An eye to the future with computer vision

Judging from current trends, it seems that computer vision will play an increasingly major role in future robot-control systems. Mark Witkowski reports on the latest developments.

SIGHT IS the primary sense used by people for a multitude of tasks; the same is also becoming true of robots. Television cameras have been interfaced to computers for many years now. These and images from other sources such as digitised medical X-rays, biological slide preparations, satellite data, bubble-chamber photographs and others have shown that our own visual processes are far from trivial.

Vision offers many advantages over other types of robot senses. A picture can be acquired relatively quickly and unlike force or touch sensors, it leaves the work-piece undisturbed.

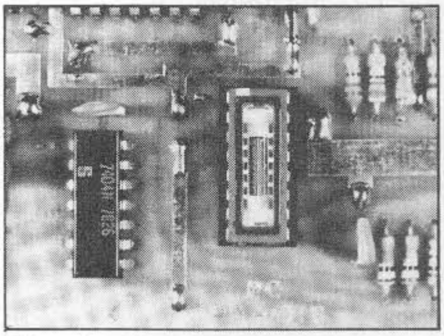
It offers many types of information about the scene viewed, recognition of objects and about their position and orientation. Vision can be used by a robot to search its environment for an object which is needed, to provide feedback to guide a manipulator or vehicle in some complex task.

Checks can be made to verify that a task is proceeding satisfactorily and that it is completed correctly. Furthermore, the data used is understood readily by the human user.

The essential problem in robot vision is to take a digitised image, which may be nearly 250,000 bytes of data and reduce it to something useful, such as: "it's a con-rod, located at X and Y, with an orientation of theta degrees".

Clearly such a massive data transformation is going to proceed in several stages, the algorithm for each may be very complex. Usually picture data is a matrix of numbers, each of which represents the brightness of the image at a particular point in the scene.

Robot images are seldom less than 64 by 64 in size. Below that, the image is degraded by the digitisation process, and is, usually limited to around 256 by 256



Picture 1.

due to the computation time required to process so much data.

Furthermore, each image point, or Pixel, will be digitised to a certain number of grey levels. That might be a simple binary image or one containing 256, 8 bits, different brightness levels. Imaging devices are usually solid-state photo-sensitive arrays of matrices, or standard Vidicon tube cameras.

Picture 1 shows a 64 × 1 photodiode array which are scanned using external clocking logic. The output from each light-sensitive diode is digitised and stored in computer memory in turn. Photodiode arrays up to 1,024 diodes in length are available.

They are particularly useful where the object to be viewed is essentially flat and moving, such as items on a conveyor belt. The CONSIGHT-1 system at General Motors is an example, figure 1 — Holland, Rossol and Ward, 1979.

Solid-state imaging matrices are manufactured both using photodiodes — the IPL 2D1 device is 64 by 64 diodes, manufacturer's literature, Integrated Photomatrix Ltd — and light-sensitive, charged coupled devices CCD.

The CCD201 is a 100 × 100 device — manufacturer's literature, Fairchild — which has been used in a number of robot projects. New solid-state units are being introduced and should soon be available in broadcast resolutions.

Solid-state cameras are smaller, lighter, consume less power and are more robust than their Vidicon counterparts. Some care is needed if they are to generate pictures with many grey levels — they are prone to uneven sensitivity across the picture, though that can be corrected by compensating circuits — Green, 1980.

Drive circuits must be calibrated carefully if their full dynamic range is to be usable. As the photo-sensitive array is manufactured precisely, there is little distortion of the image. In high-resolution applications the linearity of scan in Vidicon-based cameras may be a problem.

Vidicon cameras are used where high resolutions are needed, from 256 × 256 to 1024 × 1024 Pixels, often to eight bits of grey level. Each digitisation of a 512-Pixel line from a standard 625-line television camera must be completed, including storing the resulting data in about 100 nanoseconds.

As that is considerably shorter than the cycle time of most mini-and micro-computers, the normal technique used is to store a single line of the picture, which is user-selected, in a fast buffer and later read into computer memory through a parallel interface or DMA device at a more leisurely rate.

Picture-frame capture-times vary considerably according to the hardware design and computer system used, but in every case the picture must remain stable

Figure 1a.

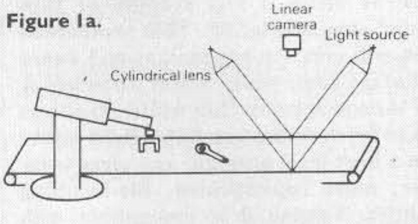
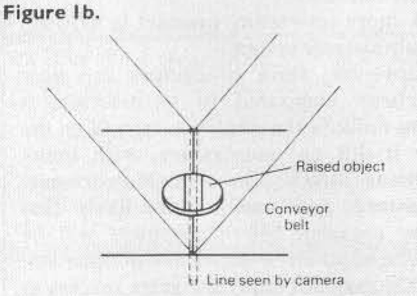


Figure 1b.



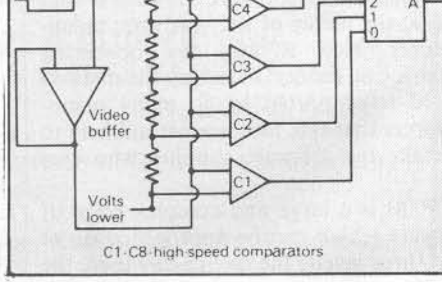


Figure 2.

