

Wykład 4

- Równanie barwy – str 2
- Model RGB – str 5
- Model CMY – str 8
- Model CMYK – str 10
- Przykład 3 (RGB na CMY) – str 11
- Przykład 5 (RGB NA CMYK) – str 12

Wykład 5

- Modele zależne i niezależne od urządzenia – str 14
- Model CIE Lab – str 15
- Model CIE XYZ – str 18
- Model HSL – str 19
- Model HSV – str 22
- Przykład 1 (RGB NA HSV) – str 25
- Przykład 2 (RGB na HSL) – str 27
- Przykład 3 (RGB na CIE XYZ) – str 29
- Przykład 4 (RGB na CIE Lab) – str 31

Wykład 6

- Separacja RGB – str 33
- Zasady doboru barw – str 35
- Bitmapa i jej charakterystyka – str 37
- Kryteria doboru rozdzielczości i głębi – str 43
- Formaty plików – str 45
- Kryteria doboru formatu – str 51

Wykład 7

- Przetwarzanie grafiki rastrowej – str 53
- Przekształcenia geometryczne - zmiana rozdzielczości i deformacja – str 54
- Przekształcenia punktowe/bezkontekstowe - operacje arytmetyczne – str 59
- Przekształcenia punktowe/bezkontekstowe - histogram – str 64
- Histogram obrazów monochromatycznych – str 72
- Histogram obrazów barwnych (RGB) – str 74

Wykład 8

- Przekształcenia kontekstowe - filtracja – str 76
- Algorytmy grafiki rastrowej – str 85

Wykład 9

- Grafika wektorowa – str 89
- Krzywe Béziera - wyliczenia numeryczne – str 92
- Krzywe Béziera - wizualizacja – str 100
- Grafika wektorowa cd – str 104
- Zalety i wady – str 112
- Rasteryzacja i wektoryzacja – str 114
- Fonty – str 120
- Wypełnianie przez spójność - wyliczenie i wizualizacja (przykład uzupełniający) – str 123

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



Patryk Obuchowicz -



online M&J

Dashboard > My courses > GU - wykład > Wykład 4 (23.03.2020) > Równanie barwy

Równanie barwy

Wszystkie systemy kolorów opierają się na trzech odrębnie postrzeganych parametrach:

- **barwa (odcień barwy),**
- **jasność,**
- **nasycenie.**

Kolor jest zjawiskiem psychofizycznym zdefiniowanym przez ww. trzy parametry.



Kilka **barw** wybranych z widma: od czerwonej do fioletowej



Próbki kolorów o tej samej barwie czerwonej ale różnej **jasności**



Próbki kolorów o tej samej barwie i jasności, ale o różnym **nasyceniu**

Przy opracowaniu modeli barw korzysta się z nie tylko z teorii fizycznej, ale także z wiedzy dotyczącej percepcji zmysłowej.

Opracowując modele użytkowe przede wszystkim należy wziąć pod uwagę:

- źródła światła emitowanego** (zasada addytywności, zwiększenie jasności obrazu wraz z wzrostem intensywności barw)
- źródła światła odbitego** (zasada subtraktywności/dopełniania, zwiększenie intensywności barwy spowoduje obniżenie jasności światła odbitego)

- a) **funkcji zależnej od długości fali** (widmowa gęstość mocy, ang. Spectral Power Distribution)
 $P(\lambda)$
- b) **ilości energii emitowanej przez źródło** (często tylko w zakresie od 400nm do 700nm)

Z kolei każda powierzchnia może odbijać lub absorbować światło w zależności od:

- a) **pigmentacji powierzchni,**
- b) **stopnia porowatości powierzchni.**

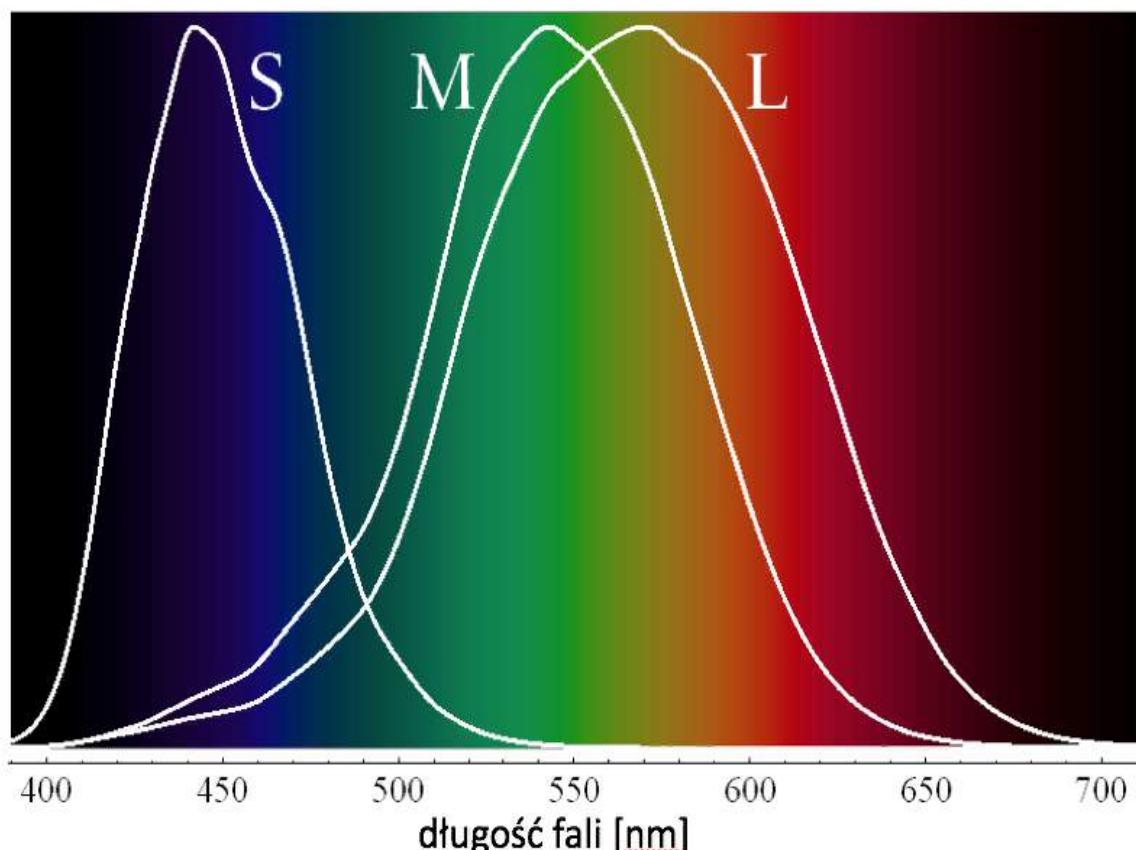
Własności powierzchni scharakteryzować można za pomocą:

- a) **funkcji zależnej od długości fali** określającej stosunek natężenia światła padającego do odbitego
- b) **dwukierunkowej funkcji rozkładu** odbicia charakteryzującej powierzchnię $E(\lambda)$

Promieniowanie docierające do oka tzw. **sygnał barwny pobudzający czopki siatkówki**, zależy od zarówno charakterystyki oświetlenia jak i własności materiału:

$$C(\lambda) = E(\lambda)P(\lambda)$$

Odpowiedź czopków zależy nie tylko od sygnału pobudzającego czopki ale również od **czułości samych czopków**:



- $S(\lambda)$ - **czułość czopków typu S** (ok. 420 nm wrażenie barwy niebieskiej),
- $M(\lambda)$ - **czułość czopków typu M** (ok. 530 nm wrażenie barwy zielonej),
- $L(\lambda)$ - **czułość czopków typu L** (ok. 700 nm wrażenie barw czerwonej).

Całkowita odpowiedź czopków jest rezultatem sumarycznym, a tryplet odpowiedzi czopków:

$$S = \int_{\lambda} C(\lambda)S(\lambda)d\lambda \quad M = \int_{\lambda} C(\lambda)M(\lambda)d\lambda \quad L = \int_{\lambda} C(\lambda)L(\lambda)d\lambda$$

stanowi podstawę widzenia barwnego a zarazem podstawę konstrukcji wszystkich modeli barwy, które z tego właśnie względu są trójwymiarowe.

Metoda matematycznego, numerycznego (obiektywnego) opisu barwy

Równanie barwy wyraża fakt zrównania bodźca światlnego o zadanym rozkładzie spektralnym S przez mieszaninę trzech niezależnych bodźców podstawowych A, B i C:

$$S(\lambda) = \frac{a}{a+b+c}A(\lambda) + \frac{b}{a+b+c}B(\lambda) + \frac{C}{a+b+c}C(\lambda) = a'A(\lambda) + b'B(\lambda) + c'C(\lambda)$$

$$a' + b' + c' = 1$$

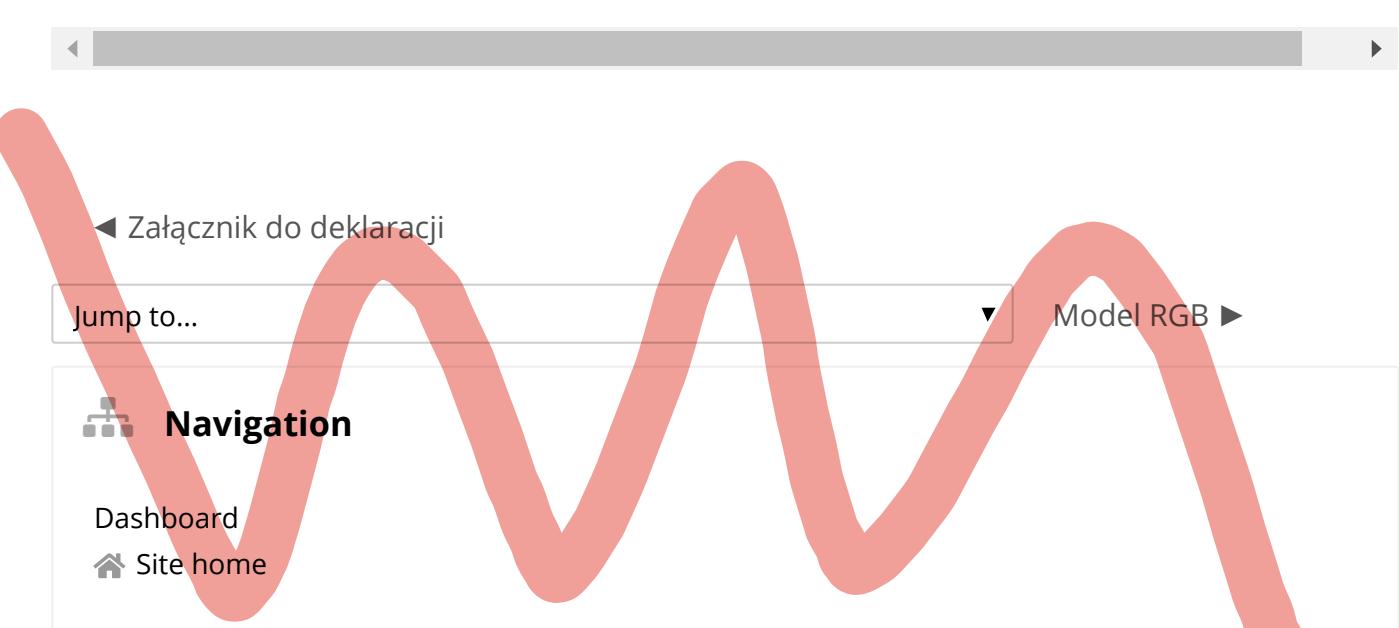
Współczynnik: a, b i c nazywane są **składowymi trójchromatycznymi**, natomiast unormowane wartości a', b' i c' nazywane są **współrzędnymi trójchromatycznymi**.

Do określenia chromatyczności bodźca (czyli barwy) wystarczy znajomość trzech składowych lub dwóch współrzędnych trójchromatycznych.

Brak możliwości jednoznacznego opisu barw w sposób użyteczny we wszystkich zastosowaniach spowodował, że wprowadzono wiele różnych modeli barw, w tym wiele dla konkretnych zastosowań.

Przestrzenie barwne konstruowane w oparciu o powyższe równanie barwy podzielić można na trzy podstawowe kategorie:

- **kubiczne** (np. RGB, CMYK),
- **walcowe** (np. HSV, HSL),
- **barw przeciwnostnych, kanałów przeciwnostnych** (np. CIE XYZ, CIE Lab).



ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



1 Patryk Obuchowicz



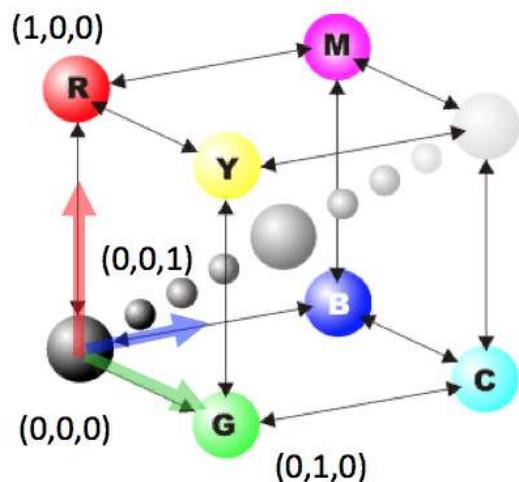
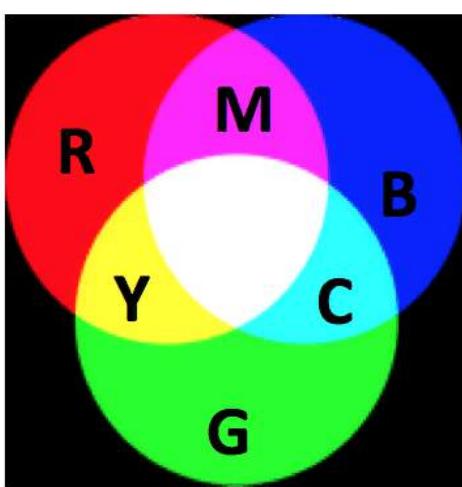
online M&J

Dashboard > My courses > GU - wykład > Wykład 4 (23.03.2020) > Model RGB

Model RGB

Cechy podstawowe:

- opisuje powstawanie kolorów światła biegącego do oka bezpośrednio ze źródła, czyli opisuje metodę powstawania koloru światła
- jest modelem addytywnym - pozwala na addytywne mieszanie barw podstawowych (R-czerwony, G-zielony, B-niebieski);
- mieszanie parami powoduje powstawanie barw dopełniających (C-cyjan, M-magenta, Y-żółty, natomiast równocenne zmieszanie wszystkich trzech barw podstawowych powoduje powstanie barwy białej; brak jakichkolwiek składowych podstawowych powoduje wrażenie barwy czarnej).



Model RGB to model bezpośrednio wywodzący się z teorii Younga-Hemholtza

- czopki niebieskie, zielone i czerwone są uporządkowane wg. maksimów ich czułości spektralnej: krótkie (S), średnie (M) i długie (L).
- mikrospektrofotometria wykazała S, M, L nie odpowiadają bezpośrednio barwie niebieskiej, zielonej i czerwonej, więc nie jest to model bezpośrednio oparty na rodzajach czopków w ludzkim oku.



- przestrzeń barw jest szeroka pozwala zakodować $2^{24} = 16.7$ mln różnych barw (człowiek rozróżnia ok. 7-10 mln barw)
- dostarcza żywych i intensywnych barw

Wady:

- nie jest jednorodny percepcyjnie (zmiany barw nie są proporcjonalne do zmian wartości)
- nie uwzględnia nielinowości procesu percepcji
- nie uwzględnia ciągłości zmiany barw

Zastosowania:

Model RGB stosuje się do opisu obrazu wyświetlanego na monitorach CRT, LCD, ekranach smartfonów, tabletów itp.

Model RGB jest wykorzystywany na stronach i w aplikacjach webowych.

Realizacja modelu:

Model RGB można przedstawić za pomocą sześciadanu (współrzędne kartezjańskie), z wersorami R, G oraz B.

Składowe R, G, B są ortogonalne, niezależne od siebie.

Paleta:

Poszczególne cyfrowe realizacje modelu RGB przewidują ustaloną liczbę bitów do opisu udziału każdego składnika w barwie opisywaną przez liczby całkowite z zakresu $<0, 2^{\text{liczba bitów}} - 1>$

8 bitowy RGB

- 3 bity = 8 poziomów dla R oraz G
- 2 bity = 4 poziomy dla B

Łącznie 256 = 8 × 8 × 4 kolorów

16 bitowy RGB

- 4 bity dla R, G oraz B oraz 4 dla przezroczystości

Łącznie 4096 = 16 × 16 × 16 kolorów

24 bitowy RGB

- 256 poziomów na każdy kanał R,G,B

Łącznie 16.777.216 = 256 × 256 × 256 kolorów

32 bitowy RGBA

- 24 - bitowy R,G,B + 8 - bitów na kanał alfa

48 bitowy RGB

- 65536 poziomów na każdy kanał R,G,B

Mieszanie barw i kodowanie na stronach www:



W systemie RGB (red, green, blue) każdy kolor jest złożeniem kolorów czystych, a udział danego składnika wyrażany jest w procentach lub za pomocą liczb rzeczywistych z przedziału (0.0, 1.0)

- 0.0 - brak danego składnika
- 1.0 - maksymalny udział

Przykłady:

- (1.0, 0.0, 0.0) - czerwony
- (0.0, 1.0, 0.0) - zielony
- (0.0, 0.0, 1.0) - niebieski
- (1.0, 1.0, 1.0) - biały
- (0, 0, 0) - czarny
- (0.208, 0.184, 0.894) – błękit
- (0.672, 0.672, 0.672) - szary

Na stronach www stosuje się zapis bazujący na 24-bitowym modelu RGB, który przewiduje dla każdego składnika barwy liczbę 8-bitową, (l. całkowitą z zakresu 0–255):

Przykłady:

- czerwony (255,0,0)
- biały (255, 255, 255)
- niebieski (0, 0, 255)
- błękit (53, 47, 226)
- szary (171, 171, 171)

RGBA - RGB rozszerzony o kanał alfa (IE9+, Firefox 3+, Chrome, Safari, Opera 10)

Przykłady:

```
#p1 {background-color: rgba(255, 0, 0, 0.3); /* red with opacity */  
#p2 {background-color: rgba(0, 255, 0, 0.3); /* green with opacity */  
#p3 {background-color: rgba(0, 0, 255, 0.3); /* blue with opacity */
```

HSL (IE9+, Firefox, Chrome, Safari, Opera 10+) z parametrami (hue, saturation, lightness)

Przykłady:

```
#p1 {background-color: hsl(120, 100%, 50%); /* green */  
#p2 {background-color: hsl(120, 100%, 75%); /* light green */  
#p3 {background-color: hsl(120, 100%, 25%); /* dark green */  
#p4 {background-color: hsl(120, 60%, 70%); /* pastel green */
```

HSLA (IE9+, Firefox 3+, Chrome, Safari, Opera 10+) z parametrami (hue, saturation, lightness, alpha), gdzie alpha 0.0 (przezroczysty) do 1.0 (nieprzezroczysty).

Przykłady:

```
#p1 {background-color: hsla(120, 100%, 50%, 0.3); /* green with opacity */
```



ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



Patryk Obuchowicz -



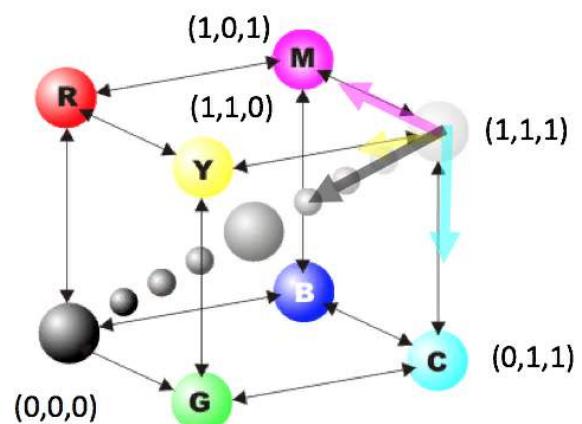
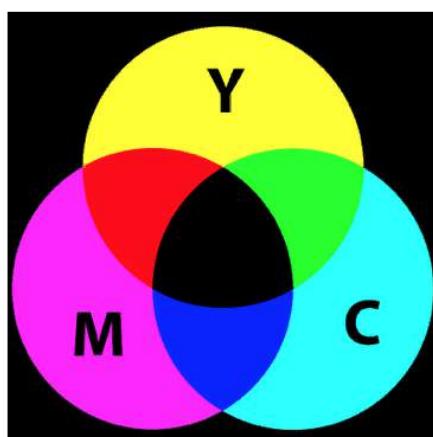
online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 4 (23.03.2020) > Model CMY

Model CMY

Cechy podstawowe modelu CMY:

- opisuje powstawanie kolorów światła odbijającego się od powierzchni, czyli opisuje metodę mieszania barwników;
- jest **modelem subtraktynym** - barwy powstają poprzez selektywne usunięcie wybranego zakresu widma światła białego (filtracja, absorpcja);
- warunkiem koniecznym jest obecność światła białego oraz substancji o selektywnej absorpcji promieniowania światlnego;
- subtraktynne mieszanie barw dopełniających powoduje powstawanie barw podstawowych oraz czerni.
- barwami podstawowymi są barwy dopełniające barw podstawowych R, G, B: cyjan (niebiesko-zielona) C, magenta (purpura) M, żółta (niezbyt jaskrawa) Y



Model CMY to model bezpośrednio wywodzący się z teorii Heringa.

Zalety:

- dogodny dla poligrafii

Wady:

- przestrzeń barw stosunkowo wąska (tusze).

- kolory wyblakłe w porównaniu do RGB,
- nie jest jednorodny percepcyjnie (zmiany barw nie są proporcjonalne do zmian wartości)
- nie uwzględnia nieliniowości procesu percepcji
- nie uwzględnia ciągłości zmiany barw

Mimo, że RGB i CMYK wydają się być zupełnym przeciwnieństwem, uzupełniają się.

Zastosowania:

Model CMY stosuje się do opisu obrazu dla poligrafii, przemysłu tekstylnego, farbiarskiego itp.

Model CMY nie jest wykorzystywany ani na stronach www ani w aplikacjach webowych, natomiast jest modelem powszechnie stosowanym w druku.

Realizacja modelu:

Model CMY można przedstawić za pomocą sześcianu (współrzędne kartezjańskie), z wersorami C, M, Y.

Paleta:

Analogiczna jak w przypadku modelu RGB, ale dostępność barwników ogranicza ją znaczco w praktyce.

Konwersja:

Model CMY jest teoretycznie w pełni konwertowalny do RGB i na odwrót.

Wzory do konwersji:

- z CMY na RGB

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} C \\ M \\ Y \end{bmatrix}$$

- z RGB na CMY

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

◀ Przykład 2

Przejdz do...

Przykład 3 ▶

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



Patryk Obuchowicz -



online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 4 (23.03.2020) > Model CMY(K)

Model CMY(K)

Model CMY(K) jest modyfikacją CMY dokonaną ze względów praktycznych.

Nazwa modelu to CMY(K) czyli Cyan (C) + Magenta (M) + Yellow (Y) + K (black):

Cechy podstawowe modelu:

- drukowanie z użyciem mieszania jest kosztowne, a triada C+M+Y pozwala uzyskać w praktyce brudny brąz a nie czarny,
- dodanie dodatkowej komponenty czerni (K) powoduje, że jest ona bardziej nasyciona i zapewnia lepszy kontrast.

Zalety:

Druk grafiki wymagającej użycia 100% Cyjan, 50% Magenta i 70% Yellow po dodaniu dodatkowego barwnika czerni 50% wymaga użycia jedynie 50% Cyjan i 20% Yellow a więc jest tańszy.

Wady:

- nie jest to model 3D,
- sposób wyznaczania wartości dla kanału K jest niejednoznaczny,
- paleta w praktyce ma ok. 1mln. barw

Model CMY(K) jest teoretycznie w pełni konwertowalny do RGB i na odwrót.

Wzory konwersji:

- z CMY(K) na RGB, wartości CMY(K) z zakresu <0,0,1,0> natomiast RGB z zakresu <0,255>

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 255(1-C)(1-K) \\ 255(1-M)(1-K) \\ 255(1-Y)(1-K) \end{bmatrix}$$



ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



1

Patryk Obuchowicz -



online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 4 (23.03.2020) > Przykład 3

Przykład 3

Kolor biały ma w przestrzeni RGB składowe (255,255,255).

Jaki będzie kod tego koloru w CMY?

**Rozwiązanie:**

Składowe wynoszą:

$$R=255$$

$$G=255$$

$$B=255$$

$$R'=255/255=1$$

$$G'=255/255=1$$

$$B'=255/255=1$$

Dokonujemy konwersji:

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1-1 \\ 1-1 \\ 1-1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$C=0, M=0, Y=0.$$

Kolor biały w CMY kodowany jest jako (0,0,0).



ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



Patryk Obuchowicz -



online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 4 (23.03.2020) > Przykład 5

Przykład 5

Kolor tła na stronie internetowej ma składowe w zapisie dziesiętnym:

- czerwony = 27
- zielony = 90
- niebieski = 104

Jaka będzie deklaracja tego koloru w CMY(K)?



Rozwiązanie:

Składowe wynoszą:

RGB (27,90,104)

R=27 G=90 B=104

Dokonujemy konwersji:

$$R' = R/255 = 27/255$$

$$G' = G/255 = 90/255$$

$$B' = B/255 = 104/255$$

$$K = 1 - \max(R', G', B') = 0.592$$

$$\begin{bmatrix} C \\ M \\ Y \\ K \end{bmatrix} = \begin{bmatrix} \frac{1 - R' - K}{1 - K} \\ \frac{1 - G' - K}{1 - K} \\ \frac{1 - B' - K}{1 - K} \\ \frac{1 - B' - K}{1 - K} \end{bmatrix} = \begin{bmatrix} 0.740 \\ 0.134 \\ 0 \\ 0.592 \end{bmatrix}$$



◀ Model CMY(K)

Przejdz do...



Przykład 6 ►

Nawigacja

Kokpit

🏠 Strona główna

Strony

Moje kursy

MAD - wykład

MAD - ćwiczenia

GU - wykład

Uczestnicy

☰ Oceny

GRAFIKA UŻYTKOWA - WYKŁAD

Informacje

Wykład 1

Wykład 2

Wykład 3

Wykład 4 (23.03.2020)

📄 Równanie barwy

📄 Model RGB

📄 Przykład 1

📄 Przykład 2

📄 Model CMY

📄 Przykład 3

📄 Przykład 4

📄 Model CMY(K)

📄 **Przykład 5**

📄 Przykład 6

🖨️ Zadanie 1

🖨️ Zadanie 2

📄 Przykład 7

🖨️ Zadanie 3

Wykład 5 (30.03.2020)

Wykład 6 (6.04.2020)

Wykład 7 (20.04.2020)

GU - laboratorium

WWW - laboratorium



ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



1 Patryk Obuchowicz -



online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 5 (30.03.2020) > Modele zależne i niezależne od urządzenia

Modele zależne i niezależne od urządzenia

Barwa zdefiniowana w modelach RGB, CMY, CMYK, HSL, HSV, HSB zależy od własności konkretnego urządzenia:

- rodzaj tuszu w drukach,
- typ luminoforu w monitorach CRT,
- rodzaj matrycy i podświetlenia w LCD.

Modele te nazywane są **modelami barw zależnymi od urządzenia**.

RGB i CMYK służą do prezentowania kolorów przez fizyczne urządzenia, modele HSB, HSL lepiej oddają sposób, w jaki kolory są interpretowane przez ludzki mózg.

Jednak aby w pełni określić barwę, nie wystarczy podać jej współrzędne w danej przestrzeni, ale potrzebne są również parametry urządzenia, na którym barwa będzie reprodukowana.

Modele CIE są **modelami niezależnymi od urządzenia**, ale dokładnymi percepcyjnie.

Wszystkie modele są wyłącznie umowną, poglądową reprezentacją prawdziwych barw - nie istnieje medium, które dysponowałoby możliwością reprodukowania wszystkich barw widzianych przez człowieka.

◀ Zadanie 5

Przejdz do...



Separacja RGB ▶



Nawigacja

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



Patryk Obuchowicz -



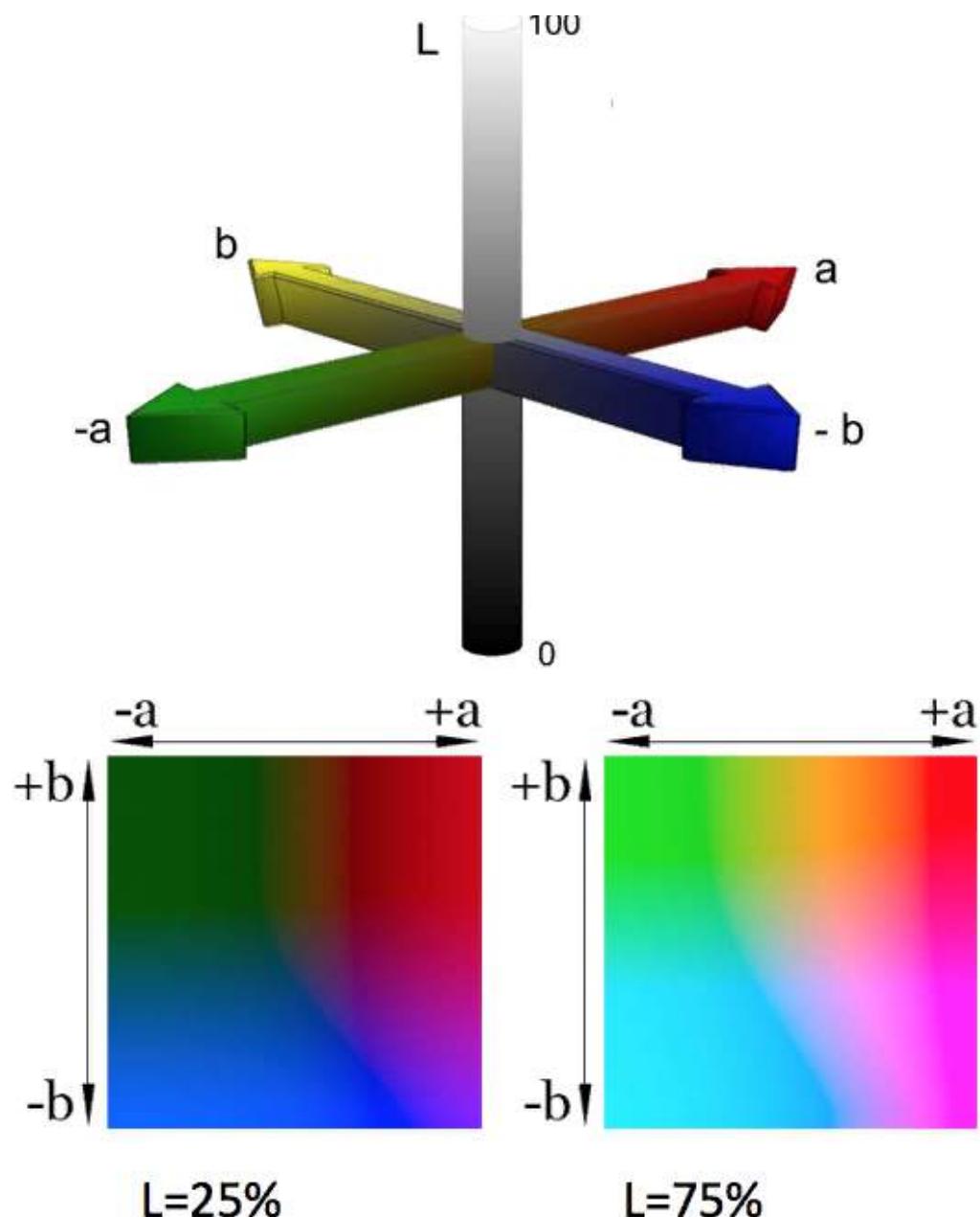
online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 5 (30.03.2020) > Model CIE Lab

Model CIE Lab

Cechy podstawowe:

- model CIE Lab (właściwie CIELab D50) został opracowany w 1976 r.,
- posiada najszerzy gamut spośród wszystkich przestrzeni jakie istnieją,
- oparty jest na postrzeganiu koloru przez ludzkie oko,
- jest równomierny i niezależny od urządzenia (zależny od oka ludzkiego),
- nie posiada tradycyjnych kanałów z pojedynczymi kolorami, ale trzy kanały:
 - Lightness (luminancja) - zawiera informację jedynie o luminancji obrazu i przyjmuje wartości od zera (czerń) do 100 (biel).
 - kanał "a" (tinta) to oś zieleń — czerwień (karmazyn), przyjmuje wartość od -120 do 120
 - kanał "b" (temperatura) to oś żółcień — ciemny niebieski (zbliżony do fioletu), przyjmuje wartość od -120 do 120.



Zalety:

- przestrzeń liniowa, co pozwala na prosty sposób opisu mieszania barw,
- zgodność ze sposobem opisu barw przez człowieka,
- możliwość precyzyjnego opisania barwy wyobrażanej,
- możliwość wiarygodnej oceny różnic w barwach.

Zastosowania:

Wykorzystywany przede wszystkim do opisu percepcji, do porównywania barw, użyteczny, jeśli chcemy korygować jedynie tonację zdjęcia.

Wzory do konwersji:

- z CIE XYZ do CIE Lab

$$L = \begin{cases} 116f\left(\frac{Y}{Y_0}\right) - 16 & \frac{Y}{Y_0} > 0.008856 \\ 903.3\left(\frac{Y}{Y_0}\right) & \frac{Y}{Y_0} \leq 0.008856 \end{cases}$$

$$a = 500 \left[f\left(\frac{X}{X_0}\right) - f\left(\frac{Y}{Y_0}\right) \right]$$

$$a = 200 \left[f\left(\frac{Y}{Y_0}\right) - f\left(\frac{Z}{Z_0}\right) \right]$$

$$f(t) = \begin{cases} t^{1/3} & t > 0.008856 \\ 7.787t + 16/116 & t \leq 0.008856 \end{cases}$$

$$X_0 = 94.81$$

$$Y_0 = 100.0$$

$$Z_0 = 107.3$$

- z CIE Lab do CIE XYZ

$$X = X_0 f^{-1} \left(\frac{L+16}{116} + \frac{a}{500} \right)$$

$$Y = Y_0 f^{-1} \left(\frac{L+16}{116} \right)$$

$$Z = Z_0 f^{-1} \left(\frac{L+16}{116} - \frac{b}{200} \right)$$

$$f^{-1}(t) = \begin{cases} t^3 & t > 6/29 \\ 7.787(t - 4/29) & t \leq 6/29 \end{cases}$$

współczynniki X_0, Y_0, Z_0 to składowe trójchromatyczne przyjętego punktu bieli.

◀ Zadanie 4 ▶ Przejdz do...

Przykład 4 ▶

 Nawigacja

Kokpit

 Strona główna

Strony

Moje kursy

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



Patryk Obuchowicz -



online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 5 (30.03.2020) > Model CIE XYZ

Model CIE XYZ

Cechy podstawowe:

- model tróichromatyczny, stworzony w 1931 r. przez Międzynarodową Komisję Oświetleniową (*Comission Internationale de l'Eclairage*) do opisania percepacji barw przez oko ludzkie,
- podstawą konstrukcji było badanie dwudziestu osób prawidłowo rozróżniających barwy. Osobom tym prezentowano w bardzo wąskim polu widzenia (2°) różne barwy i proszono o ich różnicowanie; na podstawie statystycznej analizy odpowiedzi opracowano model standardowego obserwatora opisujący uśrednione możliwości percepji barw człowieka;
- podstawą modelu jest zestaw trzech składowych tróichromatycznych, których mieszanie pozwala pokryć pełny zakres barw widmowych a wartość każdej ze składowych jest nieujemna,
- barwę w sposób jednoznaczny opisuje się w tym modelu we współrzędnych tróichromatycznych X, Y, Z, odpowiadających procentowemu udziałowi trzech podstawowych barw zależnych od względnych sprawności wizualnych czopków:
 - R (czerwonej); długość fali 700 nm
 - G (zielonej); długość fali 546.1 nm
 - B (niebieskiej); długość fali 435.8 nm
- przyjmuje się ją jako standard i punkt odniesienia do wszystkich innych przestrzeni CIE.

