Sample syllabus - students received the detailed syllabus at the beginning of the term they are enrolled in the class

CS 445 Computational Photography

Instructor: **Derek Hoiem**

Overview

Computational photography is an emerging field created by the convergence of computer graphics, computer vision, and photography. Its role is to overcome the limitations of the traditional camera by using computational techniques to capture, enhance, and combine imagery for a more vivid and lifelike visual experience.

Course Objectives

By the end of this course, you will have written programs to create optical illusions, add or remove objects from a photograph, insert 3D objects into pictures, automatically stitch together photos into panoramas, and more. Through lectures and hands-on projects, you will learn core principles of computer vision and graphics that will be of great use in robotics, psychology, media design, art, photography, information retrieval, entertainment technology, and a host of other growing areas. Beyond the practical benefits, the course also aims to provide a greater appreciation of our own amazing visual ability and to have fun in writing programs that can be used with your own photo collections.

Prerequisites

You should enter the course with basic programming skills and a working knowledge of **linear algebra** and calculus. Previous experience with Python, image processing, computer vision, or computer graphics will be very helpful but is not required. Students are recommended to own or purchase a digital **camera**, ideally with manual controls (smart phone should be fine).

General Information

Textbook: Lectures are not based on any particular textbook. The most closely related textbook is Computer Vision: Algorithms and Applications by Rick Szeliski, which is available for free online and for purchase at Amazon. You may also want to purchase either Computer Vision by Linda Shapiro and George Stockman or Computer Vision: A Modern Approach (2nd edition) by David Forsyth and Jean Ponce. I have them both in my office, so you can look them through.

Other useful books:

Linear Algebra and its Applications, Gilbert Strang *(excellent book on linear algebra)*
Multiple View Geometry in Computer Vision, Hartley & Zisserman *(bible on recovering 3D geom.)*
Photography (8th edition), London and Upton, *(a great general guide to taking pictures)*
Vision Science: Photons to Phenomenology, Stephen Palmer *(great book on human perception)*
Digital Image Processing, 2nd edition, Gonzalez and Woods *(a good general image processing text)*
The Art and Science of Digital Compositing, Ron Brinkmann *(everything about compositing)*
3D Computer Graphics (3rd Edition), Watt *(a good general graphics text)*
Fundamentals of Computer Graphics, Peter Shirley *(another good general graphics text)*

To obtain **disability-related academic adjustments** and/or auxiliary aids, students with disabilities must contact the course instructor and the Disability Resources and Educational Services (DRES) as soon as possible. To contact DRES you may visit 1207 S. Oak St., Champaign, call 333-4603 (V/TDD), or e-mail a message to disability@uiuc.edu.

Assignments and Grading

Grading is based on projects, midterm, and the final project. The final letter grade should reflect practical and theoretical grasp of concepts introduced in class, as demonstrated through regular and final projects and the exam. Although class participation and other intangibles are not formally evaluated, I may consider them to raise (but not lower) the letter grade. I will guarantee an "A" with a 95% average and guarantee a "B" with an 85% average.

Projects (60%): Students are graded out of 600 points. Each project has a "core" component worth 100 points, and there are opportunities for hundreds of additional points in "bells and whistles". This gives students an opportunity to go more in depth in the projects most interesting to them. 630/600 points is the maximum that someone can get credit for.

Midterm Exam (10%): The midterm will cover materials up to and including the single view geometry lectures. Sample problems and answers will be provided.

Final exam (15%): The final exam will cover the entire semester.

Final project (15%): Do a final project of your choice. You could implement a paper that you find interesting, something discussed in class, a big extension of one of the existing projects, or something entirely of your own design. The deliverable is a 2 page abstract describing your project and the results. The scope of the project should be similar to those of the assigned projects (excluding the "hybrid image" project).

Team project (optional): If you want to use this course to satisfy the team project requirement, you can also enroll in CS 497 (1 credit) with me. You then will need to complete a larger project with a group of 3 or more and will have two additional milestones during the semester. The project should be about twice as big (or time consuming) as a standard project of the same team size.

Late policy: Aim to get all projects in on time to stay on track in the course. You have a total of seven free late days. Use them wisely. You have a short grace period for the submission deadline, e.g. a project submitted 15 minutes late will not count as late. Additional late days come at a penalty of ten points per day late. On-time projects will generally be graded more quickly than late projects.

Academic Integrity

You are welcome to discuss projects with your classmates, but do not show or share any code. Also, you may not use any code from the Internet or any other outside sources, unless it is specifically approved by the instructor. Be sure to acknowledge any help that you do get from other students or outside works, even if it's just a small suggestion. *Violations will go on record at the university, and the minimum penalty will be a zero for the entire assignment.*