Computer Graphics

PRINCIPLES AND PRACTICE

Foley • van Dam • Feiner • Hughes

SECOND EDITION

THE SYSTEMS PROGRAMMING SERIES

Sponsoring Editor: Keith Wollman
Production Supervisor: Bette J. Aaronson
Copy Editor: Lyn Dupré
Text Designer: Herb Caswell
Technical Art Consultant: Joseph K. Vetere
Illustrators: C&C Associates
Cover Designer: Marshall Henrichs
Manufacturing Manager: Roy Logan

This book is in the **Addison-Wesley Systems Programming Series**
Consulting editors: IBM Editorial Board

Library of Congress Cataloging-in-Publication Data
Computer graphics: principles and practice/James D. Foley . . . [et al.].—2nd ed.
p. cm.
Includes bibliographical references.
ISBN 0-201-12110-7
1. Computer graphics. I. Foley, James D., 1942–.
T385.C587 1990
006.6—dc20 89-35281
CIP

Cover: "Dutch Interior," after Vermeer, by J. Wallace, M. Cohen, and D. Greenberg, Cornell University (Copyright © 1987 Cornell University, Program of Computer Graphics.)

ABCDEFGHIJ-DO-943210

continued on the back flap

To Marylou, H

To Debbie, my
my children Eli

To Jenni, my p

To my family,
my father in m

And to all of o

M_1	M_2
Translate	Translate
Scale	Scale
Rotate	Rotate
Scale (with $s_x = s_y$)	Rotate

In these cases, we need not be concerned about the *order* of matrix manipulation.

5.4 THE WINDOW-TO-VIEWPORT TRANSFORMATION

Some graphics packages allow the programmer to specify output primitive coordinates in a floating-point *world-coordinate* system, using whatever units are meaningful to the application program: angstroms, microns, meters, miles, light-years, and so on. The term *world* is used because the application program is representing a world that is being interactively created or displayed to the user.

Given that output primitives are specified in world coordinates, the graphics subroutine package must be told how to map world coordinates onto screen coordinates (we use the specific term *screen coordinates* to relate this discussion specifically to SRGP, but that hardcopy output devices might be used, in which case the term *device coordinates* would be more appropriate). We could do this mapping by having the application programmer provide the graphics package with a transformation matrix to effect the mapping. Another way is to have the application programmer specify a rectangular region in world coordinates, called the *world-coordinate window*, and a corresponding rectangular region in screen coordinates, called the *viewport*, into which the world-coordinate window is to be mapped. The transformation that maps the window into the viewport is applied to all of the output primitives in world coordinates, thus mapping them into screen coordinates. Figure 5.10 shows this concept. As seen in this figure, if the window and viewport do not have the same height-to-width ratio, a *non*uniform scaling occurs. If the application program changes the window or viewport, then new output primitives drawn onto the screen will be affected by the change. Existing output primitives are not affected by such a change.

The modifier *world-coordinate* is used with *window* to emphasize that we are not discussing a *window-manager window*, which is a different and more recent concept, and

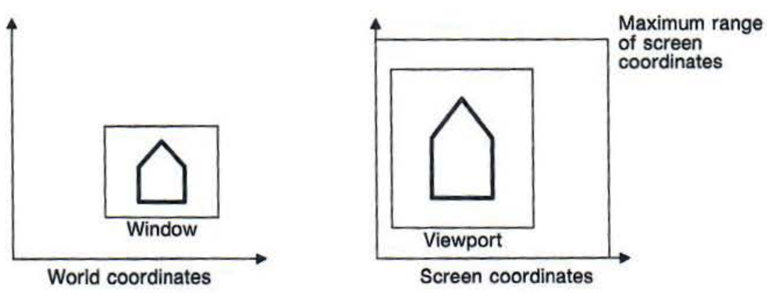


Fig. 5.10 The window in world coordinates and the viewport in screen coordinates determine the mapping that is applied to all the output primitives in world coordinates.

