Zachodniopomorska Szkoła Biznesu w Szczecinie

Bartosz Bazyluk

GRAFIKA KOMPUTEROWA

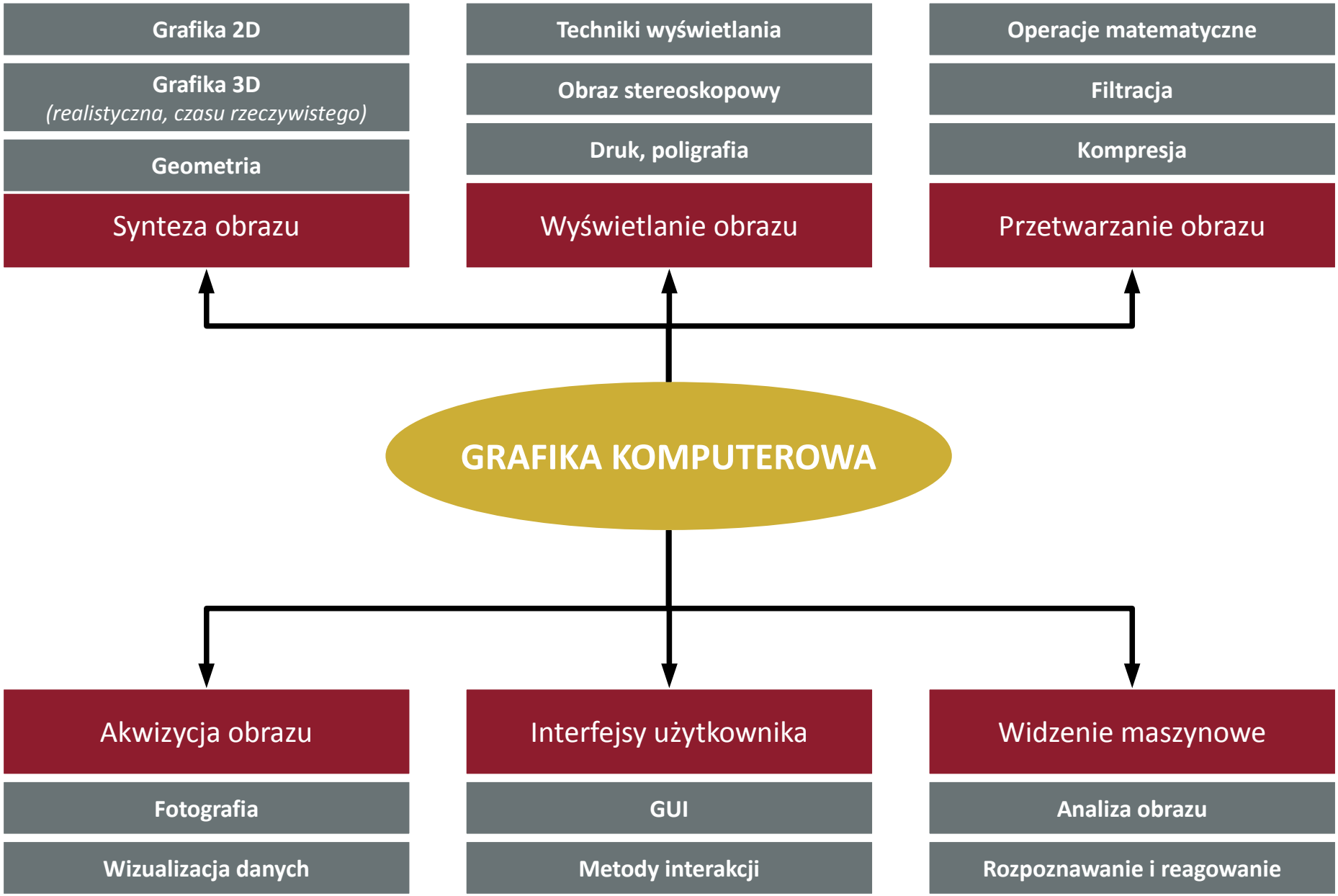
Wprowadzenie. Warunki zaliczenia.

Grafika Komputerowa, Informatyka, I Rok

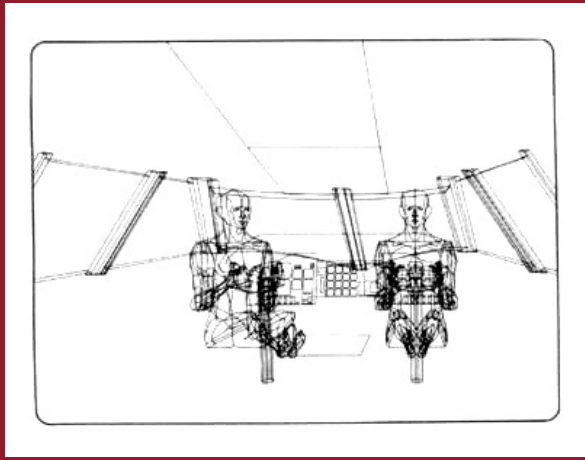
O MNIE

- dr inż. **Bartosz Bazyluk**
- bartosz@bazyluk.net
- http://bazyluk.net/zpsb
  - Wykłady, zadania, materiały, informacje

