Parallel to Perception:
Some Notes on the Problem of Machine-Generated Art

HAROLD COHEN
University of California at San Diego

In a very large number of applications the computer is used for its ability to perform a set of predetermined transformations upon a set of data, and this kind of use has become standard in 'computer art', where the data is some original provided by the artist. If the aim is to mode! human art-making behavior, rather than merely to use the machine as a too! in this quite traditional sense, such a definition of the machine's functions is inadequate. Human art-making behavior is characterized by the artist's awareness of the work in progress, and programs to mode! such behavior will need to exhibit a similar awareness. Thus, 'behavioral functions' are defined here as functions which require feedback from the results of their actions as a determinant to their subsequent actions. Programs designed upon this specification will also require appropriate schema for the description of the work in progress.

The feedback systems employed in intelligent behavior might be pictured as the asking of questions about the perceptual world whose answers will be relevant to decision-making. For the machine, 'awareness' of the work is totally defined by this question-and-answer structure, and in this sense is equivalent to the human perceptual system. It is not clear what descriptions of the work will serve for a reasonable simulation of human art-making behavior, or what questions will need to be asked. They will not necessarily reflect the 'facts' of the human system, but it seems likely that the machine's feedback system as a whole will need to possess a comparable adaptiveness to permit of the fluently changing pattern of decision-making which characterizes the practice of art.

If a photographer takes a picture, we do not say that the picture has been made by the camera. If, on the other hand, a man writes a chess-playing program for a computer, and then loses to it, we do not consider it unreasonable to say that he has been beaten by the computer. Both the camera and the computer may be regarded as tools, but it is clear that the range of functions of the computer is of a different order to that of the camera. Tools serve generally to extend or to delimit various human functions, but of all the many tools invented by man only the computer has the power to perform functions which parallel those of the mind itself, and its autonomy is thus not entirely illusory. The man actually has been beaten by the machine, and if the program was structured appropriately its performance might be considerably better, in fact, than it was when he first loaded the program.

If we acknowledge the machine's autonomy in this kind of situation, would it not seem reasonable to consider the possibility of autonomous art-making behavior, not in the trivial sense that it can control the movements of a pen, but in the sense that it can invent those movements? Would it be possible, for example, for the machine to produce a long series of drawings rather than a single drawing, different from each other in much the same way that the artist's would be different, unpredictable as his would be unpredictable, and changing in time as his might change?

If the answer to this question is that it would, then it would seem to follow that some of the machine's functions will need to parallel, at least in a primitive way, some aspects of the human perceptual process. The drawing will be the real world for the computer, just as it is a part of the real world for the artist: and just as the artist will deal with the drawing in terms of gestalts rather than in terms of raw data, so the machine will need to formulate characterizations of the current state of the drawing, rather than treating it merely as an agglomeration of marks and non-marks.

I should say that I consider the possibility of this kind of behavior to be a real one, and not merely speculative, for the reason that my own work with the computer has

Reprint from Computer Studies IV-3/4, 1973

gone some part of the way towards realizing it, as I will try to show; far enough to suggest that the rest is attainable. But I should define my position more carefully, for the simulation of human perceptual processes arises as a result, rather than as a motive, for this work. Whatever other territories may appear to be invaded, I believe that my behavior in programming the machine to simulate human art-making behavior is, in itself, primarily art-making behavior, and I have proceeded by attempting to deduce from the requirements of the venture as a whole what perception-like abilities may be appropriate. The plausibility of the resultant structure must thus rest upon the success of the whole system in satisfying its purpose, rather than upon whether it appears to provide a satisfactory model of perception. If the whole system can autonomously generate art — autonomously, that is, in the obviously qualified sense used above — then we will know something more about ways in which art may be made, and conceivably something about the way in which human beings make it; but not necessarily about the specific mechanisms upon which human art-making rests.

The purpose of this essay, then, is to say something about the nature of the characterizations, or representations, of the work in progress which the machine will need to build; and about the constraints under which they are formulated. Unfortunately, so much confusion now exists in the interface areas between art and computers, thanks to the strange manifestation popularly known as 'computer art', that some clarification will be needed before I can get to my subject.

Evidently the power of image-making retains something of its primitive magic even in a society as familiar with images as our own; and, like the camera, the computer seems to exert a democratizing influence, making this power widely available, where it was at the disposal, previously, only of an elite with the skills and abilities to exercise it. Image-making is in the province of anyone with the price of an Instamatic and a roll of film; anyone with access to a computer and a little programming experience. The programmer starts with carefully digitized Snoopy drawings, progresses to rotating polygons, and by the time he gets to polynomial functions he is ready for the annual Calcomp Computer-Art Contest.

For most people outside of art, probably, art is directed primarily at the production of beautiful objects and interesting images; and who is to argue that a complicated and intricate Lissajou figure is less beautiful than an Elsworth Kelly painting or a Jackson Pollock; or that a machine simulation of a Mondrian is less interesting than the original it plagiarizes? To talk of beauty or of interest is to talk of taste, and matters of taste cannot be argued with much profit. The fact is that art is not, and never has been, concerned primarily with the making of beautiful or interesting patterns. The real power, the real magic, which remains still in the hands of the elite, rests not in the making of images, but in the conjuring of meaning. And I use the word meaning in a sense broad enough to cover not only the semantic content of the image itself, but all that is involved in the making of the image.

