CAAD as a Thinking Medium: A Strategy for Computer Aided Architectural Design Education

Murali Paranandi

This paper is organized into two parts. The first part is a commentary on the state of CAAD in response to the three very important pedagogical issues raised by Klerker in the conference theme narrative. The second part presents an introductory course developed and taught by the author that is based on a very simple premise; until intelligent CAAD (Computer Aided Architectural Design) programs become available, use what is available intelligently. The pedagogical model employed is explained. Some of the work done by the students of this course is included. The impact of this course on the curriculum in the school, the marketability of our students, and our future plans are summarized in the concluding remarks.

Part 1: Some reflections on CAAD

CAAD has gone through the following growing pains over the past three decades:

Pedagogical aberration:
Most architecture schools, particularly in the US, over past three decades have developed and taught courses in computing on the sidelines with emphasis on either tool using or tool building; former emphasized vocational training in using popular software for generation of imagery and the latter programming computers to perform design tasks. Regardless, both of these approaches emphasized the tool over the product (architecture). Notwithstanding the groundbreaking work that has resulted due to these efforts, very little has been done to create better architecture.

Profession's retrogression:
Architecture profession's total indulgence with computer usage for productivity gains (mostly drafting and database applications such as cost estimation, facilities management etc.) left no motivation for the profit-minded vendors to develop creative computer applications. The productivity tools have reached a state of saturation based on their current paradigms (which derive from manual drafting methods).

Academic disjunction:
The senior faculty forming the critical mass of architecture schools frequently oppose CAAD. Those interested in promoting CAAD frequently feel their energy is better spent in research than fighting for curricular development (this could be one of the reasons for the activity on the sidelines pointed above).

Emerging Trends:
It is promising to note that the above tendencies seem to the changing with the advent of the following developments:

Increased access
In the previous decades, due to the high costs, computers were the game of the elite (very large AEC companies). Computer time was considered too valuable to experiment with; it had to be put to use in proven ways that guaranteed the returns. CAD technology has been democratized due to recent developments in the computer industry, and has become accessible for even small offices on a budget which are eager to use them in creative ways to compete with the bigger offices.

Paradigm shift
A number of engineering disciplines have embraced 3D CAD and are streamlining the design development/production processes to improve the product efficiency as well as lowering the costs. Prominent architects like Frank Gehry have successfully borrowed some of these engineering design paradigms and demonstrated their suitability for architectural design. The real significance of this type of work is in that it represents a type of architecture that can not exist if it were not for the computers.

**Shifts in the perception of using computers**

Using computers in architectural design as concept generators has recently become trendy, thanks to the work of such signature architects as Peter Eisenman and architectural theorists as Jeff Kipnis, Greg Lynn etc. Especially younger generation faculty together with enthusiastic and talented students are actively forging new frontiers in design computing. Educational and professional accreditation agencies are promoting the use of computers both in schools and practice.

**Influence of Internet**

Explosion of WWW over past couple of years is drawing much attention to digital media and is affecting everyday life significantly. Collaborative design is becoming a buzzword in both academia and the industry. Emerging internet and VR (Virtual Reality) technologies are transforming architectural practice into new realms [9].

**Part 2**

This section describes an introductory course in CAAD developed and taught by the author that addresses Klerker's issues to some extent. A wide assortment of popular commercial graphics software programs were used in this class. The methods for their usage were inspired by Sutherland's [5] remark,

"As soon as the process of computer-aided design is considered as building a computerized description of the object being designed rather than as the process of drawing the object being designed, horizons become tremendously expanded."

Method and product were emphasized over the means. Also discussed are the methods developed to compile and share these experiences within the school.

**Background**

Despite its rich tradition in advanced research and applications, the Department of Architecture at Ohio State did not offer courses in CAAD for its undergraduates. The undergraduate students were sent to the Department of Engineering Graphics to fulfill their computer literacy requirement until 1990 which proved to be very ineffective. The department has decided to teach computer literacy to its undergraduates internally and Arch 371 was created. In 1992, I was asked to teach it and was given complete freedom to redesign its content and structure. I have developed and taught this course until 1995 during which it has evolved significantly.

