

Exercícios Revisão 01 - Computação Gráfica

Gustavo Lopes Rodrigues

7 de setembro de 2021

Transformações Geométricas

- 1) Coordenadas homogêneas permite o tratamento algébrico de pontos no infinito.
- 2)
- 3) Essa movimentação só pode ser evitada, garantindo que todo objeto tenha um ponto central, dessa forma, para fazer as operações, o vetor para escalonar é a diferença do vértice e esse ponto central.
- 4)
- 5)
 - a. $A(-1,-3) \rightarrow A'(-2,2);$
 $B(-2,8) \rightarrow B'(-3,13);$
 $C(9,2) \rightarrow C'(8,7);$
 - b. $A(-1,-3) \rightarrow A'(-2.36,-2.09);$
 $B(-2,8) \rightarrow B'(2.26,7.92);$
 $C(9,2) \rightarrow C'(8.79,-2.76);$
 - c. $A(-1,-3) \rightarrow A'(-2.09,-2.36);$
 $B(-2,8) \rightarrow B'(-2,8);$
 $C(9,2) \rightarrow C'(2.76,8.79);$
 - d. $A(-1,-3) \rightarrow A'(-0.5,-6);$
 $B(-2,8) \rightarrow B'(-1,24);$
 $C(9,2) \rightarrow C'(4.5,21);$
 - e. $A(-1,-3) \rightarrow A'(1,-3);$
 $B(-2,8) \rightarrow B'(2,8);$
 $C(9,2) \rightarrow C'(-9,2);$

Rasterização de Retas

6. O valor de delta refere-se ao dx e dy, ou seja, a diferença entre ao par (x,y) do ponto inicial e final da reta. Logo, para atravessar o canvas e preencher os pontos da reta, o número de iterações será essa diferença.
- 7.

DDA

8. Os valores são arredondados apenas na visualização, pois o posicionamento dos pixels são representados por número inteiros. Como o algoritmo DDA trabalha com pontos flutuantes, precisamos arredondar esses valores na hora de exibí-los no canvas

9.

10. a. AB – A(-1,4) e B(5, 7)

-1, 4

0, 5

1, 5

2, 6

3, 6

4, 7

5, 7

- b. BA – B(5, 7) e A(-1, 4)

5, 7

4, 7

3, 6

2, 6

1, 5

0, 5

-1, 4

- c. CD – C(-1, 4) e D(3, 8)

-1, 4

0, 5

1, 6

2, 7

3, 8

- d. EF – E(2, 0) e F(6, 0)

2, 0

3, 0

4, 0

5, 0

6, 0

- e. GH – G(1, 3) e (1, 6)

1, 3

1, 4

1, 5

1, 6

Bresenham

11. O algoritmo de bresenham trabalha com inteiros em vez de pontos flutuantes(floats), isso permite com que as linhas sejam mais precisas, quando comparadas ao algoritmo DDA. Sem contar o gasto em memória é menor, já que precisamos de menor bytes para armazenar valores inteiros do que pontos flutuantes.
- 12.
- 13.
14. Quando a variável **p** for positiva, incrementamos o valor do **y**, isso acontece pois **p** é a variável de decisão que irá conduzir ao algoritmo qual pixel ele deve tomar na hora de formar a linha. Como **p** é positivo, significa que a diferença entre d1 e d2(o pixel anterior e posterior) é maior que zero, por isso, precisamos que o valor de y incremente.
15.
 - a. AB – A(-1,4) e B(5, 7)
 - 1, 4
 - 0, 5
 - 1, 5
 - 2, 6
 - 3, 6
 - 4, 7
 - 5, 7
 - b. BA – B(5, 7) e A(-1, 4)
 - 5, 7
 - 4, 6
 - 3, 6
 - 2, 5
 - 1, 5
 - 0, 4
 - 1, 4
 - c. CD – C(-1, 4) e D(3, 8)
 - 1, 4
 - 0, 5
 - 1, 6
 - 2, 7
 - 3, 8
 - d. EF – E(2, 0) e F(6, 0)
 - 2, 0
 - 3, 0
 - 4, 0
 - 5, 0
 - 6, 0
 - e. GH – G(1, 3) e (1, 6)
 - 1, 3
 - 1, 4
 - 1, 5
 - 1, 6

Rasterização de Circunferências

16. O algoritmo projeta um círculo perfeito, logo, ele só precisa calcular apenas um octante, e o resto ele apenas clona, porém uma diferente posição.
- 17.
- 18.
- 19.

Recorte

- 21.

Cohen-Sutherland

- 22.
23. Há duas condições de paradas para o algoritmo, no primeiro caso, ambos os pontos estão dentro da área de clipping, no outro caso, ambos os pontos estão fora da área;
- 24.
25. c1 e c2 são as variáveis que vão computar a posição dos pontos da reta inserida no algoritmo. Ambas são números onde cada bit representa a direção que o ponto está, em correspondência a área de clipping. O resultado dessas variáveis serão armazenados e então é feita a operação descrita " $c1 \& c2 \neq 0$ ", que irá fazer um bitwise and. Qualquer número diferente de 0 indicará que a computação de c1 e c2 resultou em um número que está fora da área de clipping.
- 26.
27. Primeira Linha(AB) $\rightarrow (-1,-3) \rightarrow (-2,-8)$
Linha calculada:
x1: -1 ,x2: -2 ,y1: -3 ,y2: 8
Linha rejeitada
Segunda linha (BC) $\rightarrow (-2,-8) \rightarrow (9,2)$
Linha calculada:
x1: -2 ,x2: 9 ,y1: 8 ,y2: 2
Linha calculada:
x1: 1 ,x2: 9 ,y1: 6 ,y2: 2
Linha aceita:
x1: 1 ,x2: 6 ,y1: 6 ,y2: 3
Segunda linha (AC) $\rightarrow (-1,-3) \rightarrow (9,2)$
Linha calculada:
x1: -1 ,x2: 9 ,y1: -3 ,y2: 2
Linha calculada:
x1: 5 ,x2: 9 ,y1: 0 ,y2: 2
Linha aceita:
x1: 5 ,x2: 6 ,y1: 0 ,y2: 0

