

Algorytm Bresenham

$dx, dy, 2dy-2dx$
 $P_0=2dy-dx$

$P_k < 0 \Rightarrow (X_{k+1}, Y_k), P_{k+1}=P_k+2dy$
 $P_k > 0 \Rightarrow (X_{k+1}, Y_{k+1}), P_{k+1}=P_k+2dy-2dx$

ANTYALIASING

- wstawianie dodatkowych jaśniejszych pikseli
- piksel znajdujący się bliżej odcinka jest ciemniejszy
- zwiększenie rozdzielczości

ZMIANA ROZDZIELCZOŚCI

- $P_1 = \alpha B + (1-\alpha)A$
- $P_2 = \alpha D + (1-\alpha)C$
- $N = \beta P_2 + (1-\beta)P_1$

DPI – ROZDZIELCZOŚĆ LINIOWA

- zamiana na cale – podzielić rozdzielczość przez dpi

BARWY W GRAFICE

- RGB (addytywny)
- CMY (substratywny)
- CMYK
- HSV (intuicyjny)
- HLS
- CIEXYZ (barwy fikcyjne)
- $L^*a^*b^*$ (urządzenia odbijające)
- $L^*u^*v^*$ (urządzenia emitujące)
- YUV
- YCCb

DEFINICJA BARWY BIAŁEJ

- ciało doskonale czarne ogrzane do $6500^* K$

ALGORYTM MID POINT

- Jeśli punkt środkowy leży wewnątrz okręgu bierzemy pixel z poza okręgu

WYPEŁNIANIE KOLREM

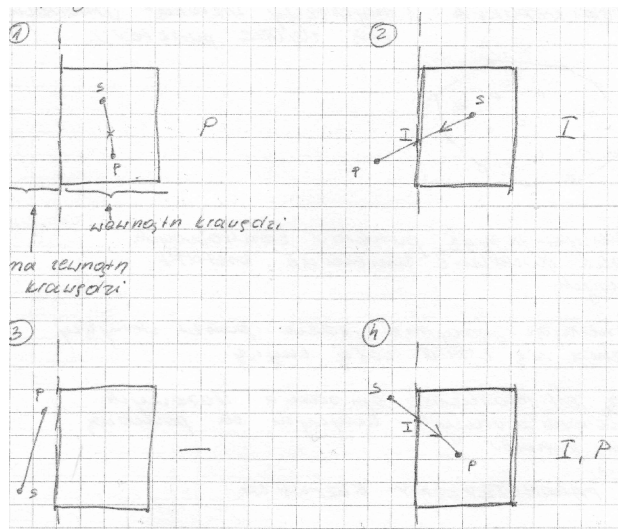
- Scan line
 - szukamy przecięć z krawędziami i numerujemy je. Potem bierzemy po kolei parę liczb z tabeli punktów i wypełniamy linię kolrem.
- Seed
 - wybieramy sobie początkowy punkt, szukamy we wszystkie strony od niego po każdej z linii do napotkania krawędzi.
- Szukanie 4-ech sąsiednich punktów i wypełnianie ich.

WYCINANIE

1. dwie fazy
 - dzielenie obiektu na 9 części i eliminacji części obszaru bez znaczenia
 - badanie obiektu i obcięciu jego fragmentów z poza obszaru lub dzielenie odcinka na coraz mniejsze.

OBCINANIE WIELOKĄTA

- numerowanie odpowiednich węzłów



- po ponumerowaniu zostaje tylko ta część wielokąta z punktów ponumerowanych

KRZYWE BEZIERA

- definiują krzywą po porzez podanie określonej ilości punktów sterujących
- krzywa zawsze przechodzi przez punkt początkowy i końcowy
- odcinek stworzony przez połączenie dwóch pierwszych i dwóch ostatnich punktów jest styczny do krzywej
- krzywa zawsze leży we wnętrzu otoczki wypukłej

TRANSFORMOWANIE GEOMETRYCZNE

- translacja (przesunięcie)
 - $x' = x + tx; y' = y + ty$
 - względem $p(0,0)$
 - $x' = x \cos \Theta - y \sin \Theta; y' = x \sin \Theta + y \cos \Theta$
 - względem $p(xr, yr)$
 - $x' = xr + (x - xr) \cos \Theta - (y - yr) \sin \Theta; y' = yr + (x - xr) \sin \Theta + (y - yr) \cos \Theta$
- składowanie
 - względem $p(0,0)$
 - $x' = x * Sx; y' = y * Sy$
 - względem $p(xf, yf)$
 - $x' = x * Sx + xf(1 - Sx); y' = y * Sy + yf(1 - Sx)$