Zalety:

- przestrzeń liniowa, co pozwala na prosty sposób opisu mieszania barw,
- zgodność ze sposobem opisu barw przez człowieka,
- możliwość opisania barwy wyobrażanej,

Wady:

- przestrzeń barw jest odczuwalnie „nieliniowa” z punktu widzenia możliwości rozróżniania barw przez człowieka (obszary zielone są znaczco „rozciągnięte”, obszary brązów i żółci są mocno „ściśnięte”),
- model ten nie jest percepcyjnie jednorodny tzn. nie odzwierciedla sposobu postrzegania barw przez człowieka,
- nie wszystkie barwy CIEXYZ mogą być odtworzone w RGB.



ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



1

Patryk Obuchowicz -



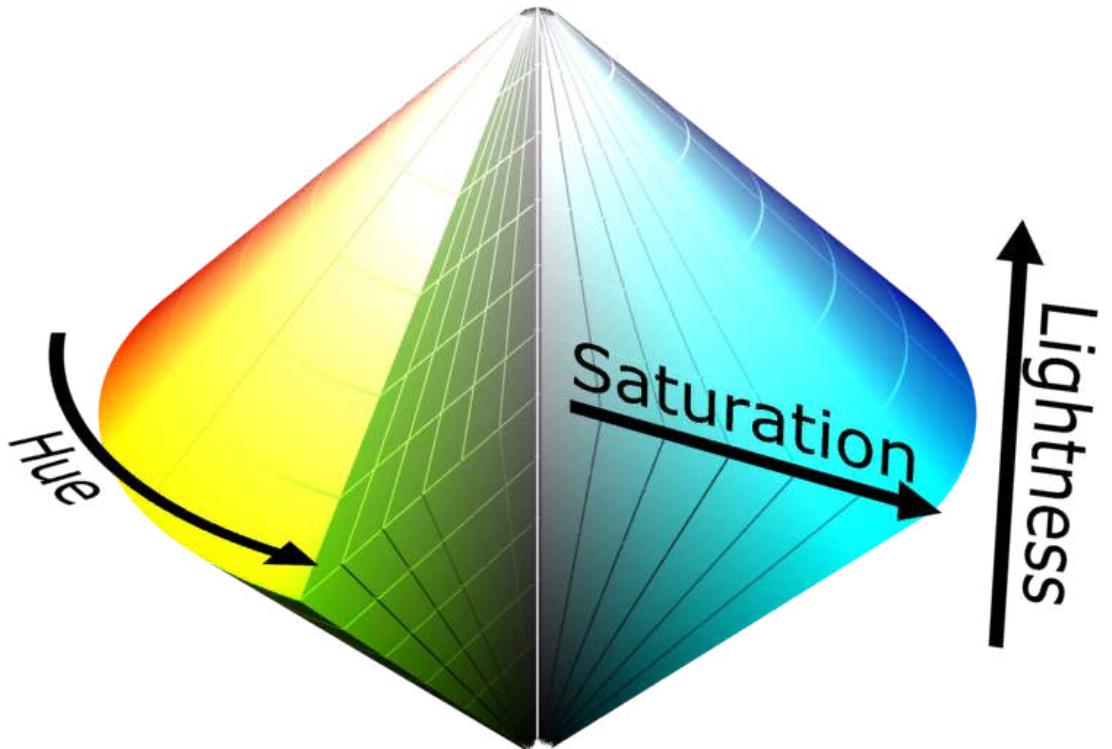
online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 5 (30.03.2020) > Model HSL

Model HSL

Cechy podstawowe:

- podobnie jak w przypadku modeli HSV/HSB w odróżnieniu od modeli RGB i CMYK kolor nie jest w tej przestrzeni definiowany jako mieszanina barw podstawowych czy dopełniających,
- stworzony w latach 1926-1930 jako telewizyjny, później zastąpiony przez inne dedykowane modele,
- model nawiązuje do naturalnego sposobu interpretacji i opisu barw za pomocą trzech atrybutów:
 - Hue: kolor rozumiany klasycznie (np. czerwony, zielony, niebieski) - wyrażony poprzez wartość kąta zdefiniowanego na kole barw Maxwella; przyjmuje wartości z zakresu od 0° do 360°; każdy stopień odpowiada określonej barwie, pozwala zmieniać odcień np. zieleni z niebieskie do żółtawej,
 - Saturation (Chroma): nasycenie, intensywność, zróżnicowanie koloru zależne od jasności, odcienie szarości pomiędzy bielą a czernią opisane przy pomocy wartości od 0 do 100%, zmiana pozwala uzyskać barwy czystsze,
 - Lightness (Luminancja): jasność od ciemna do jasna 0 do 100% (od czerni do biegi), zmiana pozwala uzyskać barwy jaśniejsze (najważniejszy parametr tego modelu).
- jest to model nielinowy, reprezentacją jest złożony podstawami stożek, który jest symetryczny a osią stożka jest oś jasności (Lightness),
- określany jako model dla artystów,
- model jest w pełni konwertowalny na RGB/CMY.



Zalety:

- intuicyjność - zgodność ze sposobem opisu barw przez człowieka,
- możliwość prostego opisania barwy wyobrażanej,
- separacja wielkości opisujących wrażenia chromatyczne (HS) od achromatycznych (V),
- możliwość opisu barwy poprzez tylko dwie składowe: H oraz S, co pozwala wykorzystywać model HSV w zastosowaniach do rozpoznawania obiektów na podstawie koloru (barwy) niezależnie od zmian jasności,
- pozwala zdefiniować efektywne miary w przestrzeni obrazu:
 - liczba unikalnych kolorów mierzonych jako liczba odcieni,
 - nasycenie pikseli (pixel saturation) jako stosunek liczby pikseli o maksymalnym nasyceniu do liczby pikseli nienasyconych oraz często stosowana miara odległości w histogramach HSV.

Wady:

- osobliwości podczas konwersji z RGB do HSV (dotyczy barw achromatycznych i czerni),
- percepcyjna niejednorodność pomimo zorientowania modelu na intuicyjność.

Zastosowania:

Wykorzystywany w aplikacjach specjalistycznych do obróbki zdjęć, analizy obrazu:

- rozjaśnianie jasnych obrazów,
- identyfikacja obiektów kodowanych barwą,
- sortowanie i klasyfikacja.

Model odpowiedni do stosowania w interfejsach użytkownika.

Realizacja modelu:

Model HSV można przedstawić za pomocą stożka.

Składowe: H (wyrażona w stopniach $<0^{\circ}, 360^{\circ}>$) oraz S i V z przedziału $<0,1>$ są niezależne od siebie.

Wzory do konwersji:

- z RGB do HSL

$$M = \max(R, G, B)$$

$$m = \min(R, G, B)$$

$$c = M - m$$

$$L = \frac{M+m}{2}$$

$$H = \begin{cases} \left(\frac{G-B}{c} \bmod 6\right) \cdot 60^\circ & R=M, c \neq 0 \\ \left(\frac{B-R}{c} + 2\right) \cdot 60^\circ & G=M, c \neq 0 \\ \left(\frac{R-G}{c} + 4\right) \cdot 60^\circ & B=M, c \neq 0 \\ 0 & c=0 \end{cases}$$

$$S = \begin{cases} 0 & c=0 \\ \frac{c}{1-|2L-1|} & c \neq 0 \end{cases}$$

- z HSL do RGB

$$c = (1 - |2L - 1|) \cdot S$$

$$m = L - c/2$$

$$x = c \left(1 - \left| \frac{H}{60^\circ} \bmod 2 - 1 \right| \right)$$

$$(R, G, B) = \begin{cases} (c+m, x+m, m) & H = 60^\circ \\ (x+m, c+m, m) & H = 120^\circ \\ (m, c+m, x+m) & H = 180^\circ \\ (m, x+m, c+m) & H = 240^\circ \\ (x+m, m, c+m) & H = 300^\circ \\ (c+m, m, x+m) & H = 360^\circ \\ (m, m, m) & \text{inne} \end{cases}$$

Składowe RGB w ww. wzorach przyjmują wartości w zakresie od 0 do 1.



ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



1

Patryk Obuchowicz -



online M&J

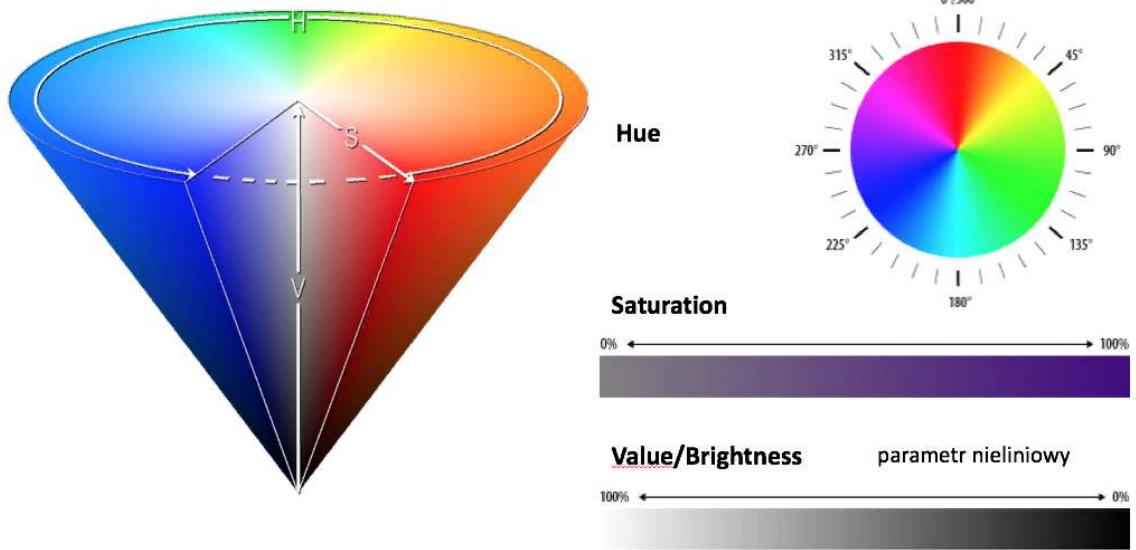
Kokpit > Moje kursy > GU - wykład > Wykład 5 (30.03.2020) > Model HSV

Model HSV

Cechy podstawowe:

- zaproponowany w 1978 roku przez A.R. Smitha,
- w odróżnieniu od modeli RGB i CMYK kolor nie jest definiowany jako mieszanina barw podstawowych czy dopełniających,
- nawiązuje do naturalnego sposobu interpretacji i opisu barw za pomocą trzech atrybutów:
 - Hue: kolor rozumiany klasycznie (np. czerwony, zielony, niebieski) - wyrażony poprzez wartość kąta zdefiniowanego na kole barw Maxwella; przyjmuje wartości z zakresu od 0° do 360°
 - Saturation (Chroma): nasycenie, intensywność, odcień szarości pomiędzy bielą a czernią opisane przy pomocy wartości od 0 (brak) do 1 (pełna intensywność) (lub w procentach 0 do 100%),
 - Value (Intensity, brightness): jasność od ciemna do jasna 0 - 1 (lub 0 do 100%).
- czasami nazywany jest modelem HSB (hue-saturation-brightness),
- jest to model nielinowy, reprezentacją jest stożek, którego podstawę stanowi koło barw,
- model jest w pełni konwertowalny na RGB/CMY.





Zalety:

- intuicyjność - zgodność ze sposobem opisu barw przez człowieka,
- separacja wielkości opisujących wrażenia chromatyczne (HS) od achromatycznych (V),
- możliwość opisu barwy poprzez tylko dwie składowe: H oraz S, co pozwala wykorzystywać model HSV w zastosowaniach do rozpoznawania obiektów na podstawie koloru (barwy) niezależnie od zmian jasności,
- pozwala zdefiniować efektywne miary w przestrzeni obrazu:
 - liczba unikalnych kolorów mierzonych jako liczba odcieni,
 - nasycenie pikseli (pixel saturation) jako stosunek liczby pikseli o maksymalnym nasyceniu do liczby pikseli nienasyconych oraz często stosowana miara odległości w histogramach HSV.

Wady:

- osobliwości podczas konwersji z RGB do HSV,
- percepcyjna niejednorodność pomimo zorientowania modelu na intuicyjność.

Zastosowania:

- rozjaśnianie jasnych obrazów
- identyfikacja obiektów kodowanych barwą,
- sortowanie i klasyfikacja,
- odpowiedni do stosowania w interfejsach użytkownika.

Realizacja modelu:

Model HSV można przedstawić za pomocą stożka.

Składowe: H (wyrażona w stopniach $<0^\circ, 360^\circ>$) oraz S i V z przedziału $<0,1>$ są niezależne od siebie.

Wzory do konwersji:

- z RGB do HSV

$$M = \max(R, G, B)$$

$$m = \min(R, G, B)$$

$$c = M - m$$



$$V = M$$

$$S = \frac{c}{V}$$

$$H = \begin{cases} \left(\frac{G-B}{c} \text{ mod } 6 \right) \cdot 60^\circ & R = M, c \neq 0 \\ \left(\frac{B-R}{c} + 2 \right) \cdot 60^\circ & G = M, c \neq 0 \\ \left(\frac{R-G}{c} + 4 \right) \cdot 60^\circ & B = M, c \neq 0 \\ 0 & c = 0 \end{cases}$$

- z HSV do RGB

$$c = V \cdot s$$

$$m = V - c$$

$$x = c \left(1 - \left| \frac{H}{60} \text{ mod } 2 - 1 \right| \right)$$

$$(R, G, B) = \begin{cases} (c+m, x+m, m) & H \in [0^\circ, 60^\circ] \\ (x+m, c+m, m) & H \in (60^\circ, 120^\circ) \\ (m, c+m, x+m) & H \in (120^\circ, 180^\circ) \\ (m, x+m, c+m) & H \in (180^\circ, 240^\circ) \\ (x+m, m, c+m) & H \in (240^\circ, 300^\circ) \\ (c+m, m, x+m) & H \in (300^\circ, 360^\circ) \\ (m, m, m) & \text{inne} \end{cases}$$

Składowe RGB w ww. wzorach przyjmują wartości w zakresie od 0 do 1.

◀ Zadanie 3

Przejjdź do...

Przykład 1 ▶



Nawigacja

Kokpit

Strona główna

Strony

Moje kursy

MAD - wykład



ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie

1

Patryk Obuchowicz -

online M&J



Kokpit > Moje kursy > GU - wykład > Wykład 5 (30.03.2020) > Przykład 1

Przykład 1

Kolor tła na stronie internetowej ma składowe w zapisie dziesiętnym:

- czerwony = 27
- zielony = 90
- niebieski = 104

Jaki będzie kod koloru w przestrzeni HSV?



Rozwiązanie:

Składowe wynoszą:

RGB (27,90,104)

R=27 G=90 B=104

Dokonujemy konwersji:

$$R'=R/255=27/255$$

$$G'=G/255=90/255$$

$$B'=B/255=104/255$$

$$M=\max(R',G',B')=0.407=B'$$

$$m=\min(R',G',B')=0.105$$

$$c=M-m=77/255=0.301>0$$

Ponieważ $M=B'$ oraz $c>0$ wybieramy trzeci wariant wzoru



$$H = \left(\frac{27-90}{77} + 4 \right) \cdot 60^\circ = 191^\circ$$

$$V = M = \frac{104}{255} \cdot 100\% = 41\%$$

$$S = \frac{c}{V} = \frac{77}{104} \cdot 100\% = 74\%$$

◀ Model HSV Przejdz do... ▶

Model HSL ▶

Nawigacja

- Kokpit
- 🏠 Strona główna
- Strony
- Moje kursy
 - MAD - wykład
 - MAD - ćwiczenia
 - GU - wykład
 - Uczestnicy
- 📊 Oceny
- GRAFIKA UŻYTKOWA - WYKŁAD
- Informacje
- Wykład 1
- Wykład 2
- Wykład 3
- Wykład 4 (23.03.2020)
- Wykład 5 (30.03.2020)
 - ?
 - Lista obecności 30.03.2020 - kliknij i zaznacz
- 📄 Model HSV
- 📄 **Przykład 1**
- 📄 Model HSL
- 📄 Przykład 2
- 📄 Model CIE XYZ
- 📄 Przykład 3
- 📝 Zadanie 4
- 📄 Model CIE Lab
- 📄 Przykład 4
- 📝 Zadanie 5
- 📄 Modele zależne i niezależne od urządzenia
- Wykład 6 (6.04.2020)

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



1

Patryk Obuchowicz -

online M&J



Kokpit > Moje kursy > GU - wykład > Wykład 5 (30.03.2020) > Przykład 2

Przykład 2

Kolor tła na stronie internetowej ma składowe w zapisie dziesiętnym:

- czerwony = 27
- zielony = 90
- niebieski = 104

Jaki będzie kod koloru w przestrzeni HSL?



Rozwiążanie:

Składowe wynoszą:

RGB (27,90,104)

R=27 G=90 B=104

Dokonujemy konwersji:

$$R'=R/255=27/255$$

$$G'=G/255=90/255$$

$$B'=B/255=104/255$$

$$M=\max(R',G',B')=0.407=B'$$

$$m=\min(R',G',B')=0.105$$

$$c=M-m=77/255=0.301>0$$

$$L=(M+m)/2=0.512/2=0.256$$

Ponieważ $M=B'$ oraz $c>0$ wybieramy trzeci wariant wzoru



$$H = \left(\frac{27-90}{77} + 4 \right) \cdot 60^\circ = 191^\circ$$

$$L = 0.256 \cdot 100\% = 25.6\%$$

$$S = \frac{c}{1 - |2L - 1|} = 0.587 \cdot 100\% = 58.7\%$$

◀ Model HSL ▶ Przejdz do... ▶ Model CIE XYZ ▶

Nawigacja

- Kokpit
- Strona główna
- Strony
- Moje kursy
- MAD - wykkład
- MAD - ćwiczenia
- GU - wykład
- Uczestnicy
- Oceny
- GRAFIKA UŻYTKOWA - WYKŁAD
- Informacje
- Wykład 1
- Wykład 2
- Wykład 3
- Wykład 4 (23.03.2020)
- Wykład 5 (30.03.2020)
- Lista obecności 30.03.2020 - kliknij i zaznacz
- Model HSV
- Przykład 1
- Model HSL
- Przykład 2**
- Model CIE XYZ
- Przykład 3
- Zadanie 4
- Model CIE Lab
- Przykład 4
- Zadanie 5
- Modele zależne i niezależne od urządzenia
- Wykład 6 (6.04.2020)
- Wykład 7 (20.04.2020)
- GU - laboratorium

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



Patryk Obuchowicz -



online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 5 (30.03.2020) > Przykład 3

Przykład 3

Kolor tła na stronie internetowej ma składowe w zapisie dziesiętnym:

- czerwony = 27
- zielony = 90
- niebieski = 104

Jaki będzie kod koloru w przestrzeni CIE XYZ?



Rozwiążanie:

Składowe wynoszą:

RGB (27,90,104)

R=27 G=90 B=104

R'=0.10588 G'=0.35294 B'=0.40784

Formalnie poprawna konwersja:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.71516 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \begin{bmatrix} 0.10588 \\ 0.35294 \\ 0.40784 \end{bmatrix} = \begin{bmatrix} 0.24346 \\ 0.36673 \\ 0.48114 \end{bmatrix}$$

daje rezultat odbiegający od oczekiwania pod względem percepcyjnym.

W praktyce wykonujemy konwersję gamma R'G'B' wszystkich współczynników

$$R'' = \left[\frac{R' + 0,055}{1,055} \right]^{2.4} = 0.01095971$$

$$G'' = \left[\frac{G' + 0,055}{1,055} \right]^{2.4} = 0.10224103$$

$$B'' = \left[\frac{B' + 0,055}{1,055} \right]^{2.4} = 0.13842936$$

a dopiero potem wykorzystujemy macierz współczynników opisującą konwersję z przestrzeni sRGB do XYZ:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.71516 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \begin{bmatrix} 0.01095971 \\ 0.10224103 \\ 0.13842936 \end{bmatrix} = \begin{bmatrix} 0.06606 \\ 0.08544 \\ 0.14395 \end{bmatrix}$$

Po zamianie na procent: X=6.606 Y=8.544 Z=14.395.

The screenshot shows a web-based application interface. On the left, there is a vertical sidebar with a red decorative background. The sidebar contains the following navigation items:

- ◀ Model CIE XYZ
- Zadanie 4 ►
- Nawigacja**
 - Kokpit
 - Strona główna
 - Strony
 - Moje kursy
 - MAD - wykład
 - MAD - ćwiczenia
 - GU - wykład
 - Uczestnicy
 - Oceny
 - GRAFIKA UŻYTKOWA - WYKŁAD
 - Informacje
 - Wykład 1
 - Wykład 2
 - Wykład 3
 - Wykład 4 (23.03.2020)
 - Wykład 5 (30.03.2020)
 - Lista obecności 30.03.2020 - kliknij i zaznacz
 - Model HSV
 - Przykład 1
 - Model HSL
 - Przykład 2

In the center, there is a white content area with a small button labeled "Przejdz do..." and a dropdown arrow icon.

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



1

Patryk Obuchowicz -



online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 5 (30.03.2020) > Przykład 4

Przykład 4

Kolor tła na stronie internetowej ma składowe w zapisie dziesiętnym:

- czerwony = 27
- zielony = 90
- niebieski = 104

Jaki będzie kod koloru w przestrzeni CIELab?



Rozwiązanie:

Składowe wynoszą:

RGB (27,90,104)

po konwersji do CIE XYZ (Przykład 3) otrzymujemy:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.71516 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \begin{bmatrix} 0.01095971 \\ 0.10224103 \\ 0.13842936 \end{bmatrix} = \begin{bmatrix} 0.06606 \\ 0.08544 \\ 0.14395 \end{bmatrix}$$

Następnie dokonujemy konwersji do CIE Lab zgodnie ze wzorami transformacyjnymi.

Przed wykonaniem operacji należy wartości X, Y, oraz Z zamienić na procent

X=6.606 Y=8.544 Z=14.395

$$L = \begin{cases} 116f\left(\frac{Y}{Y_0}\right) - 16 & \frac{Y}{Y_0} > 0.008856 \\ 903.3\left(\frac{Y}{Y_0}\right) & \frac{Y}{Y_0} \leq 0.008856 \end{cases}$$

$$a = 500 \left[f\left(\frac{X}{X_0}\right) - f\left(\frac{Y}{Y_0}\right) \right]$$

$$a = 200 \left[f\left(\frac{Y}{Y_0}\right) - f\left(\frac{Z}{Z_0}\right) \right]$$

$$f(t) = \begin{cases} t^{1/3} & t > 0.008856 \\ 7.787t + 16/116 & t \leq 0.008856 \end{cases}$$

$$X_0 = 94.81$$

$$Y_0 = 100.0$$

$$Z_0 = 107.3$$

$$\begin{bmatrix} L \\ a \\ b \end{bmatrix} = \begin{bmatrix} 35.0927 \\ -14.646 \\ -13.796 \end{bmatrix}$$

◀ Model CIE Lab Przejdz do... ▶

Zadanie 5 ►

 **Nawigacja**

- Kokpit
-  Strona główna
- Strony
- Moje kursy
 - MAD - wykład
 - MAD - ćwiczenia
 - GU - wykład
 - Uczestnicy
-  Oceny
- GRAFIKA UŻYTKOWA - WYKŁAD
- Informacje
- Wykład 1
- Wykład 2
- Wykład 3

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie

1

Patryk Obuchowicz -

online M&J



Kokpit > Moje kursy > GU - wykład > Wykład 6 (6.04.2020) > Separacja RGB

Separacja RGB

Separacją barw nazywamy proces dekompozycji obrazu barwnego na kanały (tzw. wyciągi barwne) z których każdy reprezentuje intensywność innej składowej barwy.

Separację barw wykonuje się w celu:

- przygotowania materiałów do druku lub naświetlania np. druk offsetowy CMYK (pozwala na uzyskanie różnobarwnej odbitki przy wykorzystaniu techniki separacji kolorów oraz odpowiednio przygotowanych matryc - płaskie blachy aluminiowe, klisze lub blachy nawinięte na bęben; w procesie druku cztery kolory farb C, M, Y, K zostają osobno nałożone na podłoże);
- wyodrębnienie informacji zawartej tylko w wybranym kanale np. kolor podstawowy (RGB), dopełniający (CMYK), jasność (HSV, HSL);
- wyeliminowania szumów (w RGB najczęściej szumów zazwyczaj zawiera kanał niebieski).

Kolor obrazu w danym kanale jest umowny - informuje jedynie o tym jaki to jest kanał, dla interpretacji danych istotne są jedynie różnice w poziomie sygnału w danym kanale.

Separacja RGB

Kanały identyfikowane kolorem





Oryginał
Blue

Red

Green

Kanały nieidentyfikowane kolorem



Oryginał
Blue

Red

Green

◀ Modele zależne i niezależne od urządzenia

Przejdz do...

→ Wyszukiwanie obrazów

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



1

Patryk Obuchowicz -



online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 6 (6.04.2020) > Zasady doboru barw

Zasady doboru barw

Zasady doboru kolorów

W trakcie doboru kolorów przydatne jest koło barw. Kolory dobierać można stosując reguły harmonii:

- **analogiczne (pokrewne)** - harmonijne, przyjazne dla oka, typowo jeden kolor to dominanta, drugi wspierający, trzeci akcenty;
- **monochromatyczne (jeden kolor)** - czyste, eleganckie, harmonijne, spójne, monotonne, wykorzystuje różnice w jasności i nasyceniu;
- **triada (trzy kolory)** - mocny kontrast, wykorzystuje kolory równomiernie rozłożone wokół trzech równo oddalonych punktów na kole kolorów - jeden dominujący dwa pozostałe to akcenty;
- **komplementarne, dopełniające (dwa kolory oddalone o 180°)** - efekt łączenia kolorów zimnych i ciepłych, bazuje na kolorach, które znajdują się po przeciwnych stronach koła kolorów, zapewnia wysoki kontrast ale jest złym wyborem dla tekstów,
- **złożone** - połączenie kolorów komplementarnych i analogicznych, dwa kolory o tej samej barwie, które są analogiczne z kolorem bazowym, kolor bazowy oraz dwa kolory komplementarne do koloru bazowego, mocny akcent wizualny ale łagodniejszy niż komplementarnych,
- **odcienie** - kolory o tej samej barwie i nasyceniu, lecz o różnych wartościach jasności.

Optymalne wybory:

- **monochromatyczne** - wybór z jednej palety, o zawartości tego samego koloru podstawowego (np. odcienie czerwieni);
- **harmonijne** - zestaw kolorów położonych w tej samej ćwiartce koła;
- **kontrastowe** - wybór dwóch przeciwnieległych kolorów, np. brąz z czerwienią.

Na co zwrócić uwagę:

- nie wszystkie kolory na pozór podobne należą do tej samej palety;
- nie wszystkie na pozór różne kolory są kontrastowe;
- wzbudzanie agresji - źle dobrane kolory kontrastowe;
- zmęczenie wzroku - źle dobrane kolory kontrastowe;
- znaczenie kulturowe - kolory mają zupełnie różne znaczenie w różnych kulturach;

- znaczenie psychologiczne - wpływ na umysł (koło Goethego z zajęć #3).

Inne istotne czynniki:

- światło - należy stosować barwy ciemne i nasycone w warunkach intensywnego oświetlenia, natomiast barwy jasne i mniej nasycone w warunkach słabego oświetlenia, kolory jasne powiększają optycznie przedmioty, ciemne pomniejszają;
- faktura powierzchni - najlepiej wybierać barwy o stopień jaśniejsze dla powierzchni porowatych i o ton ciemniejsze dla powierzchni gładkich;
- połysk - w przypadku matu kolor będzie wydawał się cieplejszy, natomiast przy wysokim stopniu połysku - chłodniejszy.

Zalecenia Gestalt

Dotyczące barwy

- barw należy używać z powściągliwością tylko gdy to jest konieczne,
- minimalizować liczbę użytych barw do kliku – na tyle koncentruje się człowiek,
- szczególnie uważać przy reprezentacji cech ilościowych za pomocą barwy,
- stosować barwy o tej samej jasności,
- stosować barwy percepcyjnie równoodległe,
- nie stosować odciennego tego samego koloru dla obiektów bez związku,
- nie stosować chromatyczności dla elementów interfejsów, opisów legend,
- jako tło rysunku o wybranej barwie stosować barwę dopełniającą,
- w przypadku obrazów barwnych stosować tło neutralne
- uwzględniać psychologiczne znaczenie barwy.

Dotyczące szczegółów i konturów

- szczegóły wyróżniać z tła korzystając z jasności a nie barwy,
- nie stosować barw dla obiektów małych (słaba widoczność),
- obszary barw sąsiadujących i źle harmonizujących oddzielać linią,
- stosować obwódki dla ułatwienia percepji konturu.

UWAGA: Brak braku regulacji prawnych = jednoznacznych reguł stosowania i doboru barwy!



ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



1

Patryk Obuchowicz -



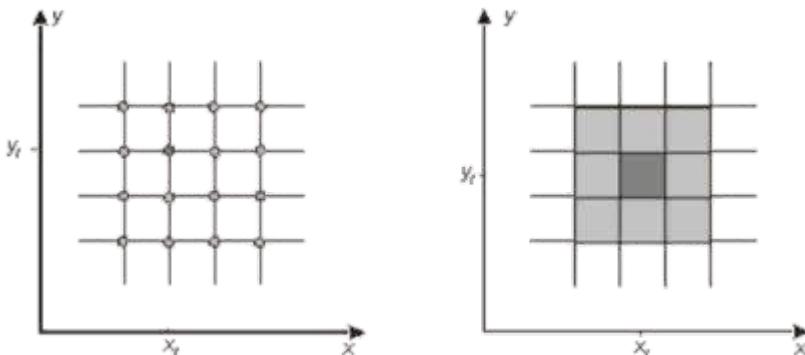
online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 6 (6.04.2020) > Bitmapa i jej charakterystyka

Bitmapa i jej charakterystyka

Bitmapa, mapa pikseli/bitowa

Mapa pikseli/bitowa (bitmapa) to matryca pikseli w postaci prostokątnej siatki (rastra).

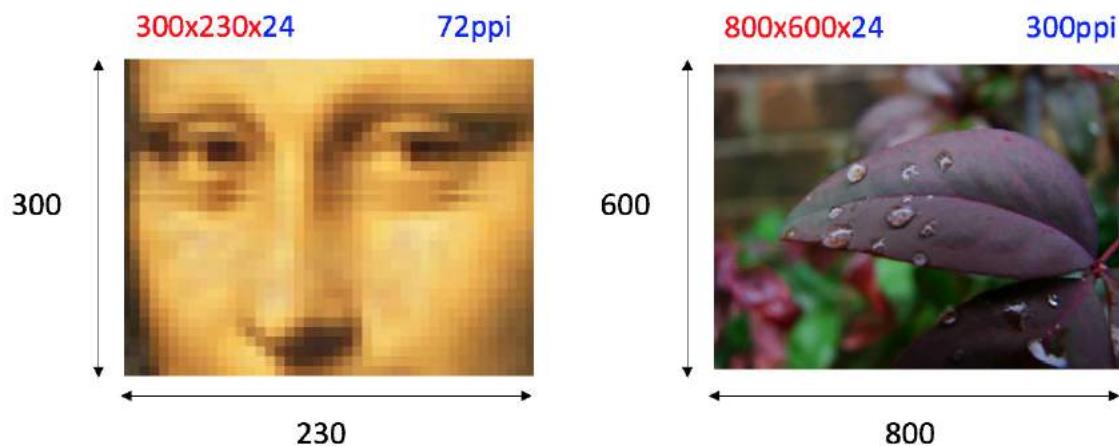


W bitmapie:

- piksele są przedstawiane jako węzły siatki,
- piksele są traktowane jako obszary o określonej powierzchni i są reprezentowane przez oczka siatki prostokątnej,
- kolor każdego piksela jest definiowany osobno,
- piksel jako najmniejsza komponenta może być definiowany zależnie od kontekstu (druk, LCD, fotosensory).

Bitmapę charakteryzuja

- **rozdzielcość pikselowa** – wysokość i szerokość bitmapy w pikselach (np. 300x230 dla obrazu po lewej i 800x600 dla obrazu po prawej)
- **rozdzielcość liniowa** – wyliczana na podstawie rozdzielczości pikselowej (np. 72 ppi dla obrazu po lewej i 300 ppi dla obrazu po prawej)
- **głębina kolorów** – liczba bitów na piksel (np. 24 dla obu obrazów)



Rozdzielcość pikselowa

Obraz jest reprezentowany przez matrycę **m x n** pikseli, gdzie **m - pozioma rozdzielcość pikselowa obrazu, n - pionowa rozdzielcość pikselowa obrazu.**

Rozdzielcość pikselowa precyzuje za pomocą ilu pikseli obraz jest reprezentowany, ale nie jest cechą stałą obrazu gdyż przy powiększaniu mapy bitowej występuje efekt powiększania piksela i utrata jakości.

Rozdzielcość liniowa

Rozdzielcość liniowa urządzenia reprodukującego, określona jest przez liczbę podstawowych elementów (pikseli, kropek, linii) przypadających na jednostkę długości:

- **PPI (points per inch)** - w odniesieniu do monitorów/wyświetlaczy,
- **DPI (dots per inch)** - w odniesieniu do drukarek/skanerów,
- **LPI (lines per inch)** - w odniesieniu do papieru/druku.

Im większe PPI/DPI/LPI tym ostrzejszy obraz i tym większy rozmiar pliku jest wymagany.

Związek pomiędzy rozdzielcością pikselową a liniową ilustruje poniższy przykład.

Przykład

Dla ekranu o rozdzielcości pikselowej 800 x 600 i przekątnej 15" określić rozdzielcość liniową w poziomie wyrażoną w ppi.

Przyjąć, że stosunek boków ekranu wynosi $a : b = 4 : 3$.

Rozwiążanie:

Z równania Pitagorasa $15^2 = a^2 + b^2$

$$a = 4x$$

$$b = 3x$$

$$15^2 = 16x^2 + 9x^2$$

$$x = 3"$$

$$a = 12" \text{ i } b = 9"$$

Przykład:

Jakie będą rozmiary obrazu o rozdzielczości pikselowej 800 x 600 reprodukowanego za pomocą drukarki o rozdzielczości 300 dpi a jakie za pomocą drukarki 600 dpi?

- **drukarka 300 dpi**

wymiary obrazu będą wynosiły odpowiednio: $800/300 = 2.66"$ = 6.77 cm i $600/300 = 2"$ = 4.08 cm czyli 6.77cm x 4.08 cm.

- **drukarka 600 dpi**

wymiary obrazu będą wynosiły odpowiednio: $800/600 = 1.33"$ = 3.39 cm i $600/600 = 1"$ = 2.54 cm czyli 3.39 cm x 2.54 cm.

Kryteria doboru rozdzielczości:

Grafika prezentacyjna

- rozdzielcość dobrana przy uwzględnieniu wielkości docelowego ekranu na którym będzie prezentowana
(obraz wielkości 3" x 5" o rozdzielczości 150 ppi można dobrze skalować do 6" x 10" o rozdzielczości 75 ppi ale nie bardziej)
- obrazów o rozdzielczości powyżej 600 ppi nie można zobaczyć na żadnym ekranie

Grafika na potrzeby druku

- zapewniająca minimum 300 dpi maximum 1200 dpi,
- dopasowana do sposobu druku (plakat, baner czy książka).

Grafika webowa

- zalecana minimalna wielkość ekranu 1024 x 768 pikseli
- ekran CRT, LCD ma rozdzielcość liniową 67-130 ppi, wyświetlacze w telefonach komórkowych 300-400 ppi,
- na ekranie 72 ppi a 300 ppi nie różnią się ale 300 ppi ma ok. 4 razy większą objętość

Wady grafik o zbyt dużej rozdzielczości dla www

- zabierają dużo miejsca,
- wolno się ładują,
- obciążają pasmo,
- jeśli wymagane jest skalowanie należy przygotować kilka wariantów grafiki.

Głębica bitowa

Głębica bitowa to liczba bitów użytych do reprezentacji koloru danego piksela w danym modelu przestrzeni barw. Im wyższa, tym większa jest rozdzielcość tonalna kolorów. Wyraża się ją w jednostkach bpp czyli liczbie bitów na piksel.

