25
Irises in a Landscape: An Experiment in Dynamic Interaction and Teaching Design Studio

John W. Danahy

Centre for Landscape Research
School of Architecture and Landscape Architecture
University of Toronto

The capacity of most computer-aided design systems is inadequate to represent landscape architectural ideas and compute landscape scenes quickly. As part of our teaching agenda, we decided to write software for the Silicon Graphics Iris workstations to tackle this problem directly. This paper begins with a discussion of our concerns about the use of CAD tools in the representation of landscape architectural space. Secondly, we discuss the approach that Toronto takes to computing and teaching with particular emphasis on the use of computers to support an integrated representational work environment. Finally, a fourth-year design studio that used our software is reviewed. Static illustrations of the system are presented here, although there is a videotape that demonstrates the dynamic nature of the system.

Landscape Architectural Representation

Concerns about the deficiencies of commercial CAD and solid-modeling systems for three-dimensional architectural representation and design have been discussed in the literature.[1] Significant research into much more subtle, semantically correct approaches to operating on architectural models has been started at a number of architecture schools, such as Ohio State, SUNY Buffalo, Harvard, UCLA, Carnegie Mellon, and Michigan. But no such first tier of research has been devoted to the issues of modeling *landscape* architectural form. The effort reported here is to build tools and methods that are relevant to the issues raised in landscape architectural studios. The first issue of concern to us is the representation of landscape form and space.

Landscape architects make extensive use of traditional geometric conventions for the conceptualization of built form and can benefit from advances in architectural modeling. But geometric form is only one of at least three major categories of space and form that a landscape architect must be concerned with to have an adequate palette to represent a landscape. An elementary palette that we have found most useful is one consisting of terrain, built form, and vegetation. A useful system must support at least these three basic types of form.[2]

Most architectural modeling systems are useful for the construction of building form. But there are few useful tools for the modeling of terrain and vegetation at a variety of levels of abstraction or detail. Available tools can seldom respond efficiently and flexibly to issues of landscape design such as vegetation growth. Our experience in building models with full terrain and species-specific vegetative representations shows a three to ten fold increase in crafting time over that required to produce an 'equivalent' model assuming a flat empty site.

Another serious issue in the practical use of existing CAD systems arises when one tries to model the complexity of landscape scenes. The computational and concomitant data storage problems that develop when representing landscapes overwhelm systems that are adequate for the representation of buildings at equivalent levels of abstraction. Images of landscapes that yield significantly more information than manually crafted sketches take far too long to generate on most microcomputers to be useful. The microcomputers capable of providing fast three-dimensional architectural information useful in the design process become unacceptably slow if image complexity is pushed to the levels that we have found necessary in professional landscape architectural work.[3] Dynamic feedback and the opportunity to cycle through iteratations in design development are all but lost under such conditions.

Toronto's Approach to Computing and Teaching

We place a strong emphasis on visual representation and simulation media in the Landscape Architectural Programme and the Centre for Landscape Research. We are very interested in the development of tools and strategies to improve students' and professionals' visual thinking and decision-making skills. We have found that the computer makes complex spatial configurations more visually accessible through an array of sophisticated visualization tools available to us for several years through our links with the Dynamic Graphics Project of the Computer Systems Research Institute at the University of Toronto. These general-purpose representation tools have been used by us on a series of professional projects and studios in which we focused our creative attention on the definition of adequate design vocabulary and procedure. That experience has served as the basis for our recent efforts in software development.[4]

We try to teach students to examine issues of information organization, classification, interrelation, and substitution. For now, most of this must be handled in the brain of the computer operator and, for the most part, our use of computers has been task oriented. Until very recently, sophisticated computer rendering tools have been extremely primitive from the standpoint of knowledge structure and dynamic interaction with design information. As a result, the onus for the organization and creation of information has been the responsibility of the user.