**Teaching philosophy**

One of the motivations for using computers is that they allow iterative processes, allowing the study of variations of an evolving theme. In practice, we use computer graphics programs for CAAD and they require a high level of geometric specificity for description of an artifact [3].
When incorporating such programs for design, students, most of whom are inexperienced in design, develop a fixation to their design ideas because of the enormous amount of energy expended in modeling the artifacts and loose inclination to experiment. Also, as Gross [2] points out, beginning student's preoccupation with using computer programs can limit the design possibilities to a certain impressive features of the program.

Over the history of evolution of built form, architects have accumulated some very valuable visual problem techniques that are based on "graphics analysis" and "constructive geometry". In Arch 371, computers were introduced as natural extensions to these skills instead of substitutes. Also emphasized was the need for multiple representations for design problem solving using a scheme with solid modeling at its core, supported by interrelated sub schemes including animation, drafting, geometric modeling, image processing, painting, rendering, and word processing. As pointed out by Eastman, "Multiple representations break down the issues and allow them to be dealt with piecemeal; otherwise they would be overwhelming in their complexity." This scheme employed the several popular commercial software on Macintosh platform.

Course Objectives

Arch 371, being the only undergraduate course dedicated to computer applications in the department, it had to have a rather broader agenda. No prior computer experience on the part of the students was assumed or required. The objectives of this course were to introduce the basic principles of the use of computers, electronic media, and computer graphics for architectural design. Upon completion of the course, the student was expected to demonstrate:

1. a working knowledge of the operation and application of microcomputers,
2. the ability to integrate simple applications in the continuum of CAAD software, including sketching, painting, drafting, and modeling programs,
3. the application of software and computer skills to assist in other courses in the School and the University,
4. an understanding of the integration of different computer software for appropriate use in design studio settings,
5. an awareness of the scope and direction of the various CAAD courses offered by the Department of Architecture.
6. a basic understanding of principles of CAAD software to recognize the underlying similarities of different commercial software and to be able to quickly learn any kind of software in an office or a studio setting.

Course Structure

First, students select either a published project of their interest or project that they have recently designed and deconstruct the design process and present it from conception to completion using traditional presentation techniques including sketches, photographs, drawings, and chipboard and plaster models. Operational models of various digital media are then presented to the students and their inherent semantics and building blocks such as pixels, lines, polygons, surfaces, cubes etc., and the rules their composition are explained. The underlying similarities and differences between the manual and digital methods are discussed. The students are then encouraged to draw metaphoric analogies between the digital and the architectural semantics, and to gradually discover various software tools. Subsequent exercises promote an understanding and usage of digital tools in the spirit of design representation as a dynamic and fluid process facilitating exploration as a discovery process rather than merely representing an assemblage of pre-conceived notions.
Teaching method

The pedagogical model of "learning by doing and doing by learning" was used in this course. About 8 projects (lasting a week each) were designed to meet this objective. Students started by learning the underlying principles of respective digital representation and then grasp them by actually working on the projects. The course comprised of weekly segments consisting of two three-hour sessions. Most of the first session focused on lectures and discussions pertaining to theory and principles of CAAD as they relate to the projects, accompanied by demonstration of relevant features of various software programs. Remaining time was spent in the lab where students work on computers under the supervision of the instructor and a graduate teaching associate.

Underlying Metaphors

The sequence in which various digital media were introduced paralleled the natural order of learning comparable skills in the analog world. Also, as a point of departure we used the corresponding metaphors of analog world. For example, students first approach word processing using their typewriting skills and experiences as a basis. Gradually they uncover the strengths of the new medium, such as the possibilities to change type face, spell checking, mass mailing etc. They also understand the limitations of word-processing using microcomputers compared with typewriting (for example: typewriters are better suited for filling an application form or typing a short letter in case when a printer may not be available etc.).

<table>
<thead>
<tr>
<th>Digital medium</th>
<th>Real world skills/experiences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word-processing</td>
<td>Type writing skills</td>
</tr>
<tr>
<td>Sketching</td>
<td>Etch-a-sketch® experiences</td>
</tr>
<tr>
<td>Painting</td>
<td>Painting by numbers, finger painting, erasing</td>
</tr>
<tr>
<td>Drawing</td>
<td>Sketching on a grid paper, Object drawing skills</td>
</tr>
<tr>
<td>Drafting</td>
<td>Drawing graphs in science classes, drafting using pin-registered overlays, stencils for lettering and patterns, templates for drawing circles, furniture symbols etc.</td>
</tr>
<tr>
<td>Solid Modeling</td>
<td>Playing with Lego® blocks</td>
</tr>
<tr>
<td>(SPLINE-based)</td>
<td>Model building using plaster, wood, chipboard etc.</td>
</tr>
<tr>
<td>Surface Modeling</td>
<td>Paper mache models</td>
</tr>
<tr>
<td>Rendering</td>
<td>Photography, rendering with color</td>
</tr>
</tbody>
</table>

