CSci 4968 and 6270
Computational Vision,
Fall Semester, 2011
Course Syllabus

Prof. Chuck Stewart
Mon, Thurs 10:00 - 11:30, Low 3045
Office: MRC 330B
Email: `stewart@cs.rpi.edu` or `cvstewart@gmail.com`
Phone: x6731
Hours: After class, Tue 1-3, by appointment, drop-in
URL: `http://www.cs.rpi.edu/~stewart/cv_2011`

Overview

Cameras and digital images are everywhere, with billions of cameras being used to take trillions of images, producing far more data than humans can possibly absorb. The result is a wide-spread need for algorithms and software that can combine, summarize and interpret these images, a need for the fields of computer vision and image analysis. Prototypical applications of computer vision include face, object and scene recognition, security and biometrics, photomontaging and virtual tours, special effects in graphics, photography and the cmovies, autonomous robots, human-computer interaction, and medical diagnosis and treatment. Given the exponential growth of digital devices, the potential for new applications seems unlimited.

In sharp contrast to this potential are the difficult challenges of computer vision. Each image is a large, noisy, quantized, two-dimensional array of intensity values. Each intensity value is created by light reflected off a surface sitting in the three-dimensional world, focused by a lens, digitized and recorded by a camera. Information is lost and noise is added at each step. To address these challenges, computer vision research scientists and applications programmers are employing a wide variety of physical, engineering, mathematical, statistical, algorithmic and software techniques to develop computer vision systems.

Our goal in this course will be to learn about the challenges, the techniques, and the applications of computer vision. Since this is a computer science course, much of our focus will be on computational aspects of the computer vision problem.

Learning Objectives

At the end of this course, each successful student will be able to

- Apply techniques of calculus and linear algebra to solve problems involved in building the components of a computer vision system.
- Develop efficient algorithms for solving problems in computer vision.
- Write small-sized and intermediate-sized programs for solving problems in computer vision.
- Map potential applications of computer vision into specific technical problems.

- Assess the difficulty of specific technical problems in computer vision and select potential solution techniques.

Prerequisites

Students should have had courses in programming, in data structures and in algorithms (e.g. CSci 2300). Mathematical background should include a course in multivariable calculus and linear algebra (Math 2100). This requirement is somewhat flexible since some students have done well in previous semesters without this background. In addition, we will be discussing the some of the necessary mathematical techniques as we proceed through the semester. Students may find my mathematical methods lectures notes helpful:

`http://www.cs.rpi.edu/~stewart/math_techniques`

Requirements

Student grades will be determined by the following

- 10% — class participation
- 60% — five homework assignments,
- 30% — semester project

Letter grades will be determined based on the rounded, combined averages. Final cut-offs will be at the instructors discretion, but will be no lower than 92 for an A, 89 for an A-, 86 for a B+, 82 for a B, etc.

Class participation will be measured in two ways. The first is the usual notion of asking questions and contributing to the discussion in class. The second will be to review lecture notes to pose follow-up questions about (a) ideas or techniques that were not clear from the just-completed lecture, (b) the utility of a technique described in lecture, or (c) the relationship between a discussed method and related techniques that have been covered in class. Students will be assigned to pose these questions on a rotating basis, with advanced warning in all cases. Questions about one lecture must be emailed to the instructor by 7 am on the morning of the next lecture (e.g. 7 am Thursday after a Monday lecture or 7 am Monday after a Thursday lecture). The questions will be judged at the level of 0, 1, or 2 (with 2 being the most insightful), and the most significant will be used in completing the preparation for that day's class.

Class attendance is expected and will figure into the participation grade. Use of laptops in class for anything other than note taking is strongly discouraged — it is distracting for everyone in the class.

Homework and Programming

Homework will involve solving mathematical problems, developing algorithms, writing programs, and analyzing results. The mixture of these will vary between assignments, although the programming aspect and resulting analysis will tend to be the most important.

