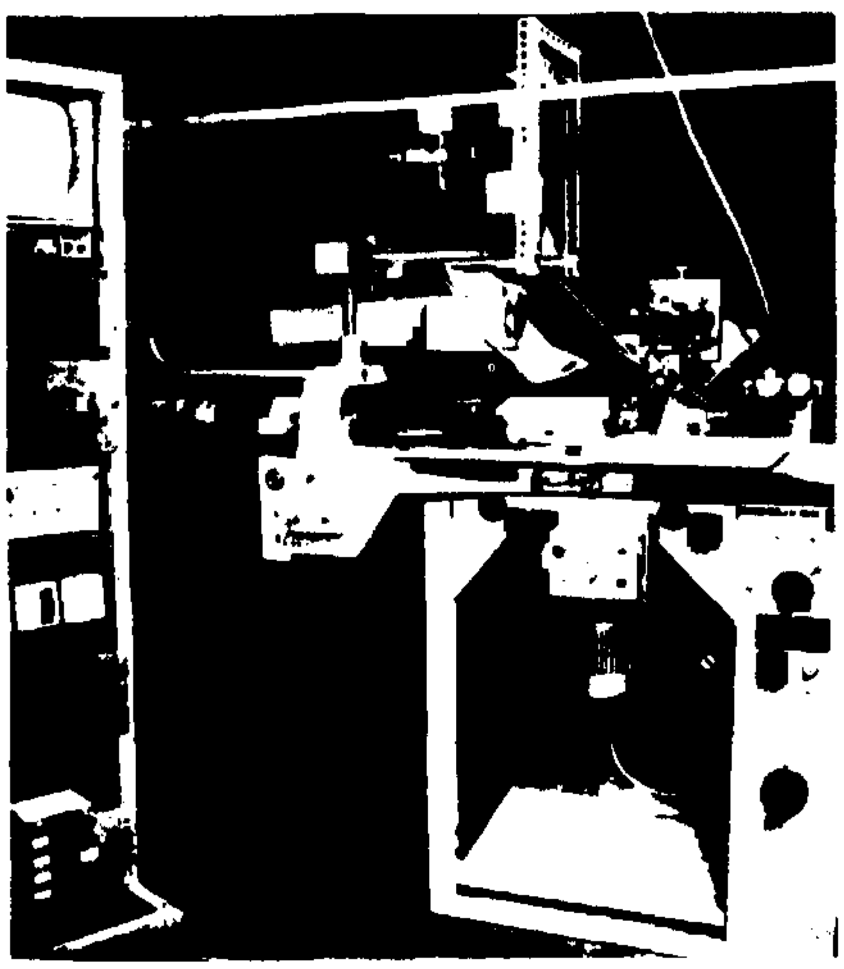


1. The Manual Drilling of Printed Circuit Boards

The operator views the board through the eyepiece, which contains a crosswire indicating the position of the drill (Fig.lb). To drill a hole, he moves the board until the point to be drilled coincides with the crosswire, and then actuates the drill. The experimental equipment built at *MRL* to automate this process and replace the human operator by a computer vision system, is illustrated diagrammatically in Fig.2 and by the photograph in Fig.3. The board is viewed, via a half-silvered mirror, by a TV camera which is interfaced to a Honeywell 516 computer. The TV video signal can be sampled with a maximum resolution of 300 x 400 picture elements (pixels) over the field of view of the camera, and each pixel can be digitised to 5 bits.



