CMSC 635 Advanced Computer Graphics

Course Syllabus, Spring 2013
Instructor: Jian Chen
Location: Janet & Walter Sondheim 112

Welcome! This is a graduate-level computer graphics course. The primary goal of this course is to expose students to a wide range of subjects in computer graphics, taking a sophisticated approach to each one. A secondary goal is to give students in-depth experience with a particular topic via the final project. Although we aim to exercise students' software design skills and teach them new tools (software and math), the majority of the time spent in the course will be related to learning fundamental methods and theories in computer graphics.

Many students take CMSC 635 as the second half of a two-course sequence starting with CMSC 435. CMSC 435 is prerequisite for the course. The course will move quickly and cover many topics. A strong emphasis will be placed on student initiative. The class will focus on a significant final project (50% of your final points). Most final projects should be publishable or sufficient to be a Master thesis with or without extensions.

There are plenty of conferences and journals for publishing your results, e.g., IEEE virtual reality, visualization, SIGGRAPH, SIGGRAPH asia, TOG, I3D, 3DUI, UIST, and TVCG etc. Paper written in the ACM SIGGRAPH or IEEE Visualization formats is mandatory.

The Staff

Prof. Jian Chen (jichen@...) is teaching this course.
Dr. Keqin Wu (keqin@...) will be a faculty assistant to help answer your questions.

Special thanks to Dr. John Hughes, Dr. Andries van Dam, and his TA for their help!

Office Hours

Jian Chen: ITE 357; 12:15pm-2:15pm on Mondays
Keqin Wu: ITE 376; 11:00-12:00 noon on Thursdays

Classes meet at 2:30-3:45 pm Monday and Wednesday, Jan 28 - ;

Textbooks (references)

Real-time rendering, by Akenine-Moller, Haines and Hoffman (3rd edition)
Computer Graphics: Principles and Practice, by Foley, van Dam, Feiner, and Hughes.

Expectations and Assignments

As mentioned, CMSC 635 is a graduate-level course, and we have high expectations of the students. Students are expected to be mature and professional about their work habits and should expect to spend 15–20 hours per week on the course. The course will rely on strong software engineering skills and some mathematical maturity, especially in linear algebra and probability. The programs test more complex concepts than those in CMSC 435, and usually require significant thought before any code is written. In several cases, the time spent understanding the problem and devising a solution will constitute a majority of the time devoted to the assignment.

CMSC 635 will require the completion of 3 written homeworks, 3 rendering and 1 slightly longer interaction assignments (of about 1 week each), and one large (5-week) project, typically a two- or three-member team effort.

Your final grade will break down as follows:

Rendering assignments (15%) + Homeworks (15%) + Section participation (5%) + Interaction assignments (15%) + Final project (50%)

Workload: This set of assignments probably looks like a burden, but in fact, if approached sensibly (i.e., working steadily), you will have sufficient time for each and every one. The normal load is about 15-20 hours per week. Good time management will make this course much more enjoyable.

Extra credit: There is ample room for bells, whistles, and other credit-garnering efforts on the part of ambitious programmers. You are invited to get creative, as long as it does not make you late. Rewarding bells and whistles with extra credit is left to the discretion of the instructor, so I strongly encourage you to discuss your creative plans with the professor before you forge ahead to make sure that they are considered appropriate for credit. Also, keep in mind that bells and whistles should only be done after the standard assignment is fully working since they won't count in lieu of missing or buggy features! CMSC 635 is a great vehicle for extra credit: if you finish a program a little early, seek appropriate inspiration and add something fancy.

Final project: The final project is perhaps the best-known and most rewarding part in this course. It is important that students keep the final project in mind throughout the entire semester. Groups that form research ideas for topics early will have more time to realize their goals than a group that devises their topic on the night before the project proposal is due.

Late policies: All coding assignments are to be handed in on the due date by 11:59 PM. All written homework's are due at the beginning of class. Programming Assignments: Handing in late will cost 10% for every 24 hours. The penalties come in units of 10%, so that if a program is 10 minutes late, it loses 10%. This means that a B project becomes a C project. There is one exception: final projects may not be handed in late. You will not receive credit for late homeworks. (**Late Days)** Everyone is allowed two late days. (This does not mean 48 "late hours"). If you wish

to use a late day on an assignment, put it in the README that you hand in with the assignment and your due date will be extended by 24 hours. They will not be tallied automatically, but you may use them as you see fit. You cannot retroactively use a late day nor can you use it for your final project. For the final project: you cannot submit the final project later than the indicated date. If you do so, we will grade it NC without looking at it! You also must attend all final project demos on May 20.

Support Code:

We will be using the G3D innovation Engine as the main support code for most (if not all) of the projects. G3D is a commercial-grade C++ 3D graphics engine that supports, in addition to the features we will use, hardware accelerated real-time rendering, off-line rendering, and general purpose GPU computation. You can find the G3D Innovation Engine at http://g3d.sourceforge.net. Somewhat the engine is created and maintained by a bunch of Brown'ers! And the documentation is somewhat poor. You might want to submit to their mailing list where you could ask programming related questions. If you want to do virtual reality related project, ask Jian. She will give you a version of VRG3D where most hassles of setting up hardware facilities are very well handled.

It will be good to download the code as soon as possible and start trying out some examples.

Community Spirit Credit

In keeping with the goal of avoiding needless work, we will have a policy in which community service of particular kinds is rewarded. If, for example, you start working on an assignment and you find a bug in the support code (we hope this won't happen, of course!), you can tell the instructor. Not only will they fix the bug, but also they'll reward you with some number of points for community service. If you find a bug and fix it, you get more points.

