BÁO CÁO BTCN-01: VẼ CÁC ĐỐI TƯỢNG ĐỒ HỌA 2D

Nguyễn Trần Hậu - MSSV 1612180

Tháng 10, 2018

Mục lục

Tóm tắt nội dung

Trình bày cài đặt thuật toán vẽ các đối tượng 2D là đường thẳng, đường tròn, elip, parabol, hyperbol bằng các thuật toán là DDA, Bresenham, MidPoint. So sánh và đánh giá từng thuật toán khi vẽ 1000, 5000, 10000 đối tượng. Vẽ đường thẳng thêm thuật toán Xiaolin Wu.

1 Thuật toán

1.1 Đường thẳng

1.1.1 DDA

Vẽ một đường thẳng từ điểm (x_1, y_1) đến (x_2, y_2). Tư tưởng chính là cho x chạy từ x1 đến x2, sau đó tính lại y theo phương trình đường thẳng $y = ax + b$.

Để dễ minh họa, xét trường hợp đơn giản $x_1 < x_2$ và $y_1 < y_2$. Xét độ dốc m của phương trình đường thẳng:

$$m = dy/dx = (x_2 - x_1)/(y_2 - y_1)$$

Nếu $m < 1$, x tăng nhanh hơn y và ngược lại khi $m > 1$ thì y tăng nhanh hơn x. Khi x tăng nhanh hơn y, cho x tăng dần từ x_1 đến x_2 rồi tính y theo x, thì có thể vẽ được nhiều điểm hơn để tạo thành đường thẳng. Khi y tăng nhanh hơn x thì làm ngược lại.

1.1.2 Bresenham

Sử dụng thuật toán DDA để vẽ đường thẳng, sẽ phải tính toán với số thực, vì khi tính y theo x có thể là số thực. Dùng thuật toán Bresenham thì chỉ tính toán với số nguyên.

Khi $x_1 < x_2$ và $y_1 < y_2$ và $dy/dx < 1$, thì x tăng nhanh hơn y. Do đó điểm tiếp theo của điểm (x_i, y_i) là $(x_i + 1, y_i)$ hoặc $(x_i + 1, y_i + 1)$. Thực tế thì điểm tiếp theo phải là $(x_i + 1, y)$ nhưng y có thể là số thực. Để xác định được cần chọn điểm nào, dùng:

$$d_1 = y - y_i$$

$$d_2 = y_i + 1 - y$$

Nếu $d_1 < d_2$ chọn y_i, ngược lại chọn $y_i + 1$. Đặt $p_i = dx(d_1 - d_2)$, thì ta có:

$$p_{i+1} - p_i = 2dy - 2dx(y_{y+1} - y_i)$$

Với $p_i < 0$

$$p_{i+1} = p_i + 2dy$$

.

Với $p_i >= 0$

$$p_{i+1} = p_i + 2dy - 2dx$$

.

Bắt đầu vẽ từ điểm (x_1, y_1), do đó

$$p_0 = 2dy - dx$$

Thuật toán bắt đầu từ việc xác định p_0. Sau đó trong lúc cho x chạy từ x_1 đến x_2, kiểm tra p_{i+1} để chọn điểm vẽ tiếp theo là $(x_i + 1, y_i)$ hay $(x_i + 1, y_i + 1)$. Đối với trường hợp y tăng nhanh hơn x, tráo thứ tự x và y.

1.1.3 MidPoint

Thuật toán MidPoint cũng chỉ tính toán với số nguyên. Vẫn là giả sử $x_1 < x_2$, $y_1 < y_2$ và $dy/dx < 1$ để x tăng nhanh hơn y.

Nếu như thuật toán Bresenham dùng d_1 và d_2 để xác định điểm tiếp theo thì thuật toán MidPoint sử dụng điểm $(x + 1, y + 1/2)$ bằng hàm $F(x, y) = Ax + By + C$. $F(x, y) > 0$ điểm nằm trên đường thẳng, ngược lại điểm nằm dưới đường thẳng.