2. *Schematic of the Visually Controlled Machine*

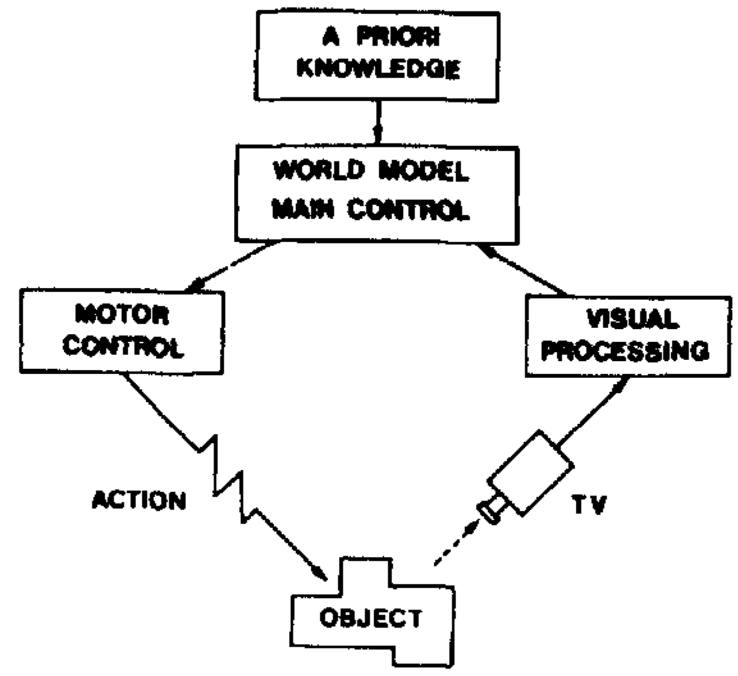


3. **Photograph of the Visually Controlled Machine**

The board can be moved over the table by means of stepping motors which are also computer controlled. The function of the crosswire in the manual machine is accomplished by projecting a graticule (a Maltese cross) into the field of the TV camera. This also allows for automatic compensation of any drift in the TV system.

The Structure of a Practical Visually Controlled Machine

The type of visually controlled machine in which we are interested (fig.4) can be characterised by four main attributes. Firstly the machine is assumed to be working on a defined task within a limited real world environment. A conceptual description of the task and the environment is given to the machine as a-priori information.

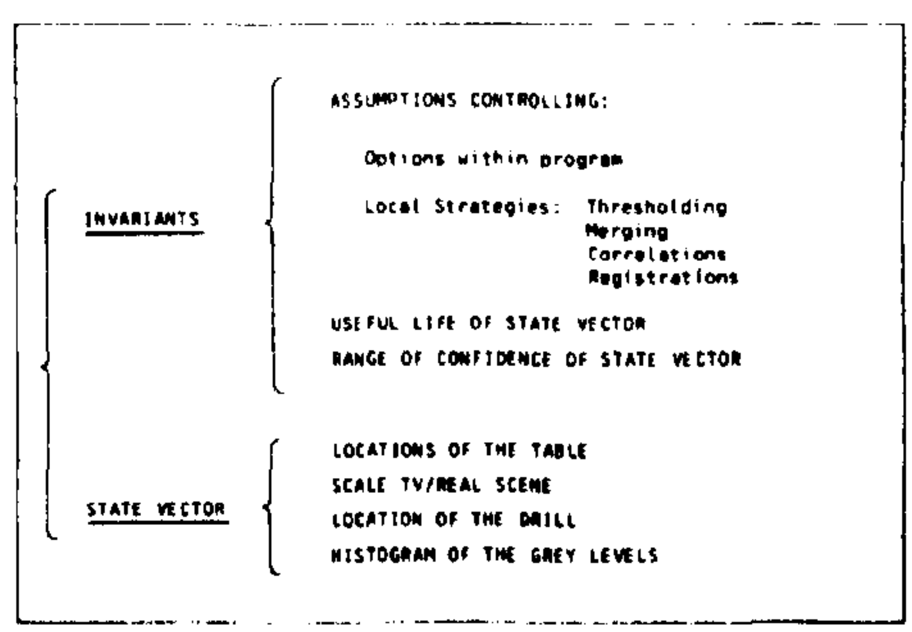


4. **Conceptual Model of a Visually Controlled Machine**

Secondly the machine is equipped with sensors which enable it to gather from the environment the informat ion it requires to perform the task. Thirdly by using both the a-priori and the sensed iniormation, the machine can construet its own internal concept ion of the real wurld i.e. its 'world model' (8,9). The model is assumed to be dynamic and may be updated by new data from the sensors. Fourthly the main control unit can use the world model to interpret sensed information and as a result instruct the motor control to take actions in pursuance of the defined task.

The World Model

We distinguish two eategor ies of iniormat ion within the world model (Fig.5), a set of 'Invariants' and a "state vector' (It)). The Invariants inchide the assumptions about the rea1 world specified by the program designer, e.g. that the TV raster is parallel to the X displacement of the drilling machine table. It might also state that a linear relationship exists between movements in the real world (table movements A_{TB}) and corresponding displacements in the television picture (image *movements* A_{TV}). The actual *value ol* the scale given by A_{TB}/A_{TV} is one of the elements of the state vector.