The particular kind of usage the computer has received in almost all 'computer art' offers some clue as to why 'computer art' can barely claim consideration as art at all. With a few notable exceptions the machine has been used as a picture-processor, which is to say that it is programmed to perform a number of transformations upon material previously defined by the artist. In this role it has something in common with other processes used traditionally by the artist, and yet it has failed to support the dynamic interplay we normally expect between a process, the art-making intentions which give rise to its use, and the formal results of that use. The cause of this failure may be the relative inflexibility of the processes available, but I am inclined to believe that it is dictated by the whole structure rather than by inadequate implementation of the structure. It should go without saying that it is beyond the power of a process to invest an image with significance where none existed before; that if you cannot draw without a computer — and by drawing I mean the conjuring of meanings through marks, not just the making of marks — it seems unlikely that you can draw with one. At all events, it is clear that the use of the computer as a tool in the sense that a camera is a tool represents the antithesis of autonomy, and is thus not my subject here.

All the same it will be worth examining the notion of picture processing as a starting point in order to see how other possibilities relate to it. Diagramatically (Fig. 1) we might think of the processor as a black box, with a slot at the top through which original material is fed, and a slot in the bottom through which the processed material exits. The range of possible processes, or transformations, is fixed for any given configuration of the machine, but they may be selected and concatenated by the user, who also has a measure of quantitative control over their application — if he chooses to transform an image by rotating it, then he can specify how much it is to be rotated, for example. The actual number of processes which can be programmed is large, but, as one might almost anticipate, the few examples of exceptional quality which have occurred have tended to make use of

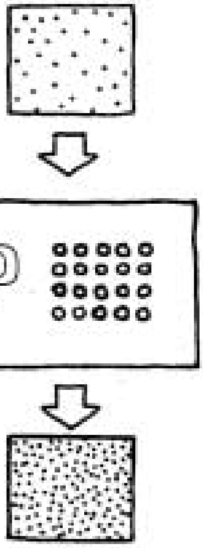

Fig. 1

quite minimal processing functions, and instead of burying the original image, lay emphasis precisely on the metamorphosis itself. One such example would be Charles Csuri's manipulations of the Leonardo *Vitruvian Man* (Fig. 2), in which the figure retains its identity even after being distorted, rubber-sheet fashion, in a number of unlikely ways. The manipulations are effective for reasons of content rather than form, however, and are even part of that content, since the *Vilruvian Man* is a symbol for those notions about the proportions of the human body which run through neo-Pythagorean cosmology for some hundreds of years of European thought, and it is precisely the proportions of the figure which Csuri has chosen to manipulate.

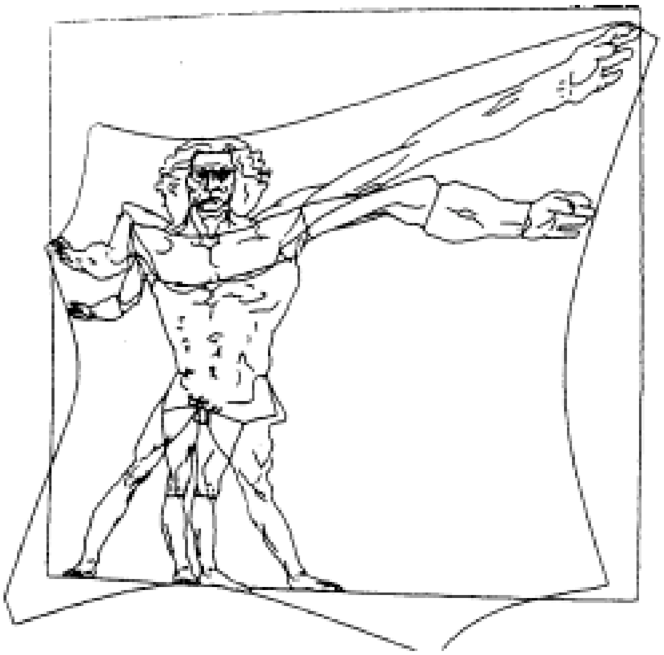

Fig. 2. Charles Csuri, "Circle to square transformations, based on Leonardo's *Vitruvian Man"*.

From a processing point of view, the drawing (Fig. 3) by Kenneth Knowiton and Leon Harmon does no more than to replace small square patches within the original photograph with an alphabetic character, chosen from a range of different type faces, each with a proportion of white character to black space equivalent to the gray level in that patch. The processing function is a simple, one-step affair, and again, the interest lies behind the image itself, since Knowiton and Harmon are concerned primarily here with the nature of visual information; not only, WHAT is read, but, HOW it is read. Even so, the choice of original material is evidently by no means entirely neutral, since the text which is used for transfiguring the picture of the Vietnamese child is in fact the Declaration of Human Rights.