Programming can be done in Matlab with the Image Processing Toolkit, in Python, in Java, in C, or C++ using OpenCV or VXL. If you use one of the latter, you will need to include some way of examining images and other results. My strong suggestion for this course is use of Matlab. If you continue with computer vision beyond this course, especially for working in industry, you should eventually transition to writing in C, C++ or Java.

Some homework assignments must be done individually, while others may be done in teams of two.

Final Project

The final project is a worth 30% of the semester grade. Each student will choose a problem or application of computer vision, study the problem carefully, learn about relevant techniques, and then implement, test and evaluate a solution.

There will be multiple submission dates for the project, starting in mid November. These will include a proposal and proposal discussion, a progress report, a software demonstration, and a final report. Points will be accumulated for these submissions, with the last two having the most weight. Students may work individually or in a team of two.

Late Policy

Students have three "late" days they can use on homework throughout the semester, with at most two used for any one homework. A late day is defined as any whole or partial day after the submission deadline. After all late days have been used, no late homework submission will be accepted without an excuse from the Dean of Students office. Late days may not be used for any part of the final project.

You do not need to use late days for personal emergencies. Just get an excuse from the Dean of Students office and then together we can arrange a suitable time to complete any missed work.

Undergraduate vs. Graduate Level

This course is a combined undergraduate and graduate course. One obvious difference is that the undergraduate course is worth four credits, whereas the graduate course is worth three. In addition, there will be some differences in the homework assignments and in the expectations on the final project. These will be made clear as we proceed through the semester.

Textbook

There is no official textbook for this course, although we will be making heavy use of Dr. Rick Szeliski's on-line draft of his textbook. An up-to-date reference will be provided on the course website. Additional references will be provided there to other on-line and printed material.

Academic Integrity

Here are the guidelines for academic integrity on the various types of assignments:

- On assignments that require individual work, students are expected to submit their own solutions and write-ups. Students may work together to understand problem requirements, sketch solution ideas, and discuss results. Implementations and write-ups must be individual.

- On assignments that allow students to work together, the work should be shared equally, and each student in the group will be responsible for understanding all aspects of the assignment and solution. Discuss between groups is allowed on the same level as discussion between individuals for single-person assignments.

- Each individual or group will be working on a different final project. Therefore, discussion between individuals and groups is greatly encouraged. Within a group, once again, all work must be shared, and each individual is responsible for understanding all aspects of the project. Students who find resources to help with the final project must (a) reference these resources in their write-up and (b) add to these with their own work; each individual or team will be graded on the work that is its own.

Topic Schedule

Here is a summary of the topics we will be covering during the semester.

Topic	**Lecture**	**Date(s)**
Introduction	1	8/29
Images and image processing	2-3	9/1-9/8
Feature extraction	4-5	9/12-9/15
Segmentation	6-8	9/19-9/26
Transforms & cameras	9-11	9/29-10/6
SIFT	12-13	10/11 - 10/13
Calibration	14	10/17
Image mosaics	15-16	10/20 - 10/24
Intro. to recognition	17	10/27
Classifiers & PCA	18-19	10/31-11/3
Instance recognition	20-21	11/7-11/10
Object detection	22-23	11/14-11/17
Category recognition	24	11/21
Motion & structure	25-26	11/28-12/1
Stereo	27	12/5
Catch-up / presentations	28	12/8

Submission Schedule

Finally, here is a **tentative** schedule of homework submission dates.

Assignment	**Topic**	**Due Date**
HW 1	Image processing	9/19
HW 2	Feature extraction	9/29
HW 3	Motion segmentation	10/13
HW 4	Image mosaics	10/31
FP 1	Final project proposal	11/17
HW 5	Recognition	11/21
FP 2	Final project progress report	12/1
FP 3	Final project demo and report	12/14

The final project reports will be due by 5 pm on Wednesday December 14. Earlier that day and on the 13th students will have 20 minute demo and discussion meetings with the professor.