5. **Contents of World Model**

In general the state vector includes those model parameters which might be changed when new information is received from the sensors. In the case of the drilling machine, state vector elements include the current location of the table, the drill, and the projected graticule in the field of view. Also contained within the state vector is a histogram of the grey levels in the picture being processed. If at any time, the machine fails to calculate a value for any part of the state vector, it may either call for help or utilise the 'default value' which is included within the a-priori information given to the machine by the operator when he sets it up.

The Use of Visual Feedback

As well as using sensed information to determine a course of action, the sensors can also be used to monitor the progress of the action (11-13). If the action appears to be incorrect, this may be because the motor control is faulty, the mechanism is inaccurate, the machine's vision is distorted, the world model is incorrect, or the a–priori information was false.

The machine can probably compensate for errors in the motor control or the mechanism by observing the results of the action and giving new instructions. This suggests that machines equipped with visual feedback may require less accurate mechanics. If the corrective actions do not succeed it is probably because the world model is no longer valid (perhaps the magnification of a zoom lens hs changed). If the main control decides that it is the world model which is at fault then it can enter a model updating phase similar to the model initialisation phase (Section 3.1). False a-priori information or distorted vision are likely to be hard to cope with, and in practice would probably result in calls for external help.

Hand-Eye Machine in University and Industry

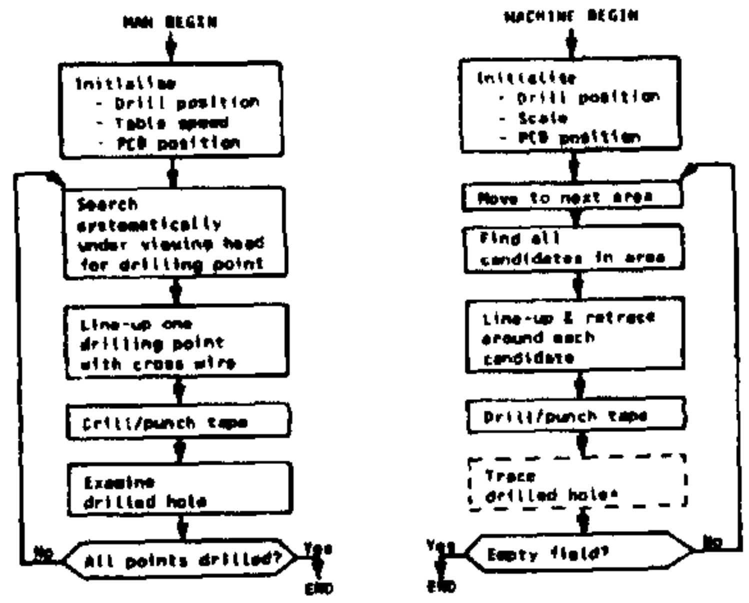
Hand-eye machines of the type described above developed in various Artificial Intelligence Laboratories. In many cases the aim of these projects has been the development of intelligent machines for their own sake, although the ideas have sometimes also been applied to practical problems in order to demonstrate the versatility of the system (14,15).

In industry, the situation is reversed. We are trying to solve real problems at the minimum cost and we only use machine intelligence where it offers a real advantage. Instead of trying to make machines solve complex problems, we try to make the problems and the equipment required as simple as possible. These differences in approach will be illustrated by the description of the visually controlled drilling machine.

Strategy and Program

As mentioned in the introduction, the visually controlled drilling machine is intended to replace a manually operated machine.

The human operator's task of drilling the board is approximately described by the flow charts of Fig.6a. He first undergoes an



