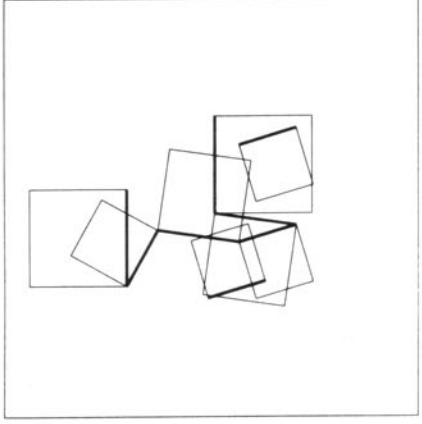
Not only computing—also art

JOHN LANSDOWN

Once again Manfred Mohr (*Computer Bulletin* March 1976) has given us some beautifully simple and attractive art works—this time based on a two-dimensional representation of a four-dimensional hypercube, that is a set of eight cubes put together in such a way that each edge of a cube simultaneously belongs to three different but adjacent cubes. Some of the drawings and construction he showed at his recent *Dimensions* exhibition in the Gallerie D & C Mueller-Roth in Stuttgart were created by choosing one distinct edge from each cube and combining these into a structure. To emphasise the ambiguity between the spatial representation and the two-dimensional drawing, squares were sometimes added to these edges (Figures 1 and 2). In another set, 16 drawings were arranged, magic-square fashion, in a 4 × 4 matrix so that any four drawings vertically, horizontally or diagonally added up to the complete hypercube representation (Figure 3).

Manfred has previously explored the way we perceive cubes and one of the most interesting aspects of the works based on the hypercube is that they cannot be understood in isolation from one another. Essentially, understanding comes from seeing all the works together—indeed perhaps from seeing all the works just as one work. Those unsympathetic to modern art often fail to appreciate the need to see such efforts in context and as part of a continuing examination of a particular idea. They expect all art to be as immediately accessible as the art of the past, not realising that the context in which that art was created is already familiar to us. One of the aims of the Computer Arts Society is to try and set the work of computer artists into a context more readily understandable to the public.

Figure 1

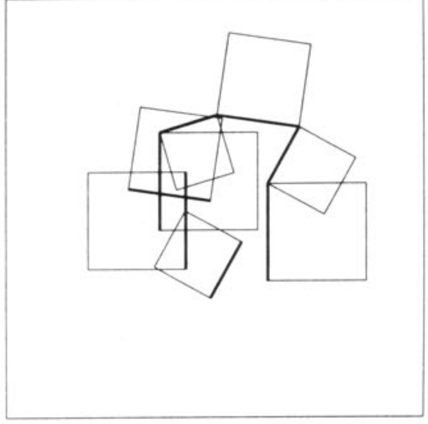


Hiding the line

A great deal of research and implementating effort is being given to the so-called 'hidden line' problem which arises when computers create perspective drawings of scenes and objects. Computers have little difficulty in drawing such things as if they were transparent thus showing all the edges where different planes meet. They are, however, heavily taxed when they have to eliminate all the hidden lines, edges and planes which could not be seen if the objects were solid. This is essentially a computing problem which has no counterpart in real life. When confronted with a 'wire-frame' drawing of even quite a complicated object, people have little difficulty in recognising it and transforming the image into a drawing of a solid object showing only the lines and faces that would actually be seen in reality. Strangely, it is probable that this ability is not innate but is both learned and culture-dependant because the appreciation of perspective as we know it is a (comparatively) recent phenomenon. The reaction of the late 19th century Japanese to the first exhibition of western art to be held in Japan was that the pictures appeared to be too flat and had no depth. (Compare this with the western view of oriental art!)

Many solutions to the hidden-line problem have been put forward and incorporated in programs but all require quite substantial increases in computer time over that needed for wire-frame drawings. A new and clever solution has been put forward by J. G. Griffiths of Salford University and is outlined in the March 1979 issue of *Computer Aided Design* where the reader should look for further details. His method relies on fast sorting of data (one of the most important

Figure 2



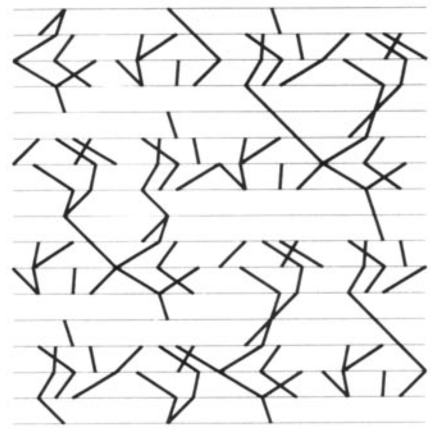


Figure 3

Figure 4

aspects of nearly all solutions, as was pointed out by Sutherland, Sproull and Schumacker in their now-famous 1974 *Computing Surveys* article), but it is in the address calculation algorithm for quickly finding the faces which might possibly obscure a given edge that Griffiths scores. His technique allows very complex objects such as Figure 4 to be drawn in a reasonable time so that, for objects with a large number of faces, the computing time rises as the 1.33-power as opposed to the 2-power of many other methods. Figure 4 consists of four tables and 16 chairs and required 227 seconds of computing. Other similarly complex drawings are shown in his article.

