

SPIS TREŚCI

2. Co to jest grafika komputerowa, przetwarzanie obrazu i rozpoznawanie obrazu
3. Klasy danych obrazowych
5. Przedstawić za pomocą algorytmu zasadę wyznaczania histogramu rozkładu jasności obrazu kolorowego o 24-bitowej strukturze piksela
6. Przedstawić za pomocą algorytmu zasadę wyznaczania macierzy sąsiedztwa
7. Metoda poprawy jakości obrazu poprzez modelowanie histogramu
10. Metoda poprawy jakości obrazu poprzez korekcję tonalną
12. Metoda poprawy jakości obrazu poprzez filtrowanie przestrzenne
14. Narysować przebiegi czasowe sygnałów synchronizacji poziomej HSYNC i pionowej VSYNC. Zaznaczyć odcinki czasowe związane z generacją linii i pikseli
16. Wyznaczyć szerokość pasma VBW rozdzielczość 1280x1024 częstotliwością 60Hz Czasy wygasz. HBlank=1us i Vblank=0.067ms
17. Wyznaczyć częst. Syg. Synchronizacji poziomej HFreq rozdzielczość 1280 x 1024 częst. 60Hz. HBlank=1us VBlank=0.667ms
18. Omówić budowę i zasadę działania projektorów CRL LCD i DLP
20. Omówić algorytm Bresenhama kreślenia odcinka
22. Omówić algorytm Bresenhama kreślenia okręgu
23. Modelowanie powierzchni metodą Coonsa
24. Modelowanie krzywych i powierzchni funkcjami Beziera
26. Modelowanie krzywych i powierzchni funkcjami B-sklejanymi
27. Modelowanie krzywych i powierzchni funkcjami Beta sklejanymi
28. Metoda modelowania brył za pomocą drzew ósemkowych
29. Metoda modelowanie brył poprzez zakreślanie przestrzeni
31. Metoda modelowania brył CSG
32. Omówić na przykładzie funkcji biblioteki OpenGL przekształcenia geometryczne w przestrzeni 3D
33. Omówić na przykładzie funkcji biblioteki OpenGL rzutowanie perspektywistyczne
34. Omówić na przykładzie funkcji biblioteki OpenGL rzutowanie ortogonalne
36. Jakie czynniki fizyczne wpływają na wrażenie koloru? Podać ich interpretacje..
38. Narysować i opisać model procesu rzutowania 3D
39. Narysować i opisać model koloru CIE La*b. W jakim zakresie def się par. tego modelu
40. Narysować i opisać model koloru RGB i CMY. Podać równanie konwersji RGB->CMY
42. Omówić model koloru CMYK. Podać równanie konwersji CMY -> CMYK
43. Narysować i opisać model koloru HLS. W jakim zakresie def. się par. tego modelu
45. Narysować i opisać model koloru HSV. W jakim zakresie def. się par. tego modelu
47. Operacje w przestrzeni kolorów RGB
48. Cieniowanie powierzchni metodą Gourauda
48. Cieniowanie powierzchni metodą Phong
49. Model oświetlenia Phong
51. Model oświetlenia Halla
53. Równanie rendeingu

Co to jest grafika komputerowa, przetwarzanie obrazu i rozpoznawanie obrazu

a) Grafika komputerowa - tworzenie obrazów na podstawie informacji opisowej

- szerokie zastosowania
- różna złożoność programów i nakładów obliczeniowych koniecznych do utworzenia obrazu
- wykresy funkcji
- kompozycje obrazów
- sceny w symulatorach

b) Przetwarzanie obrazu - dotyczy zagadnień, w których dane wejściowe i wyjściowe mają postać obrazów

- usuwanie zakłóceń, poprawa jakości obrazu
- filtrowanie obrazu
- kompresja danych obrazowych
- redukcja obrazu do konturów
- redukcja obrazu konturowego do figur kreskowych
- tworzenie przekrojów obrazu

c) Rozpoznawanie obrazu - tworzenie opisu obrazu, zakwalifikowanie obrazu do pewnej szczególnej klasy.

- odwrotność grafiki komputerowej
- w wyniku analizy obrazu powstaje opis matematyczny, ciąg symboli lub graf
- dalsze przetwarzanie danych opisowych pozwala na wyciąganie wniosków analitycznych

Klasy danych obrazowych

- 4 klasy danych obrazowych
- podzial odnosi sie do sposobu reprezentacji (opisu) nie do tresci wizualnej
- rozne metody i algorytmy przetwarzania dla kazdej klasy

KLASA 1. Obrazy realistyczne monochromatyczne i kolorowe

- dokladnie odzwierciedlaja rzeczywistosc
- reprezentowane jako macierze pamieci
- elementy obrazu (pixele) sa komorkami macierzy
- rozne struktury komorek macierzy np.
- 1 bajt na element obrazu
- 1 bajt na kazdy z 3 kolorow podstawowych (True Color)

KLASA 2. Obrazy binarne

- obrazy monochromatyczne - czarno-biale (tekst)
- element obrazu opisywany przez 1 bit
- reprezentowane jako macierze pamieci lub mapy bitowe
- brak standardowego sposobu grupowania bitow w bajt a bajtow w slowo w pamieci systemu graficznego

KLASA 3. Linie proste i krzywe

Przyklady:

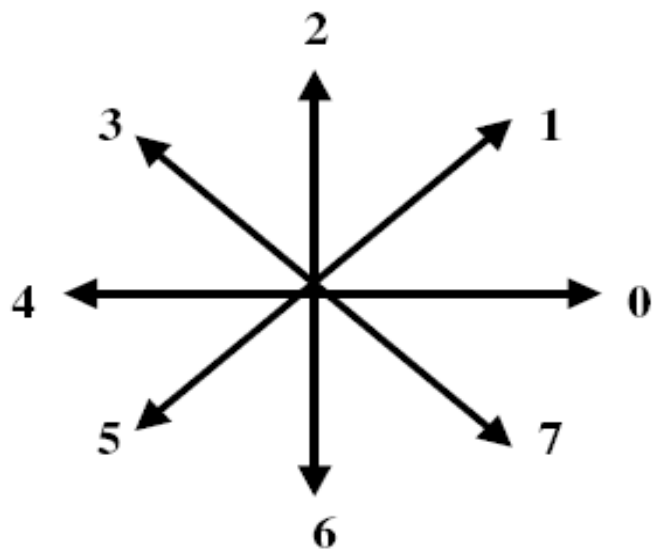
- kontury obszarow
- wykresy
- krzywe nieregularne

Reprezentacja danych:

a) współrzędne X i Y kolejnych punktów

b) przyrosty ΔX i ΔY

c) kody lancuchowe, w których wektor łączący dwa kolejne punkty jest określony jednym symbolem ze skończonego zbioru symboli



przyrosty: $\Delta X \{0,1,2,3,4\}$ 4 bity/punkt

$\Delta Y \{0,6,7,1,2\}$

kod lancuchowy: $\{0,1,2,3,4,5,6,7\}$ 3 bity/punkt

d) różnicowy kod lancuchowy - gdzie reprezentacja każdego punktu jest różnicą pomiędzy dwoma kolejnymi kodami. Zbiór symboli jest również 8-elementowy $\{0, \pm 1, \pm 2, \pm 3, 4\}$. Lecz prawdopodobieństwo ich pojawienia się nie jest takie same. Można zatem użyć kodu o zmiennej długości.

KLASA 4. Punkty i wieloboki

- klasa najczęściej wykorzystywana w grafice komputerowej
- obraz reprezentowany jest przez współrzędne punktów i funkcje
- punkty charakterystyczne połączone są liniami prostymi lub krzywymi opisanymi przez funkcje
- ronorodność funkcji i metod opisujących kształt generowanych obiektów

Przedstawić za pomocą algorytmu zasadę wyznaczanie histogramu rozkładu jasności obrazu kolorowego o 24-bitowej strukturze piksela

```
Int hR[256];
Int hG[256];
Int hB[256];
Int hist[256];
Czyszczenie hR, hG, hB, hist;
Struct pixel{
    Unsigned char r;
    Unsigned char g;
    Unsigned char b;
};
pixel obraz[K+1][L+1];
for(i=1;i<=K;i++){
    for(j=1;j<=L;j++){
        hR[ obraz[l][j].r]++;
        hG[ obraz[l][j].r]++;
        hB[ obraz[l][j].r]++;
        byte wart_max=max(obraz[i][j].r,obraz[i][j].g,obraz [i][j].b);
        byte wart_min=min(obraz[i][j].r,obraz[i][j].g, g,obraz [i][j].b);
        hist[(int) (wart_max+wart_min)/2]++;
    }
}
```

Przedstawić za pomocą algorytmu zasadę wyznaczania macierzy sąsiedztwa

```
for wszystkie powiazania r miedzy P i Q do
begin
wyzeruj tablice Cr(Z,Y);
for wszystkie elementy P obrazu do
begin
if Q jest pikselem sasiednim do P zgodnie z
powiazaniem r then zwieksz Cr(f(P), f(Q)) o 1;
end begin;
end begin.
```

Macierze sasiedztwa

Def. 1.

$p_1(P,Z)$ - funkcja gestosci prawdopodobienstwa, ze element obrazu P ma poziom jasnosci Z.

Def. 2.

Macierz sasiedztwa oznaczamy jako:

$C_r(Z,Y)$ gdzie r oznacza rodzaj powiazania pomiedzy pikselami P i Q.

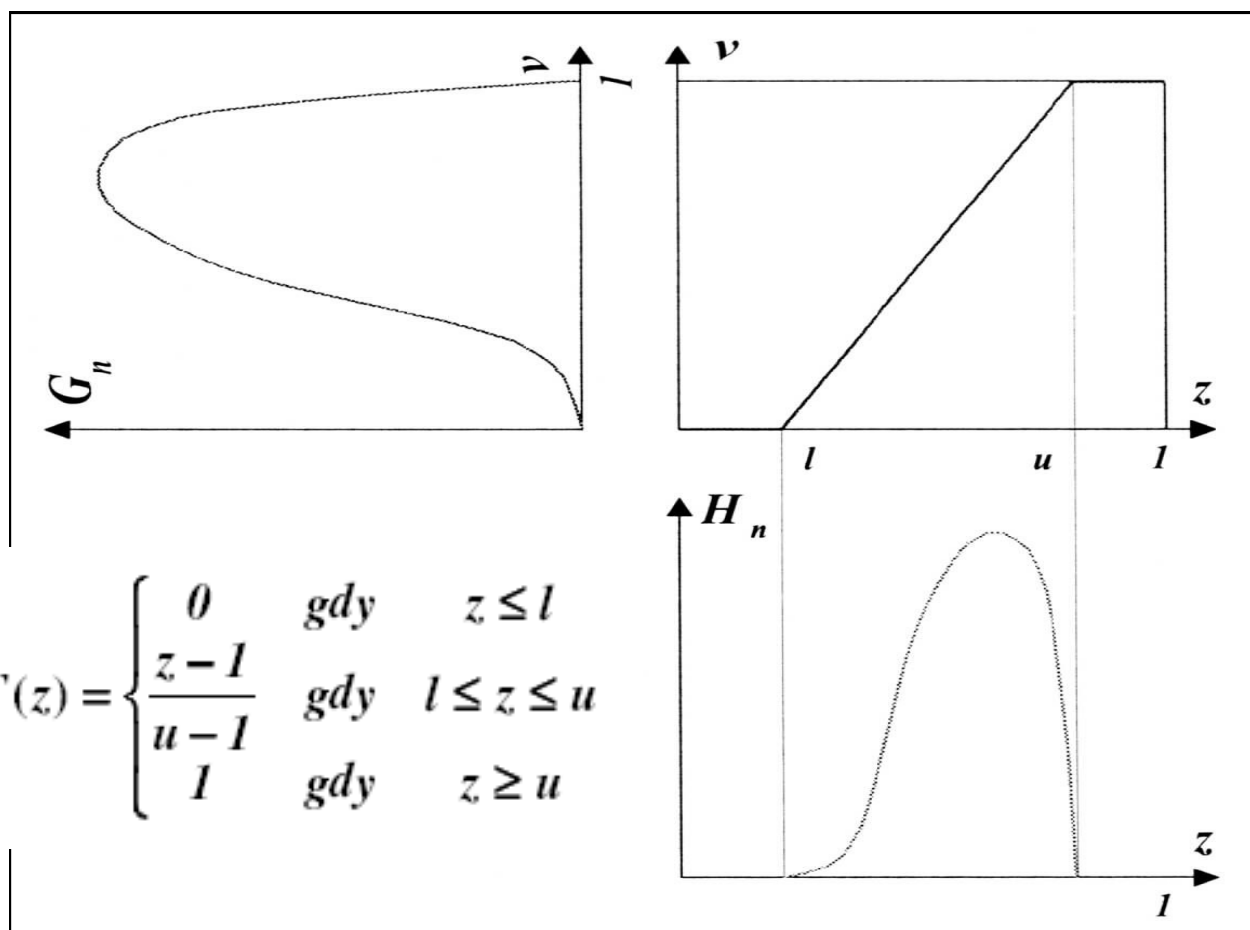
Metoda poprawy jakości obrazu poprzez modelowanie histogramu

Modelowanie histogramu ma na celu poprawę jakości obrazu. Obejmuje procesy:

1. rozciągania histogramu;
2. wyrownywania histogramu;
3. normalizacji histogramu.