- Czym się zajmuję:
  - Programowanie **grafiki czasu rzeczywistego**
  - **Eye tracking** jako narzędzie badań i interakcji
  - **Fotografia HDR** i przetwarzanie **obrazów HDR**
  - **Percepcja głębi** w obrazach syntezowanych komputerowo
  - Narzędzia do **projektowania oświetlenia** w budynkach


Grafika 2D
Grafika 3D
(realistyczna, czasu rzeczywistego)
Geometria
Synteza obrazu
Techniki wyświetlania
Obraz stereoskopowy
Druk, poligrafia
Wyświetlanie obrazu
Operacje matematyczne
Filtracja
Kompresja
Przetwarzanie obrazu
GRAFIKA KOMPUTEROWA
Akwizycja obrazu
Fotografia
Wizualizacja danych
Interfejsy użytkownika
GUI
Metody interakcji
Widzenie maszynowe
Analiza obrazu
Rozpoznawanie i reagowanie

GRAFIKA KOMPUTEROWA

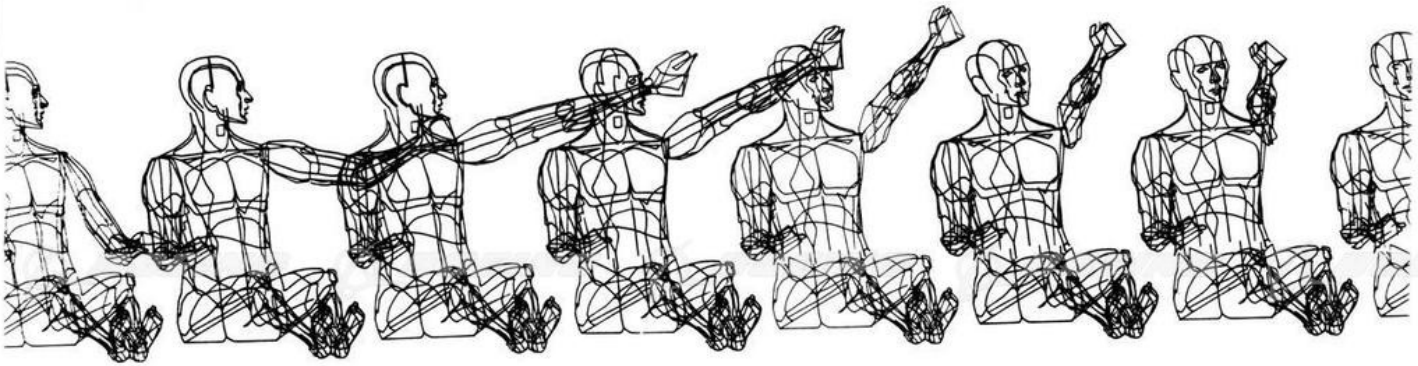


Uważa się, że **termin** ***grafika komputerowa*** został pierwszy raz użyty przez Williama Fettera i Verne Hudsona w latach '60, by opisać stosowane przez nich metody ulepszania ergonomii kokpitu samolotów Boeing *[van Dam 2016 / Fetter 1966]*

Techniki grafiki komputerowej, wyświetlania obrazu i manipulowania nim rozwijały się jednak już wcześniej.

Źródło obrazu:
Boeing; Fetter

- Elementem **wspólnym** dla wszystkich aspektów które składają się na dziedzinę grafiki komputerowej jest **praca z obrazem**:
  - **Tworzenie**
  - **Przechowywanie**
  - **Manipulacja**
  - **Komunikowanie**



ZAGADNIENIA

- Czego będzie dotyczył przedmiot?

  - **Obraz**
    - Reprezentacja, przechowywanie, wyświetlanie
  - **Przetwarzanie** obrazów
    - Kompresja, filtracja
  - **Synteza grafiki** trójwymiarowej
    - Modelowanie sceny 3D z użyciem Blendera
  - Trójwymiarowa **grafika czasu rzeczywistego**
    - Programowanie sceny 3D z użyciem OpenGL
  - Badania **eyetrackingowe**
    - Analiza uwagi wzrokowej podczas oglądania obrazów
  - **Fotografia** i obrazowanie **HDR**
    - Techniczne aspekty fotografii cyfrowej, obrazy HDR

ZALICZENIE

- **Zajęcia laboratoryjne**
  - **Praca na zajęciach** laboratoryjnych
  - **Mini-projekty** z bloków tematycznych
    - Jeden z projektów w ramach zajęć wykładowych
- **Egzamin ustny**
  - Wiedza pozyskana na wykładach
  - **Ocena 5.0 z laboratoriów zwalnia z egzaminu**