The more general availability of raster scan graphics devices where full colour shaded drawings can be produced has given a boost to further research into the hidden line or, to be more accurate in this case, the hidden-surface problem. One can imagine an easy way of drawing a fully coloured picture of an object would be to draw the surfaces in order with the surfaces furthest from the eye drawn first and the nearer and nearer surfaces superimposed over these finally leaving visible only those faces or parts of faces which would be seen in reality. In fact, it is simple and quick to test which surfaces could *never* be seen (because they face away from the viewer) and these need not be drawn at all so that only those which *might* be seen need to be included in the superimposing process. In the case of a convex object, that is one without any re-entrant surfaces such as Platonic solid, this latter test is sufficient to eliminate all the hidden surfaces anyway. In complicated objects, however, the problem in deciding which of a pair of faces is nearer to the viewer is not always straightforward and crude methods of distance determination do not work. P. J. Willis of Sussex University has ingeniously solved this problem in a way which makes it always possible to use unambiguously a simple measure of distance from the eye as the criterion for deciding which faces obscure others. The method is written up in *The Computer Journal* 1977, vol 20 No 4 and *Computers and Digital Techniques* 1978, vol 1 No 4. Essentially his simple technique involves 'unclustering' and dividing the surfaces of a complex solid in a way which allows one to use the distance from the eye to any convenient point on the surface—such as a vertex—to establish the correct priority order for processing. The real value of the idea is that the unclustering procedure is independent of the viewpoint taken and needs to be done only once for a given object (or collection of objects in fixed relationship) so that the process of producing a series of pictures of a scene from different viewpoints is substantially shortened. Of course, the reason that people find the hidden line and surface problem fairly easy and computers find it hard is that people know things about reality that computers don't and, until computers can learn about such things as transparency and opacity, this situation is likely to continue.

Art and perception

In the static exhibition of the Computing funfair in January, six computer assisted drawings of Dominic Boreham were of particular interest. Copies of some of these are on loan to the BCS and are displayed at Headquarters and anyone who missed them at the funfair should make a point of seeing them there. Like Manfred Mohr, Dominic is interested in the way we interpret the things we see and his drawings are not meant to be pictures of the world but explorations of how we see the world. In his explanatory notes to the exhibition he says:

One of the most interesting clues which inform our perception of objects in space is the obscuring of a distant object by a nearer one. Overlay is just one of a number of depth cues, that is, cues which enable us to build a mental picture of objects existing in space.

He points out that we interpret an outline drawing of one rectangle overlapping another as if the overlapped rectangle is behind the other and that this is a mental process which is upset only when we encounter ambiguous drawings. Of course, rectangles can be suggested not only by outline drawings but also by a series of properly arranged parallel lines, such as those shown in Figure 5. If this drawing is overlaid on a ground such as Figure 6 we obtain Figure 7 in which some squares seem to be in front of the ground and others behind it whilst yet others seem to disappear altogether or fragment into smaller rectangles. The whole series of drawings are worth the closest study and were prepared on an ancient Benson plotter connected online to a NOVA 2 mini at the Slade School of Fine Art where Dominic is a postgraduate student.

Data base take-away

For the past year, I have been engaged on a detailed study of data base design and evaluation. Can anyone explain why so many of the workers in this field have Chinese names? Or am I mistaken in my impression that the proportion of Chinese dealing with this subject is greater than that in computing as a whole?

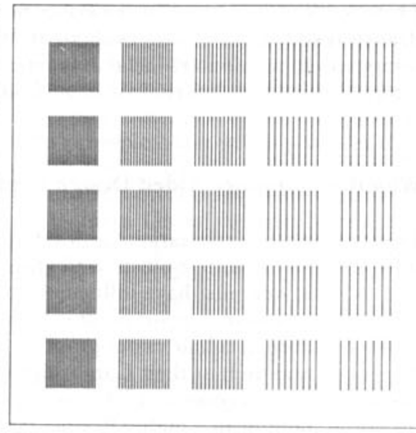


Figure 5

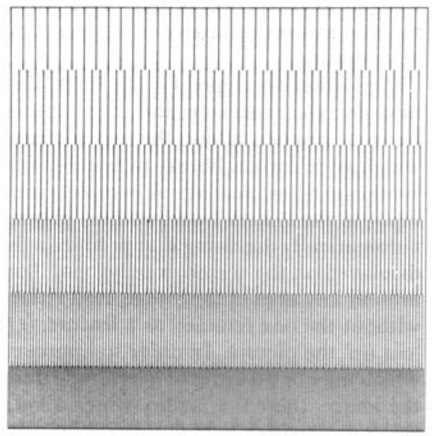


Figure 6

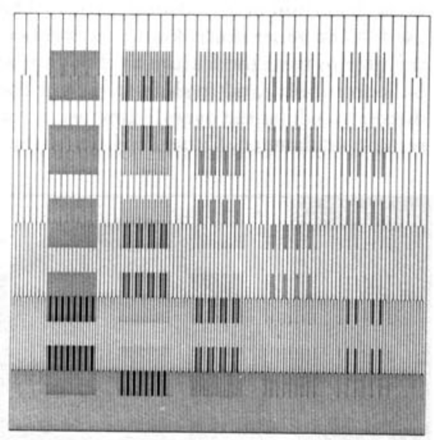


Figure 7