Table: Mapping of analog skills to digital realm

Required readings:

Required readings included topical notes prepared by the author on digital sketching, digital painting, drafting, and solid modeling which explained the underlying mechanics of each representation. These notes were typically 2 to 3 pages long and contained specific references to relevant topics as discussed by Mitchell [5] and Schmitt [6] in their books.

Reference material:

The reference material distributed to students for working with various computer programs included worksheets, quick reference cards, list of commands, tutorial notes for frequently asked questions. Whenever possible we prepared and used our own worksheets for various software, instead of using the manuals or tutorials published by vendors. We believed that when employing commercial computer graphics system for design tasks, 90% of the work is
accomplished with 10% of its capabilities [Doug stoker 1992]. Software manuals tend to be a comprehensive documentation of the feature sets and tutorials are written to showcase their salient features. Our notes were prepared in parallel to the concepts explained in lectures and readings accompanied by specific references to the software used. This allows students to distinguish the concepts from the mechanics and possibly use the same worksheet as a guide to learn another “brand” of software.

We also discouraged memorizing the commands, instead we encouraged students to think about the task they want to accomplish when they work with a software program, and then look for a command/operation that program offers that is closest to a verb that describes the intended action.

Whenever an individual student has hardship accomplishing a specific task while working on their project they could get help from the instructor (but only after they try talking to fellow students, graduate assistants, and referring to the manuals). Students were encouraged to document the help sessions into a one or two page tutorial notes. Such efforts were rewarded by an extra credit of 5% of the course grade. This was primarily due to the fact the students problems were typical; once we stockpiled the tutorial notes and trouble shooting recipes for these situations, all we had to do was to point the student to the relevant worksheet. These sessions also helped the instructor modify the subsequent teaching strategies.

### Software demonstrations:

After experimenting with both "trade-show" and "trade-school" methods, we decided to discard them both and chose to develop our own method of demonstrating the software to students. Our methods concentrated on mapping the principles and methods discussed in lectures and readings to the relevant mechanics of the specific software being used. We depended on the students to learn by actually trying it themselves (with the help of worksheets as discussed in previous section). Our methods had the following positive results:

1. Better retention of working skills,
2. The learning process gave students a sense of accomplishment and built confidence because they uncovered software features, rather than becoming "spoon-fed button pressers",
3. An awareness of being able to accomplish a given task in more than one way.

### Previous projects:

We also accumulated what we considered good, average, and bad projects from previous quarters and showed them to the students before they started working on their projects. In some cases we made the copies of such projects available to students so that they could dissect them to observe the methods and mechanics used. This also helped students establish a focus and budget their time.

### Hands-on help:

Instructor and a graduate teaching associate were available for consultation and troubleshooting during class sessions as well as during designated hours. Students also learned a lot from one another.

### Sequence of projects:

The projects were structured on the premise that the students would have taken at least two design studios at Ohio State, before they are allowed into this class. This means that they are familiar with graphic analysis and representation of architectural design using such analog media as collage, freehand sketches, hard-line drawings, chipboard models, wood models, and photography. First project in Arch 371 provides the point of departure for the rest of the
quarter, where the students deconstruct the project of their choice and present the graphic analysis using such 2D aids as sketching and drawing (pictorials expressing planimetric syntax), documentation of complete geometric description of the project as photocopies through literature research, and 3D aids including plaster or wood massing model (expressing volumetric syntax), and chipboard models (expressing the interior spatial organization).

First three projects introduced 2D software programs, where they replicated the corresponding manual methods used to describe their projects in their analysis. The projects resulted in making a sketch in Canvas, a painting in Canvas, a scanned image manipulated in Photoshop, a page layout in PageMaker, and a floor plan and a section drawn in AutoCAD R-12. At this stage they got familiar with the characteristics of the tool and attempted a metaphoric translation of their graphics analysis and construction geometry skills into digital realm. We encourage the students to take a note of the obvious advantages of digital media such as being able to undo, flexibility in output, cleanliness, accuracy etc., we also directed the students attention to the following issues:

1. The infamous "garbage in garbage out" axiom. i.e. computers do not possess magical powers to fix their drawings or designs.