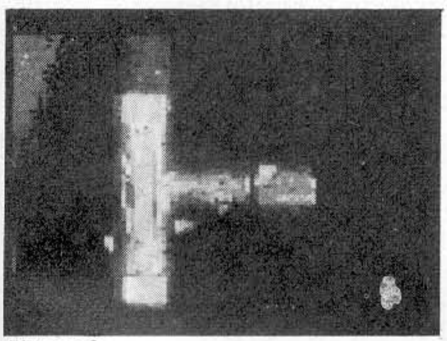
during the whole of the acquisition phase. Picture-input times vary from four to 205 seconds for a 1024 × 1024 image — manufacturer's literature, Hamamatsu Ltd.

If rapid data capture is of prime importance, it is possible to digitise and store a complete picture in one frame-time, about 40mS, for later processing by a host computer — manufacturer's literature, Microconsultants Ltd and Taylor, 1977.

Picture data signals represent a particular problem for analogue-to-digital conversion systems. The techniques mentioned in part three, ramp and successive approximation, are too slow. Figure 2 shows the principle of the fully-parallel flash 'a' to 'd' converter, capable of converting an analogue voltage into a binary number in less than 40 nanoseconds.

Each of the eight voltage comparators, Cl to C8, are fed with a different voltage from the resistor chain. The video waveform goes to the other input of all the comparators. None, some or all of the comparators will be active, according to whether the voltage at the resistor-divider chain input is greater or less than that of the video signal. The larger the video signal the more comparators will be switched on.

The outputs of all the comparators are



Picture 2.

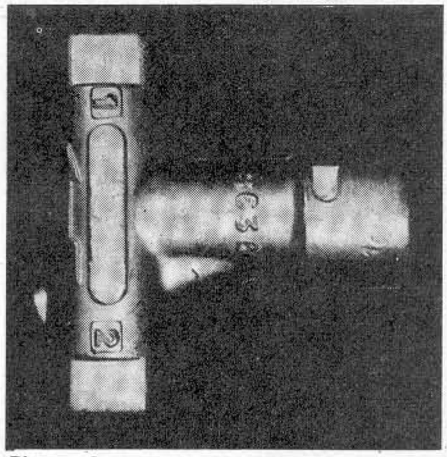
fed to an eight-to-three-line (binary) priority encoder. The three-bit result shows the highest comparator to be switched on. Flash 'a' to 'd' converters are available in both module form, up to eight bit, 50nS conversion — manufacturer's literature, Date Ltd — in which the conversion takes place in two four-bit stages and as integrated circuits.

The chip contains the resistor chain, 255 comparators, a 255 to eight-bit priority encoder and latch — manufacturer's literature, TRW LSI products. A number commercially-available television-to-computer input systems are reviewed by Onda and Ohashi (1979).

Figure 3 shows a simple form of visual inspection which could be used to check castings and the like during manufacture. Only if all the uninverted photocells, C, D and E, are lit, and the inverted ones, A and B, dark, will the output indicate that the part is usable.

The casting has to be held in a precise position for the check to work — it is a simple form of template matching. Kirsch, 1979, describes how a laser beam is passed back and forth round an object, using mirrors to test it during manufacture.

Uno *et al.*, 1979, use template-matching techniques on the output of a CCD image to attach water hoses to test water pumps during manufacture. Hale and Sarago,



Picture 3.

1975, used a template-matching process to align printed-circuit lead pads ready for automatic drilling.

Template matching works only if the object is very nearly aligned with the sensor. Different techniques are required to locate and identify work-pieces.

Picture 2 shows a brass blank during manufacture, digitised using a 100-by-100 CCD solid-state camera. The output is digitised to four bits with a flash 'a' to 'd' converter. In the system, used pairs of four-bit Pixel values are stored in a single eight-bit byte of an 8K memory block on a 6800 microprocessor.

The memory block is isolated from the processor bus logically for the frame duration — about 20 mS — while the picture is loaded into a 5K portion of the memory. When frame capture is complete, the picture appears in the microcomputer memory space with no interruption of processing. Picture 3 shows the original item.

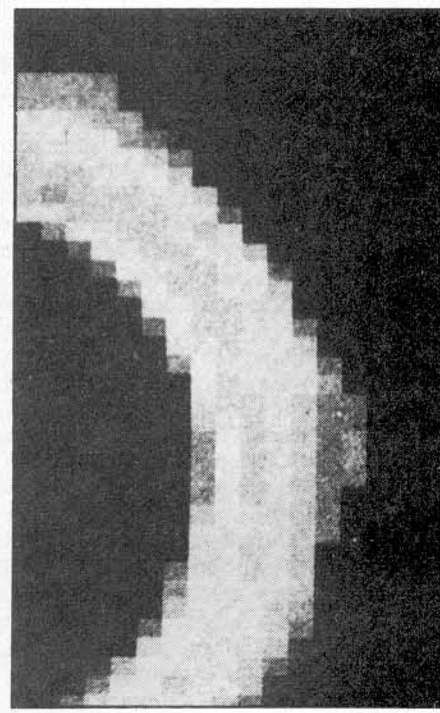
Only a few industrial robot systems use any grey-level information. The picture is pre-processed so that a binary image is formed. If the light intensity of any Pixel is less than a certain threshold, it is taken to be dark, otherwise it is taken as light.

A silhouette image is formed in which useful information about the object and any hole in it is contained in the size and shape of its overall outline. Obviously, great care must be taken with the lighting of the object. Shadows on the background or badly-lit recesses in the object could change the apparent shape of the image completely.

Picture 4 illustrates the potential problem with silhouette algorithms. The image is a section of a hexagonal nut, the background clearly dark and the nut light. However, the Pixels on the boundary are various shades of grey, caused, for the most part, by light from both the light and dark regions falling on the same photosensitive element and being averaged together.

