Robert Collins
CSE586

CSE 586, Spring 2010
Computer Vision II

Course Introduction

Course Goals

Gain practical knowledge in Computer Vision

Develop skills for being a successful researcher

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

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.

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.

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

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.

How Course is Conducted

Each student will work on sample code (perhaps working in teams if you wish) implementing the method, and trying it out on some simple problems.

We will each read a set of assigned papers that provide examples of using the method to solve a vision problem.

One student will present each assigned paper. These will be short 20 minute (conference-style talks, so really only 15 minutes + 5 for questions). The idea is that you will be presenting the paper "at a conference", as if it was your own work.

How Course is Conducted

During a week you are not presenting, each student will write a short critique evaluating each assigned paper in addition to one of their own selection (see below), comparing and contrasting the papers' use of the method we are discussing.

Students will find another related paper on their own to read and critique. After discussing all assigned papers (which may take one or more class periods), 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.

Programming

We will implement practical sample programs to reinforce learning the mathematical method being discussed. By having a short programming assignment each class, we will incrementally (and painlessly) build up nontrivial programs that do useful things, so that at the end of each unit, we will have a project that demonstrates the code on some typical vision problem (either one of the sample ideas I will give, or one that you think of yourself).

You can program individually or in teams, in any programming language that you like.

Exams

None.

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

Grading

- *Written Critiques:* 30%
- *Oral Presentations:* 20%
- *Programming/Projects:* 40%
- *Class participation:* 10%