Zachodniopomorska Szkoła Biznesu w Szczecinie

Bartosz Bazyluk

GRAFIKA KOMPUTEROWA

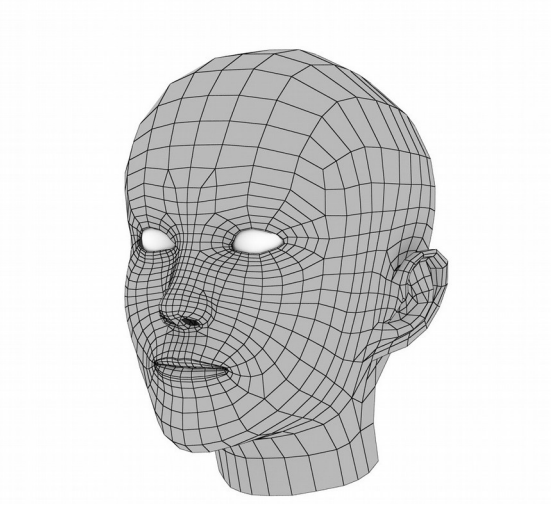
Obraz. Reprezentacja, przechowywanie, kompresja.

Grafika Komputerowa, Informatyka, I Rok

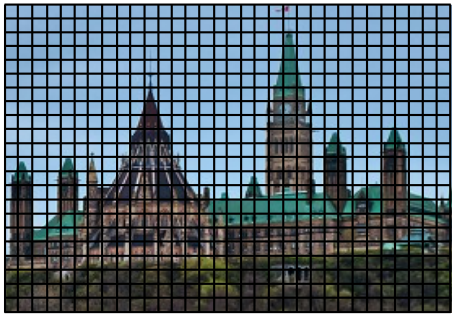
SPOSOBY REPREZENTACJI OBRAZU

Chcąc ***przetwarzać*** obraz w graficznym systemie komputerowym, ten obraz musi być w jakiś sposób ***reprezentowany*** w jego pamięci

- **Sposoby reprezentacji** obrazu komputerowego:
  - Podejście oparte na **geometrii**
    - Obraz ***wektorowy***



  - Podejście oparte na **próbkowaniu**
    - Obraz ***rastrowy***



Źródło obrazu:
Turbosquid

SPOSOBY REPREZENTACJI OBRAZU

Najbardziej charakterystyczna różnica występuje gdy **powiększymy** obraz:

W przypadku **obrazu rastrowego** nie uzyskamy więcej szczegółow, niż mamy dostępnych pikseli.

W przypadku **obrazu wektorowego** bazujemy na matematycznym opisie figur geometrycznych, możemy więc powiększać obraz zachowując dalej kształty.

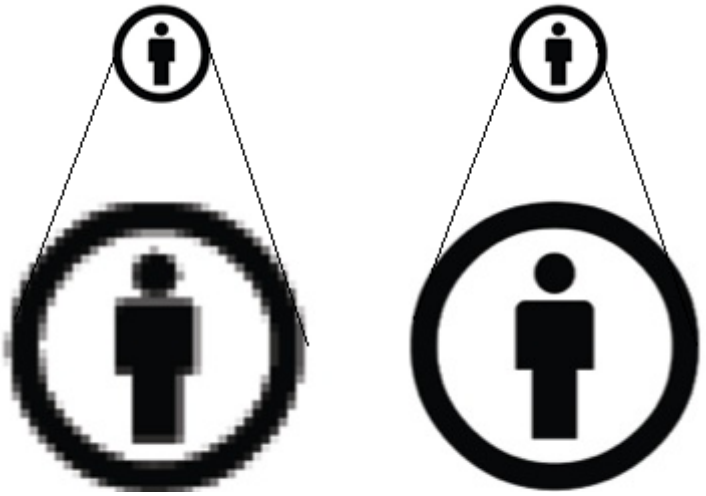
Źródło obrazu:
http://www.printwearmag.com