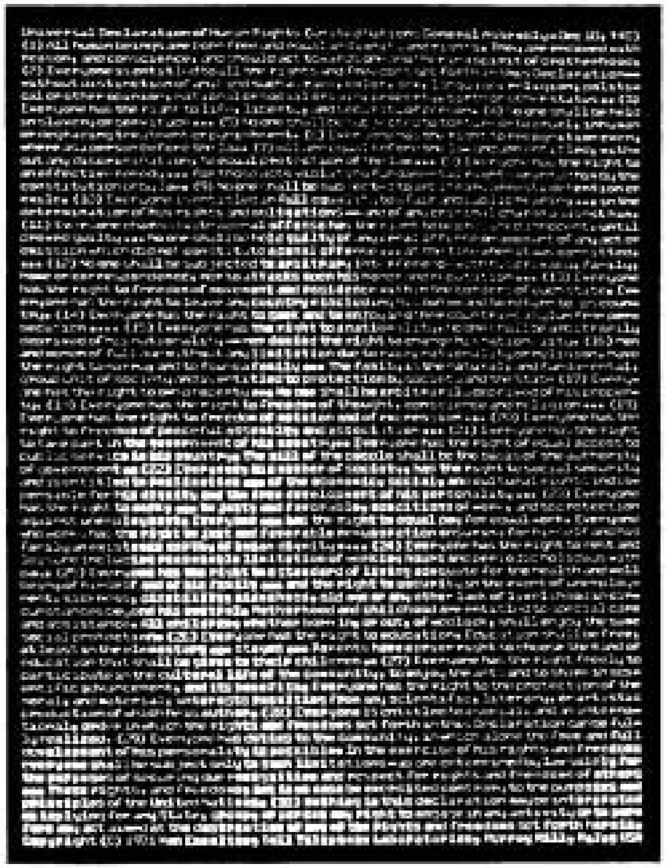

Fig. 3. Kenneth Knowiton and Leon Harmon, "processed photograph – Vietnamese child."

Processes may range from the simplest geometrical transformations to highly complex systems in which images of objects are not only rebuilt from the collection of three-dimensional coordinates which represent them, but can be shown moving around in correct perspective, complete with the shading and shadows caused by any specified lighting conditions. In practice, the machinery for operating these transformational systems may vary enormously, and the wholly electronic nature of the more modern devices allows them to operate at speeds which give the illusion of a direct interaction between the user and the process. Rather than looking at the processed image drawn on paper, then resetting the parameters for the processing functions and starting again, the image can be displayed on a screen, and will change as fast as the knobs are twiddled (Fig. 4). (In fact, operated in this way, the processor inevitably takes on some curious affinities to musical instruments: one seems to be playing it, rather than playing the image with it, and the problems relating to the (outside of music) unresolved differences between improvisation and composition which have already appeared in the use of the video-synthesizer must arise here also.)

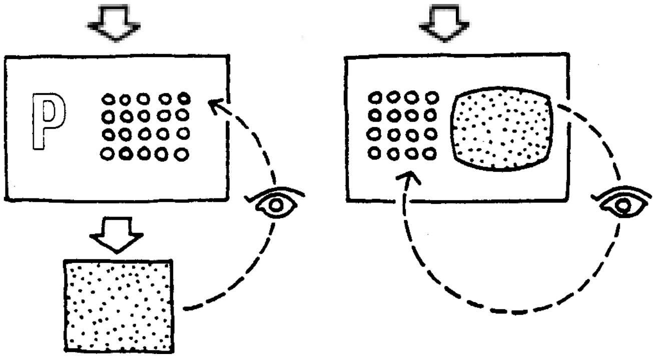

Fig. 4

As far as the manipulation of the image is concerned, however, the speeding up of the system might induce a change of attitude, but does not represent a change of structure. What all these things have in common is their diagrammatic representation — the original going in at the top, the processed image coming out at the bottom, selective and quantitative control only over the process — and also the fact that at no point does the machine need to read back what it has done. By definition processing is a deterministic affair, and for any single run its functions are predetermined and invariant. Feedback from the result to the functions themselves has no part in this process. On the other hand feedback is clearly a part of the human art-making process, or indeed of any intelligent process, and if the only feedback possible within the computer environment is via the human user, then the computer is a tool in no essential way different from any other tool: and it is evidently capable, up to this point at least, of handling material of much less complexity, and in much cruder ways, than most of the artist's more fertile, and more traditional, tools.

Suppose, now, that we wished to modify this schema in some structural way, hopefully to arrive at something corresponding, in itself, more closely to human art-making behavior. What modifications would be possible, and what could we deduce in relation to them? We have already seen that increasing the number and range of the processes — adding more knobs and switches to the control panel, as it were — would make no fundamental difference: just as increasing the speed of the system would make no fundamental difference. Closing off the slot in the bottom of the box would simply render the system inoperative, in that there would be no result. If the slot in the top were closed off, could the system provide itself with original material upon which to function?

The first answer would seem to be that it could. In fact, of course, the machine works on a description of the original rather than on the original itself, and although I have tended to write as though the description were always given as a set of points — that is to say, as a digitized version of an original — there are other ways of describing pictures. For that matter, sets of points could be included in the program which defines the processor rather than entering them after the program had been set up, although doing so would be merely a device, and would severely limit the versatility of the system. Other kinds of description, like the use of mathematical equations to describe curves, probably could not be entered conveniently after the program was set up, and would more properly form part of the program itself. Here, too, versatility would be seriously limited, since there is a relatively low limit on the number of kinds of curves which one might realistically hope to describe by means of mathematical functions. In any case, what becomes clear is that the question more usefully to be asked is not whether the machine could function with its input slot closed, but whether its program could actually GENERATE material, as opposed to being given it as it needs it on the one hand, or being given it in advance on the other.