Evolution of Our Computing Tools

The first studio we attempted (four years ago) used terminals connected to a Vax computer over telephone lines. We have used a variety of tools for creating polygon models of landscape. AutoCAD has been extensively as a digitizing and polygon-modeling tool. This has been accomplished with a completely customized AutoLISP interface. The process of constructing models this way is laborious and demands of the user a massive commitment to maintain a cognitive map of the intended shape or space while operating the tools.

A range of traditional image-processing and rendering software is currently available to students on Irises, Macs, and IBM-ATs; however, the students prefer to focus their use on two new programs under development in the research lab. These programs are referred to as TRIM (Toolkit for Representation & Modeling) and Polyed (a polygon modeler with some specialized functions for landscape architects not found in traditional modeling packages).[5]

In the spring of 1989, the Landscape Architecture Programme had a unique opportunity to run a studio using our new software, specifically designed for use by a landscape architect working on an IRIS computer. The eight students in the fourth year were able to use four Silicon Graphics Iris computers, three Mac II micro-computers, and an IBM-AT running AutoCAD with a Targa framebuffer and an E size digitizer. The Macs were connected to the Irises on 19.2 baud serial communication lines. The Irises and the AT were connected by Ethernet.

The software utilized was TRIM. This is a software package being written at the Centre for Landscape Research as a prototypical work environment for environmental designers. The specific orientation of the system is toward the support of work on urban design and landscape architectural problems.

The title, TRIM, stands for the four major conceptual areas of design process that we support.

Toolkit: a general means of operating on and communicating information
Representation: a means of articulating ideas
Interaction: a means of instructing and manipulating data representations
Modelling: a means of creating a representation of an idea

TRIM allows students to load polygon files of site models and move about the models as if walking or flying through the model in perspective (see figures 1 and 3). The Irises are capable of generating shaded perspectives quickly enough to permit a freewheeling approach to the exploration of the data. TRIM also permits the user to pick a parametric object or file and perform cloning, scaling, translation, and rotation operations by opening up a dialog box for the object. Parametric objects also permit one to undertake a variety of parametric operations related to shape, costing, and density calculations for buildings. (Figure 3 shows a tower block being illustrated using these functions).

Polyed (figure 2) has been developed as an interim replacement for the polygon modeling tools we have been using for some time on AutoCAD, and to provide parametric assembly programs in UNIX using the Makefile facility. The previous studios we ran relied on an extensively customized version of AutoCAD using AutoLISP to craft and digitize three-dimensional polygon models of buildings, terrain, and vegetation. The resulting DXF files were transferred to the Irises, where a C program scanned the DXF files and converted them into our laboratory's ASCII polygon format. (Figures 14, 15, and 16 are examples of a model produced this way.)

For the first time, in this studio, students were able to freely move about an arbitrarily-defined polygon data base, with a shaded-lighting model running in real time. Previously, on the older Iris 3000 series machine, they could examine a model in wireframe (with no hidden surfaces) in real time. If they wanted to test the model with hidden surfaces or lighting, they had to run one of our conventional cpu-based, general-purpose rendering packages. This process would require a minimum of several minutes to invoke. The system used in this studio permits students to dynamically manipulate the geometry or placement of objects while viewing the model in perspective with a shaded-lighting model. They can do more work directly on the computer, without resorting to sketching at every turn in order to adequately represent and understand a design idea.

We are interested in studying the degree to which the real-time shaded graphics and ten million instructions per second computing speed of the 4D-20 and 70/GT Iris computers can adequately deal with the complexity of landscape representations.

In our CAD teaching, one issue is how to make the best use of existing computer-aided design tools through the careful organization and management of design knowledge. In our case it is done the traditional way-in the mind of the student. For the most part, organizing and experimenting with the classification, structure, and hierarchy of the information in the machine is the users responsibility. We believe that it is premature to try to use highly structured knowledge base systems on complex problems in the landscape architectural design studio.

