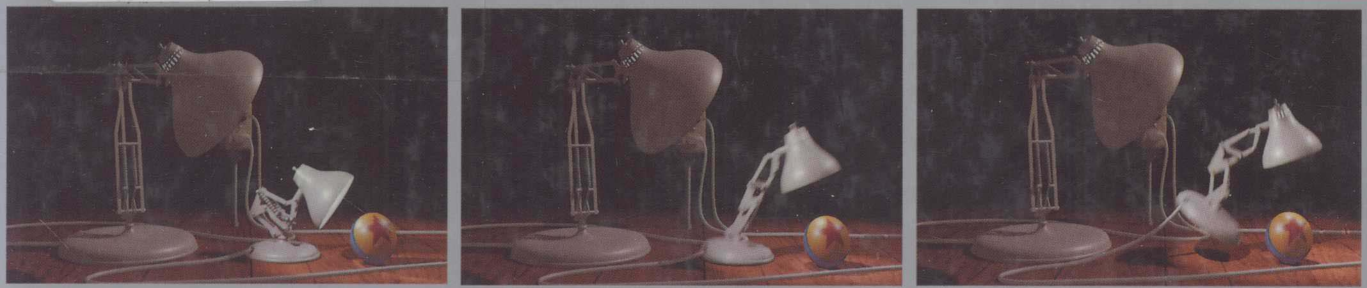
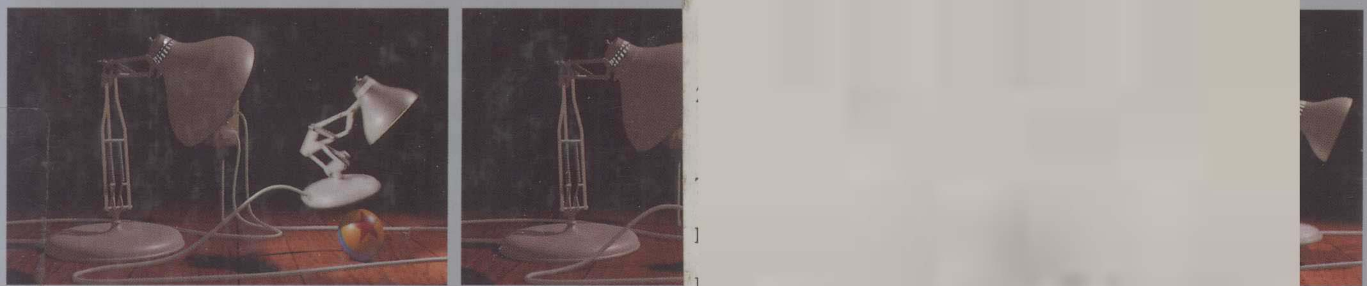
INTRODUCTION TO



COMPUTER GRAPHICS



FOLEY • VAN DAM • FEINER • HUGHES • PHILLIPS

Introduction to Computer Graphics

James D. Foley
Georgia Institute of Technology

Andries van Dam
Brown University

Steven K. Feiner
Columbia University

John F. Hughes
Brown University

Richard L. Phillips
Los Alamos National Laboratory and The University of Michigan

ADDISON-WESLEY PUBLISHING COMPANY

Reading, Massachusetts • Menlo Park, California • New York
Don Mills, Ontario • Wokingham, England • Amsterdam • Bonn
Sydney • Singapore • Tokyo • Madrid • San Juan • Milan • Paris

Sponsoring Editor *Peter S. Gordon*
Senior Production Supervisor *Jim Rigney*
Copy Editors *Lyn Dupré and Joyce Grandy*
Text Designer *Sandra Rigney*
Technical Art Consultant *Joseph K. Vetere*
Illustrators *C&C Associates and Tech Graphics*
Cover Designer *Eileen Hoff*
Senior Manufacturing Manager *Roy Logan*
Marketing Manager *Robert Donegan*

This book is an abridged and modified version of *Computer Graphics: Principles and Practice,* Second Edition, by Foley, van Dam, Feiner, and Hughes, published in 1990 in the Addison-Wesley Systems Programming Series, IBM Editorial Board, consulting editors.