Note that, although it is factually true that a mathematical function can generate a set of points, I have treated it as a storage device rather as a generator, precisely equivalent to the list of points it will generate. I have done so for the reason that a curve is fully described by its equations, just as it is fully described by the set of points of which it is comprised. But suppose we were to find some way of writing a program that required no preliminary input, no 'original', that did not make use of mathematical functions in place of input; that nonetheless succeeded in generating a graphic result; would it not then be true that the program as a whole fully described the image? That it was, in effect, exactly equivalent in that respect to the mathematical function?

The answer I will give to this question is that programs can be written which do not fully describe the images they generate in the same sense that a mathematical function does. But we should examine the implications behind this answer with some care, since it appears to involve the question of whether a machine might be capable of non-deterministic behavior. I have some doubt whether any definitive answer can be given to this question: whatever more rigorous definitions of the term 'non-deterministic' might be available in other disciplines, it seems to me that here it relates to what we think human behavior is like at least as much as it does to what we think machine behavior is like. Thus it seems to become a problem of definition rather than a problem of identification, and my own question was proposed as a more meaningful alternative to it. And in answering this question more fully, I will try to demonstrate the possibility of what I will term behavioral functions, which differ from mathematical functions in that they require feedback from the image — the image in progress, that is — and contain the necessary feedback mechanisms within their own structures (Fig. 5). It would seem reasonable to say of such functions that they do not fully describe the images they generate in the sense that mathematical functions do.

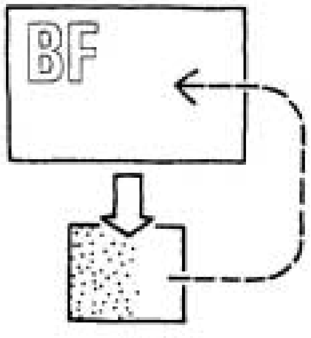

Fig. 5

Before going on to describe programs of this sort, and to talk about the nature of the feedback, and of feedback interpretation, I should deal more thoroughly with the initial premise; the notion, that is, of a machine which can provide itself with its own original material. This innocent-sounding suggestion reveals itself to be even more troublesome than the idea of non-deterministic behavior, or perhaps simply a more troublesome formulation of the same idea: since what appears to be implied is nothing less than the machine initiating its own performance, and initiating formal material. Once again, the question is heavily colored by our beliefs concerning the nature of human behavior. But to what extent could we reasonably maintain that the human mind initiates? Concepts are formed on the basis of prior concepts, decisions are made on the basis of feedback from the environment and from the results of previous decisions. The probability is that, if one could identify the starting point for an artist's whole life's work, one would find a set of concepts completely formulated if not completely digested, given to him and not initiated by him. We habitually speak of the artist 'beginning to find himself at some date much later than this starting point: the artist himself will tend rather to speak of his life and his work as a continuous self-finding process.

Thus the question of starting points and starting material is misleading in relation to the machine's performance, not because the machine could or could not initiate material, but because the idea of the machine being loaded with a program, running the program, and stopping, forms a discrete unit which has no real parallel in human behavior. What we would need to imagine to establish a reasonable parallel is a machine equipped with an archival memory, running a self-modifying program not once, but hundreds or even thousands of times, modifying future performance on the basis of past performance (Fig. 6). In this state, the nature of the initial input might

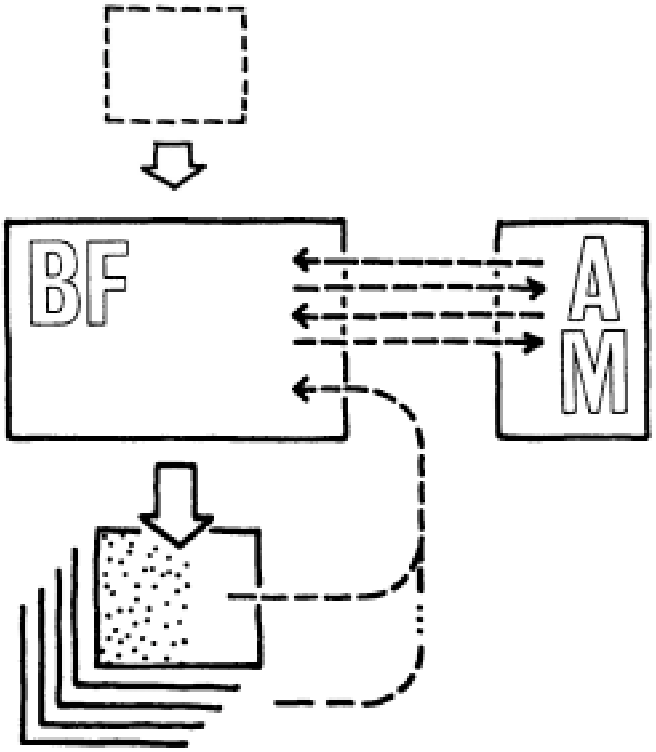

Fig. 6

be of no more importance to the final outcome than the name and style of an artist's first teacher.