Obraz **rastrowy**

- **Macierz pikseli**

Obraz **wektorowy**

- **Zbiór opisanych obiektów**



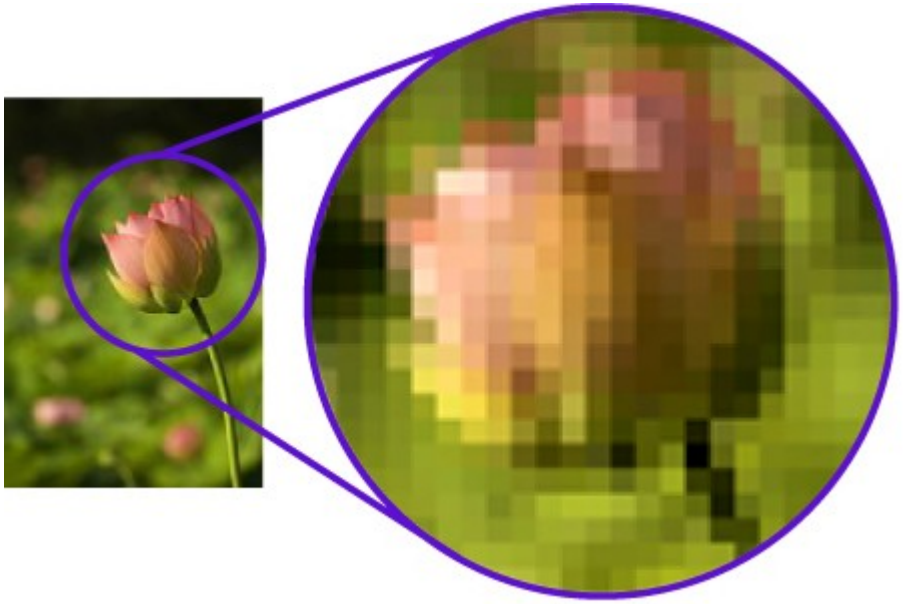
OBRAZ RASTROWY

- Obraz to **macierz pikseli** – tzw. *mapa bitowa*, popularnie *bitmapa* (od ang. *bitmap*)
- Bitmapy są *najczęściej* dwuwymiarowe, prostokątne
- Piksel jest **najmniejszym fizycznym elementem obrazu**, który możemy kontrolować i adresować

Określenie ***mapa bitowa*** odnosi się do obrazu, w którym każdy piksel jest reprezentowany przez **1 bit**.

Powszechnie stosuje się je jednak w odniesieniu do wszystkich obrazów rastrowych, niezależnie od tego ile pamięci jest przeznaczone na każdy z pikseli.

Poprawnym, uniwersalnym, choć praktycznie nigdy nie stosowanym określeniem powinna być **pikselmapa** lub ***pixmapa***.

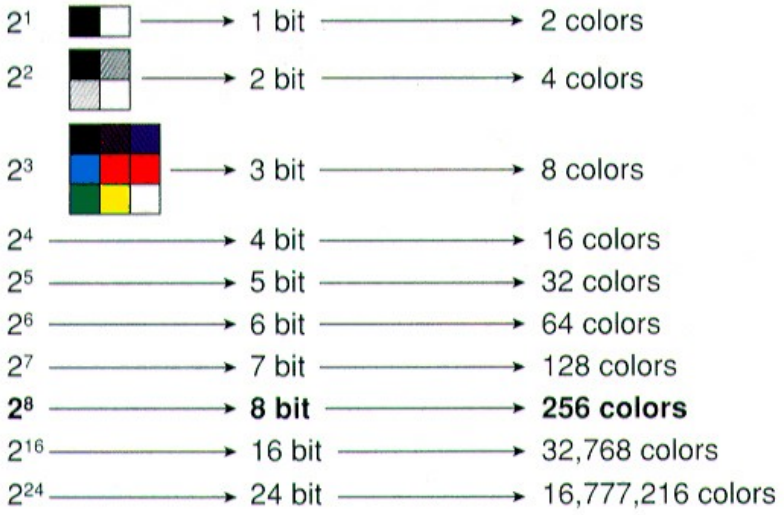


Źródło obrazu:
http://www.vector-conversions.com

OBRAZ RASTROWY

Liczbę bitów przeznaczonych do opisania jednego piksela nazywa się często **głębią koloru** (ang. *color depth*) danego obrazu.

- Bitmapę charakteryzują:
  - **szerokość**, **wysokość** [px]
    - np. 1680×1050px
  - **liczba kanałów**
    - np. RGB (*Red/Green/Blue*), czyli 3 kanały
    - np. RGBA (RGB+Alpha, czyli przezroczystość)
  - **rozmiar piksela** [bpp]
    - np. 24bpp w przypadku RGB = 8 bitów na kanał
      - oznacza to, że każdy z kanałów może przyjąć 2^8 różnych wartości



Źródło obrazu:
http://www.foothill.edu

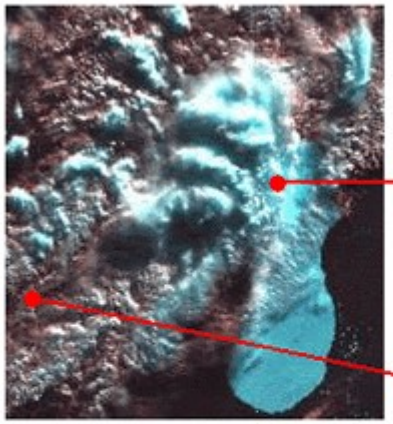
OBRAZ RASTROWY

Dawniej indeksowanie koloru wykorzystywano do uzyskania ciekawej, **prostej animacji obrazu**. Wartości pikseli były stałe, ale **co klatkę zmieniana była paleta**. To podejście nosi nazwę ***color cycling***.