We foresee the use of didactic systems for introducing new areas of knowledge to students so that they can learn very specific lessons. The computer would be used to eliminate 'wasted' time in crafting models and to direct the students' intellectual attention to the specific principle or theory behind the exercise. We use this approach to introduce students to the fundamental spatial concepts and semantics of urban design with our TUMMS software on the Iris computers (see figure 4)6 However, at the advanced levels of studio teaching we believe that we must emphasize the development of experience with the formulation of personal design- thinking methods.

To support this notion, we believe the computing environment should be configured by the user to suit his or her interpretation of issues and of subsequent relevance of specific forms of information representation. Therefore, our focus is on the development and use of computers to support intuitively driven and structured design processes. The didactic nature of a future system could result from the user's knowledge, the professor's, or some normative professional knowledge base such as those typically associated with traditional models of expert systems. In the meantime, until good theory and organizational methods are available in landscape architecture, we are not making extensive use of sophisticated knowledge base software.

Using the Computer in a Fourth-Year Studio

Course Objectives

Our specific educational objectives in the studio were:

-to understand the importance of perceived details, dimension, and time in the realization of design concepts;
-to develop better visual analysis and design skills through the use of computers and traditional visual design media;
-to learn the appropriate use of the computer as a means of testing the translation of two-dimensional ideas into three-dimensional form and experience in time.