If the binary threshold is, therefore, set high, many of these grey points will become black and the area and perimeter of the object will appear to shrink. Similarly, if it is set low, they will be taken as white and the object expand.

Great care is, therefore, needed for the lighting and camera aperture settings which should remain constant during and between sessions. Camera calibration may also involve showing the system grey step cards to determine the threshold point. There are also software calibration



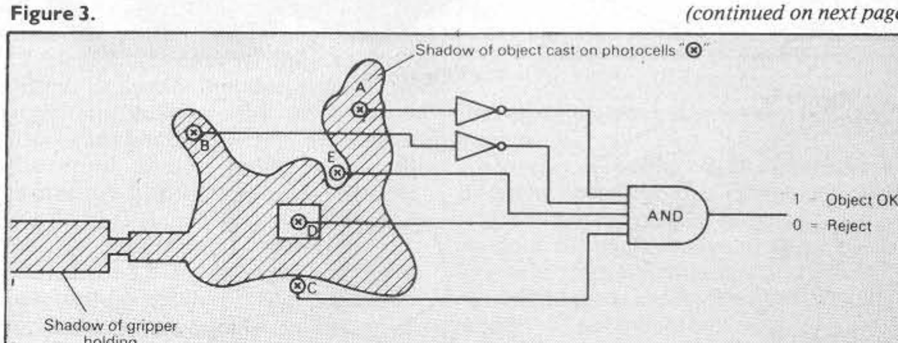
Picture 4.

techniques using histograms of the grey levels to determine the settings needed.

Several industrial robot systems, or at least prototypes, process binary images. Saraga and Skoyles, 1976, have a system to sort small objects delivered in a random orientation from a bowl feeder, to be picked-up by a three-degree-of-freedom

(continued on next page)

Figure 3.



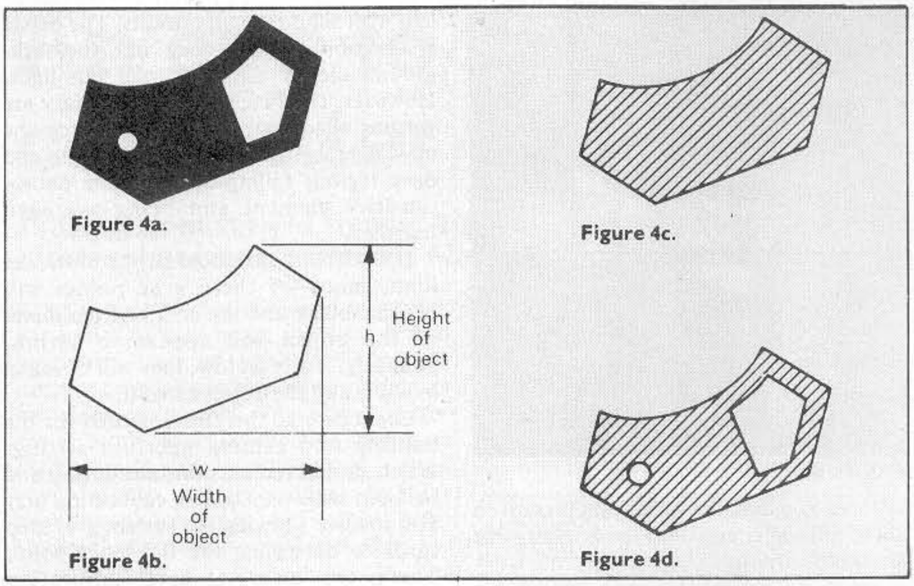


Figure 4a. Figure 4b. Figure 4c. Figure 4d.

(continued from previous page)

arm and deposited correctly in a sorted hopper.

As the arm is designed only to pick objects at one particular point, the table can be rotated and translated in X and Y to position objects correctly. There are two cameras, one directly overhead and one to give a horizontal side-view of the object.

To give a clear binary image, the object can be lit from underneath, eliminating shadowing problems and spurious effects caused by highly-reflective objects.

SRI has developed a system for partially assembling car air-conditioning compressors using visual feedback to control a Unimate 2000B manipulator, McGhie and Hill, 1978. The task chosen was to view a compressor body from above against a black background, illuminated so that the flat top surface appeared bright, but that holes in it formed shadows and appeared dark.

The position and orientation of the compressor body was computed from two large, circular blobs — the holes forming the piston bores. A matching compressor cover was then located on the body and vision was again used to find the bolt holes.

A third system to use binary images was Freddy, part of whose algorithm was described in part five. Freddy had to isolate a part from a heap, recognise it and place it in a standard position so that it could later be used by an assembly program.

The output of the linear scanning device on CONSIGHT-1 was also limited to a binary signal. The designers of the system were very concious of the problem of visual noise introduced by the grease, dirt and other debris to be found in a typical industrial environment.

Figure 1b shows the lighting arrangement they adopted for use with objects of a reasonable thickness on a conveyor belt. Their research dealt with such components as car engine connecting rods and universal joint yokes.

When the conveyor was empty, the array saw the bright line of light. Whenever an object was illuminated, it would appear, from above, that the light beam was deflected to one side, out of the camera's field of view.

In fact, two projectors had to be used to prevent shadows, since the object first intercepts the beam but is not yet directly under the camera. A full binary image of the object was obtained by continually scanning as it was drawn through the field of view. Needless to say, that requires a good deal of housekeeping on the image.

