THE ROLE OF DRAWING/VISUALIZATION IN THE DIGITAL DESIGN PROCESS

Ewan Steel

ABSTRACT

This paper examines the role of drawing/visualization in the digital design process, focusing on the part it has to play in enabling creative group thinking in the development of interactive multimedia. In the context of this discussion, interactive multimedia is defined as digital media that intends the user to have an aesthetic experience in interaction. The design of interactive multimedia is a synthesis between programming and digital design/aesthetics; this paper seeks to explore the cognitive processes that generate and interpret drawings that are part of the development of this synthesis. This paper will examine and analyze the differences between the creative/cognitive thinking of visual designers and programmers. It seeks to establish a common creative approach to the design of interactive multimedia and make this approach concrete as visual symbol. This paper will make a contribution to establishing a paradigm of digital design practice and education that uses experiments and exercises in drawing/visualization as a fundamental creative activity and means of communication.

Keywords: Drawing, programming, cognitive process

1. INTRODUCTION

The development of aesthetics led interactive multimedia can be a process without defined goals. These applications are judged on their inherent ‘playfulness’, and the quality of user-interaction with visual/sound content is deemed the most significant design factor. In effect, an outcome can be an exploration of intuitive design decisions and the result of a shared creative experience within a design team.

In such a fluid environment, good communication between designer and programmer and a shared aesthetic are essential in the development process. Shared experiences in drawing/visualization are important in the development of this relationship, both as an educational tool and as part of a practice methodology.

Drawing/visualization has always been seen as central to the development of ideas in design. What form this drawing takes can vary greatly dependent on individual skills and preferences, shared experience in working groups and different traditions in professional practice. Most professions use a form of drawing where utility is dependent on shared experience of a specialized visual language. It functions as a highly formalized depiction of desired reality. Architectural and engineering plans use a graphic language that is easily understood by fellow professionals and related disciplines. They are coded messages that drive productivity. This is a limitation as observed by Massironi,

"A code can work when it is shared by emitter and receiver, but to be shared it needs structure and rules. Rules make communication possible, but they also reduce the possibilities of adaptation to reality." [1]
These kinds of codified drawings represent the design process at a stage once removed from direct observation and pure conceptualization. The drawing provides a visual environment in which creative insights are discussed and developed, but it is less likely that these insights were inspired by the actual 'act' of drawing. The reality of the proposed structure becomes clearer, and as a result, is more open to creative intervention, but the drawing process is too mechanized and formalized to provide an organic experience similar to that of the artist or child engaged in concentrated freehand drawing where evolving experience and understanding of the object perceived or conceived is affected by the actual drawing process. However, any educational exercise or practice methodology that uses drawing as a means of communication has to embed an agreed upon structure to the drawing process, rules that govern what is to be observed and developed at the outset. The style of drawing chosen to express observations and development ideas should be a matter of individual choice. Individual visual expression can be used as a focus for group discourse. In traditional art school pedagogy, group discussion of individual stylistic approach is used to help students develop their critical faculties and visual logic.

2. DRAWING AND PERCEPTION

Recording (or inventing) complex visual data through drawing 'forces' us to develop organizational patterns of thought that simultaneously control observation and depiction. These patterns are instrumental in the creation of artistic style. The interplay between underlying logic and observation is refined within an overarching and consistent aesthetic approach, guided by experience and expressive response, and manifest as personal style. The drawing is a trace of a human decision making process that combines logic and expressiveness. Real creative drawing is a mixture of aesthetic experience and a developing understanding of form and function. Aesthetic response is stimulated by our mind finding a recognizable sequence or pattern in the visual world, the very building blocks of perception. The psychologist Steven Pinker has observed,
"First we seem to get pleasure out of looking at purified, concentrated versions of the geometric patterns that in dilute form give us pips of micro-satisfaction as we orient ourselves toward informative environments and fine-tune our vision to give us a clear picture of them." [2]
Drawing mimics our actual experience of perception, which uses the motor of visual aesthetics to forward understanding. The plotted line in an architectural plan develops this understanding without the immediate aesthetic feedback of seeing the mark as part of an organic whole, both in terms of aesthetics and cognition. This positive feedback is seen as an essential part of any drawing experience that would be of value to those working at the design/programming interface.