Library of Congress Cataloging-in-Publication data

Introduction to computer graphics / James D. Foley . . . [et al.] .].
p. cm.
Includes bibliographical references and index.
ISBN 0-201-60921-5
1. Computer graphics. I. Foley, James D., 1942– .
T385.I538 1993
006.6—dc20 93-16677
CIP

Reprinted with corrections.

3 4 5 6 7 8 9 10 MA 989796959493

Preface

This book is an adaptation of *Computer Graphics: Principles and Practice*, Second Edition (*CGPP*), by Foley, van Dam, Feiner, and Hughes. *Introduction to Computer Graphics* was created by abridging and modifying that comprehensive teaching and reference work to suit the needs of different courses and different professional requirements. While this book is half the size of its parent, it is not merely a shorter version of it. Indeed, it features new material and, in some cases, a different approach to exposition, all added with the needs of its intended audience in mind.

This book is designed to be used in a one- to two-semester course in computer graphics in any four-year college or university and, assuming only a small amount of mathematical preparation, for a one-semester course in community colleges or other two-year institutions. *Introduction to Computer Graphics* is also an ideal book for the professional who wants to learn the rudiments of this dynamic and exciting field, whether to become a practitioner or simply to gain an appreciation of the far-ranging applications of computer graphics.

This book is not meant to supplant *CGPP* as being more current or in any way more comprehensive. There are chapters, however, where, because of the dizzying pace at which the field is moving, older material was dropped and hardware performance and cost figures were updated. One such example can be found in Chapter 4, where the statement from *CGPP*—which, bear in mind was just published in 1990—that "… a graphics workstation typically consists of a CPU capable of executing at least several million instructions per second (MIPS) …" was updated to reflect the fact that 20–100 MIPS are now commonplace.

Other major differences and strengths of *Introduction to Computer Graphics* are:

- The computer language used throughout the book, both in pseudocoded program fragments and complete working programs, is modern ANSI C. The use of C, rather than Pascal as in *CGPP*, is consistent with both current teaching and professional practice, especially in graphics.
- As a direct benefit of the use of C in the book, there is now a one-to-one correspondence between the data types and functions of the code used in this book with those of the SRGP and SPHIGS software packages that are available (free of charge to adopters) to accompany the book (see page 559).
- The SPHIGS package mentioned above has been substantially enhanced with many new features, such as multiple light sources, improved rendering, and improved pick correlation for better interactive manipulation.
- The book features several worked-out examples, some of which are quite extensive. These examples are strategically located in chapters where they best enhance the exposition of difficult concepts. One such example is a complete working program for interactively defining Bézier parametric cubic curves.
- The importance of computer graphics to the emerging field of multimedia is introduced by describing some examples, complete with figures, and providing their supporting references.
- A mathematical preliminaries section has been added to the chapter on Geometrical Transformations. This section provides sufficient information for the reader to understand and use all subsequent mathematically oriented material in the book.

Potential Syllabi

There are several paths that a reader can take through this book. A few are suggested here, but it is entirely feasible to select one suiting the reader's circumstances. Even the order of study can be permuted. For example, the material on hardware could come either earlier or later in a syllabus than is suggested by Chapter 4's ordinal positioning.

A minimal one-semester course emphasizing 2D graphics. Where the goal is to provide a good, but not rigorous, overview of elements of mostly 2D graphics, this course of study will be appropriate for students in a two- or four-year college or university.

Chapter	Sections
1	All
2	Sect. 2.1–2.2
3	Sect. 3.1–3.3
4	Sect. 4.1, 4.2, 4.3, and 4.5
5	Sect. 5.1 (as appropriate), 5.2, 5.3, 5.4

Chapter	Sections
6	Sect. 6.1, 6.2, 6.3, 6.4.1, 6.4.2
8	All
9	Sect. 9.1, 9.2.1–9.2.3
11	Sect. 11.1–11.2
12	Selected reading to demonstrate advanced capabilities

A one-semester course providing an overview of 2D and 3D graphics. This syllabus will provide a strong founding in graphics for readers who are mathematically well prepared.