Tryb koloru	Liczba barw	Wartości barwy	Głębica bitowa [bpp]	Liczba bitów pamięci do przechowywania obrazu mxn	Liczba kanałów	Przykład
			[bpp]	obrazu mxn	kanałów	
bitmapowy	$2^1 = 2$	{0,1}	1	$m \cdot n \cdot 1$	1	
skala szarości	$2^8 = 256$	$\{0,1,...255\} = <0,255>$	8	$m \cdot n \cdot 8$	1	
8 bitowy kolor	$8 \times 8 \times 4 = 256$	$\{0,1,...255\} = <0,255>$	8 (3R+3G+2B)	$m \cdot n \cdot 8$	3	
True Color	$2^{24} = 16.777.216$	{R,G,B}, gdzie R,G,B $<0,255>$	24	$m \cdot n \cdot 24$	3	

Tryb koloru

Tryb koloru jest to sposób reprezentacji kolorów w obrazach rastrowych, który opisuje, jak fizycznie reprezentowane są dane o obrazie. Dotyczy zapisu w plikach graficznych, systemach lub urządzeniach wyświetlających. Tryb koloru bezpośrednio opiera się na modelach barw.

Rozróżniamy następujące tryby koloru:

- **czarno-biały** (bitmap mode) – dozwolone są dwie wartości koloru pikseli (czarna i biała), piksel zajmuje jeden bit,
- **bichromia** (duotone mode) – monochromatyczny, bichromatyczny, trichromatyczny lub kwadrochromatyczny - dozwolone są 1,2,3 lub 4 kolory,
- **skala szarości** (grayscale mode) – oprócz czarnego i białego występują piksele o jasności pośrednich, piksel zajmuje 1, 2 lub więcej bajtów – wartość liczbową jest wprost powiązana z jasnością (liniowo lub nieliniowo),
- **kolorowy** – przyporządkowanie pikselom kolorów z przestrzeni barw RGB itp.
 - **indeksowany** (indexed color mode) – obraz składa się z dwóch części: tablicy kolorów CLUT oraz obrazu, w którym każdy piksel jest indeksem do koloru w tablicy (typowy dla formatu gif); maksymalnie 256 kolorów,
 - **bezpośredni** – każdy piksel przechowuje indywidualnie kolor; dla RGB rozróżnia się: High Color (32 lub 64 tys.) – piksel zajmuje 2 bajty oraz True Color (16 mln.) – piksel zajmuje 3 lub 4 bajty; w CMYK każdemu pikselowi są przypisane procentowe wartości każdej z czterech farb podstawowych; w Lab obrazy mogą mieć nawet głębię 48-bitową (16 bitów na kanał).

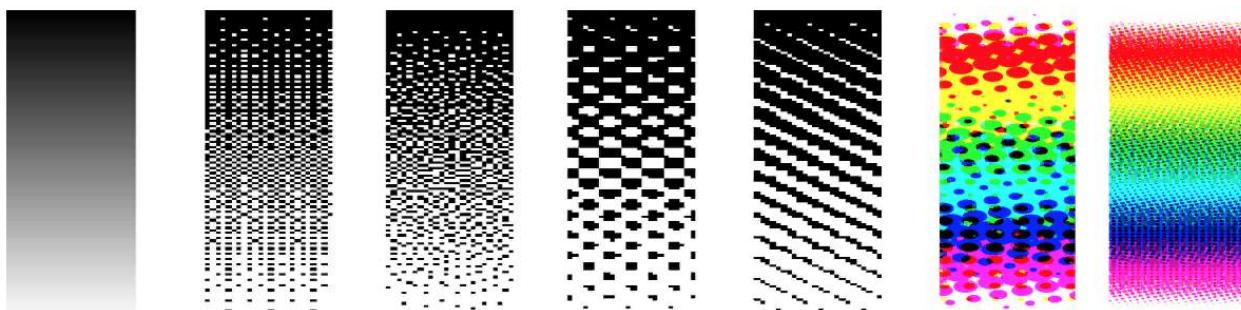
Zmiana trybu pozwala na zmianę palety barw obrazu ale jednocześnie może powodować nieodwracalną utratę informacji o kolorach.

Tonalność a ciąglotonalność

Użycie tego samego koloru w różnych odcieniach, czyli roznaczyc się nasyceniem i jasnością, natomiast ciągłość tonalność to ciągłe, a zarazem płynne przejścia pomiędzy różnymi odcieniami tej samej barwy oraz pomiędzy różnymi kolorami.

Roztrząsanie („Dithering”, „Error diffusion” lub „Dither Option”)

Dithering to symulowanie barw występujących w obrazie, ale niedostępnych dla środowiska operacyjnego. Opiera się on na złudzeniu optycznym wykorzystującym fakt, że półtonalne kwadraty mogą skutecznie symulować barwy spoza tabeli kolorów. Jest to technika zalecana dla obrazów o małej liczbie kolorów i zawierających przejścia tonalne ale niewskazana dla obszarów o jednolitym kolorze.



Rodzaje ditheringu:

- zamiast danego koloru używa się najbardziej zbliżonego (wada: ostre przejścia między poszczególnymi odcieniami koloru),
- nieuporządkowane rozpryskiwanie kolorów w odpowiednich proporcjach dostępnych w palecie
- stosowanie wybranych losowo pikseli ułożonych we wzory w celu symulacji brakujących kolorów (Mac)

Antialiasing

Antialiasing jest techniką pozwalającą zapobiegać nienaturalnemu odwzorowaniu ukośnych krawędzi określonym jako tzw. schodkowanie. Stosuje się go do prawie wszystkich krzywych poza pionowo/poziomo ułożonymi liniami prostymi, których grubość jest równa wielokrotności pojedynczego piksela.



Polega na:

- uzupełnianiu tworzonego obrazu kolorami pośrednimi w celu wygładzenia brzegów rysunków,
- dodawaniu do krawędzi rysunku kolorów pośrednich między barwą obiektu a tłem.

Techniki antialiasingu:

- supersampling (zwiększone próbkowanie a następnie uśrednianie) - mało wydajny,
- multisampling (usunięcie obszarów zbędnych jak cieniowania i tekstury i próbkowanie pozostałej jego części) - szybszy ale mniej efektowny.

Zalety: zapobiega się powstawaniu „ząbkowanych” krawędzi.

Wady: zmniejsza stopień maksymalnej kompresji tworzonych plików graficznych, wymaga większej mocy obliczeniowej.

Przeplot (*ang. interlace*)

Przy pomocy klasycznej metody progresywnej obraz video jest wyświetlany od pierwszej do ostatniej linii skanowania, w kolejności od góry do dołu, ale zawsze w całości.

Wadą tego rozwiązania jest konieczność zapewnienia dużej przepustowości łącza.



Przeplot polega na wyświetlaniu co drugiej linii podczas pierwszego ładowania obrazu i dodaniu brakujących linii dopiero w kolejnym „przejściu”.

Mózg potrafi dopełnić obraz i go zinterpretować zanim brakujące linie w obrazie zostaną wyświetcone.



Zalety:

- stworzony na potrzeby TV analogowej i zarzucony okazał się dogodny dla TV cyfrowej,
- rozwiązuje problem małych przepustowości łączy,
- przeplot jest sposobem na uzyskanie możliwie najbardziej płynnego obrazu video,
- stanowi alternatywę dla wyświetlania dużej liczby klatek na sekundę,
- niektóre formaty graficzne obsługują przeplot co jest dogodne do przyspieszania wyświetlania obrazów na stronach www.

Wady:

- naprzemienne wyświetlanie dwóch niepełnych półpól sprawa, że obraz jest podatny na migotanie i rozmycie.

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



Patryk Obuchowicz -



online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 6 (6.04.2020) > Kryteria doboru rozdzielczości i głębi

Kryteria doboru rozdzielczości i głębi

Kryteria doboru rozdzielczości

Grafika prezentacyjna

- rozdzielcość dobrana przy uwzględnieniu wielkości docelowego ekranu na którym będzie prezentowana
(obraz wielkości 3" x 5" o rozdzielczości 150 ppi można dobrze skalować do 6" x 10" o rozdzielczości 75 ppi ale nie bardziej)
- obrazów o rozdzielczości powyżej 600 ppi nie można zobaczyć na żadnym ekranie

Grafika na potrzeby druku

- zapewniająca minimum 300 dpi maximum 1200 dpi,
- dopasowana do sposobu druku (plakat, baner czy książka).

Grafika webowa

- zalecana minimalna wielkość ekranu 1024 x 768 pikseli
- ekran CRT, LCD ma rozdzielcość liniową 67-130 ppi, wyświetlacze w telefonach komórkowych 300-400 ppi,
- na ekranie 72 ppi a 300 ppi nie różnią się ale 300 ppi ma ok. 4 razy większą objętość

Wady grafik o zbyt dużej rozdzielczości dla www

- zabierają dużo miejsca,
- wolno się ładować,
- obciążają pasmo,
- jeśli wymagane jest skalowanie należy przygotować kilka wariantów grafiki.

Kryteria doboru głębi bitowej

Grafika prezentacyjna i webowa

- zależnie od typu monitora/ekranu co najmniej 16-bitowa optymalnie 32-bitowa (32-bitowa przydatna do obsługi przezroczystości, gradientów, cieni)

Grafika na potrzeby druku

- zależnie od typu maszyny drukarskiej i jej zdolności reprodukcji kolorów

Zalety grafik o dużej głębi:

- obsługa przezroczystości,
- płynne gradienty,
- realistyczne cienie,
- jaskrawsze kolory
- jeśli wymagane jest dostosowanie do urządzeń można to uwzględnić.

Wady grafik o zbyt dużej głębi:

- zła reprodukcja kolorów,
- zabierają dużo miejsca,
- wolno się ładują,
- obciążają procesor/kartę graficzną.

Wady grafik o zbyt małej głębi:

- odczuwalny dyskomfort
- utrata istotnych informacji

◀ Bitmapa i jej charakterystyka

Przejdz do...

Zadanie 7 ►

Nawigacja

Kokpit

🏠 Strona główna

Strony

Moje kursy

MAD - wykład

MAD - ćwiczenia

GU - wykład

Uczestnicy

📊 Oceny

GRAFIKA UŻYTKOWA - WYKŁAD

Informacje

Wykład 1

Wykład 2

Wykład 3

Wykład 4 (23.03.2020)

Wykład 5 (30.03.2020)

Wykład 6 (6.04.2020)

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



1

Patryk Obuchowicz -



online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 6 (6.04.2020) > Formaty plików

Formaty plików

GRAFIKA WWW – FORMATY PLIKÓW

Minimum informacji o mapie bitowej konieczne do wyświetlenia obrazu na ekranie

Do wyświetlenia obrazu na ekranie konieczne są następujące dane, które muszą zostać zapisane w pliku:

- rozmiar rysunku (szerokość i wysokość),
- liczba bitów na piksel (głębina),
- rodzaj grafiki monochromatyczna / True color / kolor indeksowany,
- paleta kolorów, jeśli obraz jest opisany za pomocą palety,
- tablica pikseli.

Zasadnicze różnice pomiędzy formatami plików graficznych

- kolejność umieszczonych w nagłówku informacji może być różna dla różnych formatów plików,
- niektóre formaty, przystosowane do konkretnej karty graficznej, nie muszą zawierać palety jedynie tablice pikseli,
- linie obrazu mogą być ułożone w kolejności od góry do dołu lub od dołu do góry,
- jeżeli wartości pikseli są składowymi RGB, to kolejność zapisu tych składowych może być różna,
- wartości pikseli mogą być zapisane w postaci pakietów lub planów obrazu:
 - w przypadku pakietów bity, odpowiadające kolejnym pikselom, są zapisywane jeden za drugim,
 - w przypadku planów bity poszczególnych pikseli są zapisywane na tej samej pozycji w kolejnych planach,
- tabele pikseli mogą być skompresowane.

Podział formatów

1. Bez kompresji:

- **BMP** – mapa bitowa,
- **XCF (eXperimental Computing Facility)** – mapa bitowa programu GIMP; może przechowywać wiele warstw,
- **XPM** – format zapisu plików przy pomocy znaków ASCII,

2. Kompresja bezstratna:

- **PNG (Portable Network Graphics)** – popularny w www, obsługiwany przez większość przeglądarek, obsługuje kanał alfa,
- **GIF (Graphics Interchange Format)** – popularny w www, obsługuje przezroczystość, animacje,
- **TIFF**

3. Kompresja stratna

- **JPEG (Joint Photographic Experts Group)** - używany zarówno w sieci www, w aparatach cyfrowych,
- **JPEG 2000** – nowsza wersja formatu JPEG, oferująca lepszą kompresję,
- **DjVu** – format stworzony do przechowywania zeskanowanych dokumentów,
- **TIFF (Tagged Image File Format)** – popularny w poligrafii, udostępnia wiele poziomów kompresji (stratnej jak i bezstratnej) umożliwia przechowywanie kanału alfa.

BMP (Graphics Interchange Format)

- opracowany przez IBM dla systemu OS/2, wykorzystywany później Ms Windows,
- wolny od patentów, dostępny, popularny,
- zawiera opcjonalną prostą kompresję bezstratną RLE (krotkość powtórzenia),
- obsługuje tryby RGB oraz RGBA (1-, 4-, 8-, 16-, 24-, 32-bit),
- maksymalna wielkość pliku 32Kx32K oraz 2Gx2G.

Zalety:

- **prostota formatu,**
- duża szybkość przetwarzania skomplikowanych obrazów,
- **powszechność formatu na wielu platformach sprzętowych,**
- prosta technika animacji tzw. technika podwójnego bufora (Double Buffer),
- przygotowanie w pamięci ramki animacji i szybkie jej przepisanie do pamięci obrazu.

Wady:

- **duża pamięciożerność/długi (czas ładowania www),**
- mała kompresja (obrazy 4- i 8-bitowe można kompresować metodą RLE (Run Length Encoding) lub brak kompresji obrazu),
- słabe skalowanie grafik,
- wyświetlanie „od dołu” brak obsługi przezroczystości,
- **niedogodny dla www (podatny na ataki).**

GIF (Graphics Interchange Format), 1987 r. (GIF87a), zmodyfikowany w 1989 r. (GIF89a)

- licencjonowany przez firmę CompuServe INC.
- najlepszy algorytm kompresji bezstratnej LZW (kompresja poziomych bloków pikseli tego samego koloru pozwala czterokrotnie zmniejszyć wielkość pliku);
- obraz zapisywany jest w sposób „poprzeczny” (linia po linii),

- **grafika w 256 kolorach (Indexed Color) 1 do 8 bitowa,**
- maksymalna wielkość obrazu 64Kx64K pikseli,
- format Gif89a pozwala na zapis przezroczystości (tło transparentne, ale piksel jest albo pełni przezroczysty albo ma pełen kolor) i wykorzystanie przeplotu (ładowana jest kolejno 1/8 obrazka, 1/4 obrazka, 1/3 obrazka i pozostała część),
- pozwala na zapis prostych animacji,

Zastosowania

- tworzenie napisów, przycisków nawigacyjnych, znaków graficznych, logo i rysunków ubogich w odcieniu,
- tworzenie obrazów składających się z warstw,
- proste animacje,
- około 80% grafiki webowej.

Wady

- **nie nadaje się do fotografii - ograniczona paleta barw**

PNG (Portable Network Graphics), 1995 r.

- opracowany przez IETF (Thomas Boutell, Tom Lane) jako następca GIF po ogłoszeniu przez Unisys oraz CompuServe roszczeń patentowych dotyczących kompresji LZW używanej w formatach GIF oraz TIFF,
- format wolny od patentów, znoszący ograniczenia GIF, stale ulepszany,
- **kompresja bezstratna** (można zapisać bezstratnie dowolne grafiki RGB),
- algorytm kompresji deflate (słownikowa Lempel-Ziv77=LZ77 + kodowanie Huffmana statystyczne/adaptacyjne),
- **stopniowana przezroczystość (tzw. kanał alfa, 256 stopni przezroczystości),**
- przeplot (pierwszy obraz PNG pojawia się 8x szybciej niż GIF-ie, tekst zapisany w PNG staje się czytelny ok. 2x szybciej niż w GIF-ie) i wyświetlanie sekwencyjne - progresywne (7 poziomów),
- **od 1- do 48-bitowa głębia kolorów** (GIF obsługiwał tylko 256 kolorów),
- obrazy w odcieniach szarości – 16-bit (informacja o jasności),
- obsługa korekcji gamma,
- maksymalna wielkość obrazu 2Gx2G piksele.

Zastosowania:

- PNG-8 (odpowiada GIF) i PNG-24 (odpowiada JPEG),
- wielkość plików PNG-8 jest porównywalna do GIF,
- PNG-24 do obróbki fotografii (pliki większe niż JPEG).

JPEG (Joint Photographic Expert Group), 1983; JPEG 2000 (nowszy)

- stworzony przez C-Cube Microsystems, zbiór standardów ISO/IEC JTC1 SC29 Working Group 1 m.in. IS 10918-1 (ITU-T T.81)
- przetwarzanie obrazów naturalnych charakteryzujących się płynnymi przejściami barw oraz brakiem lub małą ilością ostrych krawędzi i drobnych detali
- realizuje typ kompresji stratnej - sekwencyjny, oparty na dyskretnej transformacie kosinusowej oraz jego różne rozszerzenia; kompresja od 10:1 do 100:1

- przy małym współczynniku kompresji (do 20) strata jakości niezauważalna, przy większym – widoczne kwadratowe pola i zaburzenia wokół konturów
- **tryb progresywny kompresji (płynny stosowany w www), hierarchiczny (fotografia),**
- wielkość pliku zależy od rozmiarów ilustracji, nie zaś od liczby barw
- **maksymalnie kolor 24 bitowy, maksymalny rozmiar obrazu 64Kx64K piksele**
- **JPEG nie udostępnia przezroczystości, tworzenia animacji**

Zastosowania:

- prezentacja fotografii
- obrazy z łagodnymi przejściami tonalnymi
- **obok GIF i PNG najczęściej stosowany format grafiki na stronach WWW**
- nie nadaje się do obrazów o małej liczbie kolorów, gdyż powoduje ich zniekształcenia większe niż powodowane przez zmniejszenie liczby barw GIF

LWF (LuraWave Format) implementacja JPEG 2000

- stosunkowo nowy, stworzony w 1993 r. German National Space Agency (DLR), Stanford FHTW Berlin
- stosowany do obrazów satelitarnych
- stosowany w telefonii komórkowej
- kompresja falkowa, wielostopniowa DWT, dzieli obraz na wysokie i niskie częstotliwości

Zalety:

- daje lepsze efekty od JPEG,
- bardziej popularny niż nowszy DjVu,
- możliwa jest dwupoziomowa ochrona,
- praw autorskich – numer lub znak wodny,
- obsługuje CMYK i inne przestrzenie barwne,
- pozwala dodawać profile ICC,
- **4-bitowa skala szarości – 24-bitowy kolor.**

Wady:

- wymaga pluginów lub osobnej aplikacji

TIFF (ang. Tagged Image File Format)

- opracowany w 1986 przez Aldus Corporation, Microsoft i Hewlett-Packard do druku postscriptowego,
- najbardziej rozpowszechniony i uniwersalny z formatów,
- umożliwia stosowanie **kompresji bezstratnej typu LZW oraz CCITT.4 oraz CCITT.6 stratnej JPEG (TIFF 6.0)** - tag TIFF Tag 259 przechowuje informację o rodzaju kompresji

- obsługuje warstwy (strukturę drzewa)
- typ mime zdefiniowany RFC 3302, RFC 3949, RFC 3950
- obsługuje przestrzeń CMYK, YCbCr, HalftoneHints, CIE L*a*b*
- **używany w DTP (podstawowy format), grafice 3D, w obrazowaniu medycznym itd.**

Zalety - pozwala na zapisywanie obrazów stworzonych:

- w trybie kreskowym,
- skali szarości i wielu głębiach bitowych koloru,
- **przechowuje ścieżki i kanały alfa,**
- **profile koloru,**
- **przechowuje warstwy,**
- komentarze tekstowe (udostępnia własne tagi wykorzystane w Geo TIFF, NASA).

PORÓWNANIE

Opcje	BMP	GIF	PNG	JPG /JPEG	TIF/TIFF
głębina bitowa	24	8	48	24	16, 24 lub 32
przestrzeń barwna	RGB, RGBA	indeksowana	RGB, RGBA	RGB, CMYK	CMYK, CIE L*a*b*
przezroczystość	NIE	TAK (jest-nie ma)	TAK (256 stopni/piksel)	NIE	NIE
przeplot/progresja	NIE/NIE	TAK/NIE	TAK/TAK	NIE/TAK	NIE
animacja	NIE (z wyjątkiem)	TAK	NIE	NIE	NIE
kompresja	NIE (ew. RLC)	TAK (bezstratna, LZW)	TAK (bezstratna, LZ77)	TAK (stratna!!!)	TAK (stratna JPG, bezstratna LZW)

rozmiar pliku	BARDZO DUŻY	ŚREDNI	MAŁY	BARDZO MAŁY	DUŻY
maksymalny rozmiar	32Kx32K oraz 2Gx2G	64Kx64K	2Gx2G	64Kx64K	2Gx2G
Typ mime (IANA)					
[RFC6838]	image/bmp	image/gif [RFC2045] [RFC2046]	[Glenn Randers Pehrson]	image/png [RFC 1341]	image/tif [RFC 3302, RFC 3949, RFC 3950]
Zastosowanie	zdjęcia do dalszej obróbki	ozdobniki graficzne, przyciski, banery	ozdobniki graficzne, przyciski	zdjęcia dla www	dtp grafika medyczna

◀ Zadanie 7

Przejdź do...



Kryteria doboru formatu ▶

Nawigacja

Kokpit

Strona główna

Strony

Moje kursy

MAD - wykład

MAD - ćwiczenia

GU - wykład

Uczestnicy

Oceny

GRAFIKA UŻYTKOWA - WYKŁAD

Informacje

Wykład 1

Wykład 2

Wykład 3

Wykład 4 (23.03.2020)

Wykład 5 (30.03.2020)

Wykład 6 (6.04.2020)

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



1

Patryk Obuchowicz

online M&J



Kokpit > Moje kursy > GU - wykład > Wykład 6 (6.04.2020) > Kryteria doboru formatu

Kryteria doboru formatu

Kryteria doboru formatu

Grafika prezentacyjna i webowa:

- zależnie od rodzaju grafiki formaty gif, jpg, png

Grafika na potrzeby poligrafii

- tif, ewentualnie PNG-24

Konsekwencje doboru złego formatu:

- niezamierzone obcięcie głębi,
- zła reprodukcja kolorów (przekłamania kolorów),
- brak przejść tonalnych lub skokowe przejścia tonalne ,
- brak obsługi przezroczystości nieczytelność grafiki,
- brak uniwersalności,
- zabieranie zbyt dużej ilości miejsca spowalnianie dalszej obróbki lub ładowania stron.

Zalecenia grafik dla dla www:

- 72-120 ppi
- 30-35 kB
- czas wyświetlania strony nie powinien być dłuższy niż 60 sekund

Standardowe metody doboru formatu:

- zastosowanie odpowiedniej rozdzielczości (ppi/dpi)
- ograniczenie liczby kolorów do najważniejszych w paletie (głębii bitowej)
- (poprawienie kompresowalności pliku zapisywanego w formacie GIF)
- przeskalowanie/przycięcie (zmniejszenie rozmiarów)
- (JPEG redukuje wielkość pliku, ale nie liczbę kolorów)
- zastosowanie kompresji (bezstratnej/stratnej)
- odpowiedni format zapisu grafiki rastrowej (BMP, TIFF, PNG, JPEG, GIF)

Komplementarne metody doboru formatu:

- format dedykowany w przypadku nietypowych zastosowań (DjVu, RAW)
- zmiana z grafiki rastrowej na grafikę wektorową (wektoryzacja, trasowanie) eliminuje zależność grafiki od rozdzielczości!

The screenshot shows a web-based learning platform with a sidebar navigation menu. The menu items include:

- ◀ Formaty plików
- Przejdz do... ▾
- Zadanie 8 ▶
- Nawigacja**
- Kokpit
- 🏠 Strona główna
- Strony
- Moje kursy
 - MAD - wykład
 - MAD - ćwiczenia
 - GU - wykład
 - Uczestnicy
 - █ Oceny
- GRAFIKA UŻYTKOWA - WYKŁAD**
- Informacje
- Wykład 1
- Wykład 2
- Wykład 3
- Wykład 4 (23.03.2020)
- Wykład 5 (30.03.2020)
- Wykład 6 (6.04.2020)
 - ❓ Lista obecności 6.04.2020 - kliknij i zaznacz
 - 📄 Separacja RGB
 - 📄 Zasady doboru barw
 - 📝 Zadanie 6
 - 📄 Bitmapa i jej charakterystyka
 - 📄 Kryteria doboru rozdzielczości i głębi
 - 📝 Zadanie 7
 - 📄 Formaty plików
 - 📄 Kryteria doboru formatu**
 - 📝 Zadanie 8
- Wykład 7 (20.04.2020)
- GU - laboratorium
- WWW - laboratorium

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



Patryk Obuchowicz -



online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 7 (20.04.2020) > Przetwarzanie grafiki rastrowej

Przetwarzanie grafiki rastrowej

Przetwarzanie grafiki rastrowej to operacje w których dane wejściowe, jak i wyjściowe są obrazami rastrowymi.

Zaliczamy do nich:

- **przekształcenia geometryczne**
 - zmiana rozdzielczości – zwiększenie, zmniejszenie;
 - różnego rodzaju deformacje – liniowe i nieliniowe;
- **przekształcenia punktowe/bezkontekstowe**
(operacje niezależne od otoczenia analizowanego piksela)
 - operacje na histogramie obrazu (wyrównanie, rozciąganie);
 - operacje logiczne i arytmetyczne,
- **przekształcenia kontekstowe – filtry**
(operacje zależne od otoczenia analizowanego piksela (problem pikseli brzegowych))
 - **filtры liniowe**, w których filtracja przybiera postać liniowej kombinacji określonych pikseli obrazu wejściowego
 - filtre dolnoprzepustowe – wygładzające, usuwają szumy*
 - filtre górnoprzepustowe – wyostrzające, uwypuklają kontury, krawędzie*
 - **filtre nieliniowe**, w których filtracja przybiera postać nieliniowej funkcji określonych pikseli obrazu wejściowego
 - filtre logiczne - stosowane w przypadku obrazów binarnych;*
 - filtre adaptacyjne - swoje działanie dostosowują do cech obrazu;*
 - filtre medianowe - bardzo efektywne, wykorzystują medianę punktów sąsiadujących, zamiast wartości średniej;*
 - **erozja, dylatacja, otwarcie, zamknięcie, szkieletyzacja, obcinanie**

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



1

Patryk Obuchowicz -



online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 7 (20.04.2020) > Przekształcenia geometryczne - zmiana rozdzielczość...

Przekształcenia geometryczne - zmiana rozdzielczości i deformacja

PRZEKSZTAŁCENIA GEOMETRYCZNE:

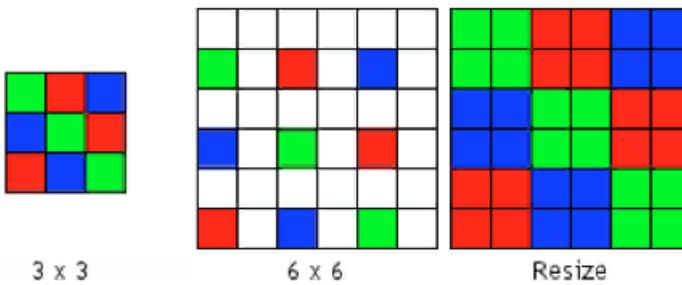
Przekształcenia geometryczne pozwalają na zmianę formatu obrazów (wielkość, proporcje) co jest przydatne przy tworzeniu skalowalnych stron www, łączeniu obrazów (operacje arytmetyczne), porównaniu (wyznaczanie różnic), korekcji (eliminowanie zniekształceń i wad wprowadzanych przez sprzęt - obiektywy) lub dostosowaniu/uproszczeniu obrazu do dalszego przetwarzania. W tej grupie wyróżniamy operacje takie jak:

- **zmiana rozdzielczości – zwiększenie, zmniejszenie**
zmiana rozdzielczości przestrzennej (metody interpolacyjne)
- **różnego rodzaju deformacje – liniowe i nieliniowe**
 - *przesunięcie (translacja),*
 - *odbicie symetryczne,*
 - *obrót,*
 - *zniekształcenia,*
 - *powielanie skrajnych wierszy lub kolumn*

Zmiana rozdzielczości przestrzennej

- **Metoda najbliższego sąsiada:**
 - wartość pikseli nowopowstałych w obrazie obliczana jest poprzez wybór wartości jednego z czterech najbliższej położonych pikseli obrazu źródłowego - analizowany piksel przyjmuje wartość piksela znajdującego się najbliżej niego w sensie odległości euklidesowej,
 - w przypadku zwiększenia rozdzielczości – powielenie pikseli,
 - w przypadku zmniejszania rozdzielczości – eliminacja niektórych pikseli

- w przypadku zmniejszania rozdzielczości – eliminacja niektórych pikseli.



powiększenie 200%
jeden piksel zostanie powiększony
do powierzchni 2×2
tj. 4 pikseli

Wady:

- brak nowych wartości wprowadzanych do obrazu

Zalety:

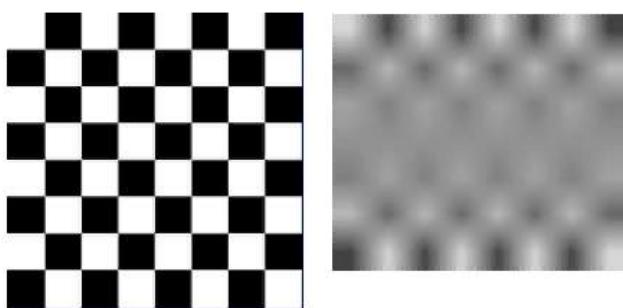
- brak interpolacji nie powoduje zmniejszenia ostrości krawędzi
- zadowalające wyniki jedynie w przypadku zwielokrotniania rozdzielczości
- w innych przypadkach rezultat nie jest optymalny

UWAGA:

Jeżeli dwa piksele obrazu źródłowego są równo oddalone od nowego piksela, wybór może przebiegać według dowolnej metody, lecz powinna być ona stosowana konsekwentnie względem wszystkich nowopowstałych pikseli.

• Metoda interpolacji dwuliniowej

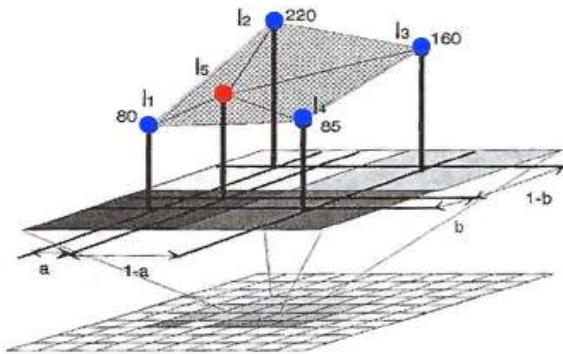
- wszystkie piksele z sąsiedztwa (a nie tylko jeden) mają wpływ na wartość nowopowstałego piksela
- im bliżej analizowanego punktu położony jest piksel z jego sąsiedztwa, z tym większą wagą wpływa na wartość obliczanego piksela
- interpolacji liniowej poddawane są piksele występujące najczęściej w układzie 4-sąsiedztwa



4 piksele w najbliższym 2×2 sąsiedztwie

piksele w obrazie

Wartości nowopowstałych pikseli obliczane są następująco:



$$I_6 = I_1(1-a)(1-b) + I_2(1-a)b + I_3ab + I_4a(1-b)$$

Wady:

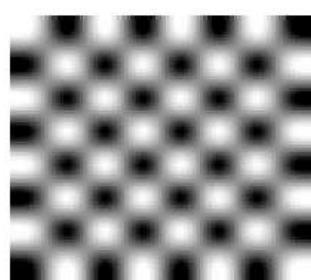
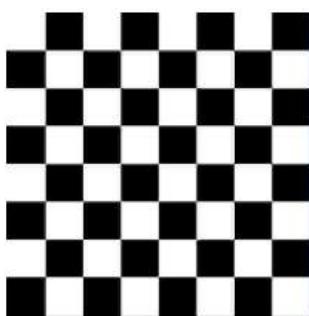
- kontury obiektu ulegają rozmyciu

Zalety:

- w tej metodzie powstają nowe wartości pikseli (nieobecne w obrazie źródłowym)

- **Metoda interpolacji dwukubicznej**

- wszystkie piksele z sąsiedztwa (a nie tylko jeden) mają wpływ na wartość nowopowstałego piksela
- im bliżej analizowanego punktu położony jest piksel z jego sąsiedztwa, z tym większą wagą wpływa na wartość obliczanego piksela
- branych jest pod uwagę 16 sąsiednich pikseli



16 pikseli w najbliższym 4×4 sąsiedztwie

piksla w obrazie

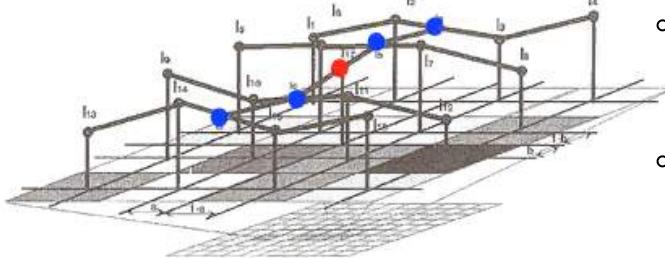
Wartości nowopowstałych pikseli obliczane są ze wzoru:

$$I_{17} = (I_d - I_c + I_b - I_a)b^3 + (I_c - I_d - 2I_b + 2I_a)b^2 + (I_c - I_a)b + I_d$$

$$I_i = (I_4 - I_3 + I_2 - I_1)a^3 + (I_3 - I_4 - 2I_2 + 2I_1)a^2 + (I_3 - I_1)a + I_2 \quad \text{gdzie } i = a, b, c, d$$

Wady:

- znaczna złożoność obliczeniowa



- rozmycie krawędzi

Zalety:

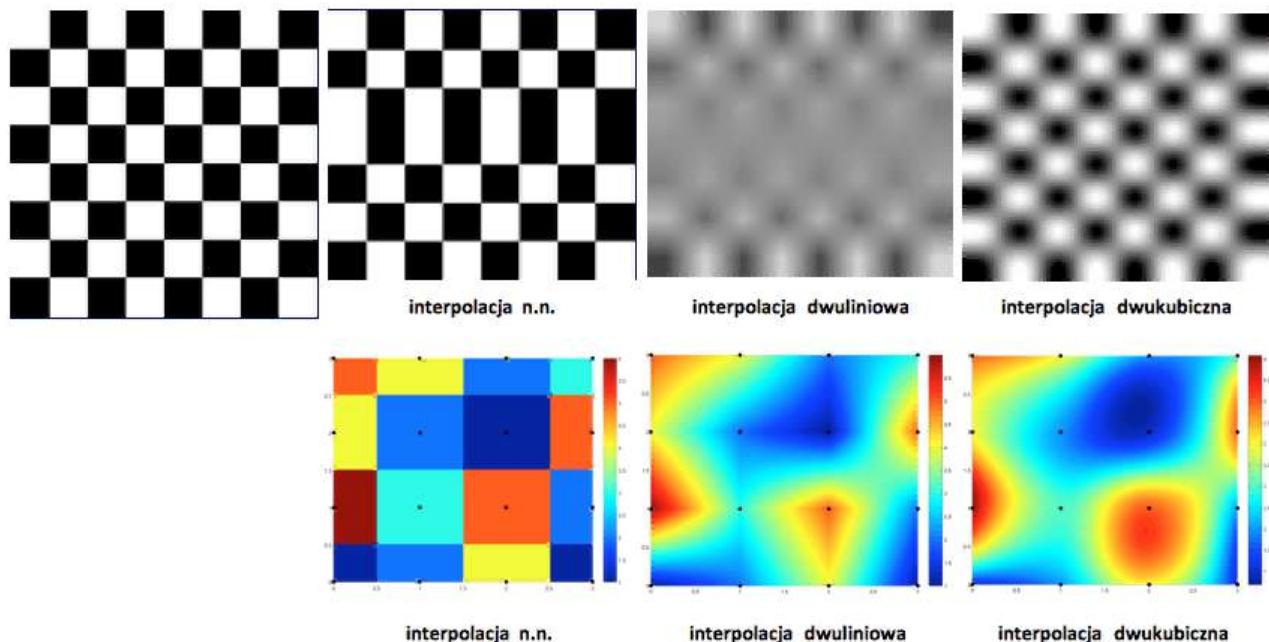
- w tej metodzie powstaj \acute{a} nowe wartości pikseli (nieobecne w obrazie źródłowym)

UWAGA:

Metoda ta jest najczęściej używana przez oprogramowanie do edycji obrazu,

sterowniki drukarek i aparatów cyfrowych w celu ponownego próbkowania obrazów.