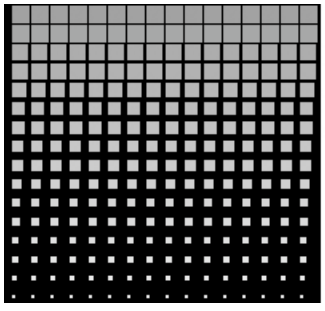
3. AESTHETICS

If we look beyond formalized aesthetics, primarily a manifestation of 18th century attitudes to fine art, a visual aesthetic can be seen as an emotional response to the systematic interpretation of visual data. The artist/programmer Harold Cohen has observed,

"... I regard "style" - surely the most difficult word in the entire vocabulary of art - as the signature of a complex system ... an emergent property arising from the interaction of so many interdependent processes." [3]
Cohen's interpretation of 'style' suggests that we can experience an aesthetic response to any complex system. The artist/designer takes pleasure in arrangements of concrete visual data; the programmer derives the same sense of satisfaction in recognizing patterns in the organization of abstract data. For designer and programmer to experience a shared aesthetic there must exist some understanding of what constitutes 'style' in their mutual disciplines. It is a difficult task to communicate aesthetic response, essentially intuitive/emotional and not readily quantifiable. But in any educational environment, group experience and discussion can create areas of commonality and enhance learning. The focus and structure of exercises that promote dialogue and mutual understanding/appreciation at the interface between visual and programming aesthetics will be discussed in the following sections.

4. VISUALIZATION IN PROGRAMMING EDUCATION

Computer based program visualizations are used to help student programmers understand concepts related to procedural and object oriented programming. Visual content can represent data values and relationships; the behavior of algorithms plotted in algorithm animations; the abstraction inherent in object-oriented concepts can be expressed as visual constructs where color, shape and relative placement signify object dynamics. Visualizations can be used to help students understand both programming languages and program constructs. In a design team engaged in the development of a creative application, mutual understanding between designer and programmer of the program construct is essential. Language syntax does not contribute significantly to creative synthesis.
Naps et al. have stated "... visualization technology, no matter how well it is designed, is of little educational value unless it engages learners in an active learning activity."[4]
A widely used programming/visualization tool that encourages this kind of engagement is 'Processing' (http://processing.org/). This open source programming language and environment is being utilized by a growing number of educational institutions and individuals. Its focus on user participation in the programming of visual content (computer generated drawing) has made it a meeting place for artists/designers and programmers and a model for drawing exercises in this field. The drawings (fig 1) are visual traces of code structures.



```
float box_size = 11;
float box_space = 12;
int margin = 7;

size(200, 200);
background(0);
noStroke();

// Draw gray boxes

for(int i = margin; i < width-margin; i += box_space) {
  for(int j = margin; j < height-margin; j += box_space) {
    fill(255 - box_size*10);
    rect(j, i, box_size, box_size);
  }
  box_size = box_size - 0.6;
}
```

Figure 1. Embedding Iteration in Processor

'Processing' is more than a mere educational tool, aimed at developing programming skills. These 'software drawings' possess an inherent aesthetic and successfully make a connection between visual and programming logic. In most cases, however, they tend to generate an over-formalized graphic-style (an emergent manifestation of code driven process) that lacks individual expression and spontaneity. Freehand drawing can provide a parallel experience, but one that is more intuitive.

5. DRAWING AND PROGRAMMING

In art and design, drawing has traditionally been used to explore and rehearse the possible interactions between artist and concrete object, placing the artist in a spatial and temporal relationship to methods and objects that exist in the concrete world. The focus of this paper is to use freehand drawing as a medium to explore creativity that interacts with virtual objects, manifestations of programming concepts. Visual logic, if not exactly analogous to programming logic, is close in many respects. We can describe the drawing process using a language that is consistent with basic programming concepts. Programming structures appear to have the inherent qualities of a complex drawing developed through a controlled combination of iterative mark making. Programmed objects can be seen as the constituent parts of a drawing, repeated marks or shapes contained within or controlled by parent shapes. The parent-child relationship is key to controlling the mark-making process in a complex drawing. If a drawing is seen to possess internal harmony, its detailed observations are nested inside larger, progressively abstracted shapes that dictate overall compositional movement in both the creation and perception of the drawing. The parent shapes must interact within a consistent visual logic; this is analogous to artistic style. Nested objects/shapes/marks are realized in repetitive loops that build complexity, following their own internal logic while interacting with other objects in a consistent fashion that follows pre-planned objectives. Stylistic approach is inherited from the initially created classes. Inheritance is deeply rooted in the drawing process; the first marks on the paper can influence the stylistic approach of the whole drawing process where each new mark contains an echo of a previous mark, geometric shapes repeat, becoming smaller as the drawing becomes more resolved towards an acceptable reality (fig 2).

