Teaching portfolio

Fuhua (Frank) Cheng (Fall, 2004)

A. Teaching Evaluation

1. Reflective Statement

My teaching has three goals: (1) to make sure that students understand the course materials well, (2) to make sure that students know how to use/apply the materials they learn in class, and (3) to make sure that the students are evaluated fairly.

To achieve the first goal,

- I use a motivation-driven approach in my lecture, i.e., I give the background and applications of the result first, and then explain the theory that leads to the result.
- I give many examples in my notes (see, e.g., my CS535 and CS633 notes).
- I encourage the students to be involved and active during lectures. (However, for those who find it difficult to do so, I welcome them to ask questions or make comments after class.)
- I make all my class notes available on line so that, instead of copying my notes in class, they can closely follow my lectures on course materials. (My class notes such as CS633, CS631, CS535, and CS321 have been used by some of my students and colleagues in their own classes.)

To achieve the second goal,

- I give applications for each covered result.
- I provide students with sample programs to help them initiate their work (see my web pages).
- I encourage students to share their ideas.
- I award students with extra credit if they have new ideas on assignments.

To achieve the third goal,

- I always let the students know at the outset of the course exactly what is expected. I clearly specify the requirements of the course such as materials to be covered, grading policy, program requirements (see, e.g., my CS535 and CS633 Programming Requirements), late penalty, and numerical scale to be used in the evaluation, on the first day of class.
- I provide students with solution sets for all homework assignments and exams (see my webpages) so they would not only know the solutions to the questions, but also know if their works are graded fairly.

I have different expectations for graduate and undergraduate students though. For a undergraduate or programming-extensive course, the students are evaluated based on two subjects: programming assignments and tests. I usually put equal weight on both sides so the effort of the students can be evaluated fairly. However, I encourage students to do critical thinking and they get extra credit if they do so such as providing comments or improvement on existing techniques. For a seminar course or advanced topics, I evaluate the students mainly based on the quality of the work, i.e., I will follow the numerical scale, but a student with good ideas will get more extra credit than the ones who don't.

2. **List of Courses taught**

Term	Course	Title	Enrollment
Spring 03	CS633	Computer Animation (new)	11
Fall 03	CS631	Computer Aided Geometric Design (revised)	6
	CS535	Intermediate Computer Graphics	24
Spring 04	CS633	Computer Animation	13
Fall 04	CS321	Intro. Numerical Methods	30

Computer Aided Geometric Design (CS631): Computer Aided Geometric Design (CAGD) provides the mathematical basis for dealing with geometric data. Applications of CAGD include: representation of large data sets, visualizing products, automatically producing sectional drawings, designing pipe systems, modeling surfaces arising in the construction of cars, ships and airplanes, production and quality control, robot path description, and milling machine control.

This course presents the mathematical foundations of geometric data processing. We discuss the mathematical methods that underlie CAGD. Our aim is to provide mathematical techniques which will enable a student to develop his/her own software for generating, describing, and modifying freeform curves, surfaces, and volumes.

Intermediate Computer Graphics (CS535): This course covers two dimensional graphics such as rasterization of lines/polygons/curves, clipping, anti-aliasing and three dimensional graphics such as modeling, viewing, lighting, shading, hidden line/surface removal. More advanced topics such as solid modeling, curves and surfaces, advanced raster graphics architecture and algorithms, advanced modeling techniques, and animation will also be covered. The supporting graphics system used in this course is OpenGL.

Computer Animation (CS633) (new): This course presents algorithms and programming techniques for specifying and generating motion for graphical objects. It addresses practical issues and provides accessible techniques and straightforward implementations. It is not intended for animators using off-the-shelf animation software, nor does it address the issue of *computer-assisted animation*, i.e., the computerization of conventional hand-drawn techniques. This course is primarily concerned with 3D computer animation.