2. The representation schemes used up to this point are static\(^\text{16}\) and rigid\(^\text{17}\) in nature.

We attributed the static nature to the 2D nature of the media and rigidity as a characteristic of the software limitations in understanding the architectural intent. We then sought to overcome the static nature by using 3D media, and rigidity by data organizational schemes.

The second stage started by building a massing model of their project where students had to demonstrate a grasp of inherent dynamism of solid models by associating multiple views to 2D representations such as plans, sections, elevations as well as 3D representations such as axonometrics, perspectives etc. This exercise lasted for about at week. The Instructional emphasis was to foster fluidity in the modeling methodology. It had to be a generative process where models with low-level detail were built, and organized, such that any or all of it can be developed into a higher level detail for further study. We called this process "doodles\(^\text{18}\) to models". The idea being the lower-level models can be generated effortlessly playing the role of doodles in the digital realm, that can be developed into various higher levels of abstract states of varying detail. From this point on, we spent about 2 weeks in mastering mechanics and applications of selected derivative methods for developing higher level detail from model with lower level.

![Diagram of software tools](image)

*Figure. Various software tools used as part of architectural design representation scheme.*

The third and final stage concentrated on demonstrating the skills of developing higher level representations from "doodle like" models with the help of various programs. Students revisited the 2D programs to take advantage of their capabilities for developing higher level detail as appropriate. An example could be making a floor plan representation by taking a horizontal section of a solid model built in form\(\text{Z}\) and developing it further by adding more details and drafting conventions such as dimensions, text, line weights, hatch patterns etc. in AutoCAD. This process was used as a two way street, i.e. certain 2D tasks better supported
by AutoCAD, are handled in that program and transferred back to form•Z. Similarly, image processing in PhotoShop was used to manipulate a scanned site plan which was later used as an underlay to build the site context model.

Conclusions

Our experiences with Arch 371 were successful in fostering a computer culture across the department and enticing more students to use computers in design studio and other courses. Exposing students to popular drafting program (AutoCAD) had very positive influence on student self-confidence and many have secured the mis-titled positions such as AutoCAD architect, MicroStation architect in AEC firms. Many students were pleasantly surprised to see the interviewers being inquisitive about the computer generated 3D imagery in their portfolios. Although most of them started out with routine drafting jobs using computers, many were able to convince their bosses to expand the use computers to 3D modeling/rendering/animation and decision making. Some of them have been instrumental in advancing the state of computing in their offices particularly to 3-D modeling, rendering, and animation.

We are convinced that learning digital techniques for architectural design representation occurs faster when they are based on corresponding analog metaphors. Another benefit to this method is that it keeps the students "digital skills set" generic making it possible to adopt them to any specific commercial computer graphics program easily. The mechanics involved in developing higher level details from a lower-level, mapping mechanisms between various programs are very procedural and do not require anymore intelligence than that is required for operating a car. With practice, they can become second nature to the students. Once this happens the focus shifts form how they are doing to what they are doing.

We expect to build on this positive experience and extend these methods to such courses as Graphics Analysis of Architectural Precedent, Freshman Graphics, and Design Studio at the Department of Architecture, Miami University. Also, we are investigating funding possibilities to build a "collective knowledge base" of our experiences on an intranet and gradually link it to the internet for sharing with others.

Acknowledgments

The ideas and methods described here owe much to the following individuals: Chris Yessios for thoughtful input for the course content; Jose Oubrerie for the encouragement and for suggesting the "worksheet" idea¹; all the graduate teaching associates who assisted in this course, especially Ellen Gelrich for her constant input in the course development.

References:


Notes

1  Klerker succinctly points out the three important issues particularly relevant to architectural education today: (1) need for improving pedagogic efficiency in the education of architects in general and making more natural space for CAAD in particular; (2) dilemma of professional training vs. broad based educational; (3) how to build a stock of practical experiences to be used for educational purposes?

2  In his article, John McIntosh recalls the forum held for the initiation of ACADIA during Fall 1980, where Eastman points to the lack of pedagogy to drive technology in right directions being a problem over the unavailability of technology itself. In the same article McIntosh comments that the situation has not changed to date. McIntosh, John. "The founding of ACADIA". ACADIA Quarterly, Fall 1995. Pp. 3.

