Project Topic Ideas

Rendering (15-862)
24 Oct. 1995
-
Paul Heckbert

The goal of the project is to give you practice at doing research in the field of rendering. Most projects will involve implementing an algorithm, but some could be more theory-oriented or survey-like. You should (1) do literature search on related previous work, (2) do your new work, (3) write up a description of your work, with results (preferably beautiful pictures), and comparison of your results to previous work, and (4) give a presentation in class. Papers should be 8-12 pages in length. The "new work" of item (2) above could be either design and implementation of a new algorithm, novel theoretical analysis of an algorithm, or a thorough survey and taxonomy of previous work on some topic.

SCHEDULE:

- **Meet with me to discuss your project idea: 26 or 27 Oct.**
- **Turn in one-page proposal of what you plan to do: 31 Oct.** in class.
- Presentations: 30 Nov., 5 Dec., 7 Dec. (2-3 per class meeting)
- Final report due: 8 Dec.

The following are some possible project topics. The list is not exclusive! You can also modify an idea listed here, or propose something totally new. If you have other ideas, please talk with me.

VISIBILITY ALGORITHMS

Analyze computational complexity of realistic image synthesis algorithm(s): To talk meaningfully about the computational complexity of a realistic image synthesis algorithm, you need a "realism" or "accuracy" standard. Propose a simple accuracy measure and use it to analyze, theoretically or empirically global illumination algorithms on simplified scenes (perhaps 2-D?).

Propose a z-buffer based shadow algorithm that simulates penumbra: Can it be done? [14, 9, 1]

ANTIALIASING

Implement and compare several adaptive stochastic sampling methods: [6, 7] (and others)

Implement antialiased Ghostscript: The Ghostscript program (gs) is wonderful in most respects, but it doesn't do antialiasing. Modify its source code to compute and use 3 bit per pixel antialiased fonts. (For serious hackers only).

IMAGE-BASED MODELS

Interactive projective image mosaics: Use hardware supported fast texture mapping on a graphics workstation to create an interactive tool for laying out image mosaics in which the images differ by a projective transformation (e.g. panoramas). User would grab a point on image and pull it about with direct manipulation [3].

Registration for image mosaics: Implement a nonlinear optimization method to refine an initial (manual) registration for an image mosaic. One such optimization method is called Levenberg-Marquardt [12]. This project could be a team effort with the previous.

Generate Quicktime VR from image mosaics: Generate a cylindrical panorama image in the Quicktime VR file format, from an image mosaic (talk to Harry Shum).

Solve the hole problem in Chen-Williams' view interpolation algorithm. (good luck!)

VISUALIZATION

Visualize Surface Curvature: Write software to display the two sets of contours of principal curvature on an arbitrary parametric or implicit surface. Make some beautiful pictures of curvature on ellipsoids, tori, blobby surfaces, and teapots. [vanWijk,Hilbert,Koenderink]

Survey volume rendering algorithms: Write up a good survey of volume rendering algorithms, helping to make sense of the wide variety of them. (this will involve a lot of reading).

LOCAL ILLUMINATION

Compare reflection models: Compare some subset of Phong, Beckmann-Spizzichino, Cook-Torrance, He, or others empirically or theoretically.

GLOBAL ILLUMINATION

Simulate participating media (fog): using either zonal (radiosity) or Monte Carlo ray tracing. [10]

Implement distribution ray tracing and compare: Compare the speed and accuracy of distribution ray tracing as discussed in [5] with [13]. You'll have to implement the former, but you can get code for the latter.

Visualize a radiosity algorithm: Using plots of radiance across each surface and perhaps visualizations of the matrix solution process, explain visually how a 3-D radiosity algorithm works (in a progressive algorithm, you might show how light "bounces" around the scene; in a matrix radiosity algorithm, you might display eigenfunctions). Depict artifacts and errors of the approximation visually.

Parallel algorithms: Propose a parallel algorithm for realistic image synthesis that simulates interreflection and will work for complex scenes (i.e. does not require replication of the scene in each processor's memory).

Implement specular radiosity: Choose one of the specular models (Cook-Torrance, He, etc.) and fit it with spherical harmonics. Examine accuracy/speed tradeoff (more spherical harmonic terms allows greater accuracy, but increases cost). [11]

Analyze radiosity algorithm with matrix analysis: Use matrix analysis to analyze the convergence and errors of matrix radiosity methods and progressive radiosity methods. [4]

MISCELLANEOUS

Do texture mapping on general surfaces: Implement software to allow parameterization of and texture mapping on implicit surfaces [8] or the instancing of geometry (e.g. tree bark tiles) [2] on complex surfaces.

References

[1] Lynne Shapiro Brotman and Norman I. Badler. Generating soft shadows with a depth buffer algorithm. *IEEE Computer Graphics and Applications*, 4(10):5–24, Oct. 1984.

[2] Kurt Fleischer. In *SIGGRAPH '95 Proc.*, 1995.

[3] Michael Gleicher and Andrew Witkin. Through-the-lens camera control. *Computer Graphics*, 26(2):331–340, 1992. Proc. Siggraph '92.

[4] Steven Gortler, Michael F. Cohen, and Philipp Slusallek. Radiosity and relaxation methods: Progressive refinement is southwell relaxation. Technical report, Computer Science Dept., Princeton University, Feb. 1993. CS-TR-408-93.

[5] James T. Kajiya. The rendering equation. *Computer Graphics (SIGGRAPH '86 Proceedings)*, 20(4):143–150, Aug. 1986.

[6] Don P. Mitchell. Generating antialiased images at low sampling densities. *Computer Graphics (SIGGRAPH '87 Proceedings)*, 21(4):65–72, July 1987.

[7] James Painter and Kenneth Sloan. Antialiased ray tracing by adaptive progressive refinement. *Computer Graphics (SIGGRAPH '89 Proceedings)*, 23(3):281–288, July 1989.

[8] Hans Pedersen. In *SIGGRAPH '95 Proc.*, 1995.

[9] William T. Reeves, David H. Salesin, and Robert L. Cook. Rendering antialiased shadows with depth maps. *Computer Graphics (SIGGRAPH '87 Proceedings)*, 21(4):283–291, July 1987.

[10] Holly E. Rushmeier and Kenneth E. Torrance. The zonal method for calculating light intensities in the presence of a participating medium. *Computer Graphics (SIGGRAPH '87 Proceedings)*, 21(4):293–302, July 1987.

[11] François X. Sillion, James R. Arvo, Stephen H. Westin, and Donald P. Greenberg. A global illumination solution for general reflectance distributions. *Computer Graphics (SIGGRAPH '91 Proceedings)*, 25(4):187–196, July 1991.

[12] Richard Szeliski and James Coughlan. Spline-based image registration. Technical report, DEC Cambridge Research Lab, April 1994. TR 94/1, URL=http://www.research.digital.com/CRL/publications/crl-rr.html.

[13] Gregory J. Ward, Francis M. Rubinstein, and Robert D. Clear. A ray tracing solution for diffuse interreflection. *Computer Graphics (SIGGRAPH '88 Proceedings)*, 22(4):85–92, Aug. 1988.

[14] Lance Williams. Casting curved shadows on curved surfaces. *Computer Graphics (SIGGRAPH '78 Proceedings)*, 12(3):270–274, Aug. 1978.