1. Rozciąganie histogramu

Operacji rozciągania histogramu dokonuje się wówczas, gdy nie pokrywa on całego zakresu poziomów jasności - na histogramie nie występują bardzo ciemne i bardzo jasne składowe. Rozciąganie histogramu polega na takiej konwersji poziomów jasności obrazu wejściowego, by rozkład poziomów jasności na histogramie obrazu wyjściowego obejmował wszystkie poziomy jasności.



gdzie:

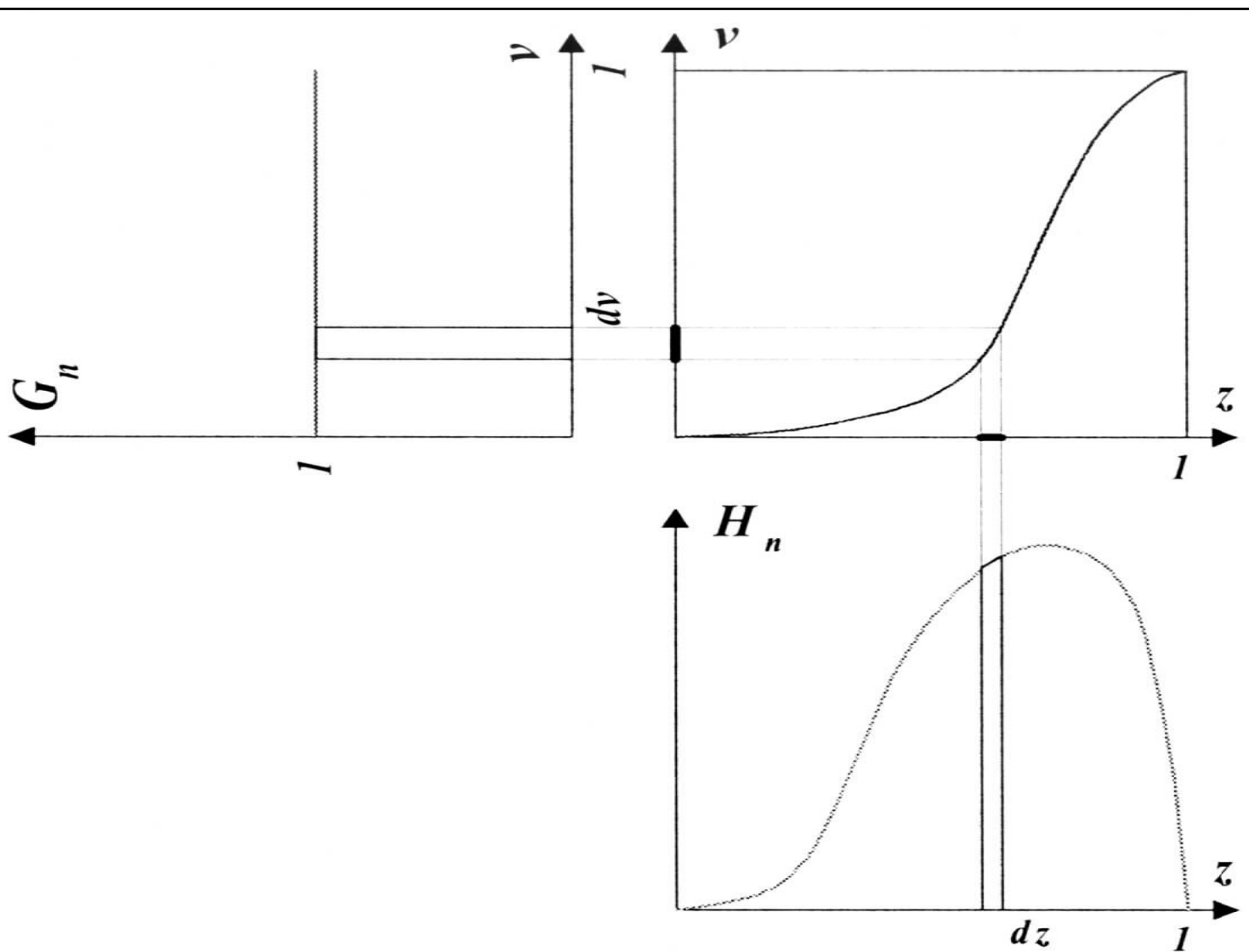
$H_n(z)$ histogram obrazu oryginalnego

$G_n(v)$ histogram obrazu po transformacji T

$v=T(z)$ transformacja

2. Wyównywanie histogramu

Celem wyównania histogramu obrazu jest uzyskanie możliwie równomiernego rozkładu poziomow jasności obrazu dla całego jego zakresu. Skutkiem wyównania histogramu jest poprawa kontrastu obrazu. Wyównanie histogramu pozwala na podkreślenie w obrazie tych szczegółow, które są mało widoczne z powodu niewielkiego kontrastu. Nie jest to metoda uniwersalna i w przypadku histogramow o określonym kształcie nie daje zadowalających rezultatow.



gdzie:

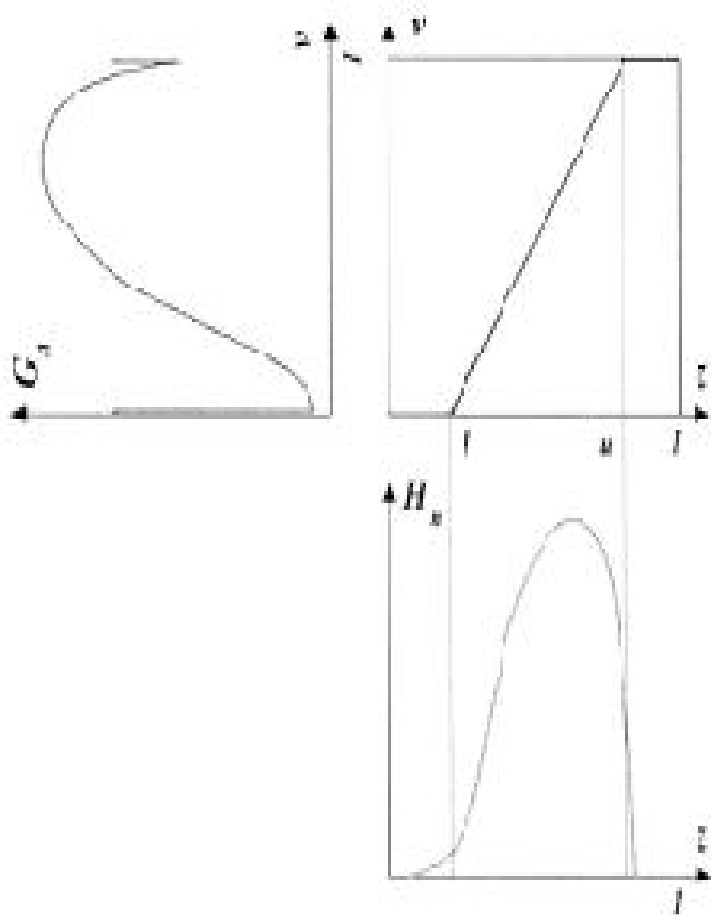
$H_n(z)$ histogram obrazu oryginalnego

$G_n(v)$ histogram obrazu po transformacji T

$v=T(z)$ transformacja

3). Normalizacja histogramu

Operacja normalizacji histogramu zmienia zakres zmienności poziomów jasności obrazu wyjściowego.



$$T(z) = \begin{cases} \int_0^z H_n(z) dz = \epsilon_l & \text{gd}y \quad z \leq l \\ \frac{z-l}{u-l} & \text{gd}y \quad l \leq z \leq u \\ \int_u^z H_n(z) dz = \epsilon_u & \text{gd}y \quad z \geq u \end{cases}$$

gdzie:

$H_n(z)$ histogram obrazu oryginalnego

$G_n(v)$ histogram obrazu po transformacji T

$v=T(z)$ transformacja

Metoda poprawy jakości obrazu poprzez korekcję tonalną

Korekcja tonalna jest przykładem operacji punktowej. W wyniku korekcji tonalnej wykonac można:

- rozjasnienie obrazu;
- przyciemnienie obrazu;
- obniżenie kontrastu;
- zwiększenie kontrastu.

Operacje punktowe to takie, dla których wynik operacji dla każdego pikselu obrazu zależy tylko od jego wartości.

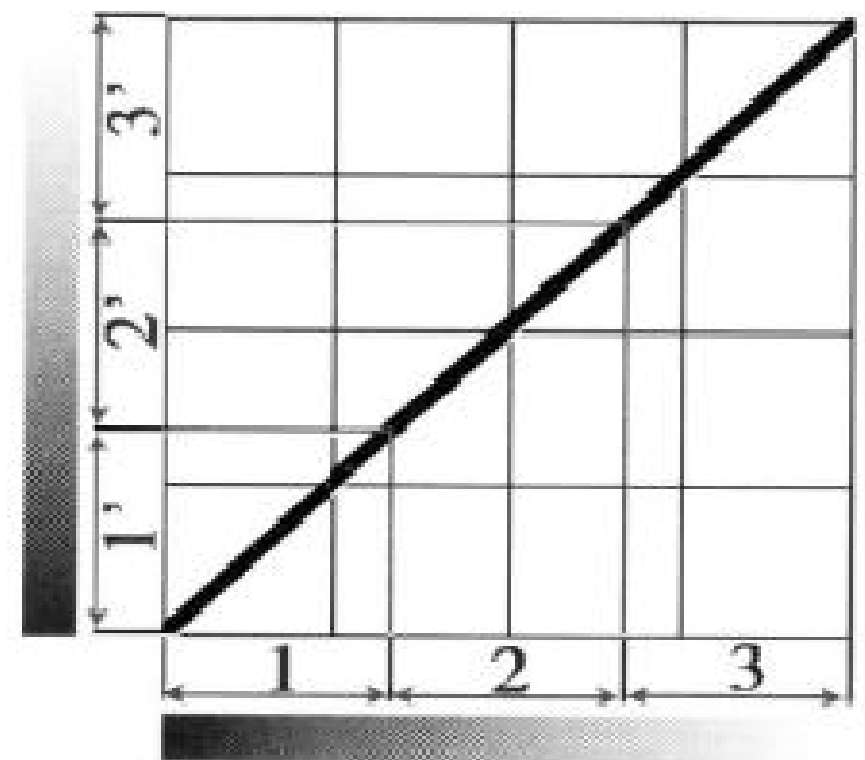
Operacje punktowe nie uwzględniają przestrzennych zależności między elementami obrazu.

Krzywa odwzorowania tonów

Krzywa jest wykresem opisującym zależność pomiędzy wielkością poziomów jasności na wejściu (obraz przed korekcją) a poziomami jasności na wyjściu (obraz po korekcji). Krzywa odwzorowania tonów jest początkowo linią prostą nachyloną pod kątem 45° . Na osi poziomej zaznaczone są poziomy jasności obrazu wejściowego, a na osi pionowej poziomy jasności obrazu wyjściowego. Przed korekcją krzywa przypisuje identyczne poziomy jasności obrazowi wejściowemu i wyjściowemu. W trakcie korekcji kształt krzywej jest zmieniany.

Krzywa posiada umowne zakresy (obszary) jasności:

1. Cienie
2. Tony srednie
3. swiatla



Do korekcji tonalnej obrazow barwnych wygodniej stosowac modele koloru HLS lub HSB, w ktorych w jawnej postaci wystepuje parametr okreslajacy jasnosc piksela (L lub B).

Wyrozniamy dwa rodzaje korekcji tonalnej:

- liniowa;
- nieliniowa.

Korekcja tonalna liniowa

Krzywa odwzorowania tonow ma postac linii prostej (lamanej) nachylonej do osi poziomej pod pewnym katem. Wada korekcji liniowej jest mozliwosc obcicia pikseli koncowych w obszarze cieni lub swiatel. Przy rozjasnianiu czesc jasnych pikseli staje sie zupełnie biala (brak szczegolow w swiatlach obrazu), przy sciemnianiu znikaja szczegoly w cieniach.

Wolna od tej wady jest korekcja tonalna nieliniowa.

Metoda poprawy jakości obrazu poprzez filtrowanie przestrzenne

Filtrowanie przestrzenne

W metodach filtrowania przestrzennego przetwarzaniu podlega bezpośrednio funkcja jasności (obraz jest reprezentowany jako macierz poziomów jasności). Funkcja działająca na obraz wejściowy można opisać wyrażeniem:

$$[g(x, y) = T f(x, y)]$$

gdzie:

$f(x, y)$ - funkcja reprezentująca obraz wejściowy;

$g(x, y)$ - funkcja reprezentująca obraz przetworzony;

T - operator lokalny działający na określonym obszarze obrazu np.: (3x3, 5x5 lub 7x7 pikseli).

Filtr rozmywający (dolnoprzepustowy)

Filtr rozmywający stosowany jest głównie w celu ukrycia drobnych zniekształceń lub redukcji szumów, powstałych np. w wyniku skanowania obrazów rastrowych (efekt mory). Filtr rozmywający nazywany jest często filtrem dolnoprzepustowym ponieważ przepuszcza elementy obrazu o małej zmienności (częstotliwości), tłumi natomiast albo blokuje elementy o większych częstotliwościach (szumy).

Filtr wyostrzający (gornoprzepustowy)

Filtry wyostrzające (gornoprzepustowe) są używane do wzmocnienia szczegółów obrazu o dużej zmienności.

Wykorzystywane są do zaakcentowania obiektów w obrazie albo ich identyfikacji. Ujemnym skutkiem filtracji gornoprzepustowej

jest wzmacnianie szumu w obrazie. Zastosowanie filtrów wyostrzających pozwala na osiągnięcie wyrazistości większej niż w oryginale. Filtry wyostrzające działają w ten sposób, że akcentują (wzmacniają) różnice pomiędzy sąsiadującymi ze sobą jasnymi i ciemnymi punktami.

Detekcja krawędzi

Operatory Laplace'a wykrywają krawędzie we wszystkich kierunkach. Metody wykrywania krawędzi korzystające z Laplasjanów dają w efekcie ostrzejsze krawędzie niż większość innych metod.