Chapter	Sections
1	All
2	All
3	Sects. 3.1–3.5, 3.10, 3.12, 3.15
4	Sect. 4.1, 4.2, 4.3, and 4.5
5	Sect. 5.1 (as appropriate), 5.2–5.5, 5.7, 5.8
6	Sect. 6.1–6.5, Sect. 6.6 (except 6.6.4), 6.7
7	Sect. 7.1–7.4, 7.10
8	All
9	Sect. 9.1, 9.2.1–9.2.3, 9.3.1–9.3.2
11	All
12	All
13	Sect. 13.1–13.2
14	Sect. 14.1–14.2

A two-semester course covering 2D and 3D graphics, modeling, and rendering. All chapters (possibly omitting selected topics from Chapters 9 and 10) plus selected topics from *CGPP*.

Since many readers of *Introduction to Computer Graphics* will be interested in consulting its more advanced and comprehensive parent, the preface to *CGPP* follows this one. There the reader will find a discussion of *CGPP*'s important features and suggestions for structuring courses based on that book.

Acknowledgments

First, it should be stated that while all the authors of *CGPP* participated to some degree in the preparation of this book, I assume full responsibility for any new errors introduced in the adaptation process.

David Sklar was a guest author for *CGPP*, and much of the material he contributed to that book remains here in Chapters 2 and 7. He was also helpful to me in locating electronic versions of computer code and artwork from the original book.

Peter Gordon, my editor, always had timely, wise, and calming advice throughout the duration of this project. Jim Rigney, my production supervisor, spent lots of helpful hours teaching me the "tricks of the trade."

Many people helped with various aspects of the book. Among them are Yvonne Martinez, Chuck Hansen, Tom Rokicki, David Cortesi, Janick (J.) Bergeron, Henry McGilton, Greg Cockroft, Mike Hawley, Ed Catmull, Loren Carpenter, Harold Borkin, Alan Paeth, and Jim White.

A special thank you is owed to Ed Angel of The University of New Mexico and his courageous students for beta—make that alpha—testing of a first draft of the book in the fall of 1992.

Finally, without D.C. this would never have happened.

Santa Fe, N.M. R.L.P.

Preface to Computer Graphics: Principles and Practice, Second Edition

> Interactive graphics is a field whose time has come. Until recently it was an esoteric specialty involving expensive display hardware, substantial computer resources, and idiosyncratic software. In the last few years, however, it has benefited from the steady and sometimes even spectacular reduction in the hardware price/performance ratio (e.g., personal computers for home or office with their standard graphics terminals), and from the development of high-level, device-independent graphics packages that help make graphics programming rational and straightforward. Interactive graphics is now finally ready to fulfill its promise to provide us with pictorial communication and thus to become a major facilitator of man/machine interaction. (From preface, *Fundamentals of Interactive Computer Graphics*, James Foley and Andries van Dam, 1982)

This assertion that computer graphics had finally arrived was made before the revolution in computer culture sparked by Apple's Macintosh and the IBM PC and its clones. Now even preschool children are comfortable with interactive-graphics techniques, such as the desktop metaphor for window manipulation and menu and icon selection with a mouse. Graphics-based user interfaces have made productive users of neophytes, and the desk without its graphics computer is increasingly rare.

At the same time that interactive graphics has become common in user interfaces and visualization of data and objects, the rendering of 3D objects has become dramatically more realistic, as evidenced by the ubiquitous computer-generated commercials and movie special effects. Techniques that were experimental in the early eighties are now standard practice, and more remarkable "photorealistic" effects are around the corner. The simpler kinds of pseudorealism, which took hours of computer time per image in the early eighties, now are done routinely at animation rates (ten or more frames/second) on personal computers. Thus "real-time" vector displays in 1981 showed moving wire-frame objects made of tens of thousands of vectors without hidden-edge removal; in 1990 real-time raster displays can show not only the same kinds of line drawings but also moving objects composed of as many as one hundred thousand triangles rendered with Gouraud or Phong shading and specular highlights and with full hidden-surface removal. The highest-performance systems provide real-time texture mapping, antialiasing, atmospheric attenuation for fog and haze, and other advanced effects.