There are a relatively small number of data extraction processes applicable to silhouette images and which are used to provide sufficient information to allow the program to identify the object from a set of possible objects and stable positions. To locate it accurately in terms of X and Y co-ordinates, or to identify sub-parts of it for detailed assembly see figure 4.

Assuming that the position and zoom settings of the camera remain unaltered and are calibrated, the area of the silhouette is a useful measure in identifying the part, figures 4c and 4d. That is extracted easily as the number of Pixels within the boundary of the image.

A second measure of the image is the length of the perimeter of the shape, obtained by counting the Pixels round the edge — figure 4c. The ratio area to perimeter length is a measure of the complexity of the shape of the object. For a given area, the greater the perimeter, the more convoluted it must be. The compressor assembly program used a measure called 'PEROUND', defined as:

(4*π*TOTALAREA)/
(PERIMETER*PERIMETER)

which returns one for a circular blob, less for more complex outlines — figure 4f.

Within the overall outline of the compressor body there were several dark blobs. All the interesting ones, bolt holes, cover aligning stud holes and the piston bores, were circular. They were isolated from other shapes of holes in the casting using peround, and then identified by their total area; the bores were the largest and the boltholes the smallest.

Once a circle is identified, its centre is found and the information is used to guide the manipulator. While bolts were screwed into the boltholes, force feedback sensors monitored the progress and a final visual check confirmed that the hole had been filled with a bolt and was now invisible.

In general, those measures work and in a reasonable time. It is generally assumed that if a computer-vision system for an industrial robot can reliably perform its

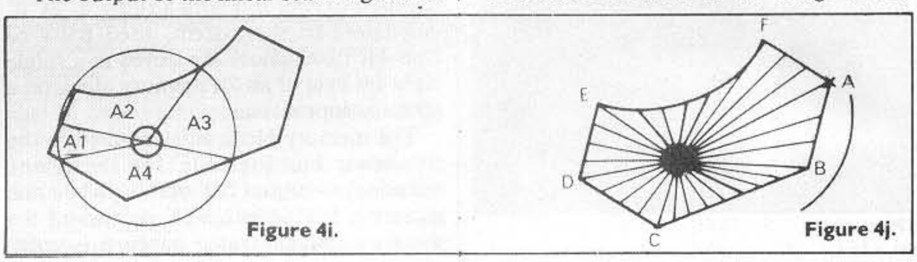


Figure 4i. Figure 4j.

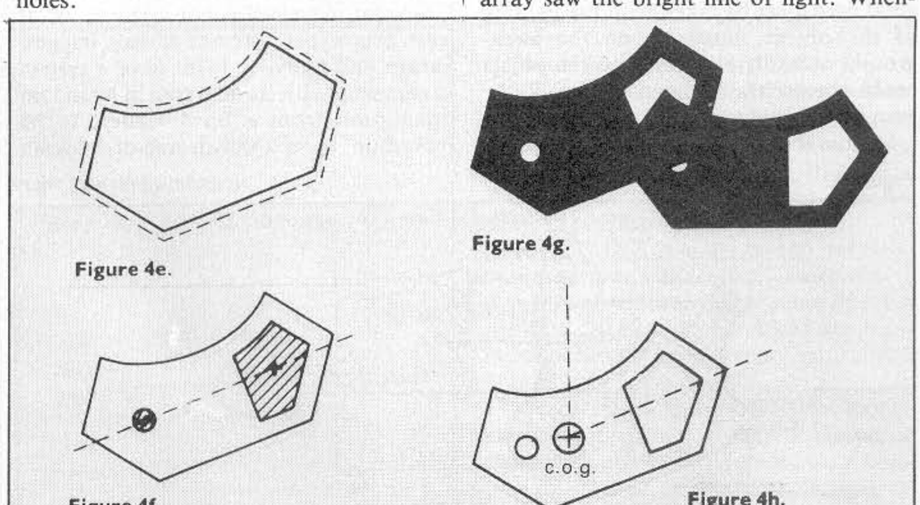


Figure 4e. Figure 4f. Figure 4g. Figure 4h.

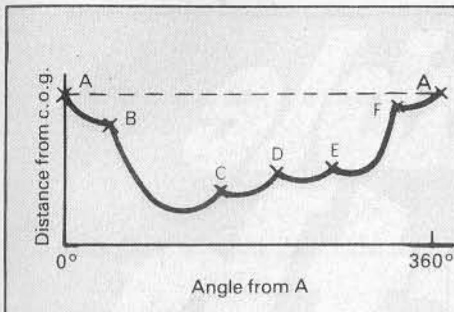


Figure 4k.

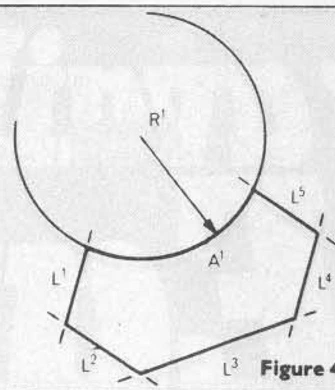


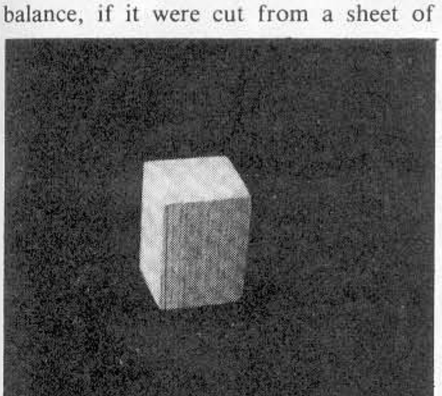
Figure 4l.

computation in a matter of seconds, it will be effective without making the manipulator wait for its instructions.