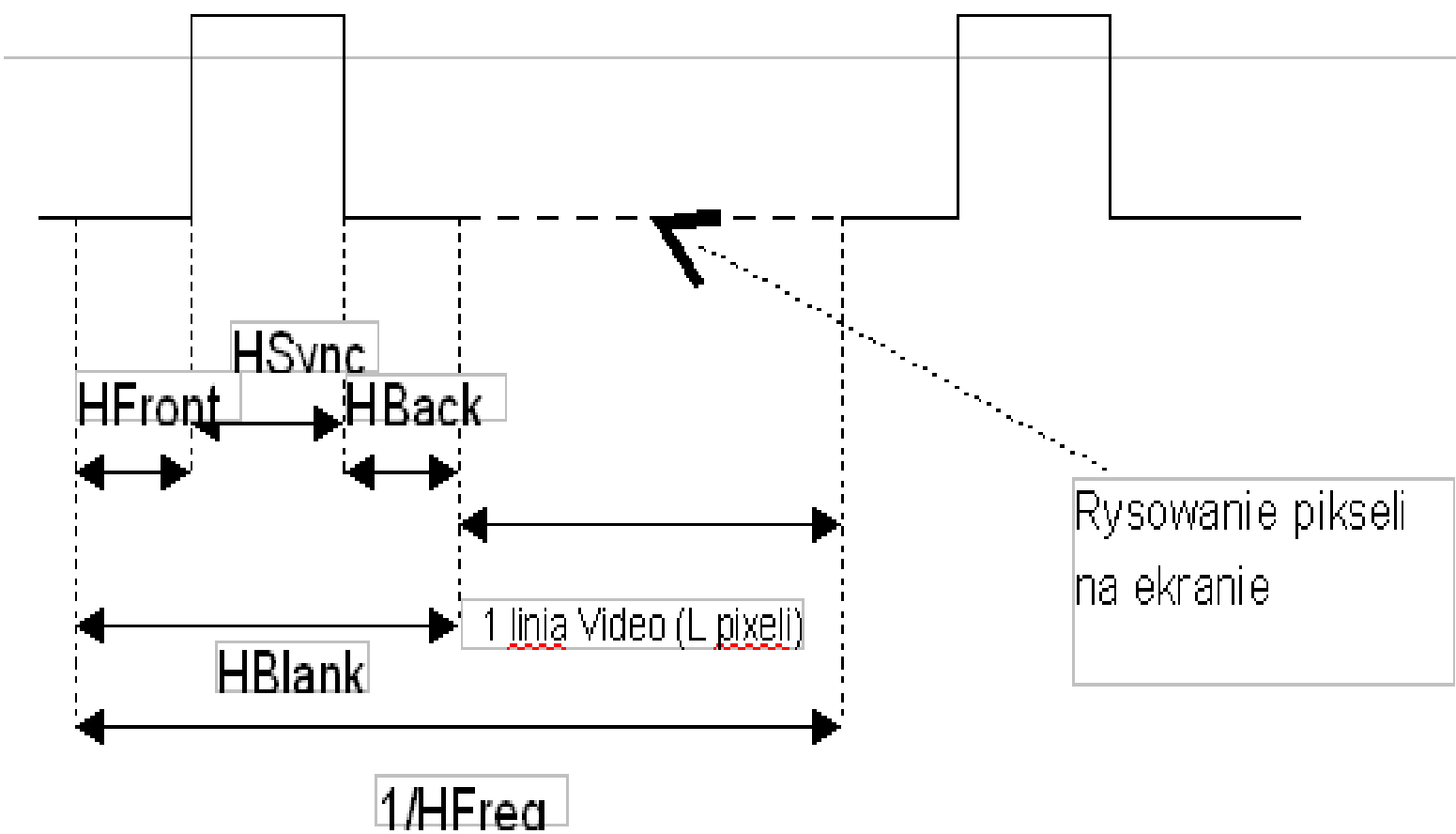
Konturowanie obrazu

Filtry używane do wykrywania krawędzi i tworzenia konturu obrazu są nazywane filtrami konturowymi. Są używane często w procesie segmentacji obrazu i rozpoznawania kształtów obiektów w obrazie. Działają one na zasadzie gradientowej. Gradient określa, jak w obrazie zmieniają się jasności pomiędzy sąsiednimi pikselami. Gradient osiąga największą wartość tam, gdzie w obrazie istnieją największe zmiany jasności przy przejściu od piksela do piksela. Przykładami filtrów konturowych są filtry Sobela i filtry Prewitta.

Filtr medianowy służy do usuwania zakłóceń typu punktowego. Działa erozyjnie, co jest zjawiskiem niepożądanym. Polega to na tym, że po wielokrotnym użyciu filtra medianowego krawędzie obiektów o różnych poziomach jasności stają się poszarpane. W wyniku działania filtra medianowego piksel obrazu wyjściowego przyjmuje wartość mediany z wartości pikseli obrazu wejściowego znajdujących się w jego otoczeniu.

Narysować przebiegi czasowe sygnałów synchronizacji poziomej HSYNC i pionowej VSYNC. Zaznaczyć odcinki czasowe związane z generacją linii i pikseli

- Synchronizacja pozioma**
1. Horizontal Line Rate HFreq [kHz]
 2. Horizontal Synchronization Width HSync [μs]
 3. Horizontal Back HBack [μs]
 4. Horizontal Front HFront [μs]



HBack i HFront określają szerokość aktywnej części obrazowania, czyli lewego i prawego marginesu.

HBlank czas wygaszania.

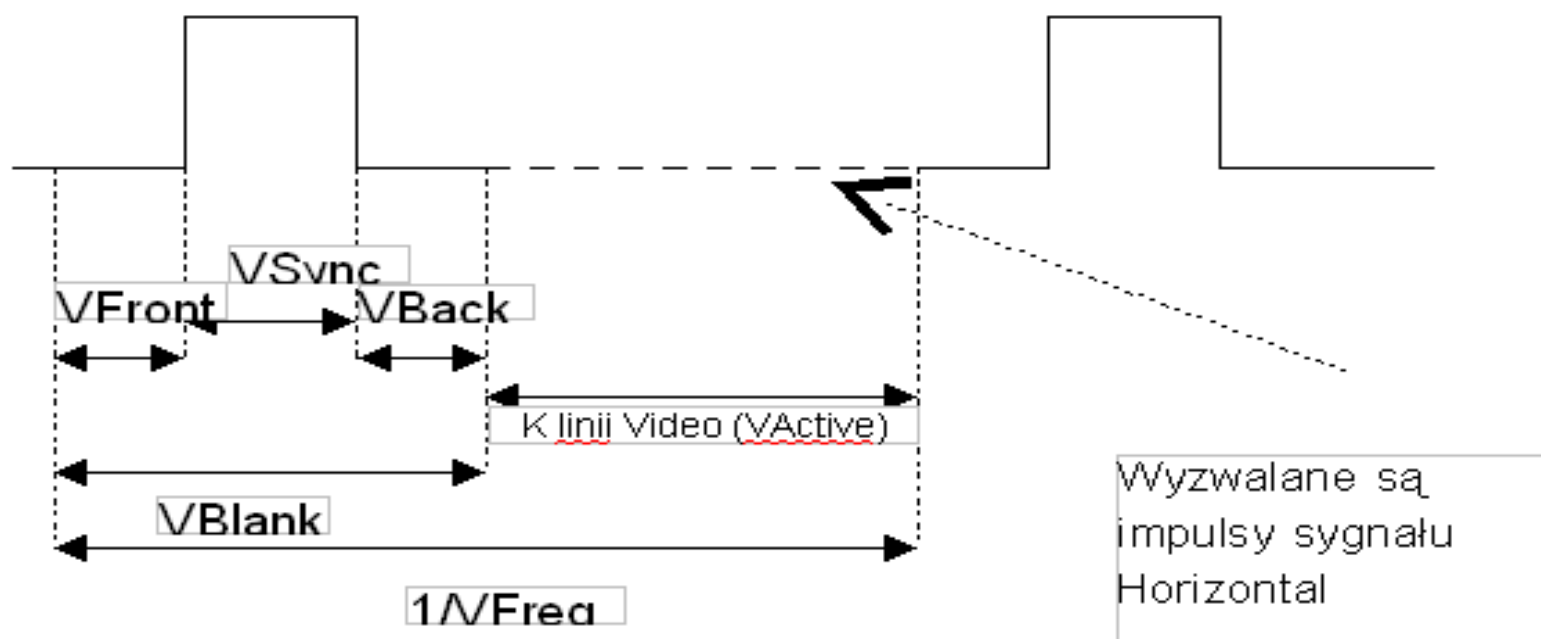
HSync czas trwania wysokiego sygnału w jednym okresie.

HFreq częstotliwość sygnału odpowiedzialnego za rysowanie

wierszy (1 impuls to 1 wiersz).

Synchronizacja pionowa

1. Horizontal Line Rate V_{Freq} [Hz]
2. Vertical Synchronization Width V_{Sync} [ms]
3. Vertical Back V_{Back} [ms]
4. Vertical Front V_{Front} [ms]

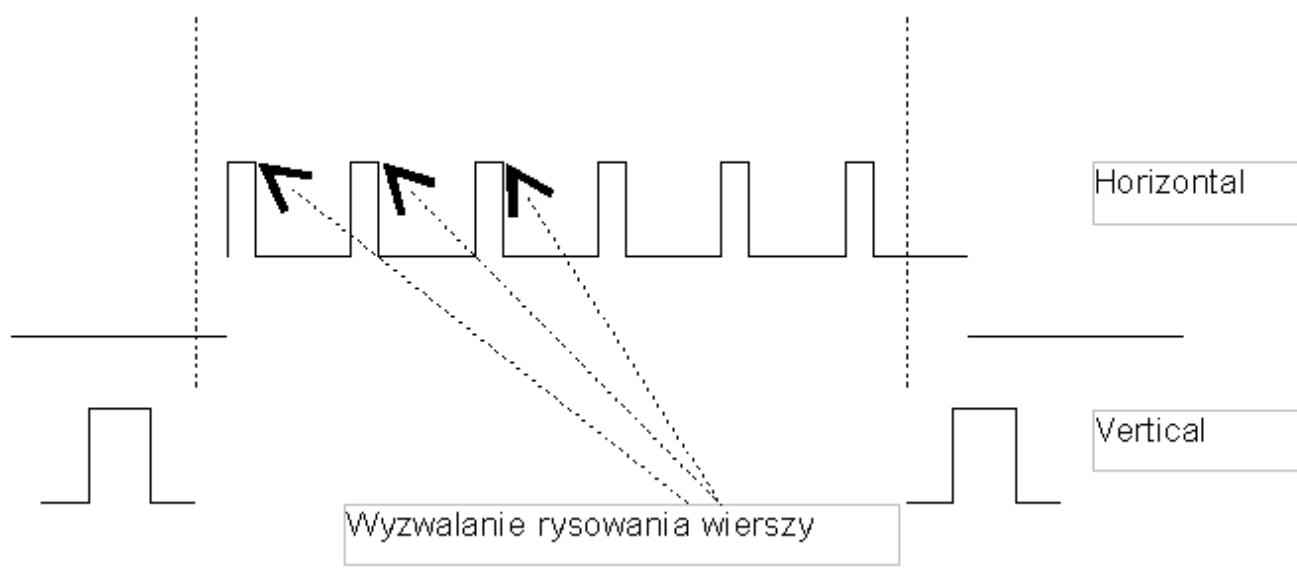


V_{Back} i V_{Front} określają szerokość górnego i dolnego marginesu.

V_{Blank} czas wygaszania.

V_{Sync} czas trwania wysokiego sygnału w jednym okresie.

V_{Freq} częstotliwość sygnału odpowiedzialnego za rysowanie pojedynczych ekranów.



3. Wyznaczyć szerokość pasma video (VBW) karty graficznej generującej obraz o rozdzielczości 1280x1024 w trybie kolejnoliniowym z częstotliwością odświeżania obrazu 60 Hz. Czasy wygaszania są równe: HBlank=1 μ s i VBlank=0.667 ms

$$\frac{1}{V_{Active}} = \frac{1}{V_{Freq}} - V_{Blank} = \frac{1}{60 \text{ Hz}} - 0,667 \text{ ms} = 16,667 - 0,667 = 16,0 \text{ ms}$$

$$H_{Freq} = \frac{1}{V_{Active}} = \frac{1024}{16} = 64 \text{ kHz}$$

$$H_{Video} = \frac{1}{H_{Freq}} + H_{Blank} = 15,62 \mu\text{s} - 1 \mu\text{s} = 14,62 \mu\text{s}$$

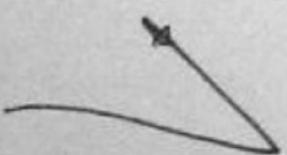
$$VBW = \frac{K}{H_{Video}} = \frac{1280}{14,62 \mu\text{s}} = 87,55 \text{ MHz}$$

✓

3. Wyznaczyć częstotliwość sygnału synchronizacji poziomej karty graficznej generującej obraz o rozdzielczości 1280×1024 w trybie kolejnoliniowym z częstotliwością odświeżania obrazu 60 Hz . Czasy wygaszania są równe: $H_{\text{Blank}} = 1 \text{ } \mu\text{s}$ i $V_{\text{Blank}} = 0.667 \text{ ms}$

$$\frac{1}{V_{\text{przeq}}} = \frac{1}{V_{\text{przeq}}} - V_{\text{Blank}} = \frac{1}{60} - 0.667 = 16.0 \text{ ms}$$

$$H_{\text{przeq}} = \frac{L}{V_{\text{przeq}}} = \frac{1024}{16} = 64 \text{ kHz}$$



Omówić budowę i zasadę działania projektorów CRL LCD i DLP

Projektor CRT (Cathode Ray Tube) składa się z trzech obiektywów - lamp projekcyjnych. Każda lampa emituje obraz w jednej z trzech podstawowych barw (R, G, B). Obraz wynikowy powstaje przez nałożenie się na ekranie poszczególnych składowych. Zastosowanie w systemach projekcji klasy HIGH-END.

Projektor 3-matrycowy

Światło z lampy jest filtrowane i rozszczepiane na trzy składowe R, G i B. Każda z tych składowych jest kierowana na osobną matrycę LCD, której komórki są sterowane zgodnie z wartością odpowiedniej składowej koloru piksela. Każdemu pikselowi obrazu przyporządkowana jest odpowiednia ilość składowej koloru. Strumienie światła po przejściu przez matrycę LCD są następnie syntetyzowane w pryzmacie. Tak wygenerowany obraz wyświetlany jest na ekranie za pośrednictwem obiektywu.

Projektor 1-matrycowy

Światło z lampy przechodzi przez pojedynczą matrycę LCD. Każdy piksel matrycy składa się z 3 komórek LCD z filtrami R, G i B na każdej komórce. Włączenie i wyłączenie komórki jest sterowane zgodnie z wartością odpowiedniej składowej koloru piksela. Strumień światła po przejściu przez matrycę tworzy obraz. Tak wygenerowany obraz wyświetlany jest na ekranie za

posrednictwem obiektywu.

Jednoukładowy projektor DLP

Światło z lampy jest rozszczepiane na trzy składowe R, G i B za pomocą wirującej z częstotliwością 60 Hz tarczy z filtrami R, G i B (180 wiązek światła na sekundę). Tak powstałe wiązki kierowane są na układ DMD (Digital Micromirrors Device), który składa się z bardzo dużej ilości ruchomych mikro-lusterek (na każdy piksel obrazu przypada 1 lustro). Aby uzyskać różne nasycenie danej składowej, część padającego światła jest odbijana w stronę obiektywu, a część rozpraszana zgodnie z wartością odpowiedniej składowej koloru piksela. Odbite światło przechodzi następnie przez układ optyczny. Obraz końcowy jest syntetyzowany przez nakładanie na przemian obrazów odpowiedzialnych za każdą ze składowych R, G i B.

Trzyukładowy projektor DLP

Projektory z 1 układem DMD nie dają wystarczającej ostrości obrazu. W projektorach DLP z 3 układami DMD, na każdy przetwornik trafia jedna wiązka o barwie R, G lub B. Projektory takie generują najlepszy obraz ze wszystkich urządzeń projekcyjnych. Koszt zakupu takiego urządzenia jest kilkakrotnie wyższy niż w przypadku zwykłych modeli DLP i większości projektorów LCD. Kompromisem są urządzenia z dwoma układami DMD, gdzie dodatkowy zestaw lusterek odpowiada za odbijanie wyłącznie światła o barwie czerwonej.