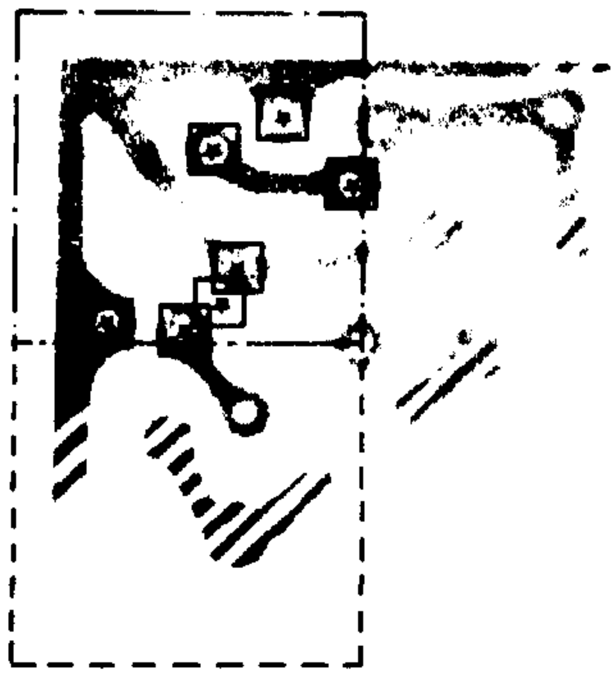
(a) (b)

6. **Flow Charts of PCB Drilling:**

(a) by a human operator
(b) by the visually controlled machine

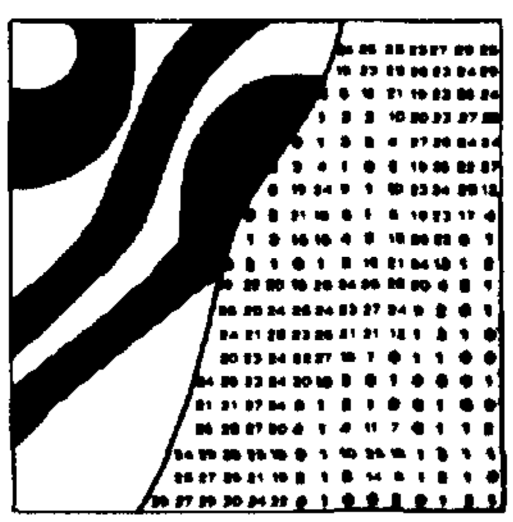
'initialisation' phase during which he establishes his 'world model'. He checks such things as the accuracy of the correspondence between tho position of the drill and the crosswire, and the relationship between moving the board on the table and the resulting movement of the image of the board in his viewer. Once the initialisation is complete he systematically moves the board under the viewer, scanning the image for drilling points. As soon as he spots something that might be a drilling point, he stops moving the board and examines the candidate point more carefully. If he confirms it is a drilling point, he moves it to coincide with the crosswire, and drills a hole. At this point he can check whether the drilled hole is exactly in the centre of the drilling point. If he notes a consistent error in the position of the hole, he may decide to adjust his 'world model' accordingly. The above sequence of operations is repeated until all holes have been drilled.

The computer program in the H516 follows a sequence analogous to the human operator's, as illustrated in Fig.6b. It first enters the initialisation phase during which the scale between table and image movement, the position of the projected graticule, and the relationship between the graticule and a drilled hole arc determined. The main drilling phase is then entered during which the board is systematically searched for drilling points by scanning a sequence of adjacent sub-areas (each 57 x 57 pixels) of the board (Fig.7). The TV interface samples the TV video and converts it into a digital signal of 5 bits per pixel (32 grey levels in which 0 is white and 31 is black). By this means, an image such as that illustrated in Fig.8 is stored in the computer.

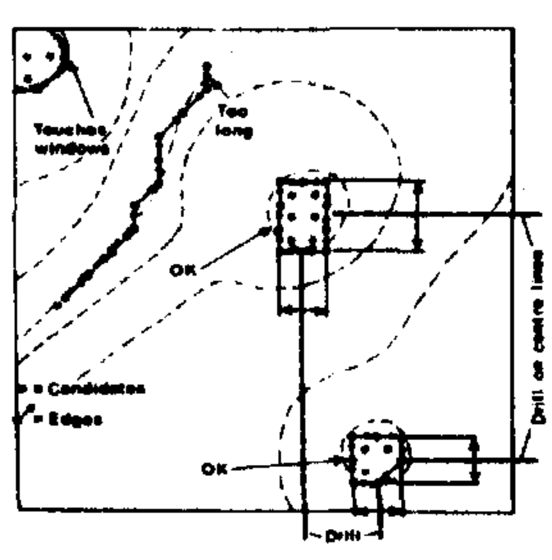


7. **Strategy for Drilling-Point Location**

The stored image of each sub-area is examined for patterns which correspond to 'candidate' drilling points and their positions are recorded. A candidate is detected by a 'blob operator' such as that described in Section 4.1. Once the positions of all the candidates within a sub-area have been determined, the table is moved to bring each candidate in turn, to the centre of the field of view of the TV camera.