Those times are possible with current low-cost computers, and the algorithms are available as hardware/software packages — C1064, C1285, manufacturer's literature, Hamamatsu Ltd.

As more objects are added to the working set and the number and size of the differences between them becomes correspondingly smaller, so ever-more detailed features and differences have to be used to distinguish them. Figure 4l shows how the circumference of the shape can be represented as straight lines and arcs of circles.

Object orientation is a similar problem. If there are obvious features, such as the bore holes in the compressor body, their centres can be used to compute the information, i.e., figure 4f. Failing that, the centre of gravity of the image is evaluated — the point at which it would balance, if it were cut from a sheet of

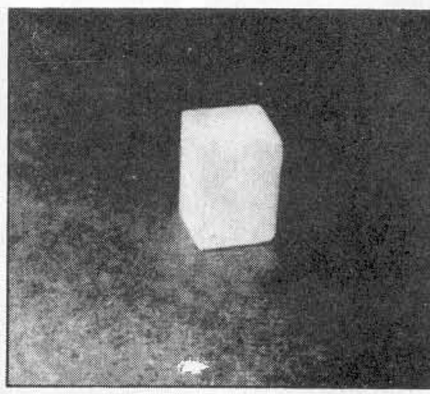


Picture 5.

metal. Figure 4h shows that by calculating the longest line to the edge, the orientation may be found.

Figure 4h is an alternative computation if this result were ambiguous — the ratios of A1 to A4 may also be used as a recognition measure. Saraga and Skoyles, 1976, discuss some criteria for choosing the size of the circle. Figure 4j extends these principles by computing many values from centre to edge and plotting them as an angular displacement graph — figure 4k. Martini and Nehr, 1979, cover the technique in more detail.

Dreyfus, 1974, describes hardware capable of scanning the outline of an object thousands of times a second. The device was used for feature tracking in both reconnaisance satellites and homing missiles. Kulpa, 1978, discusses possible problems to be encountered with calcul-



Picture 6.

ations for the centre-of-gravity method.

Perkins at General Motors, Perkins 1977 and 1978, used grey-level images, 256 × 256 to five bits from which the edges of the objects were extracted, resulting in a picture similar to a line drawing of the parts.

He then fitted a set of concurves to the line images, so that they were made from short line segments and arcs of circles. These relatively-organised descriptions could be compared to ideal descriptions of all the parts it might be.

As more of the features match between the current image and the stored model, the greater chance that the part will be identified correctly. As long as the match is above a certain confidence level, the process can stop.

So for most components only a portion of the object need be visible. In this way, where parts overlap or are only partially within the field of view, data from the model can supply the missing information.

Typically, it would be folly to try to attempt to isolate suitable features for recognition and use with an industrial robot 'by-hand'. After an initial calibration phase, during which the cameras are aligned and grey levels set, etc., the system is shown workpieces in typical orientations and stable states.

As each part is presented, the system runs through each of the parameter-extracting routines and builds-up its model of the parts. That process may be guided by the user if he feels some feature or property of the object to be particularly significant. When that is done, the manipulator can be programmed to use the information from the vision system to complete the task.

Three-dimensional vision, as is required by mobile robots, is an altogether stickier problem. Even after more than 12 years of intensive research by hundreds of people, an elegant, and computationally-fast solution to the problem remains elusive.

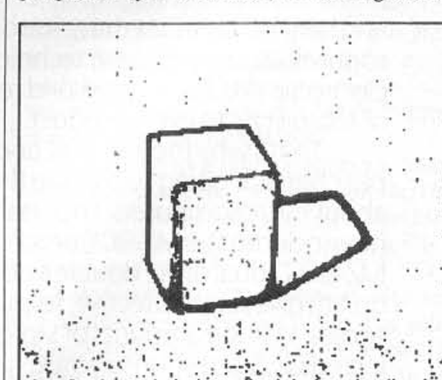
The techniques described for processing silhouette images are inadequate. Once an image is made binary, about half of the objects merge into the background, or with each other and become totally invisible. As the vehicle moves round, the apparent shape of all the objects changes, so simple model-matching will not work either.

One possible starting point is to extract all the lines that bound various objects in the scene. As an illustration, picture 5 and 6 show a small wooden block and the result of digitisation to eight bits.

One algorithm which might be applied is the first difference between each Pixel and its neighbours — picture 7. At each boundary, the accumulated difference will be high and if it above a specified threshold, the Pixel is printed black.

Such an operator is far from ideal — low-contrast lines, or ones which change over a relatively broad front, shadows, for example, are lost easily. Small random discontinuities and grey-level perturbations in the image are favoured strongly and noise can be a significant problem.

Unfortunately, as the threshold is lowered to capture more of the lines, still more noise is introduced. To combat those problems, operators which detect lines locally have been introduced. Picture 8 shows the effect of the Walsh operator, proposed by O'Gorman, 1976. In general,

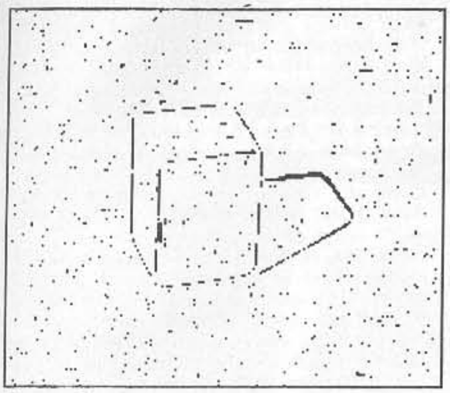


Picture 7.

the results are more manageable and freer from noise.