3  Understanding the reasons for this can be a dissertation in itself.

4  I chose to use this word to imply that what existed as high-end technology on mainframe computers until recently is becoming available on an affordable micro computer.

5  As Gehry acknowledges in his Lecture on April 16, 1996 at Ohio State University, that computers are instrumental in his work to concoct the form, to convince the clients and more importantly the contractors.

6  Previously students were to choose between "computers" and "design" and a lot of talented designers were drawn away from computers in favor of design. This trend is changing.

7  NAAB requires some computer component in architectural curricula for accreditation. From year 1997, NCARB will be administering the architectural licensing exams on computers.

8  See a chapter dedicated to the Department of Architecture at the Ohio State University in Pioneers of CAAD, 1985.

9  Peter Eisenman, Ben Gianni, Jeff Kipnis etc. have used computers in their studio classes at Ohio State as concept generators for architectural design that have gained international acclaim in design community.

10  auto•des•sys form•Z RenderZone® v 2.6 for solid modeling and rendering; auto•des•sys Imager® for batch rendering of hidden line images of solid models; Electric Image® v 1.6 for Rendering (Phong shading) and animation and slide shows; AutoDesk AutoCAD® R-12 for drawing and drafting; Adobe PhotoShop® 3.0 for image processing, painting, compositing; Adobe PageMaker® 5.0 for page layout; Deneba's Canvas® 3.0 for painting and drawing; Microsoft Word® 5.0 for word processing; Ofoto® software for 2D scanning.

11  In the words of an overwhelmed student 10 pounds of potatoes packed into a 5 pound bag.

12  This issue could be moot for next generation of students who are growing up with video games, CD-ROM tutors for coloring/drawing before they even go to preschool.

13  Etch-a-sketch® is a popular toy for children in many countries. Using this toy a child makes sketches by controlling a marker through a rotary dial that etches lines over transparent medium. This is perhaps first experience in transferring the ideas second hand.

14  Another childhood toy that deals with modular spatial and object composition.

15  "trade-show" demos that focus on showcasing the software feature set. "trade-school"
instructional demos follow a prescriptive model where students sit by individual terminals and learn how to press buttons in predetermined sequence imitating the actions of an instructor projected on to a bigger screen.

16 View dependent.
17 Not very easily modified for studying variations.
18 Architects are known to doodle (sketch as they think, or otherwise pre-occupied on cocktail napkins, notepads, news papers, or just about anything they have at hand) to brainstorm.
19 Jose Oubrerie, Chair of Architecture, dragged the author to the fax machine in the department and pointed at the "three line" instruction sheet written by the department secretary taped to it, and said that although he did not know anything about fax machines, this sheet enabled him to get his work done! Why can't this be done with computers? While one can question comparing a such specific linear task as faxing with complex non-linear nature of using computers, I think that Prof. Oubrerie does have a point (breaking the ice).

