

Computational Photography (CS 445) – Fall 2020

Instructor: **Derek Hoiem**

TAs:

- Zhi Chen (zhic4)
- Shengyu Fen (shengyu8)
- Eric Lee (kylee5)
- Dominic Roberts (djrbrts2)
- Yuan Shen (yshen47)

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). For the image-based lighting project, you may need a mirrored ball that can be purchased on Amazon.

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. Letter grades will be assigned based on the following thresholds:

97	94	90	87	84	80	77	74	70	67	64	60	<60
A+	A	A-	B+	B	B-	C+	C	C-	D+	D	D-	F

In summary, the point distribution is:

- Assigned Projects: 55%
- Exams: 30%
 - Midterm: 15%
 - Final: 15%
- Final project: 15%
 - Proposal: 1%
 - Report / code: 14%

Projects (55%): There are five projects, each with a “core” worth 100 points. Each project also has “bells and whistles” that give you the opportunity to explore projects of interest at greater depth and earn additional points. The number of required points depends on whether you are enrolled in the 3 credit undergraduate or 4 credit graduate version of the course:

- **3 credit: 425 points required**

- **4 credit: 500 points required**

You can earn **extra credit for up to 25 additional points**. So, for example, in the four credit version, if a student earns 540 points, her “Assigned Projects” portion of the grade will be 525/500. *Due to limitations in Coursera’s gradebook, extra points and late penalty will be tracked separately and will not be reflected in the grade shown in Coursera.* In final grade computation, it does not matter for which projects you earned the points or whether they are core or bells and whistles --- only the total matters. Projects that are not submitted within two weeks of the deadline will receive 0 points, but those points can be made up with bells & whistles from other projects if necessary.

Midterm Exam (15%): The midterm will cover materials up to and including the single view geometry lectures. The midterms and final exams will be administered on ProctorU using PrairieLearn. Sample questions 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 2-3 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). You can work with a group of up to three, but you are responsible for forming and managing the group. Details are in a separate document. The proposal is worth 1%, and the final submission is worth 14%.

Late policy: Aim to get all projects in on time to stay on track in the course. You have a total of ten free late days for regular projects. Use them wisely. Additional late days come at a penalty of five project points per day late. *To receive any credit, your project must be submitted within two weeks of the deadline, with no exceptions, regardless of whether you use free late days.* The reason for this is to ensure that the class is roughly in sync. You have a short grace period for the submission deadline, e.g. a project submitted less than one hour late will not count as late. The final project cannot be submitted late.

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.*