

ĐỒ HỌA MÁY TÍNH I

Phạm Tiến Sơn

Đà Lạt, 2005

Mục lục

Lời nói đầu	7
1 Các thuật toán vẽ đường cong trên thiết bị raster	9
1.1 Đoạn thẳng	9
1.1.1 Thuật toán số gia	11
1.1.2 Thuật toán điểm giữa	13
1.1.3 Một số vấn đề liên quan đến thuật toán vẽ đoạn thẳng	18
1.1.4 Các thuộc tính của đoạn thẳng	21
1.2 Đường tròn	22
1.2.1 Đối xứng tám điểm	22
1.2.2 Thuật toán điểm giữa vẽ đường tròn	23
1.3 Đường cong ellipse	28
1.3.1 Ellipse có dạng chính tắc	29
1.3.2 Ellipse trong trường hợp tổng quát	34
2 Hình học của các đường cong và mặt cong	47
2.1 Mở đầu	47

2.2	Đường cong Bezier	48
2.2.1	Thuật toán de Casteljau	48
2.2.2	Đa thức Bernstein và đường cong Bezier	52
2.3	Các tính chất của đường cong Bezier	55
2.3.1	Điều khiển địa phương	59
2.4	Đa thức từng khúc và các hàm spline	60
2.4.1	Sử dụng các hàm spline như các hàm trộn	63
2.4.2	Xây dựng các hàm trộn	65
2.4.3	Đường cong spline và các hàm cơ sở	66
2.4.4	Các hàm B-spline cơ sở	66
2.4.5	Sử dụng các knot bội	71
2.4.6	Vector knot chuẩn	73
2.5	Các tính chất của đường cong B-spline	75
2.6	Nội suy các điểm điều khiển bằng đường cong B-spline	77
2.7	Thiết kế các mặt Bezier và B-spline	80
2.7.1	Patch Bezier	80
2.7.2	Dán các patch Bezier	81
2.7.3	Patch spline	82
3	Giao của các đối tượng	83
3.1	Mở đầu	83
3.2	Giao của hai đoạn thẳng	83
3.2.1	Phân tích	84

3.2.2	Thuật toán xác định giao hai đoạn thẳng	86
3.3	Đoạn thẳng và hình chữ nhật	87
3.3.1	Tìm giao bằng cách giải hệ các phương trình	89
3.3.2	Thuật toán chia nhị phân	89
3.3.3	Thuật toán Cohen-Sutherland	93
3.3.4	Thuật toán Liang-Barsky	97
3.4	Giao của đoạn thẳng và đa giác lồi	100
3.4.1	Vị trí tương đối của một điểm với đường thẳng	100
3.4.2	Thuật toán tìm giao của đoạn thẳng và đa giác lồi	102
3.5	Giao hai đa giác	107
3.5.1	Thuật toán Sutherland-Hodgman	108
3.5.2	Thuật toán Weiler-Atherton	111
3.5.3	Các phép toán tập hợp trên các đa giác	113
3.6	Ray tracing hai chiều: phản xạ trong buồng kín	114
3.6.1	Vector phản xạ	115
3.6.2	Giao của tia sáng và đường thẳng	117
3.6.3	Giao của tia sáng với đường tròn	121
3.6.4	Xây dựng ví dụ ray tracing	124
3.6.5	Buồng kín là ellipse	126
4	Tô màu vùng	127
4.1	Các định nghĩa	127
4.1.1	Vùng định nghĩa bởi pixel	127

4.1.2	Vùng định nghĩa bởi đa giác	129
4.2	Thuật toán tô màu theo vết dầu loang	129
4.3	Thuật toán tô màu theo con chạy	131
4.4	Thuật toán tô màu theo biên	134
4.5	So sánh các thuật toán	144
4.6	Tô màu các hình chữ nhật	145
4.7	Thuật toán tô màu đa giác	147
4.7.1	Các dòng quét ngang	150
4.7.2	Các mảnh vụn	151
4.7.3	Liên kết cạnh và thuật toán tràn	151
4.7.4	Tô màu các đa giác chồng nhau	158
4.8	Tô màu theo mẫu tô	161
Phần phụ lục: Thư viện graph2D.h		163
Tài liệu tham khảo		171

Lời nói đầu

Đồ họa máy tính là một lĩnh vực hấp dẫn của khoa học máy tính. Chúng ta sử dụng đồ họa máy tính như một công cụ để quan sát thông tin trong nhiều lĩnh vực khác nhau, bao gồm khoa học và công nghệ, hoá học, kiến trúc và giải trí. Các chương trình đồ họa tương tác cho phép người sử dụng làm việc theo cách tự nhiên nhất: người sử dụng cung cấp thông tin cho trình ứng dụng thông qua các hoạt động bên ngoài của họ và sẽ nhận được thông tin trở lại bằng hình ảnh. Đồ họa máy tính đang giúp con người thay đổi về quan niệm và cách thức sử dụng máy tính.