Hueckel, Hueckel, 1977, proposed a different operator, used by Perkins, but which is more expensive computationally. As none of the methods can be relied on to give perfect line drawings of the scene, various methods are employed to compute where the lines should be.

(continued on page 95)



Picture 8.

(continued from page 93)

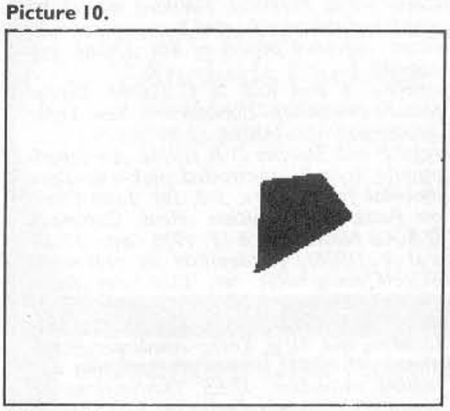
It is interesting to note that those programs seldom reconstruct the lines where a person would. A great many algorithms for analysing and post-processing images digitally are covered by Rosenfeld and Kak, 1976.

Once a scene has been reduced to a line drawing, a number of classic systems have been devised, such as those by Roberts, Waltz and Guzman — reviewed by Winston, 1972, or most books on artificial intelligence, e.g., Boden, 1977. Those systems variously separate, fit models to and identify parallel-piped, blocks-world, objects within the image.

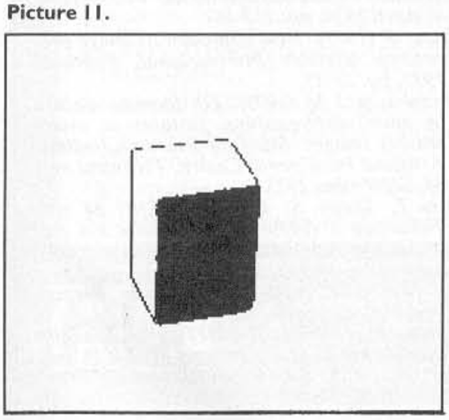
Such programs are based on the observations that lines constituting such objects form readily identifiable patterns where they join and cross. They infer the three-dimensional structure from a two-dimensional picture because the surfaces of the objects are constrained to meet in well-defined ways.

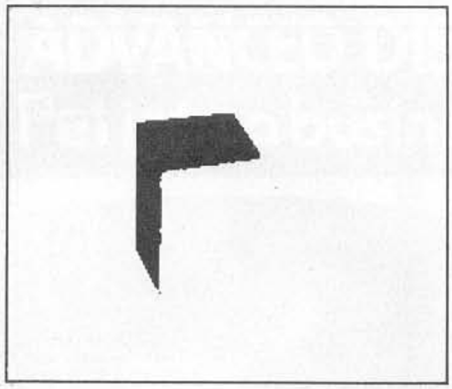
Those algorithms do not extend easily

Picture 10.



Picture 11.

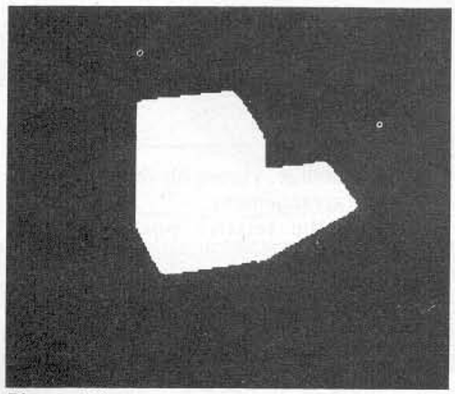




Picture 9.

to scenes containing objects with curved or irregular surfaces which are those of most interest to the robot user.

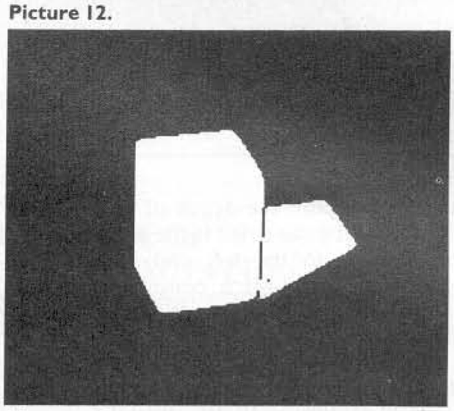
Perhaps a more fruitful approach is to



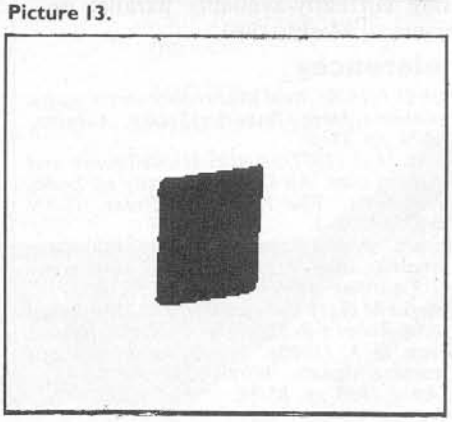
Picture 14.

attempt to isolate the regions in the image which correspond to the surfaces in the original scene, rather than to try directly to isolate the boundaries between them as

Picture 12.



Picture 13.