Przesunięcie

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & dx \\ 0 & 1 & dy \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Mnożenie maci

$$\begin{aligned} x' &= 1x + 0y + dx \\ y' &= 0x + 1y + dy \\ 1 &= 0x + 0y + 1 \end{aligned}$$

Składowanie

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} Sx & 0 & 0 \\ 0 & Sy & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

2-ma obrot

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \Theta & -\sin \Theta & 0 \\ \sin \Theta & \cos \Theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- operacja pochylenia

$$SH_x = \begin{bmatrix} 1 & a & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad SH_y = \begin{bmatrix} 1 & 0 & 0 \\ b & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

GRAFIKA 3D

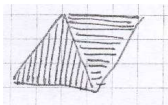
- metody modelowania
 - drutowy
 - BREP (reprezentacja brzegowa)
 - CSG (opis bryły za pomocą innych mniejszych)
 - SWEEP (przesuwamy przekrój obiektu)
 - OCTREE (struktura drzewiasta)
 - dzielimy obszar gdzie znajduje się obiekt na 4
 - dopuki rozpatrywane pola są większe od 1 px dzielimy dalej
 - pole przestajemy dzielić jeśli nie znajduje się w nim kawałek obiektu

ELIMINOWANIE RZECZY NIEWIDOCZNYCH

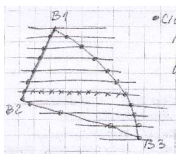
- Metoda węzła normalnego
 - działa tylko dla brył wypukłych
- Metoda malarska
 - rysowanie obiektów od najdalszego
 - wymagane posortowanie obiektów
- Metoda Z-buffer
 - przypisujemy do pamięci obrazu każdemu pixelowi barwę tła
 - dla każdej komórki zapisujemy w z-buforze największą wartość jaką może przyjąć z
 - jeśli $Z_a < Z_{buf}$ to w pamięci obrazu wstawiamy na miejsce barwy tła barwę punktu A, a w z-buforze na miejsce Z_{max} wstawiamy Z_a
 - jeśli $Z_b < Z_{buf}$ to Z_a zastępuję wartością Z_b
 - jeśli $Z_c > Z_{buf}$ to nic nie zmieniamy
 - Algorytm jest uniwersalny, nie ma żadnych ograniczeń
 - Algorytm ma same zalety, zawsze można z niego skorzystać
 - Obiekty można przeglądać w dowolnej kolejności

CIENIOWANIE

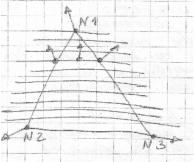
- ogólne
 - światło pada równoległe do światła naturalnego
 - w zależności jak światło pada na ściankę przyjmuje ona inny odcień
 - oblicza się kąt nachylenia ścianki do promieni
 - zakłada się także że jest światło rozproszone nie mające źródła, przez to obiekt jest widoczny
 - trzeba tu rozwiązać problem widoczności od strony padającego światła i obserwatora
- ciągle
 - aproksymacja powierzchni bocznej: trójkąty
 - widoczna jest struktura trójkątów
 - kolory nie przechodzą stopniowo



- GOURAUD
 - polega to na kolorowaniu pixeli lini rastra
 - interpolujemy punkty pomiędzy B1-B2, B2-B3, B1-B3
 - interpolujemy punkty przecięć z liniami rastra
 - interpolujemy punkty na liniach rastra
 - cieniowanie stopniowe



- PHONG
 - obliczanie wartości normalnych dla każdego wierzchołka
 - interpolacja pomiędzy wierzchołkami
 - interpolacja pomiędzy punktami przecięć
 - oblicza barwę każdego piksela z osobna
 - jest to najlepsza metoda cieniowania, ale kosztowna



- Śledzenia promieni
 - śledzenie promieni od obserwatora
 - do problemu cieniowania wykorzystuje się dodatkowe promienie pomocnicze
 - najczęściej czasu znalezienie miejsc przecięcia się promieni z trójkątami
 - metody:
 - szukamy przecięć ze wszystkimi trójkątami
 - dzielimy trójkąty na grupy otoczone kulą, szukamy przecięć z kulami
 - dzielimy scenę na sześciany, w każdym z sześcianów znajdują się jakieś obiekty, rozpatrujemy tylko te obiekty, które znajdują się wewnątrz sześcianu przez który przechodzi promień
 - metoda daje bardzo dużą dokładność, jest bardzo pracochłonna

WYZNACZANIE BARWY W PRZESTRZENI

- Równanie PHONG'a

TEKSTUROWANIE