8. **Digital Image of PCB Extracted from the TV Picture**



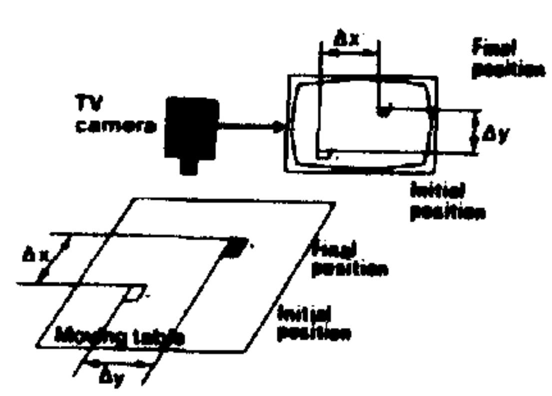
9. **Edge Tracing of the Picture of Fig.8 around Candidates**

The region encompassing the candidate is then rescanned to obtain a higher resolution image. This is thresholded to a binary image which is examined by tracing the black white boundary (Fig.9) (16). If the shape of the boundary is consistent with that of a drilling point (i.e. it approximates to a circle), the table is moved to bring the centre of the drilling point (the centre of the circle) into coincidence with the assumed drill location, and a hole is drilled. This process continues until all sub-areas have been scanned and all drilling points have been drilled.

Initialisation

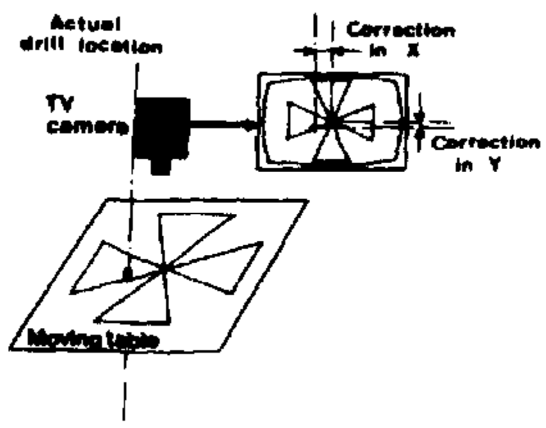
The object of the initialisation procedure is to assign an initial numerical value to the elements of the state vector (XSCALE, YSCALE, XDRILL, YDRILL, XGRAT, YGRAT, XTAB, YTAB). The determination of these values can be either entirely automatic or in an interactive mode in which the user monitors each operation. If any automatic operation appears to the machine to have failed, the program assigns the default value to that element of the state vector.

The initialisation procedure assumes that the board contains an 'empty region' without any pattern except a Maltese cross (which may either have been etched into the copper or stuck on to the board by the user). The user positions the board so that this cross is in the field of view of the TV camera and the program locates the centre of the cross using a two pass process. In the first pass the TV picture is coarsely sampled (every 16th point from the rectangular grid) and a simple operator, described in Section 4.2, is used to find 'candidate' locations for the centre of the cross. In the second pass the area around each candidate is rescanned at a higher resolution, and examined with the more elaborate operator described in Section 4.3. The centre of the Maltese cross should be detected at one of the candidate locations. The board is then shifted a known distance in X and Y and the new position of the cross in the TV picture is determined. XSCALE and YSCALE can now be calculated (see Fig.10).



10. **Scale Determination**

The purpose of the projected graticule is to act as a secondary datum defining the position of the drill, the primary datum being the actual position of a drilled hole (Fig.11). The graticule projector is mounted, independently from the TV camera, on the main frame of the drilling machine. Therefore, the relative positions, in the TV picture, of the graticule and the drill are unaffected by an mechanical movements or electronic drifts in the camera. Although by prior adjustment the centre of the Maltese cross is arranged to correspond to the drill position, mechanical drifts in the projector can introduce a displacement in their relative position. This offset is calculated by drilling a test hole in the board as described below.