Figure 2. Progressive stages of drawing

Freehand drawing, however, is an essentially linear process where inheritance cannot be used as a method of creating flexibility in the properties of visual data. This is a very important issue, which highlights a significant difference in thinking between visual designers and programmers. Object inheritance is a difficult concept for the artist/designer to grasp. He tends to see the world as being constructed in a way consistent with a conception of art and design that is rooted in classical notions of concrete reality, and is continually imposing this unidirectional perception of a concrete

world on the computer environment. Design decisions are made consistent with this perception. As inheritance is central to programming flexibility and shared visual design/programming creativity, experiments in visualizing inheritance are seen as an essential element of any group drawing exercise.

6. CONCLUSIONS

There can be no definitive structure to drawing exercises that enhance creative group thinking in this area. Drawing strategies will be determined by group dynamics. All the members of the design team should agree on drawing strategies. All should experience the drawing process. It is necessary that the act of drawing be as simple as possible so that no member of the group feels isolated. It is essential that the act of drawing be seen as an experience and drawing processes and outcomes a focus for group discussion (not to be judged as independent works of art).

The following is an outline of the goals of such a drawing program...

1. All investigation has its origin in direct observational drawing, however simple, and subsequent development drawings are abstracted interpretations of initial observations.
2. The drawing process should develop understanding of programmed environments.
3. The drawing process should encourage aesthetic responses to OOPS structures.
4. Development drawings should attempt to synthesize visual and programming logic.
5. Group discussion should facilitate shared understanding of stylistic approach.

We can adapt established design drawing traditions of selection and development from initial object drawing where identification and analysis of iterative form is used to create the visual properties of distinct 'classes' and growth/organizational patterns used to mimic 'methods'. Development drawings can be directed to create visual metaphors for parent-child relationships and inheritance through the promotion of conscious mixing of forms to create new 'classes'. The development drawings are in essence object hierarchy maps that focus on the aesthetic potential of analyzing the drawn image in a manner consistent with a programming structure.

As has been outlined, the concept of object inheritance is a crucial area for visual experimentation and discussion. In addition to basic drawing exercises, program visualization tools such as "Processing" can be used to extend experience. These tools are better suited to visualize advanced programming concepts such as inheritance and polymorphism.

The object of any drawing exercise is not to build a workable programming structure or a drawing that has artistic merit. Drawing is seen as a shared experience within a community of designers and programmers, a shared experience that encapsulates the creative style of any possible joint creative project.

In today's multimedia design environment, a general style that encompasses both programming and visual design is seen as an important development in this field. The two disciplines are increasingly seen as interdependent and it is essential that creative drawing is part of an evolving educational experience and professional practice.

REFERENCES

[1] Massironi, M., *The Psychology of Graphic Images,* Lawrence Erlbaum Associates, New Jersey, 2002, p.71.

[2] Pinker, S., *How the mind works,* Penguin Books, 1997, p.527.

[3] Cohen, H., How to draw three people in a Botanical Garden, *AAAI*, Vol. 89, 1988, pp. 846-855.

[4] Naps, T.L., Rößling, G., Almstrum, V., Dann, W., Fleischer, R., Hundhausen, C.,

Korhonen, A., Malmi, L., McNally, M., Rodger, S. & Velázquez-Iturbide, J.Á Exploring the role of visualization and engagement in computer science education, *SIGCSE Bulletin*, 35(2), pp. 131-152.

Ewan Steel BA(Hons)
Graphic Design department
Duncan of Jordanstone College of Art & Design
University of Dundee
Dundee DD1 4HT
United Kingdom
T: (UK) 01382 345290
F: (UK) 01382 201378