Porównanie różnych metod interpolacji:



Deformacje

Większość podstawowych operacji można opisać za pomocą przekształceń afinicznych, które definiuje się wykorzystując operację mnożenia macierzy:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} y \\ y \\ 1 \end{bmatrix}$$

gdzie x',y' współrzędne po transformacji, x,y przed transformacją a macierz $A = [a_{ij}]$ opisuje przekształcenie afiniiczne.

- **Translacja (przesunięcie):**

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \left[\begin{array}{ccc|c} 1 & 0 & 0 & dx \\ 0 & 1 & dy & 0 \\ 0 & 0 & 1 & 0 \end{array} \right] \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

gdzie dx określa przesunięcie w poziomie a dy w pionie, tzn. [dx,dy] jest wektorem translacji.

- **Skalowanie:**

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \left[\begin{array}{ccc|c} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & 1 & 0 \end{array} \right] \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

gdzie a określa skalowanie w poziomie a b w pionie; w przypadku a=b skalowanie jest proporcjonalne, natomiast gdy a>0 a, b=1 to jest to skalowanie panoramiczne.

- **Obrót (rotacja):**

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \left[\begin{array}{ccc|c} \cos\alpha & -\sin\alpha & 0 & 0 \\ \sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \end{array} \right] \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

gdzie α określa kąt rotacji.

- **Naprężenie ścinające:**

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \left[\begin{array}{ccc|c} 1 & a & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{array} \right] \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

gdzie a określa parametr ścinania.

Składanie transformacji afonicznych wymaga dokonania mnożenia macierzy opisujących te transformacje. Należy pamiętać, że składanie operacji nie jest przemienne.



ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



Patryk Obuchowicz -



online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 7 (20.04.2020) > Przekształcenia punktowe/bezkontekstowe - operacje...

Przekształcenia punktowe/bezkontekstowe - operacje arytmetyczne

PRZEKSZTAŁCENIA PUNKTOWE/BEZKONTEKSTOWE

operacje niezależne od otoczenia analizowanego piksela

OPERACJE ARYTMETYCZNE:

Wykonywane są na każdym pikselu obrazu źródłowego, którego barwa (lub stopień jasności) jest przeliczana w oparciu o zadaną funkcję. Każdej wartości barwy (lub jasności) obrazu źródłowego przypisywana jest wartość, jaką uzyska w obrazie wynikowym po wykonaniu danego przekształcenia:

$$j_i^W(x,y) = \psi(j_i(x,y))$$

gdzie ψ to funkcja (liniowa lub nieliniowa), która modyfikuje wartości obrazu wejściowego.

Operacje arytmetyczne liniowe

- **dodanie lub odjęcie** od obrazu stałej wartości powoduje jego rozjaśnienie lub przyćiemnienie (zwiększenie lub zmniejszenie jasności).

$$j_i^W(x,y) = j_i(x,y) \pm b \quad \text{gdzie } b = \text{const}$$

Istnieje niebezpieczeństwo przekroczenia maksymalnej lub minimalnej wartości jasności.



- **mnożenie przez stałą** - zwykle stosowane w celu poprawy jakości

$$j_i^w(x,y) = a \cdot j_i(x,y) \quad a = \text{const}$$

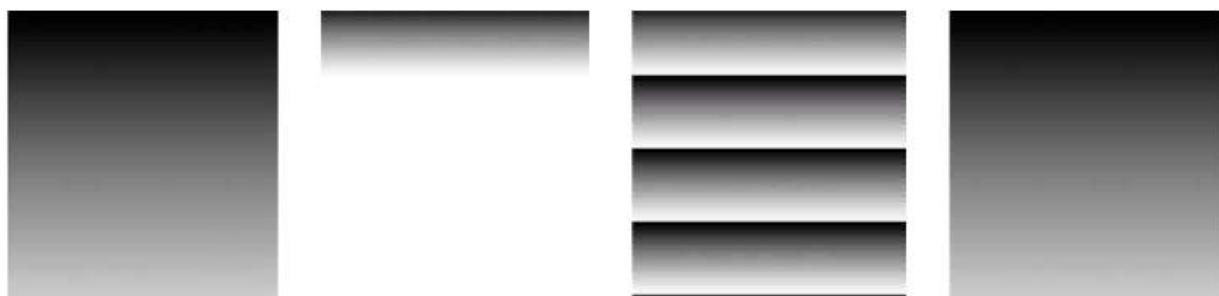
W wyniku uzyskujemy:

- zwiększenie kontrastu (większe zróżnicowanie szarości), jeżeli $a > 1$,
- zmniejszenie kontrastu, jeżeli $0 < a < 1$.

Przy przekroczeniu maksymalnej lub minimalnej wartości jasności może nastąpić utrata części danych.



Zapobieganie przekroczeniu maksymalnej bądź minimalnej wartości parametru jasności (lub barwy):



stosowanie zasad
arytmetyki
przepiętniowej
(obcinanie wartości na
poziomie dopuszczalnego
maksimum czy minimum)

modulo p (p=256)

normalizacja jasności

wykonywane są dla każdego piksela zgodnie ze wzorem:

$$j_i^W(x,y) = \varphi(j_i(x,y)) \quad \text{gdzie } \varphi - \text{funkcja nieliniowa}$$

- **potęgowanie:**

$$j_i^W(x,y) = [j_i(x,y)]^\alpha \quad \text{gdzie } \alpha > 0$$

Jeżeli założymy, że minimalna jasność wynosi 0, to wzór przyjmuje postać:

$$j_i^W(x,y) = 255 \left[\frac{j_i(x,y)}{j_{max}(x,y)} \right]^\alpha$$

to dla

- **$\alpha=2$ lub $\alpha=3$** - większe zróżnicowanie jasności
po zastosowaniu normalizacji, powoduje przyciemnienie obrazu oraz zmniejszenie zróżnicowania niskich wartości parametru jasności.
- **$\alpha < 1$** - zwiększenie kontrastu w ciemnych partiach obrazu (o małej jasności)
(korekcja gamma ma na celu poprawę percepcji zmian jasności wraz ze zmianą luminancji)

$\alpha < 1$



$\alpha = 2$



- **logarytmowanie obrazu:**

Przy założeniach podobnych, jak dla potęgowania, otrzymujemy wzór:

$$j_i^W(x,y) = \log[1 + j_i(x,y)]$$

Ze względu na nieokreśloność funkcji w zerze, wykres musi być przesunięty o jeden.

$$j_i^W(x,y) = 255 \cdot \frac{\log(1 + j_i(x,y))}{\log(1 + j_{max}(x,y))}$$

W wyniku operacji następuje silne rozjaśnienie ciemnych partii obrazu.



Problemy związane ze stosowaniem operacji arytmetycznych

Przy konieczności przeprowadzania obliczeń dla każdego piksela przekształcenia piksel po pikselu są bardzo czasochłonne.

PRZYKŁAD:

Należy dokonać korekty jasności obrazu 3000x3000 (obraz w skali szarości). Wartość jasności każdego z pikseli obrazu należy przeliczyć według wzoru na potęgowanie.

Rozwiązanie:

Pikseli w obrazie jest 9 mln. zatem wzór na przeliczenie jasności będzie musiał zostać zastosowany 9 mln. razy i za każdym razem wynik będzie wstawiany do tablicy pikseli. Ale obraz jest w skali szarości więc dane wejściowe zawierają 256 różnych wartości - jasności pikseli w przedziale 0 do 255 a nie 9 milionów. Stosujemy tablicę przekodowania - tablicę LUT (Look Up Table) - o liczbie elementów równej liczbie możliwych wartości na wejściu 256 elementów). W każdej komórce tablicy wpisujemy wynik przeliczania wzoru dla poszczególnych wartości pikseli: $LUT[i] = \text{funkcja}(i)$; gdzie funkcja to przekształcenie, które chcemy zrealizować.

Wartości poziomów jasności wszystkich punktów - tabela LUT:

wartość jasności obrazu źródłowego	wartość jasności obrazu wynikowego
00000000	J_0
00000001	J_1
...	...
11111111	J_{255}

I UZUŁASZ PRZETWARZANIA OBRAZU KORZYSTAJĄC Z TABLICY LUT - POMINIĘTYM WŁAŚCIWOŚĆI Z TABLICY ZAMIASM WYNIKÓW ponownie.

Zastosowanie tablicy LUT zmniejsza liczbę obliczeń z 9 milionów do 256.

◀ Przekształcenia geometryczne - zmiana rozdzielczości i deformacja

Przejdz do...

Przekształcenia punktowe/bezkontekstowe - histogram ►

Nawigacja

Kokpit

🏠 Strona główna

Strony

Moje kursy

MAD - wykład

MAD - ćwiczenia

GU - wykład

Uczestnicy

☰ Oceny

GRAFIKA UŻYTKOWA - WYKŁAD

Informacje

Wykład 1

Wykład 2

Wykład 3

Wykład 4 (23.03.2020)

Wykład 5 (30.03.2020)

Wykład 6 (6.04.2020)

Wykład 7 (20.04.2020)

❔ Lista obecności 20.04.2020 - kliknij i zaznacz

📄 Przetwarzanie grafiki rastrowej

📄 Przekształcenia geometryczne - zmiana rozdzielczość...

📄 Przekształcenia punktowe/bezkontekstowe - operacje...

📄 Przekształcenia punktowe/bezkontekstowe - histogram

📄 Histogram obrazów monochromatycznych

📄 Histogram obrazów barwnych (RGB)

📝 Zadanie 9

GU - laboratorium

WWW - laboratorium

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



1

Patryk Obuchowicz -



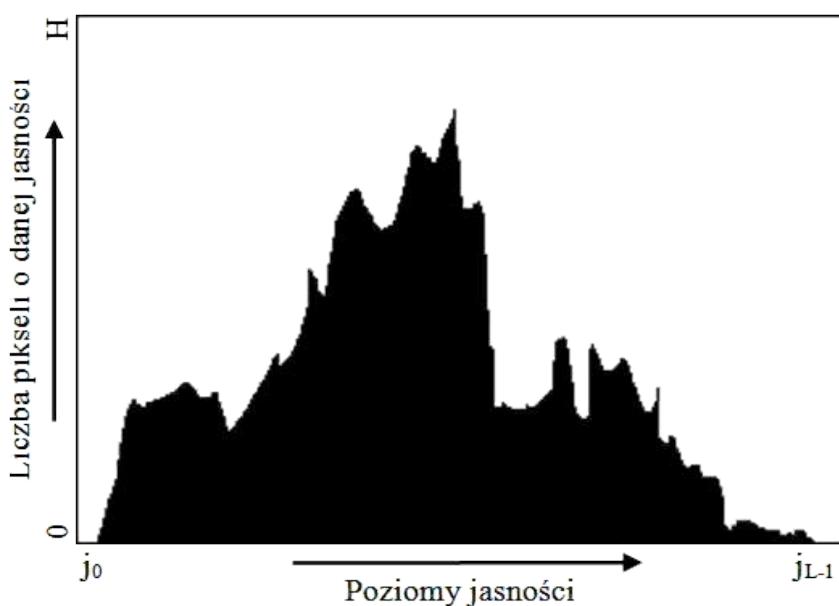
online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 7 (20.04.2020) > Przekształcenia punktowe/bezkontekstowe - histogram

Przekształcenia punktowe/bezkontekstowe - histogram

HISTOGRAM

Histogram to jeden z graficznych sposobów przedstawiania rozkładu cechy. **Histogram obrazu jest to funkcja, pozwalająca scharakteryzować w sposób globalny rozkład pikseli o zadanym poziomie jasności.**



$$H(j_i) = n_i = \sum_{x=0}^M \sum_{y=0}^N k_i(x,y) \quad i=0, 1, 2, 3, \dots, L-1$$

$$k_i(x,y) = \begin{cases} 1 & j(x,y) = i \\ 0 & p.p. \end{cases}$$

gdzie

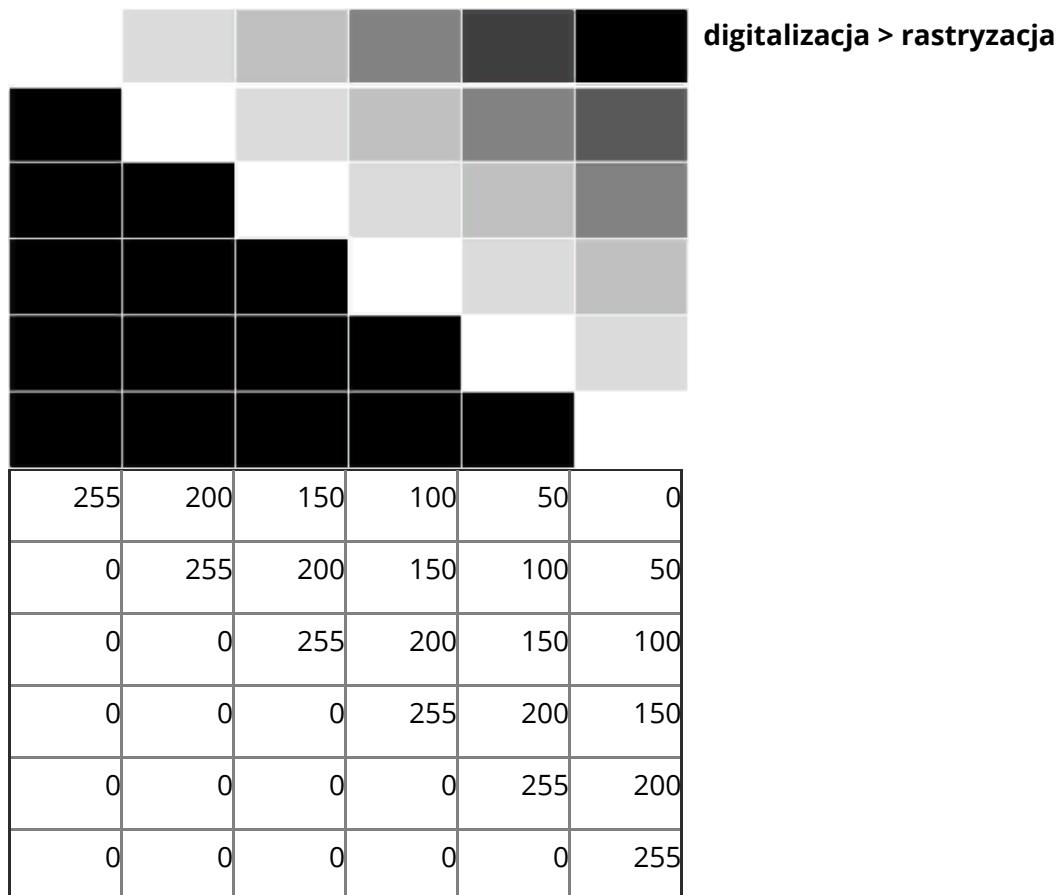
M – rozmiar obrazu w kierunku x,

N – rozmiar obrazu w kierunku y,

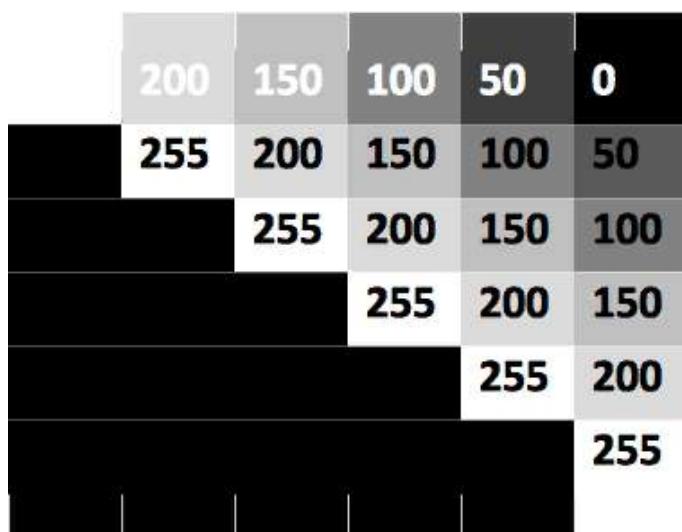
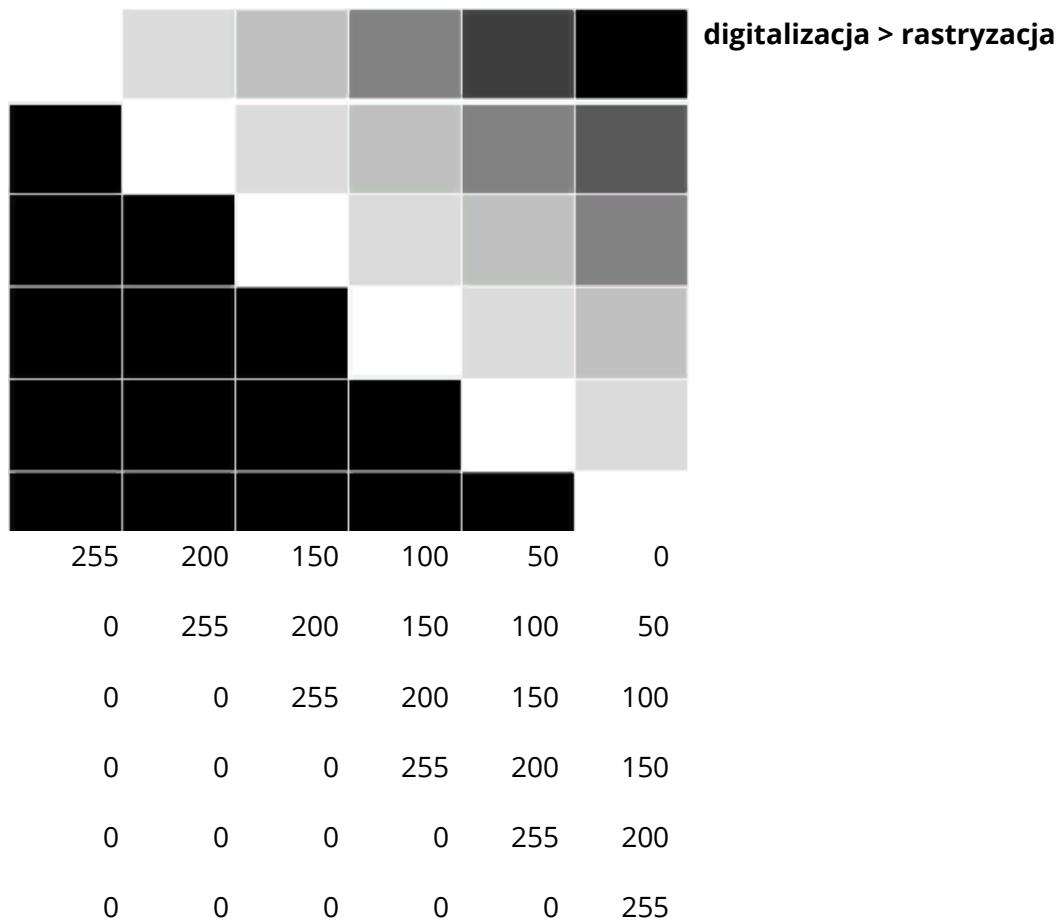
Na osi odciętych histogramu oznaczane są wartości jasności pikseli obrazu j_0, j_1, \dots, j_{L-1} , (L liczba osiągalnych poziomów jasności).

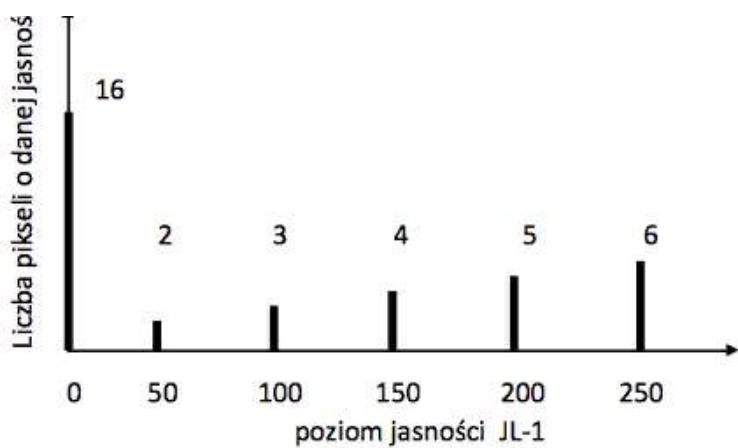
Natomiast na osi rzędnych znajdują się wartości histogramu $H(j_0), H(j_1), \dots, H(j_{L-1})$ reprezentujące liczbę pikseli w obrazie o danych poziomach jasności.

Przykładowy obraz i histogram ilościowy cechy jasności tego obrazu

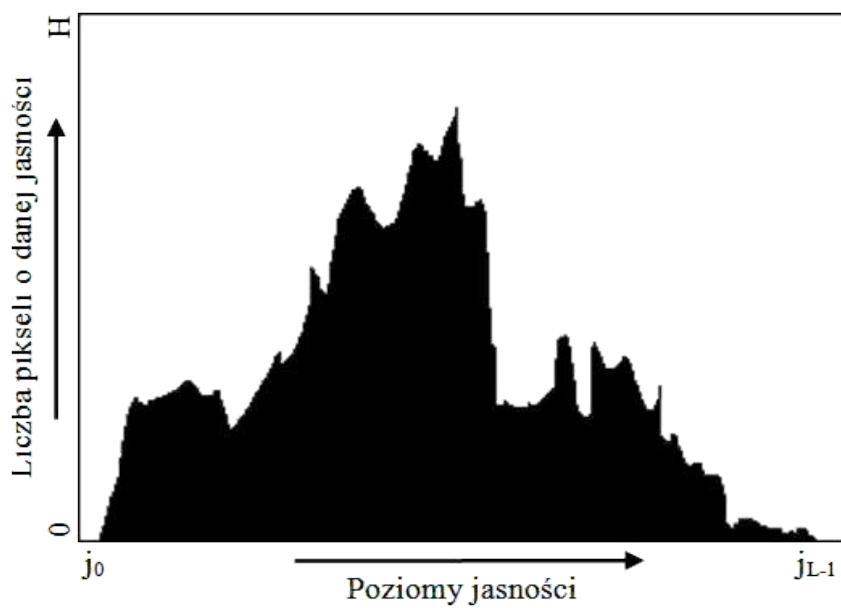


W przykadku obrazu cyfrowego w reprezentacji ustrojowej wektor jasności J_{L-1} powinien skadac się z liczb całkowitych z przedziału $<0,256$)





W praktyce często wykorzystuje się **histogram znormalizowany**.



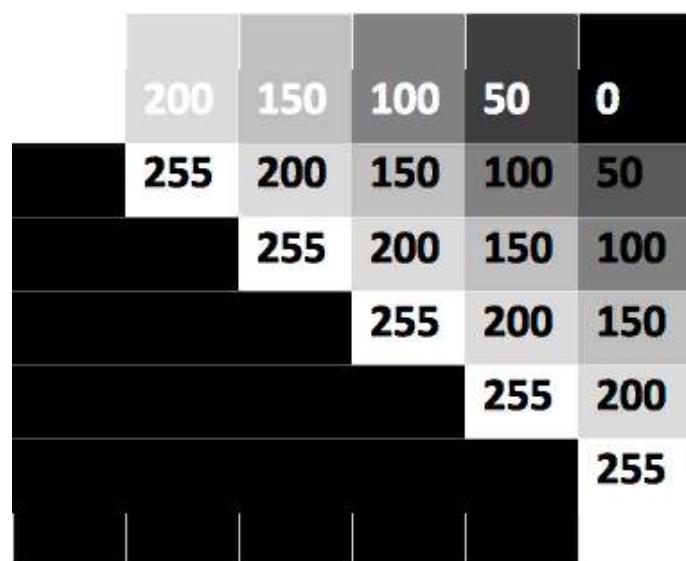
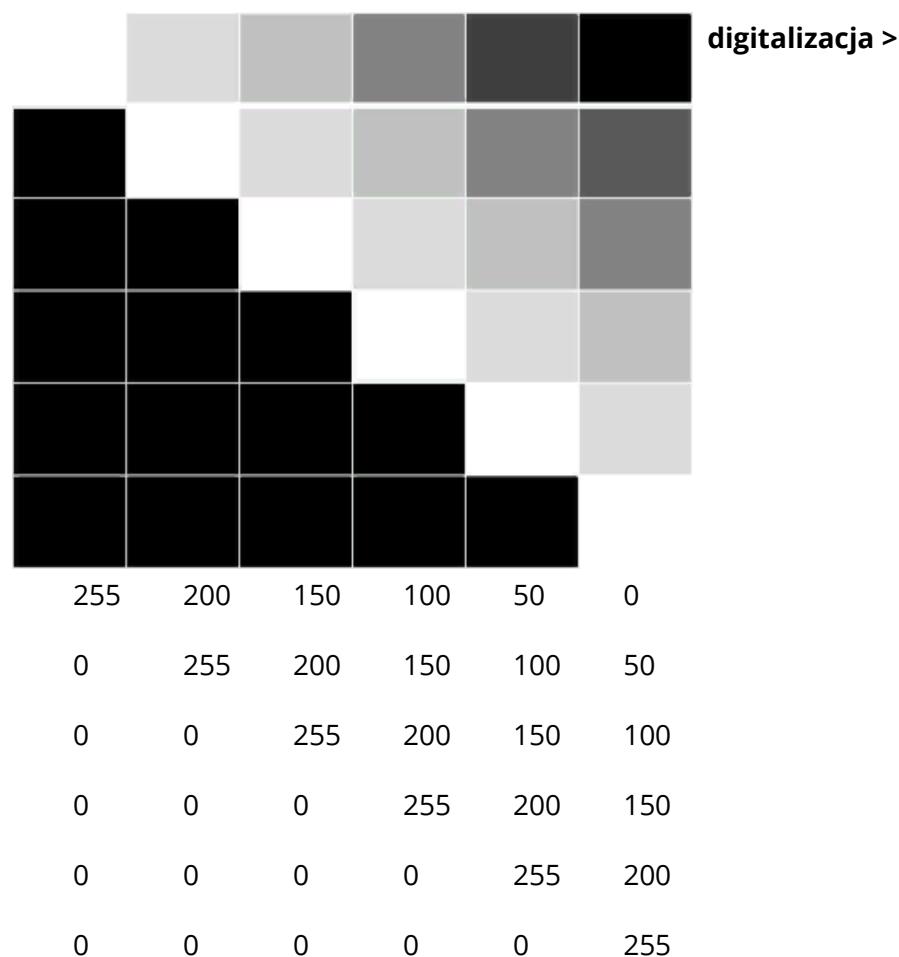
$$H(j_i) = n_i = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} k_i(x, y) \quad i = 0, 1, 2, 3, \dots, L-1$$

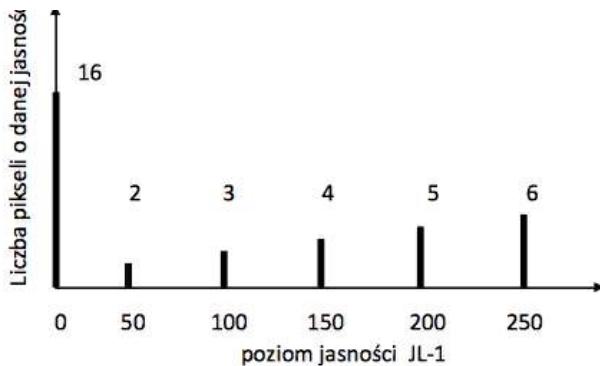
$$k_i(x, y) = \begin{cases} 1 & j(x, y) = i \\ 0 & p.p. \end{cases}$$

- oś odciętych: wartości jasności pikseli obrazu j_0, j_1, \dots, j_{L-1} ,

- oś rzędnych: unormowane wartości histogramu $H(j_0), H(j_1), \dots, H(j_{L-1})$

Przykładowy obraz i histogram ilościowy cechy jasności tego obrazu $6 \times 6 = 36$





Typowe niedoskonałości obrazów:

- **zbyt ciemny**
w obrazie występują głównie piksele o niskiej jasności
brak pikseli o wysokiej jasności
- **zbyt jasny**
w obrazie występują głównie piksele o wysokiej jasności
brak pikseli o niskiej jasności
- **mały kontrast**
w obrazie występują piksele o jasności z wąskiego zakresu
- **duży kontrast**
w obrazie występują piksele o wszystkich wartościach jasności
- **poprawny**
wykorzystanie pełnego zakresu poziomów jasności

Metody poprawy jakości:

- rozciąganie histogramu
- wyrównywanie histogramu

Rozszerzanie zakresu jasności:

Jeżeli zakres dostępnych poziomów danej cechy nie jest w pełni wykorzystany, stosuje się modyfikacje histogramu, prowadzące do zmiany kształtu jego obwiedni.

Zastosowanie:

zakres jasności pikseli obrazu nie obejmuje całego dostępnego zakresu

Transformacja:

piksele o wartościach minimalnych i maksymalnych przyjmują dostępne ekstremalne jasności, natomiast pomiędzy pośrednimi odległości zwiększały się.

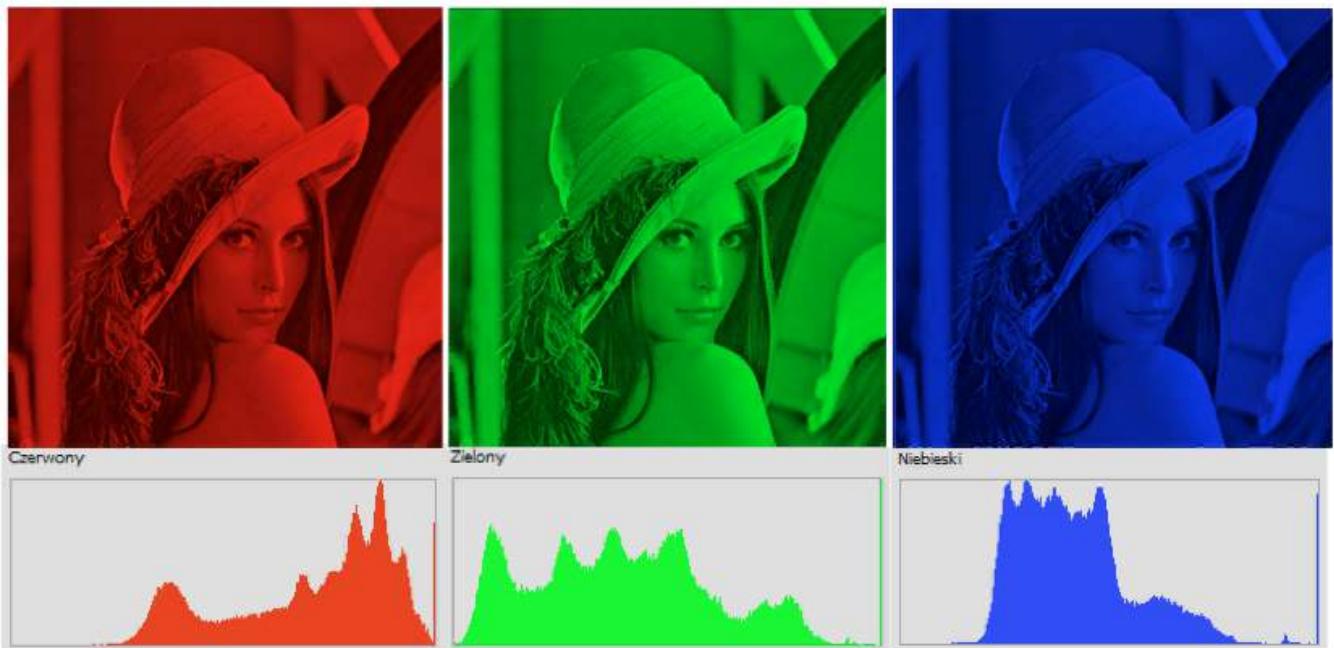
$$J_W(x,y) = \frac{255}{j_{max} - j_{min}} (j(x,y) - j_{min}) \quad j_{min} \leq j(x,y) \leq j_{max}$$

Rezultat :

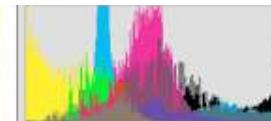
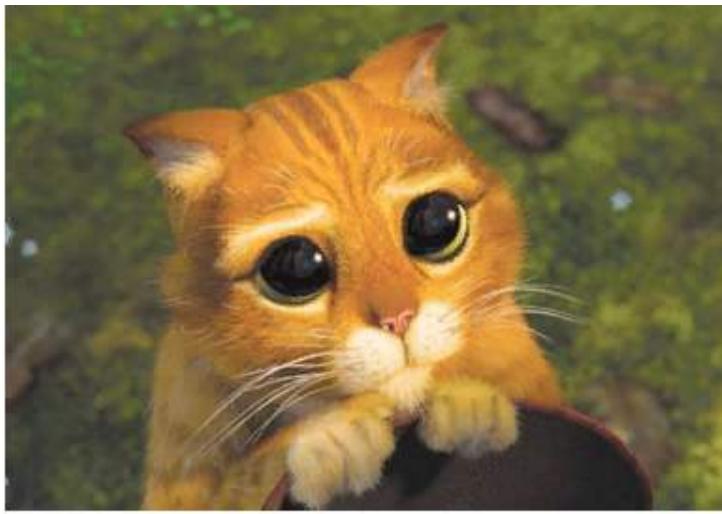
zwiększenie kontrastowości obrazu

HISTOGRAM RGB

Separacja obrazu na składowe (np. R, G, B) pozwala wyznaczyć histogramy składowe (histogramy każdego kanału):



Separacja obrazu na składowe (np. C, M, Y, K) pozwala wyznaczyć histogramy składowe:



Źródło: Cały obraz
Średnia: 346.88
Odch. stand.: 54.37
Median: 39
Liczba pikseli: 290560
Poziom bufora: 1

Niebiesko-czarny

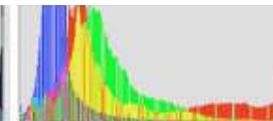
Karmazynowy

Zółty

Czarny

Niebieski

Czerwony



Źródło: Cały obraz
Średnia: 346.88
Odch. stand.: 54.37
Median: 39
Liczba pikseli: 290560
Poziom bufora: 1

Ciemny



RGB

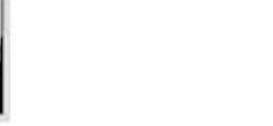


CMYK



Zółty

Czarny



Niebieski

◀ Przekształcenia punktowe/bezkontekstowe - operacje arytmetyczne

Przejdz do...

Histogram obrazów monochromatycznych ►

Nawigacja

Kokpit

Strona główna

Strony

Moje kursy

MAD - wykład

MAD - ćwiczenia

GU - wykład

Uczestnicy

Oceny

GRAFIKA UŻYTKOWA - WYKŁAD

Informacje

Wykład 1

Wykład 2

Wykład 3

Wykład 4 (23.03.2020)

Wykład 5 (30.03.2020)

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



1

Patryk Obuchowicz -



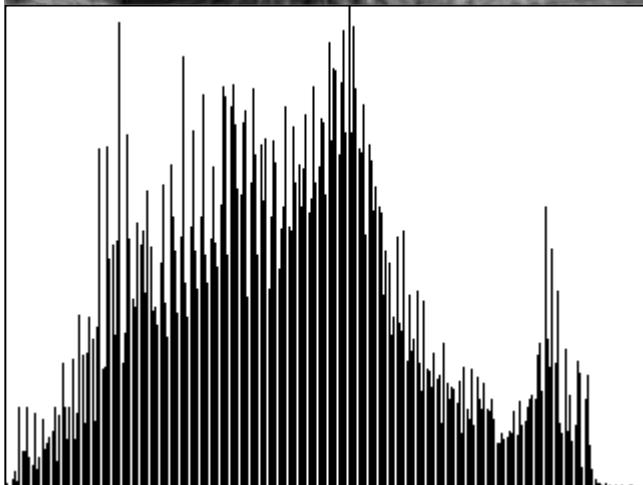
online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 7 (20.04.2020) > Histogram obrazów monochromatycznych

Histogram obrazów monochromatycznych

Obrazek:

Kot ▾



◀ Przekształcenia punktowe/bezkontekstowe - histogram

Przejdz do...