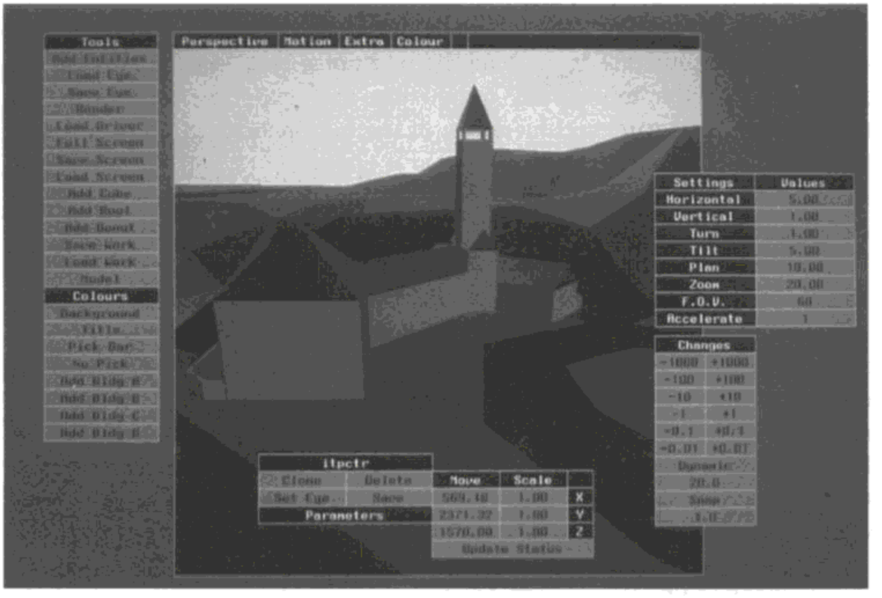

Figure 1 TRIM

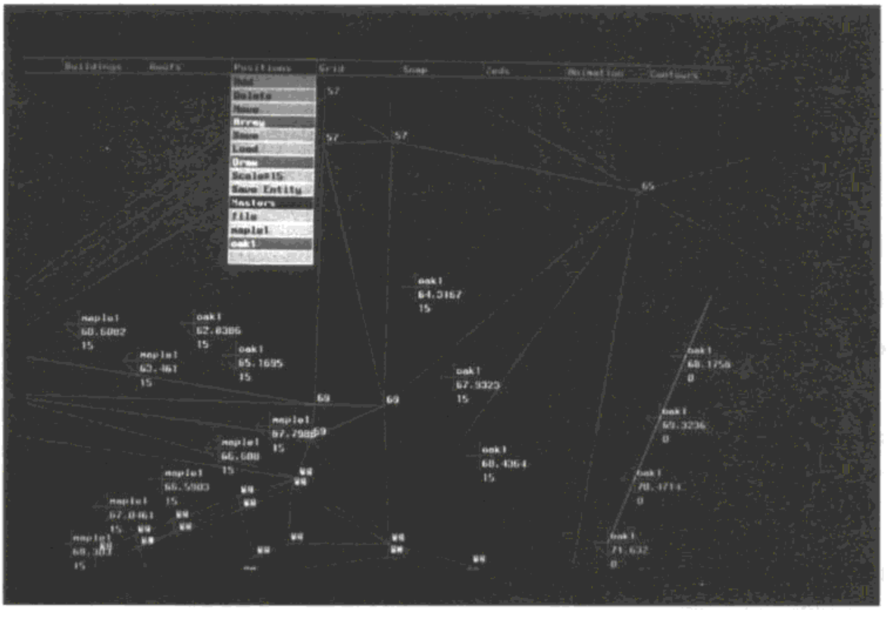

Figure 2 Polyed modeler

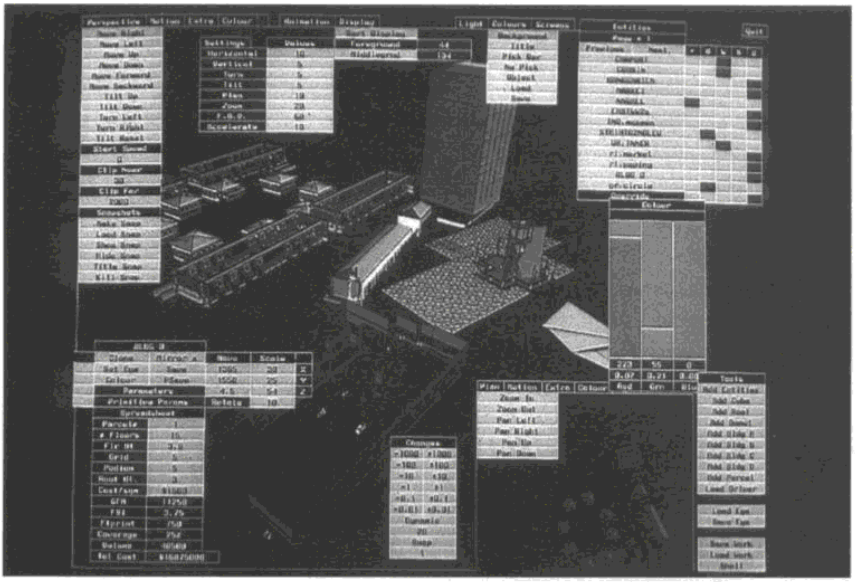

Figure 3 Parametric modeling

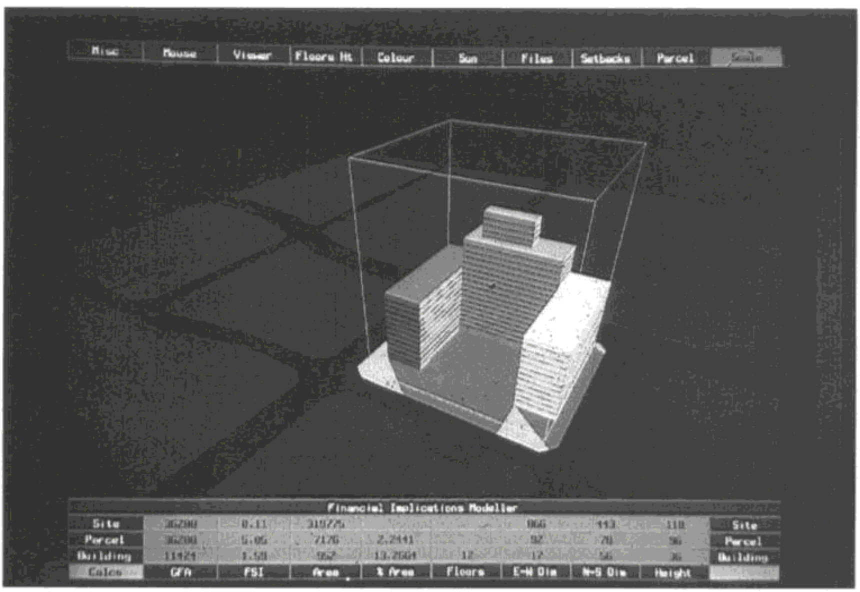

Figure 4 TUUMS modeler

In this studio the computer was introduced to students in a way that did not require them to understand a programming language. It was expected that they would know how to operate a text-editing program and a 'paint' program and that they could undertake simple system operations on the Macintosh. Exercises began with operations that took no more than a couple of pages and a fifteen- to twenty-minute tutorial to describe. These exercises were used to gradually introduce students to the other equipment and software available in the course.

A great deal of the preparation and drafting of base material that is normally associated with a studio project was eliminated from these exercises. The base mapping and three-dimensional computer model of the sites under study were prepared ahead of time and were available on the computers' data library. A preliminary symbol library of urban design and landscape elements, such as tree symbols, was also ready for use. Students were able to start on the Iris computers the first day and test some preliminary design ideas.

The Studio Projects

Part 1: Translation and Representation

The first two exercises of the studio were intended to be used as an introduction to the computer tools and as a look at normative visual design composition issues. The computer provided students with the opportunity to work with perspective composition from the standpoint of perceived eye-level experience of the user. The goal was for the student to gain a better intuitive understanding of the translation of two-dimensional symbolic representation of spatial ideas into perceived form. These two exercises ran concurrently during the first two weeks of the term. Students worked in groups of two on both exercises.

Exercise 1: Exploring Archetypal Space and Learning the CAD Tools.