Przykład: ten sam obraz z dwiema różnymi paletami:

Źródła obrazów:
http://effectgames.com
http://www.hdfgroup.org

- Bitmapa może posiadać **kolory indeksowane**
  - zamiast konkretnych wartości RGB pikseli, zapisane są tylko **identyfikatory kolorów** (oszczędność pamięci)
  - powiązanie identyfikatorów z konkretnymi kolorami (**paleta**) może być dołączone do bitmapy jako swoisty *słownik*
  - identyfikatory mogą też dotyczyć np. standardowej **palety systemowej**



8-bit raster image

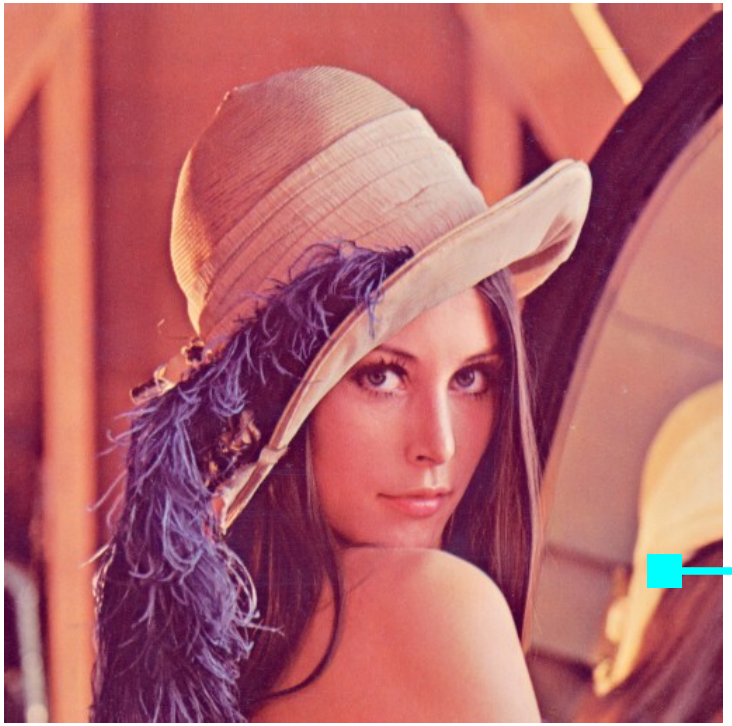
	RED	GREEN	BLUE	RGB
0	10	9	15	
1	28	130	124	
2	152	131	131	
3	109	65	69	
4	152	200	203	
5	64	196	203	
6	34	68	77	
7	96	135	141	
8	209	196	199	
9	36	100	118	
10	97	166	168	
11	61	25	30	
12	206	152	155	
13	153	227	233	
14	113	98	101	
15	153	166	168	
16	209	229	227	
17	90	227	236	
18	80	67	71	

Color look-up table (palette)

OBRAZ RASTROWY

Niezależnie od tego *ile* obiektów znajduje się na obrazie i *jak wiele* posiadają one detali, **ilość pamięci niezbędna do przechowania obrazu rastrowego w postaci nieskompresowanej zawsze będzie stała** i zależeć będzie wyłącznie od liczby pikseli i głębi koloru.

- Ilość **pamięci** potrzebna do zapisania mapy bitowej:
  - szerokość × wysokość × rozmiar piksela
  - Przykład:



512px

512px

24bpp

(512 × 512 × 24) / 8 = 786.432B = **768kB**

ponieważ 1B = 8b

OBRAZ RASTROWY

Podział na metody *stratne i bezstratne* nie dotyczy wyłącznie kompresji obrazu.

Wszędzie tam gdzie kompresując dane można pozwolić sobie na utratę jakości z uwagi na np. niedoskonałość aparatu percepcyjnego człowieka, warto rozważyć **metodę stratną**.

Gdy dane muszą zostać odtworzone w niezmienionej postaci (np. treść dokumentu, kod programu), wówczas słuszna będzie wyłącznie **metoda bezstratna**.

Źródło obrazu:
http://cscie12.dce.harvard.edu

- Zapisując bitmapy w pamięci masowej w celu przechowania, często stosuje się **kompresję**
- Algorytmy kompresji można podzielić na dwie podstawowe grupy:
  - Kompresja **bezstratna**
    - Możliwe jest całkowite **odtworzenie pierwowzoru**
  - Kompresja **stratna**
    - W wyniku kompresji dochodzi do **utraty jakości**, np. szczegółów lub koloru
    - Powstają zniekształcenia, tzw. *artefakty*

OBRAZ RASTROWY

Należy rozróżnić **format pliku** (tzw. *kontener*) od **formatu danych** obrazu, **metody kompresji** i **kodeka** służącego kompresji.

Format pliku mówi o kolejności w jakiej dane zapisane są w pliku, włączając w to strumień danych samego obrazu jak i dane jemu towarzyszące.