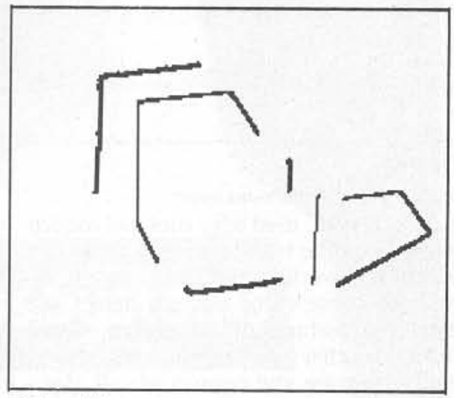


lines. Pictures 9, 10, 11 and 12 show a region segmentation of the image in picture 6.

In this case, a histogram of the grey value of every Pixel is formed on the assumption that each region will have a similar brightness — figure 5. The regions are then extracted as the major peaks in the histogram. Pictures 13 and 14 show the result of a simple but effective noise-removal program to tidy spurious parts of the image.

The histogram obviously works well when there are only a few peaks in separate places. As the scene contains more surfaces, so the grey-level histogram peaks will start to overlap and the segmentation becomes less reliable.

Surfaces with heavy-texture markings may also be segmented in a less-than-



Picture 15.

desirable manner. However, once the regions have been isolated, a good line extraction is made easily.

Picture 15 is an exploded composite obtained by expanding each of the regions by one Pixel round its perimeter and then ANDing it with each of its similarly-expanded neighbours; the boundaries clearly show where they overlap.

Brice and Fennema, 1970, isolated objects in Shakey's world using a region-extraction algorithm but as their model matching was limited to cubic- and prismic-shaped objects, the data produced may have been less than ideal.

In response to an earlier problem, image segmentation can be made on texture measures; Pixels are part of a uniform texture belonging to the same surface, Skalansky, 1978.

Tenenbaum, 1973, proposed a system to analyse pictures of an SRI office using colour and range information. From the range data it is possible to compute the orientation of surfaces, although small errors in range lead to gross shifts in apparent orientation.

The scene could be analysed using that data and knowledge about SRI offices. For instance, the floor is to be found at the bottom of the picture, horizontal and with a specifically-coloured carpet. Desks rest on the floor, have a different height and colour; telephones sit on desks, are black and so on.

(continued on next page)

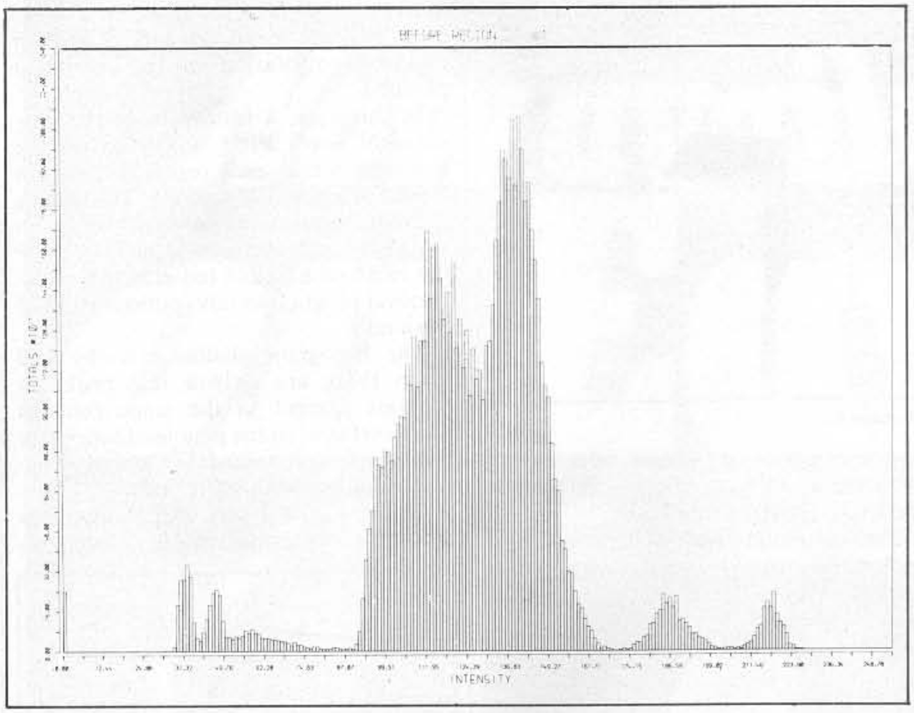


Figure 5.

(continued from previous page)

Shirai, 1978, used edge cues and models which could be transformed to fit images within real-world scenes, such as a desktop. All those vision systems detect and catalogue features of the objects within the scenes they analyse; the area, shape, colour, texture and corners are all identifiable. Features can be complex structures, with lesser features of their own, a telephone may be identified by its dial, a ring of 10 elipses, within a larger elipse.

Some or all of those different modalities can and should be used by a vision program, the danger being that the program will grow to exceed the machine and that the time it takes will increase from seconds to minutes, to hours.

All those systems use static information, The mobile robot is, however, able to move and capture another view of the same objects to resolve any difficulties. By projecting light beams into the scene, information may be obtained that is difficult to infer from conventional two-dimensional images.

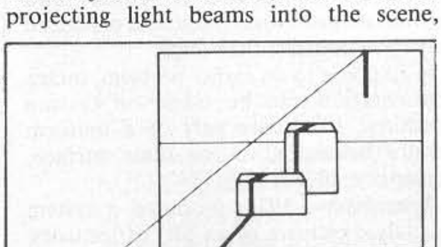


Figure 6a.