There are other kinds of community service as well: in some projects, you'll work with various sorts of data. If you created a viewer for datasets to help with debugging, and posted news about it (or gave your viewer to the instructor to put in the course bin directory that will be created later), you'd get points as well. If you have an improved Makefile for some project, that's a contribution as well.

Answering other students' questions will be considered a contribution as well.

Of course, it's not counted as community spirit if you post the solutions for an assignment to the course newsgroup. In general you should present community spirit contributions to the instructor before posting to the newsgroup.

Asking Questions

For asking your questions, we are going to use Google Groups (https://groups.google.com/forum/?hl=en&fromgroups#!forum/cmsc635spring20

13) to answer the majority of your questions. If you have a question whose answer that does not give away the key points of the assignment, please post it to the newsgroup. If in doubt, mail or ask the instructor and the faculty assistant. The instructor and the faculty assistant reserve the right to anonymously post questions that she receives and answers them in the newsgroup.

Example of an OK Newsgroup Question: "In my ray tracing I keep getting little black dots in my image. I've tried small test cases and know that my reflection model works. Any ideas?"

Answer: "Check your epsilon values." (Note that the answer did not say exactly what was wrong)

Bad Newsgroup Question: "I can't get specular highlighting to work. What's wrong?" (Or worse yet, "What's specular highlighting").

Really Bad Newsgroup Question: "Here's my rendering algorithm: . . . Does anyone think that I could work progressive refinement in for extra credit?"

Presentation Sections (TBD)

In sections, you will discuss and occasionally present an important graphic research paper to your fellow students. Sections are designed to expose you to a variety of final project ideas as well as give you the experience necessary to read graphics research papers with sophistication. Each person will be assigned to a section in the first week of class; attendance is required. You may not shift sections.

Section will run for about five and five and half weeks of the course; for each week of section there will be three graphics research papers discussed in sections. Each week we will randomly select three people who will each present one paper during the following week. Presenters for the first week of the presentation will be announced in class.

As a presenter you will prepare a short (about 20 minutes) presentation. You should also be prepared to take questions from fellow students after your presentation. You should attempt to thoroughly understand the paper you are presenting. Your presentation should outline the purpose and techniques used in the paper, list any problems that you feel the paper does not adequately address, and evaluation whether the paper presents its ideas clearly.

Whether you are presenting a paper or not you must fill out a paper evaluation form. The form and instructions will be available on the website during the first week of the class.

If you are presenting that week, you are only responsible for reading your paper. Everyone else must read all three papers and get a general understanding of each. Read each paper once, think about what parts seem a little unclear, and read the paper again.

The presenter is supposed to be the authority on the paper.

Lectures

Lectures are Mondays and Wednesdays, from 2:30-3:45 in the Janet & Walter Sondheim 112.. Sometime shortly after Spring Break, normal classes will end. In mid-April, there will be an intermediate demo where you will show the status of your final project to the rest of the class.

The VanGoph group'ers meet every Thursday at 4pm in the VanGoph lab ITE 352. You are invited to join the group.

We might have some visitors to present project ideas. The instructor will update that information with you.

We might find time for the paper presentation sections. I will work out a schedule if possible.

The final project demo day will be Monday May 7th. Detailed instructions will be online in the first week of the class. Attendance is mandatory. If you believe that you cannot attend, contact the instructor or the faculty assistant immediately.

Class Information

Almost all course information will be available on the Web (http://www.csee.umbc.edu/courses/graduate/635/spring13)

If there is information relevant to the class that does not yet appear on the web page, feel free to talk to the instructor or the faculty assistant, and they will consider placing it there.

You are responsible for knowing all the information in all articles posted in the newsgroup — read it frequently! (You're also responsible for all in-class announcements and material. "I missed that class" is not an excuse.)

Tentative Schedule

Here is a tentative schedule: (note: last day to Add/drop classes: Feb 8)

		Project	HWs	RDs
1/28	Missives		Math out	
1/30	Physics and measurement of light			
2/4	Light transport and scattering: scattering equation; rendering equation; review of raytracing	Ray out	Math due	
2/6	Software interface to scattering; transmission and attenuation; probability I		Render out	
2/11	Probability II; Monte Carlo integration		Render	

	and importance samples; path tracing overview / color?		due	
2/13	Path tracing details / color? Photon mapping overview	Ray due; Photon out	Photon out	
2/18	Photon mapping		Photon due	
2/20	Teddy introduction; advanced 3D interaction			
2/25	CrossShade	Photon due; Teddy out		
2/27	Some programming support: mesh; final project discussions			
3/4	As rigid as possible (ARAP) overview		Form final project group	
3/6	ARAP details	Teddy due; ARAP out	Final project groups due	
3/11	More ARAP (Laplacian coordinates??)			
3/13	Mesh deformation		Final project proposals due	
3/18-20	Spring break (Jian @ IEEE VR conference)	ARAP due		
3/25-27	VisWeek paper submission deadline – no class		Literature review due on the 27th	
4/1	Diffusion curves			
4/3	Octree textures			
4/8 – 5/7 Final projects meetings with Jian TBD: literature review; intermediate report and presentation				
4/8	Proposal presentation			
4/24	Intermediate demos			
5/7	Project demo and presentation			
5/12		Project paper due		

Special Accommodations:

Student requiring special accommodations should send a letter at the beginning of the semester to make arrangements.

Finally

CMSC 635 will be a continually evolving course. This is the instructor's first year at UMBC and first time teaching this course. As such, I am bound to have my own 'bugs' hiding in the corners. Please read everything handed out very carefully. If there is something which you do not understand, or which is not stated very clearly, please let me or the faculty assistant know so we can fix it.

Have fun!