Omówić algorytm Bresenhama kreślenia odcinka

Rysowanie odcinka metoda Bresenhama opiera się na zmiennej decyzyjnej, której znak decyduje o kolejnym kroku algorytmu. Zmienna określa kierunek poziomy, pionowy lub idealnie skosny rysowania kolejnego piksela odcinka.

Algorytm działa na liczbach całkowitych.

Założenia:

- odcinek określony przez współrzędne początku i końca (x_0, y_0) i (x_k, y_k) ;

- $x_0 < x_k$;

- współczynnik kierunkowy odcinka spełnia nierówność:

$$0 < dy/dx \leq 1$$

gdzie: $d_y = y_k - y_0$; $d_x = x_k - x_0$;

Rysowanie zaczynamy od piksela $P_0 = (x_0, y_0)$. Ponieważ kąt nachylenia odcinka jest ograniczony do przedziału $[0, 45^\circ]$, więc po znalezieniu kolejnego $P_i = (x_i, y_i)$ następny piksel wybieramy z spośród tylko dwóch:

$$S_{i+1} = (x_i + 1, y_i) \text{ i } T_{i+1} = (x_i + 1, y_i + 1)$$

Wielkości s i t są określone równaniami:

$$s = (dy/dx) \cdot (x_i + 1 - x_0) - (y_i - y_0)$$

$$t = (y_i + 1 - y_0) - (dy/dx) \cdot (x_i + 1 - x_0)$$

Odejmując te równania stronami i mnożąc przez dx otrzymujemy:

$$d_i = dx(s - t) = 2dy(x_i - x_0) - 2dx(y_i - y_0) + 2dy - dx$$

Ponieważ $dx > 0$, więc znak d_i określa, która z wielkości s i t jest większa.

Jeśli $d_i > 0$, to i za P_{i+1} przyjmujemy piksel T_{i+1} , w przeciwnym razie,

jeśli $d_i < 0$, wybierzemy piksel S_{i+1} . Równość $d_i = 0$ oznacza, że oba

piksele S_{i+1} i T_{i+1} leżą w tej samej odległości od odcinka i wtedy możemy

arbitralnie decydować, np., że $P_{i+1} = T_{i+1}$.

Dla $i+1$ wzór (1) ma postać:

$$d_{i+1} = 2dy(x_{i+1} - x_0) - 2dx(y_{i+1} - y_0) + 2dy - dx$$

Odejmując od niego stronami równanie (1) uzyskujemy zależność:

$$d_{i+1} - d_i = 2dy(x_{i+1} - x_i) - 2dx(y_i - y_i)$$

stad:

$$d_{i+1} = d_i + 2dy - 2dx(y_i - y_i)$$

gdzie: $x_{i+1} - x_i = 1$.

Jeśli $d_i \geq 0$ (wybieramy wtedy $P_{i+1} = T_{i+1}$) to $y_i = y_i + 1$ i zależność

rekurencyjna (2) upraszcza się do postaci:

$$d_{i+1} = d_i + 2dy - 2dx$$

a jeśli $d_i < 0$ (wybieramy wtedy $P_{i+1} = S_{i+1}$) to $y_i = y_i$ i mamy:

$$d_{i+1} = d_i + 2dy$$

Dla $i = 0$, ze wzoru (1) dostajemy wartość początkową zmiennej decyzyjnej $d_0 = 2dy - dx$

Omówić algorytm Bresenhama kreślenia okręgu

Zalozenia:

- promien okregu R jest liczba naturalna, a jego srodek lezy w poczatkku ukladu wspolrzednych;
- osmiokierunkowy wybor piksela;
- ze wzgledu na symetrie okregu ograniczymy wyznaczanie piksela tylko do $1/4$ okregu (dla $a = 1$ wystarczy rozwazenie $1/8$ okregu).

Rysowanie zaczynamy od piksela $P_0=(0,R)$, nastepne wyznaczamy zgodnie z kierunkiem obrotu wskazowek zegara. Po znalezieniu piksela P_i , wybor nastepnego P_{i+1} ogranicza sie do jednego z trzech pikseli oznaczonych na rysunku literami A, B lub C. Punkt Z, w ktorym wspolczynnik kierunkowy wektora stycznego:

$$(dx/dy) = -f_x/f_y = -(2p^2x)/(2q^2y) = -(p^2x)/(q^2y)$$

jest rowny -1 , dzieli cwiartke okregu na dwa wycinki:

- w wycinku 1: $p^2 * x < q^2 * y$ - wybieramy piksel P_{i+1} sposrod pikseli A i B zwikszajac kolejno wartosc x ;
- w wycinku 2: $p^2 * x \geq q^2 * y$ - wybieramy piksel P_{i+1} sposrod pikseli B i C zmniejszajac kolejno wartosc y .

Wybor piksela leacego bliiej okregu (A czy B lub B czy C)

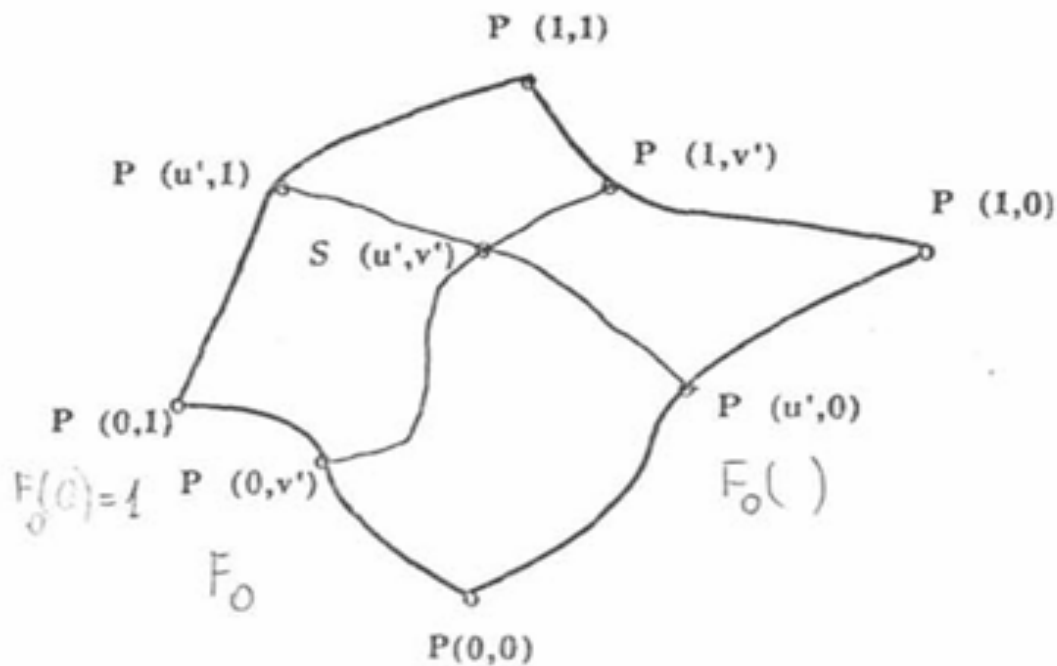
dokonujemy wyznaczajac wartosci $f(x,y)$ dla odpowiednich wspolrzednych x i y . Sposob nie efektywny dziala na liczbach rzeczywistych. W przypadku krzywych opisanych rownaniem wyszego stopnia zloony obliczeniowo. Wygodniej zastosowac inne kryterium wyboru - von Akenema.

Modelowanie powierzchni metodą Coonsa

Metoda ta zakłada, że dane są cztery odpowiednio przecinające się krzywe definiujące brzeg wycinka powierzchni. Niech wycinek powierzchni $S(u,v)$ będzie znormalizowany do kwadratu jednostkowego $0 \leq u \leq 1, 0 \leq v \leq 1$, i krzywe brzegowe można wówczas przedstawić jako $P(u,0), P(u,1), P(0,v)$ i $P(1,v)$. Wycinek powierzchni Coonsa buduje się

interpolując przeciwległe

brzeży.



$$S(u, v) = \sum_{i=0}^1 F_i(u) P(i, v) + \sum_{j=0}^1 F_j(v) P(u, j) - \sum_{i=0}^1 \sum_{j=0}^1 F_i(u) F_j(v) P(i, j)$$

gdzie:

- $P(0,0)$ jest punktem przecięcia krzywych $P(u,0)$ i $P(0,v)$;
- $P(0,1)$ jest punktem przecięcia krzywych $P(u,1)$ i $P(0,v)$;
- $P(1,0)$ jest punktem przecięcia krzywych $P(u,0)$ i $P(1,v)$;
- $P(1,1)$ jest punktem przecięcia krzywych $P(u,1)$ i $P(1,v)$.

Funkcje F_i są to dowolnie dobrane funkcje, spełniające warunki brzegowe:

$$F_0(0)=1, F_0(1)=0, F_1(0)=0, F_1(1)=1$$

Najprostszymi funkcjami F_i są funkcje:

$$F_0(u)=1-u \quad \text{i} \quad F_1(u)=u$$

innymi przykładowymi funkcjami F_i mogą być funkcje:

$$F_0(u)=\cos^2\left(\frac{\pi}{2}u\right) \quad \text{i} \quad F_1(u)=\sin^2\left(\frac{\pi}{2}u\right)$$

Modelowanie krzywych i powierzchni funkcjami

Beziera

Krzywe Beziera definiujemy przez lamana kontrolna o $N+1$ wierzchołkach P_0, P_1, \dots, P_N . Początkowy i końcowy punkt leżący na krzywej pokrywa się z pierwszym i ostatnim punktem kontrolnym. Krzywe Beziera są styczne do lamanej kontrolnej w punkcie początkowym i końcowym. Umożliwia to łączenie ze sobą krzywych i powierzchni Beziera z zachowaniem ciągłości klasy C_1 połączenia. Powierzchnia Beziera jest powierzchnią parametryczną $S(u, v)$ wyznaczona na podstawie siatki punktów kontrolnych, zwanej grafem kontrolnym określonej wg. odpowiedniego wzoru.

Powierzchnia Beziera jest iloczynem tensorowym krzywych Beziera. Większość własności krzywych Beziera przenosi się również na powierzchnie.

Powierzchnia:

$$S(u, v) = \sum_{i=0}^N \sum_{j=0}^M P_{ij} B_{iN}(u) B_{jM}(v)$$

gdzie: u i v należą do przedziału $\langle 0, 1 \rangle$ a B_{ik} są wielomianami Bernsteina:

$$B_{ik}(w) = \frac{k!}{i!(k-i)!} w^i (1-w)^{k-i}$$

Powierzchnia Beziera jest iloczynem tensorowym krzywych Beziera i większość własności krzywych Beziera przenosi się również na powierzchnie.

Krzywa:

$$C(t) = \sum_{i=0}^N P_i B_{iN}(t)$$

gdzie: t należy do przedziału $\langle 0, 1 \rangle$, a $B_{iN}(t)$ jest wielomianem Bernsteina:

$$B_{iN}(t) = C_i^N t^i (1-t)^{N-i}$$

gdzie: $C_i^N = \frac{N!}{(N-i)!i!}$

Właściwości krzywych i powierzchni Beziera:

- nie przechodzą przez wszystkie punkty kontrolne
- krzywe pozostają w wypukłej części łamanej kontrolnej, a powierzchnie leżą w powłoce wypukłej grafu kontrolnego
- nie umożliwiają lokalnej kontroli kształtu. Punkty kontrolne mają wpływ nie tylko na punkty leżące w najbliższym otoczeniu, lecz także na punkty na całej długości krzywej lub powierzchni. Przesunięcie dowolnego punktu kontrolnego powoduje zmianę położenia wszystkich punktów tworzących krzywą lub powierzchnię
- dla N punktów kontrolnych generowana jest krzywa stopnia $N-1$. Wpływa to na złożoność obliczeniową algorytmów realizujących operacje na krzywych i powierzchniach

Modelowanie krzywych i powierzchni funkcjami B-sklejanymi

Znana jest łamana kontrolna o $N+1$ wierzchołkach i określone są węzły t_i . Funkcja bazowa $N_{i,k}(t)$ jest wielomianem stopnia $k-1$, który zapewnia ciągłość klasy C^{k-2} na całej długości definiowanej krzywej. Wartość krzywej zależy od położenia węzłów t_i i t_{i+k} . modyfikacja punktu kontrolnego zmienia lokalnie kształt krzywej. Kształt krzywej może być modyfikowany poprzez wielokrotne zdefiniowanie tego samego punktu kontrolnego. Stopień k funkcji bazowej wpływa na odległość krzywej od punktów definiujących. Powierzchnia B-sklejana definiowana jest iloczynem tensorowym krzywych B-sklejanych.