Liang-Barsky

28. Apenas no final é feito a atualização, pois primeiro é preciso verificar se a reta está dentro da área de clipping, mas também é preciso verificar qual é o ajuste necessário para que a reta fique dentro da área desejada
29. As estruturas condicionais são aninhadas, pois o programa verifica pelos cantos da área de clipping, ou seja, os cantos direito e esquerdo, tanto inferior e superior.
30. u_1 e u_2 são parâmetros de interseção da linha, quando u_1 é maior que u_2 , rejeitamos essa linha, pois significa que ela não está mais dentro da área de clipping, logo, iniciamos u_1 com 0 e u_2 com 1, pois $0 < 1$.
31. Primeira Linha(AB) $\rightarrow (-1,-3) \rightarrow (-2,-8)$
 $dy = 11.0$
Linha rejeitada
- Segunda linha (BC) $\rightarrow (-2,-8) \rightarrow (9,2)$
 $dy = -6.0$ $t_2 = 0.72$ $t_1 = 0.33$
Linha aceita:
 $x_1: 1$ $x_2: 6$ $y_1: 6$ $y_2: 3$
- Segunda linha (AC) $\rightarrow (-1,-3) \rightarrow (9,2)$
 $dy = 5.0$ $t_2 = 0.7$ $t_1 = 0.6$
Linha aceita:
 $x_1: 5$ $x_2: 6$ $y_1: 0$ $y_2: 0$

Sutherland-Hodgeman

- 32) O algoritmo Sutherland – Hodgman funciona estendendo cada linha do polígono do clipe convexo por vez, e selecionando apenas os vértices do polígono que estão visíveis.
- 33)
- 34)

Preenchimento de Áreas

- 35) a. Boundary fill é o algoritmo usado com frequência em computação gráfica para preencher uma cor desejada dentro de um polígono fechado com a mesma cor de limite para todos os seus lados.
Vantagens : Lógica simples e simples de implementar
Desvantagens : A cor da borda deve ser a mesma para todas as arestas do polígono.
- b. Flood Fill é o algoritmo que determina e altera a área conectada a um determinado nó em uma matriz multidimensional com algum atributo correspondente.
Vantagens : o preenchimento colore uma área inteira em uma figura fechada por meio de pixels interconectados usando uma única cor.
- É uma maneira fácil de preencher as cores nos gráficos. Um apenas toma a forma e começa o flood fill.
 - O algoritmo funciona de forma a dar a todos os pixels dentro do limite a mesma cor
- Desvantagens** : não é adequado para desenhar polígonos preenchidos, pois perderá alguns pixels em cantos mais agudos.

- c. ScanLine é o algoritmo que processa uma linha por vez, em vez de processar um pixel (um ponto na exibição raster) de cada vez.

Vantagens :

- Classificar vértices ao longo da normal do plano de varredura reduz o número de comparações entre as bordas
- Não é necessário traduzir as coordenadas de todos os vértices da memória principal para a memória de trabalho - apenas os vértices que definem as arestas que cruzam a linha de varredura atual precisam estar na memória ativa, e cada vértice é lido apenas uma vez.

Desvantagens :

- É um algoritmo mais complexo.
- Requer todos os polígonos enviados ao renderizador antes de desenhar.

36)

- 37) a. Boundary fill
- b. Flood Fill
- c. ScanLine

Antialiasing

38)

- a. Superamostragem é o algoritmo onde a intensidade do pixel é calculada em uma resolução mais alta, para ser visualizada em uma resolução mais baixa

Vantagens : fácil de implementar

Desvantagens : Força bruta, mais espaço de armazenamento, mais tempo de processamento

- b. Amostragem por áreas é o algoritmo de intensidade do pixel onde é calculado pelo tamanho da área do pixel que é interceptada/sobreposta pelo objeto

Vantagens :

- Quase elimina o aliasing devido à subamostragem;
- Permite o pré-cálculo de filtros caros;
- Excelente qualidade de imagem;

Desvantagens : Para formas de amostragem diferentes de polígonos, este algoritmo pode ter um preço computacional maior.

- c. Uso de máscaras é um algoritmo onde a intensidade do pixel é calculada a partir da distribuição de pesos entre os pixels dentro de uma grade

Vantagens :

- Aumenta a resolução da imagem original;
- Encontra as melhores posições dos pixels atribuídos. Este procedimento reduz a percepção do efeito escadinha ao retornar uma imagem com melhor qualidade visual;

Desvantagens : Para formas de amostragem diferentes de polígonos, este algoritmo pode ter um preço computacional maior.

- d. Pixel Phasing é um algoritmo onde as extremidades são suavizadas, aqui, as posições dos pixels são deslocadas para posições mais aproximadas especificadas pela geometria do objeto. O algoritmo também permite o ajuste de pixels individuais para um meio adicional ocorrendo uma distribuição de intensidade

Vantagens :

- Aumenta a resolução da imagem original;
- Encontra as melhores posições dos pixels atribuídos. Este procedimento reduz a percepção do efeito escadinha ao retornar uma imagem com melhor qualidade visual;

Desvantagens : Para formas de amostragem diferentes de polígonos, este algoritmo pode ter um preço computacional maior.