Histogram obrazów barwnych (RGB) ►

Nawigacja

Kokpit

Strona główna

Strony

Moje kursy

MAD - wykład

MAD - ćwiczenia

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



1



Patryk Obuchowicz

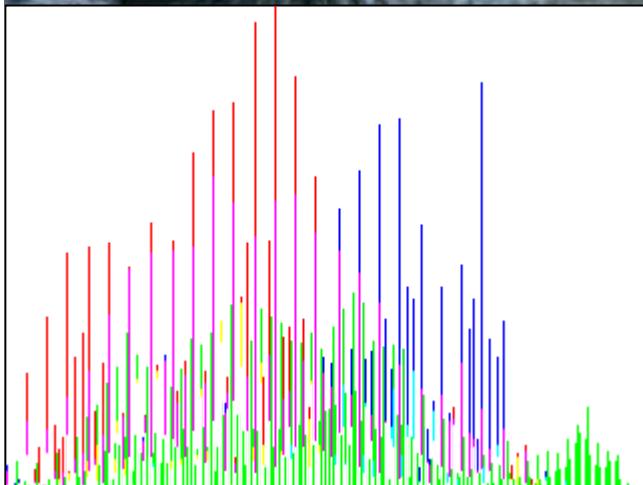


online M&J

Kokpit > Moje kursy > GU - wykład > Wykład 7 (20.04.2020) > Histogram obrazów barwnych (RGB)

Histogram obrazów barwnych (RGB)

Obrazek: ▾



Rodzaj histogramu: RGB ▼

◀ Histogram obrazów monochromatycznych

Przejdź do...

Zadanie 9 ►



Nawigacja

Kokpit

🏠 Strona główna

Strony

Moje kursy

MAD - wykład



Przekształcenia kontekstowe - filtracja

FILTRACJA

Filtrowanie obrazu to transformacja złożona, **kontekstowa**. Operacje wykonywane są nie tylko na pojedynczym pikselu, ale także na pikselach należących do jego otoczenia, z wyłączeniem pikseli brzegowych.

Z matematycznego punktu widzenia filtr jest funkcją wieloargumentową przekształcającą obraz źródłowy w nowy metodą „piksel po pikselu”.



Podział filtrów:

- **filtracja liniowa** w oparciu o liniową kombinację pikseli obrazu źródłowego z reguły są prosta w działaniu ale o ograniczonych możliwościach
- **filtracja nieliniowa**, skomplikowana, ale stwarzająca bardzo duże możliwości (stosuje się maski przyjmujące jako wynikową wartość piksela wartość statystyczną – wartość jednego z punktów otoczenia)

Cechy:

- Podstawową wadą filtrów liniowych jest fakt, że filtrują informację w obrazie, jak i szum.
- Filtry nieliniowe, które pozwalają na oddzielenie zakłóceń od zasadniczej treści obrazu.

Cele filtracji obrazu:

- redukcja niepożdanego szumu,
- poprawa jakości obrazów nieostrych, poruszonych lub o niewielkim kontraste,
- usunięcie określonych wad obrazu,
- wzmacnianie pewnych elementów obrazu,
- rekonstrukcja obrazu
- inne: artystyczne/estetyczne

Zastosowania filtrów:

- medycyna – techniki obrazowania, teledetekcja, robotyka – systemy wizyjne, TV

FILTRACJA LINIOWA

Nawigacja

Kokpit

Strona główna

Strony

Moje kursy

MAD - wykład

MAD - ćwiczenia

GU - wykład

Uczestnicy

Oceny

GRAFIKA

UŻYTKOWA -
WYKŁAD

Informacje

Wykład 1

Wykład 2

Wykład 3

Wykład 4
(23.03.2020)

Wykład 5
(30.03.2020)

Wykład 6
(6.04.2020)

Wykład 7
(20.04.2020)

Wykład 8
(27.04.2020)

Lista obecności
27.04.2020 -
kliknij i zaznacz

Przekształceni a kontekstowe - filtracja

Filtracja z
zadaną maską
filtra

Filtracja z
dowolną maską
filtra

Algorytmy
grafiki
rastrowej

Zadanie 10

Zadanie 11

GU - laboratorium
WWW - laboratorium

Obraz może być traktowany jako funkcja dwóch dyskretnych zmiennych wartości punktów w dwóch osiach x i y. W praktyce często stosuje się przekroje obrazu (pionowe/poziome), które wizualizują rozkład jasności (dla obrazu monochromatycznego) bądź nasycenia (RGB).

Funkcje realizujące filtrację cyfrową można opisać przy pomocy splotu (konwolucji) funkcji:

$$y(x) = \int_{-\infty}^{+\infty} f(x-t)h(t)dt = f(t) \cdot h(t)$$

gdzie: f, h – splotane funkcje.

Splot jest zdefiniowany w całym zbiorze liczb rzeczywistych ale dziedzina funkcji h może być ograniczona, wówczas splot z funkcją h można traktować jako filtr.

Operacja filtrowania jest przeprowadzana jako sumowanie jasności pikseli $J(x-i,y-j)$ należących do otoczenia K danego punktu (x,y) z odpowiednimi wagami $w(i,j)$.

$$j^w(x,y) = \sum_{i,j \in K} j(x-i,y-j) \cdot w(i,j)$$

gdzie $w(i,j)$ – filtr, tablica (maska) współczynników, K – otoczenie. Im większy rozmiar otoczenia (K), tym wyraźniejszy efekt działania filtra.

$$\sum_{i,j \in K} w(i,j) = 1 \quad \text{jasność obrazu nie ulega zmianie}$$

$$\sum_{i,j \in K} w(i,j) > 1 \quad \text{rozjaśnienie obrazu}$$

$$\sum_{i,j \in K} w(i,j) < 1 \quad \text{przyciemnienie obrazu}$$

Ze względu na prostotę i szybkość obliczeń zwykle przyjmuje się wagę całkowite a maski definiuje jako macierze n x n najczęściej 3 x 3, 5 x 5.

Jeżeli suma współczynników przekracza 1, konieczne jest zastosowanie normalizacji jasności, aby otrzymane wyniki sprowadzić do przedziału dostępnych poziomów jasności.

$$j^w(x,y) = \frac{\sum_{i,j \in K} j(x-i,y-j) \cdot w(i,j)}{\sum_{i,j \in K} w(i,j)}$$

Jeżeli w filtrze występują współczynniki ujemne, w obrazie wynikowym mogą pojawić się ujemne wartości atrybutu jasności.

Metody stosowane zapobiegawczo:

- obcinanie wartości na poziomie dopuszczalnego minimum/maksimum (arytmetyka przepelnieniowa),
- zastosowanie wartości bezwzględnej do nowej wartości jasności,
- zastosowanie normalizacji

Normalizacja jasności:

$$j^w(x,y) = 2^\beta \frac{j^w(x,y) - j_{min}^w(x,y)}{j_{max}^w(x,y) - j_{min}^w(x,y)}$$

gdzie β – głębia bitowa obrazu

$j_{min}^w(x,y)$ – minimalna wartość jasności w obrazie wynikowym

$j_{max}^w(x,y)$ – maksymalna wartość jasności w obrazie wynikowym

• FILTR UŚREDNIAJĄCY

Filtr tłumii składowe widma.

Zastosowanie:

- wygładzanie obrazu

Wada:

- zmniejszenie ostrości i wyrazistości obrazu.

Stosowanie dużych masek:

- znacznie zmniejsza ostrość obrazu
- generuje wyraźny wzrost obliczeń dlatego stosuje się maski 3x3 lub 5 x 5 i nie większe.

Typowe maski:

$$w(i,j) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad j^w(x,y) = \frac{\sum_{i,j \in K} j(x-i, y-j)}{9}$$



$$w(i,j) = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix} \quad j^w(x,y) = \frac{\sum_{i,j \in K} j(x-i, y-j)}{25}$$

• **FILTR DOLNOPRZEPUSTOWY**

Filtr tłumia składowe widma o wyższej częstotliwości, pozostawiając bez zmian składowe o niższej częstotliwości.

Zastosowanie:

- redukcja szumów i zakłóceń,
- wygładzanie drobnych zawierań krawędzi,
- usuwanie efektów „falowania” jasności obiektów i tła.
- mniejszy efekt rozmycia

Wada:

- zmniejszenie ostrości i wyrazistości obrazu

Typowe maski:

$$w(i,j) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & a & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad w(i,j) = \begin{pmatrix} 1 & b & 1 \\ b & b^2 & b \\ 1 & b & 1 \end{pmatrix}$$

Uwagi:

- a = 0 - obraz wynikowy będzie generowany jedynie w oparciu o otoczenie,
- dla a = 1 (b = 1) - filtr uśredniający,
- dla b > 1 - filtr Gaussa

Dla uproszczenia obliczeń zamiast maski kwadratowej stosowana jest maska w kształcie krzyża (filtr kołowy).

PRZYKŁAD

Maska 1

$$w(i,j) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & a & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

dla $a=1$ lub $a=2$ lub $a=4$

$$w(i,j) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad w(i,j) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad w(i,j) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 4 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Maska 2

$$w(i,j) = \begin{pmatrix} 1 & b & 1 \\ b & b^2 & b \\ 1 & b & 1 \end{pmatrix}$$

dla $b=2$

$$w(i,j) = \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$$



Filtre dolnoprzepustowe stosowane są przede wszystkim do usuwania szumu i zakłóceń, które:

- wyraźnie odróżniają się od tła
- występują w postaci izolowanych punktów na dużych, gładkich powierzchniach.

• FILTR GÓRNOPRZEPUSTOWY

Filtr wzmacnia składowe widma o wyższej, tłumi składowe o niższej częstotliwości.

Zastosowanie:

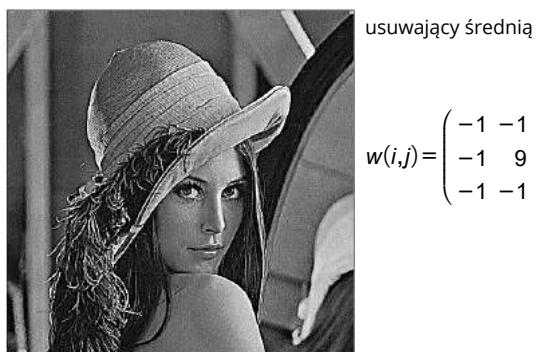
- zwiększenie ostrości obrazu,
- podkreślenie elementów, charakteryzujących się szybką zmianą jasności
- (krawędzie, kontury, kontrastowe tekstury)

Wada:

- wzmacnienie zakłóceń

Typowe maski:

- centralne elementy masek mają wysokie współczynniki
- elementy brzegowe mają ujemne współczynniki



$$w(i,j) = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{pmatrix}$$



$$w(i,j) = \begin{pmatrix} 1 & -2 & 1 \\ -2 & 5 & -2 \\ 1 & -2 & 1 \end{pmatrix}$$

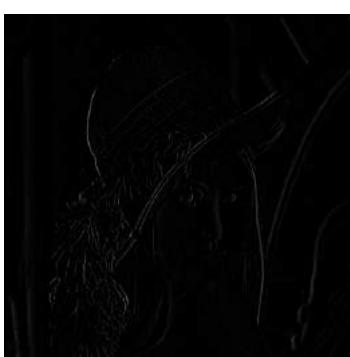
- **FILTR PRZESUWANIA/ODEJMOWANIA**

wykonują kopię, transformację przesunięcia kopii o 1 piksel i odjęcia obrazu od jego kopii

W obszarach, gdzie występują szybkie zmiany jasności, w wyniku operacji pojawia się piksel o wysokiej wartości, w obszarach ciągłotonalnych jasności pikseli są niskie.

Zastosowanie:

- wykrywanie krawędzi pionowych i poziomych



$$w(i,j) = \begin{pmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$



$$w(i,j) = \begin{pmatrix} 0 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$



$$w(i,j) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{pmatrix}$$

- **FILTRY KRAWĘDZIOWE**

- **Operatory Prewitta** - filtry działające w oparciu o analizę pierwszej pochodnej wyprowadzone jako rozszerzenie operatorów Robertsa na okna o wielkości 3 x 3.

Gradient może być analizowany w 8 kierunkach (najwyższa wartość współczynnika wskazuje kierunek).



$$w(i,j) = \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}$$



$$w(i,j) = \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix}$$

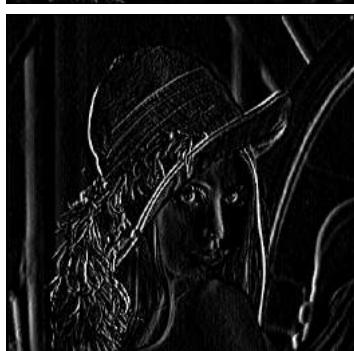
◦ Operatory Sobela

Podobnie, jak filtry Prewitta opierają się na analizie gradientu. Wykorzystują wyższe wartości współczynników. Efekt ich działania jest wyraźniejszy (większa „czułość”), a wykryte krawędzie mają większą grubość. Filtry obarczone są wyższym współczynnikiem błędu.

Maski Sobela mogą być wyznaczane dla 8 kierunków.



$$w(i,j) = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$$



$$w(i,j) = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}$$

◦ Operatory krawędziowe kierunkowe Robinsona

Filtry o maskach:

$$135^\circ \quad w(i,j) = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -2 & -1 \\ 1 & -1 & -1 \end{bmatrix}$$

$$90^\circ \quad w(i,j) = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{bmatrix}$$

$$45^\circ \quad w(i,j) = \begin{bmatrix} 1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & -1 & 1 \end{bmatrix}$$

$$180^\circ \quad w(i,j) = \begin{bmatrix} 1 & 1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & -1 \end{bmatrix}$$

$$0^\circ \quad w(i,j) = \begin{bmatrix} -1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & 1 & 1 \end{bmatrix}$$

$$270^\circ \quad w(i,j) = \begin{bmatrix} 1 & -1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$325^\circ \quad w(i,j) = \begin{bmatrix} -1 & -1 & -1 \\ 1 & -2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$225^\circ \quad w(i,j) = \begin{bmatrix} -1 & -1 & 1 \\ -1 & -2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

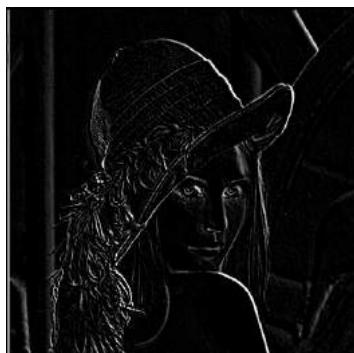
PRZYKŁAD



Maska Robinsona wykrywająca narożniki dla kierunku 180°.

◦ FILTRY GRADIENTOWE KIERUNKOWE

Definiowane przy użyciu kierunków geograficznych wyróżniają krawędzie o kierunku zbliżonym do zadanego tożsame z filtrami Robertsona



$$w(i,j) = \begin{pmatrix} 1 & -1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & 1 \end{pmatrix}$$

południowy zachód

270°

◦ FILTRY UWYPUKLAJĄCE

Definiowane przy użyciu kierunków geograficznych. Ich celem jest nadanie obrazowi wrażenia wypukłości.



$$w(i,j) = \begin{pmatrix} -1 & 0 & 1 \\ -1 & 1 & 1 \\ -1 & 0 & 1 \end{pmatrix}$$

uwypuklający - wschód

◦ FILTRY WYOSTRZAJĄCE – LAPLASJANY

Służą do wykrywania i wydobywania krawędzi niezależnie od kierunku ich przebiegu (mają charakter bezkierunkowy). Budowane w oparciu o sumę pochodnych funkcji gradientu jasności dla kierunków poziomego i pionowego, bądź dla wszystkich kierunków (z uwzględnieniem kierunków ukośnych).



LAP 1

$$w(i,j) = \begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$



LAP 2

$$w(i,j) = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}$$



LAP 3

$$w(i,j) = \begin{pmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{pmatrix}$$

FILTRY STATYSTYCZNE, NIELINIOWE

- **FILTR MEDIANOWY**

wykorzystywana jest mediana, redukcja szumu jest efektem eliminacji pikseli, których jasność odbiega znacznie od średniej wartości pozostałych.

nie prowadzi do rozmywania, ale powoduje utratę części danych, nie wymaga normalizacji obrazu, filtr eliminuje narożniki



- **FILTR MINIMALNY (kompresujący albo erozyjny)**

wybór piksela najmniejszej jasności, wielokrotne stosowanie powoduje zmniejszanie jasności obiektów krawędziowych w obrazie

- **FILTR MAKSYMALNY (dekompresujący lub ekspansywny)**

wybierana jest największa wartość jasności, powoduje zwiększenie jasności elementów krawędziowych, co w efekcie daje wrażenie powiększania się obiektów.



FILTRY NIELINIOWE ADAPTACYJNE

Charakterystyka działania zmienia się zależnie od analizowanego fragmentu obrazu.

Etapy działania adaptacyjnego filtra usuwającego szum:

- wyznaczenie dla każdego punktu wartości parametru, który pozwoli zdecydować czy należy on do krawędzi (np. wartość gradientu),
- zastosowanie filtracji dolnoprzepustowej selektywne jedynie dla pikseli, które nie zostały zakwalifikowane jako przynależne do krawędzi.

Zaletą stosowania tego typu filtrów jest możliwość usunięcia szumu bez utraty ostrości.



Copyright 2020 M&J Latosińskie. All rights reserved.

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie

online M&J

Patryk Obuchowicz



Kokpit > Moje kursy > GU - wykład > Wykład 8 (27.04.2020) > Algorytmy grafiki rastrowej

Algorytmy grafiki rastrowej

Algorytmy rastrowe

- służą do odwzorowania w jak najwierniejszy sposób prymitywów geometrycznych (punkt, odcinek, wielokąt, elipsa) na dyskretnym zbiorze punktów reprezentującym piksele ekranu.
- rysowanie „prymitywów” w grafice rastrowej – przejście od układu współrzędnych rzeczywistych do układu współrzędnych całkowitych.

Pożądane cechy algorytmów:

- minimalny błąd aproksymacji,
- możliwość redukcji artefaktów w postaci postrzępionych krawędzi (aliasing),
- możliwe jak najniższa złożoność czasowa oraz obliczeniowa,
- możliwość efektywnej implementacji sprzętowej

Prymitywy

Punkt

- zadanie polega na narysowaniu na rastrze piksela o określonej barwie

Algorytm

- sprowadza się do podania współrzędnych i koloru piksela.

Odcinek

- zadanie polega na narysowaniu odcinka o zadanych współrzędnych początku i końca,
- zadaniem algorytmu konwersji odcinka jest wyznaczenie pozycji pikseli, które leżą na linii prostej położnej na dwuwymiarowej siatce,
- optymalnie byłoby sekwencja pikseli stanowiących odcinek leżała, jak najbliżej idealnego odcinka, zakładamy odcinki o grubości jednego piksela,
- odcinki są rysowane na siatce reprezentującej bufor ramki, gdzie piksele znajdują się na węzłach siatki.

Algorytm naiwny

- dla wszystkich wartości x (pikseli) od początku do końca odcinka wyznaczamy $y[i]$ dla $x[i]$ z równania prostej (1 mnożenie, 1 dodawanie)
- znajdujemy najbliższą $y[i]$ wartość całkowitą $y_c[i]$ (1 zaokrąglenie)
- zaznaczamy piksel $(x[i], y_c[i])$

Algorytm działa poprawnie dla odcinków o nachyleniu od 0 do 45 stopni.

Dla odcinków o nachyleniu od 45 do 90 stopni należy podążać wzduż osi y .

Przykład:

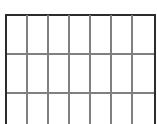
Wykreślić odcinek o końcach w punkcie $(0,0)$ oraz $(5,3)$.

Równanie prostej na której leżą punkty $(0,0)$ oraz $(5,3)$ ma postać $y=0.6 \cdot x$

Położenia pikseli wyznaczone algorytmem naiwnym:

x	0	1	2	3	4	5
y	0	0.6	1.2	1.8	2.4	3
yc	0	1	1	2	2	3

Efekt kreślenia odcinka na rastrze:



Nawigacja

Kokpit

Strona główna

Strony

Moje kursy

MAD - wykład

MAD - ćwiczenia

GU - wykład

Uczestnicy

Oceny

GRAFIKA

UŻYTKOWA - WYKŁAD

Informacje

Wykład 1

Wykład 2

Wykład 3

Wykład 4
(23.03.2020)

Wykład 5
(30.03.2020)

Wykład 6
(6.04.2020)

Wykład 7
(20.04.2020)

Wykład 8
(27.04.2020)

Lista obecności
27.04.2020 - kliknij i zaznacz

Przekształcenia kontekstowe - filtracja

Filtracja z zadaną maską filtra

Filtracja z dowolną maską filtra

Algorytmy grafiki rastrowej

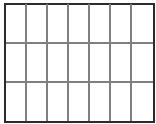
Zadanie 10

Zadanie 11

GU - laboratorium

WWW - laboratorium





Algorytm DDA (ang. Digital Differential Algoritm)

- algorytm naiwny można poprawić korzystając z faktu, że nachylenie odcinka jest stałe
- przy przesuwaniu wartości x o 1, różnica między kolejnymi wartościami y jest stała i wynosi $dy=m$ ($0 < m < 1$)
- kolejne wartości y można obliczyć ze wzoru $y[i+1] = y[i]+dy$

Algorytm oszczędza 1 mnożenie.

Działa poprawnie dla odcinków o nachyleniu od 0 do 45 stopni.

Może być stosowany dla linii w dowolnym oktancie układu współrzędnych, pod warunkiem modyfikacji kierunku i znaku przesunięcia.

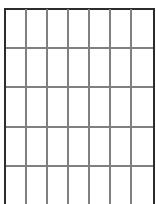
Przykład:

Wyznaczyć współrzędne pikseli do wykreślenia odcinka z punktu (0,0) do (3,5).

Wartość parametru $dy=m=0.6$.

x	0	0+1=1	1+1=2	2+1=3	3+1=4	4+1=5
y	0	0+0.6=0.6	0.6+0.6=1.2	1.2+0.6=1.8	1.8+0.6=2.4	2.4+0.6=3
y_c	0	1	1	2	2	3

Efekt kreślenia odcinka na rastrze:



Algorytm Bresenhama

W 1962 r. Jack Elton Bresenham zaproponował całkowitoliczbowy algorytm rasteryzacji, który z modyfikacjami jest stosowany do dziś. Jest to algorytm przyrostowy bazujący całkowicie na arytmetyce całkowitoliczbowej, znacznie szybszy.

Założenia:

- odcinek ma nachylenie mniejsze niż 45 stopni m należy do przedziału $[0,1]$
- (x_0, y_0) są współrzędnymi dolnego lewego końca odcinka
- (x_1, y_1) są współrzędnymi górnego prawego końca odcinka

Założmy, że został już znaleziony piksel w kolumnie x_i i mamy znaleźć piksel w następnej kolumnie x_{i+1} . Wybór ogranicza się tylko do dwóch pikseli: tego leżącego w tym samym wierszu i tego w wierszu powyżej.

Bresenham zaproponował, żeby kryterium wyboru stanowił znak różnicy odległości ($D_1 - D_2$) pikseli od punktu leżącego na rzeczywistym odcinku. Przy znaku minus będzie to górny piksel, przy znaku plus dolny piksel.

Algorytm:

- 1) Znając współrzędne końców odcinka należy wybrać koniec odcinka o mniejszej współrzędnej x. Wybrany punkt (x_0, y_0) jest pierwszym punktem rysowanego odcinka.
- 2) Obliczyć pomocnicze wielkości: $\Delta x = x_2 - x_1$, $\Delta y = y_2 - y_1$, $a = 2\Delta y$, $b = 2\Delta y - 2\Delta x$ oraz wartość początkową pomocniczego parametru decyzyjnego $p_0 = 2\Delta y - \Delta x$
- 3) Dla kolejnych kolumn o współrzędnych x_k , zaczynając od $k = 0$ należy sprawdzić znak wartości pomocniczego parametru p_k

$p_k < 0$ następny punkt ma współrzędne (x_{k+1}, y_k) i nowa wartość parametru decyzyjnego jest wyliczana jako $p_{k+1} = p_k + a$

$p_k > 0$ następny punkt ma współrzędne (x_{k+1}, y_{k+1}) i nowa wartość parametru decyzyjnego jest wyliczana jako $p_{k+1} = p_k + b$

4) Krok 3 jest powtarzany dopóki nie dojdziemy do końca odcinka.

W każdym kroku, w celu wyznaczenia kolejnego punktu musimy wykonać tylko jedno dodawanie całkowitoliczbowe.



Przykład:

Wyznaczyć współrzędne pikseli do wykreślenia odcinka z punktu (0,0) do (3,5).

1. Wybieramy pierwszy punkt odcinka.

Jest to punkt (0,0).

2. Obliczamy pomocnicze wielkości

$$\Delta x = x_2 - x_1 = 3 - 0 = 3$$

$$\Delta y = y_2 - y_1 = 5 - 0 = 5$$

$$a = 2\Delta y = 2 * 3 = 6$$

$$b = 2\Delta y - 2\Delta x = 6 - 10 = -4$$

$$p_0 = 2\Delta y - \Delta x$$

$$p_0 = 6 - 5 = 1$$

$p_0 > 0$ następny punkt ma współrzędne (1,1), natomiast $p_1 = p_0 + b = 1 - 4 = -3 < 0$

$p_1 < 0$ następny punkt ma współrzędne (2,1), natomiast $p_2 = p_1 + a = -3 + 6 = 3 > 0$

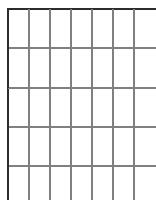
$p_2 > 0$ następny punkt ma współrzędne (3,2), natomiast $p_3 = p_2 + b = 3 - 4 = -1 < 0$

$p_3 < 0$ następny punkt ma współrzędne (4,2), natomiast $p_4 = p_3 + a = -1 + 6 = 5 > 0$

$p_4 > 0$ następny punkt ma współrzędne (5,3)

x	0	1	2	3	4	5
p	1	-3	3	-1	5	
y _c	0	1	1	2	2	3

Efekt kreślenia odcinka na rastrze:


Problemy dotyczące rysowania odcinków:

- w przypadku, gdy punkt przecięcia leży w połowie odległości między pikselami wybór piksela poprzez „zaokrąglenie” jest różny w zależności od kierunku kreślenia odcinka,
- zmiana jasności odcinka w funkcji nachylenia,
- wygląd odcinka narysowanego na rastrze odbiega od idealnego,
- w trakcie zmieniania nachylenia odcinka zmienia się wzór ułożenia pikseli co jest skutkiem skończonej rozdzielczości rastra,
- głównym efektem rastra jest powstawanie „schodkowości” linii innych niż wielokrotność 90° - tzw. aliasing, można wyeliminować ten efekt używając metod antialiasingu (zamiast ostrego rozgraniczenia miedzy kolorem linii/krawędzi i kolorem tła wprowadza się kolory pośrednie pomiędzy tymi dwoma)
- kreślenie linii grubych: wyznaczenie pikseli dla linii pionowych i poziomych jest bezproblemowe, natomiast dla ukośnych wiąże się z niepewnością, które piksele powinny zawierać się w grubej linii, jeśli wybrane zostaną tylko piksele zawierające się w obszarze, nastąpi optyczne „wyszczuplenie” linii, dlatego dodaje się piksele bliskie wybranej grubości linii.

Łuki, okręgi, elipsy

- zadanie polega na narysowaniu krzywej np. do kreślenia okręgu wykorzystuje się równanie okręgu

Algorytm symetrii oktantowej

- bazuje na podziale okręgu na 8 symetrycznych części
- ze wzoru na równanie okręgu obliczamy y oraz L

$$y = \pm \sqrt{R^2 - x^2}$$

$$L = R \frac{\sqrt{2}}{2}$$



Algorytm z punktem środkowym

- jako punkt下一个 po $P_i(x_i, y_i)$ wybieramy piksel powyżej (A) albo poniżej (B)
 - $F(x,y) = x^2 + y^2 - R^2 = 0$
 - $F(x,y) = 0$, dla punktów leżących na okręgu,
 - $F(x,y) > 0$, dla punktów leżących poza okręgiem,
 - $F(x,y) < 0$, dla punktów leżących wewnątrz okręgu.
- zmienność decyzyjna
- dla $d_i < 0$ wybieramy piksel A. Następny punkt środkowy znajdzie się w odległości 1 przyrostu wzdłuż osi OX
- czyli wówczas przyrost wynosi $2x_i + 3$
- dla $d_i \geq 0$ wybieramy piksel B, punkt środkowy znajdzie się w odległości 1 wzdłuż osi OX i OY czyli wówczas przyrost wynosi $2x_i - 2y_i + 5$

Algorytm Bresenham'a

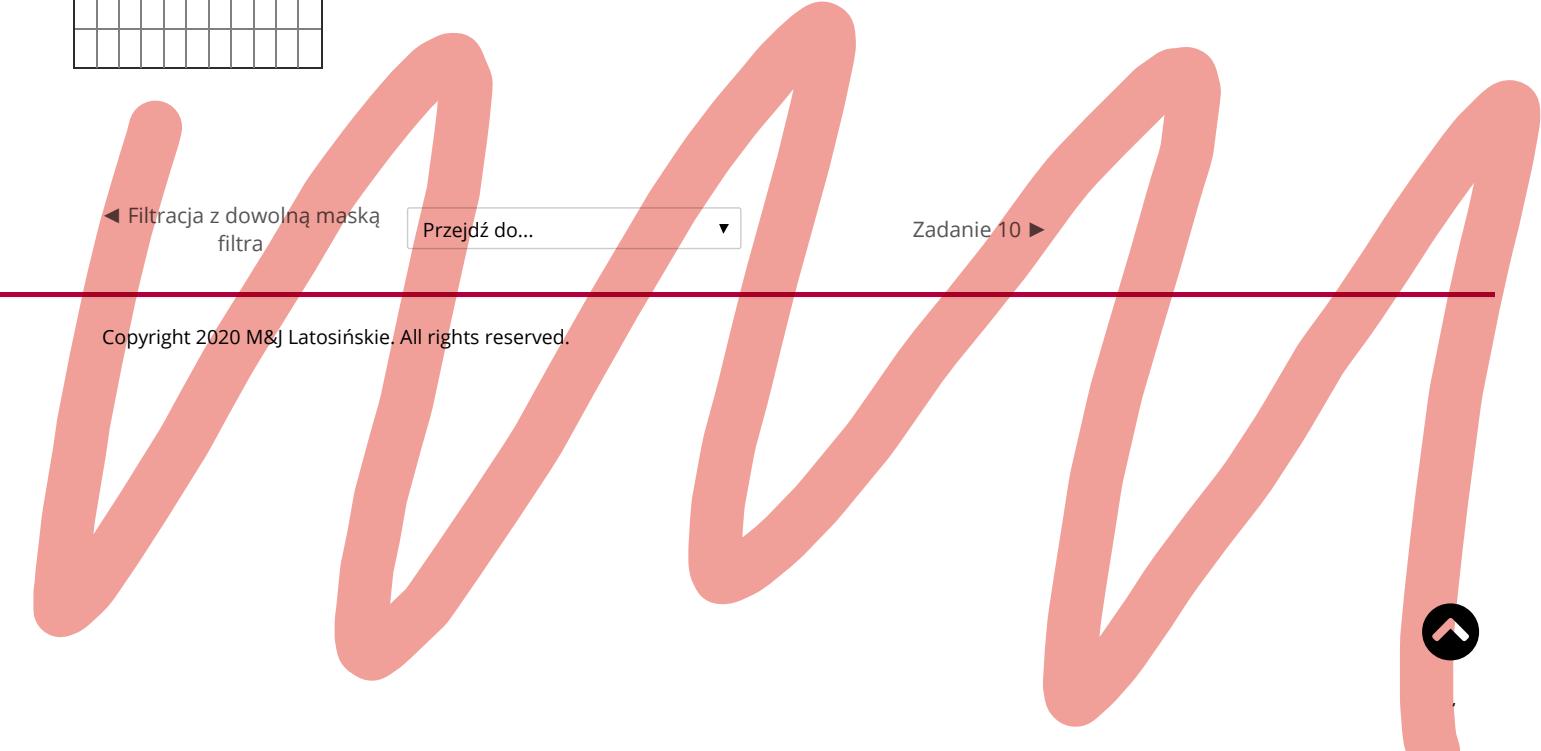
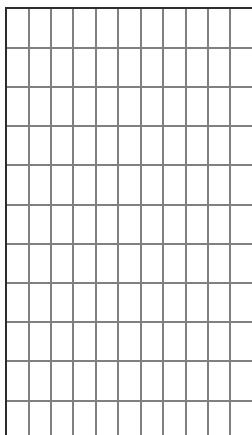
- dla okręgu w 1 oktancie zachodzi zależność analogiczna jak dla odcinka - wartość y kolejnego piksela będzie równa wartości aktualnej lub o 1 mniejszej, a kryterium wyboru stanowi to czy punkt środkowy znajduje się wewnątrz czy na zewnątrz
- korzystając z zasad symetrii można wyznaczyć punkty tylko w 1 oktancie koła
- dla 1 oktantu koła $(0,r)$ oraz par (x,y) takich że $x=y$, kryterium wyboru między pikselami stanowi znak parametru $p[i]$, którego początkowa wartość wynosi $p[0]=5/4 - r$
- gdy $p[i] < 0$ wybierany jest piksel $(x[i+1], y[i])$ a zmienność $p[i+1] = p[i] + 2x[i+1] + 1$
- gdy $p[i] \geq 0$ wybierany jest piksel $(x[i+1], y[i+1])$ a zmienność $p[i+1] = p[i] + 2x[i+1] + 1 - 2y[i+1]$

Przykład:

Narysować okrąg o promieniu 5.

Kroki algorytmu:

1. punkt startowy piksel $(0,5)$ - [zaznaczony na czerwono]
2. $p[0] = 5/4 - 5 = -3,75 < 0$
3. piksel $(1,5)$
4. $p[1] = -3,75 + 2 + 1 = -0,75$
5. $p[1] = -0,75 < 0$
6. piksel $(2, 5)$
7. $p[2] = -0,75 + 4 + 1 = 4,25$
8. $p[2] = 4,25 > 0$
9. piksel $(3,4)$ - [zaznaczony na niebiesko]
10. stop



ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



2

Patryk Obuchowicz -



online M&J

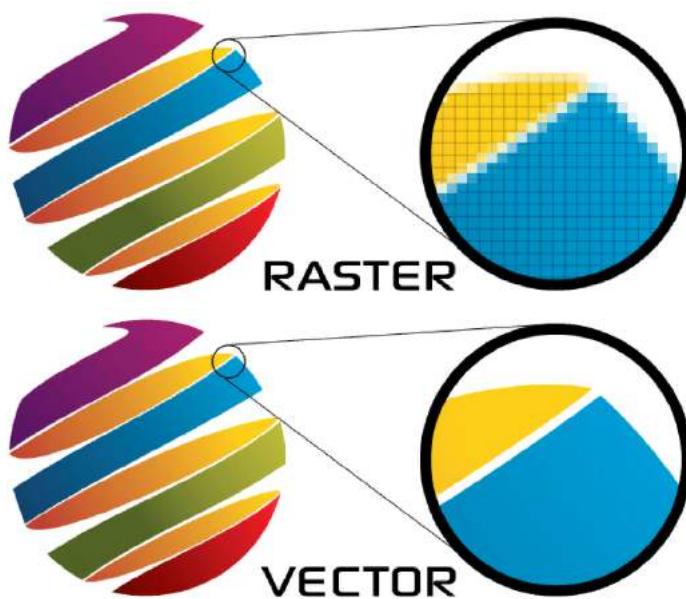
Dashboard > My courses > GU - wykład > Wykład 9 (4.05.2020) > Grafika wektorowa

Grafika wektorowa

Grafika wektorowa

W grafice rastrowej obraz generowany jest jako mapa pikseli na podstawie zestawu danych opisujących ich położenie i kolor. Obszary zastosowań grafiki rastrowej to przede wszystkim zapis zdjęć i realistycznych obrazów (każdy piksel może mieć inną barwę i nasycenie), ich obróbka (retusz, filtracja), przygotowanie ilustracji i obrazów dla aplikacji multimedialnych. Ma ona również zastosowanie w efektach specjalnych w filmach, gdyż nadaje się do opisu prostych animacji, morfingu.