Graphics software standards have also advanced significantly since our first edition. The SIGGRAPH Core '79 package, on which the first edition's SGP package was based, has all but disappeared, along with direct-view storage tube and refresh vector displays. The much more powerful PHIGS package, supporting storage and editing of structure hierarchy, has become an official ANSI and ISO standard, and it is widely available for real-time geometric graphics in scientific and engineering applications, along with PHIGS+, which supports lighting, shading, curves, and surfaces. Official graphics standards complement lower-level, more efficient de facto standards, such as Apple's QuickDraw, X Window System's Xlib 2D integer raster graphics package, and Silicon Graphics' GL 3D library. Also widely available are implementations of Pixar's RenderMan interface for photorealistic rendering and PostScript interpreters for hardcopy page and screen image description. Better graphics software has been used to make dramatic improvements in the "look and feel" of user interfaces, and we may expect increasing use of 3D effects, both for aesthetic reasons and for providing new metaphors for organizing and presenting, and navigating through information.

Perhaps the most important new movement in graphics is the increasing concern for modeling objects, not just for creating their pictures. Furthermore, interest is growing in describing the time-varying geometry and behavior of 3D objects. Thus graphics is increasingly concerned with simulation, animation, and a "back to physics" movement in both modeling and rendering in order to create objects that look and behave as realistically as possible.

As the tools and capabilities available become more and more sophisticated and complex, we need to be able to apply them effectively. Rendering is no longer the bottleneck. Therefore researchers are beginning to apply artificial-intelligence techniques to assist in the design of object models, in motion planning, and in the layout of effective 2D and 3D graphical presentations.

Today the frontiers of graphics are moving very rapidly, and a text that sets out to be a standard reference work must periodically be updated and expanded. This book is almost a total rewrite of the *Fundamentals of Interactive Computer Graphics,* and although this second edition contains nearly double the original 623 pages, we remain painfully aware of how much material we have been forced to omit.

Major differences from the first edition include the following:

- The vector-graphics orientation is replaced by a raster orientation.
- The simple 2D floating-point graphics package (SGP) is replaced by two packages—SRGP and SPHIGS—that reflect the two major schools of interactive graphics programming. SRGP combines features of the QuickDraw and Xlib 2D integer raster graphics packages. SPHIGS, based on PHIGS, provides the fundamental features of a 3D floating-point package with hierarchical display lists. We explain how to do applications programming in each of these packages and show how to implement the basic clipping, scan-conversion, viewing, and display list traversal algorithms that underlie these systems.
- User-interface issues are discussed at considerable length, both for 2D desktop metaphors and for 3D interaction devices.
- Coverage of modeling is expanded to include NURB (nonuniform rational B-spline) curves and surfaces, a chapter on solid modeling, and a chapter on advanced modeling techniques, such as physically based modeling, procedural models, fractals, L-grammar systems, and particle systems.

- Increased coverage of rendering includes a detailed treatment of antialiasing and greatly expanded chapters on visible-surface determination, illumination, and shading, including physically based illumination models, ray tracing, and radiosity.
- Material is added on advanced raster graphics architectures and algorithms, including clipping and scan-conversion of complex primitives and simple image-processing operations, such as compositing.
- A brief introduction to animation is added.

This text can be used by those without prior background in graphics and only some background in Pascal programming, basic data structures and algorithms, computer architecture, and simple linear algebra. An appendix reviews the necessary mathematical foundations. The book covers enough material for a full-year course, but is partitioned into groups to make selective coverage possible. The reader, therefore, can progress through a carefully designed sequence of units, starting with simple, generally applicable fundamentals and ending with more complex and specialized subjects.

Basic Group

Chapter 1 provides a historical perspective and some fundamental issues in hardware, software, and applications. Chapters 2 and 3 describe, respectively, the use and the implementation of SRGP, a simple 2D integer graphics package. Chapter 4 introduces graphics hardware, including some hints about how to use hardware in implementing the operations described in the preceding chapters. The next two chapters, 5 and 6, introduce the ideas of transformations in the plane and 3-space, representations by matrices, the use of homogeneous coordinates to unify linear and affine transformations, and the description of 3D views, including the transformations from arbitrary view volumes to canonical view volumes. Finally, Chapter 7 introduces SPHIGS, a 3D floating-point hierarchical graphics package that is a simplified version of the PHIGS standard, and describes its use in some basic modeling operations. Chapter 7 also discusses the advantages and disadvantages of the hierarchy available in PHIGS and the structure of applications that use this graphics package.