I do not believe that the existence of such a machine is around the nearest corner: and there is no doubt that before we get to it, and to other machines which, like it, would profoundly challenge man's thinking about his own identity, there will be emotional roadblocks of significant proportions to be taken down. But, of course, it has already been demonstrated that machines can learn, given appropriate criteria for performance: and conceivably the idea that no such criteria can exist in art will prove to be simply one of the roadblocks.

In practice, it is not possible to run a program from scratch without providing initial material. You cannot tell the machine, "draw some circles", you have to tell it how many circles you want drawn, as well as specifying a general program for drawing circles, how big you want them, and where they are to be placed. But it is possible to have the machine itself decide these things, and the programmer can make use of the machine's random number generator for this purpose. You tell the machine "draw some circles — anything from ten to thirty will do. I want them not less than an inch in diameter, and not bigger than three inches" ... and so on. We should not be too impressed with this ability: it is no more indicative of intelligence in the machine to make decisions randomly than it is for a human being to make his decisions by tossing coins. Intelligent human beings make their decisions this way only when the outcome does not matter, and what is at stake here is the programmer's tacit declaration that his program will function to give satisfactory results regardless of whether it has fifteen circles or twenty-five to work on. What we might anticipate from the hypothetical learning machine is that parameters would be initially set as random choices over very wide ranges, and that the machine would itself narrow those ranges down to the point where specific values could produce specific results.

Let us turn now, finally, to the question of feedback, and of what kind of programs one might imagine could be built up given appropriate feedback structures. Any complex, non-organic system must make use of feedback structures to keep it in a stable state, just as any organism does. In the computer there will be such mechanisms functioning at electronic level, but these are not the ones under consideration since they are operating regardless of what the machine is doing, just as the body uses feedback to control its temperature regardless of what the mind is thinking. Similarly, organisms have feedback structures to control their physical movements, and while something of the sort may be built into a computer program, they do not come high enough in the scale of activities for organisms or for mechanisms to be considered intelligent. The program which generated the drawings in Fig. 12, for example, has at its lowest level a sub-program which draws lines between pairs of points which have been determined higher up in the program. The sub-program uses a sort of homing strategy: it wants to wander freely as if it had no destination, but at each step it corrects its path, so that it arrives at its destination nevertheless, without overshooting and without needing to spiral in as a moth does around a candle flame. While the feedback structure is more sophisticated than the moth's, equivalent perhaps to those we might employ in driving a car, it is of essentially the same order, and the structures exhibited higher up in this program are of a different kind. I will return to these in a moment.

I use the term 'feedback' in the most general sense, to denote, within a system, the passage of information back TO a function FROM the result of the operation of that function, such that the operation tends subsequently to be modified. In intelligent systems, we might thus characterize feedback, in a rough *ad hoc* sort of way, as the asking of questions relevant to continuing operation, and might even describe the complexity of the feedback system in terms of the number of questions the continuing operation requires.

This is not to say, however, that the complexity of the result necessarily depends upon the complexity of the system. One of the most amazing examples of an apparently simple system yielding complex results is the 'Game Of Life', which has received enough attention recently not to require further description. In this matrix-manipulating program, the asking of the same single question for each cell in the matrix — how many of its neighboring cells are live and how many are dead? — is enough to provide for the generation of a rich set of patterns often possessing remarkable characteristics. If I would exclude the 'Game Of Life' from the class of systems I am discussing it is because, although it appears to be asking its question of the result of an operation, the operation itself never actually changes at all, and the result of one application of the operation simply becomes the input, the original, for the next application of the same operation (Fig. 7). In other words, the system should properly be considered as an iterative processor of ingenious design, in which the complexity of the result stems from the iteration rather than from the process itself.

In practice, it is quite difficult to avoid the appearance of this processing structure in computing, since the whole methodology of programming is built upon the notion of