Đặt $p_i = 2F(x + 1, y + 1/2)$, thì ta có

$$p_{i+1} = p_i + 2dy - 2dx(y_{i+1} - y_i)$$

Dễ thấy thuật toán MidPoint cho đường thẳng tính ra hệ số p_i và p_{i+1} giống thuật toán Bresenham. Do đó đối với vẽ đường thẳng, thì MidPoint và Bresenham là một.

1.1.4 Xiaolin Wu

Thuật toán Bresenham vẽ đường thẳng nhanh nhưng không cho phép anti-aliasing. Thuật toán Xiaolin Wu vẽ đường thẳng tương đối nhanh và cho phép anti-aliasing.

Giả sử $x_1 < x_2$ và $y_1 < y_2$. Xét trường hợp x tăng nhanh hơn y. Đặt $gradient = dy/dx$. Khi x tăng dần từ x_1 đến x_2, y lúc này là số thực tăng dần từ y_1 với mỗi lần tăng $gradient$. Thuật toán vẽ 2 điểm $(x, int\_part(y))$ và $(x, int\_part(y)+1)$ với màu sắc lần lượt là $1-float\_part(y)$ và $float\_part(y)$. Với int_part là phần nguyên, $float\_part$ là phần thực. Khi y tăng nhanh hơn x thì hoán đổi vai trò của x và y.

1.2 Đường tròn

1.2.1 DDA

Đường tròn có thể chia ra thành 8 phần, trong đó mỗi phần đều đối xứng với nhau. Do đó chỉ xét vẽ đường tròn từ điểm $(0, R)$ trong 1/8 đường tròn. Nhận thấy lúc này x tăng nhanh hơn y giảm. Cho x tăng từ 0, vẫn còn tăng khi điều kiện $x < y$ vẫn còn thỏa, và tính lại y theo x bằng phương trình đường tròn.

1.2.2 Bresenham

Vẫn vẽ đường tròn bằng cách xác định điểm trong 1/8 đường tròn. Đặt $F(x, y) = x^2 + y^2 - R^2$. $F > 0$ thì (x, y) nằm ngoài đường tròn, $F < 0$ thì (x, y) nằm trong đường tròn. Nếu đã vẽ được điểm $F(x_i, y_i)$, thì điểm tiếp theo cần chọn là $(x_i + 1, y_i)$ hoặc $(x_i + 1, y_i - 1)$.

Đặt $p_i = F(x_i + 1, y_i) + F(x_i + 1, y_i - 1)$[1].

Với $p_i < 0$ chọn $y_{i+1} = y_i$

$$p_{i+1} = p_i + 4x_{i+1} + 2$$

Với $p_i >= 0$ chọn $y_{i+1} = y_i - 1$

$$p_{i+1} = p_i + 4x_{i+1} + 2 - 4y_{i+1}$$

Tại vị trí bắt đầu là điểm $(R, 0)$

$$p_0 = 3 - 2R$$

Xác định p_0 , cho x chạy từ 0 và tiếp tục tăng chừng nào điều kiện $x < y$ còn thỏa mãn. Dùng điều kiện so sánh p_i để chọn điểm tiếp theo là (x_i+1, y_i) hoặc $(x_i + 1, y_i - 1)$.

1.2.3 MidPoint

Vẫn vẽ đường tròn bằng cách xác định điểm trong 1/8 đường tròn. Đặt $F(x, y) = x^2 + y^2 - R^2$. Xét điểm $F(x_i + 1, y_i - 1/2)$.

Đặt $p_i = 4F(x_i + 1, y_i - 1/2)$.