User Interface Group

Chapters 8-10 describe the current technology of interaction devices and then address the higher-level issues in user-interface design. Various popular user-interface paradigms are described and critiqued. In the final chapter user-interface software, such as window managers, interaction technique-libraries, and user-interface management systems, is addressed.

Model Definition Group

The first two modeling chapters, 11 and 12, describe the current technologies used in geometric modeling: the representation of curves and surfaces by parametric functions, especially cubic splines, and the representation of solids by various techniques, including boundary representations and CSG models. Chapter 13 introduces the human color-vision system, various color-description systems, and conversion from one to another. This chapter also briefly addresses rules for the effective use of color.

Image Synthesis Group

Chapter 14, the first in a four-chapter sequence, describes the quest for realism from the earliest vector drawings to state-of-the-art shaded graphics. The artifacts caused by aliasing are of crucial concern in raster graphics, and this chapter discusses their causes and cures in considerable detail by introducing the Fourier transform and convolution. Chapter 15 describes a variety of strategies for visible-surface determination in enough detail to allow the reader to implement some of the most important ones. Illumination and shading algorithms are covered in detail in Chapter 16. The early part of this chapter discusses algorithms most commonly found in current hardware, while the remainder treats texture, shadows, transparency, reflections, physically based illumination models, ray tracing, and radiosity methods. The last chapter in this group, Chapter 17, describes both image manipulations, such as scaling, shearing, and rotating pixmaps, and image storage techniques, including various image-compression schemes.

Advanced Techniques Group

The last four chapters give an overview of the current state of the art (a moving target, of course). Chapter 18 describes advanced graphics hardware used in high-end commercial and research machines; this chapter was contributed by Steven Molnar and Henry Fuchs, authorities on high-performance graphics architectures. Chapter 19 describes the complex raster algorithms used for such tasks as scan-converting arbitrary conics, generating antialiased text, and implementing page-description languages, such as PostScript. The final two chapters survey some of the most important techniques in the fields of high-level modeling and computer animation.

The first two groups cover only elementary material and thus can be used for a basic course at the undergraduate level. A follow-on course can then use the more advanced chapters. Alternatively, instructors can assemble customized courses by picking chapters out of the various groups.

For example, a course designed to introduce students to primarily 2D graphics would include Chapters 1 and 2, simple scan conversion and clipping from Chapter 3, a technology overview with emphasis on raster architectures and interaction devices from Chapter 4, homogeneous mathematics from Chapter 5, and

3D viewing only from a "how to use it" point of view from Sections 6.1 to 6.3. The User Interface Group, Chapters 8-10, would be followed by selected introductory sections and simple algorithms from the Image Synthesis Group, Chapters 14, 15, and 16.

A one-course general overview of graphics would include Chapters 1 and 2, basic algorithms from Chapter 3, raster architectures and interaction devices from Chapter 4, Chapter 5, and most of Chapters 6 and 7 on viewing and SPHIGS. The second half of the course would include sections on modeling from Chapters 11 and 13, on image synthesis from Chapters 14, 15, and 16, and on advanced modeling from Chapter 20 to give breadth of coverage in these slightly more advanced areas.

A course emphasizing 3D modeling and rendering would start with Chapter 3 sections on scan converting, clipping of lines and polygons, and introducing antialiasing. The course would then progress to Chapters 5 and 6 on the basic mathematics of transformations and viewing, Chapter 13 on color, and then cover the key Chapters 14, 15, and 16 in the Image Synthesis Group. Coverage would be rounded off by selections in surface and solid modeling, Chapter 20 on advanced modeling, and Chapter 21 on animation from the Advanced Techniques Group.

Graphics Packages

The SRGP and SPHIGS graphics packages, designed by David Sklar, coauthor of the two chapters on these packages, are available from the publisher for the IBM PC (ISBN 0-201-54700-7), the Macintosh (ISBN 0-201-54701-5), and UNIX workstations running X11, as are many of the algorithms for scan conversion, clipping, and viewing.