In this exercise students were asked to go through the assigned readings and "experience" or view the diagrams from the course readings that were on the Iris computers. They viewed thee archetypal spatial arrangements using the TRIM program. The assignment was to generate three unique, interpretive views of a series of diagrams (contained in the course readings) on the Macintosh computer. The images explored were submitted as diagrams in the text of a Word document that explained what each image portrayed (see figure 5).

Exercise 2: Framing the Pavilion in the Landscape.

In this exercise, students were instructed to interpret the landscape of the Parliamentary Precinct in Ottawa from the standpoint of the "roving eye." They were to find a path in space that created an interpretive trail through the Capital. In this first stage of the exercise, they identified and composed three eye-level views along the trail. The Iris computers were used to traverse the site and locate the three vantage points. (Figure 6 is an example of the Parliamentary Precinct data base with which students worked). The students then generated and saved the images on the Macintosh computers using a paint program. The paint program was then used to develop and present a set of potential viewing masks or frames to visually structure or enhance each scene. The images were submitted as diagrams in the text of a Word document that explained what each image portrayed. Stages of the second exercise were:

1. composition of a landscape scene;
2. enhancement of a scene with a two-dimensional frame;
3. translation of a two-dimensional frame into three-dimensional form;
4. interpretation of the Precinct with a moving eye.

Part II: Major Project -Community Design

The major project in this computer-aided design studio spanned eight weeks and dealt with concepts of neighborhood open space and housing design. A variety of approaches to design typology were explored. The typological approach provided students with a conceptual framework that they applied using the forms of computer representation and modeling tools available to the studio. Students evaluated a variety of typological strategies for the formation of neighborhood open space. The design strategies were examined using a range of forms of representation to develop, test, and communicate understanding of the typologies proposed. Projects were evaluated from both the standpoint of a user at eye level and as abstract relationships in a scheme.

