Skills and the Appreciation of Computer Art

Margaret A. Boden [1]

Abstract. The appreciation of art normally includes recognition of the artist's skills in making it. Most people cannot appreciate computer art in that way, because they know little or nothing about coding. Various suggestions are made about how computer artists and/or curators might design and present computer art in such a way as to make the relevant making-skills more intelligible.

1 Introduction

Philosophers of art disagree over the aesthetic relevance of making-skills. Robin Collingwood [6], for instance, thought such skills irrelevant in judging a work as "art". Skill was relevant, he said, only to crafts.

John Ruskin [**?**] didn't go that far. As co-founder (with William Morris) of the Arts and Crafts movement, he didn't take a dismissive attitude towards craftwork in general. Moreover, he compared the skills of different art masters in judging who produced "better" art than others. Nevertheless, he thought skill to be much less important to a work's status as art than the free expression of the individual human spirit. That's why, in his panegyric on "The Gothic", he favoured the roadside carvings of untutored rustics over theadmittedly very beautifulmasterpieces of Classical or Islamic art.

Ordinary people, by contrast, usually think skill to be importanteven essential. (Hence the common dismissal of a work with the complaint: "Call that art? A six-year-old could have painted it!") Their interest, however, is usually less in deciding whether the piece actually is art than in deciding whether it's good, whether they like it. In brief: they're into art appreciation, not art identification.

Computer art, notoriously, raises difficult questions regarding art identification. Can something really be "art" if it was generated by a computer? And, of course, all the familiar identification-questions remain. For example: if a work is exhibited in a gallery, or sold by a recognized art-dealer, does that make it art? But we shan't be concerned with those questions here. Rather, we'll focus on the role of making-skills in influencing the appreciation of computer art. In particular, does unfamiliarity with computers compromise a person's chances of enjoying computer art?

2 The recognition of skill

Setting aside Collingwood's and Ruskin's hesitations, let's grant that for something to be a good piece of art it has to be made in a skilled fashion. How can that skill be recognized?

It's not always easy to judge just what skills are being displayed in a particular artwork, or even in an artist's entire oeuvre. One and the same paint-stroke, for instance, could have resulted from a deliberate intention to suggest emotional abandon, or from carelessness. Consistency across many different paintings might suggest deliberate (and admirable) skill rather than momentary carelessness, but this artist might be careless all the time. And what is to count as "careless"? Whether Jackson Pollock's paint-splashes, for instance, should be so described is debatable. They aren't meticulously fashioned like the paint in a Dutch still life. But they aren't introduced willy-nilly, either.

Viewers will be in a better position to discriminate skilled intent from carelessness if they have some understanding of just how difficult achieving a particular effect actually is, and just how easy it is to fall short. That understanding can come from practical personal experience and/or from impersonal knowledge about artistic techniques.

All of us have wielded a paintbrush, if only at the kitchen table when we were children. So we all know something about how difficult it would be to do that, and how distressingly easy to do this instead. But the more experience we have of such matters, the better we are able to appreciate the skilled achievements of a master. Likewise, someone who knows (from reading art-history, not from art-making experience) that priming a canvas with white enables certain colour-effects to be produced is in a better position to appreciate the challenges faced by the artist, and his/her success in meeting them.

Potters and metal-workers face more difficulty than painters in having their skills recognized. Very few of us have ever glazed a pot, or used a wheel. Even fewer have etched images onto metal plates. We can of course view etchings as drawings, in which case we are on accessible (kitchen-table) ground again. But skills specifically related to working, or working on, metals are foreign to all but a tiny number. Abstract knowledge about the techniquesand difficultiesinvolved is all the more necessary in such cases. Without such knowledge, we're reduced to observing the final result and asking whether it is beautiful, or disturbing, or aesthetically interesting in some other way. Those questions aren't irrelevant, of course. Art appreciation involves them, too. But they have nothing to do with the recognition of skill as such.

In short, both experience and knowledge are normally involved in art appreciation. They enable us to recognize skill, and to compare different degrees of skill, when we see it.

3 Recognizing computer skills

Here, computer art faces a problem. Or rather, much computer art faces a problem.

Computer-assisted art (CA-art), such as David Hockney's Photoshopped images, isn't highly problematic. Many visitors to Hockney's recent exhibition at London's Royal Academy will have played around with Photoshop themselvesincreasingly many of them, as the future unfolds, when they were still children at the kitchen table. And it's as easy to find a friendly advisor, a mini-demonstration, or a written manual, on Photoshop as it is to locate a book on art-

[1] Department of Informatics, University of Sussex, Brighton BN1 9QJ, UK. Email: m.a.boden@sussex.ac.uk

history–and almost as easy to understand. Moreover, Hockney's own goals and intentions are relatively accessible. Even without his self-explanations in the exhibition's catalogue, it's relatively easy for the gallery-visitor to understand what he is up to. For what he is up to has nothing essentially to do with computers. The computers help him to do it, but he was trying to do it anyway.