Với $p_i < 0$ chọn $y_{i+1} = y_i$

$$p_{i+1} = p_i + 8x_{i+1} + 4$$

Với $p_i >= 0$ chọn $y_{i+1} = y_i - 1$

$$p_{i+1} = p_i + 8x_{i+1} + 4 - 8y_{i+1}$$

Tại vị trí bắt đầu là điểm $(R, 0)$

$$p_0 = 5 - 4R$$

Cách chạy thuật toán MidPoint giống như thuật toán Bresenham sau khi biết p_0 và p_i.

1.3 Elip

1.3.1 DDA

Hình Elip có thể chia ra làm 4 phần, mỗi phần đối xứng với nhau. Do đó chỉ xét vẽ từng điểm trong 1/4 hình elip. Trong đoạn này chia ra làm 2 đoạn nhỏ tùy vào $dy/dx = -(xb^2)/(ya^2)$. Đoạn đầu tiên x tăng nhanh hơn y giảm khi và chỉ khi $dy/dx > -1$. Đoạn thứ hai y tăng nhanh hơn x giảm khi và chỉ khi $dy/dx < -1$.

Đoạn đầu tiên vẽ từ điểm $(0, b)$, sau mỗi lần tăng x, tính lại y theo phương trình elip. Vòng lặp chạy trong điều kiện $dy/dx > -1$. Đoạn thứ hai vẽ từ điểm $a, 0$, sau mỗi lần tăng y, tính lại x. Vòng lặp chạy trong điều kiện $dy/dx < -1$.

1.3.2 Bresenham

Sử dụng cách chia elip như thuật toán DDA. Đặt $F(x, y) = x^2/a^2 + y^2/b^2 - 1$.

Đoạn đầu tiên vẽ từ điểm $(0, b)$, có

$$p_0 = 2b^2 - 2a^2b + a^2$$

$$p_i = a^2b^2(F(x_i + 1, y_i) + F(x_i + 1, y_i - 1))[2]$$

Với $p_i < 0$, $y_{i+1} = y_i$

$$p_{i+1} = p_i + 4b^2x_{i+1} + 2b^2$$

Với $p_i >= 0$, $y_{i+1} = y_i - 1$

$$p_{i+1} = p_i + 4b^2x_{i+1} + 2b^2 - 4a^2y_{i+1}$$

Đoạn thứ hai vẽ từ điểm $(a, 0)$, có

$$p_0 = 2a^2 - 2ab^2 + b^2$$

$$p_i = a^2b^2(F(x_i, y_i + 1) + F(x_i - 1, y_i + 1))$$

Với $p_i < 0$, $x_{i+1} = x_i$

$$p_{i+1} = p_i + 4a^2y_{x+1} + 2a^2$$

Với $p_i < 0$, $x_{i+1} = x_i - 1$

$$p_{i+1} = p_i + 4a^2y_{x+1} + 2a^2 - 4b^2x_{i+1}$$

Thuật toán vẽ ở một đoạn chọn x hay y tăng liên tục, tùy theo giá trị nào tăng nhanh hơn, sau đó dùng p_i để xác định điểm còn lại.

1.3.3 MidPoint

Sử dụng cách chia elip như thuật toán DDA. Đặt $F(x, y) = x^2/a^2 + y^2/b^2 - 1$.

Đoạn đầu tiên vẽ từ điểm $(0, b)$, có

$$p_0 = 4b^2 - 4a^2b + a^2$$

$$p_i = 4a^2b^2F(x_k + 1, y_k - 1/2)$$

Với $p_i < 0$, $y_{i+1} = y_i$

$$p_{i+1} = p_i + 8b^2x_{i+1} + 4b^2$$

Với $p_i >= 0$, $y_{i+1} = y_i - 1$

$$p_{i+1} = p_i + 8b^2x_{i+1} + 4b^2 - 8a^2y_{i+1}$$

Đoạn thứ hai vẽ từ điểm $(a, 0)$, có

$$p_0 = 4a^2 - 4ab^2 + b^2$$

$$p_i = 4a^2b^2F(x_i - 1/2, y_i + 1)$$

Với $p_i < 0$, $x_{i+1} = x_i$

$$p_{i+1} = p_i + 8a^2y_{k+1} + 4a^2$$

Với $p_i >= 0$, $x_{i+1} = x_i - 1$

$$p_{i+1} = p_i + 8a^2y_{i+1} + 4a^2 - 8b^2x_{i+1}$$

Thuật toán vẽ ở một đoạn chọn x hay y tăng liên tục, tùy theo giá trị nào tăng nhanh hơn, sau đó dùng p_i để xác định điểm còn lại.