Acknowledgments

This book could not have been produced without the dedicated work and the indulgence of many friends and colleagues. We acknowledge here our debt to those who have contributed significantly to one or more chapters; many others have helped by commenting on individual chapters, and we are grateful to them as well. We regret any inadvertent omissions. Katrina Avery and Lyn Dupré did a superb job of editing. Additional valuable editing on multiple versions of multiple chapters was provided by Debbie van Dam, Melissa Gold, and Clare Campbell. We are especially grateful to our production supervisor, Bette Aaronson, our art director, Joe Vetere, and our editor, Keith Wollman, not only for their great help in producing the book, but also for their patience and good humor under admittedly adverse circumstances—if we ever made a promised deadline during these frantic five years, we can't remember it!

Computer graphics has become too complex for even a team of four main authors and three guest authors to be expert in all areas. We relied on colleagues and students to amplify our knowledge, catch our mistakes and provide constructive criticism of form and content. We take full responsibility for any remaining sins of omission and commission. Detailed technical readings on one or more chapters were provided by John Airey, Kurt Akeley, Tom Banchoff, Brian Barsky, David

Bates, Cliff Beshers, Gary Bishop, Peter Bono, Marvin Bunker, Bill Buxton, Edward Chang, Norman Chin, Michael F. Cohen, William Cowan, John Dennis, Tom Dewald, Scott Draves, Steve Drucker, Tom Duff, Richard Economy, David Ellsworth, Nick England, Jerry Farrell, Robin Forrest, Alain Fournier, Alan Freiden, Christina Gibbs, Melissa Gold, Mark Green, Cathleen Greenberg, Margaret Hagen, Griff Hamlin, Pat Hanrahan, John Heidema, Rob Jacob, Abid Kamran, Mike Kappel, Henry Kaufman, Karen Kendler, David Kurlander, David Laidlaw, Keith Lantz, Hsien-Che Lee, Aaron Marcus, Nelson Max, Deborah Mayhew, Barbara Meier, Gary Meyer, Jim Michener, Jakob Nielsen, Mark Nodine, Randy Pausch, Ari Requicha, David Rosenthal, David Salesin, Hanan Samet, James Sanford, James Sargent, Robin Schaufler, Robert Scheifler, John Schnizlein, Michael Shantzis, Ben Shneiderman, Ken Shoemake, Judith Schrier, John Sibert, Dave Simons, Jonathan Steinhart, Maureen Stone, Paul Strauss, Seth Tager, Peter Tanner, Brice Tebbs, Ben Trumbore, Yi Tso, Greg Turk, Jeff Vroom, Colin Ware, Gary Watkins, Chuck Weger, Kevin Weiler, Turner Whitted, George Wolberg, and Larry Wolff.

Several colleagues, including Jack Bresenham, Brian Barsky, Jerry Van Aken, Dilip DaSilva (who suggested the uniform midpoint treatment of Chapter 3) and Don Hatfield, not only read chapters closely but also provided detailed suggestions on algorithms.

Welcome word-processing relief was provided by Katrina Avery, Barbara Britten, Clare Campbell, Tina Cantor, Joyce Cavatoni, Louisa Hogan, Jenni Rodda, and Debbie van Dam. Drawings for Chapters 1–3 were ably created by Dan Robbins, Scott Snibbe, Tina Cantor, and Clare Campbell. Figure and image sequences created for this book were provided by Beth Cobb, David Kurlander, Allen Paeth, and George Wolberg (with assistance from Peter Karp). Plates II.21–37, showing a progression of rendering techniques, were designed and rendered at Pixar by Thomas Williams and H.B. Siegel, under the direction of M.W. Mantle, using Pixar's PhotoRealistic RenderMan software. Thanks to Industrial Light & Magic for the use of their laser scanner to create Plates II.24–37, and to Norman Chin for computing vertex normals for Color Plates II.30–32. L. Lu and Carles Castellsagué wrote programs to make figures.

Jeff Vogel implemented the algorithms of Chapter 3, and he and Atul Butte verified the code in Chapters 2 and 7. David Sklar wrote the Mac and X11 implementations of SRGP and SPHIGS with help from Ron Balsys, Scott Boyajian, Atul Butte, Alex Contovounesios, and Scott Draves. Randy Pausch and his students ported the packages to the PC environment.

