SCHOOL OF COMPUTING
THE UNIVERSITY OF UTAH

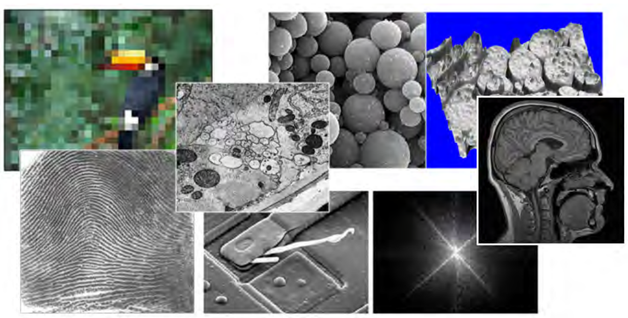
SYLLABUS CS-BIOEN 6640 FALL 2014

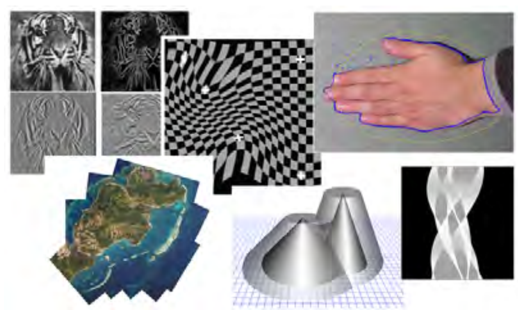
Introduction to Digital Image Processing

CS Catalog Number 6640, jointly listed as BIOEN 6640
University of Utah, Fall Semester 2014
M,W 1.25 - 2.45 WEB 1230

Guido Gerig (gerig@sci.utah.edu)

Class web-page (sample from 2012):
`http://www.sci.utah.edu/~gerig/CS6640-F2012/CS6640-F2012.html`





Goal and Objectives:

This is an introductory course in processing grey-scale and color images — taught at the graduate level. This course will cover both mathematical fundamentals and implementation. It will introduce students to the basic principles of processing digital signals and how those principles apply to images. These fundamentals will include sampling theory, transforms in appearance and geometry, filtering and object segmentation. The course will also cover a series of basic image-processing problems including enhancement, reconstruction, segmentation, feature detection, and compression. Assignments will include several projects with software implementations and analysis of real data, and with emphasis on a project report with summary of the approach and critical assessment of results and experiments.

- To introduce the fundamental problems of digital image processing (DIP).
- To introduce the main concepts and techniques used to solve those.
- To enable participants to implement solutions for reasonably complex problems, and to apply those to own images.
- To enable participants to understand basic DIP methodology that is discussed in the image processing literature, and applications of DIP across a broad range of scientific disciplines.
- To motivate students to experience that DIP is a field where signal processing algorithms and methods become visual. Beside helping scientists to use imaging as a scientific instruments, this also lead to attractive and creative results.

Who should attend this course?

Graduate students who are interested in learning the fundamental concepts of DIP or desire to use image processing techniques in their research. Research in image processing is closely related to computer vision and computer graphics, but also most essential in all areas where researchers make use of imaging technology (digital images, 2D and 3D scanners (CT,MRI), microscopy, range imaging and more) and need to *extract quantitative information* from those digital image data.

Prerequisites:

Most of the knowledge required should be part of the standard undergraduate/graduate background in Computer Science, Bioengineering and Electrical and Civil Engineering, including undergraduate/graduate Mathematics, Algorithms, and Programming. Students with different background and curriculum need to discuss suitability and options with the teacher.

Learning approach

- Students should read the relevant chapters of the books and/or reading assignments before the class.
- In the course, the material will then be discussed in detail on the board and via slides and motivated with real world examples and applications.
- There will be assignments with theoretical & programming questions to provide students with practical experience of the techniques. We will also do a few short quizzes in class to evaluate the level of understanding.
  - *"Learning by Doing"*: Image Processing is particularly attractive to experience algorithms and methods since one immediately *sees what is done by displaying results via images and graphing image-derived quantitative information.*
  - "Creativity"*: Projects will be designed to motivate and encourage students to become creative by going further than the minimal requirements and exploring algorithms on own pictures and/or images taken from their research projects.*

– "Project Report": *While programming a working solution based on the course materials is a first important task, reporting on the solution strategy and demonstrating results and experiments will be seen at least as important as the coding. The practical part should be written with a text system and should include equations, hints to the programming solution, outline of solution strategy, graphs and results, and a critical discussion of results and eventual obstacles.*

- There will also be a final project where students will solve a real world problem using DIP techniques. Students will be able from a few projects proposed by the instructor.