1.4 Parabol

1.4.1 DDA

Xét phương trình Parabol $y = ax^2/b$. Parabol đối xứng có thể chia làm 2 phần đối xứng, do đó chỉ cần vẽ một phần của parabol, phần còn lại lấy đối xứng qua. Phương trình có độ dốc $dy/dx = 2ax/b$.

Giả sử $a/b > 0$, x tăng nhanh hơn y khi và chỉ khi $dy/dx < 1$. Ngược lại y tăng nhanh hơn x khi và chỉ khi $dy/dx > 1$.

Với đoạn x tăng nhanh hơn y, x chạy từ $(0, 0)$ đến chừng giới hạn của khoảng không gian 2d được vẽ. Trong lúc đó y được tính lại theo x. Khi y tăng nhanh hơn x thì làm ngược lại.

1.4.2 Bresenham

Như thuật toán DDA, chỉ cần vẽ 1/2 của parabol. Đặt $F(x, y) = y - ax^2/b$.

Trong đoạn x tăng nhanh hơn y, điểm hiện tại đang vẽ là (x_i, y_i)

$$d_1 = y - y_i$$

$$d_2 = y_i + 1 - y$$

Với $d1 - d2 < 0$ thì điểm tiếp theo là $(x_i + 1, y_i)$, ngược lại thì điểm tiếp theo là $(x_i + 1, y_i + 1)$. Đặt $p_i = b(d1 - d2)$, thì:

$$p_0 = 2a - b$$

Với $p_i < 0$, thì $y_{i+1} = y_i$

$$p_{i+1} = p_i + 4ax_{i+1} + 2a$$

Với $p_i > 0$, $y_{i+1} = y_i + 1$

$$p_{i+1} = p_i + 4ax_{k+1} + 2a - 2b$$

Trong đoạn y tăng nhanh hơn x, điểm hiện tại đang vẽ là (x_i, y_i)

$$d_1 = x^2 - x_i^2$$

$$d_2 = (x_i + 1)^2 - x^2$$

Đặt $p_i = a(d1 - d2)$. p_0 được tính từ điểm cuối cùng của đoạn x tăng nhanh hơn y.

Với $p_i < 0$, thì $x_{i+1} = x_i$

$$p_{i+1} = p_i + 2b$$

Với $p_i >= 0$, thì $x_{i+1} = x_i + 1$

$$p_(i + 1) = p_i + 2b - 4ax_{i+1}$$

Thuật toán vẽ ở một đoạn chọn x hay y tăng liên tục, tùy theo giá trị nào tăng nhanh hơn, sau đó dùng p_i để xác định điểm còn lại.

1.4.3 MidPoint

Như thuật toán DDA, chỉ cần vẽ 1/2 của parabol. Đặt $F(x, y) = y - ax^2/b$.

Trong đoạn x tăng nhanh hơn y, xét điểm $(x + 1, y + 1/2)$. Đặt $p_i = 2bF(x + 1, y + 1/2)$, thì:

$$p_0 = b - 2a$$

Với $p_i < 0$, thì $y_{i+1} = y_i + 1$

$$p_{i+1} = p_i + 2b - 4ax_{i+1} - 2a$$

Với $p_i >= 0$, thì $y_{i+1} = y_i$

Trong đoạn y tăng nhanh hơn x, xét điểm $(x + 1/2, y + 1)$. Đặt $p_i = 4bF(x + 1/2, y + 1)$. p_0 là giá trị tại điểm cuối cùng của đoạn x tăng nhanh hơn y.