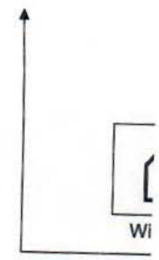


Fig. 5.11 The eff primitives specifyin changed to viewpc package to draw th

which unfortunately of window is meant

If SRGP were tc the current canvas, be able to change t specified output pri included a different in positions differen

A window man which case not all c Chapter 10, we fi viewports, and win

Given a window from world coordi developed as a thr

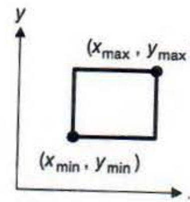


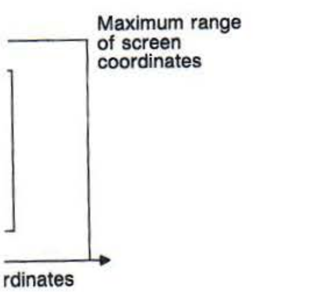
Fig. 5.12 The s

ıf matrix manipulation.

RMATION

ıtput primitive coordinates in a units are meaningful to the ;ht-years, and so on. The term ;enting a world that is being

linates, the graphics subroutine screen coordinates (we use the specifically to SRGP, but that rm *device coordinates* would be g the application programmer to effect the mapping. Another rectangular region in world :esponding rectangular region in ırld-coordinate window is to be viewport is applied to all of the into screen coordinates. Figure ɔw and viewport do not have the rs. If the application program es drawn onto the screen will be affected by such a change. ' to emphasize that we are not ıt and more recent concept, and



viewport in screen coordinates primitives in world coordinates.

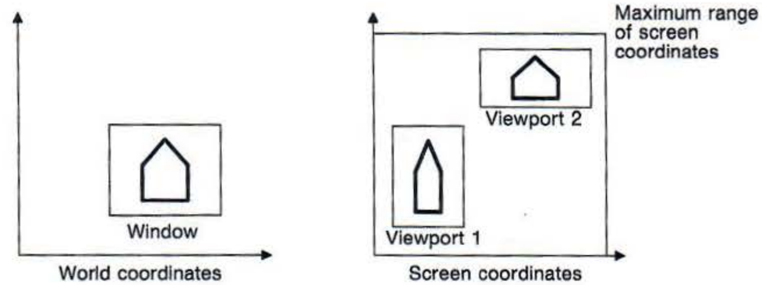


Fig. 5.11 The effect of drawing output primitives with two viewports. Output primitives specifying the house were first drawn with viewport 1, the viewport was changed to viewport 2, and then the application program again called the graphics package to draw the output primitives.

which unfortunately has the same name. Whenever there is no ambiguity as to which type of window is meant, we will drop the modifier.

If SRGP were to provide world-coordinate output primitives, the viewport would be on the current canvas, which defaults to canvas 0, the screen. The application program would be able to change the window or the viewport at any time, in which case subsequently specified output primitives would be subjected to a new transformation. If the change included a different viewport, then the new output primitves would be located on the canvas in positions different from those of the old ones, as shown in Fig. 5.11.

A window manager might map SRGP's canvas 0 into less than a full-screen window, in which case not all of the canvas or even of the viewport would necessarily be visible. In Chapter 10, we further discuss the relationships among world-coordinate windows, viewports, and window-manager windows.

Given a window and viewport, what is the transformation matrix that maps the window from world coordinates into the viewport in screen coordinates? This matrix can be developed as a three-step transformation composition, as suggested in Fig. 5.12. The

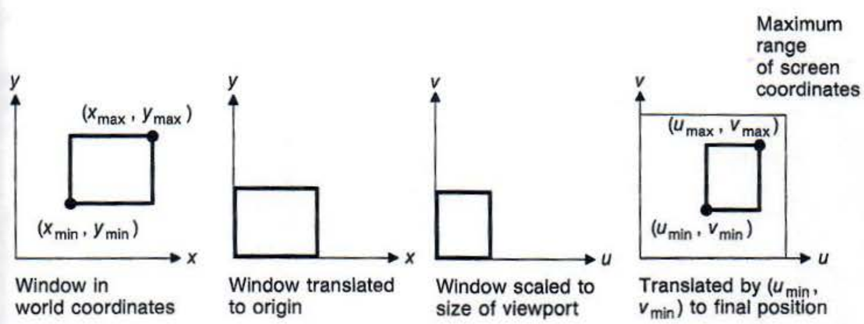


Fig. 5.12 The steps in transforming a world-coordinate window into a viewport.