Format danych jest standardem w jakim zapisany jest strumień bitów opisujących kolejne piksele.

Metoda kompresji jest algorytmem służącym zamianie oryginalnego zbioru wartości pikseli w strumień bitów.

Kodek jest konkretną implementacją algorytmu kompresji.

- Popularne **formaty** plików, zapisu i kompresji bitmap:
  - **BMP**
    - duża dowolność formatu danych
    - może wykorzystywać kompresję bezstratną *RLE*
  - **JPEG**
    - kompresja stratna
    - brak obsługi przezroczystości
    - 24 bpp
  - **PNG**
    - kompresja bezstratna
  - **GIF**
    - kompresja bezstratna
    - zazwyczaj max. 8 bpp
    - możliwość zapisu sekwencji obrazów jako animacja
  - **TIFF**
    - duża dowolność formatu danych i sposobów kompresji

OBRAZ RASTROWY

Często standardy opisują zarówno format pliku, format danych jak i metodę kompresji.

Przykładem jest *Portable Network Graphics (PNG)*.

Idea formatu pliku, choć nastawiona na zastosowanie jednego formatu danych, służy przyszłemu rozwojowi i kompatybilności wprzód.

- **Kompresja *JPEG***
  - Standard *JPEG* opisuje działanie **metody kompresji** i **format danych**, a nie **format pliku**
  - Najpopularniejsze **formaty plików**:
    - *JPEG/Exif*
    - *JPEG/JFIF*
  - Dane obrazu JPEG mogą być też zawarte w plikach wielu innych formatów, np. w muzycznym formacie *mp3*.

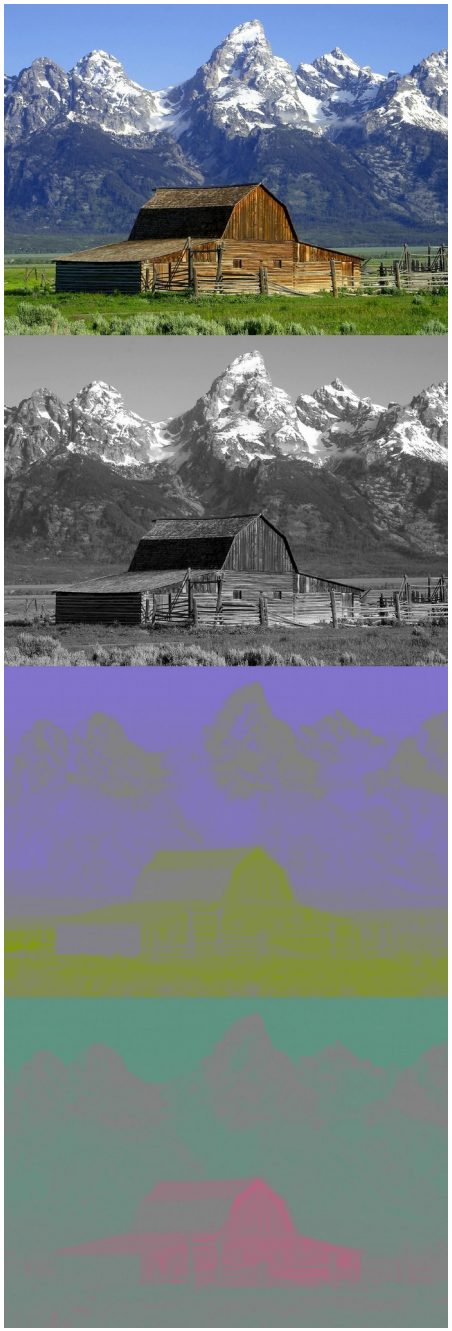
OBRAZ RASTROWY

Ludzki **aparat widzenia** jest **bardziej czuły na zmiany kontrastu, niż zmiany koloru**.

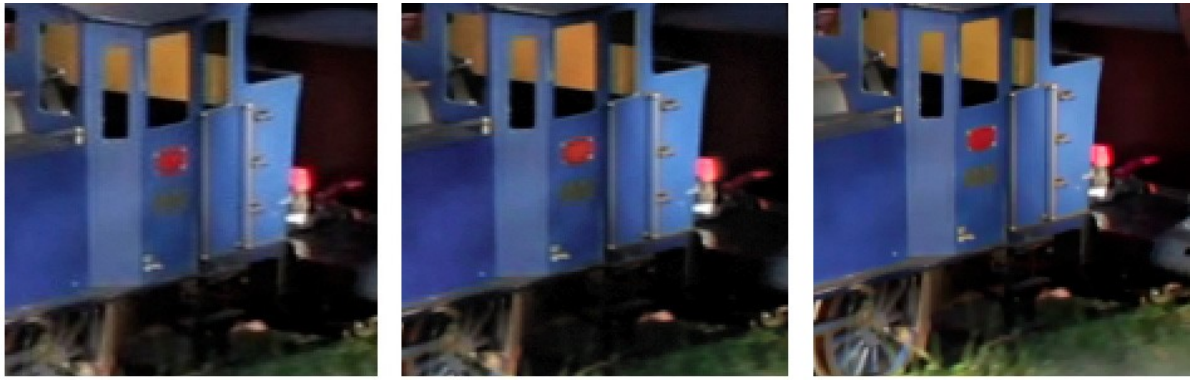
Dlaczego więc nie pokusić się o zmniejszenie rozdzielczości informacji o kolorze, zachowując oryginalną rozdzielczość informacji o kontraście?