Với $p_i < 0$, thì $x_{i+1} = x_i$

$$p_{i+1} = p_i + 4b$$

Với $p_i >= 0$, thì $x_{i+1} = x_i + 1$

$$p_{i+1} = p_i + 4b - 8ax_{i+1} - 8a$$

Thuật toán vẽ ở một đoạn chọn x hay y tăng liên tục, tùy theo giá trị nào tăng nhanh hơn, sau đó dùng p_i để xác định điểm còn lại.

1.5 Hyperbol

1.5.1 DDA

Phương trình hyperbol có dạng $x^2/a^2 - y^2/b^2 - 1$. Hyperbol chia làm 4 phần đối xứng với nhau, do đó chỉ cần xác định điểm trên 1 phần, còn lại dùng phép đối xứng. Phương trình có $dy/dx = xb^2/ya^2$. y tăng nhanh hơn x khi và chỉ khi $dy/dx > 1$, x tăng nhanh hơn y khi và chỉ khi $dy/dx < 1$.

Trong đoạn y tăng nhanh hơn x, cho x chạy từ $a, 0$ đến điểm giới hạn trong không gian vẽ 2d. Tính lại y theo x. Tiếp theo là đoạn x tăng nhanh hơn y, thì tính lại y theo x.

1.5.2 Bresenham

Như thuật toán DDA, chỉ cần xác định điểm cần vẽ trên 1/4 hyperbol. Đặt $F(x, y) = x^2/a^2 - y^2/b^2 - 1$.

Xét trong đoạn y tăng nhanh hơn x.

$$d_1 = x^2 - x_i^2$$

$$d_2 = (x_i + 1)^2 - x^2$$

Đặt $p_i = b^2(d_1 - d_2)$[4], thì

$$p_0 = 2a^2 - 2ab^2 - b^2$$

Với $p_i < -b^2/2$, thì $x_{i+1} = x_i$

$$p_{i+1} = p_i + 4a^2 y_{i+1} + 2a^2$$

Với $p_i >= -b^2/2$, thì $x_{i+1} = x_i + 1$

$$p_{i+1} = p_i + 4a^2 y_{i+1} + 2a^2 - 4b^2 x_{i+1}$$

Xét trong đoạn tiếp theo x tăng nhanh hơn y.

$$d_1 = y^2 - y_i^2$$

$$d_2 = (y_i + 1)^2 - y^2$$

Đặt $p_i = a^2(d_1 - d_2)$. p_0 là giá trị tại điểm cuối cùng của đoạn y tăng nhanh hơn x.

Với $p_i < -a^2/2$, thì $y_{i+1} = y_i$

$$p_{i+1} = p_i + 4b^2 x_{i+1} + 2b^2$$

Với $p_i >= -a^2/2$, thì $y_{i+1} = y_i + 1$

$$p_{i+1} = p_i + 4b^2 x_{i+1} + 2b^2 - 4a^2 y_{i+1}$$

Thuật toán vẽ ở một đoạn chọn x hay y tăng liên tục, tùy theo giá trị nào tăng nhanh hơn, sau đó dùng p_i để xác định điểm còn lại.

1.5.3 MidPoint

Như thuật toán DDA, chỉ cần xác định điểm cần vẽ trên 1/4 hyperbol. Đặt $F(x, y) = x^2/a^2 - y^2/b^2 - 1$.