Programming:

This course will make use of MATLAB for assignments, and the TA will be able to support students with Matlab questions. Matlab is installed and accessible on computers of the Utah COE Cade lab, can be accessed remotely (although slow) or purchased as individual copy at the student store. Students can also make use of C++ or Java for projects but will have to be self-supporting w.r.t. programming. For Matlab novices, we will give additional introductory lectures for Matlab use, and documentation, test programs and other materials will be available for download.

Grading

Weighted contribution of projects and exams to final grade:

Projects/Assignments (5)	55%*
Midterm:	15% (on paper, theoretical questions)
Final project (programming/report)	20%
Class participation/quizzes	10%**

Final project is required to get a passing grade

* Late policy on projects: 10% grade deduction per late day, assignments will no more be accepted after 3 days.

** Attendance to lectures is required to be eligible for office hours by the instructor and the TA (unless absence is officially documented). Attendance and regular active participation is required to get the participation grade. Absences count as unexcused absences unless students present documents signed by officials (e.g. medical certificate), with advanced notice to course instructor and TA.

		87-89	B+	77-79	C+	67-69	D+		
93-100	A	83-86	B	73-76	C	63-66	D	0-59	E
90-92	A-	80-82	B-	70-72	C-	60-62	D-		

Overview of the Course

- Introduction to Probability and Images: Images, Points, Functions

- Point Operations: Histogram analysis, and mapping of intensity distributions.
- Filtering with Neighborhoods: Linear Filtering
- Filtering with Neighborhoods: Nonlinear Filtering
- Object detection by template matching
- Edge and Line Detector (Canny optimal detector)
- Fourier Transforms and Filtering of image data
- Geometric Transformations and Warping
- Image Mosaicing/Stitching
- Grouping of pixels to structures: Hough Transform (HT)
- Generalized Hough Transform (GHT)
- Deformable model segmentation: *Snakes*
- Mathematical morphology (binary): Erosion, Dilation, Opening, Closing
- Mathematical morphology (gray level)

Web-Page, Textbook and Handouts

We will use the UofU web-based canvas system for organization of this course. Handouts, assignments, and student solutions will all be organized electronically via web-based download and upload of documents. Canvas will also be used to communicate grading to students.

The traditional textbook for this course is "Digital Image Processing, 3rd Edition , Rafael C. Gonzalez and Richard E. Woods, Prentice Hall" `http://www.imageprocessingplace.com/DIP-3E/dip3e_main_page.htm`, and a majority of lectures will follow materials from this book. Additional material (instructor slides, scientific papers, chapters from other textbooks etc.) will be distributed during the course and made available via the course web-site on the UofU Canvas system.

Organization

Teaching:	Guido Gerig email: gerig@sci.utah.edu Tel: (801) 585-0327
Lecture time and place:	M,W 1.25 - 2.45 / room WEB 1230
Office hours:	tbd, office tbd
TA:	tbd tbd TA Office hours: tbd room: tbd.
Material:	Book "Digital Image Processing, 3rd Edition , Rafael C. Gonzalez and Richard E. Woods, Prentice Hall" `http://www.imageprocessingplace.com/DIP-3E/dip3e_main_page.htm`. and handouts on WebCT
Assignments:	Theorectial questions, programming work with Matlab (or ev. other languages on own responsibility), practical examples with sample images and images generated by students using own cameras.

Honor Policy

Students are expected to work *on their own*, as instructed by the Professor. Students may discuss projects with other individuals either in the class or outside the class, but they may not copy code from others or receive code or results electronically from any other source. *Students must write their own code, conduct their own experiments, write their own reports, and take their own tests.* Any use of sources (for projects or tests) that are not specifically given to the student by the Professor or TA, must be discussed with the Professor or TA or documented in the report. Any student who is found to be violating this policy will be given a failing grade for the course and will be reported to the authorities as described in the University's Student Code (Code of Student Rights and Responsibilities) (`http://regulations.utah.edu/academics/6-400.php`).

Some useful links

- The Computer Vision Homepage (`http://www-2.cs.cmu.edu/afs/cs/project/cil/ftp/html/vision.html`)
- CVonline (`http://www.dai.ed.ac.uk/CVonline/`)
- CVonline Wikipedia (`http://en.wikipedia.org/wiki/CVonline`)

Draft syllabus, subject to change, Guido Gerig August 24, 2014