Course Information

1 Staff
Instructor: Jovan Popović 32-D534
jovan@csail.mit.edu
Teaching Assistant: Eugene Hsu 32-D530
ehsu@csail.mit.edu
World Wide Web: http://stellar.mit.edu/S/course/6/sp08/6.839

2 Information
Advanced computer graphics will teach you how to represent geometry, motion, and appearance in complex three-dimensional scenes. Upon conclusion of this course, you will know how to apply computational methods to represent any scene, real or imagined. You will know how to analyze research publications in computer graphics. And you will know how to combine these methods to solve new challenges in computer graphics. We hope to inspire passion for computer graphics and its many applications, instill fearlessness needed to implement even the most sophisticated computational methods, and encourage lifelong learning from research publications.

3 Prerequisites
A strong understanding of programming and a solid background in linear algebra and multivariable calculus are necessary prerequisites to this course. You are expected to have taken Linear Algebra (18.006 or equivalent), Introduction to Algorithms (6.046J or equivalent). A computer graphics course such as 6.837 is strongly recommended for undergraduates. Advanced undergraduates and all graduate students need not have taken such a course in the past. If you do not meet these requirements, arrange to talk with the instructor.
4 Course Website

Please bookmark the course website:

http://courses.csail.mit.edu/6.839

This link will take you automatically to the Stellar Web site:

http://stellar.mit.edu/S/course/6/sp08/6.839,

which will contain electronic copies of handouts, assignments, and special announcements. Visit this site regularly to be aware of any changes in the course schedule, updates to your instructors’ office hours, etc. Major announcements will also be emailed to everyone in the class.

5 Topics

The detailed schedule for the course is available on the Web. These topics are divided into four parts:

Rendering. We formulate a general rendering equation for light transport and examine numerical solutions with Monte Carlo integration (stochastic ray tracing).

Animation. We formulate classical dynamics of constrained particles and rigid bodies and examine computational methods for efficient simulation and control.

Modeling. We derive subdivision curves and surfaces with a general and constructive approach. We also explore variational descriptions of non-rigid geometry and its control via inverse problems.

Publications. We analyze academic publications in computer graphics. For example, we review data-driven graphics and its fundamentals: memory-based learning and regression. We will also select additional topics according to student interest.

These topics will also lead us review the basics of multivariable calculus and linear algebra (linear transformations, change of bases, eigenanalysis and singular value decomposition) in the context of concrete computer graphics problems.

6 Lectures

Students are responsible for all material presented in the course. There are no required textbooks for this class. Students are expected to learn the material from lectures, readings, and handouts. All handouts will be available on the course web page in formats suitable for printing. The students should also take their own notes as lectures will often depend on the blackboard to convey the details and the information that is not adequately covered in other course material. Lectures will be held in Room 4–145 from 11 A.M. to 12:30 P.M. EST on Tuesdays and Thursdays.

Students without prior computer graphics experience will benefit from two more resources. The first book is 3D Computer Graphics: A Mathematical Introduction with OpenGL by Samuel
R. Buss. You do not need to purchase a hardcopy version of this book. There is a free online version available from Books24x7 (http://libraries.mit.edu/get/books24x7, you will need your MIT Personal Certificate). The book also has an associated web page that contains sample code and known errata. The second book is *The OpenGL Programming Guide* (also known as the red book). An earlier edition of this book (1.1) is available for free on the Web http://www.opengl.org/documentation/red_book_1.0/. The latest edition (1.4) is not available online, but everything that we will need in this course is addressed in the first edition.

7 Assignments

The course will include four graded assignments and several ungraded homeworks. The assignments develop the material that’s best learned through programming and implementation. They are often involved and require synthesis of more than one class topic. The fourth assignment is student-lead and allows you to extend one of the three core assignments or pursue another topic of interest.

The course calendar shows tentative due dates for all four assignments. You will be required to turn in your solutions and artifacts that demonstrate a working version of your program. Each student will also present their artifacts in short in-class presentations.

Late assignments will receive no credit but each assignment has a built-in grace period: between the turn-in date and the in-class presentation. However, if we can’t retrieve your artifact in time for the in-class presentation, you will receive no credit for that assignment. If there are extenuating circumstances, you should make prior arrangements with the instructor. An excuse from the Dean’s Office will be required if prior arrangements have not been made. You are responsible for ensuring that the correct solution is uploaded on time.

8 Grading Policy

The final grade will depend on the four graded assignments (80%), class participation (10%), and the final exam (10%). The grade for class participation will depend on your attendance, participation, and the discussion of reading materials. Note that you cannot earn a top grade without participating in the class.

9 Collaboration Policy

Collaboration on homeworks is encouraged, but you must write all solutions and programs on your own. If you are stuck and not able to make progress you should contact the teaching assistant during the office hours. If you collaborate with others on any part of the homework, you should identify your collaborators by name. If you did not collaborate with anyone then you should explicitly state so. If you obtain a solution through research (e.g., on the Web), cite the source, but write the solution or code on your own. For example, if you cannot explain the solution to a member of the course staff then you have violated our collaboration policy.
Collaboration is not permitted on the final exam. Students in violation of this collaboration policy will receive no credit for the assignment or exam and their actions will be reported to the Committee on Discipline. If you have any doubt about this collaboration policy, you should talk to the instructor immediately. If you’d like to learn more about academic integrity and the MIT policy, you can consult the MIT handbook for students


and the MIT libraries guide to avoiding plagiarism

http://libraries.mit.edu/tutorials/general/plagiarism.html