Właściwości krzywych i powierzchni B-sklejanych:

- definiowane są na podstawie łamanych lub grafów kontrolnych
- umożliwiają lokalną kontrolę kształtu tworzonej krzywej lub powierzchni
- stopień k funkcji bazowej zapewnia ciągłość klasy C^{k-1} na całej długości definiowanej krzywej
- gładkość powierzchni jest kontrolowana przez stopień k i l funkcji bazowych
- stopień funkcji bazowych nie zależy od liczby punktów kontrolnych

Modelowanie krzywych i powierzchni funkcjami

Beta sklejanymi

Właściwości krzywych i powierzchni β -sklejanych:

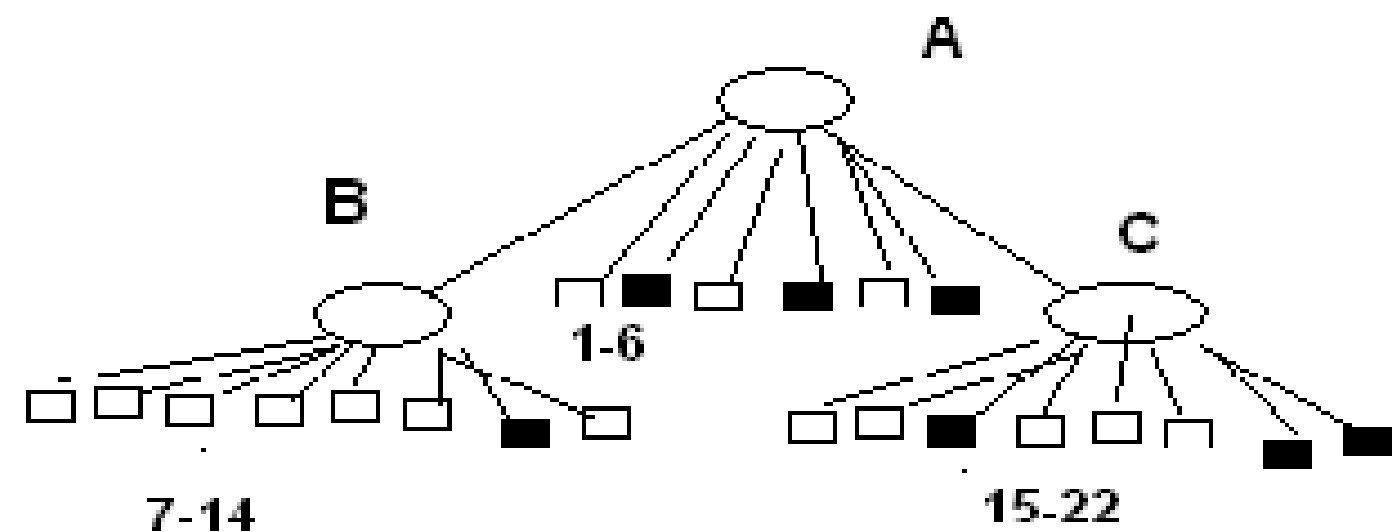
- mogą być wykorzystywane w interakcyjnych systemach graficznych, w których wymagana jest dokładana kontrola kształtu generowanych obiektów
- kontrola kształtu oraz transformacje mogą być dokonywane lokalnie
- parametry β_1 i β_2 mogą być przedstawione w formie funkcji, co umożliwia ciągłą kontrolę kształtu
- stopień bazowych funkcji β -sklejanych jest stały (3 dla krzywych i 6 dla powierzchni)
- nie przechodzą przez punkty kontrolne, nawet przez punkt początkowy i końcowy. Dla zapewnienia przejścia przez określony punkt kontrolny należy do łamanej kontrolnej lub grafu dołożyć dodatkowy punkt

Parametr β_1 wpływa na symetrię krzywej, natomiast β_2 kontroluje stopień przylegania krzywej do łamanej kontrolnej.

Jeżeli $\beta_1=1$ i $\beta_2=0$ to funkcje bazowe β -sklejane redukują się do funkcji bazowych B-sklejanych 3-go stopnia.

Metoda modelowania brył za pomocą drzew ósemkowych

Jest to uogólnienie koncepcji opisu obszarów dwuwymiarowych za pomocą drzew czworkowych na przypadek trójwymiarowy. W metodzie tej obiekt przestrzenny wpisujemy w szescian, któremu odpowiada korzeń drzewa ósemkowego. Szescian dzielimy na osiem mniejszych zwanych oktantami. Jeżeli oktant leży całkowicie wewnątrz bryły, to odpowiadającemu mu węzłowi nadajemy wartość **FULL**. Gdy cały oktant położony jest na zewnątrz bryły, to odpowiadającemu mu węzłowi nadajemy wartość **EMPTY**. Oktanty niejednorodne, tylko częściowo zawarte w bryle, dzielimy dalej na mniejsze, aż do uzyskania wszystkich oktantów jednorodnych lub mniejszych, równych ustalonej minimalnej wielkości. Postępowanie kończymy również w przypadku uzyskania pożądanej dokładności.



Metoda modelowanie brył poprzez zakreślanie przestrzeni

Przez pojęcie zakreslania przestrzeni (ang. sweeping) rozumiemy budowanie bryły przez przemieszczanie jej przekroju (płaskiej figury) wzdłuż pewnej trajektorii. Najprostszymi przykładami takiego postępowania jest przesunięcie równoległe i obrot wokół osi.

Szczególnymi przypadkami obiektów utworzonych w wyniku przesunięcia równoległego są:

- prostopadłoscian: figura zakreslająca jest prostokąt;
- walec : figura zakreslająca jest koło;
- graniastosłup : figura zakreslająca jest wielobok;

Typowo, bryły obrotowe są tworzone w wyniku obrotu płaskiej figury wokół osi o kąt 360 stopni.

Obracając dowolną krzywą parametrycznie zależnościami $x=x(t)$, $y=y(t)$ wokół osi y otrzymujemy powierzchnię o równaniu: $x=x(t)\cos(u)$, $y=y(t)$, $z=z(t)\sin(u)$ przy czym u należy do $[u_1, u_2]$.

Granice u_1 i u_2 przedziału zmienności parametru u określają położenie krzywej w przestrzeni i kąt obrotu. Przykładowo, obracając okrąg o równaniu:

$$X=R*\cos(t), y=\sin(t), t \text{ należy do } [0, 2*\pi]$$

Wokół osi y , dla u należącego do $[0, 1.5*\pi]$

otrzymamy czesc torusa:

Generacja bryly nastepuje w wyniku przemieszczania jej przekroju (plaskiej figury) wzdluz pewnej trajektorii. Najprostszymi przykladami takiego postepowania jest przesuniecie rownolegle i obrot wokol osi. W ogolnym przypadku tworzenia bryly obrotowej parametry zwiazane z ta metoda moga sie zmieniac:

- zmiana polozenia punktow wyznaczajacych os obrotu
- figura zakreslajaca przestrzen moze byc przesuwana w dowolnym kierunku
- figura zakreslajaca przestrzen moze byc skalowana
- ksztalt figury zakreslajacej moze zmieniac sie podczas obrotu

Metoda modelowania brył CSG

CSG (Constructive Solid Geometry)

Metoda budowy brył z ustalonych podstawowych elementów przestrzennych (prymitywów).

Zdefiniowane operacje wykonywane na elementach podstawowych to dodawanie, odejmowanie i iloczyn.

Elementem podstawowym może być dowolny wycinek przestrzeni, czyli zbiór punktów przestrzeni spełniający nierówność $f(x,y,z) \geq 0$. Dowolna bryła opisuje się drzewem, którego liśćmi są prymitywy przestrzenne, a węzły wewnętrzne odpowiadają działaniom na tych prymitywach (dodawanie, odejmowanie, iloczyn) lub transformacjom (obrot, przesunięcie, składowanie).

Omówić na przykładzie funkcji biblioteki OpenGL przekształcenia geometryczne w przestrzeni 3D

Przekształcenia geometryczne w przestrzeni 3D odnoszą się do modyfikacji następujących atrybutów obiektu (lokalizacja, orientacja, rozmiar):

Translacja – zmiana lokalizacji (przesunięcie o wektor)

Skalowanie – zmiana rozmiaru, skalowanie ze współczynnikami $S = \langle SX, SY, SZ \rangle$ $P(PXSX, PYSY, PZSZ)$

Obrot – rotacja (zmiana orientacji)

Przekształcenia te zwane są transformacjami punktowymi, ponieważ realizowane są na każdym punkcie P obiektu.

Składanie przekształceń realizowane jest poprzez mnożenie macierzy

$$C' = CM$$

Gdzie:

C – aktualna macierz przekształceń

M – macierz opisująca dodawane przekształcenie

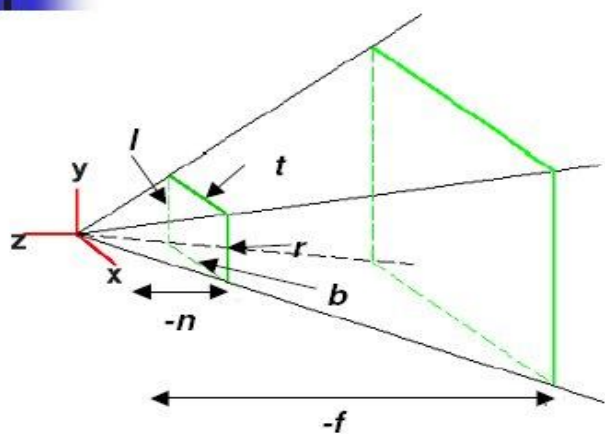
C' – macierz wynikowa

Obraz wierzchołka v po przekształceniu CM

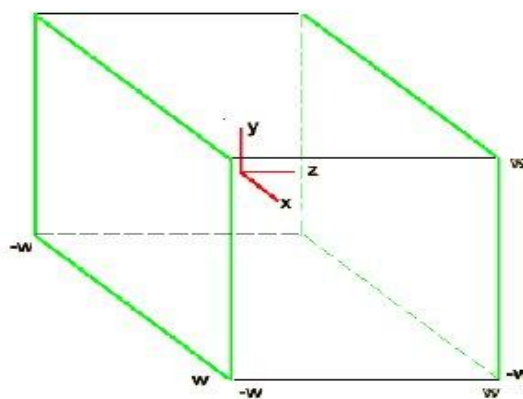
$$v' = C'v$$

Omówić na przykładzie funkcji biblioteki OpenGL rzutowanie perspektywistyczne

Rzut perspektywiczny nadaje realizm symulacjom i animacjom oraz imituje wrażenie głębi obrazu. Wielkość rzutu obiektu jest odwrotnie proporcjonalna do odległości obiektu od środka rzutowania. Rzuty tego typu nie nadają się jednak do rejestrowania kształtu obiektu i dokonywania pomiarów (np. kąty na ogół nie są zachowywane). Punkt zbieżności dla pęków prostych równoległych do danej osi układu współrzędnych, określamy jako osiowy punkt zbieżności. W zależności od liczby istniejących osiowych punktów zbieżności rozróżniamy rzuty jedno-, dwu- i (rzadko stosowane) trzy-punktowe. Liczba osiowych punktów zbieżności zależy od tego ile osi układu współrzędnych przecina rzutnia. Jeżeli punkt centralny leży na ujemnej części osi Z i ma współrzędne $\langle 0, 0, C_z \rangle$, a płaszczyzna rzutowania jest XY, wówczas naszym obrazem punktu $V = \langle V_x, V_y, V_z \rangle$ jest punkt $W = \langle W_x, W_y, 0 \rangle$, gdzie $W_x = V_x / (1 - V_z/C_z)$ oraz $W_y = V_y / (1 - V_z/C_z)$



Układ współrzędnych obserwatora



Układ współrzędnych odciętych

$$\text{glFrustum}(l, r, b, t, n, f) \quad \mathbf{R} = \begin{bmatrix} \frac{2n}{r-l} & 0 & \frac{r+t}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{-(f+n)}{f-n} & \frac{-2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

Omówić na przykładzie funkcji biblioteki

OpenGL rzutowanie ortogonalne

Rzuty ortogonalne są najprostszym rodzajem rzutów, dla którego obraz punktu jest śladem normalnej do płaszczyzny rzutowania przechodzącej przez ten punkt. Jeżeli płaszczyzna rzutowania jest XY wówczas współrzędnej Z przypisuje się wartość 0 . Macierz opisująca taki rzut wygląda następująco:

Tego typu rzutowanie określa się stosując prostokątna lub sześcienna bryła rzutowania. Nic, co znajduje się poza tą bryłą nie jest rzutowane. Bryła rzutu określa się podając bliższa, dalsza, lewa, prawa, gorna i dolna płaszczyzne obcinania. Rzuty te łatwo uzyskać jednak nie dają one obrazów realistycznych chyba, że obserwator znajduje się daleko od obiektu. Ten rodzaj rzutowania stosuje się w projektowaniu architektonicznym i programach CAD.

```
void glOrtho(GLdouble lewa, GLdouble prawa,
             GLdouble dolna, GLdouble gorna, GLdouble
             bliższa, GLdouble dalsza).
```

Definiuje ona płaszczyzny przycinania, oraz

wskazuje na wymiary jednostek miar, używanych w rysowaniu. Kolejne argumenty tej funkcji to kolejno współrzędne lewa, dolna, bliska, prawa, gorna, dalsza prostopadloscianu przyciania. Obserwator znajduje sie w punkcie (0,0,0).

$$\begin{pmatrix} \frac{2}{\text{right} - \text{left}} & 0 & 0 & t_x \\ 0 & \frac{2}{\text{top} - \text{bottom}} & 0 & t_y \\ 0 & 0 & \frac{-2}{z\text{Far} - z\text{Near}} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where

$$t_x = - \frac{\text{right} + \text{left}}{\text{right} - \text{left}}$$

$$t_y = - \frac{\text{top} + \text{bottom}}{\text{top} - \text{bottom}}$$

$$t_z = - \frac{z\text{Far} + z\text{Near}}{z\text{Far} - z\text{Near}}$$

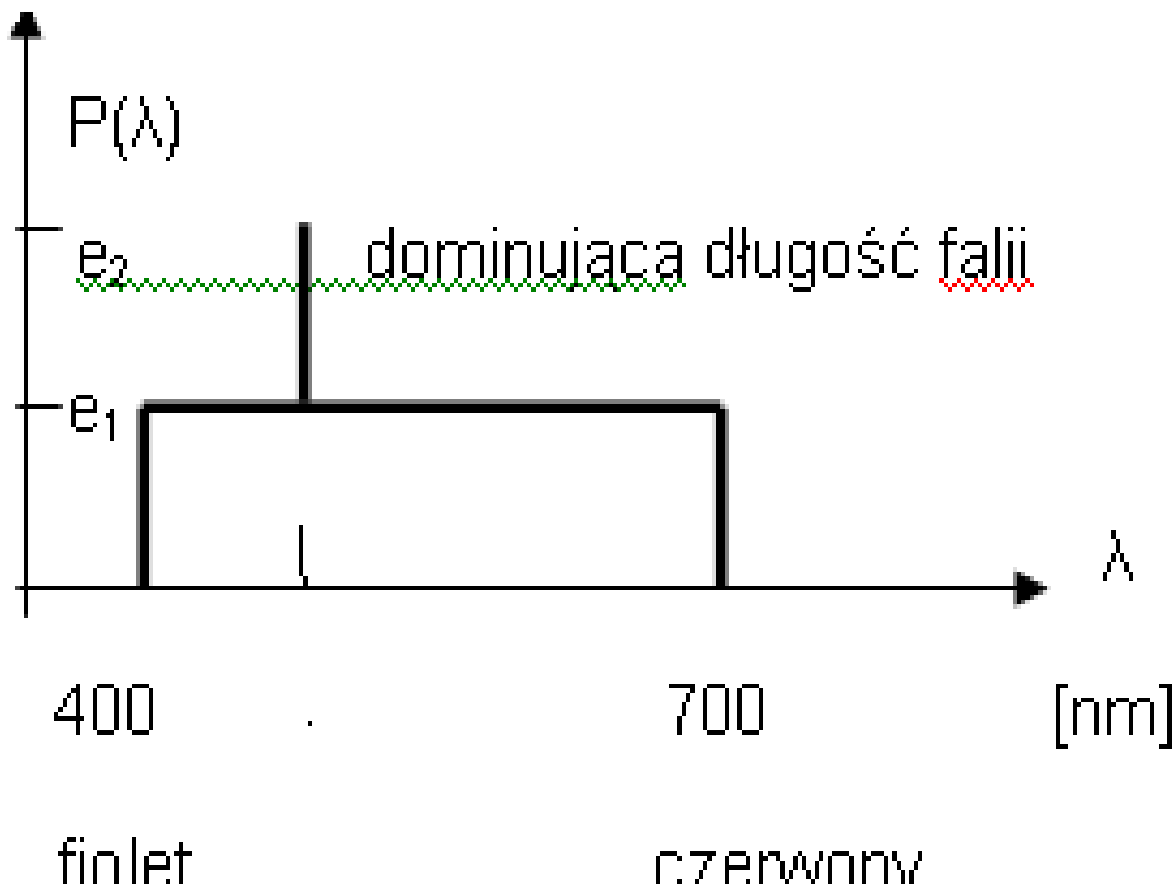
Jakie czynniki fizyczne wpływają na wrażenie koloru? Podać ich interpretacje..

