CS855: Data Visualization

International Institute of Information Technology, Bangalore

Term I 2014-15

General Course Information

Course Name	Data Visualization
Instructors	Prof. Jaya Sreevalsan-Nair jnair@iiitb.ac.in
Course Credits (Level)	4 (Level-2)
Grading	9-point scale (A, A-, B+, B, B-, C+, C, D, F)
Pre-requisites	GEN501: Mathematics for IT (specifically the module on "Linear algebra & Matrix theory"), CS501: Algorithms, Working knowledge of programming (C++/OpenGL), Working knowledge of numerical analysis, calculus, and matrix operations, preferrably CS606: Computer Graphics, preferrably GEN601: Introduction to Scientific Computing.

Course Overview

This course is a graduate-level course for which the goal is "to provide students with concepts and a firm mathematical foundation, as well as technical aspects of algorithms. Practical skills in programming visualization algorithms, using commercial visualization tools, and applying methodologies and techniques to new problems are taught in accompanying exercises." – this is as stated in "Curriculum for a Course on Scientific Visualization," a peer-reviewed paper by Rotard et. al in Proceedings of Eurographics/ACM Siggraph Workshop on Computer Graphics Education, in 2004, and can be extended to information visualization as well.

In short, this course will cover techniques and evaluation of visualizations of scientific and information data. The outcome of this course is to bring up the knowledge and practice of visualization in students to

requirements of a graduate level course. The lectures cover the areas of scientific and information visualization.

Course Contents

1. Topics from textbook "Data Visualization: Principles and Practice,":
   (a) Introduction to visualization and data, including data types; graphics and visualization pipelines (3 lectures).
   (b) Scientific visualization techniques (icon-based, topology-based, etc.)
      - Scalar field visualization (2 lectures),
      - Vector field visualization (2 lectures),
      - Tensor field visualization (2 lecture),
      - Surface and volume visualization (3 lectures).
   (c) Information visualization techniques: tables, networks, matrices, multivariate, text, etc. and visual analytics (5 lectures).
   (d) Evaluating visualizations and introduction to visualization open-source applications such as paraview, vtk, etc. (3 lectures).
2. Supplementary material from textbook "Mathematical Principles for Scientific Computing and Visualization":
   (a) Computing with Multivariate Functions
   (b) Facets
   (c) Background: Computer Graphics
3. Additionally, the students will be informed of new directions of research in the visualization community towards visual analytics. (5 lectures)

Grading

- Assignments: 75% (Individual breakdown given in the assignment booklet).
- Midterm: 15%
- Final: 10%

Specifics on the assessment:

- **Academic Plagiarism:** This course has zero tolerance for cheating and plagiarism. Any violation may result in an F grade and further disciplinary action may be initiated as per the Institute's policies. Ignorance of what constitutes cheating and plagiarism is not an excuse! If you have any doubts, contact your instructor. All material that will be used for assessment of the student's performance must be original work.

- **Assignments:** An assignment booklet will be provided to the students in the first class which will give a complete description and logistics of all the assignments for the course. There will be biweekly homework sets and four programming assignments. "*Start early and finish on time*" is the guiding principle for all assignments in this course.

- Rules on implementation:

  - Regarding the choice of paper for the reading assignment, no two students can work on the same paper. Hence, *first-come, first-served* policy is undertaken for the assignment of papers to the sudents. All students have to inform the instructor on the choice of the paper via e-mail. If students do not own up a paper by instructed deadline, a paper will be randomly assigned to them on the day.
  - All programming assignments should be implemented as per the instructions provided in the assignment booklet.
  - All programming assignments should be implemented on a Linux OS, preferrably Ubuntu or Fedora.

- Rules on submissions:

  - The list of submission deadlines will be provided in the assignment booklet.
  - Several submissions can be made for the programming assignments and the technical report. Only the last submission will be considered for evaluation.
  - To incentivize early submissions and discourage late submissions the following bonus scheme will be used on the total for final grade:
    - $+1$ for submission before the designated Sunday, -0.5 for submission on the subsequent Monday or Tuesday, -1 for submission before the next Sunday, -3 any later.
  - All submissions must be sent to the instructor via e-mail.
  - Submissions should be named in the format: <RollNumber>_Assignment<Number>.* where * is tar.gz or pdf
  - Submission for a programming assignment would be a tarred-gzipped folder comprising of the source files, header files, README, subfolder containing screenshots and a Makefile.
    - There will be penalty for submissions containing intermediate files (e.g., *.o, *.C , etc.).
    - README files should contain information on sources referred to for help on the assignment, instructions on how to compile and run the application, expected input-outputs, and any notable defects/effects when running the application.
  - Submission for a technical report will be a single .pdf file.
    - Reports should be written using LaTeX and IEEE style file.
    - The technical report should follow the style of an IEEE conference paper where the body of the text should include the following sections: Abstract, Introduction, Related work, Background of the assigned paper, Discussions, and Conclusions, and References. The technical report should be original work. The report should include the findings of the assigned paper, its influence on the graphics community, and its consequences in terms of its succeeding techniques, algorithms, and/or applications.

    - A class presentation of the report will need to be done, which will also be used for evaluating the reading assignment.

- **Examinations:** There will be a written examination during the Mid-term examination week and a viva voce for finals.
  - The examinations will be based on the topics covered in the class until the date of the examination and the viva voce may also include questions based on the assignments.
  - The viva voce will be conducted by two examiners, including the primary instructor. The examination will be conducted for 15 minutes per student. Each student will be allocated a time-slot for the examination day.
- **Evaluations:** All programming assignments will be evaluated based on a demonstration and code-review. The final version of all programming assignments have to be demonstrated to the instructor before 4pm on the following Monday after the deadline. The technical report will be reviewed by two reviewers, including the primary instructor.

References/Reading Material

Textbooks:

1. Alexandru C. Telea, "Data Visualization: Principles and Practice," A. K. Peters Ltd, 2008.
2. Gerald Farin, Dianne Hansford, "Mathematical Principles for Scientific Computing and Visualization," A. K. Peters Ltd., 2008.
3. Tamara Munzner, "Visualization Analysis and Design," A. K. Peters Ltd, to appear October 2014. (preprint-draft-4 will be used: http://www.cs.ubc.ca/~tmm/courses/533/book/)