Giáo trình *Đồ họa máy tính I* cung cấp một số kỹ thuật cơ bản của đồ họa máy tính hai chiều. (Đồ họa máy tính ba chiều, một phần quan trọng không thể thiếu được sẽ được đề cập trong một giáo trình khác). Để có một khung cảnh toàn diện và sâu sắc về những nguyên lý và thực hành của đồ họa máy tính, xem các tài liệu dẫn [9] và [11]. Các phương pháp phân tích và thiết kế các thuật toán trong giáo trình cho phép sinh viên có thể viết dễ dàng các chương trình minh họa. Giáo trình được biên soạn cho các đối tượng là sinh viên Toán-Tin và Tin học.

Giáo trình sử dụng ngôn ngữ C để minh họa, tuy nhiên có thể dễ dàng chuyển đổi sang các ngôn ngữ khác; và do đó, sinh viên cần có một số kiến thức về ngôn ngữ C. Ngoài ra, hầu hết các chương trình thao tác trên cấu trúc dữ liệu như danh sách liên kết, nên đòi hỏi sinh viên phải có những kỹ năng lập trình tốt.

Sinh viên cũng cần có cơ sở toán học của những năm đầu đại học: hiểu biết về đại số tuyến tính và hình học giải tích, phép tính vi tích phân.

Mục đích của giáo trình là, ở mức độ nào đó, cho thấy các trình ứng dụng đồ họa được tạo ra như thế nào: Chúng ta cần viết và chạy thử các chương trình. Một trong những mục đích chính của giáo trình là giúp sinh viên nắm vững các phương pháp, trước hết toán học hoá các khái niệm hình học và sau đó chuyển tải thành các đoạn mã chương trình.

Giáo trình bao gồm bốn chương và một phần phụ lục với những nội dung chính như sau:

- Chương thứ nhất đề cập đến các phương pháp vẽ các “nguyên sơ” của đồ họa máy tính: đoạn thẳng, đường tròn và ellipse.
- Phân tích và thiết kế bằng hình học là nội dung chính của Chương 2. Hầu hết các phần mềm đồ họa đều có những chức năng tạo ra các đường cong dựa trên các điểm mà người sử dụng lựa chọn. Chương này cung cấp những nguyên lý và cách tiếp cận thực hành mà các trình ứng dụng đồ họa áp dụng.
- Chương 3 giải quyết bài toán xác định giao của những nguyên sơ đồ họa: Giao hai đoạn thẳng, giao của đoạn thẳng và đa giác lồi (bao hàm các hình chữ nhật) và giao của hai đa giác. Cuối chương là một ví dụ của kỹ thuật “ray tracing” hai chiều: Chuyển động của tia sáng trong buồng kín có chứa các “chướng ngại vật”.
- Chương 4 đề cập đến những thuật toán tô màu vùng bất kỳ: Vùng định nghĩa bởi phần trong, bởi đường biên và vùng là đa giác.
- Phần phụ lục là thư viện các cấu trúc dữ liệu và các hàm cần thiết và thường xuyên sử dụng trong giáo trình.

Trong lần xuất bản thứ hai này, chúng tôi đưa thêm các ví dụ tính toán nhằm minh họa cho phần lý thuyết cũng như giúp sinh viên nắm vững kiến thức đã học. Ngoài ra, các lỗi trong xuất bản lần trước cũng đã được chỉnh lý; mặc dù vậy, tác giả vẫn mong có những đóng góp từ bạn đọc.

Tôi xin cảm ơn những giúp đỡ đã nhận được từ nhiều người mà không thể liệt kê hết, đặc biệt là các bạn sinh viên, trong quá trình biên soạn giáo trình này.

Đà Lạt, ngày 10 tháng 1 năm 2005

PHẠM Tiến Sơn

Chương 1

Các thuật toán vẽ đường cong trên thiết bị raster

Chương này trình bày các thuật toán vẽ đoạn thẳng, đường tròn và ellipse trên lattice nguyên \mathbb{Z}^2 . Các thuật toán chỉ thao tác trên những số nguyên và trong các vòng lặp chỉ sử dụng phép toán cộng nên rất hiệu quả.