Czynniki mające wpływ na wrażenie koloru:

barwa - fizycznie określa ją długość fali świetlnej (zakres widzialności dla człowieka wynosi od 400 do 700 nm). Barwa zależy od wartości na osi poziomej.

jasność - określa ją wielkość energii związanej z promieniowaniem (nie ma żadnych ograniczeń jeśli jest bardzo duża, może spowodować uszkodzenie oka). Jasność zależy od wartości na osi pionowej.

nasycenie - stosunek różnicy energii dla dominującej fali świetlnej (e_2) i wypadkowej energii dla wszystkich pozostałych długości fali (e_1) podzielona przez wartość energii dla dominującej fali świetlnej (e_2), czyli $(e_2 - e_1) / e_2$. Nasycenie (czystość) barwy jest tym większe, im większa jest wartość e_2 w stosunku do e_1 . Dla $e_2 = e_1$ mamy nasycenie zerowe, gdy zaś $e_1 = 0$, nasycenie wynosi 100%. Nasycenie fizycznie odpowiada ilości fal o różnej długości składających się na dany kolor. Poniższy rysunek przedstawia wykres rozkładu energii światła w funkcji długości fal promieniowania widzialnego.



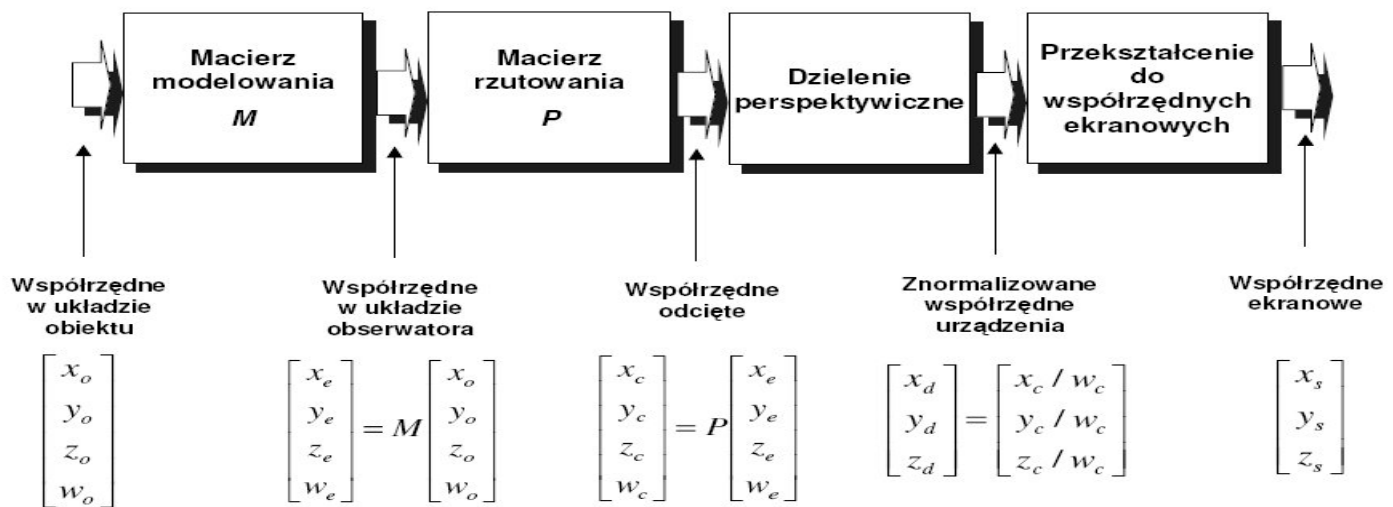
e_2 - oznacza poziom dominującej składowej światła
 e_1 - wypadkowy poziom wszystkich pozostałych, dających światło białe

Nasylenie (czystość) barwy jest tym większe, im większa jest wartość e_2 w stosunku do e_1 . Dla $e_2 = e_1$ mamy nasylenie zerowe, gdy zaś $e_1 = 0$, wynosi ono 100%.

Przeciętnie człowiek rozróżnia ok. 150 czystych barw widma słonecznego

Narysować i opisać model procesu rzutowania

3D



WEJŚCIE: Modele obiektów 3D (układ współrzędnych obiektu)

1. Macierz modelowania sceny: (układ współrzędnych rzeczywistych ziemskich)

Translacja, Skalowanie, Obrót

2. Macierz rzutowania: (układ współrzędnych płaszczyzny rzutowania)

Rzut perspektywiczny, Rzut ortogonalny

3. Przekształcenia normalizujące: (układ współrzędnych urządzenia zobrazowania współrzędne pikselowe)

4. Definiowanie okna zobrazowania (układ współrzędnych obszaru zobrazowania współrzędne ekranowe)

WYJŚCIE: obraz sceny w oknie zobrazowania.

Narysować i opisać model koloru CIE $L^*a^*b^*$. W jakim zakresie def się par. tego modelu

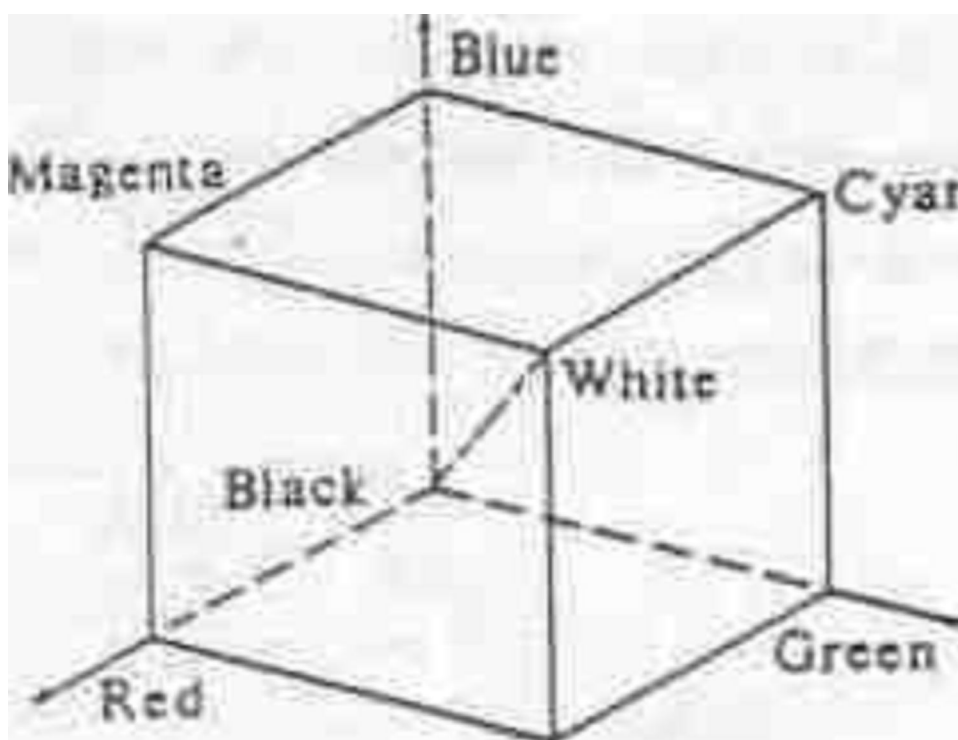
Opracowany przez Międzynarodową Komisję Oświetleniową (CIE) model barw CIE $L^*a^*b^*$ jest najważniejszym modelem w odniesieniu do grafiki komputerowej. Opisywane barwy mieszają się w elipsoidzie o trzech prostopadłych osiach. Wzdłuż osi a barwy przechodzą od zielonej do czerwonej, wzdłuż osi b od żółtej do niebieskiej, w punkcie zbiegu jest biel (szarość). Wzdłuż osi pionowej L określającej jasność, biegną barwy achromatyczne - od bieli do czerni.

Barwy objęte w kuli CIE $L^*a^*b^*$ tworzą najszerszą gamę barw, model został opracowany na bazie modelu CIE Yxy . Zastosowana transformacja współrzędnych umożliwia łatwiejsze wyliczenie odległości pomiędzy dwoma barwami w przestrzeni. Każdy inny model barw da się opisać we współrzędnych Lab , ale nie na odwrót. Czyny to model ten niezależnym od urządzeń wejścia/wyjścia. Model wykorzystywany jest do obliczeń na barwach przez systemy zarządzania barwami CMS.

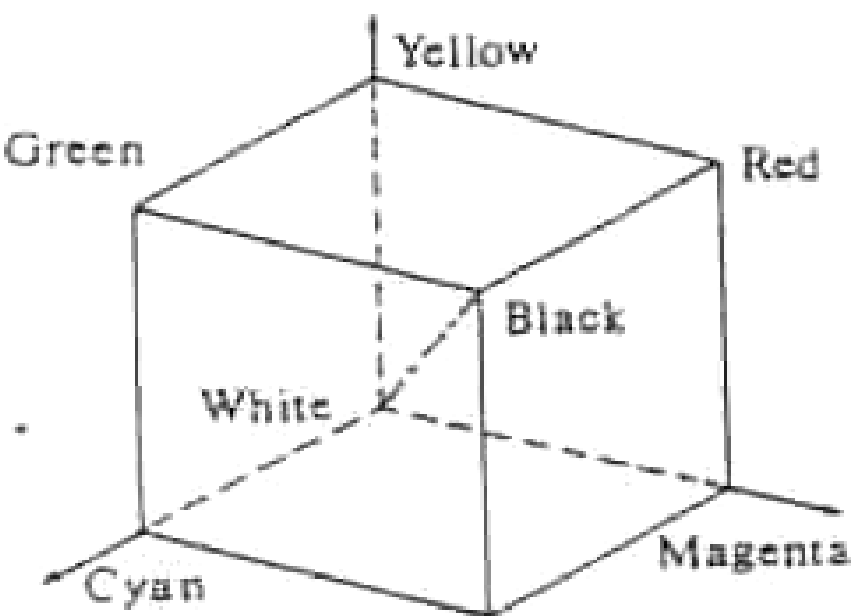
Narysować i opisać model koloru RGB i CMY.

Podać równanie konwersji RGB->CMY

RGB 3 barwy podstawowe: czerwony, zielony i niebieski (Red, Green, Blue) odpowiadają sprzętowemu sposobowi generowania kolorów w monitorze rastrowym. W modelu RGB wszystkim możliwym do uzyskania barwom odpowiadają punkty szescianu jednostkowego, rozpiętego na osiach barw podstawowych. Początek układu $(0,0,0)$ reprezentuje barwę czarna, wierzchołki szescianu leżące na osiach barwy podstawowe, a przeciwległe do nich wierzchołki barwy dopełniające. Przekątna $(0,0,0) \rightarrow (1,1,1)$ odpowiada różnym poziomom szarości, od białego do czarnego. Model ten opisuje addytywny proces otrzymywania barw w wyniku dodawania (mieszania) w różnych proporcjach trzech barw podstawowych.



CMY barwy podstawowe: zielono-niebieska (Cyan), karmazynowa (Magenta) i zolta (Yellow). Stanowią one dopełnienia barw czerwonej, zielonej i niebieskiej i są nazywane podstawowymi barwami subtraktywnymi, gdyż usuwają poprzednie podstawowe barwy ze światła białego – wypadkowa pozostałych odbitych składników odbieramy jako daną barwę. Model CMY jest subtraktywny i służy do określenia kolorów rysunków na ploterach i drukarkach. Można przedstawić go w postaci sześcianu jednostkowego. Osie układu odpowiadają barwom zielono-niebieskiej, karmazynowej i żółtej. Punkt (0,0,0) określa barwę białą, maksymalne (rowne 1) składowe barw podstawowych dają barwę czarną (1,1,1). W praktyce nie absorbują one łącznie wszystkich długości widzialnych fal i światło odbite od powierzchni pokrytej tymi trzema kolorami odbieramy jako ciemnoszare. Dlatego stosuje się druk czterobarwny, dodatkowo używając czarnej farby.



Równanie konwersji RGB -> CMY

$$C = 1 - R$$

$$M = 1 - G$$

$$Y = 1 - B$$

Omówić model koloru CMYK. Podać równanie konwersji CMY -> CMYK

CMYK - (Cyan, Magenta, Yellow, Key Black)

Model ten wykorzystuje zasadę substraktywnego mieszania barw i jest szeroko stosowany w poligrafii.