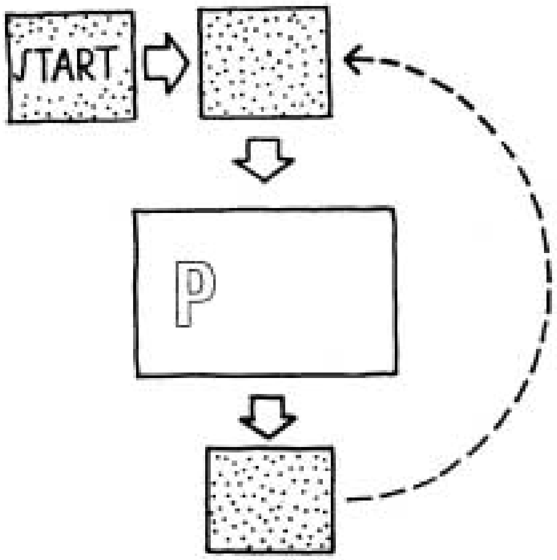

Fig. 7

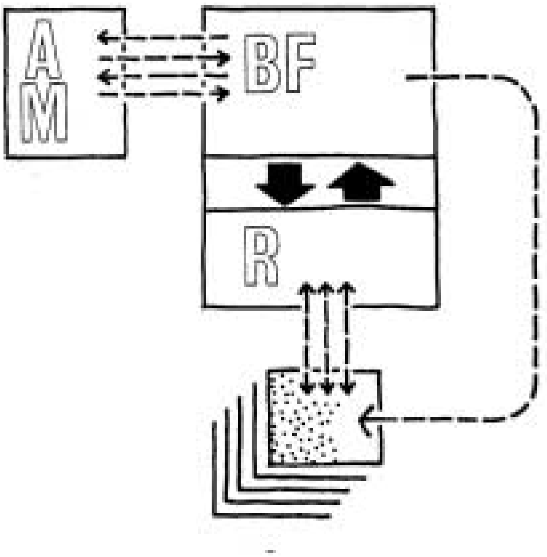

Fig. 8

iteration. But to say that processing structures are embedded in a program is not the same as saying that the program itself functions as a processor. The distinction becomes important if we pursue the idea that feedback complexity may be measured by the number of questions which need to be asked in order to determine subsequent operation, for clearly the total number of questions to be asked within the whole program to provide a single unit of information will depend upon a number of issues, not least of which is the availability of that single piece of information. We might imagine an artist having a piece of work made by telephone, updating his mental image of the piece by asking questions. "How many lines are there in the drawing now?" he might need to know: and he has no interest in whether the person on the other end will need to go and count them yet again, or whether he has been smart enough to keep an updated record. For our purposes we would say that only one question has been asked. Similarly (Fig. 8), the computer program could be considered as a two-part affair, in which the upper part — the 'artist', as it were — accesses the work in progress by interrogating the lower part about it. Our measure of feedback complexity is then given by the number of loops between the two levels, and is not concerned with how complex are the functions occurring within the lower part, many of which will certainly appear as processing functions in the sense we have already discussed. In many ways, then, the upper part might be thought of as using the lower part as the human user uses a processor.

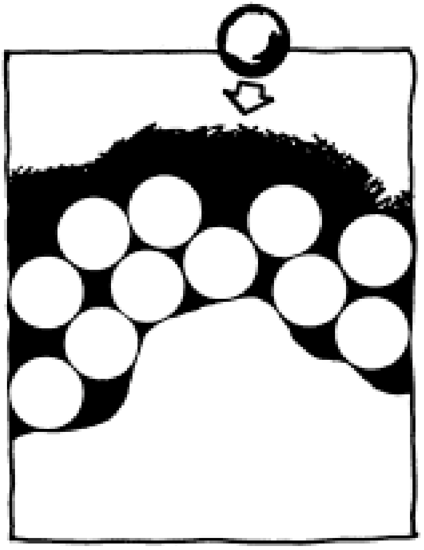

Fig. 9

If I were to write a program which packed equal sized circles into a hexagonal array, no feedback at all would be required, since I could calculate in advance where all the centers would need to be, and would know without looking that there would be no overlapping. A program which sought to pack the same circles over and around an irregular projection (Fig. 9) would be a different matter, however, since the space available for each new circle could not be known until the previous circle had been drawn. Similarly, a program which caused a number of loosely distributed circles to grow irregularly, amoeba-like (Fig. 10), until all the available space had been absorbed, would need to ask, for each new development of each developing shape: what is the current state of its boundary? What shapes lie in the direction of the proposed development, and what is the state of THEIR boundaries? Has this developing shape reached those existing boundaries yet? The end-state of the whole development (Fig. 11), which has been managed entirely by the program, has thus evolved from a unique set of events, and the large set of drawings which actually resulted from a long run of this program exhibited wide variety without any change in the operating parameters.