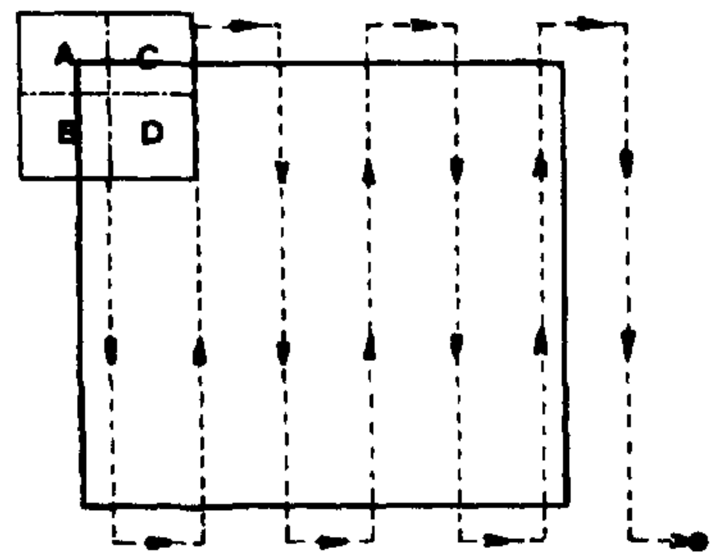
11. Determination of the Offset between the Projected Graticule and the Drill

In the next phase of the initialisation procedure the Maltese cross graticule is switched on and appears in the TV picture. Its position is determined by the same method as that used for the cross on the board. The projected graticule is then switched off, and the program relocates the cross on the board. It then searches for a small (20 x 20 pixel) empty area which it positions at XGRAT, YGRAT. The test hole, which it is assumed will fall within the scanned area, is drilled and its position determined using the method employed to locate drilling points (Section 3). Since the position of the graticule has already been found, the offset between drill and graticule can now be calculated.

Movement Strategy and the Registration of Sub-Areas

An important feature of the control program is its ability to explore the whole board without missing or reprocessing any regions. This is accomplished by starting in one corner of the board and by systematically processing successive sub-areas in a boustrophedal route as shown in Fig.12.

For each sub-area being processed for drilling an area four times the size of the sub-area is scanned, the other three sub-areas being concerned with the registration between one drilled sub-area and the next. One of these three will be the next to be drilled and this sub-area is also inspected for candidate drilling



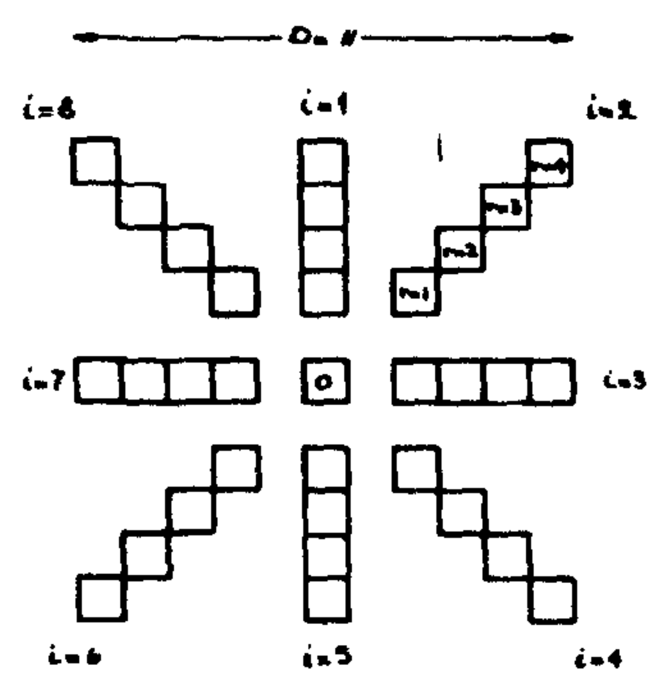
12. **Movement Strategy**

points whose locations are stored. When the board is shifted to process this next sub-area for drilling a new list of candidates is obtained whose positions could be checked against the list obtained in the previous stop. Any misregistration which may have occurred could be computed and corrected.

Picture Processing Operators

A Simple Blob Detection Operator

During the execution of the program as described above, the computer has to recognise 'blobs' defined as dark objects surrounded by a closed ring of picture elements which are lighter than the object itself. The shape of this ring is not defined, but its dimension must be smaller than D, where D is the size of the operator used to recognise the blob. (D is an odd number). The operator (Fig.13) is a square template composed of eight radiating limbs.