W grafice wektorowej obraz ma postać kodu matematycznego, który następnie jest renderowany na urządzeniu docelowym przez np. system operacyjny, program graficzny czy przeglądarkę internetową. W wielu obszarach zastosowań grafika wektorowa stanowi bardzo dobrą alternatywą dla grafiki rastrowej, a nawet jest od niej pod wieloma względami znacznie lepsza.



Typowe zastosowania grafiki wektorowej



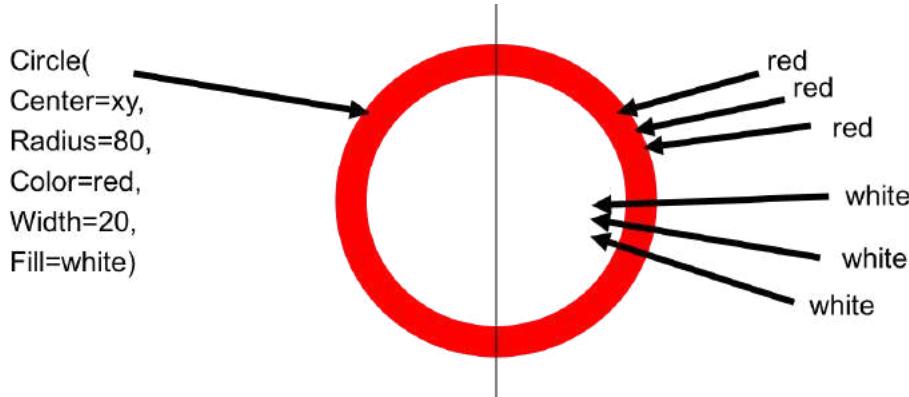
- logotypy, napisy, flagi, godła, herby,
- wykresy matematyczne i statystyczne,
- plany i mapy,
- rysunki techniczne, ilustracje,
- grafika 3D, w tym CAD,
- projekty architektoniczne i inżynierskie,
- modelowanie,
- grafika fraktalna,
- grafika komiksowa.

Podstawowe cechy grafiki wektorowej

- generowana komputerowo,
- opis za pomocą prymitywów,
- każdy element obrazu jest opisany za pomocą pewnej liczby cech (położenie, barwa itp.), których wartości można zmieniać,
- obraz składa się ze stosu elementów ułożonych w odpowiedniej kolejności,
- jest „kreślony” element po elemencie,
- może być 2D lub 3D.

Prymitywy w grafice wektorowej

Grafika wektorowa opiera się na elementarnych obiektach, które nazywane są prymitywami. Prymitywy w grafice rastrowej są kreślone jako punkty na rastrze (opis okręgu po prawej), natomiast wszystkie prymitywy w grafice wektorowej są opisywane za pomocą równań matematycznych (opis okręgu po lewej). Porównanie opisu obiektu w grafice rastrowej (po prawej) i wektorowej (po lewej):



Rodzaje prymitywów w grafice wektorowej

- **Punkty** - współrzędne w układzie kartezjańskim, biegunowym lub innym. Punkt nie ma wymiarów liniowych, dlatego do jego wizualizacji zawsze używa się jakiegoś umownego znacznika (markera).
- **Odcinki linii prostych** - współrzędne punktów: początkowego i końcowego
- **Linie łamane** - współrzędne ciągu wierzchołków
- **Prostokąty** - położenia jednego wierzchołka, szerokość i wysokość
- **Wielokąty** - obszary ograniczone dowolną linią łamaną zamkniętą
- **Linie krzywe gładkie** i z załamaniemi w węzłach - za pomocą punktów węzłowych, przez które linia przechodzi, oraz punktów kontrolnych nie należących do niej, lecz wpływających na jej kształt. Kreślenie linii wymaga równań algebraicznych, których współczynniki zależą od współrzędnych punktów węzłowych i kontrolnych.

Najważniejsze typy linii:

2D:



- **krzywe Hermite'a** - określone poprzez dwa punkty początkowy P1 i końcowy P4 oraz przez dwa wektory styczne w punktach końcowych.
- **krzywe sklejane Béziera 2. lub 3. stopnia** - parametryczne, powszechnie stosowane w programach do projektowania CAD. Każda współrzędna punktu krzywej jest pewną funkcją liczby rzeczywistej, żeby kreślić krzywą na płaszczyźnie potrzebne są dwie funkcje, żeby kreślić krzywą w przestrzeni - trzy, itd.
- **krzywe B-Spline** - sklejane wielomianowe 3. stopnia; powszechnie stosowane, złożone z kawałków gładko połączonych krzywych wielomianowych bądź wymiernych.

3D

- **krzywe wymierne** (krzywa Béziera jest krzywą wielomianową - jej współrzędne opisują wielomiany, natomiast współrzędne krzywej wymiernej są opisywane przez wyrażenia wymierne i kreślone w oparciu o ten sam algorytm)
- **Obszary krzywoliniowe** - obszary ograniczone linią krzywą zamkniętą,
- **Łuki okrągów i elips** oraz wycięte za ich pomocą koła i elipsy - współrzędne środka i promienia oraz zakresy kątów,
- **Sektorzy i odcinki kół oraz elips** - obszary wycięte za pomocą łuku okręgu lub elipsy, odpowiednio zamkniętego,
- **Teksty** - treść informacji przechowywana jest jako ciąg znaków; o jego pozycjonowaniu, rozmiarach i innych właściwościach prezentacyjnych decydują dodatkowe atrybuty,
- **Powierzchnie i bryły różnego typu** - tylko w grafice trójwymiarowej,
- **Inne obiekty** - zależnie od możliwości edytora i formatu rysunku.

◀ Zadanie 11

Jump to...

Krzywe Béziera - wyliczenia numeryczne ►

Navigation

- Dashboard
-  Site home
- Site pages
- My courses
 - MAD - wykład
 - MAD - ćwiczenia
 - GU - wykład
 - Participants
 -  Grades
 - GRAFIKA UŻYTKOWA - WYKŁAD
 - EGZAMIN
 - Informacje
 - Wykład 1
 - Wykład 2
 - Wykład 3



ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



Patryk Obuchowicz -



online M&J

Dashboard > My courses > GU - wykład > Wykład 9 (4.05.2020) > Krzywe Béziera - wyliczenia numeryczne

Krzywe Béziera - wyliczenia numeryczne

Krzywe Béziera

Krzywe Béziera są krzywymi parametrycznymi, wprowadzonymi w latach 60 do opisu karoserii samochodowych, niezależnie przez Pierre Etienne Béziera, pracującego dla Renault oraz Paula de Casteljau, pracującego dla Citroëna. Obecnie krzywe Béziera 2D i 3D stosowane są powszechnie w programach inżynierskich typu CAD, podczas projektowania grafik komputerowych (również wektorowych) a nawet fontów komputerowych.

Każdy punkt krzywej Béziera jest definiowany za pomocą pewnej funkcji liczby rzeczywistej zwanej parametrem, a każda krzywa jest definiowana za pomocą łamanej kontrolnej.

Krzywa Béziera definiowana jest przy pomocy $n + 1$ punktów kontrolnych P_0, P_1, \dots, P_n tworzących tzw. łamankę kontrolną. Kształt krzywej opisywany jest przez wielomiany (Bernsteina) zapisane w postaci rekurencyjnej $B_i^n(t) = (1-t)B_i^{n-1}(t) + tB_{i-1}^{n-1}(t)$ gdzie $t \in [0,1]$ jest parametrem. Stopień wielomianów zależy od liczby punktów kontrolnych. Do wyznaczania punktów krzywej Béziera wykorzystywany jest algorytm de Casteljau.

Algorytm de Casteljau

Jest to algorytm rekurencyjny za pomocą którego na podstawie wierzchołków łamanej znajdowany jest zbiór punktów wyznaczających krzywą Béziera. Punkty początkowy i końcowy łamanej wyznaczają początek i koniec krzywej Béziera. Liczba punktów na krzywej jest zależna od gęstości podziału przedziału $<0,1>$.

Kroki algorytmu:

1. dana jest dowolna łamana zdefiniowana przez n wierzchołków oraz parametr (liczba) t ,
2. każdy odcinek łamanej jest dzielony w stosunku $t:(1-t)$ wyznaczając n wierzchołków, które definiują nową łamankę,

3. proces powtarzany jest do chwili, aż zostanie jeden punkt, co wymaga wykonania n kroków,
4. zmieniając parametr t otrzymujemy ciąg punktów (indeks oznacza krok algorytmu), które wyznaczają krzywą Béziera.

Przykład

Wyznaczyć współrzędne punktów krzywej Béziera trzeciego stopnia zbudowanej w oparciu o cztery punkty P_0, P_1, P_2, P_3 .

Idea algorytmu:

1. dana jest łamana zdefiniowana przez punkty P_0, P_1, P_2, P_3 punkty P_0 ($t=0$) oraz P_3 ($t=1$) są punktami stałymi, wyznaczają początek i koniec krzywej Béziera, pozostałe dwa punkty P_1, P_2 to punkty kontrolne.
2. wyznaczamy trzy punkty, każdy na jednym z trzech odcinków łamanej:

- $P_{01} = (1-t)P_0 + tP_1$
- $P_{12} = (1-t)P_1 + tP_2$
- $P_{23} = (1-t)P_2 + tP_3$

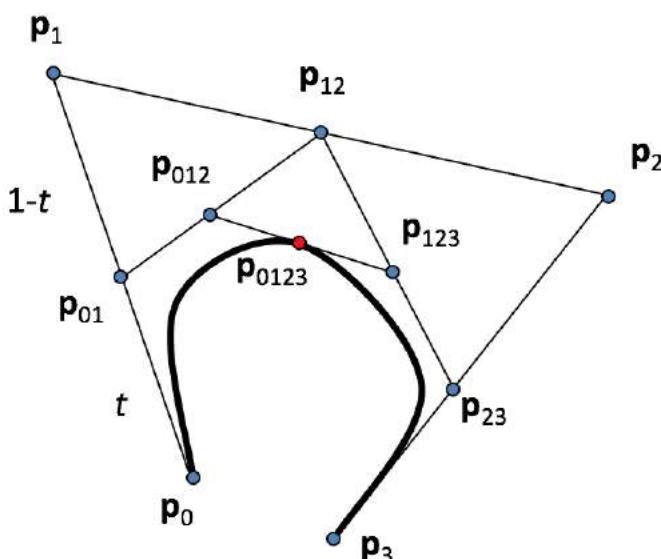
które definiują nową łamankę $P_0, P_{01}, P_{02}, P_{03}, P_3$

3. powtarzamy procedurę - wyznaczamy kolejne punkty:

- $P_{012} = (1-t)P_{01} + tP_{12}$
- $P_{123} = (1-t)P_{12} + tP_{23}$
- $P_{0123} = (1-t)P_{012} + tP_{123}$

Punkt podziału P_{0123} wraz z P_0 oraz P_3 definiuje krzywą Béziera wyznaczając dwa segmenty:

- segment $P_0P_{0123}: P_0, P_{01}, P_{012}, P_{0123}$
- segment $P_{0123}P_3: P_{0123}, P_{123}, P_{23}, P_3$



Wyliczenie numeryczne

Położenia punktów można wyliczyć wykorzystując równania macierzowe:

$$x(t) = \begin{bmatrix} x_0 & x_1 & x_2 & x_3 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} t^3 \\ t^2 \\ t \\ 1 \end{bmatrix}$$

$$y(t) = \begin{bmatrix} y_0 & y_1 & y_2 & y_3 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} t^3 \\ t^2 \\ t \\ 1 \end{bmatrix}$$

gdzie x_i, y_i współrzędne punktów odpowiednio $P_0(x_0, y_0), P_1(x_1, y_1), P_2(x_2, y_2), P_3(x_3, y_3)$, $x(t), y(t)$ współrzędne nowych punktów a t parametr $0 \leq t \leq 1$.

Przykład

- Dane są punkty: początkowy $P_0(4, -2)$, punkty kontrolne $P_1(2, 5)$ oraz $P_2(8, 8)$, punkt końcowy $P_3(11, 6)$ a parametr t zmienia się z krokiem 0.25, co oznacza 3 kroki rekurencyjne (3 nowe punkty).

Współrzędne punktów:

dla $t=0$ punkt początkowy $P_0(4, -2)$

dla $t=0.25$ współrzędne punktu

$$x(t) = \begin{bmatrix} 4 & 2 & 8 & 11 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.25^3 \\ 0.25^2 \\ 0.25 \\ 1 \end{bmatrix} = 3.828$$

$$y(t) = \begin{bmatrix} -2 & 5 & 8 & 6 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.25^3 \\ 0.25^2 \\ 0.25 \\ 1 \end{bmatrix} = 2.484$$

dla $t=0.5$ współrzędne punktu

$$x(t) = \begin{bmatrix} 4 & 2 & 8 & 11 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.5^3 \\ 0.5^2 \\ 0.5 \\ 1 \end{bmatrix} = 5.625$$

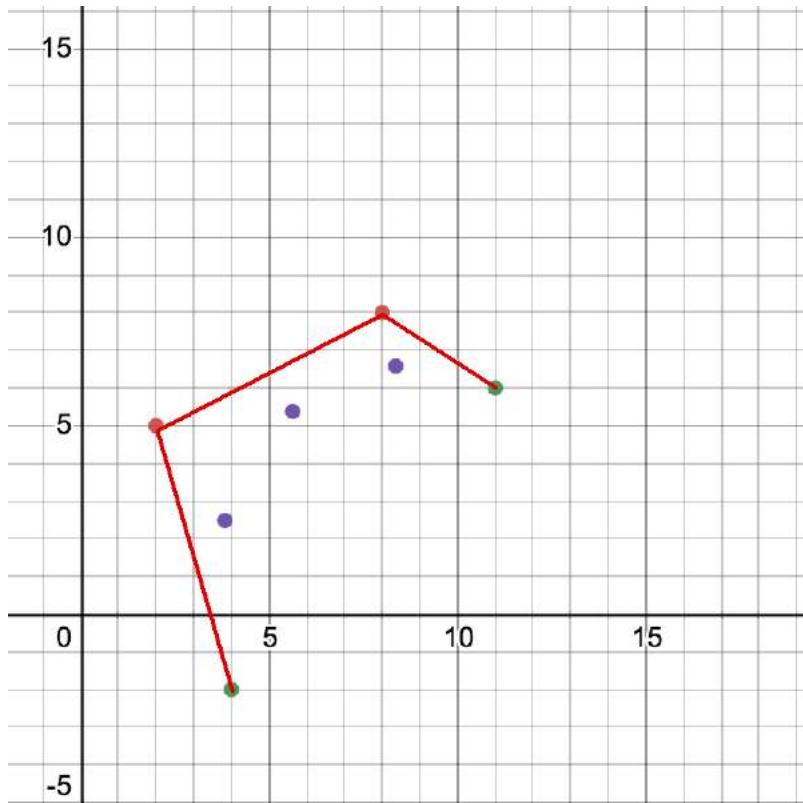
$$y(t) = \begin{bmatrix} -2 & 5 & 8 & 6 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.5^3 \\ 0.5^2 \\ 0.5 \\ 1 \end{bmatrix} = 5.375$$

dla $t=0.75$ współrzędne punktu

$$x(t) = \begin{bmatrix} 4 & 2 & 8 & 11 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.75^3 \\ 0.75^2 \\ 0.75 \\ 1 \end{bmatrix} = 8.359$$

$$y(t) = \begin{bmatrix} -2 & 5 & 8 & 6 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.75^3 \\ 0.75^2 \\ 0.75 \\ 1 \end{bmatrix} = 6.578$$

dla t=1 punkt końcowy $P_3(11,6)$



- Dane są punkty: początkowy $P_0(4, -2)$, punkty kontrolne $P_1(2, 5)$ oraz $P_2(8, 8)$, punkt końcowy $P_3(11, 6)$ a parametr t zmienia się z krokiem 0.1, co oznacza 9 kroków rekurencyjnych.

dla t=0 punkt początkowy $P_0(4, -2)$

dla t=0.1 współrzędne x(t) oraz y(t) wyznaczamy za pomocą wzorów:

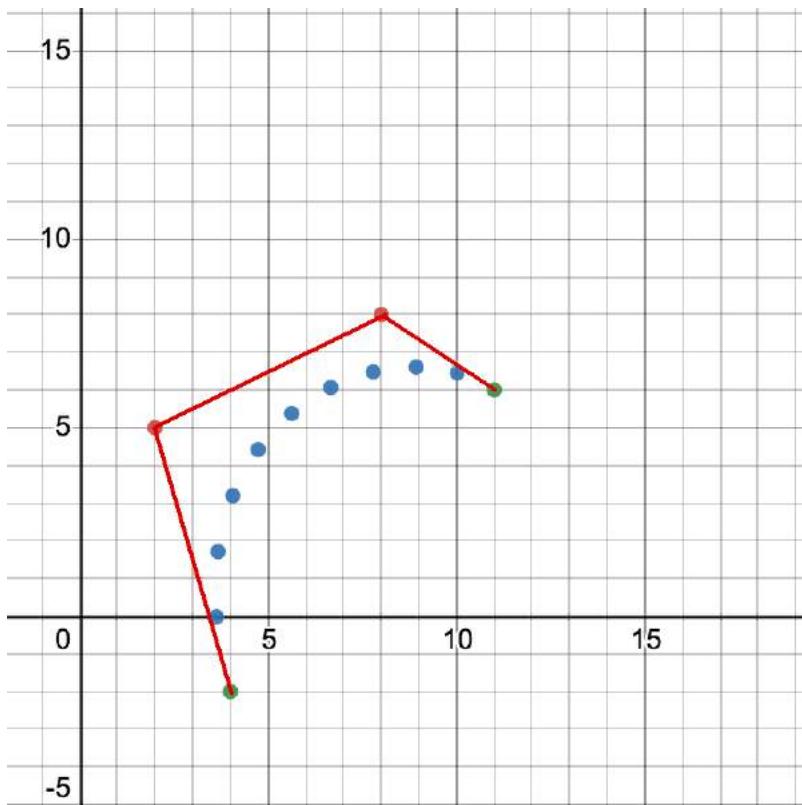
$$x(t) = \begin{bmatrix} 4 & 2 & 8 & 11 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.1^3 \\ 0.1^2 \\ 0.1 \\ 1 \end{bmatrix} = 3.621$$

$$y(t) = \begin{bmatrix} -2 & 5 & 8 & 6 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.1^3 \\ 0.1^2 \\ 0.1 \\ 1 \end{bmatrix} = -0.021$$

w kolejnych krokach podstawiamy do powyższych wzorów $t=0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8$ i 0.9 i znajdujemy pozostałe punkty,

dla $t=1$ punkt końcowy $P_3(11,6)$

Uzyskujemy ciąg punktów wyznaczający krzywą Béziera przedstawiony na rysunku za pomocą niebieskich kropek.



Alternatywny sposób wyznaczenia krzywej Béziera

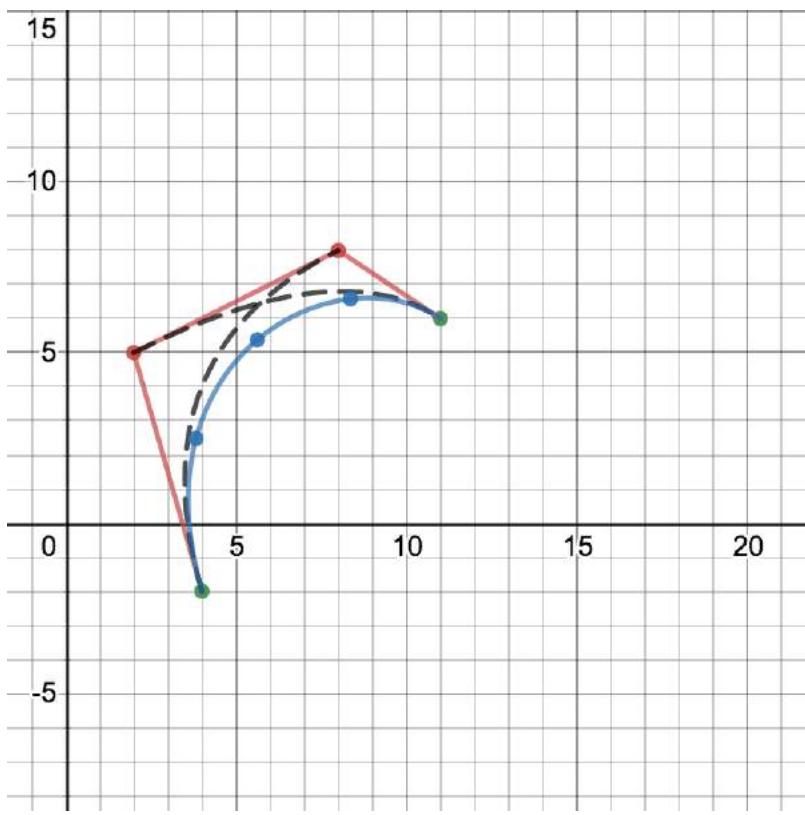
Położenie krzywej można wyliczyć wykorzystując równanie parametryczne

$$((1-t)((1-t)(x_0 + tx_1) + t((1-t)x_1 + tx_2)) + t((1-t)((1-t)x_1 + tx_2) + t((1-t)x_2 + tx_3)), (1-t)((1-t)(y_0 + ty_1) + t((1-t)y_1 + ty_2)) + t((1-t)((1-t)y_1 + ty_2) + t((1-t)y_2 + ty_3))) \quad 0 \leq t \leq 1$$

Podstawiając współrzędne punktów $P_0(4,-2)$, $P_1(2,5)$, $P_2(8,8)$ oraz $P_3(11,6)$ otrzymujemy

$$((1-t)((1-t)(4(1-t)+2t) + t(2(1-t)+8t)) + t((1-t)(2(1-t)+8t) + t(8(1-t)+11t)), (1-t)((1-t)(-2(1-t)+5t) + t(5(1-t)+8t)) + t((1-t)(5(1-t)+8t) + t(8(1-t)+6t))) \quad 0 \leq t \leq 1$$

Krzywa Béziera została przedstawiona niebieską linią ciągłą na poniższym wykresie. Niebieskie punkty na tej krzywej ilustrują identyczność obu wariantów rozwiązań.



Przykład

Wyznaczyć współrzędne punktów krzywej Béziera drugiego stopnia zbudowanej w oparciu o trzy punkty P_0, P_1, P_2 .

Idea algorytmu:

1. dana jest łamana zdefiniowana przez punkty P_0, P_1, P_2 punkty P_0 ($t=0$) oraz P_2 ($t=1$) są punktami stałymi, wyznaczają początek i koniec krzywej Béziera, trzeci punkt P_1 , to punkt kontrolny.
2. wyznaczamy dwa punkty, każdy na jednym z dwóch odcinków łamanej:

- $P_{01} = (1-t)P_0 + tP_1$
- $P_{12} = (1-t)P_1 + tP_2$

które definiują nową łamankę.

3. punkt podziału P_{012} wraz z P_0 oraz P_2 definiuje krzywą Béziera

Wyliczenie numeryczne

Położenia punktów można wyliczyć wykorzystując równania macierzowe:

$$x(t) = \begin{bmatrix} x_0 & x_1 & x_2 \end{bmatrix} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} t^2 \\ t \\ 1 \end{bmatrix}$$

$$y(t) = \begin{bmatrix} y_0 & y_1 & y_2 \end{bmatrix} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} t^2 \\ t \\ 1 \end{bmatrix}$$

- Dane są punkty: początkowy $P_0(4, -2)$, punkt kontrolny $P_1(2, 5)$ oraz punkt końcowy $P_3(11, 6)$ a parametr t zmienia się z krokiem 0.25.

dla $t=0$ punkt początkowy $P_0(4, -2)$

dla $t=0.25$ współrzędne punktu

$$x(t) = \begin{bmatrix} 4 & 2 & 11 \end{bmatrix} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.25^2 \\ 0.25 \\ 1 \end{bmatrix} = 3.688$$

$$y(t) = \begin{bmatrix} -2 & 5 & 6 \end{bmatrix} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.25^2 \\ 0.25 \\ 1 \end{bmatrix} = 1.125$$

dla t=0.5 współrzędne punktu

$$x(t) = \begin{bmatrix} 4 & 2 & 11 \end{bmatrix} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.5^2 \\ 0.5 \\ 1 \end{bmatrix} = 4.75$$

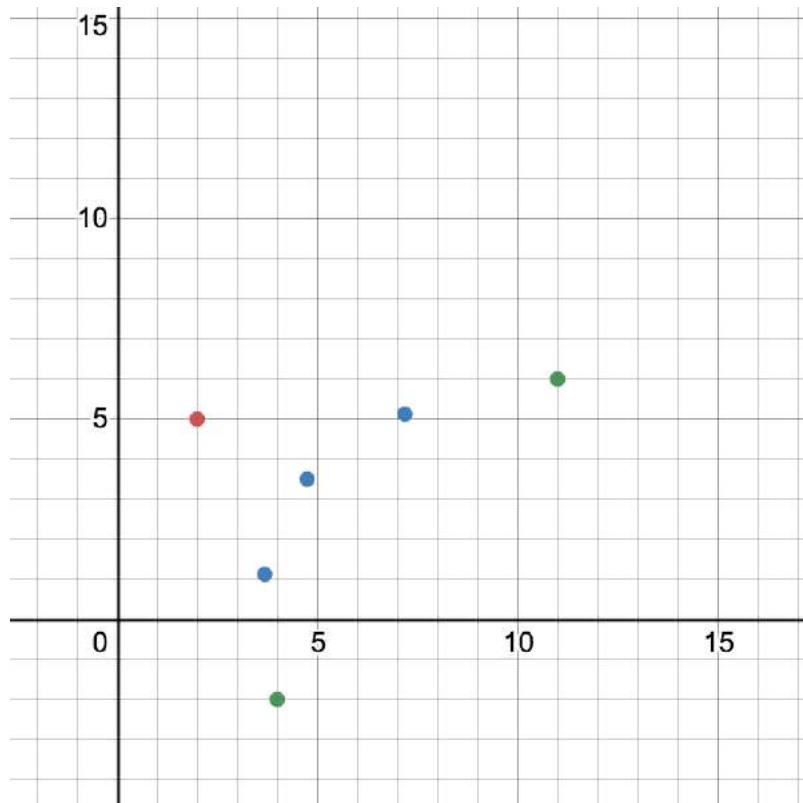
$$y(t) = \begin{bmatrix} -2 & 5 & 6 \end{bmatrix} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.5^2 \\ 0.5 \\ 1 \end{bmatrix} = 3.5$$

dla t=0.75

$$x(t) = \begin{bmatrix} 4 & 2 & 11 \end{bmatrix} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.75^2 \\ 0.75 \\ 1 \end{bmatrix} = 7.188$$

$$y(t) = \begin{bmatrix} -2 & 5 & 6 \end{bmatrix} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.75^2 \\ 0.75 \\ 1 \end{bmatrix} = 5.125$$

dla t=1 punkt końcowy P₂(11,6)



Alternatywny sposób wyznaczenia krzywej Béziera

Położenie krzywej można wyliczyć wykorzystując równanie parametryczne

$$((1-t)((1-t)x_0 + tx_1) + t((1-t)x_1 + tx_2), (1-t)((1-t)y_0 + ty_1) + t((1-t)y_1 + ty_2))$$

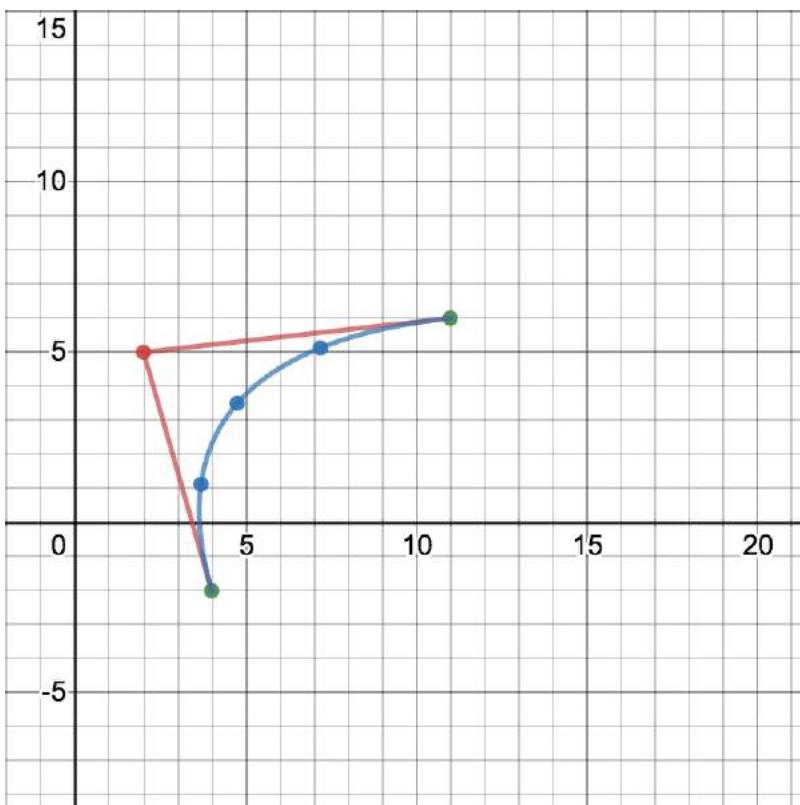
$$0 \leq t \leq 1$$

Podstawiając współrzędne punktów $P_0(4, -2)$, $P_1(2, 5)$ oraz $P_3(11, 6)$ otrzymujemy:

$$((1-t)(4(1-t)+5t)+t(5(1-t)+11t), (1-t)(-2(1-t)+5t)+t(5(1-t)+6t))$$

$$0 \leq t \leq 1$$

Krzywa Béziera została przedstawiona niebieską linią ciągłą na poniższym wykresie. Niebieskie punkty na tej krzywej ilustrują identyczność obu wariantów rozwiązania.



◀ Grafika wektorowa

Jump to...

Krzywe Béziera - wizualizacja ►

Navigation

Dashboard

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



2

Patryk Obuchowicz -



online M&J

Dashboard > My courses > GU - wykład > Wykład 9 (4.05.2020) > Krzywe Béziera - wizualizacja

Krzywe Béziera - wizualizacja

Krzywe Béziera

Krzywe Béziera trzeciego stopnia (sześciennne)

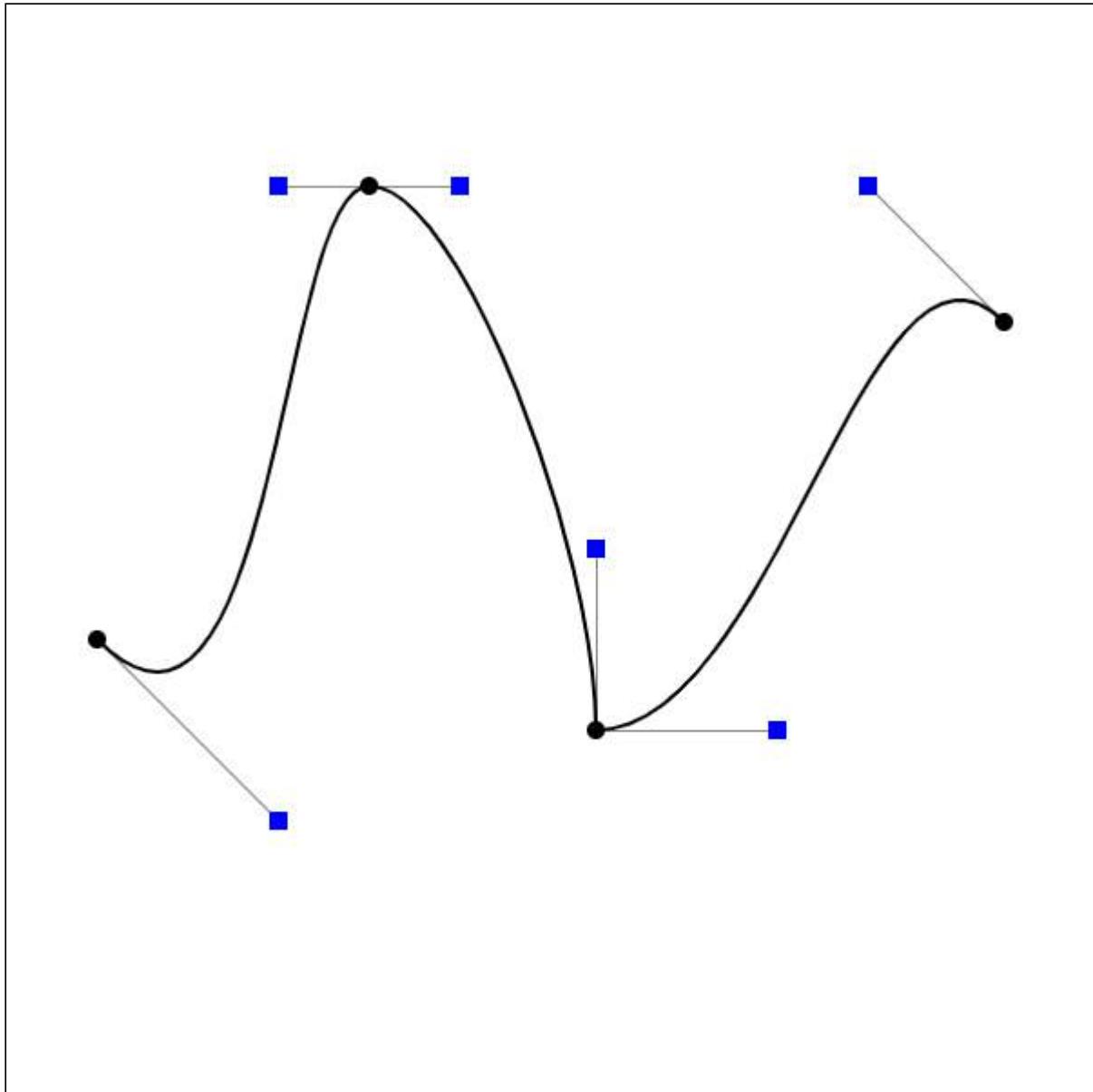
Krzywa Béziera trzeciego stopnia zbudowana jest w oparciu o dwa punkty końcowe i dwa kontrolne. Każdy punkt kontrolny definiuje styczną w odpowiednim punkcie końcowym. Łącznie cztery punkty P_0 , P_1 , P_2 oraz P_3 definiują krzywą Béziera trzeciego stopnia. Niech $B_{P_i, P_j, P_k}(t)$ oznacza krzywą Béziera drugiego stopnia zdefiniowaną przez punkty P_i, P_j, P_k wtedy krzywą Béziera trzeciego stopnia można zapisać w postaci liniowej kombinacji dwóch krzywych Béziera drugiego stopnia: $B(t) = (1 - t)B_{P_0, P_1, P_2}(t) + B_{P_1, P_2, P_3}(t)$, gdzie $t \in [0, 1]$. Po przekształceniu funkcja $B(t)$ przyjmuje postać: $B(t) = (1 - t)^3 P_0 + 3(1 - t)^2 t P_1 + 3(1 - t)t^2 P_2 + t^3 P_3$, gdzie $t \in [0, 1]$, z której jawnie wynika sześcienna zależność funkcji $B(t)$ od parametru t .