The students were presented with several authors' ideas about typology. The authors obviously held contradictory views of the "correct" way to use typology, which each student had to resolve and incorporate into his or her design thinking. The goal was to have students develop their own design vocabulary. In this studio they were asked to organize their definitions of urban landscape space into patterns and units that facilitate design, analysis, implementation, maintenance, and behavioral use. The units they define form a typology that was to respond to the requirements of each individual design program. Stages of the major project were:

1. Site analysis
2. Search for housing precedents
3. Sketch problem of College Street infill project
4. Formation of a computable working design vocabulary for community design (Reading week)

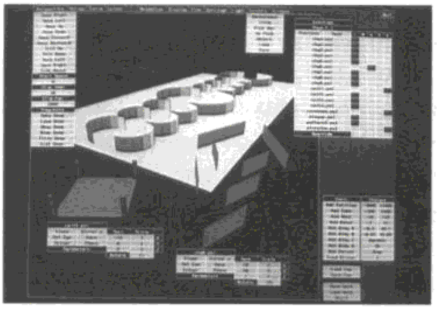

Figure 5 Experiencing a diagram from class readings

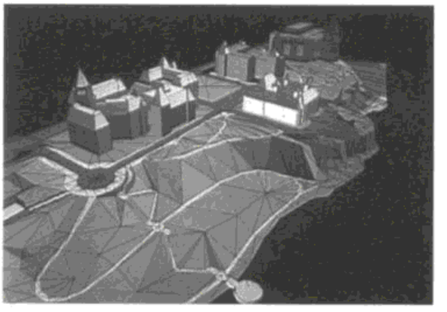

Figure 6 Parliamentary precinct: data given to students

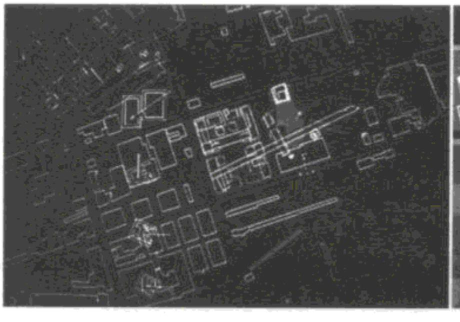

Figure 7 Student Project

Figure 8 Student Project

Figure 9 Student project

Figure 10 Student project

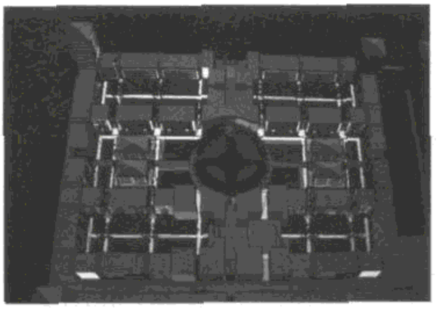

Figure 11 Student project

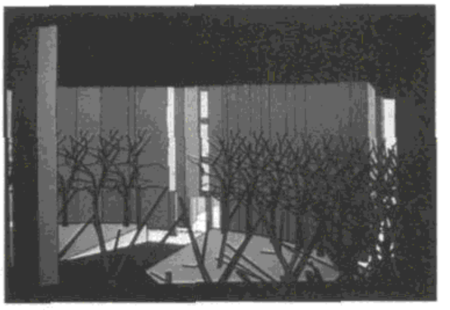

Figure 12 Student project

Figure 13 Student project

Figure 14 Student project

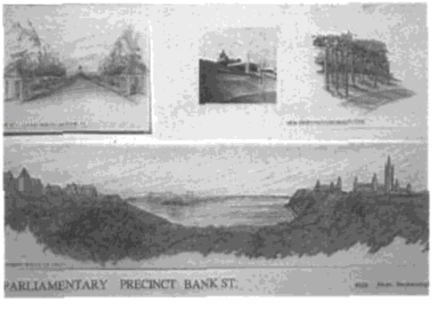

Figure 15 Student project

Figure 16 Student project

5. Morphology of neighborhood: community open-space structure proposals
6. Housing project: program and space-planning concept specification
7. Preliminary design development
8. Design development
9. Revisions and documentation
10. Final presentations

Documentation/Submission

Submissions for the project were made digitally, on video, and on paper. Presentations in studio (both formal and desk critiques) took place directly on the Iris computers. Also required is a videotape of three to five minutes in length that overviewed the project. The paper submitted for the major project consisted of an 11 by 17 inch newspaper-like report given to all critics. All text and titles were to be computer-generated fonts. The document was to be developed and presented at each formal stage of review and revised for the final submission. (Selected images generated by students for the major project appear in figures 7-16.)

