Lecture 7

Clipping Concluded; Introduction to Curve Representations

Wednesday, February 9, 2000

William H. Hsu
Department of Computing and Information Sciences, KSU
http://www.cis.ksu.edu/~bhsu

Readings:
- Sections 11.1, 11.2.1-11.2.2, Foley et al
- Optional references: Hearn and Baker 2nd, Angel 2nd

Introduction to Curve Representations

- What are cubic curves?
- How would we use them to represent surfaces?
- Read about polygon meshes, Bezier curves, B-splines for next time!

Next Lecture: Cubic Curves and Surfaces

Quick Review: 3D View Volumes

Based on Figure 7.21, [Angel, 2000]

View volume (frustum)

Based on Figure 6.25, [Foley et al, 2000]
Figure 5.25, [Angel, 2000]
And Figure 12-30(b) [Hearn and Baker, 1997]

Perspective View Volume

View plane

Front clipping plane

Back clipping plane

Projectors

Quick Review: From Perspective to Parallel

- Perspective-to-Parallel Viewing Transformation
  \[ N_{pers} \cdot M = S_{pers} \cdot S_{trans} \cdot T_{trans} \cdot R \cdot T \cdot V_{pers} \] (Equation 6.49, FVD)
- \[ S_{pers} \cdot S_{trans} \cdot T_{trans} \cdot R \cdot T \cdot V_{pers} \] matrix stack
- All point (VFP) to origin - translate
- Eye point (COP) to origin - translate
- Center line of view volume aligned with y - shear
- Normalize to (+1, -1) canonical view volume - scale
- But wait...
  - Q: What is M?
  - A: perspective-to-parallel viewing transformation

Significance
- Q: Why is this important?
- A: Clipping is much uglier without M
- Q: What kind of transformation is M?
- A: Nonuniform scaling (see 6.5, FVD; 12.4, Hearn and Baker)

Quick Review: Specifying an Arbitrary 3D View

Based on Figure 6.25, [Foley et al, 2000]
Figure 5.25, [Angel, 2000]
And Figure 12-30(b) [Hearn and Baker, 1997]

glFrustum (xmin, xmax, ymin, ymax, near, far)

NB: (x, y) extrema on front clipping plane

Clipping

- Problem Definition
  - Given: coordinates for primitives (line segments, polygons, circles, ellipses, etc.)
  - Determine: visible components of primitives (e.g., line segments)

- Objectives
  - Correctness: display primitive components if visible
  - Efficiency
    - Fewer tests
    - Fewer floating point operations per test

- Solution Approaches
  - Solving simultaneous equations (quick rejection: testing endpoints)
  - Solving parametric equations

- Clipping in 3D
  - 2D algorithms extendible to 3D: Cohen-Sutherland, Liang-Barsky
  - Specification and implementation of view volumes needed

- Transparent Implementation in Graphics APIs: Last Lecture
Clipping Lines [1]: Simultaneous Equations

- Recall: Cohen-Sutherland Quick Rejection Method
  - Unique 4-bit binary number (outcode) for each of 9 regions
  - Check clipping cases (NB: not necessarily conclusive)
    - 2 outcodes per line segment: \( o_1, o_2 \)
      - Case 1: \( o_1 \neq 0000, o_2 \neq 0000 \)
      - Case 2: \( o_1 = 0000, o_2 = 0000 \) (or vice versa)
      - Case 3: \( o_1, o_2 \neq 0000 \)
      - Case 4: \( o_1 \neq 0000, o_2 = 0000 \)

- Using Cohen-Sutherland
  - Check all segments with inconclusive outcodes (Case 4)
    - Compute line-rectangle intersection using simultaneous equations
      - \( y = y_{min} \) or \( y = y_{max} \)
      - In general: parametric tests require more floating-point operations, but better performance in worst-case (or nearly so – see Homework 1)

- Based on Figure 7.21, (Angel, 2000)

Clipping Polygons

- Section 3.10, FVD; Section 6.8, Hearn and Baker; Section 7.9, Angel
- Intuitive Idea
  - “Can use line-clipping procedures”
- Implementation
  - Algorithm (Sutherland-Hodgeman)
    - Process polygon boundary (all segments) against each edge
    - Correctness issue: concave polygons (may have multiple clipped polygons)
    - Scan conversion: sweeping algorithm \( (x, y, \text{or polar coordinates}) \)
    - Can also modify Liang-Barsky (similar idea)

- Optional reference: Section 6.7 Hearn and Baker 2e

Clipping Lines [2]: Parametric Equations

- Intuitive Idea
  - Use parametric line formulation: \( P(t) = P_0 + (P_1 - P_0)t \)
  - Find 4 \( t \) values for 4 clip edges
  - Decide which of these form true intersections
  - Calculate \( (x, y) \) for those only
  - There can be only... 2

- Cyrus-Beck: Earliest Parametric Line Clipper (Figure 3.44, 3.12.4 FVD)
  - Independently developed
  - Slightly faster for clipping against rectangles (extra rejection testing)
  - Main references
    - Section 5.12, FVD
    - vanDam, Lecture 4-3 (09/30/1999)
  - Optional reference: Section 6.7 Hearn and Baker 2e