Po włączeniu opcji "zablokuj pary punktów kontrolnych", dwa punkty kontrolne powiązane z danym punktem końcowym przesuwają się razem, generując gładką krzywą.

zablokuj punkty kontrolne

ukryj punkty kontrolne





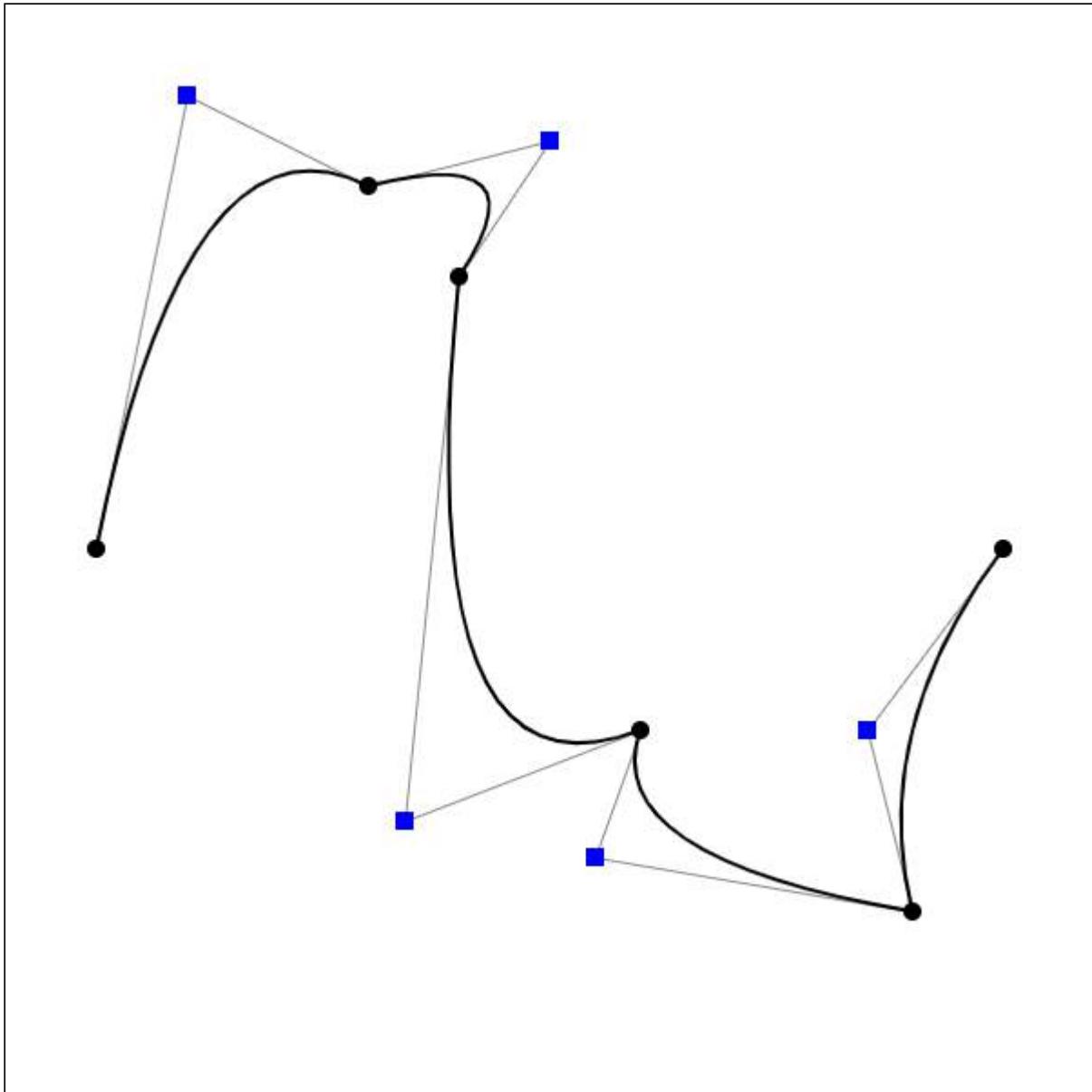
Krzywe Béziera drugiego stopnia (kwadratowe)

Krzywa **Béziera** drugiego stopnia zbudowana jest w oparciu o dwa punkty końcowe i jeden kontrolny. Punkt kontrolny definiuje styczne w obu punktach końcowych. Do wykreślenia krzywej Béziera drugiego stopnia potrzebne są trzy punkty P_0 , P_1 oraz P_2 . Definiuje ją wzór: $B(t) = (1 - t)[(1 - t)P_0 + tP_1] + t[(1 - t)P_1 + tP_2]$, gdzie $t \in [0,1]$.

Po przekształceniu funkcja $B(t)$ przyjmuje postać: $B(t) = (1 - t)^2P_0 + 2(1 - t)tP_1 + t^2P_2$, gdzie $t \in [0,1]$, z której jawnie wynika kwadratowa zależność funkcji $B(t)$ od parametru t .

ukryj punkty kontrolne



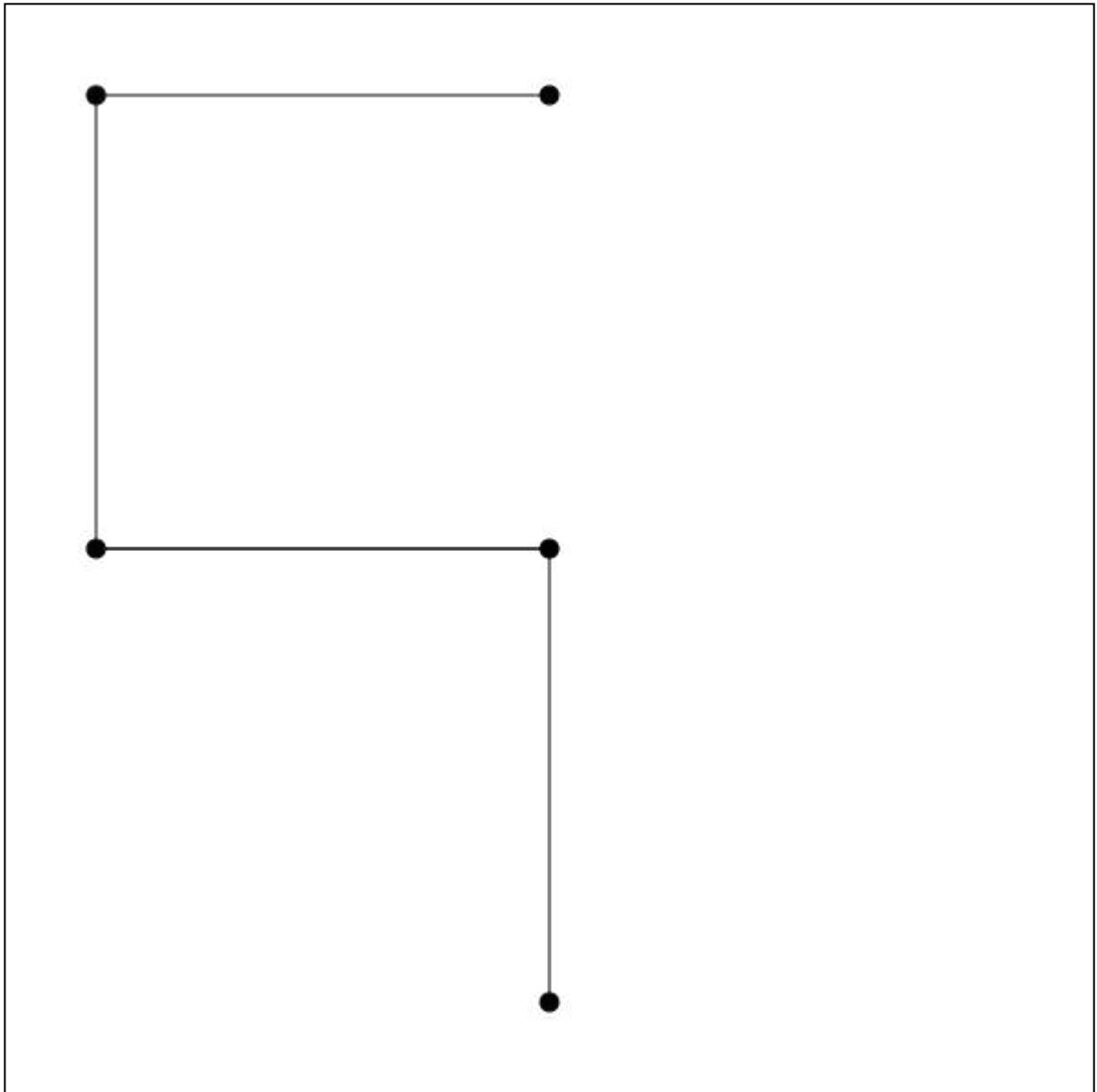


Krzywe Béziera pierwszego stopnia (liniowe)

Dwa punkty P_0 oraz P_1 pozwalają wyznaczyć odcinek, który jest formalnie krzywą Béziera pierwszego stopnia.

Funkcja $B(t)$ ma postać: $B(t) = P_0 + t(P_1 - P_0) = (1 - t)P_0 + tP_1$, gdzie $t \in [0,1]$. Jest to funkcja liniowa parametru t .





◀ Krzywe Béziera - wyliczenia numeryczne

Jump to...

Grafika wektorowa cd ►

Navigation

 Dashboard

 Site home

Site pages

My courses

MAD - wykład

MAD - ćwiczenia

GU - wykład

Participants

 Grades



ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



2

Patryk Obuchowicz -



online M&J

Dashboard > My courses > GU - wykład > Wykład 9 (4.05.2020) > Grafika wektorowa cd

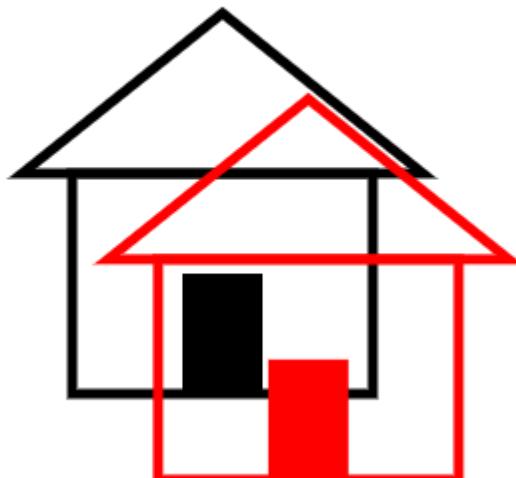
Grafika wektorowa cd

Obiekty złożone w grafice wektorowej

Tworzenie złożonych obiektów w grafice wektorowej wymaga posłużenia się operacjami:

- **teoriomnogościowymi**
 - suma,
 - różnica,
 - dopełnienie,
 - część wspólna,
 - różnica symetryczna.
- **geometrycznymi** (w szczególności operacje opisywane metodami algebry liniowej - transformacje afoniczne):
 - wizualnie (np. przez wskazanie kąta obrotu),
 - numerycznie (np. przez podanie macierzy przekształcenia).

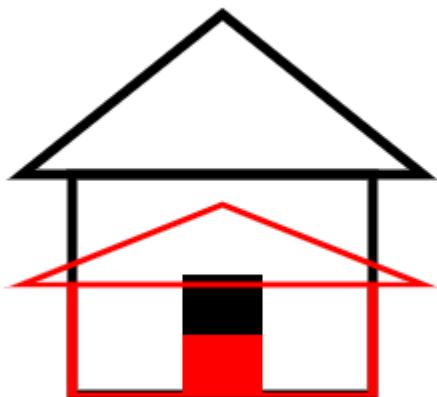
Translacja



Rotacja



Skalowanie



Operacje na obiektach

- **składanie przekształceń** (w ustalonej kolejności),
- wybrane **przekształcenia nieliniowe** np. rzutowanie perspektywiczne,
- **nakładanie obiektów** dwuwymiarowych na dowolne powierzchnie
- **układanie tekstu** wzdłuż linii krzywej,
- **powielanie obiektów** metodami kopiowania i klonowania (kopia jest niezależnym obiektem; klon pozostaje zależny od oryginału i reaguje na jego modyfikacje)
- **grupowanie obiektów** (grupa jest obiektem złożonym traktowanym jako całość, lecz dającym się rozmontować)
- **scalanie obiektów** (efektem scalenia jest pojedynczy obiekt; nie da się go rozbić)
- **przycinanie obiektów** (ustalanie widoczności zależnie od położenia względem innego obiektu)
- **konwersje obiektów** różnych typów - zastąpienie napisu linią obrysowującą znaki fontu
- **przenikanie obiektów** - Podczas prezentacji scen przestrzennych na dwuwymiarowym ekranie obiekty mogą się wzajemnie przesłaniać, w grafikach dwuwymiarowych wzajemne przesłanianie obiektów wynika z kolejności ich rzutowania na płaszczyznę rysunku (tzw. indeks z, czyli z-ordering).

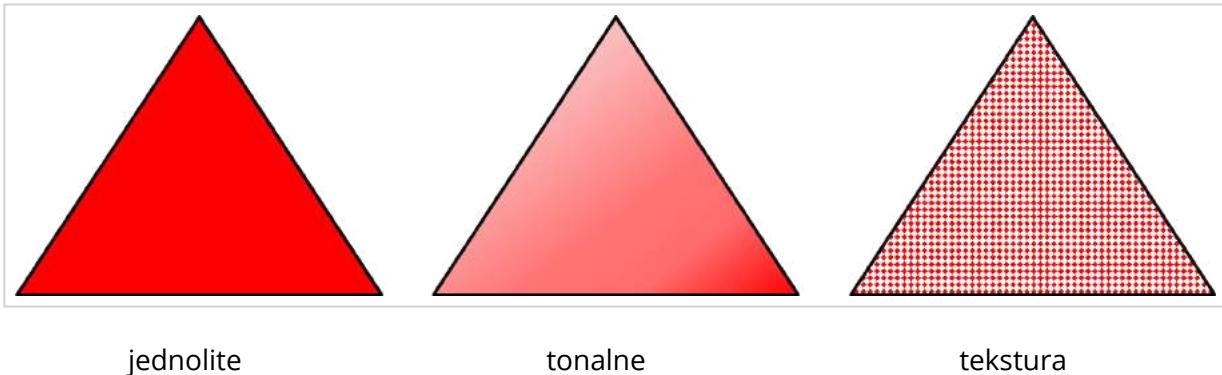
Właściwości obiektów wektorowych

Każdemu obiektovi w grafice wektorowej można nadać lub łatwo zmienić nadane właściwości takie jak:

- kolor i wzorzec linii lub obramowania,
- grubość, sposób kreślenia linii,
- kolor i wzorzec wypełnienia obiektu,
- jednolity kolor, gładkie przejście kolorów, wzorzec rastrowy,
- kolor i postać znaczników punktów węzłowych - mogą np. być oznaczone kółkiem lub strzałką,
- kształt połączeń linii w węzłach ważne w rysunku technicznym,
- właściwości prezentacyjne tekstu - nazwa używanego fontu, odmiana i stopień pisma,
- właściwości niegraficzne np. hipertekstowe.

Wypełnianie obszaru kolorem/teksturą

Polega na nadaniu każdemu pikselowi rastra **wewnętrz konturu** barwy lub odwzorowania tekstury. Możliwe są różne warianty wypełnienia – jednolite, tonalne (gradientowe), tekstura.

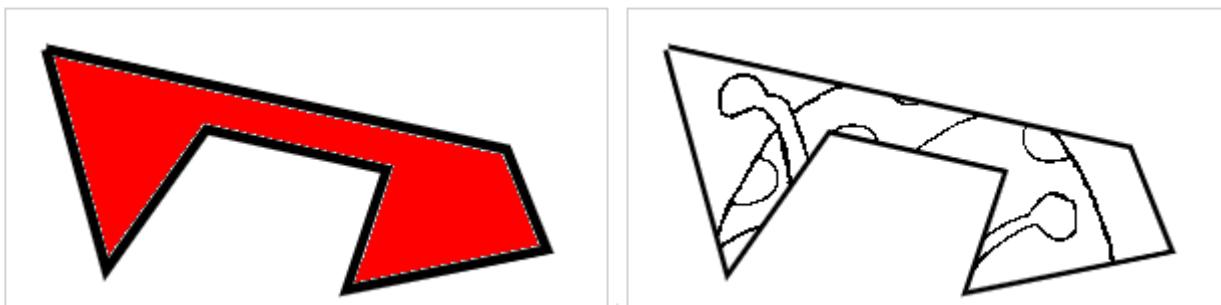


jednolite

tonalne

tekstura

Wypełnianie nieskomplikowanych konturów (np. prostokątnych) jest proste do wykonania, natomiast w przypadku figur o dowolnym kształcie, zarówno wypukłych, jak i wklęsłych, zadanie to może być nietrywialne.



Kontur może być dany jako opis wektorowy wieloboku lub jako opis na płaszczyźnie rastra. Algorytmy wypełniania mają zastosowanie zarówno w odniesieniu do grafiki rastrowej jak i wektorowej.

Algorytmy wypełniania dzielimy na:

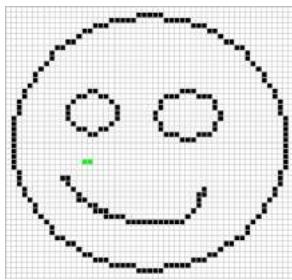
- wypełniania z punktem początkowym,
- wypełniania przez spójność,
- śledzenia konturu,
- kontroli parzystości.

Algorytmy wypełniania z punktem początkowym

- wypełnianie zaczyna się od znanego punktu należącego do wnętrza figury,
- w każdym kroku próbuje się wypełnić piksele sąsiednie w stosunku do rozpatrywanego piksela (cztery sąsiednie w metodzie z czterema sąsiadami),
- kolejność analizowanych pikseli może być różna, z tym, że zawsze rozpatrywany piksel należy do listy aktywnych pikseli - tych, które leżą na obrzeżu już wypełnionego obszaru (rozważa się

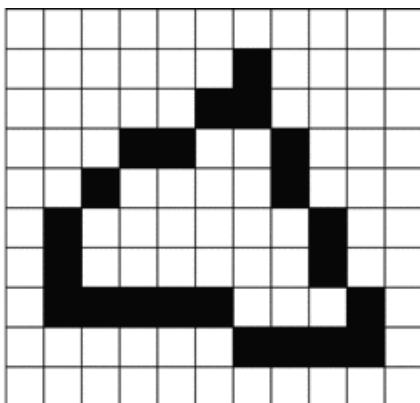
wyłącznie punkty należące do wnętrza figury),

- algorytm może wypełniać wielokąty wypukłe i niewypukłe!

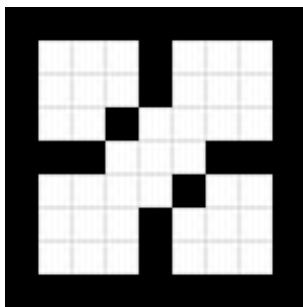


Algorytm wypełniania przez spójność

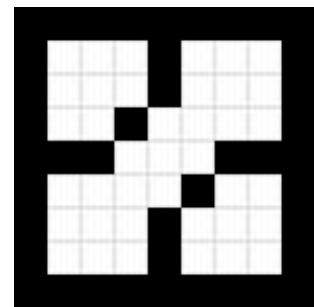
- algorytm używany do wypełniania zamkniętych obszarów bitmapy,
- pozwala również na klasyfikację które piksele należą do którego obszaru (przydatne w pierwszej fazie wektoryzacji lub rozpoznawania tekstu),
- korzysta z kolejki lub stosu,
- najprostszy algorytm wypełniania przez spójność polega na rekurencyjnym przeglądaniu otoczenia punktu startowego – ziarna, przy czym punkt startowy musi należeć do wnętrza wypełnianego obszaru
- parametrami są położenie, kolor początkowy, kolor końcowy piksela,
- wady: kosztowny, barwa piksela badana wielokrotnie.



warianty algorytmu:



4 - punktowy (nie przelewa się przez krawędź)

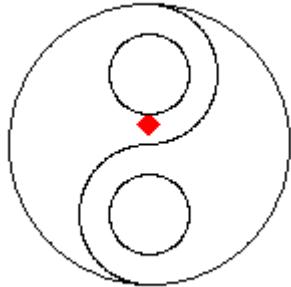


8 - punktowy (przelewa się przez krawędź)

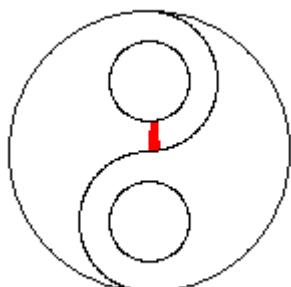
Obszar jest **obszarem 4 - spójnym**, gdy zawarte w nim piksele są tego samego koloru oraz gdy do każdego z nich można dotrzeć przesuwając się tylko w 4 kierunkach - góra, dół, prawo, lewo (4 - punktowy). Obszar jest obszarem **8 - spójnym**, gdy dozwolone są również kierunki leżące na przekątnych (8 - punktowy).

Algorytmy śledzenia konturu (przeglądania wierszami)

- w pierwszej fazie, dla każdego wiersza rastra znajduje się przecięcia z konturami obiektów i ustala się pary punktów przecięcia, między którymi należy dokonać wypełnienia,
- w drugiej fazie ma miejsce wypełnianie odcinków wyznaczonych w pierwszej fazie (zostaną wypełnione odcinki ograniczone przez pary punktów),
- algorytm nie stwarza problemów implementacyjnych w przypadku wielokątów wypukłych, zwłaszcza trójkątów, natomiast trzeba zwracać uwagę na wierzchołki,
- na ogół konieczne jest wyróżnienie dwóch klas wierzchołków takich dla których dwie stykające się w wierzchołku krawędzie leżą po jednej stronie linii rastra oraz takich dla których krawędzie, leżą po obu stronach linii rastra.



wypełnianie rekurencyjne (bez stosu)

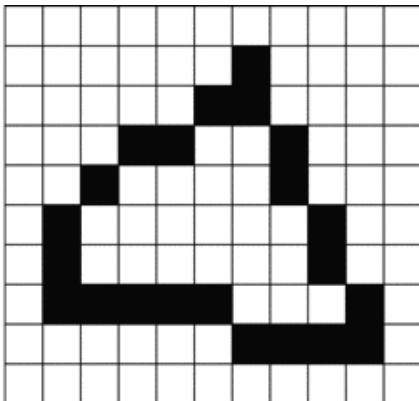


wypełnianie ze stosem

Algorytm XY – z kontrolą parzystości (przeglądania wierszami)

Jego działanie polega na znalezieniu par punktów należących do konturu i leżących na tych samych poziomych (lub pionowych) liniach rastra. Nieparzyste punkty przecięcia odpowiadają wejściu do wnętrza wypełnianego obszaru, a parzyste – wyjściu z niego. Tym samym piksele leżące pomiędzy takimi parami muszą zostać wypełnione barwą. Poszczególne kroki algorytmu:

1. zbudować listę ze wszystkich punktów należących do konturu,
2. posortować listę tak aby element (x_1, y_1) poprzedzał element (x_2, y_2) jeśli $(y_1 > y_2)$ lub jeśli $(y_1 = y_2)$ i $(x_1 < x_2)$,
3. pogrupować elementy w pary tak, aby każda para odpowiadała końcom odcinka leżącego wewnętrz konturu,
4. narysować wszystkie odcinki używając barwy wypełnienia.



Do prawidłowego działania algorytmu należy rozwiązać problem punktów w wierzchołkach wielokąta. Należy dodawać do listy dwukrotnie punkt w wierzchołku jeśli przyległe krawędzie wznoszą się lub opadają niemonotonicznie.

Wybór algorytmu używanego do wypełniania w znacznej mierze zależy od sposobu w jaki definiowany jest kontur obszaru.

W przypadku konturu opisanego rastrowo (poprzez zbiór pikseli o jednakowej barwie) najczęściej stosuje się algorytmy wypełniania przez spójność, natomiast w przypadku wektorowej definicji konturu (bazującej na krzywych zadanych wzorami i parametrami) używa się przede wszystkim algorytmów przeglądania wierszami z kontrolą parzystości.

Formaty graficzne - pliki wektorowe

SVG - (Scalable-Vector-Graphics) - format grafiki dwuwymiarowej, statycznej oraz animowanej, stworzony w 1999 roku z myślą o stronach WWW.

FLA, SWF - technologia tworzenia animacji z wykorzystaniem grafiki wektorowej na zasadzie klatek kluczowych, Adobe FLASH

DXF (data-exchange-format) - najpopularniejszy „format inżynierski” wspierany lub natywny dla aplikacji inżynierskich m.in. AutoCAD opracowany przez Autodesk w 1982 r.

CDR - format wewnętrzny programu CorelDRAW

AI - format wewnętrzny programu Adobe Illustrator

WMF - WMF (Windows Metafile Format) - format graficzny stosowany w systemach Windows, zawierający instrukcje dla systemu operacyjnego, jak wyświetlać grafikę wektorową i rastrową

PostScript - język wektorowego opisu wydruku odpowiedni zwłaszcza dla gotowych dokumentów nie przeznaczonych do dalszej edycji lecz wydruku lub do rozpowszechniania w formie elektronicznej. W ten sposób udostępnia się np. gotowe artykuły, raporty i dokumentację (zawierające np. tekst z elementami grafiki). Jest pokrewny do formatu PDF.

EPS - format, będący podzioborem języka PostScript, którego głównym przeznaczeniem jest przechowywanie pojedynczych stron (ilustracji); dokumenty mogą być wielostronowe, pliki zazwyczaj przechowują wraz z głównym dokumentem miniaturę, która jest wykorzystywana do szybkiego

podglądu zawartości takiego pliku. EPS jest nieformalnym standardem wymiany obrazów wektorowych.

VRML - język grafiki wektorowej, który służy do modelowania i opisu obiektów 3D, nadaje się do opisywania nawet bardzo skomplikowanych układów wielu obiektów a odpowiednie oprogramowanie pozwala na ich wizualizację z różnych miejsc, uwzględniając perspektywę, wzajemne przesłanianie, oświetlenie i ewentualny ruch.

Format SVG

Format SVG jest formatem wektorowym, rekomendowanym przez W3C dla grafiki wektorowej 2D. Został stworzony przez Adobe Systems i stopniowo ewoluował od wersji 1.0 do wersji 3.0:

- SVG 1.1 - rekomendowany przez W3C od 14.01.2003 (różnica SVG 1.1 a SVG 1.0 nieznaczna)
- SVG Tiny 1.2 od 22.12.2008
- SVG 1.1 wersja II, zawierająca poprawki 2011
- SVG 2 integrowana z CSS, HTML5 oraz WOFF
- najnowsza wersja SVG 3.0
- prace nad jego specyfikacją są jawne i nie podlegają żadnym prawom patentowym,
- nie używa algorytmów kompresujących,
- stanowi konkurencję przede wszystkim dla formatu GIF,
- w SVG oprócz standardowych obiektów (prostokąty, elipsy, krzywe) można opisywać efekty specjalne (filtry), maski przezroczystości, tworzyć animacje.

Implementacja:

- SVG bazuje na XML (zawartość to kod podobnie jak w HTML-u),
- tekst zagnieźdzony w pliku *.svg można kopiować tak jak zwykły tekst w dokumencie HTML,
- może być katalogowany i wyszukiwany przez wyszukiwarki,
- pliki można edytować przy wykorzystaniu dowolnego edytora tekstowego,
- pliki *.svg można tworzyć za pomocą zwykłego edytora tekstu, Inkscape, bądź przy pomocy najnowszych wersji pakietów Corel Draw i Adobe Illustrator z odpowiednimi pluginami
- dozwolone są trzy rodzaje obiektów:
 - grafika wektorowa (<PATH>, <RECT>, <CIRCLE>, <ELLIPSE>, <LINE>, <POLYLINE> oraz <POLYGON>)
 - grafika rastrowa (<IMAGE>)
 - tekst (<TEXT>)

Przykłady

- **Kreślenie odcinka:**

```
<svg height="210 " width="500 ">  
  <line x1="0 " y1="0 " x2="50 " y2="30 " style="stroke:rgb(255,0,0);stroke-width:2 " />  
</svg>
```

Osadzanie w kodzie html

```
<!DOCTYPE html>
<html>
<body>
<h1>Red line </h1>

<svg width="100 " height="100 ">
    <line x1="0 " y1="0 " x2="50 " y2="30 " style="stroke:rgb(255,0,0);stroke-width:2 "/>
</svg>

</body>
</html>
```

- **Kreślenie okręgu**

```
<svg width="100 " height="100,>

<circle cx="50" cy="50" r="40" stroke="green" stroke-width="4" fill="yellow" />

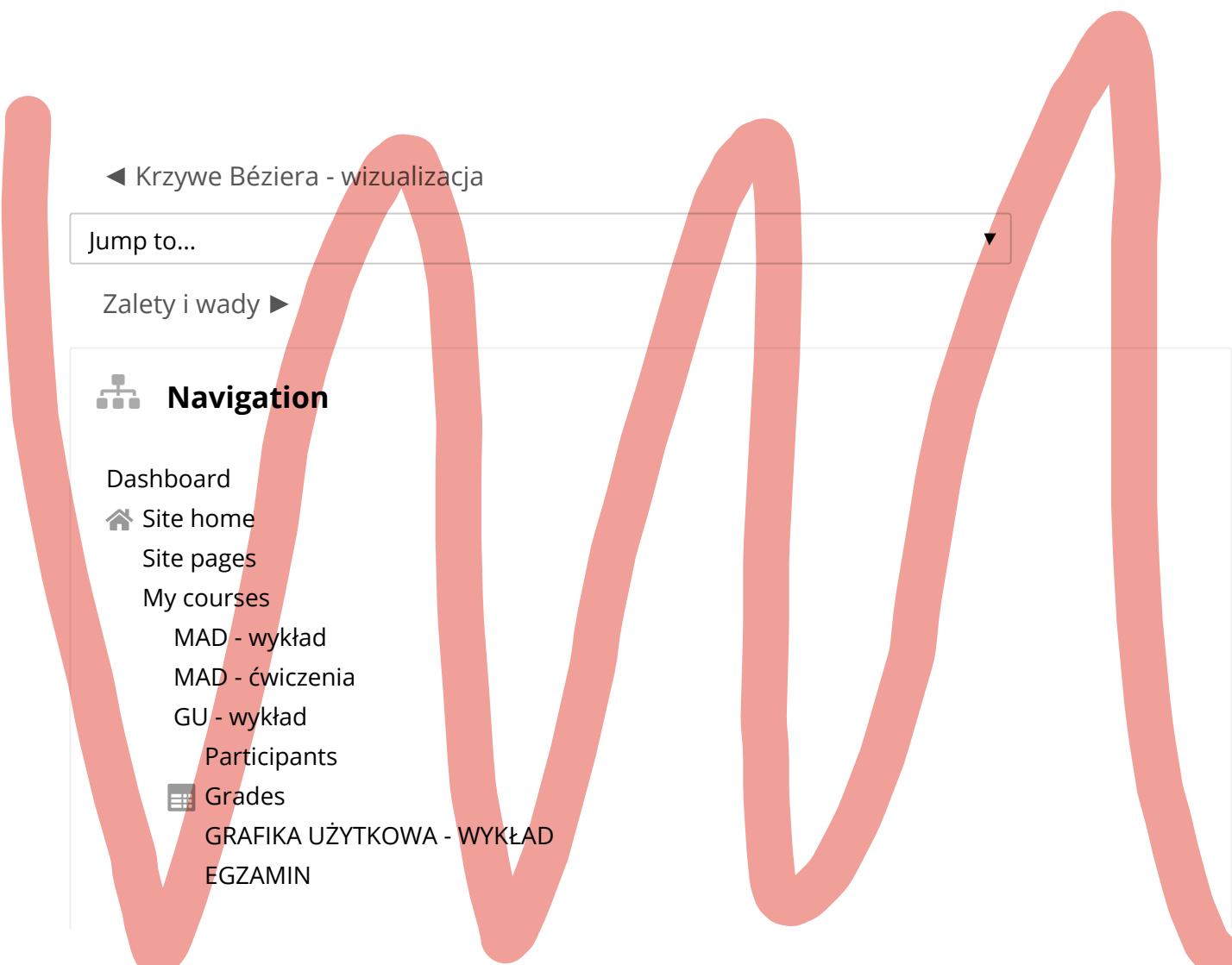
</svg>
```

- **Kreślenie kwadratu**

```
<svg xmlns="http://www.w3.org/2000/svg" version="1.1">

<rect width="150" height="150" fill="rgb(0, 255, 0)" stroke-width="1" stroke="rgb(0, 0, 0)" />

</svg>
```



◀ Krzywe Béziera - wizualizacja

Jump to... ▶

Zalety i wady ►

Navigation

- Dashboard
- Site home
- Site pages
- My courses
 - MAD - wykład
 - MAD - ćwiczenia
 - GU - wykład
 - Participants
- Grades
- GRAFIKA UŻYTKOWA - WYKŁAD
- EGZAMIN

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



Patryk Obuchowicz -



online M&J

Dashboard > My courses > GU - wykład > Wykład 9 (4.05.2020) > Zalety i wady

Zalety i wady

Zalety grafiki wektorowej

- małe pliki, rozmiar obrazu nie wpływa na rozmiar pliku,
- mniejszy rozmiar w przypadku zastosowań niefotorealistycznych (schematy techniczne, loga, flagi i herby, wykresy itp.),
- mniejszy zbiór punktów niż mapa pikselowa,
- skrócenie czasu niektórych obliczeń,
- mniej miejsca w pamięci,
- krótszy czas ewentualnej transmisji obrazu,
- prostota opisu,
- każdy obiekt może być traktowany niezależnie, podczas gdy w przypadku mapy pikselowej mamy do czynienia z całym obrazem - poszczególne obiekty nie mają indywidualnego charakteru,
- potrzebny jest wydajny procesor i algorytm generowania obrazu,
- skalowalność - opis wektorowy jest łatwo skalowalny – np. operację skalowania wykonuje się jedynie w odniesieniu do punktów definiujących obiekty, krawędzie nawet po skalowaniu są gładkie,
- modyfikowalność - parametry obiektów są pamiętane i mogą być modyfikowane (grafika fraktalna),
- transformacje - translacja, skalowanie, rotacja, wypełnianie nie degradują rysunku,
- wymiary podaje się w jednostkach niezależnych od urządzenia (optymalna rasteryzacja przy konwersji na postać rastrową),
- końcowa rasteryzacja będzie wykonana w odniesieniu do już przeskalowanego obiektu, a więc zawsze będzie wykonana z rozdzielcością, z jaką obraz będzie reprodukowany,
- dostępny jest opis przestrzeni trójwymiarowych,
- renderowanie cienia w grafice wektorowej 3D jest bardziej realistyczne, ponieważ cienie mogą być abstrahowane do promieni światła, z których wynikają (renderowanie z realizmem bliskim klasycznej fotografii),
- możliwość użycia ploterów zgodnie z metodą ich pracy,
- bardzo dobre możliwości konwersji do grafiki rastrowej.

Wady grafiki wektorowej

- obraz ma niewiele „detali”,

- przybliżenie wielokątami ujawnia niewłaściwą reprezentację matematyczną,
- problem grubości linii, utrata proporcji,
- wąska paleta barw!
- trudne wypełnianie kolorem obiektów niestandardowych,
- wygładzanie linii (nawet rysowanych „z wolnej ręki”),
- trudne wykorzystywanie w innych dokumentach,
- niemożność dokonania pewnych operacji z powodu wektorowego opisu krzywych,
- konieczność zapisu w postaci grafiki rastrowej do publikacji,
- ogromna złożoność pamięciowa dla obrazów fotorealistycznych,
- przy skomplikowanych obrazach rastrowych nieopłacalność obliczeniowa konwersji do formy wektorowej.

Jedną z największych zalet grafiki wektorowej jest **pełna skalowalność**, co zapewnia brak utraty jakości podczas skalowania i bezproblemową responsywność; jedną z największych wad konieczność zapewnienia wystarczającej mocy obliczeniowej by temu zadaniu podołać.

Grafika wektorowa cd

Jump to... ▶

Rasteryzacja i wektoryzacja ▶

Navigation

- Dashboard
- Site home
- Site pages
- My courses
 - MAD - wykład
 - MAD - ćwiczenia
 - GU - wykład
 - Participants
 - Grades
 - GRAFIKA UŻYTKOWA - WYKŁAD
 - EGZAMIN
 - Informacje
 - Wykład 1
 - Wykład 2
 - Wykład 3
 - Wykład 4 (23.03.2020)
 - Wykład 5 (30.03.2020)
 - Wykład 6 (6.04.2020)
 - Wykład 7 (20.04.2020)
 - Wykład 8 (27.04.2020)
 - Wykład 9 (4.05.2020)

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



Patryk Obuchowicz -



online M&J

Dashboard > My courses > GU - wykład > Wykład 9 (4.05.2020) > Rasteryzacja i wektoryzacja

Rasteryzacja i wektoryzacja

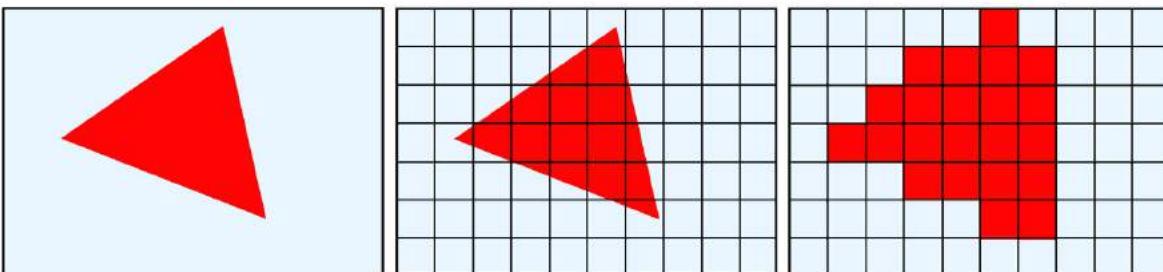
Rasteryzacja – operacja przekształcenia obrazu wektorowego w rastrowy (łatwa)

- jest to konwersja sygnału ciągłego na sygnał dyskretny, która ma na celu możliwie jak najwierniejsze odwzorowanie idealnego prymitywu geometrycznego (punkt, odcinek, figura) na zbiorze punktów o skończonym rozmiarze,
- odwzorowywany prymityw praktycznie nigdy nie przechodzi dokładnie przez węzły siatki, którą wyznaczają środki pikseli, dlatego poszukuje się algorytmów pozwalających jedynie zminimalizować błąd reprezentacji rastrowej.

