Robert Collins
CSE586

CSE 586, Spring 2011
Computer Vision II

Course Introduction

Robert Collins
CSE586

Course Goals

Gain practical knowledge in Computer Vision

Develop skills for being a successful researcher

Robert Collins
CSE586

Course Layout

Typically, this course is organized by vision topics
e.g.

- Stereo
- Tracking
- Recognition
- Segmentation

We are going to do it a little differently

Robert Collins
CSE586

Course Layout

We will organize based on mathematical methods used
e.g.

- Mixture Models / EM
- Graph Cuts
- Graphical Models
- Monte Carlo Approaches

Underlying vision info is the same, but we are indexing it differently and refocusing our effort towards really understanding the underlying solution techniques.

Robert Collins
CSE586

Why Doing it This Way?

By focusing on solutions rather than problems, we hope to gain:

- practical knowledge about solving vision problems
- intuition about when to apply each method
- competency in using the techniques

Potential drawback:
This is going to be a very mathematical course.

Robert Collins
CSE586

Course Layout

To address development of research skills, each student will be an active learner

- Implementing useful programs
- Reading primary source material (conferences, journals)
- Locating relevant papers on the web
- Writing critical analyses of papers read
- Oral presentation, ~~both formal (using projector in front of the class) and~~ informal (class discussion of papers)

Why Doing it This Way?

By focusing on active learning, we hope to gain:

- better practical understanding
- practice research skills
- help you jumpstart your own research

Potential drawback:
This course has a heavy student workload

Course Goals

Gain practical knowledge in Computer Vision
- focusing on solution methods
- understanding the underlying math
- knowing when/how to apply methods

Develop skills for being a successful researcher
- programming
- critical thinking
- locating and reading research papers
- technical writing
- presentation skills / rational debate

Sample Topics

- Mixture Models and EM
- Monte Carlo Methods
- Graphical Models
- "Sparse" Methods
- Graph Cuts / Spectral Methods
- Procrustes Analysis
- Subspace Methods
- Variational Calculus

We surely won't cover all of these. Some of them alone are suitable for a whole semester course.

How Course is Conducted

Course is broken into "units" lasting two or more weeks.

Each unit focuses on one particular solution method (e.g. EM algorithm).

I will give some lectures at the start of each unit to introduce the basic mathematical approach.

During the unit, we will implement sample code using that solution method, and read/discuss papers that are examples of using that solution method. The papers and examples will typically span different areas of computer vision.

Homework

We will have frequent homework that is a combination of "theory" and programming. We will implement many practical sample programs to reinforce learning the mathematical method being discussed. It should become second nature to you to sit down an "bang out" some simple test code to see if you understand an idea, to verify some result you computed analytically, or to generate a graphical visualization to gain more insight into a problem.

I would suggest using matlab (or octave) because it is easier to program for the kinds of things we will be doing, but you can use any language you want.

Reading/Critiques

During each unit we will each read a set of assigned papers that provide examples of using the method to solve a vision problem. You will write a short critique evaluating each assigned paper, identifying its pros and cons, and comparing and contrasting how it uses the method we are discussing. Each person will also find another related paper on their own to read and critique. After discussing all assigned papers we will go around the room and have people tell what the paper they found independently is about, why they selected that paper, and whether they recommend it or not.

This whole exercise is meant to develop skills in research, reviewing, and communication.

Final Project

We will have a final term project that is meant to be a non-trivial exploration of the use of one of the methods we have discussed to some problem and dataset that is of interest to you. The hope is that you will find something we learn in this class useful in your own research, and this project is a way to get started on that. Projects can be done either individually or in groups (I would expect a group project to be larger in scope than a single person project).

Exams

None.

Reading/writing/discussing/programming will keep you busy enough!

Grading

- *Homework:* *50%*
- *Written Critiques* *20%*
- *Final Project:* 20%
- *Class participation:* 10%

Academic Integrity: Although you are encouraged to talk to each other to understand the course material and assignment instructions, when it comes time to doing the assignments, every student is expected to submit their own original work. For programming, standard and publicly available code libraries (such as simple signal processing or linear algebra libraries) may be used after seeking consent of the course instructor or TA.