1.1 Đoạn thẳng

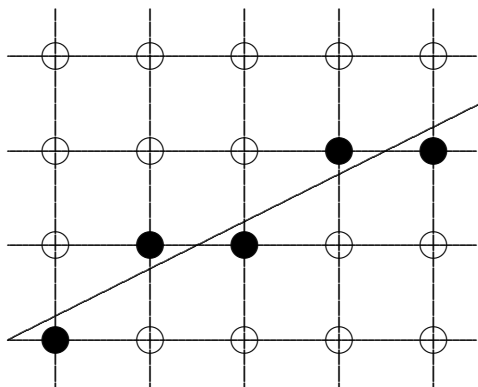
Thuật toán vẽ đoạn thẳng xác định tọa độ của các pixel nằm trên hoặc gần với đoạn thẳng thực tế nhất. Về nguyên tắc, chúng ta muốn chọn dãy các pixel gần với đoạn thẳng thực tế nhất và thẳng nhất. Xét đoạn thẳng thực tế được xấp xỉ với mật độ một pixel; ta cần có những tính chất gì? Với các đoạn thẳng có hệ số góc thuộc đoạn $[-1, 1]$, có đúng một pixel được vẽ lên trên mỗi cột; với các đoạn thẳng mà hệ số góc nằm ngoài đoạn này, có đúng một pixel được vẽ lên trên mỗi hàng. Tất cả các đoạn thẳng được vẽ với cùng một độ sáng, không phụ thuộc vào độ dài và hướng, và nhanh nhất có thể được. Thuật toán vẽ đoạn thẳng cũng cần chú ý đến các thuộc tính của đoạn thẳng như độ rộng, kiểu vẽ... Thậm chí chúng ta muốn cực tiểu hoá mức độ răng cưa do tiến trình rời rạc hoá đường thẳng thực tế nhờ sử dụng kỹ thuật antialiasing (xem [9], [11]) bằng cách áp dụng khả năng đặt cường độ của mỗi pixel trên các thiết bị hiển thị mà một pixel tương ứng nhiều bit.

Trước hết chúng ta chỉ đề cập đến các đoạn thẳng độ rộng một pixel và có đúng một pixel trên mỗi cột (hoặc hàng đối với các đoạn thẳng dốc). Phần cuối chương sẽ đề cập

đến độ rộng các nguyên sơ và các mẫu vẽ.

Một cách hình học, chúng ta biểu diễn một pixel như một chấm tròn với tâm tại vị trí (x, y) của pixel trên lưới các tọa độ nguyên \mathbb{Z}^2 . Biểu diễn này là một xấp xỉ thích hợp nhất cắt ngang trong một chu kỳ của chùm tia electron của CRT; xấp xỉ này phụ thuộc vào khoảng cách (tùy thuộc vào hệ thống) giữa các vết trên màn hình hiển thị. Trong một số hệ thống, các chấm kề nhau phủ lấp một phần lên nhau; với những hệ thống khác có những khoảng cách giữa các pixel đứng kề nhau; trong hầu hết các hệ thống, khoảng cách theo chiều ngang nhỏ hơn theo chiều đứng. Một khác biệt nữa tùy theo hệ thống trong việc biểu diễn hệ tọa độ, chẳng hạn Macintosh xem các pixel được đặt tại tâm của hình chữ nhật giữa các đường thẳng kề nhau của lưới điều khiển thay cho nằm trên các đường thẳng của lưới. Theo cách này, các hình chữ nhật (xác định bởi hai góc) gồm các pixel thuộc phần trong của nó. Định nghĩa này cho phép các vùng độ rộng bằng không: Hình chữ nhật từ (x, y) đến (x, y) không chứa pixel nào, trong khi với những hệ thống khác, có đúng một pixel tại điểm này. Dưới đây chúng ta sẽ biểu diễn các pixel như các hình tròn rời nhau có tâm nằm trên lưới.

Hình 1.1 là phóng to của đường thẳng thực tế và xấp xỉ độ rộng một pixel của nó. Các pixel được vẽ tương ứng các hình tròn màu đen và các pixel không được vẽ tương ứng hình tròn không tô. Trên màn hình thực tế, đường kính của hình tròn biểu diễn pixel lớn hơn khoảng cách giữa các pixel kề nhau, bởi vậy biểu diễn bằng ký hiệu của chúng ta là một phóng đại mức độ rời rạc của các pixel.



Hình 1.1: Đoạn thẳng xấp xỉ được biểu diễn bởi các hình tròn đen.

Vì các nguyên sơ trong hệ thống chúng ta xác định trên lưới điều khiển nguyên nên các tọa độ đầu cuối của đoạn thẳng là nguyên. Thật ra, nếu chúng ta cắt đoạn thẳng với hình chữ nhật trước khi hiển thị nó thì tọa độ các điểm đầu cuối của đoạn thẳng có thể không nguyên. (Chúng ta sẽ thảo luận các giao điểm không nguyên trong Phần 1.1.3). Giả