Idea algorytmu

1. Nałożyć siatkę kwadratów na rysunek
2. Zamalować każdy kwadrat dominującym kolorem
3. Zapamiętać układ punktów barwnych (pikseli)
4. Wyprowadzić - przeliczyć scenę i utworzyć plik wyjściowy - obraz statyczny (wyliczane są odbicia, cienie, załamania światła, mgła, atmosfera)

Przykład rasteryzacji trójkąta:



Zastosowania

- wyświetlenie na urządzeniu wizyjnym, druk lub zapis w pliku w formacie rastrowym/bitmapowym

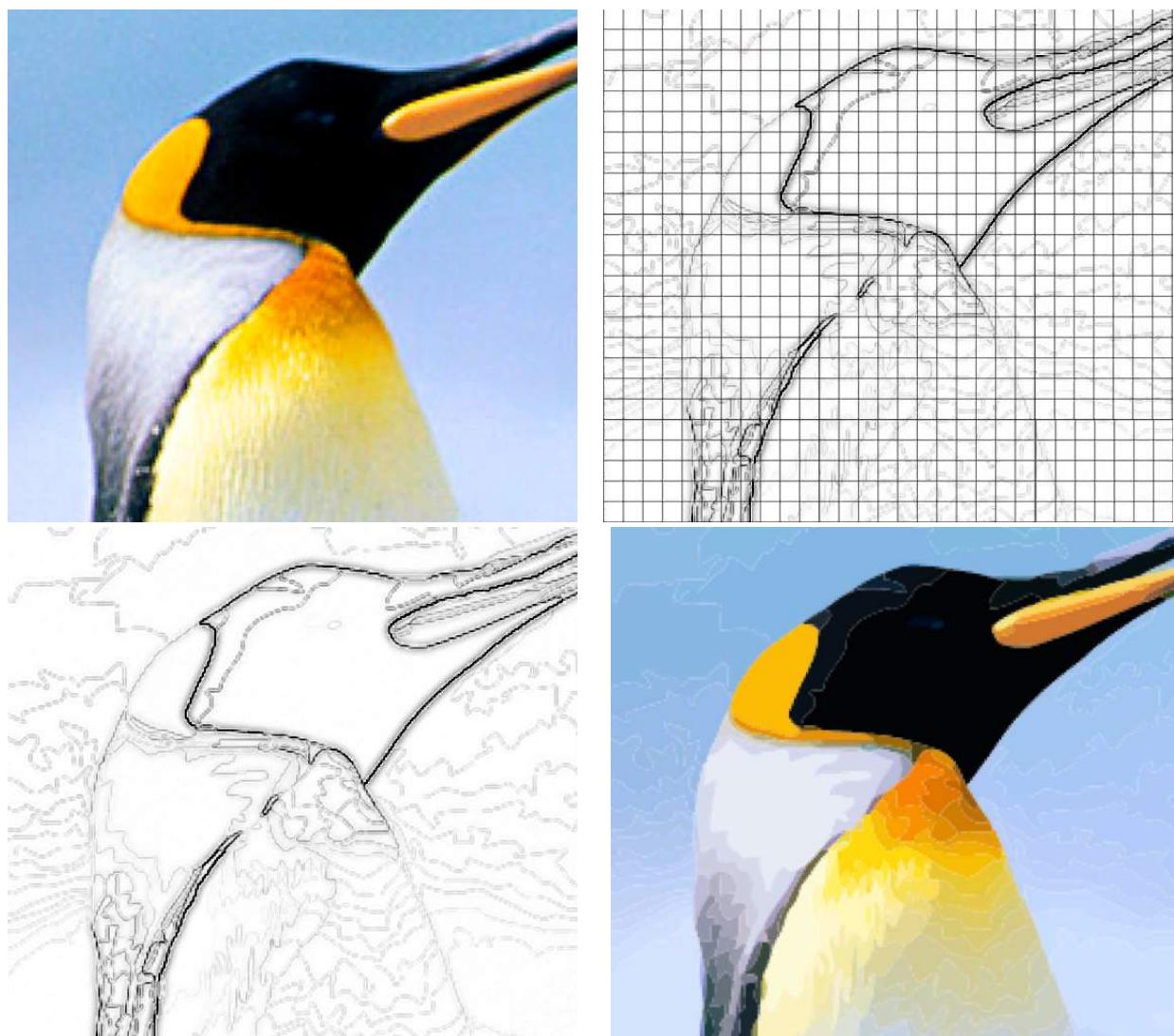
Wektoryzacja (trasowanie) – operacja przekształcenia obrazu rastrowego w wektorowy (trudna)

- polega na wykrywaniu krawędzi w obrazie rastrowym i tworzeniu w oparciu o nie listy obiektów geometrycznych,
- zwektoryzowany obraz nie jest idealnym odpowiednikiem pierwowzoru, dlatego też poszukuje się algorytmów, które zapewnią jak najdokładniejsze odwzorowanie obrazu rastrowego,
- jest to operacja czasochłonna dlatego efektywność algorytmu jest kluczowa.

Idea algorytmu

1. Przygotować obraz do wektoryzacji (często wymaga przetwarzania wstępnego jak np. redukcja liczby kolorów, szumów czy progowanie).
2. Rozpoznać prymitywy - linie, okręgi, litery, znaki, obszary jednobarwne i krawędzie (przetwarzanie właściwe - filtracja krawędziowa, pocienianie erozyjne, segmentacja, metody grafowe, algorytmy oparte na rozproszonych pikselach).
3. Dokonać korekty (uzupełnienie, eliminacja błędów - kolor, kształt, uproszczenie, usunięcie duplikatów).
4. Zapamiętać parametry obiektów i stworzyć ich listę.

Przykład wektoryzacji ilustracji:



obraz rastrowy
rozpoznanie krawędzi

analiza rastra - rozpoznanie obszarów
obraz wektorowy

Rezultat (plik svg):

```
<?xml version="1.0" encoding="UTF-8" ?><!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg width="1024pt" height="768pt" viewBox="0 0 1024 768" version="1.1"
xmlns="http://www.w3.org/2000/svg">
<path fill="#80b6e4" d=" M 814.01 15.16 C 816.30 13.76 819.18 13.12 821.86 13.16 C 823.73 14.12
825.68 14.91 827.76 15.36 C 827.92 20.40 828.00 26.59 835.33 24.95 C 835.55 27.88 834.39 29.54
831.37 29.33 C 830.71 32.64 828.80 35.65 828.94 39.10 C 830.74 37.94 832.31 36.47 834.01 35.16 C
836.19 36.30 838.46 37.27 840.68 38.35 C 842.26 41.17 843.93 43.94 845.44 46.80 C 849.08 46.74
851.86 41.92 855.53 45.13 C 854.87 47.14 855.69 49.29 857.82 49.91 C 859.64 48.32 860.17 44.81
863.03 44.94 C 863.18 47.70 859.28 53.27 866.74 49.96 C 866.58 48.27 866.53 46.57 867.11 44.94 C
868.61 45.51 868.80 46.76 868.38 48.16 C 870.06 48.02 871.44 47.00 872.84 46.14 C 872.11 50.69
876.98 48.85 879.05 47.05 C 880.31 47.84 881.39 48.90 882.65 49.72 C 883.90 48.33 885.14 46.92
886.37 45.50 C 888.89 45.44 892.44 48.28 893.80 45.00 C 893.65 46.65 894.34 47.29 895.86 46.93 C
898.10 44.12 903.49 44.74 905.15 48.00 C 902.53 47.73 899.61 46.26 897.39 48.44 C 895.72 50.10
893.43 49.46 891.36 49.45 C 890.94 50.84 890.47 52.20 888.82 52.39 C 887.40 51.78 886.00 50.77
884.39 51.13 C 883.62 52.31 883.83 55.15 881.98 54.99 C 879.14 55.64 879.62 51.83 878.61 50.15 C
874.50 47.95 873.40 52.57 870.24 53.07 C 870.13 52.31 869.91 50.78 869.81 50.02 C 867.34 51.22
865.56 53.03 865.70 55.95 C 862.66 56.26 861.13 52.78 858.80 51.34 C 857.36 51.84 856.00 53.59
854.42 52.52 C 851.16 50.70 847.13 50.97 844.03 48.95 C 843.00 47.91 841.52 47.92 840.18 48.00 C
840.06 49.29 839.95 50.58 839.86 51.86 C 837.61 51.41 835.49 50.64 835.63 47.93 C 834.01 49.62
833.17 52.79 830.24 51.98 C 830.13 51.25 829.91 49.79 829.81 49.06 C 826.66 50.67 824.22 54.79
820.32 53.93 C 820.61 49.77 816.43 50.86 813.44 51.01 C 813.50 53.51 812.26 56.87 816.12 56.32 C
813.79 59.42 809.37 59.44 806.11 57.84 C 806.32 56.57 806.69 55.34 806.97 54.08 C 805.63 54.69
804.33 55.39 802.98 55.95 C 800.84 54.52 799.63 51.94 797.26 50.90 C 797.13 51.69 796.88 53.29
796.75 54.08 C 794.89 53.82 794.59 51.56 793.00 50.93 C 791.30 51.99 790.06 53.88 787.95 54.10 C
787.65 52.22 789.36 51.49 790.60 50.54 C 790.70 48.08 790.06 45.64 790.11 43.19 C 791.26 41.28
793.62 43.45 794.90 41.80 C 795.05 40.47 794.22 39.31 793.79 38.11 C 791.53 37.30 789.93 38.94
788.33 40.21 C 788.56 38.79 788.81 37.38 788.91 35.94 C 788.99 34.59 789.08 33.23 789.17 31.87 C
790.17 32.89 790.98 34.08 792.06 35.03 C 793.60 34.93 793.97 32.93 794.65 31.80 C 794.79 28.73
791.01 28.61 788.87 27.72 C 791.38 25.58 794.97 25.95 798.06 25.19 C 799.76 24.48 800.29 27.57
801.89 26.84 C 804.93 25.12 812.22 22.23 809.24 29.99 C 812.04 30.10 815.46 29.74 817.45 27.47 C
818.99 28.20 820.36 29.24 821.91 29.97 C 824.55 27.12 820.80 23.83 819.80 21.05 C 816.80 20.84
816.72 25.37 813.71 25.73 C 810.55 27.15 810.43 22.77 808.82 21.16 C 807.29 21.03 804.23 20.78
802.71 20.65 C 806.22 17.53 810.68 20.98 814.69 20.16 C 814.38 18.50 813.93 16.86 814.01 15.16 Z" />
</svg>
```



obraz

rastrowy



obraz

wektorowy

Alternatywne podejście



kwantyzacja
segmentacja

redukacja szumów
kwantyzacja + redukcja szumów + segmentacja



obraz

wektorowy

Zastosowania:

- GIS (ang. Geographic Information System) do tworzenia, archiwizacji i przetwarzania map w postaci cyfrowej,
- przetwarzanie rysunków technicznych do postaci cyfrowej,
- systemy nawigacji,
- analiza obrazu w czasie rzeczywistym.

◀ Zalety i wady

Jump to...

▼

Fonty ►

Navigation

- Dashboard
- Site home
- Site pages
- My courses
- MAD - wykład
- MAD - ćwiczenia
- GU - wykład

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



2

Patryk Obuchowicz -



online M&J

Dashboard > My courses > GU - wykład > Wykład 9 (4.05.2020) > Fonty

Fonty

Fonty

Czcionka to metalowy słupek z wypukłą literą wykorzystywany w tradycyjnej technice drukarskiej (prasy drukarskie, maszyny do pisania).



Font to komputerowy zestaw znaków określonej odmiany kroju pisma (zbiór danych o wszystkich znakach danego kroju pisma). W praktyce pod pojęciem font rozumiemy zestaw glifów z których każdy stanowi osobny znak zaprojektowany w formie rastrowej lub wektorowej.

TIMES NEW ROMAN

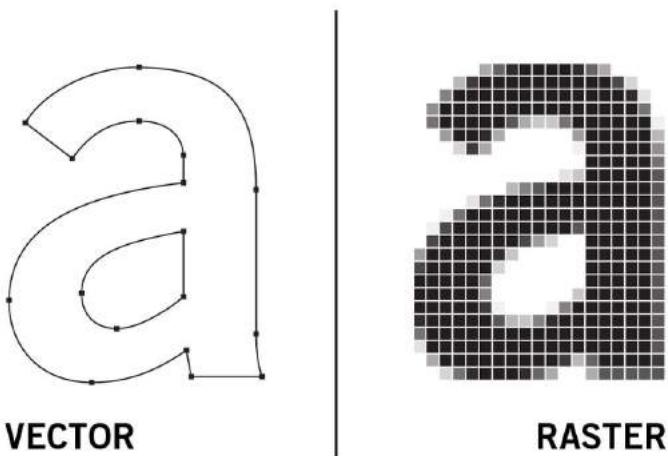
abcdefghijklmnopqrstuvwxyz

ABCDEFGHIJKLMNPQRSTUVWXYZ

1234567890

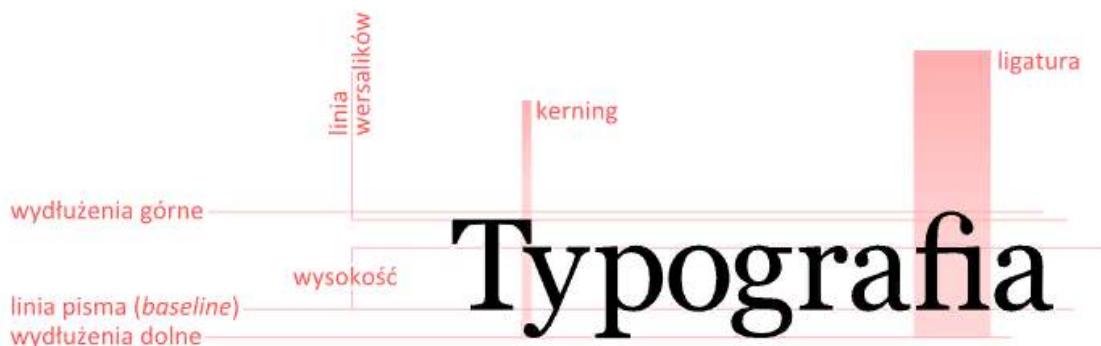
Wyróżniamy dwie klasy fontów

- **rastrowe** - każdy glif to bitmapa,
- **wektorowe** - każdy glif składa się z linii używanych do kreślenia znaku (SVG, kreślone z użyciem javascriptu, głównie na urządzeniach mobilnych)
(małe litery w tym formacie wyglądają gorzej ale format SVG to dokument XML, więc na litery można nakładać gradienty, tekstury, tworzyć animacje),



Fonty kolorowe mogą być zarówno wektorowe jak i rastrowe, w zależności od typu danych SVG, na których są oparte!

Podstawowe pojęcia



Rodzaje fontów

- **PostScript** (Adobe, ps) – wysokiej jakości obrys (obwiednie), ciągi punktów kontrolnych, kontury z krzywych 3 stopnia, Beziera, zawarte w 2 plikach standard dla dtp

- **TrueType** (Apple, ttf) – dtp, format komputerowy,
 - pliki fontów łatwo dostępne,
 - każdy glif definiuje zestaw linii, krzywych (2 stopnia, Beziera),
 - elastyczne, rozszerzalne, 1 plik, Unicode, wady - duży plik, idealne dla czytników.
- **OpenType** (Adobe & Microsoft, ttf, otf, otc, ttc) – łączą zalety TrueType i PostScript
 - obrys może być w formacie ps lub ttf, każdy glif składa się z linii, krzywych,
 - szerszy zestaw znaków (Unicode), znaki alternatywne, wsparcie dla ligatury, kerningu, linii bazowej (wyjaśnienie tych pojęć na obrazku powyżej), małych glifów, literałów, łącznie do 65.536 glifów,
 - skalowalne, niezależne sprzętowo – PC/MAC
- **OpenType TT** – wysoka jakość, szeroki zestaw znaków,
 - kompatybilne z TTF
- **OpenType PS** – idealne dla dtp, kompatybilne z PS,
 - problemy z kompatybilnością sprzętową, duże pliki
- **OpenType v. 1.8.** zmienne
 - zapewniają płynne, niedyskretne przejście między wagami, ciągłą zmienność wzdłuż osi szerokości,
 - zmiana wyglądu bez potrzeby instalowania dodatkowych plików fontów lub dodawania obrysu do tekstu.
- **webowe** – dedykowane dla www, lekkie, dostępne w postaci bibliotek np. Google Web Fonts, ale projektant nie ma kontroli nad wyglądem
 - Embedded Open Type, EOT (Microsoft) – IE
 - Web Open Font Format, WOFF – Firefox, Chrome, Safari, Opera, IE 9+

The screenshot shows a web-based application interface. On the left, there is a sidebar with a red header containing the text "Navigation". Below the header, the sidebar lists several menu items: "Dashboard", "Site home", "Site pages", "My courses", "MAD - wykład", "MAD - ćwiczenia", "GU - wykład", "Participants", "Grades", "GRAFIKA UŻYTKOWA - WYKŁAD", "EGZAMIN", "Informacje", "Wykład 1", "Wykład 2", and "Wykład 3". The "GRAFIKA UŻYTKOWA - WYKŁAD" item is currently selected, indicated by a blue background. The main content area on the right has a white background and contains a large, stylized red graphic element resembling a series of overlapping arches or waves. In the top-left corner of the main area, there is some text and a small arrow icon: "Rasteryzacja i wektoryzacja" with a left-pointing arrow, and "Jump to..." with a dropdown arrow. Below these, there is a link "Zadanie 12 ►".

ZAJĘCIA ZDALNE - prowadzący: M&J Latosińskie



Patryk Obuchowicz -



online M&J

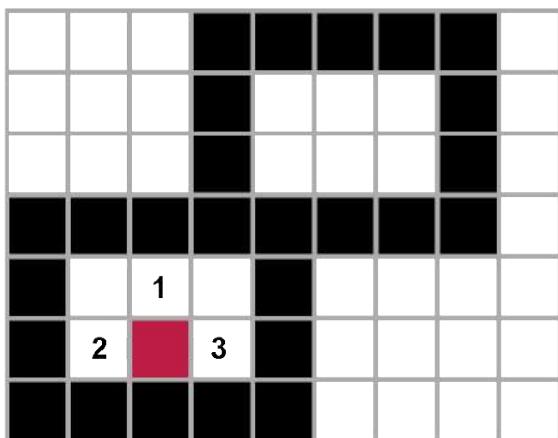
Dashboard > My courses > GU - wykład > Wykład 9 (4.05.2020) > Wypełnianie przez spójność - wyliczenie i wizualiz...

Wypełnianie przez spójność - wyliczenie i wizualizacja (przykład uzupełniający)

Wypełnianie 4-spójne

algorytm odkłada na stosie piksele powyżej, poniżej po lewej i po prawej wybranego piksela tzw. ziarna

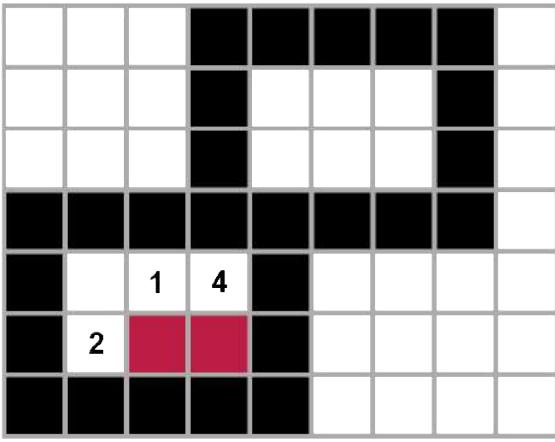
krok 1



wypełniamy kolorem piksel startowy #0, na stosie od dołu do góry są odłożone piksele #1,#2,#3

krok 2

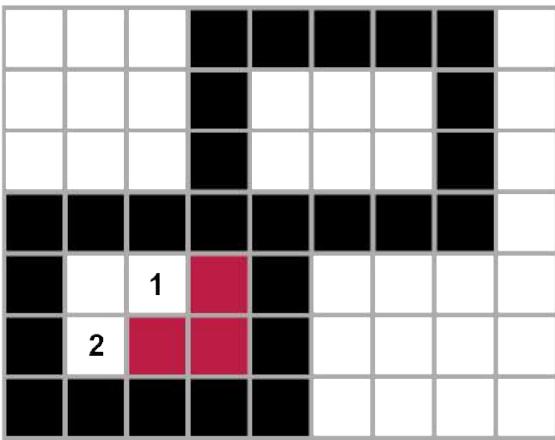
zakolorujemy #3 i zdejmujemy ze stosu



badamy otoczenie piksela #3 i korygujemy stos, na stosie są odłożone są teraz #1,#2,#4

krok 3

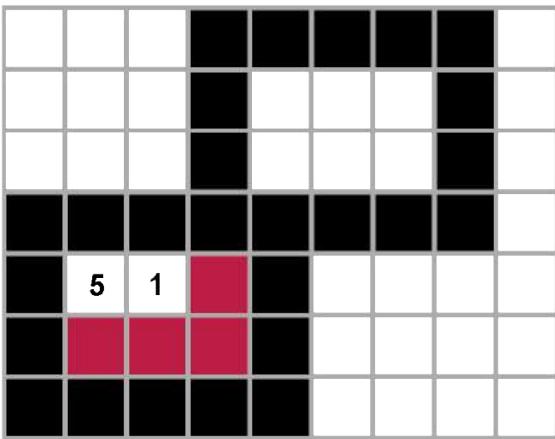
zakolorujemy #4 i zdejmujemy ze stosu



badamy otoczenie piksela #4, nie ma podstaw do korekty stosu, więc na stosie są odłożone są teraz #1,#2

krok 4

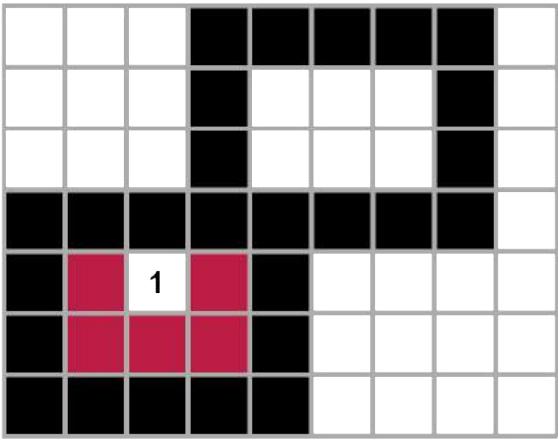
zakolorujemy #2 i zdejmujemy ze stosu



badamy otoczenie piksela #2 i korygujemy stos, na stosie są odłożone są teraz #1,#5

krok 5

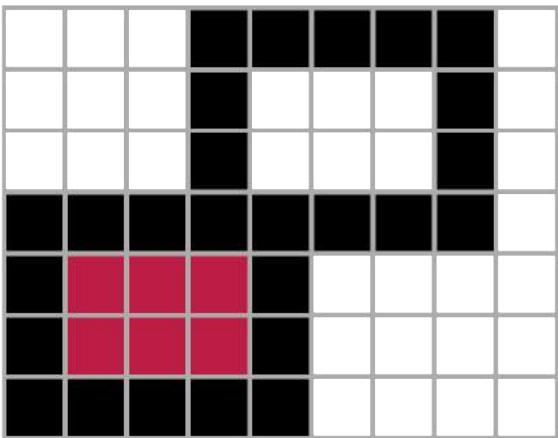
zakolorujemy #5 i zdejmujemy ze stosu



badamy otoczenie piksela #5, nie ma podstaw do korekty stosu, więc na stosie jest odłożony tylko #1

krok 6

zakolorujemy #1 i zdejmujemy ze stosu

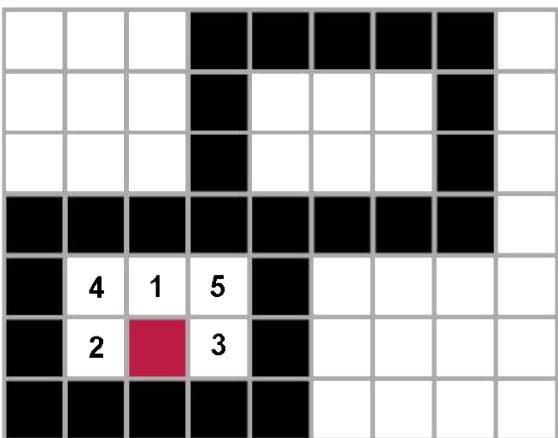


stos jest pusty, algorytm kończy działanie.

Wypełnianie 8-spójne

algorytm odkłada na stosie piksele powyżej, poniżej po lewej, po prawej, powyżej w lewo, powyżej w prawo, poniżej w lewo oraz poniżej w prawo wybranego piksela tzw. ziarna.

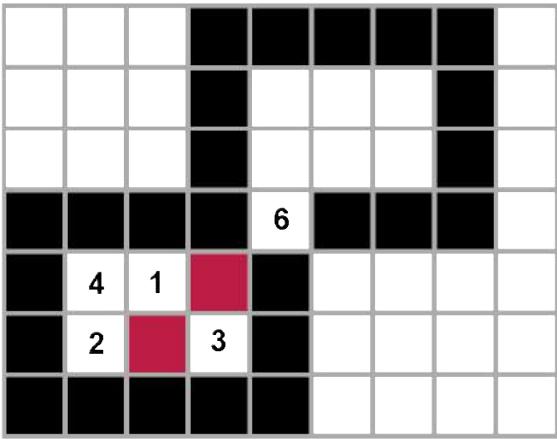
krok 1



wypełniamy kolorem piksel startowy #0, na stosie od dołu do góry są odłożone piksele #1, #2, #3, #4, #5

krok 2

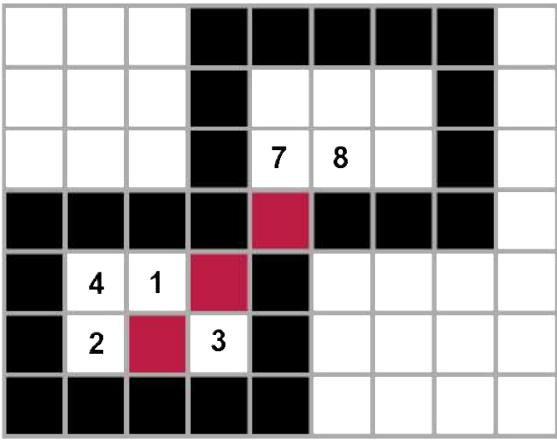
zakolorujemy #5 i zdejmujemy ze stosu



badamy otoczenie piksela #5 i korygujemy stos, na stosie są odłożone są teraz #1,#2,#3,#4,#6

krok 3

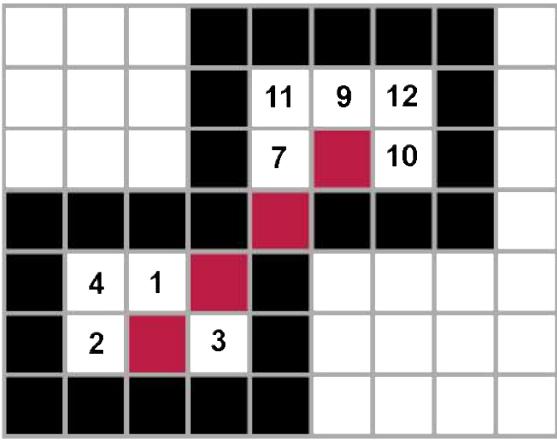
zakolorujemy #6 i zdejmujemy ze stosu



badamy otoczenie piksela #6 i korygujemy stos, na stosie są odłożone są teraz #1,#2,#3,#4,#7,#8

krok 4

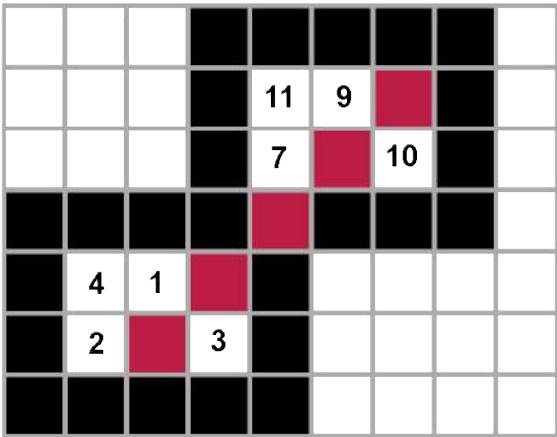
zakolorujemy #8 i zdejmujemy ze stosu



badamy otoczenie piksela #8 i korygujemy stos, na stosie są odłożone są teraz #1,#2,#3,#4,#7,#9,#10,#11,#12

krok 5

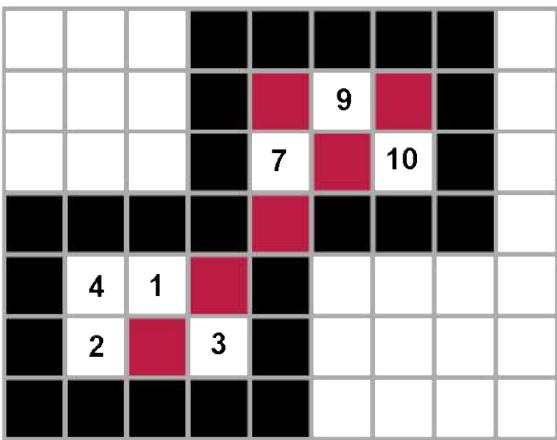
zakolorujemy #12 i zdejmujemy ze stosu



badamy otoczenie piksela #12, nie ma podstaw do korekty stosu, na stosie są odłożone są teraz #1,#2,#3,#4,#7,#9,#10,#11,

krok 6

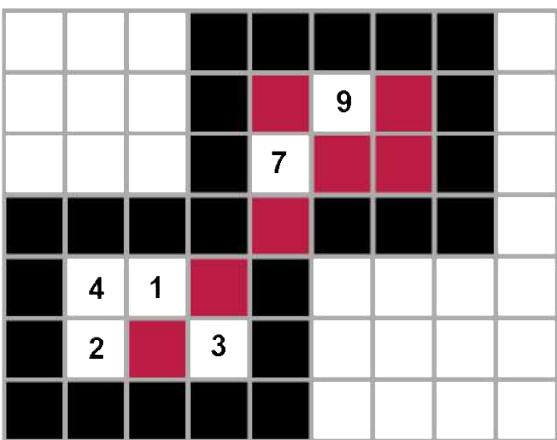
zakolorujemy #11 i zdejmujemy ze stosu



na stosie są odłożone są teraz #1,#2,#3,#4,#7,#9,#10.

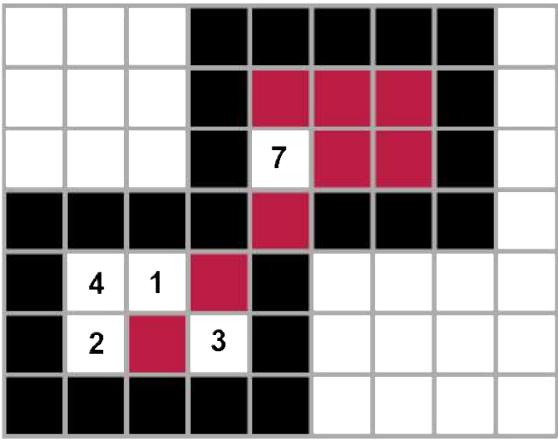
krok 7

zakolorujemy #10 i zdejmujemy ze stosu



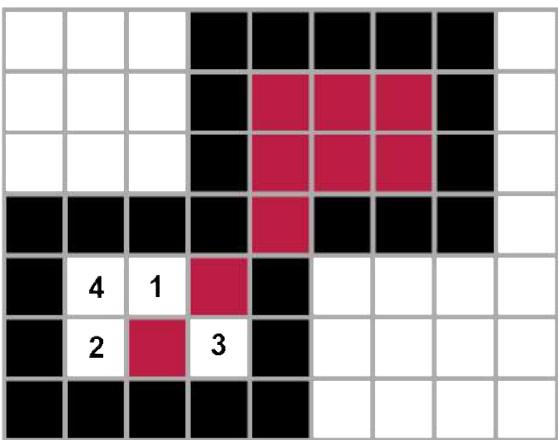
krok 8

zakolorujemy #9 i zdejmujemy ze stosu



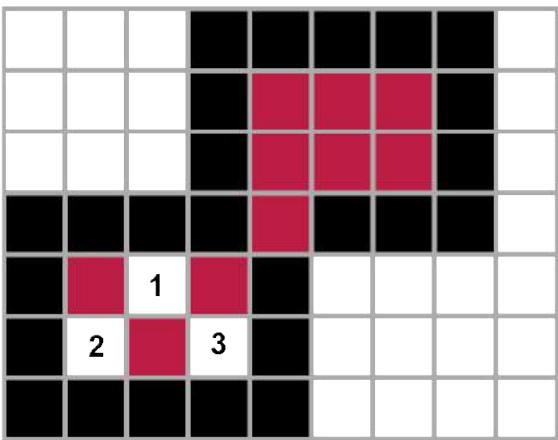
krok 9

zakolorujemy #7 i zdejmujemy ze stosu



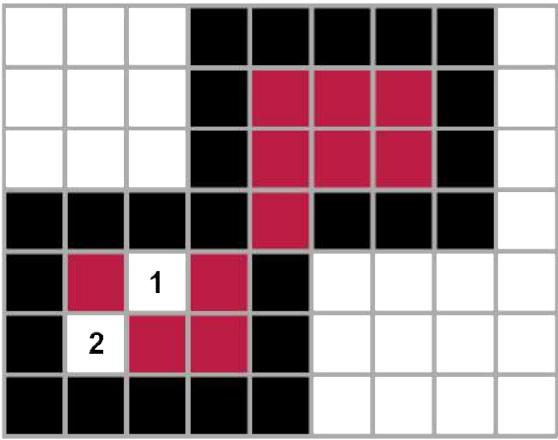
krok 10

zakolorujemy #4 i zdejmujemy ze stosu



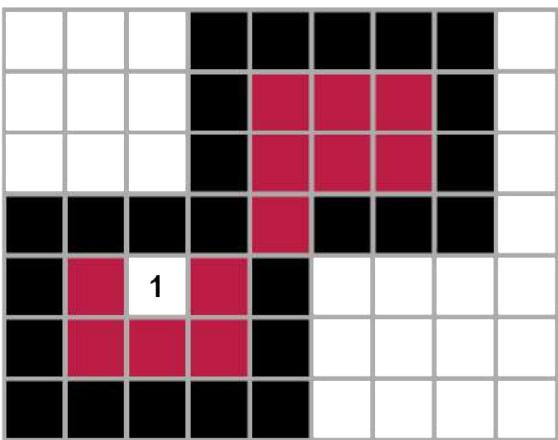
krok 11

zakolorujemy #3 i zdejmujemy ze stosu



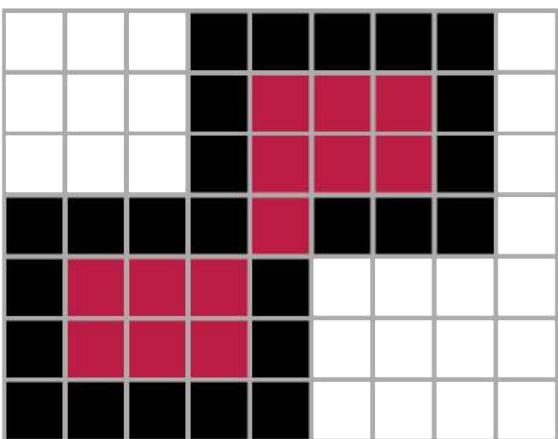
krok 12

zakolorujemy #2 i zdejmujemy ze stosu



krok 13

zakolorujemy #1 i zdejmujemy ze stosu



Kontur zostaje wypełniony a stos jest pusty.