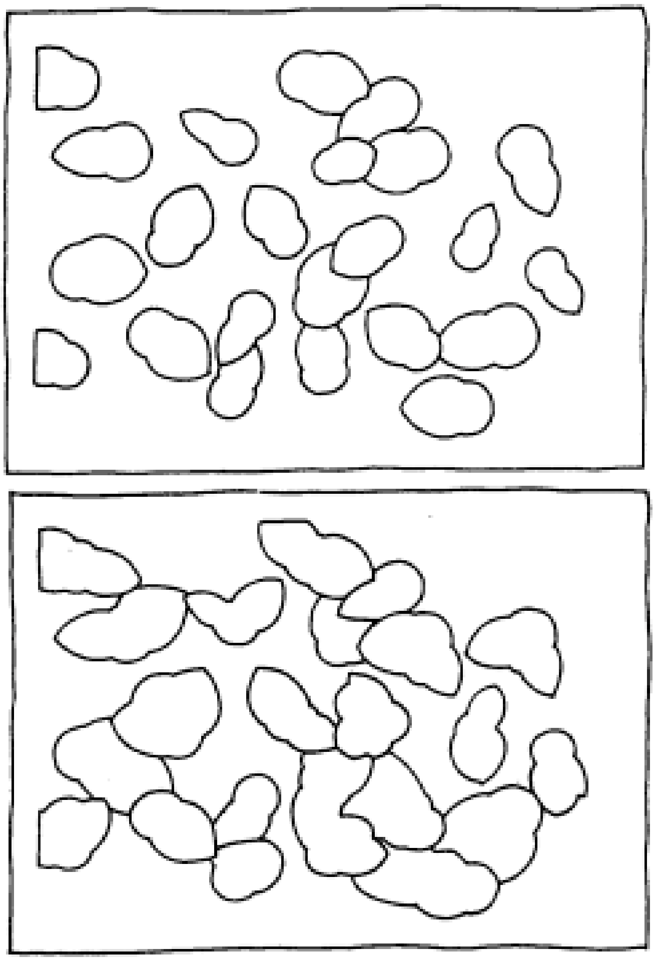

Fig. 10

A more recent program of mine ran continuously for a month during a museum exhibition, and, again, required no human intervention in making over three hundred drawings (Fig. 12) beyond changing the sheets of paper on the drawing machine and refilling the pens. The feedback in the current version of this program is more complex than in the previous example, but not by much, since to draw each new line the program needs only to know which of many possible destinations may be reached without crossing an existing line, and, of them, which is the nearest and which the furthest away. But the program is structured in such a way that more particularized decisions may now be reached on the basis of more extended information requiring more complex feedback — which of the possible destinations will result in the straightest lines? Which is closest to the center of

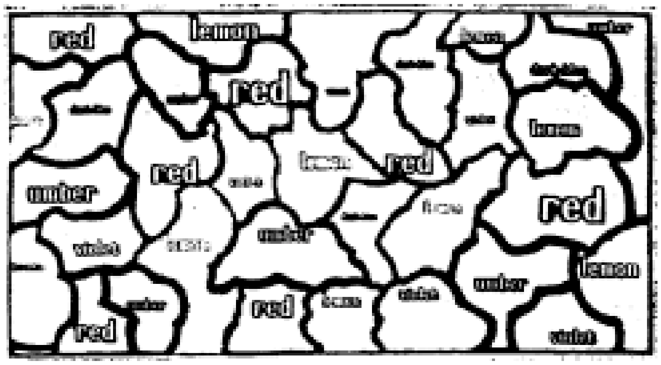

Fig. II. Harold Cohen, "Labeled Map" (1969), 102" x 192", oil on canvas.

