CSCI 507 / EENG 507 Syllabus
Introduction to Computer Vision

Instructor

Associate Professor William A. Hoff, Room BB249 (Brown Hall)
Phone: 303-273-3761 Email: whoff@mines.edu
Office hours: M 1-3, W 1-2. At other times, if I'm not having a meeting, please feel free to stop by.

Meetings

- Class is held every Monday and Wednesday in BB316A from 4:30 pm - 5:45 pm.
- Occasionally there are outside talks and seminars related to this course, which I encourage you to attend (see http://eecs.mines.edu/eecs.php/eventsAndNews/calendar).

Web Sites

- The main course website is http://eecs.mines.edu/Courses/csci507/. Visit this often, as it will contain lecture slides, homework assignments and solutions, labs, grades, etc.
- Please use Piazza to ask questions about the material – this is often the quickest way to get help. Go to http://piazza.com/ to sign up.

Description (from Bulletin)

Computer vision is the process of using computers to acquire images, transform images, and extract symbolic descriptions from images. This course provides an introduction to this field, covering topics in image formation, feature extraction, location estimation, and object recognition. Design ability and hands-on projects will be emphasized, using popular software tools. The course will be of interest both to those who want to learn more about the subject and to those who just want to use computer imaging techniques.

Prerequisites

Students should have undergraduate level knowledge of linear algebra (i.e., vectors, matrix operations, solution of linear equations), elementary statistics, and a programming language (e.g., MATLAB and/or C++).

Follow-on Course

This is course is the pre-requisite to a follow-on course called "Advanced Topics in Computer Vision" (CSCI508/EENG508), which covers topics such as depth image processing, category object recognition, and machine learning for computer vision applications.

Objectives

- To understand the fundamental concepts, problems, and solution techniques in computer vision, including image formation, structure and motion estimation, and object recognition.
- To apply computer vision techniques to solve common problems in research and industrial applications, such as image transformations, inspection, and recognition.
- To learn the use of image processing and image understanding software tools.

Computer Tools

Computer tools will be used frequently in class and for assignments. We will use MATLAB (with the image processing and computer vision toolboxes) most often, since it is interactive, easy to write code, and there is a lot of existing software available. MATLAB is installed on the PCs in Brown Hall. You will need to get an "adit" logon for these if you don't already have it – see the Computing Center website. We will also occasionally use the Open Source Computer Vision library (OpenCV), which is free from http://opencv.org/. This is a collection of algorithms written in C/C++ for various computer vision problems.

Textbooks

- The required textbook is *Computer Vision: Algorithms and Applications*, by Richard Szeliski, Springer 2011. It is available electronically through the school's library at http://link.springer.com/book/10.1007%2F978-1-84882-935-0. You must be on campus or logged in via VPN to download the book.
- Optional texts:
  - *Digital Image Processing, 3nd ed,*, by Gonzalez and Woods, Prentice Hall, 2008. This is the book used in the image processing class. It is a thorough and readable coverage of image processing techniques that are useful in computer vision.
  - There are a bunch of recent books on OpenCV (see http://opencv.org/books.html for a complete list).

Course Format

This course will be run in a partially "flipped" mode. On the indicated days, students are expected to go over the course material in advance, and class time will be used primarily for working through problems and examples. For this to work, students must come prepared to class.

Assessment and Grading

Students will be assessed with using the following elements.

- Weekly quizzes (20%). On most Mondays, there will be a short (open book) quiz at the beginning of class, covering the material from last week.
- Lab (10%). On most Wednesday s, there will be a hands-on lab assignment to be done in class, in teams of two. The lab assignment will be checked for completion in class and must be shown to the instructor to receive credit.
- Homework assignments (40%). There will be a series of homework assignments, to be done individually. Homework is due by the beginning of class on the due date. Late assignments will be reduced in grade unless there is prior approval by the instructor.
- Final project (30%). A final project will be done in teams of two. See the course website for additional details.

There are no exams.

Students taking the 498 version of the class will be given the same quizzes, assignments, and projects, as those taking the 507 version. However, the quality and/or quantity of the work expected is higher for 507 than for 498, and thus will be graded more strictly.

Using Computers in Class

It's ok to follow along the lecture slides on the computer. However, checking your email, working on other non-class related materials, web-surfing, etc., are not appropriate activities for class time. It's a huge distraction, not only to you but to the people around you. Please be respectful of your colleagues in class, and use the computers only for class activities.

Collaboration Policy for Programming Projects in CS Courses

The following policy exists for all CS courses in the EECS department. This policy is a minimum standard; your instructor may decide to augment this policy.

- If the project is an individual effort project, you are not allowed to give code you have developed to another student or use code provided by another student. If the project is a group project, you are only allowed to share code with your group members.
- You are encouraged to discuss programming projects with other students in the class, as long as the following rules are followed:
    - You view another student's code only for the purpose of offering/receiving debugging assistance.
    - Students can only give advice on what problems to look for; they cannot debug your code for you.
    - All changes to your code must be made by you.
    - Your discussion is subject to the empty hands policy, which means you leave the discussion without any record [electronic, mechanical, or otherwise] of the discussion.
- Any material from any outside source such as books, projects, and in particular, from the Web, should be properly referenced and should only be used if specifically allowed for the assignment.
- To prevent unintended sharing, any code stored in a hosted repository (e.g. on GitHub) must be private. For group projects, your team members may, of course, be collaborators.
- If you are aware of students violating this policy, you are encouraged to inform the professor of the course. Violating this policy will be treated as an academic misconduct for all students involved. See the Student Handbook for details on academic dishonesty.

Collaboration Policy for Homework

The following policy applies to homework assignments other than programming projects.

- You can discuss homework assignments with other students in the class, as long as the following rules are followed:
    - You view another student's work only for the purpose of offering/receiving assistance.
    - All work must be done by you.
    - Your discussion is subject to the empty hands policy, which means you leave the discussion without any record [electronic, mechanical, or otherwise] of the discussion.
- Any material from any outside source such as books, projects, and in particular, from the Web, should be properly referenced and should only be used if specifically allowed for the assignment.