Computer-generated art (CG-art) is very different. In CG-art, the art-producing computer is left to run by itself, with minimal or zero interference from a human being [2]. Here, what the human artist is trying to do is typically just not do-able without the computer. Moreover, the skills required to do it include computer programming. Having a good eye for colour, or symmetry, or isn't enough. One also needs to be able to code.

And there's the rub. Most people cannot code, not even at the kitchen table. And it's not easy to explain to them how to do so. This may change, if school curricula change (as they should) so as to include coding as a mandatory exercise. Even an elementary experience of programming can be enormously educative, for it shows the extreme clarity and precision that must be observed when writing programs. In that respect, there's more difference between someone who's done only a little programming and someone who's done none at all than there is between someone who's done only a little and a software wizard.

Even if the viewer does have kitchen-table experience and/or some basic knowledge of coding, they will very likely have little or no appreciation (sic) of the difficulties faced by the artist-programmer in writing the CG-art system concerned. For, quite apart from the specific difficulties attending any particular computer-art project, there are various types of CG-artand these present differing challenges to both programmer and audience.

The different genres of CG-art include evolutionary art (Evo-art), interactive art (CI-art), robotic art (R-art), and virtual-reality art (VR-art). Someone might be able to recognize–and appreciate–one of these without being able to recognize all of them.

People who are willing to play with a demonstration, enrol on a computer-programming course, and/or read a book about computer art will make some advance in understanding the specialist skills involved, of course. But what about the casual gallery visitor? They will probably need help to understand what's going on, so as to enrich their appreciation of the artist's achievement.

In short, there's a special burden of responsibility on the shoulders of artists and curators of CG-art. Given the lack of relevant knowledge on the part of their audience, they need to make the general nature of the CG-art program intelligible. Ideally, they need also to convey something about the specific difficulties involved in this particular artwork. But how can these things be done? Short of turning every exhibition catalogue into a textbook, how can artists and curators communicate the nature of CG-art skills to the uninitiated?

4 Clarifying the CG-art genre

The general nature of Evo-art is perhaps the least difficult to convey. For people are already familiar with the concept of evolution. That is, they know that if random changes are made to some structure, and if the most successful of these are selected to be the "parents" of the next generation, thenwith time–the structure is likely to improve.

They even knowalthough in an Evo-art context it could be very helpful to remind them–that what counts as "successful" can vary. Different criteria of success will lead to different kinds of improvement. In biological evolution, running-speed and camouflage (for instance) improve overall fitness in very different ways. What are some of the alternative criteria of "fitness" in CG-art? The colour palette? Symmetry? 3D verisimilitude ?

A significant problem here is that, for most Evo-Art, there is no programmed fitness function that can be communicated (whether in words or code) to the audience. Instead, the selection is made at each generation by the human artist (or, sometimes, by the audience). The basic reason for this is that it's usually very difficult, or even impossible, for the artist to state what his/her aesthetic criteria are. And even if a criterion can be stated in words ("symmetry", perhaps), defining it clearly enough for it to be coded is another matter entirely.

The human artist may not even want to state the fitness function. Suppose, for example, that he/she had discovered that a particular combination of wavelengths was judged to be attractive by most people, and had incorporated it in the fitness function accordingly. It doesn't follow that the artist would want this to be known by all and sundry. It might make his/her aesthetic skills appear to be less intriguingly mysterious, and therefore less impressive, than they would otherwise seem to be. It might even lay them open to plagiarism by other artists.

Of course, someone can know what biological evolution is without having any notion of how a computer program could evolve. Indeed, the very idea may seem absurd to someone unfamiliar with evolutionary AI. So it would be helpful if this could be explained to the gallery visitor. This shouldn't be too difficult. Mini-images, and even lines of code, could be displayed to show different mutations within one generation, and the changes ensuing over several generations. Moreover, if lines of code were shown, then distinct types of mutation (e.g. point mutations, crossovers, code-nestings, and concatenations) could be illustrated. Such a demonstration might make it clear, for instance, that a point mutation is less likely to cause a deep structural change than a code-nesting is. So the viewer might start to get some sense of the skill needed for the artist to choose the types of mutation allowed.

One might expect it to be even easier to show the audience what's going on in CI-art. For after all, the artwork here changes in some way as a result of the actions of the viewer. However, this issue isn't straightforward.