13. **Simple Blob Detection Operator**

Let $P(i,r)$ be the density of one picture element seen through the r^{th} element of limb i of the template.

$P(0)$ be the density seen through the centre of the template.

$$P_{min(i)} = \text{Minimum} \left\{ P(i,r) \mid r = 1, (D-1)/2 \right\}$$

T be a given constant offset.

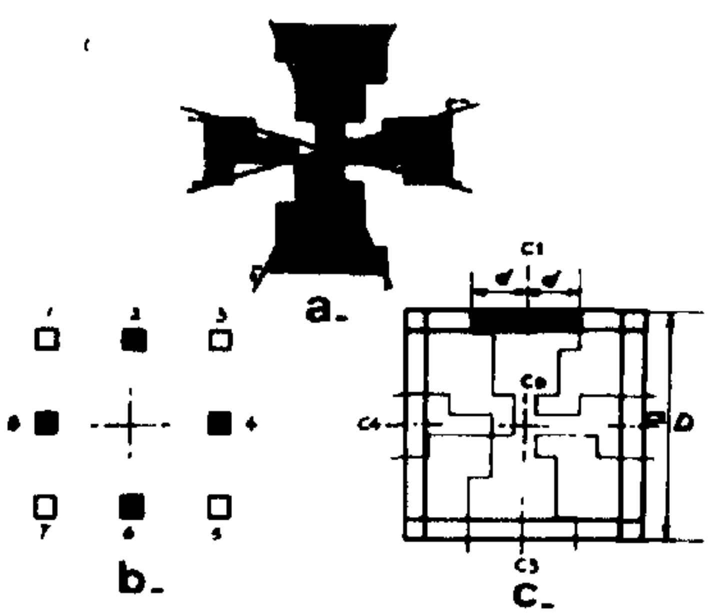
Then if the condition

$$P(0) > P_{min(i)} + T \qquad \text{for } i = 1,8$$

is satisfied, then a blob is present in the region bounded by the outer elements of the operator and one of its elements is at the centre of the operator.

<u>A Simple Local Operator for Locating a Maltese Cross</u>

This operator is designed to detect the sequence of black and white areas which surround the centre of the cross (Fig.14a). The shape of the operator, which can be regarded as a variation of the blob detector described above, is shown in Fig.14b.



14. Recognition of the Maltese Cross:

(a) binary picture
(b) simple operator
(c) more elaborate operator for locating the Maltese cross

Let $P(i)$ be the density of the picture element seen through element i of the operator.

T be a given constant offset.

$$L = P(1) + T$$

Then if the conditions

$$P(i) > L \qquad \text{for } i = 2,4,6,8$$

$$P(i) \not> L \qquad \text{for } i = 3,5,7$$

are satisfied then a candidate location for the cross is detected at the centre of the operator.

<u>A More Elaborate Local Operator for Locating a Maltese Cross</u>

This operator is designed to inspect the area surrounding the candidate location of the cross determined by the simple operator. It checks whether a cross is really present and if so determines its exact position. The operator is based on recognising a series of increasingly long vertical and horizontal dark bars on a light background (Fig.14c). First, a square search path of size D is defined, and a histogram of the density of the D pixels along each side of the square is computed. A binary pattern is obtained by thresholding at the density level defined by the first minimum in the histogram. The binary pattern is 'accepted' only if it contains a black bar whose length is of the order of $^D/2$ (within say 50% of this value). If the pattern is rejected a second threshold value (given by the next minimum in the histogram) is tried, and so on. If the pattern is accepted its centre C_1 is found and recorded. The procedure is repeated to find C_2, C_3, C_4 on the other sides of the square. A tentative centre C_D for the cross is found by the intersection of C_1 C_3 with C_2 C_4. The centre of the cluster of successive tentative centres C_D obtained for increasing values of D is taken as the centre of the cross.

<u>Performance and Conclusions</u>

Although fairly complex at first sight, the program requires only the core size of a small mini computer (16K words of 16 bits). The addition of more sophistication would require at least one level of overlay. The program is written in Fortran IV with single precision integer operations and machine code device drivers. Although the speed of the program has not been optimised, the machine can drill holes at approximately 1 every 6 seconds. This is slower than can be achieved by a human operator or when the machine is controlled from a punched tape. In this latter mode the speed of the machine is approximately 1 hole every 2.5 seconds. The comparison with numerically controlled operation leads one towards an alternative use for the visually controlled machine. Instead of drilling the board, it could be used to prepare a punched tape for future numerical control. One can also consider extensions of this type of machine to other problems of automatic digitisation.

