Lecture Notes in Computer Science

Edited by G. Goos and J. Hartmanis

421

Tamiya Onodera
Satoru Kawai

A Formal Model of Visualization in Computer Graphics Systems

Springer-Verlag
Berlin Heidelberg New York London Paris Tokyo Hong Kong

Authors

Tamiya Onodera
IBM Research, Tokyo Research Laboratory
5-19, Sanban-cho, Chiyoda-ku
Tokyo 102, Japan

Satoru Kawai
Department of Graphics and Computer Science
The College of Arts and Sciences, The University of Tokyo
3-8-1, Komaba, Meguro-ku
Tokyo 153, Japan

CR Subject Classification (1987): I.3.4

ISBN 3-540-52395-2 Springer-Verlag Berlin Heidelberg New York
ISBN 0-387-52395-2 Springer-Verlag New York Berlin Heidelberg

Printing and binding: Druckhaus Beltz, Hemsbach/Bergstr.
2145/3140-543210 – Printed on acid-free paper

Preface

The field of computer graphics has expanded rapidly during the last decade. It provides a comprehensive way for computers and human beings to communicate, and is indispensable in modern computer systems. In order to facilitate the building of such systems, the kernel systems of computer graphics are designed and implemented to provide programmers with a set of fundamental graphics functions for producing pictures.

Visualization refers to the total process of transforming graphical objects from system-specified representations into their final representations. Unfortunately, visualization in conventional kernel systems is only poorly described: the specification documents are incredibly thick and most of the essential parts are written in natural language, relying on the incomplete framework of a viewing pipeline. The model of a viewing pipeline is insufficient, since it models only the viewing process, ignoring the important process of attribute rendering of graphical objects, and does not define the central concept of graphical output primitives. Moreover, the diversity of standards and proposed standards by the ISO for kernel systems embarrasses the computer graphics community, since we do not have a model on which two standards can be compared. All of these problems motivate the construction of a general, concise, and formal model of visualization.

The main purpose of this book is to present a formal model of visualization called a *visualizing net*. We give the model in a unified and axiomatic manner based on the mathematical concept of correspondence, which is a general form of mapping. Surprisingly, various attribute rendering processes, including pattern filling, are described there in a uniform way, and graphical output primitives are defined as mappings from finite generable objects. A visualizing net is a concise model that not only allows an unambiguous and comprehensive description of systems but can also be the center of the documentation. In addition, a visualizing net is far more general than a viewing pipeline. This implies that all visualizations in conventional kernel systems can be described by means of a visualizing net: we do so in Section 3.3.

This book is primarily intended for students or researchers interested in the formal aspects of computer graphics, which have been paid little attention by most researchers. One reason is that the research done so far in this field is not closely linked to actual systems.

We try in this book to link the model to reality. That is, we present the design and implementation of a kernel system called a *graphics nucleus*, which provides the full functionality of a visualizing net. The system is based on one general and expressive drawing technique, called *geometry-driven* drawing, which is the consequence of our abstraction of attribute rendition. For instance, it can draw roads or railways as simply as dotted lines.

Kernel system designers can also benefit from our work, since it contains a discussion of what a kernel system should be and expresses a view on how to avoid building complex, hopeless systems. Chapter 2 will be helpful for those who are interested in correspondence, since it contains in-depth discussions on this subject, which has hitherto received far less attention than mapping or functions. Familiarity with computer graphics kernel systems and with first-order logic is assumed.

Acknowledgements

We are thankful to Messrs. T. Kamada, S. Matsuoka, H. Morishima, and C. Lee of the Kawai Laboratory of the University of Tokyo for their helpful discussions and valuable criticisms. We would also like to express my gratitude to Mr. H. Yokouchi of IBM Research, Tokyo Research Laboratory for his mathematical suggestions. Finally, We are grateful to Mr. McDonald of IBM Research, Tokyo Research Laboratory for his careful proof-reading.

Tokyo, January 1990

Tamiya Onodera

Satoru Kawai

Contents

List of Figures

List of Tables