Źródło obrazu:
http://wikipedia.org

- **Kompresja *JPEG* – krok 1/4**
  - Przejście z przestrzeni **RGB** do **$Y'C_BC_R$**
    - **Y'** – luma
      - Poziom jasności (nieliniowy!)
    - **C_B, C_R** – chroma
      - **C_B** – różnica koloru niebieskiego i Y'
      - **C_R** – różnica koloru czerwonego i Y'
  - Dzięki temu możliwe jest **osobne traktowanie lumy**, a co za tym idzie **kontrastu**, od informacji o kolorze



- **Kompresja *JPEG* – krok 2/4**
  - **Redukcja rozdzielczości** informacji o **kolorze** (ang. *chroma subsampling*)
    - Skoro człowiek trudniej zauważa różnicę w zmianach koloru niż kontrastu, to *po co marnować pamięć* na przechowywanie informacji o kolorze w pełnej rozdzielczości?
    - 4:4:4 – bez redukcji
    - 4:2:2 – 2-krotna redukcja w poziomie
    - 4:2:0 – 2-krotna redukcja w pionie i poziomie
    - Wyjaśnienie zapisu: http://en.wikipedia.org/wiki/Chroma_subsampling#Sampling_systems_and_ratios
  - Krok **opcjonalny**



4:2:0 4:2:2 4:4:4

Źródło obrazu: http://wikipedia.org

OBRAZ RASTROWY

Charakterystyczne, **blokowe artefakty** są wynikiem osobnego przetwarzania bloków 8×8 pikseli.

- **Kompresja *JPEG* – krok 3/4**
  - Obraz dzielony jest na **bloki 8x8 pikseli**
    - Stąd bierze się często widoczny "blokowy" artefakt w obrazach JPEG
  - *Każdy* z kanałów $Y'C_BC_R$ w *każdym* z bloków jest osobno poddawany działaniu **dyskretnej transformacji kosinusowej** (ang. *discrete cosine transform, DCT*)
    - Przejście do dziedziny częstotliwości
    - Przed zastosowaniem *DCT* wartości [0;255] przesuwane są do zakresu [-128;127]
    - Tymczasowo każdy komponent zajmuje więcej niż 8 bitów
  - Wartości po *DCT* poddawane są **kwantyzacji**
    - Jedyny (poza subsamplingiem) **stratny** element kodeka, tego etapu dotyczy ustawienie **stopnia kompresji**
    - Pozwala pominąć zmiany **wysokiej częstotliwości**
- **Kompresja *JPEG* – krok 4/4**
  - Dodatkowa, **bezstratna kompresja Huffmana**
  - Może być poprzedzona bezstratnym **kodowaniem entropii**

OBRAZ RASTROWY

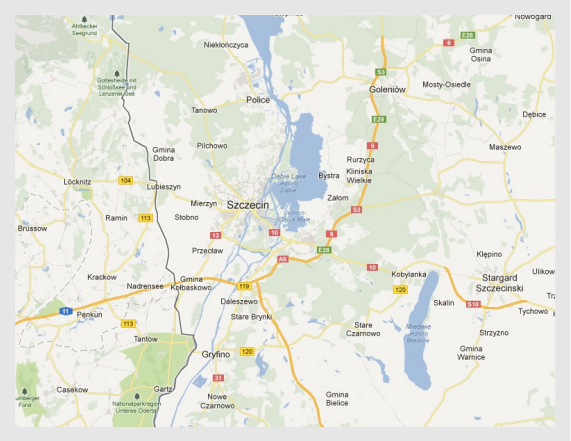
Dobierając metodę kompresji należy zwrócić uwagę na charakter kompresowanego obrazu.

Zazwyczaj obrazy z niewielką liczbą kolorów najlepiej jest kompresować korzystając z formatu wykorzystującego paletę.

Obrazy zawierające zmiany niskiej częstotliwości jak np. rastrowe ilustracje, mapy, zazwyczaj optymalnie jest kompresować z użyciem metod bezstratnych.

Zdjęcia i inne obrazy zawierające wiele szczegółów i zaszumienie zwykle dobrze sprawdzą się przy stosowaniu metod stratnych.