Motion specification techniques in two categories: *interpolation and basic techniques* and *advanced algorithms*, are studied and discussed. The interpolation and basic techniques category consists of ways in which the computer is used to fill in the details of the motion once the animator specifies the required information, such as *key framing* and *path following*. Advanced algorithms generate motion using a set of rules or constraints that specify *what* is to be done instead of *how* is to be done. Model-specific applications are also surveyed. These are grouped into two general areas: *natural phenomena* and *figure modeling*. The graphics library used in this course is OpenGL.

Intro. Numerical Methods (CS321): This course presents the basic techniques for the design, use, and understanding of numerical algorithms. The students will be given application-oriented programming assignments to enhance their ability in solve practical problems.

Students will learn basic concepts, problems and methods used in numerical computing. Specifically students will be able to

1. estimate computed errors
2. select/propose methods that yield small errors (if possible)
3. understand important properties for a number of basic methods (e.g., Gaussian elimination, Lagrange and spline interpolation, Trapezoidal and Simpson's quadratures, Newton's iteration, Runge-Kutta methods).
4. modify problems for better algorithm performance
5. analyze results computed in fl-arithmetic

3. Course Syllabi

(see attachment)

4. Student Evaluation

		2003			2004
		Spring	Fall		Spring
		633	631	535	633
	Enrollment	11	6	24	13
	Number of answers	9	5	18	8
1	Material/grading outlined	3.7	3.4	3.3	3.4
2	Textbook	3.2	3.3	3.6	3.4
3	Supplemental reading	3.3	3.6	3.3	3.5
4	Exams reflection	3.7	4.0	3.4	3.4
5	Grading fair	3.7	3.6	3.2	3.5
6	Distributing assignments evenly	3.9	3.8	3.5	3.5
7	Assignments graded promptly	3.6	3.4	3.1	3.3
8	Grading including comments	3.6	3.5	3.1	3.4
9	presentation	3.6	4.0	3.4	3.6
10	Knowledge of subject	3.9	4.0	3.7	3.6
11	Availability	3.7	3.8	3.4	3.5
12	Answer questions	3.8	4.0	3.6	3.4
13	Stimulate interest	3.7	3.6	3.6	3.1
14	Encourage participation	3.6	4.0	3.3	3.3
15	Respect viewpoints	3.3	3.6	3.5	3.6
16	Ability to analyze	3.7	3.6	3.3	3.3
17	Solve problems	3.6	3.6	3.4	3.3
18	Understand concepts	3.6	3.8	3.4	3.4
19	Read further	2.9	3.8	3.2	2.9
20	Value of course	3.4	3.8	3.4	3.3
21	Quality of teaching	3.7	4.0	3.3	3.4

5. Class Notes

(CS535, CS633, CS631 and CS321 can all be accessed from my website)

B. Advising Evaluation

1. Reflective Statement

My goal in advising a project or a thesis is to ensure that the student knows how to set up a target and how to develop a strategy to reach that target. The target must be very specific and the strategy must be practical. The idea is to let the student know how to play a game by him/her-self and to what extent that he/she should keep trying before giving up. I help the student with the technical part initially after he/she has successfully performed background study, target selecting, and strategy design.

My advising in pre-registration meetings with the students will ensure that (1) students understand the requirement of a computer science major in addition to the college and university requirements, and (2) each student develops an appropriate course plan for each semester. This will be achieved by going through a checklist with the student and showing him/her the best combination for the semester.

2. Numbers of Students Advised

	2003		2004	
	Spring	Fall	Spring	Fall
Undergraduate Program Advisees	32	29	21	27
Graduate Program Advisees	7	8	9	10
MS Committees*	2	2	2	0
PhD Committees**	5	5	5	4

* Xiaoping Lao, Lei You, Kumar S. S. Dammu, Chengdong Li, Charles S. Carpenter, Geetashree Chakravorty

** Quanren Xiong, Xiqun Wang (ME), Xiang Fang (DSIS), Wei Su (EE), Leonard Hoffnung (MATH)

3. PhD Students

- **Shuhua Lai**

  Area of Research: *Subdivision surface based one-piece representation*
  Starting Date: January 2003
  Supported Period: January 2003 - present (supported by NSF grant DMS-0310645).
  Current Status: *passed all breadth and depth exams. Working on thesis now*
  Publication: One journal paper, one conference paper, one in preparation
  Scholarship: IBM PhD Scholarship for 2005-2006 (pending)
  Anticipated Graduation Date : May 2006.