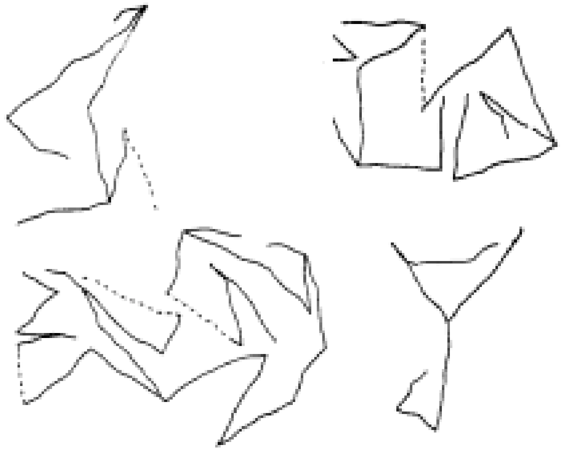

Fig. 12A

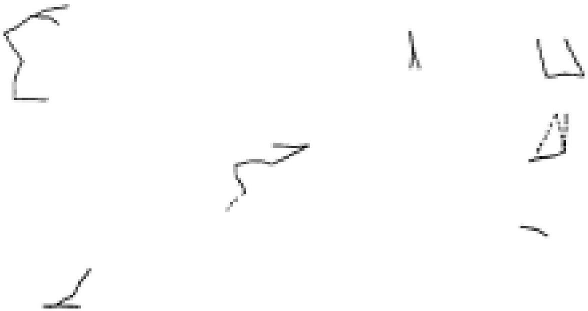

Fig. 12B

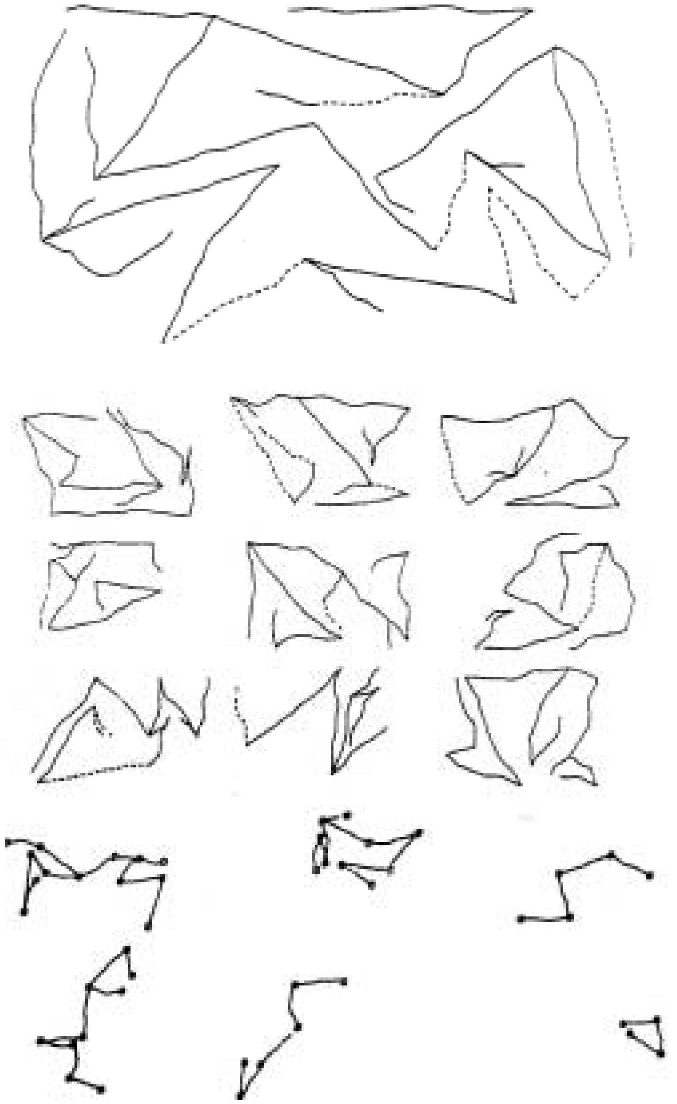

Fig. 12C, 12D, 12E

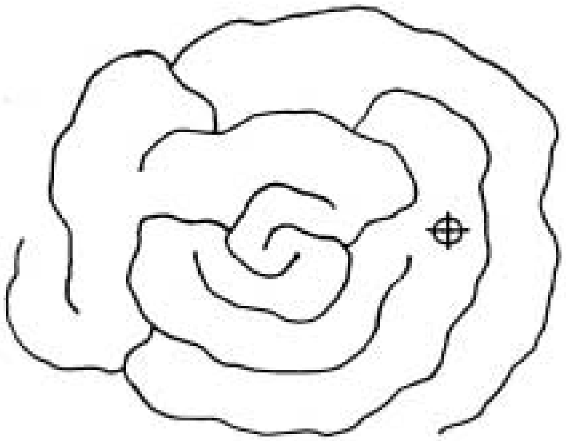

Fig. 13

the picture? Which is in the densest part of the picture, and so on: with even greater variety of output than we have at the moment.

But this brings me now to the central issue in this enquiry. I suggested before that feedback complexity might be measured by the number of questions which needed to be asked about the current state of the drawing, not by how difficult it would be to answer them. Consider the question: is the pen currently inside a closed shape? (Fig. 13). If the lower part of the program had been keeping up a running index of closed shapes, updating it every time a new closure was made, the question might be answered immediately. If not, it might not be able to give an answer at all within any reasonable time. Obviously one would not want to write a program which asked unanswerable questions: or to put the matter the other way around, one would certainly want to be sure that the lower part was in a position to answer the questions one knew the upper part would need to ask; that the required information was either explicitly available, or easily derived from what was available. What may not be equally clear is that as far as the machine is concerned its awareness of the picture exists solely and exclusively in terms of this information, and it is by no means mandatory that this information be visual, in any sense which might seem to apply to human perceptual behavior, and which makes the recognition of closed shapes a trivial human problem. I am not in a position to judge what the relationship actually is in human perception between the outside world and the internal representation : my experience in teaching students to draw suggests that the internal representation of the visual world is certainly not exclusively in visual terms, and indeed that visual information may be a good deal harder to retain than information of other sorts. As far as the machine is concerned, the internal representation similarly need not be equivalent to a complete view of the picture — such as might be given by a television scan of the picture itself, or by a fine matrix in which each cell records the presence or absence of a mark in the drawing — and for many purposes such a representation would yield up the required information very poorly. Actually, they might be better regarded as transcripts than as representations. The need to model the machine's internal representation in terms of the upper program's special preoccupations is not merely a matter of efficiency, since once it is established it places an absolute limit on what the upper program will be allowed to do. There will always be a line beyond which the upper program will not be able to go, questions in answer to which it will be told — sorry, we don't know anything about that.

There is, of course, no mode of internal representation of the work in progress which could be described, meaningfully, as 'natural' to the machine, and no single universal mode to satisfy all possible requirements. Presumably the same could be said of human internal representations of the real world, since we do find it necessary from time to time to build new models, or at least to modify old ones, pushing back the line and finding ways of asking new questions. Whatever else it is, art is primarily a model-making activity. Thus unless we were limiting the aim to simulating the work of a particular artist at a particular time (a human process known as plagiarism, not as art) it would be obviously simplistic to think in terms of the machine's upper program needing to ask the same questions about the work in progress that the human artist would ask, and of the lower program building representations of the work very much like those the human artist would build. The nature of art is not to be characterized in terms of specific sets of questions and representations, since these will be, by definition, in flux for any given artist and even perhaps peculiar to a single artist only. The interface between the questions and the representations, permitting fluid change in both, might reasonably be thought to possess more general properties, since art does change, at least within our own Western tradition. Thus I would conclude that the machine's autonomy rests upon developing total systems, in which the feedback structures linking the decision-making processes above with the characterizations below are sufficiently flexible, or adaptive, to support the changes which must occur in both; not upon pinning down particular characteristics of human perception, or particular formal aspects of human art.