Conclusions: Realization of Intentions for the Studio

Preliminary results indicate that a number of our intentions for the course are being realized and that some unforeseen issues are arising. In the Iris-based studio all of the students were able to generate more complex models and more imagery during the first three to four weeks of the term than most of the previous years studio members were able to accomplish during the whole term. We saw students test a greater number of ideas in more depth. Instead of taking hours and days to test schemes, the students actually responded to criticism during studio critiques. For the first time, we saw students model and test ideas about terrain and detailed landscape elements, but we are not yet satisfied with the sophistication of these representations. We need a more careful presentation of methodological strategies to the students and further development of efficient software.

A great surprise to us has been the desire on the part of many of the students to work almost exclusively on the computer to develop conceptual ideas and test them. (In past studios we generated two or three times as much manual drawing in perspective as a traditional studio.) We are not yet sure if this is a positive development. However, a number of the students are beginning to design primarily in perspective.

Acknowledgments

The author would like to acknowledge the contributions of Shannon McKenzie, Rodney Hoinkes, Steven Ginsberg, and Robert Wright to the development and operation of our system. We would also like to thank the Dynamic Graphics Project of the Computer Systems Research Institute for their assistance in the development of software used in these projects. Student work illustrated was produced by Eckhard Pastrik, Ann Cheung, Dennis Lago, Werner Schwar, and Sham Nankoosingh.

Major funding for the activities of the lab has been provided by Alias Research Inc., Silicon Graphics, Inc., the University Research Incentive Fund of the Province of Ontario, The Natural Sciences and Engineering Research Council of Canada, Apple Computer, Inc., and Autodesk, Inc. Databases used by the students have been created in professional association with the National Capital Commission, Ottawa, the City of Ottawa, and the City of Toronto.

Notes

1. Chris Yessios. 1986."What Has Yet To Be CAD. *ACADIA Proceedings '86*.. Houston.

2. Mark Lindhult. The Road Beyond CAD. *Landscape Architecture* (Magazine of the American Society of Landscape Architects) (July-August 1988).

3. John Danahy. 1988. 'Engaging Intuitive Visual Thinking in Urban Design Modelling: A Real-Time Hypothesis." *ACADIA Proceedings 88* . *Ann* Arbor.

4. Detailed descriptions of the techniques. projects. and impact on decision-making of our previous work can be found in the following papers:

John Danahy. 1987. "Sophisticated Image Rendering in Environmental Design Review. Human Factors in Computing Systems and Graphics Interface." pp. 211-218. CHI + GI Conference Proceedings.

John Danahy. 1987. "The Parliamentary Precinct Study: Visual Simulation in Urban Design Decision-Making and The Need for GIS-CAD Linkages." GIS' *87* (San Francisco Second Annual International Conference Exhibits and Workshops on Geographic Information Systems). Post Conference Proceedings. Volume 111.

5. We tested several Macintosh 3D-CAD programs but found no way to extract their three-dimensional databases for use on the Irises. These programs appeared extremely efficient in crafting conventional hardline geometric form. However, they were extremely limited in their capacity to represent terrain and vegetation in a manner acceptable for an advanced landscape architectural studio. One could not adequately shape and interactively change the form of exterior spaces using these packages. Finally, databases that provide any significant level of detail become laboriously slow on Mac IIs or 386 machines, let alone the IBM-AT machine we had been using. We also discovered that a fully configured 386 PC or Mac II with advanced graphics and software was costing us in the neighborhood of $15,000 to $20,000 (CDN). That is the price of Silicon Graphics Personal Irises. As a result, we decided to write our own modeling software and use Silicon Graphics computers.

6. John Danahy, 'Engaging Intuitive Visual Thinking in Urban Design Modelling." Unpublished paper.