Figure 6a shows the effect of projecting a thin sheet of light vertically into a scene, from the camera's point of view. As the light penetrates more deeply into the scene, it appears further and further to the right of the image. Figure 6b shows a plan view of the arrangement.

Knowing the relative positions and angles of the light strip projector and camera and the lenses focal length, it is easy to compute the depth of any point illuminated by the strip. If the projector is then moved to the left and right it is possible to build-up a complete depth-map of the scene — Popplestone *et al.*, 1975, and Agin, 1979.

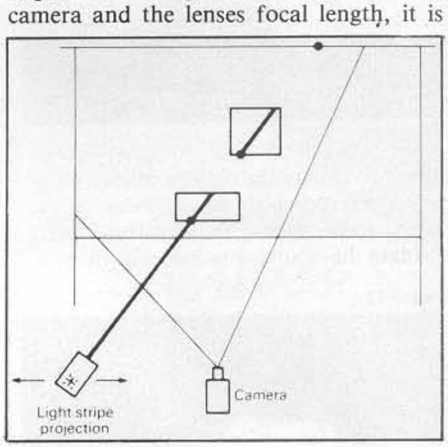


Figure 6b.

Philip Marks, has demonstrated the real-time advantages of processing images using currently-available parallel processors — Marks, 1980.

References

Agin G J (1979). Real-time robot control with a mobile camera. Robotics Today, Autumn, 1979, pp. 35-39.

Boden M A (1977). Artificial intelligence and natural man. An Open University set book. Hassocks. The Harvester Press. ISBN 0-85527-435-2.

Brice C R and Fennema C L (1970). Scene analysis using regions. Artificial Intelligence 1-3 Autumn, 1970, pp. 205-226.

Dreyfus M G (1974). Visual robots. The Industrial Robot 1-6. December 1974, pp. 260-264.

Green D K (1980). Improving photodiode camera signals. Wireless World 86-1532, April, 1980, pp. 85-88.

Hale J A G and Saraga P (1975). Control of a PCB drilling machine by visual feedback. Advance papers of 4th International Joint Conference on Artificial Intelligence. (IJCAI) held at: Tbilisi, Georgia, USSR, September 3-8, 1975. pp. 775-781.

Holland S W, Rossol L and Ward M R (1979). Consight-1: A vision-controlled robot system for transferring parts from belt-conveyors. Computer vision- and sensor-based robots. New York: Plenum Press. ISBN 0-306-40305-6. pp. 81-100.

Hueckel M H (1977). An operator which locates edges in digitised pictures. Journal of the Association of Computing Machines, Vol. 18, pp. 113-125, January 1971.

Kirsch J (1979). Progression of intelligence in limited sequence robots. Industrial robots vol. 2. Tanner W R ed., Michigan: Society of Manufacturing Engineers. ISBN 0-87263-046-3. pp. 277-284.

Kulpa Z (1978). Errors in objects positioning with centre-of-gravity method. The Industrial Robot 5-2, June 1978, pp. 94-99.

Marks P C (1980). Low-level vision using an array processor. Computer Graphics and Image Processing.

Martini P and Nehr G (1979). Recognition of angular orientation of objects with the help of optical sensors. The Industrial Robot 6-2. June 1979, pp. 62-69.

McGhie D and Hill J W (1978). Vision-controlled sub-assembly station. Robots III conference at Chicago, Illinos, November 7-9, 1978. (SRI report).

O'Gorman F (1976). Edge detection using Walsh functions. Conference proceedings of AISB Summer Conference, July 12-14, 1976. University of Edinburgh: Dept of A I, pp. 195-206.

Onda H and Ohashi Y (1979). Introduction of visual equipment to inspection. The Industrial Robot 6-3, September 1979, pp. 131-135.

Perkins W A (1977). Model-based vision system for scenes containing multiple parts. General Motors Research Laboratories Research Publication GMR-2386.

Perkins W A (1978). A model-based vision system for industrial parts. IEEE Transactions on Computers, vol. C-27-2, February 1978, pp. 126-143.

Popplestone R J, Brown C M, Ambler A P and Crawford G F (1975). Forming models of plane-and-cylinder faceted bodies from light strips. Advance papers of 4th. IJCAI. pp. 664-668.

Rosenfeld A and Kak A C (1976). Digital picture processing. London and New York: Academic Press. ISBN 0-12-597360-8.

Saraga P and Skoyles D R (1976). An experimental visually-controlled pick-and-place machine for industry. 3rd. Int. Joint Conf. on Pattern Recognition. Held: Coronado U.S.A. November 8-11 1976. pp. 17-21.

Shirai Y (1978). Recognition of real-world objects using edge cue. Computer vision systems, Hanson A R and Riseman E M eds. ISBN 0-12-323550-2. pp. 353-362. London and New York: Academic Press.

Skalansky J (1978). Image segmentation and feature extraction. IEEE Transactions on systems, man, and cybernetics. Vol. SMC-8, 4, April 1978. pp. 237-247.

Taylor R (1977). New approach to image processing. Systems International, February 1977, pp. 33-35.

Tenenbaum J M (1973). On locating objects by their distinguishing features in multisensory images. Stanford Research Institute Artificial Intelligence Centre, Technical note 84, September 1973.

Uno T, Ikeda S, Ueda H, Ejiri M and Tokunaga T (1979). An industrial eye that recognises hole positions in a water pump testing process. Computer vision- and sensor-based robots. New York: Plenum Press.

Winston P H (1972). The MIT robot. Machine Intelligence 7, Meltzer B and Michie D. eds. pp. 431-463. Edinburgh University Press, ISBN 0-85224-234-4.