We have installed an automated electronic mail server to allow our readers to obtain machine-readable copies of many of the algorithms, suggest exercises, report errors in the text and in SRGP/SPHIGS, and obtain errata lists for the text and software. Send email to "graphtext @ cs.brown.edu" with a Subject line of "Help" to receive the current list of available services.

Atlanta, GA	J.D.F.
Providence, RI	A.v.D.
New York, NY	S.K.F.
Providence, RI	J.F.H.

About the Authors

James Foley (Ph.D., University of Michigan) is founder and Director of the Graphics, Visualization & Usability Center at Georgia Institute of Technology, where he is also Professor of Computer Science and Professor of Electrical Engineering. Coauthor with Andries van Dam of *Fundamentals of Interactive Computer Graphics*, Dr. Foley is a member of ACM, ACM SIGGRAPH, ACM SIGCHI, the Human Factors Society, IEEE, and the IEEE Computer Society. He is editor-in-chief of *ACM Transactions on Graphics*, and serves on the editorial boards of *Computers and Graphics* and *Presence*. His research interests are UIDE (the User Interface Design Environment, a model-based user interface development tool), user interface software, information visualization, multimedia, and human factors of the user interface. Dr. Foley is a Fellow of the IEEE, and a member of Phi Beta Kappa, Tau Beta Pi, and Sigma Xi.

Andries van Dam (Ph.D., University of Pennsylvania) was one of the founders and the first chairman of the computer science department at Brown University. He is currently L. Herbert Ballou University Professor and professor of computer science at Brown, and senior consulting scientist at BLOC Development and Electronic Book Technologies, and on Technical Advisory Boards for ShoGraphics and Microsoft. A member of the IEEE Computer Society and ACM, he is also cofounder of ACM SIGGRAPH. Dr. van Dam helped found and was an editor of *Computer Graphics and Image Processing* and of *ACM Transactions on Graphics*. Coauthor of the widely used book *Fundamentals of Interactive Computer Graphics* with James Foley and of *Pascal on the Macintosh: A Graphical Approach* with David Niguidula, he has, in addition, published over eighty papers. In 1984 he received an IEEE Centennial Medal; in 1988 the state of Rhode Island Governor's Science and Technology Award; in 1990, the NCGA Academic Award; and in 1991, the SIGGRAPH Steven A. Coons Award. Dr. van Dam's research interests include hypermedia, electronic books, and high-performance workstations for teaching and research.

Steven Feiner (Ph.D., Brown University) is associate professor of computer science at Columbia University where he directs the computer graphics and user interface laboratory. His current research focuses on picture synthesis, applications of artificial intelligence to computer graphics, user interfaces, animation, virtual worlds, and hypermedia systems. He is also involved with the development of knowledge-based systems that automate the design and layout of graphical user interfaces. Dr. Feiner is on the editorial boards of *Electronic Publishing* and *ACM Transactions on Information Systems*, and is a member of ACM SIGGRAPH and the IEEE Computer Society. In 1991 he received an ONR Young Investigator Award. Dr. Feiner's work has been published in over forty papers and presented in numerous tutorials and panels.

John Hughes (Ph.D., University of California, Berkeley) is an assistant professor of computer science at Brown University where he codirects the computer graphics group with Andries van Dam. His research interests include the application of mathematics to computer graphics for scientific and mathematical visualization, automated computer animation, foundations of computer graphics, and interactive illustration. He is a member of ACM SIGGRAPH and the IEEE Computer Society. Recent papers have appeared in *Computer Graphics* and *Topology*, and his work on sphere eversions was described in a cover article for *Science News*.

Richard Phillips (Ph.D., University of Michigan) is principally responsible for this adaptation of *CGPP*. Dr. Phillips is Professor Emeritus of Electrical and Computer Engineering, and Aerospace Engineering at the University of Michigan. There, as founding director of the Computer Aided Engineering Network, he was instrumental in establishing a several hundred node workstation network for student and faculty use in the College of Engineering. He was also founding director of the Center for Information Technology Integration. He is currently a technical staff member at Los Alamos National Laboratory, where his current research interests are scientific visualization, multimedia workstations, distributed computing, and multimedia digital publication. He is a member of IEEE and ACM and serves on the editorial board of *Computers and Graphics*.

Contents