The relevant "actions" may be ones which the viewer doesn't normally think about, or even notice. Breathing, for instance. Or even the emission of body-heat. In addition, the effects of any given action may be unpredictable, or delayed, and accordingly difficult for the audience to recognize. CI-artists themselves differ hugely over whether they actually want their audience to realize what's going on, and whether they want them to be able to control what goes on accordingly. Different types of programming skill are involved in either case, but how this fact could be conveyed to non-programmers is unclear.

The CI-artist's decision as to how much control to hand over to the audience will depend on how strongly they want the audience to think of themselves as participants, or even co-creators, in the production of the artwork. For to cause an effect in the changing artwork isn't the same as to deliberately bring it about. Only the artist-programmer, if he/she wishes, can enable the audience to be fully engaged in the (co-)designing of the work.

Roy Ascott [1] is an example of someone strongly committed to the co-creational perspective in art. So is Ernest Edmonds. But Edmonds' CI-artworks involve both significant delays and non-obvious cause-effect (interaction) rulesboth of which distance them from the viewer's conscious intentions. The sense of control enjoyed by Ascott's co-creators is much greater than that experienced by Edmonds' audience.

Assessing the CI-artist's skills may be tricky. It's much easier to write code to delay viewer-caused changes by (e.g.) 30 seconds than to code just what those changes should be and/or what actions can cause them. Similarly, it's easier for the audience to see–in successive versions of Harold Cohen's AARON, for example (Cohen in [3, 4, 5])–that a CG-artist is gradually becoming more skilled at depicting three-dimensionality than it is for them to appreciate (sic) what sorts of code are being added (what difficulties the CG-artist is facing) as the program develops.

For instance, suppose that a beech-tree appears in a developing VR-artwork whenever the viewer raises their right arm, and that a white bird appears when they stamp their left foot. But how is the computer to know that they've raised their right arm? Is that done by using advanced computer vision, requiring significant programming skills? Or does it depend on a quick fix, enabled by the viewer's wearing a specially designed electronic glove? And what of the tree, and the bird: do they appear in plausibly realistic positions in the VR-image, and if so how is this achieved?

Many such questions could be satisfactorily answered only by considering the code in great detail. Most viewers wouldn't want to do thatand wouldn't profit from it, in any case.

R-art adds further complications. For robots involve engineering as well as programming, and the artist's engineering skills will often be germane in appreciating their art. But to what extent should the artist, and/or the curator, try to inform the audience of such matters? Although much engineering (gears and cogwheels, for instance) is visible, and intuitively intelligible, much is not. If an R-artwork embodies a clever engineering trick, is it worth anyone's while to try to tell the audience about it?

Whatever the particular genre of CG-art may be, it may not be exhibited in a gallery at all. Rather, it may be situated on a street, or other public place, to be enjoyed by passers-by. Here, there's no official curator. Is it anyone's responsibility (the CG-artist's, perhaps?) to convey some sense of the skills required to produce that artwork? If so, would that risk turning every plinth into a textbook?

5 Appreciation without recognition

If art appreciation normally involves recognition of (some of) the skill involved in producing it, it doesn't follow that appreciation must be wholly lacking in the absence of such recognition. For superficial aspects of artworks can be enjoyed without any consideration of how they were made.

Judgments on whether the work is aesthetically arresting in any number of ways can be made even by someone knowing nothing of its history as an artefact. So someone ignorant of programming, and even unfamiliar with computers, can look at CG-art and appreciate it in those terms.

Nevertheless, they'll be missing out. The element of art appreciation that depends on the recognition of art-making skills on the artist's part isn't available to them. One of the challenges facing CG-art curators (and dealers), and CG-artists too, is to design and/or present this work so that the general public isn't short-changed in that way.

REFERENCES

[1] Roy Ascott, 'Behaviourist art and cybernetic vision', *Cybernetica*, **9**(4), 247, (1966).

[2] Margaret A Boden and Ernest A Edmonds, 'What is generative art?', in *Creativity and art: three roads to surprise*, 125–161, Oxford: Oxford University Press, (2010).

[3] Harold Cohen, *On the Modelling of Creative Behavior*, volume 6681, Rand Corporation, 1981.

[4] Harold Cohen, 'The further exploits of aaron, painter', in *Constructions of the Mind: Artificial Intelligence and the Humanities, special edn.*, eds., S Franchi and G Guzeldere, volume 4, 141–60, Stanford Humanities Review, (1995).

[5] Harold Cohen, 'A million millennial medicis', in *Explorations in art and technology*, eds., L Candy and E Edmonds, 91–104, Springer, (2002).

[6] Robin George Collingwood, *The Principles of Art*, Oxford: Clarendon Press, 1938.

[7] John Ruskin, 'On the nature of gothic architecture: And herein of the true functions of the workman in art', in *The Stones of Venice*, volume 2, chap. 6, London: Smith Elder, (1953). Published separately, with an introduction by William Morris.