- **Alice J. Lin**

  Area of Research: *Highlight and Shadowgraph Lines based Surface Fairing*
  Starting Date: August 1999
  Current Status: *quit PhD program in May 2003*
  Publication: *none*.

Supported period: June 2000 - May 2003 (supported by NSF grant DMI-9912069).

- **Qi Chen**
  Area of Research: *Shadow Generation based on Angular Representation and One-Pass Z-Buffering*
  Starting Date: August 2003
  Supported Period: August 2004 - present (supported by NSF grant DMI-0422126).
  Current Status: *passed no breadth and depth exams yet.*
  Publication: one paper in preparation (One-pass based shadow generation)
  Anticipated Graduation Date : May 2007.
- **Jiangbo Li** (Temporary assignment)
  Area of Research: *Computer Networking*
  Starting Date: August 2002
  Supported Period: August 2004 - present (supported by Ken Calvert).
  Current Status: *passed two breadth exams (NA and Theory).*
  Publication: one conference paper (Challenges in implementing an ESP service)
  Anticipated Graduation Date : May 2007.
- **Shunnan Chen** (Temporary assignment)
  Area of Research: *Computer Vision*
  Starting Date: August 2003
  Supported Period: August 2004 - present (supported by Ruigang Yang).
  Current Status: *passed no breadth exams yet.*
  Publication: one paper in preparation
  Anticipated Graduation Date : May 2008.

4. **MS Students**
   - **Jidong Qu**
     Masters Project: *Shape Modeling using Subdivision Surfaces*
     Current Status: *finished vertebrae modeling using subdivision surface based one-piece representation*
     Date of graduation: December 2004.
   - **Shiming Zou**
     Masters Project: *Constrained scaling of Catmull-Clark subdivision surfaces*
     Publication: one journal paper (*Constrained Scaling of Catmull-Clark Subdivision Surfaces*)
     Date of graduation: February 2004.
   - **Gang Chen**
     Masters Project: *Subdivision Depth Computation for Extra-ordinary Patches*
     Publication: one submitted (Subdivision Depth Computation for Subdivision Surfaces)
     Date of graduation: December 2004.
   - **Xiqun Wang**
     Masters Project: *One-Pass Shadow Generation*
     Current Status: *system testing now*

Anticipated date : May 2005

- **zhaohui Ren**
  Masters Thesis: *Distance Evaluation for Catmull-Clark Subdivision Surfaces*
  Current Status: *theory development stage*
  Date of graduation: May 2005.
- **Ping Du**
  Masters Project: *Catmull-Clark Subdivision Surface Interpolation*
  Current Status: *theory development stage*
  Date of graduation: May 2005.

5. **Graduate Students**:
   - **Yuan Hong** (passed two breadth exams)
     Research topic: *Eigenvalues and Eigenvectors of Subdivision Surfaces*
     Current Status: *doing research on identifying simplified form of discrete Fourier Transform in eigenvalue computation of subdivision surfaces*
     Supervising period: 8/23/04 - present (supported by NSF grant DMI-0422126)

6. **Student Activities Summary**
   - **Summer Internship**
     Student advised: K. Michael Cheng
     Advising period: summer, 2004

Attachments

1. **Syllabi**

   CS535 - Intermediate Computer Graphics
   CS633 - Computer Animation
   CS631 - Computer Aided Geometric Design
   CS321 - Intro. Numerical Methods

2. **Materials prepared for teaching activities** (all can be accessed from my website)

   CS535 - class notes, programming assignments (with sample programs), homework assignments (with solution sets)
   CS633 - class notes, programming assignments (with sample programs), homework assignments (with solution sets)
   CS631 - class notes, programming assignments (with sample programs), homework assignments (with solution sets)
   CS321 - class notes, programming assignments (with sample programs), homework assignments (with solution sets)