In its present state, the machine can almost drill complete boards. The program still needs to be improved slightly to prevent a few drilling points being missed. This is particularly important as the practical use of the machine in the laboratory workshop is now being considered.

As described, the machine treats each hole as an individual item. In practice however, the relative position of holes in a PCB is often as important as the location of an individual hole at the centre of a drilling point. For example, the pitch of the pattern of holes for a dual-in-line IC must be regular to ensure easy insertion of the device in the board. The drilling machine could be made intelligent enough to recognise such patterns of drilling points and to drill the holes accordingly.

Other features which could be added to the program include the recognition of other shapes as drilling points and an ability to check that the holes are actually drilled in the right place with respect to the copper pattern. If the machine discovered that the holes were misplaced, procedures for updating the contents of the state vector could be invoked.

Acknowledgements

We would like to acknowledge the contributions of the other members of the team who have worked on this project, A.R. Turner-Smith and D. Paterson, We would also like to thank our Group Leader J.A. Weaver for his help and encouragement.

References

(1) A.N. RADCHENKO, E.I. YEREVICH. On a Definition of Artificial Intelligence. 1st CISM-IFTOMM Symposium Udine (Italy), Sept. 5-8, 1973.

(2) Japanese Robots now have Eyes. Press Release in The Engineer, (4th Oct. 1973), p.21.

(3) J.T. OLSZTYN, L. ROSSOL, R. DEWAR, N.R. LEWIS. An Application of Computer Vision to a Simulated Assembly Task. 1st Int. Joint Conf. on Pattern Recognition, Washington DC, Nov. 1973, p.505,513.

(4) W.B. HEGINBOTHAM. Reasons for Robots. 1st Conf. on Ind. Robot Techn., Nottingham UK (1973), p.R1/3-12.

(5) W.B. HEGINBOTHAM. Robots - The Trend to a Visual Knowledge of their Environment. Design Engineering (Sept. 1974), p.280-283.

(6) A ROSENBLATT Robot Handling More Jobs on Industrial Assembly Lines. Electronics, (July 19 1973), p.93-104.

(7) K. MORISHITA, G. BOOTHROYD. Group Technology: What Role for Robots. Automation (August 1973), p.34-39.

(8) P.F. ROWAT, R.S. ROSENBERG. Robot Simulation Sundries: Descriptions and Plans. Canadian Computer Conference, Montreal, June 1-3 1972, p.423301-423332.

(9) R.E. FIKES, P.E. HART, N.J. NILSSON. Learning and Executing Generalized Robot Plans. Artificial Intelligence 2 (1972), p.251-288.

(10) M.H. LEE, D.J. WOOD. Sensory-Motor Control Problems in a Flexible Industrial Robot System. 2nd Conf. on Ind. Robot Techn., Birmingham, March 27-29, 1974, p.C2/9-C2/20.

(11) J.R. PARKS. Sensory Devices for Industrial Manipulators and Tools. 1st CISM IFTOMM Symposium, Udine (Italy), Sept. 5-8, 1973.

(12) SHIRAI, H. INOUE. Guiding a Robot by Visual Feedback in Assembling Tasks. Pattern Recognition 5(1973), p.99-108.

(13) J. KLIR, M. VALACH. Cybernetic Modelling. Iliffe Books Ltd (London) 1967, p.257-278.

(14) A.P. AMBLER, H.G. BARROW, CM. BROWN, R.M. BURSTALL, R.J. POPPLESTONE. A Versatile Computer-Controlled Assembly System. 3rd Int. Joint Conf. Artif. Intell. Stanford, August 1973, p.298-307.

(15) A. GILL. Visual Feedback and Related Problems in Computer Controlled Hand-Eye Coordination. Stanford Artificial Intelligence Memo AIM 178, October 1972.

(16) P. SARAGA, P. WAVISH. Edge Tracing in Binary Arrays in Machine Perception of Patterns and Pictures. Institute of Physics (London) Conf. Ser. No.13, (1972), p.294-302.