Next Lecture: Cubic Curves and Surfaces

Clipping in 3D [1]: Extending Cohen-Sutherland

- How To Use Liang-Barsky in 3D
  - New need to clip 6 faces instead of 4 edges
    - Right
    - Left
    - Bottom
    - Top
    - Front
    - Back
  - Performance
    - Cyrus-Beck: for 3-D perspective pyramid
    - Liang-Barsky: can do faster using cuboids

- See Table 6.1, Figure 6.55, FVD
Curve Representations: Overview

- **Representations**
  - Polygon mesh
  - Parametric cubic equations (splines, etc.)
- **Parametric Cubic Curves (Table 11.2, FVD)**
  - Hermite curves
  - Bézier curves
  - Cubic splines
    - Uniform B-splines
    - Uniformly shaped B-splines
    - Nonuniform B-splines (rational: NURBS, nonrational)
    - Catmull-Rom
    - Kochanek-Bartels
- **References**
  - Section 11.2, FVD
  - 10.6-10.8, Neun and Baker 2nd, 10.1-1.3,10.6, Angel 2nd

Cubic Curves: Interpolation Methods

- **Cubic Curves**
  - Defined using control points
  - Intuitive idea
    - Interpolation: fitting curve
    - Touches specified control points (e.g., endpoints and midpoint)
    - Smooth (twice differentiable)
    - Fit to other control points
    - See properties in Table 11.2
    - Curve hull defined by control points?
    - Interpolation method
    - Ease of subdivision
- **Next Time**
  - Defining cubic curves
  - Computing, render interpolations
  - Rendering bicubic surfaces

Course Project: Suggested Topics

- **Photorealistic Rendering**
  - Scene illumination – ray tracing, radiatively, fast shading, etc.
  - Artificial objects – transparent surfaces, faces, natural scenes
- **Realistic Animation**
  - Animating specific entities – human faces, hair, bodies, etc.
  - Advanced topics – physically based modeling, particle systems, etc.
- **Image Processing**
  - 2D image transformation – compression, correction, etc.
  - Parametric clipping: line / rectangle intersection using parametric equation
  - Mapping – texture, bump, etc.
- **Visualization**
  - Statistical data visualization, information visualization
  - Scientific visualization – fluid dynamics, geology, groundwater, etc.
- **Mathematical Modeling**
  - Geometric models: curves, surfaces, 3D solid models, etc.
  - Fractal image synthesis, fractal image compression, other

Course Project: More Guidelines

- **Project Proposal (Due 02/14/2000)**
  - 1-3 page description of project topic, plan
  - See: implementation practicum links (Brown, Cornell, UNC, others) on 736 page
- **Guidelines**
  - Specify focus of your project
  - Implementation (e.g., “write a simple animator for articulated figures”)
  - Experiments with existing algorithms (e.g., “compare texture mappers A, B”)
  - Experiments with visualizing data and processes (e.g., “write a visualization technique to ray tracer F”)
  - Extending or combining existing algorithms (e.g., “add multiple light-source technique to ray tracer F”)
  - Applying CG techniques to specific problem (e.g., “rendering back end for solid CAD/CAM machining model”)
  - State how will you demonstrate your results
  - How will you measure or show success?
  - Make sure your goals are sufficiently narrow and realistic
- **Three Parts: Proposal (20%), Implementation (50%), Report (30%)”

Terminology

- **Review: Normalizing Transformations**
  - \[ \begin{align*}
  H_{\text{pers}} &= M \cdot S_p \cdot S_{\text{sh}} \cdot T_{\text{w}} \cdot \text{PRP} \cdot R \cdot \Pi \cdot VRP \\
  \end{align*} \] (Equation 6.49, FVD)
  - \( M \) nonuniform scaling transformation (perspective-to-parallell)
- **Line Clipping**: Determining Parts of Line Segment Primitives to Display
  - Quick rejection testing for simultaneity equations (Cohen-Sutherland)
  - Division of plane into 9 regions with (4-bit) outcomes
  - Testing endpoints of line segment
  - Parametric clipping: line / rectangle intersection using parametric equation
  - **Cyrus-Beck**: general convex 3D polyhedron
  - **Liang-Barsky**: more efficient, specialized variant (upright 2D, 3D clip regions)
- **Clipping in 3D**
  - **Cubed**: truncated viewing pyramid used to clip after \( H_{\text{pers}} \)
  - **Prismed**: truncated viewing pyramid
- **Cubic Curves**
  - Definition: representation of curve by polynomial (usually smooth) of order 3
  - Interpolation: fitting curves given specified control points

Summary Points

- **Quick Review: 3D Viewing**
  - 3D view volume
  - Perspective-to-parallel transformation (Section 6.5, FVD)
- **Clipping (Concluded)**
  - Quick review: problem definition, objectives, general approaches
  - Simultaneous equations and quick rejection test (Cohen-Sutherland)
  - Parametric clipping (Cyrus-Beck / Liang-Barsky)
  - Clipping in 3D (extending Cohen-Sutherland, Liang-Barsky)
  - Clipping polygons, curves
  - Clipping against windows (exterior clipping)
- **Introduction to Curve Representations**
  - Cubic curves: Bézier curves, cubic B-splines
  - Cubic surfaces: next
  - Read about polygon meshes, Bézier curves, B-splines for next time!
- **Next Lecture: Cubic Curves and Surfaces**