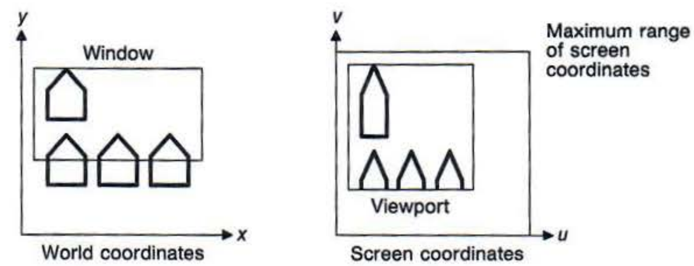


Fig. 5.13 Output primitives in world coordinates are clipped against the window. Those that remain are displayed in the viewport.

window, specified by its lower-left and upper-right corners, is first translated to the origin of world coordinates. Next, the size of the window is scaled to be equal to the size of the viewport. Finally, a translation is used to position the viewport. The overall matrix M_{wv} is:

$$M_{wv} = T(u_{min}, v_{min}) \cdot S\left(\frac{u_{max} - u_{min}}{x_{max} - x_{min}}, \frac{v_{max} - v_{min}}{y_{max} - y_{min}}\right) \cdot T(-x_{min}, -y_{min})$$

$$= \begin{bmatrix} 1 & 0 & u_{min} \\ 0 & 1 & v_{min} \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \frac{u_{max} - u_{min}}{x_{max} - x_{min}} & 0 & 0 \\ 0 & \frac{v_{max} - v_{min}}{y_{max} - y_{min}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & -x_{min} \\ 0 & 1 & -y_{min} \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \frac{u_{max} - u_{min}}{x_{max} - x_{min}} & 0 & -x \cdot \frac{u_{max} - u_{min}}{x_{max} - x_{min}} + u_{min} \\ 0 & \frac{v_{max} - v_{min}}{y_{max} - y_{min}} & -y \cdot \frac{v_{max} - v_{min}}{y_{max} - y_{min}} + v_{min} \\ 0 & 0 & 1 \end{bmatrix}. \qquad (5.33)$$

Multiplying $P = M_{wv} \, [x \quad y \quad 1]^T$ gives the expected result:

$$P = \left[(x - x_{min}) \cdot \frac{u_{max} - u_{min}}{x_{max} - x_{min}} + u_{min} \quad (y - y_{min}) \cdot \frac{v_{max} - v_{min}}{y_{max} - y_{min}} + v_{min} \quad 1\right]. \qquad (5.34)$$

Many graphics packages combine the window–viewport transformation with clipping of output primitives against the window. The concept of clipping was introduced in Chapter 3; Fig. 5.13 illustrates clipping in the context of windows and viewports.

5.5 EFFICIENCY

The most general composition of R, S, and T operations produces a matrix of the form

$$M = \begin{bmatrix} r_{11} & r_{12} & t_x \\ r_{21} & r_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}. \qquad (5.35)$$

The upper 2 × 2
are composite translat
nine multiplies and s
simplifies the actual

reducing the process
since the operation
Thus, although 3 ×
tions, we can use th
structure. Some harc
diminishing or remo
Another area wh
such as a molecule o
view can be created
object will appear t
each individual poin
(Eq. (5.6)) require f
recognizing that, be
approximation, Eq.

which requires just
significant on comp
Equation (5.37)
small error is built i
error gets a bit large
correct values, and t
lines.
A better approx

$y' = x'$

This is a better
corresponding 2 ×
unchanged.

5.6 MATRIX

Just as 2D transfo
coordinates, so 3D
homogeneous coo
representing a poi

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.