Xét trong đoạn y tăng nhanh hơn x. Đặt $p_i = 4a^2b^2F(x + 1/2, y + 1)$, thì

$$p_0 = 4ab^2 + b^2 - 4a$$

Với $p_i < 0$, $x_{i+1} = x_i + 1$

$$p_{i+1} = p_i + 8b^2 x_{i+1} - 8a^2 y_{i+1} - 4a^2$$

Với $p_i >= 0$, $x_{i+1} = x_i$

$$p_(i + 1) = p_i - 8a^2 y_{i+1} - 4a^2$$

Xét đoạn tiếp theo, x tăng nhanh nhanh hơn y. Đặt

$$p_i = 4a^2b^2F(x + 1, y + 1/2)$$

Với $p_i < 0$, $y_{i+1} = y_i$

$$p_{i+1} = p_i + 8b^2x_{i+1} + 4b^2$$

Với $p_i >= 0$, $y_{i+1} = y_i$

$$p_(i+1) = p_i + 8b^2x_(k+1) + 4b^2 - 8a^2y_(k+1)$$

Thuật toán vẽ ở một đoạn chọn x hay y tăng liên tục, tùy theo giá trị nào tăng nhanh hơn, sau đó dùng p_i để xác định điểm còn lại.

2 Đánh giá

So sánh thời gian vẽ đường thẳng, đường tròn, elip, parabol, hyperbol khi vẽ một cùng bộ random các đối tượng.

	DDA	Bresenham	MidPoint	Xiaolin Wu
Đường thẳng	192ms	194ms	195ms	469ms
Đường tròn	124ms	127ms	126ms	X
Elip	335ms	329ms	335ms	X
Parabol	295ms	295ms	296ms	X
Hyperbol	657ms	660ms	652ms	X

Bảng 1: Vẽ 1000 đối tượng

	DDA	Bresenham	MidPoint	Xiaolin Wu
Đường thẳng	965ms	963ms	964ms	2333ms
Đường tròn	643ms	654ms	657ms	X
Elip	1664ms	1634ms	1649ms	X
Parabol	1422ms	1421ms	1416	X
Hyperbol	3275ms	3273ms	3463ms	X

Bảng 2: Vẽ 5000 đối tượng

	DDA	Bresenham	MidPoint	Xiaolin Wu
Đường thẳng	1933ms	1933ms	1945ms	4829ms
Đường tròn	1287ms	1314ms	1312ms	X
Elip	3643ms	3292ms	3316ms	X
Parabol	2878ms	3089ms	2884ms	X
Hyperbol	6798ms	6646ms	6643ms	X

Bảng 3: Vẽ 10000 đối tượng

Nhận xét:

- Về thời gian: Ở vẽ đường thẳng, thuật toán Xiaolin Wu cho thời gian chạy lâu hơn những thuật toán khác. Thuật toán DDA, Bresenham, MidPoint cho thời gian chạy chênh lệch từ 1ms đến 20ms.

- Về độ chính xác: Trong quá trình test, kết quả vẽ được từ thuật toán DDA, Bresenham, MidPoint tương đối giống nhau qua mắt thường. Còn thuật toán Xiaolin Wu vẽ đường thẳng cho kết quả mịn hơn là 3 thuật toán còn lại.

Tài liệu

[1] A Fast Bresenham Type Algorithm For Drawing Circles
`https://web.engr.oregonstate.edu/~sllu/bcircle.pdf`

[2] A Fast Bresenham Type Algorithm For Drawing Ellipses
`https://dai.fmph.uniba.sk/upload/0/01/Ellipse.pdf`

[3] Midpoint Ellipse Algorithm
`https://www.cpp.edu/~raheja/CS445/MEA.pdf`

[4] Efficient integer algorithms for the generation of conic sections
`http://graphics.di.uoa.gr/Downloads/papers/journals/p7.pdf`

[5] Generating Conic Sections Using an Efficient Algorithms
`https://www.iasj.net/iasj?func=fulltext&aId=38879`

[6] Midpoint circle algorithm
`https://en.wikipedia.org/wiki/Midpoint_circle_algorithm`

[7] Bresenham's line algorithm
https://en.wikipedia.org/wiki/Bresenham%27s_line_algorithm