Appendix-A: Typical schedule of projects

Appendix-B: An example of modeling process as practiced in this class

Appendix-C: A typical student project at the end of the quarter
<table>
<thead>
<tr>
<th>#</th>
<th>Project Title</th>
<th>Topics</th>
<th>Software</th>
<th>Start on</th>
<th>Finish on</th>
<th>Weight</th>
<th>No. of Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Hand Simulation</td>
<td>Project Selection, analysis and hand simulation of various computer representation techniques as discussed in class. Must include a word processed essay (300 words min, 1 page max) about your building.</td>
<td>Word</td>
<td>3/30</td>
<td>4/4</td>
<td>5%</td>
<td>10</td>
</tr>
<tr>
<td>1</td>
<td>Illustration</td>
<td>An expression of your project using Sketching, Painting, Text Layout techniques.</td>
<td>CANVAS</td>
<td>4/4</td>
<td>4/11</td>
<td>5%</td>
<td>10</td>
</tr>
<tr>
<td>2.1</td>
<td>Drafting-1</td>
<td>Getting started with Computer Aided Drafting using AutoCAD. Preparation of a title block and bubble diagram / part of your project to demonstrate setting UNITS, limits, basic drawing commands, and plotting.</td>
<td>AutoCAD</td>
<td>1/17</td>
<td>4/18</td>
<td>5%</td>
<td>10</td>
</tr>
<tr>
<td>2.2</td>
<td>Drafting-2</td>
<td><strong>3D computer representational issues:</strong> Grids, Scale, Blocks, Layers, UCS. <strong>Drawing techniques:</strong> Grid Snaps, Object Snaps, Array, Mirror, Divide, Offset, Trim, Extend. <strong>Drafting Conventions:</strong> Text, Dimensions, Hatch Patterns, Line Weights. Preparation of a floor plans to demonstrate the above.</td>
<td></td>
<td>1/24</td>
<td>4/25</td>
<td>10%</td>
<td>10</td>
</tr>
<tr>
<td>3.1</td>
<td>3D Modeling-1</td>
<td><strong>Introduction to 3D modeling:</strong> Planes, Viewing, Grids, Units, Visualization. <strong>Understanding 3D data organization model.</strong> Drafting / Modeling relation and basics of Interface in formZ. <strong>Building a simple massing model of your project using Basic Creation and editing Operations including extrusions, derivations, revolutions, sweeps, booleans, geometric transformations, topological editing operations.</strong></td>
<td>FormZ</td>
<td>1/31</td>
<td>5/2</td>
<td>10%</td>
<td>10</td>
</tr>
<tr>
<td>3.2</td>
<td>3D Modeling-2</td>
<td>Hierarchical Modeling: Building a more complete 3D model of your project with interior walls, openings, floors and roofs, stairs, and the site with trees, path ways, and landscaped form in hierarchical manner.</td>
<td>FormZ</td>
<td>2/7</td>
<td>5/9</td>
<td>15%</td>
<td>15</td>
</tr>
<tr>
<td>3.3</td>
<td>3D Visualization</td>
<td>Introduction to Static Animation / rendering. A perspective tour of your building to be presented as a slide show as well as hard copies. You will learn how to set up perspective views and render them in a &quot;batch mode&quot; using FormZ Imager or by porting your model to a dedicated rendering program such as Electric Image and composing these images as a slide show.</td>
<td>FormZ</td>
<td>2/14</td>
<td>5/16</td>
<td>10%</td>
<td>10</td>
</tr>
<tr>
<td>2.3</td>
<td>Drafting-3</td>
<td><strong>Methods of generating a complete set of floor plans, sections, and elevations for your project using 3D model built in formZ.</strong> One Section and One Elevation in a presentable quality are required. You may use the floor plan done in Project 2.2 to complete the set.</td>
<td>FormZ</td>
<td>2/21</td>
<td>5/23</td>
<td>10%</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>Portfolio</td>
<td><strong>Integration of all your work and Final Presentation.</strong> Must include: a. Front Cover: your name, Course Number, Quarter, Instructor’s Name, Date. b. free hand sketch done for Project 1. c. at least one Floor Plan, One Section, One Elevation from Project 2.3. d. Renderings from Project 3.2: Birds Eye View of your 3D model showing your building and site. At least one Exploded Axon or Sectional Perspective from Project 3.2. e. At least 4 interior, 4 exterior views of your building that you have generated for Project 3.3. f. Back Cover: Blank or your favorite drawing done for this class.</td>
<td></td>
<td>3rd week of the Quarter</td>
<td></td>
<td>15%</td>
<td>15</td>
</tr>
</tbody>
</table>

*Number of hours suggested here represents the actual work hours you are expected to put in to the project... not including the class meetings.*
Building Ando's Chapel in form•Z with varying levels of details. Once the massing is geometrically described the rest of the details are developed from it using derivative operations.

Instructor: Murali Paranandi
The Giovanniti House by Richard Meier is that of a building that is shaped by its limited site. The organization of the house is based on the consideration of the private to the public. The public pertains to the views that the homeowner is granted by the controlled directionality of the solid to translucent forms which make up this house. Expanding the feeling of the space is done by allowing the floors to open to one another at certain areas, achieving a larger more interesting living environment.

This portfolio consists of a progression of learning on the computer the abundance of design and organizational programs available. It represents a compilation of my knowledge of tools, each used for the purpose at which they are meant to excel, to compose sketches, hard-line drawings, 3D models, and text that accompanies. Software used includes Canvas, PhotoShop, PageMaker, AutoCAD R-12, formZ, form-Z Imager, Electric Image.

Chris Maresca, Arch 371, Au 94. Instructor: Murali Paranandi.
Order a complete set of eCAADe Proceedings (1983 - 2000) on CD-Rom!

Further information:
http://www.ecaade.org