Przejsie z CMY->CMYK (bezwzględne obliczenia różnic dla składowych C, M, Y):

$$C = C - k * \min(C, M, Y)$$

$$M = M - k * \min(C, M, Y)$$

$$Y = Y - k * \min(C, M, Y)$$

$$K = k * \min(C, M, Y)$$

Gdzie k współczynnik regulacji kontrastu o wartościach z zakresu $\langle 0;1 \rangle$, najczęściej ma wartość 1

Aby przejść od modelu CMYK do CMY należy wykorzystać zależność (by zorski):

$$\{C, M, Y\} = \{C(1-K)+K, M(1-K)+K, Y(1-K)+K\}.$$

Narysować i opisać model koloru HLS. W jakim zakresie def. się par. tego modelu

Każdy kolor definiuje się za pomocą trzech wielkości :

H barwa (ang. Hue)

L jasność (ang. Lightness)

S nasycenie (ang. Saturation)

(H) Barwa związana bezpośrednio z długością fali promieniowania widzialnego. Jest ona

reprezentowana za pomocą okręgu (0° - 360°)

3 kolory podstawowe (R,G,B) i 3 dopełniające (cyan, magenta, yellow) rozmieszczone są na okręgu

tworząc sześciokąt regularny, 0° czerwony, 120°

zielony, 240° niebieski, barwy dopełniające dla

danej otrzymujemy przez zwiększenie kąta o 180°

(L) Jasność - związana ze skalą kolorów

achromatycznych, definiuje wszystkie odcienie

szarości wartości od 0 (kol. czarny) do 1 (kol. biały)

(S) Nasycenie ilość fal o różnej długości

składających się na jeden kolor

kolor czysty, lub w pełni nasycony, to 1

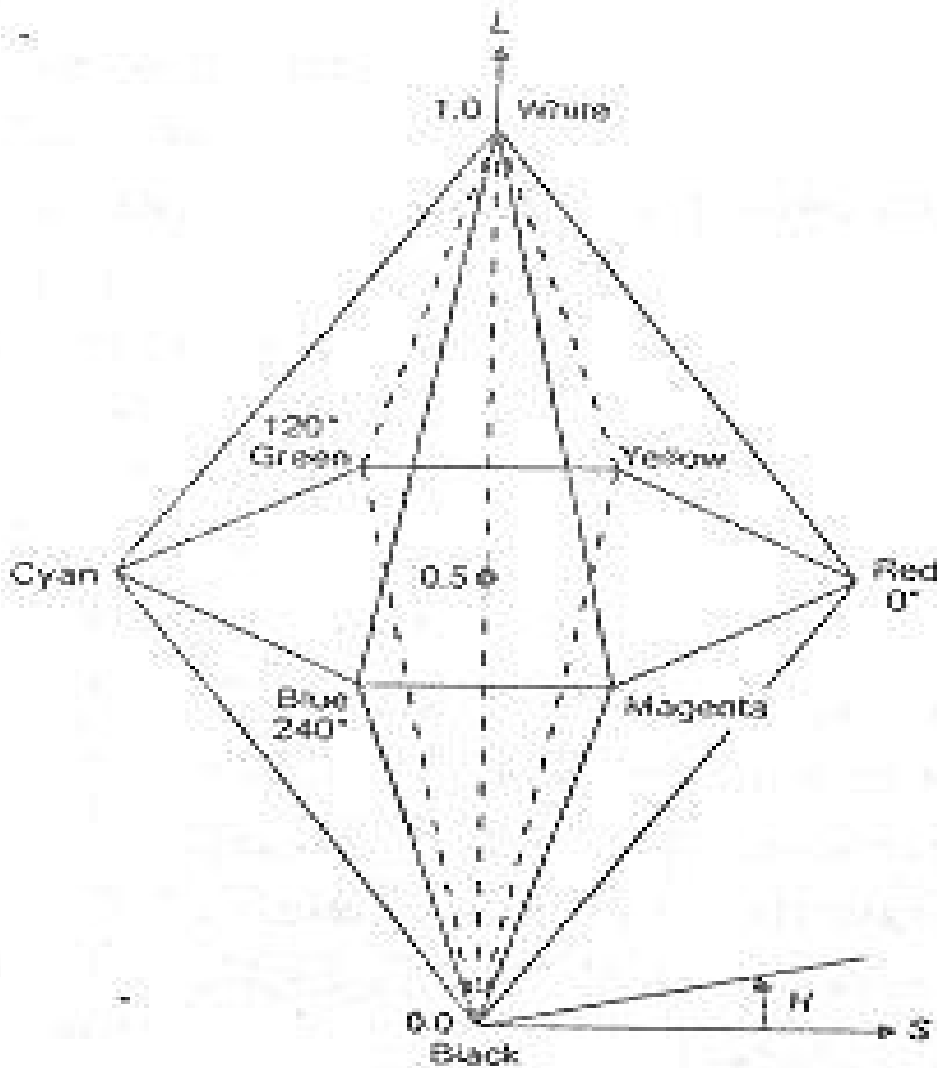
kolor szary utworzony za pomocą fali o szerokim

przedziale długości 0

Barwa w modelu HLS jest wyrażana jako część

ułamekowa jednostkowego kąta pełnego. Dzięki

temu wszystkie kolory mogą być definiowane wektorem o składowych $\langle H, L, S \rangle$ których wartości należą do przedziału $\langle 0; 1 \rangle$ Przykład: Red = $\langle 0.25, 0.5, 1 \rangle$ itd. dla pozostałych kolorów. Model HLS możemy przedstawić jako podwojny stożek:



Barwa (H) mierzona jako wartość kąta wokół osi L
Jasność (L) określona jest jako współrzędna na pionowej osi L
Nasycenie (S) mierzony jako odległość od osi pionowej

Narysować i opisać model koloru HSV. W jakim zakresie def. się par. tego modelu

Model został zaprezentowany przez Smitha, a znany jest jako model HSV od angielskich określeń:

barwy (Hue)

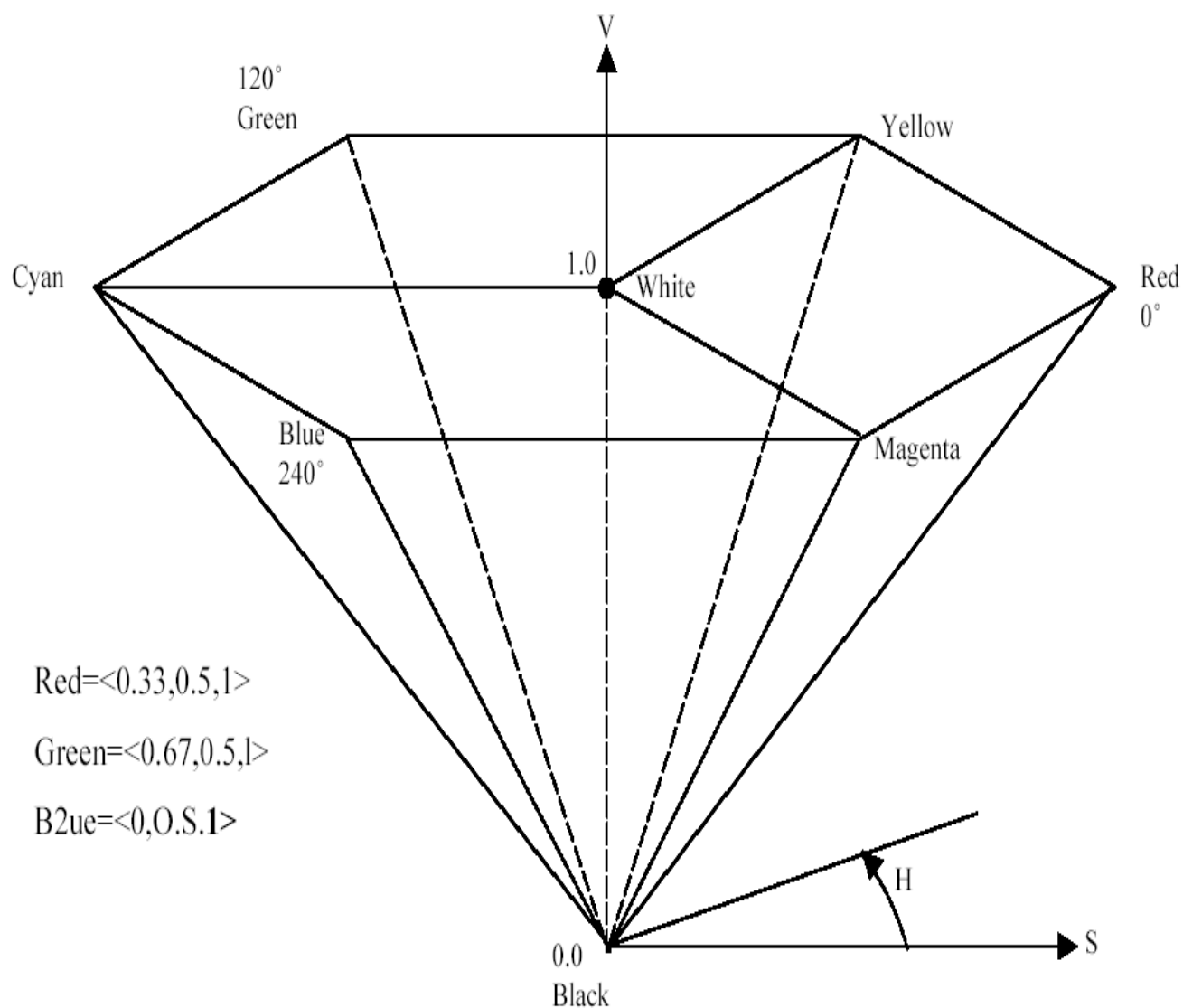
nasycenia (Saturation)

wartości (Value).

Model ten często nosi nazwę HSB, gdzie B oznacza jasność. Przedstawia się go w postaci ostrosłupa prawidłowego o podstawie sześciokątnej. Wierzchołki podstawy odpowiadają barwom spektralnym, wierzchołek ostrosłupa odpowiada barwie czarnej, a spodek wysokości barwie białej.

Barwa mierzona jest kątem β wokół osi V: $\beta=0$ dla barwy czerwonej; $\beta=120$ dla barwy zielonej; $\beta=240$ dla barwy niebieskiej. Barwę dopełniająca do danej otrzymuje się przez zwiększenie β o 180. Wartość nasycenia określa względna odległość punktu od pionowej osi, dla punktów leżących na osi V wartość nasycenia jest równa zero, a dla punktów leżących na

trojkatnych scianach ostroslupa wartosc nasycenia wynosi jeden. Barwa ma najmniejsza wartosc $V=0$ w wierzcholku ostroslupa, maksymalna wartosc $V=1$ osiaga w punktach podstawy. Okreslanie barwy uzytkownik moze rozpoczac od podania kata β i $V=S=1$; wybiera on tym samym czysta barwe spektralna. Nastepnie sciemnia ja (dodaje barwe czarna zmniejszajac odpowiednio wartosc V) lub rozjasnia (dodaje barwe biala zmniejszajac wartosc nasycenia S).



Operacje w przestrzeni kolorów RGB

zmiana jasności

filtr kolorowy

zwiększenie kontrastu

utrata ostrości

wygladzanie (smoothing)

uwypuklenie (relief)

redukcja liczby barw

zmiana nasycenia

zmiana barwy

przejsie na obraz monochromatyczny

progowanie

fotoinwersja

Cieniowanie powierzchni metodą Gourauda

- 1) Wyznaczenie barwy w wierzchołkach wielokąta (np. trójkąta)
 - wyznaczenie wektorów normalnych do wszystkich wierzchołków jako średniej arytmetycznej wektorów normalnych wszystkich ścian, do których ten wierzchołek należy.
 - na podstawie wektora normalnego wyznaczenie barwy wierzchołka korzystając z wybranego modelu oświetlenia

- 2) Liniowa interpolacja barwy dla wszystkich punktów wielokąta zgodnie z wzorami:

$$I_D = I_A \frac{y_B - y_D}{y_B - y_A} + I_B \frac{y_D - y_A}{y_B - y_A}$$

$$I_F = I_A \frac{y_C - y_F}{y_C - y_A} + I_C \frac{y_F - y_A}{y_C - y_A}$$

$$I_E = I_D \frac{x_F - x_E}{x_F - x_D} + I_F \frac{x_E - x_D}{x_F - x_D}$$

Cieniowanie powierzchni metodą Phong

Polega na interpolacji wektora normalnego dla każdego punktu wielokąta, a następnie na tej podstawie intensywności barwy.

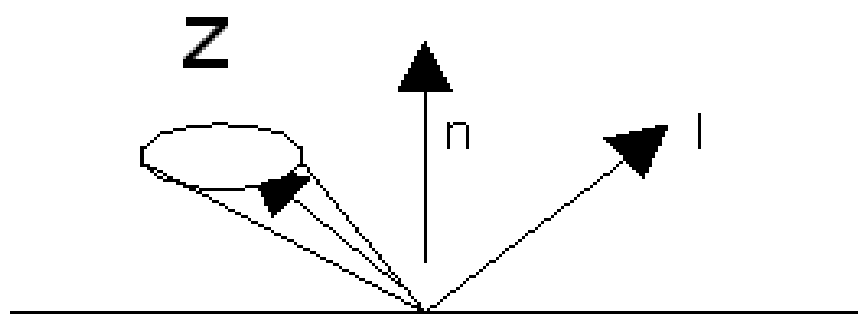
Pierwszy etap cieniowania jest dokładnie taki sam jak w metodzie Gourauda. Metoda ta jest 2x bardziej złożona obliczeniowo od metody Gourauda, przez co wykonanie tego cieniowania jest wykonywane 2x dłużej. Zysk między tymi metodami jest widoczny dla powierzchni lustrzanych.

Model oświetlenia Phong'a

modele przejściowe oświetlenia:

Model Phong'a bardziej skomplikowany, uwzględnia drugi rodzaj odbicia odbicie kierunkowe (zwierciadlane)

Złudzenie odblasku światła na powierzchni obiektów.



Stozek dla powierzchni matowych wezszy niz dla błyszczących

$$I = I_a \cdot k_a + \sum_i \frac{I_i \cdot (k_d \cdot \cos \alpha_i + k_s \cdot \cos^k \alpha_i)}{D + K}$$

k^s współczynnik odbicia kierunkowego;

n współczynnik definiujący właściwości lustrzanej powierzchni; decyduje o średnicy stożka pokazanego na rysunku powyżej. Im bardziej matowa powierzchnia, tym n i podstawa stożka jest

mniejsza, co daje mniejszy kat odbicia kierunkowego.

σ_i - kat pomiędzy kierunkiem odbicia a kierunkiem widzenia powierzchni;

I - intensywnosc swiatla odbitego od powierzchni;

I_α - intensywnosc swiatla rozproszonego posredniego (z wtornych zrodel)

K_α - wspolczynnik odbicia rozproszonego posredniego swiatla (z wtornych zrodel)

I_i - intensywnosc i-tego pierwotnego zrodla swiatla

K_d - wspolczynnik odbicia rozproszonego bezposredniego swiatla (z pierwotnych zrodel)

i - numer pierwotnego zrodla

α_i - kat pomiędzy wektorem normalnym do powierzchni a kierunkiem padania swiatla;

D - odleglosc oswietlanej powierzchni od obserwatora

K - arbitralnie dobierana stala zapewniajaca najlepszy efekt wizualny

Model oświetlenia Halla

modele symulacyjne:

Model Halla uwzględnia nie tylko źródła pierwotne i wtórne, uwzględniają przezroczystość wizualizowanych obiektów. Uwzględniano w tym modelu zależność intensywnego odbitego światła oraz współczynników odbicia od długości fali λ . Opis zachowania się światła na granicy dwóch ośrodków opisują wprowadzone współczynniki Fresnela $F_s(\lambda)$ i $F_t(\lambda)$.

$$I(\lambda) = k_a(\lambda) * I_a + k_d(\lambda) * \sum_i \cos \alpha_i * I_i + k_s * F_s(\lambda) * \sum_i (\cos \phi_i)^n * I_i + k_t * F_t(\lambda) * \sum_i (\cos \phi_i')^{n'} * I_i + k_s * F_s(\lambda) * I_s * T_s^{ds} + k_t * F_t(\lambda) * I_t * T_t^{dt}$$

λ - długość fali światła

$I(\lambda)$ - intensywność światła odbitego od powierzchni w kierunku obserwatora

I_a - stała intensywność światła otaczającego

I_i - intensywność i-tego źródła światła

I_s - oświetlenie pośrednie z kierunku, w którym promień został odbity

I_t - oświetlenie pośrednie z kierunku, w którym promień został załamany

α_i - kąt pomiędzy wektorem normalnym do pow. (N), a kierunkiem padania światła

Φ_i - kat pomiędzy wektorem normalnym do mikropowierzchni odbijającej (H) a wektorem (N)

Φ_i' - kat pomiędzy wektorem normalnym do mikropowierzchni zalamującej (H') a wektorem (N)

T_s^{ds} - współczynnik transmisji osrodka, w którym poruszał się promień odbity, podniesiony do potegi równej odległości, która promień przebył w osrodku

T_t^{ds} - współczynnik transmisji osrodka, w którym poruszał się promień zalamany, podniesiony do potegi równej odległości, która promień przebył w osrodku

k_d - współczynnik związany z powierzchnią, określający ilość światła rozproszonego

k_s - współczynnik związany z powierzchnią, określający ilość światła kierunkowo odbitego

k_t - współczynnik związany z powierzchnią, określający ilość światła kierunkowo zalamanego

k_a - współczynnik określający ilość światła otaczającego odbitego od powierzchni

n - współczynnik określający właściwości mikropowierzchni odbijającej

n' - współczynnik określający właściwości mikropowierzchni zalamującej

Równanie renderingu

Sformalizowanie założeń dotyczących modelu odbicia doprowadziły do powstania

Równania wizualizacji (renderingu Kajiya 1986)

Równanie wizualizacji propagacja energii świetlnej, oparte na równaniu transferu ciepła i przepływu energii droga promieniowania.

Wielokrotne rozwiązanie równania wizualizacji dla każdego punktu powierzchni prowadzi do syntezy obrazu realistycznego:

$$I(x, x') = g(x, x') \cdot [e(x, x') + \int p(x, x', x'') \cdot I(x', x'') dx'']$$

Gdzie:

$I(x, x')$ suma intensywności światła emitowanego i odbijanego w punkcie x w kierunku x'

$g(x, x')$ czynnik geometryczny

$e(x, x')$ emisja światła z punktu x w kierunku x'

$p(x, x', x'')$ współczynnik odbicia

kierunkowego światła $I(x, x')$ w punkcie x ;

kierunek padania światła określa położenie

punktu x'' , natomiast kierunek odbicia wyznacza

lokalizację punktu x

Podstawowy empiryczny model oświetlenia

Model lokalny - rozpatruje wpływ pierwotnych źródeł światła na oświetlenie powierzchni.

Model globalny - uwzględnia zarówno pierwotne jak i wtórne źródła światła.

modele empiryczne oświetlenia uwzględniają jedynie pierwotne źródła światła i wprowadzają empiryczny, stały współczynnik, kompensujący w pewnym stopniu brak obliczeń dla wtórnych źródeł światła

(Lambertowska charakterystyka światła odbitego).

Analiza rozchodzenia się światła w środowisku ogranicza się do tłumienia jego energii proporcjonalnie do odległości pomiędzy powierzchnią odbijającą a obserwatorem;

$$I = I_{\alpha} K_{\alpha} + \sum_i \frac{I_i \cdot K_d \cdot \cos \alpha}{D + K}$$

I - intensywność światła odbitego od powierzchni;

I_{α} - intensywność światła rozproszonego pośredniego (z wtórnych źródeł)

K_{α} - współczynnik odbicia rozproszonego pośredniego światła (z wtórnych źródeł)

I_i - intensywność i -tego pierwotnego źródła światła

K_d - współczynnik odbicia rozproszonego bezpośredniego światła (z pierwotnych źródeł)

i - numer pierwotnego źródła

α_i - kat pomiędzy wektorem normalnym do powierzchni a kierunkiem padania światła;

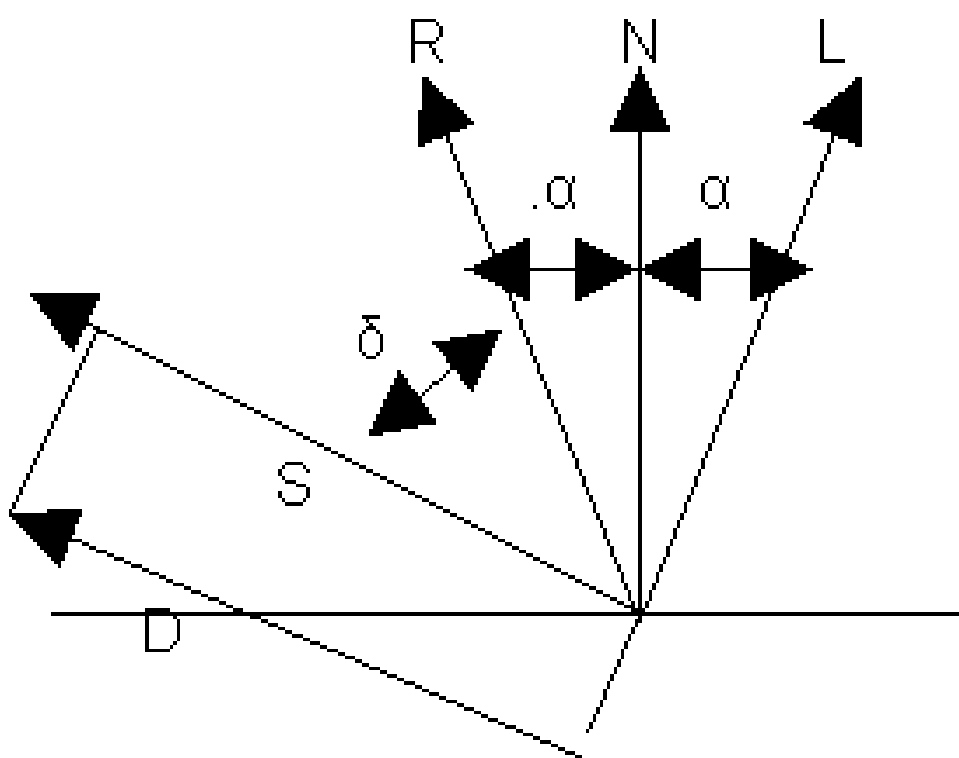
D - odległość oświetlanej powierzchni od obserwatora

K - arbitralnie dobierana stała zapewniająca najlepszy efekt wizualny

Te współczynniki k to są charakterystyczne dla danej powierzchni i przyjmują wartości z przedziału $\langle 0,1 \rangle$.

Interpretacja geometryczna

Oznaczenia przyjęte w empirycznym modelu oświetlenia:



N - wektor prostopadły do powierzchni

L - wektor w kierunku źródła światła

R - wektor w kierunku światła odbitego

S - wektor w kierunku oka obserwatora;

Segmentacja

Segmentacja - identyfikowanie obszarów obrazu, których wygląd jest dla obserwatora jednolity.

- identyfikowanie oznacza jednoznaczne ustalenie, do jakiego obszaru (składowej pierwotnej) należy dany punkt obrazu;
- obszary obrazu są to składowe pierwotne obrazu, mogą nimi być: płaszczyzny o zdefiniowanych kształtach, krawędzie, kontury itp.;
- jednolitość może być definiowana na różne sposoby (poziom jasności, barwa, nasycenie koloru, tekstura itp.).

Klasy algorytmów segmentacji (podział ze względu na metodę)

1. Segmentacja przez progowanie
2. Segmentacja przez wykrywanie krawędzi
3. Segmentacja przez rozrost obszaru

Segmentacja jest jednym z najbardziej intensywnie studiowanych zagadnień w procesie rozpoznawania obrazu i widzenia komputerowego.

Segmentacja przez progowanie
Wartość jasności każdego elementu obrazu (piksela) jest porównywana z wartością progową. Element jest przydzielany do jednej z dwóch kategorii: o wartości progowej przekroczonej lub nieprzekroczonej (binaryzacja obrazu). Wyboru wartości progowej dokonuje się na podstawie histogramu.

Segmentacja przez wykrywanie krawędzi Krawędzie między obszarami wykrywane są poprzez zastosowanie operatora detekcji krawędzi. Najczęściej stosowane operatory to:

- filtr Sobela;
- filtr Prewitta;
- operator Laplace'a;
- gradient w kierunku poziomym lub pionowym.

Wynik operacji podlega progowaniu. Elementy obrazu, których wartość po operacji przekroczy wartość progowa kwalifikowana jest jako punkt należący do krawędzi. Ostatnim etapem metody jest połączenie punktów zidentyfikowanych jako krawędzi w celu uformowania krzywej zamkniętej otaczającej obszar.

Segmentacja przez rozrost obszaru Metody segmentacji przez progowanie i wykrywanie krawędzi zajmują się różnicami między wartościami elementów obrazu. W metodzie segmentacji przez rozrost działa się w przestrzeni obrazu poszukując grup elementów o zbliżonej wartości.

Najprostszą postać metody to rozpoczęcie algorytmu od jednego arbitralnie wybranego elementu obrazu (ziarna) i sprawdzanie czy piksele przylegające (sasiadujące) mają podobną jasność spełniają test jednolitości. Jeżeli tak, to są one grupowane w obszar. Następnie sprawdzane są kolejne sasiadujące piksele. W ten sposób powstają coraz większe, rozrastające się obszary.

Przedstawić za pomocą algorytmu zasadę wyświetlania obrazu w urządzeniach realizujących grafiki wektorowa

- urządzenia realizujące grafiki wektorowa tworzą obrazy klasy 3 i 4

- rozkazy elementarne:

$p(x, y)$ ustaw plamkę światła w punkcie (x, y)

$s(z)$ ustaw jasność zgodnie z wartością z

- sekwencja rozkazów odpowiada wyświetleniu obiektu

- program sterujący wyświetleniem obrazu:

START: //S1...Sn - sekwencje rozkazów

S1

:

Sn

IF brak przerwania THEN GOTO START

ELSE zmień sekwencje

GOTO START

END

Aby zmodyfikować obraz należy zmienić odpowiednią sekwencję. Zmiana obrazu może być sygnalizowana poprzez przerwanie. Za każdą petlą odnawiany jest ekran. Częstotliwość odnawiania jest odwrotnie proporcjonalna do długości petli. Jeśli czas ten jest dłuższy od czasu fluorescencji ekranu to następuje migotanie i rozmycie ekranu. Stąd istnieje górna granica długości petli, a tym samym złożoności obrazu.

Przedstawić za pomocą algorytmu zasadę wyświetlania obrazu w urządzeniach realizujących grafiki rastrowa

- urządzenia realizujące grafiki rastrowa tworzą obrazy klasy 1 i 2, mogą również symulować grafiki wektorowe

- parametry punktów ekranu są przechowywane w pamięci

- operacje elementarne:

read (I, x, y, z) czytaj komórkę pamięci I oraz określ z na podstawie zawartości I (x i y są określone adresem I)

write (x, y, z) pisz na ekranie w punkcie o współrzędnych (x, y) wartość luminescencji określoną przez z

- główna pętla wyświetlania:

START:

FOR I= 1 TO liczba punktów ekranu DO

BEGIN

read (I, x, y, z)

write (x, y, z)

END;

GOTO START

END.

Algorytm wypełniania obrazu przez sianie.

Wnętrze obszaru jest zbiorem czterospojnym, a ograniczający je brzeg osmiospojnym. Brzeg obszaru jest narysowany kolorem cb. Wnętrze obszaru wypełniamy kolorem cn. Znamy położenie ziarna, czyli piksela leżącego wewnątrz obszaru. Od tego piksela rozpoczynamy działanie algorytmu wypełniania. Piksel wypełniamy nowym kolorem i następnie sprawdzamy w czterech kierunkach czy piksele należą do wnętrza obszaru i czy nie zostały jeszcze wypełnione nowym kolorem. Jest to algorytm rekurencyjny, zwany także wypełnianiem przez sianie. Rekurencyjny opis przedkłada się zazwyczaj na rekurencyjną implementację algorytmu, co jest jego główną wadą (łatwo można doprowadzić do przepelnienia stosu). Drugą wadą jest rozrzutność algorytmu objawiająca się wielokrotnym badaniem koloru tego samego piksela. W niektórych przypadkach kolor pojedynczego piksela badany jest nawet pięciokrotnie.

Algorytm wypelniania przez sianie

- x,y współrzedne piksela;
- color kolor sprawdzanego piksela;
- cb kolor brzegu obszaru;
- cn kolor wypelniania obszaru.

```
procedure fill4(integer: x, y, cb, cn)
```

```
begin
```

```
  getpixel(x, y, color);
```

```
  if (color  $\neq$  cb) and (color  $\neq$  cn) then begin
```

```
    putpixel(x, y, cn);
```

```
    fill4(x, y-1, cb, cn);
```

```
    fill4(x, y+1, cb, cn);
```

```
    fill4(x-1, y, cb, cn);
```

```
    fill4(x+1, y, cb, cn);
```

```
  end;
```

```
end.
```

Algorytm wypełniania wielokątów.

przeładowane są kolejne linie poziome wyznaczone są punkty przecięcia linii z krawędziami wielokąta wypełniane są segmenty pikseli leżące wewnątrz wielokąta punkty przecięcia w kolejnych liniach obliczane są metodą przyrostową (iteracyjnie)

Niech: $y_{\min} = \text{Int}(y_1)$; $y_{\max} = \text{Int}(y_3)$;

gdzie: $\text{Int}(r)$ - zaokrąglenie liczby rzeczywistej r do najbliższej liczby całkowitej.

Wyznaczamy współczynniki kierunkowe nachylenia boków trapezu:

$c_l = (x_4 - x_1) / (y_4 - y_1)$; $c_p = (x_3 - x_2) / (y_3 - y_2)$;

Przy tych założeniach algorytm wypełniania trapezu jest następujący:

start:

dla $y = y_{\min}, y_{\min} + 1, \dots, y_{\max}$ wykonaj:

1). wyznacz przecięcia x_l i x_p linii poziomej y z prostymi:

$x_l = x_1 + (y - y_1)c_l$ i $x_p = x_2 + (y - y_2)c_p$;

2). wypełnij nowym kolorem (wzorcem) piksele leżące na tej linii

od $\text{Int}(x_l)$ do $\text{Int}(x_p)$;

end.