- Różne metody kompresji nadają się do **różnego rodzaju obrazów**
- Należy zawsze wziąć pod uwagę charakter kompresowanego obrazu by uzyskać **optymalny stosunek jakości do rozmiaru pliku**
- Większość metod kompresji pozwala **dobrać jej parametry**
- Przykłady:



Brak kompresji: **1386kB**
PNG: **96kB**, jakość *idealna*
JPG: **99kB**, jakość *b.zła*
JPG: **285kB**, jakość *db.*

Brak kompresji: **1875kB**
PNG: **1268kB**, jakość *idealna*
JPG: **480kB**, jakość *b.db.*
JPG: **246kB**, jakość *akceptowalna*

OBRAZ RASTROWY

Transformacje geometryczne, takie jak **skalowanie** lub **rotacja**, wymagają **ponownego próbkowania** obrazu rastrowego (ang. *resampling*).

W jego wyniku znajdowane są wartości nowych pikseli na podstawie pikseli z obrazu źródłowego, poddawanego transformacji.

Ponowne próbkowanie najczęściej prowadzi do utraty informacji z obrazu i **pogorszenia jego jakości**.

Jest wiele metod resamplingu, np. metoda najbliższego sąsiada, filtracja dwuliniowa, *mipmapping*, filtracja trójliniowa...

- Gdy bitmapa ma zostać wyświetlona w rozmiarze **mniejszym niż oryginalny**, musi zostać przeprowadzony *downsampling*

96×96px → 64×64px

Metoda **najbliższego sąsiada** (*nearest neighbour*)

Metoda **filtracji dwuliniowej** (*bilinear filtering*)

OBRAZ RASTROWY

Istnieją bardziej skomplikowane algorytmy *upscalingu*, które biorą lub próbują wziąć pod uwagę treść powiększanego obrazu.

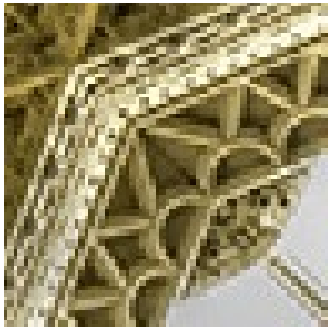
Podejście nazywane ***content-aware scaling*** może opierać się na rozmywaniu obrazu dalekiego od wyraźnych krawędzi lub uzupełnianiu obrazu charakterystycznymi obiektami z pozostałej części źródła.

Źródło obrazu:
Adobe

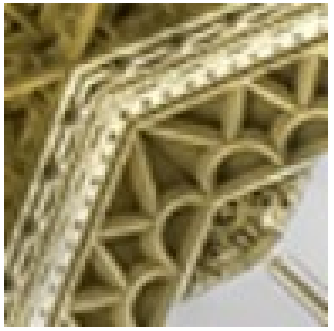
- Gdy bitmapa ma zostać wyświetlona w rozmiarze **większym niż oryginalny**, musi zostać przeprowadzony *upsampling*

64×64px → 96×96px

Metoda **najbliższego sąsiada** (*nearest neighbour*)



Metoda **filtracji dwuliniowej** (*bilinear filtering*)



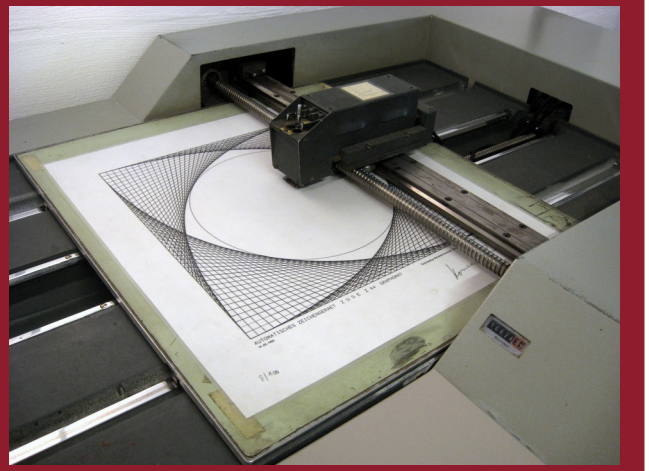
OBRAZ RASTROWY

- **Cechy charakterystyczne**:
  - **Transformacje obrazu (obrót, skalowanie) są trudne** i prowadzą do **pogorszenia jakości** obrazu
  - Wymaga **tej samej ilości pamięci** niezależnie od tego *co* przedstawia obraz i jak bardzo jest on złożony
  - **Pozwala na manipulację** kolorystyką i zawartością obrazu stosunkowo małym kosztem
  - Dobry do reprezentowania obrazów, które przedstawiają **nieprzewidywalnie zmienne obiekty**, naturalne sceny, zaszumione obrazy (np. fotografie)

OBRAZ WEKTOROWY

Wyświetlenie obrazu wektorowego na medium rastrowym, np. na typowym monitorze komputerowym, wymaga wcześniejszej **rasteryzacji** obrazu.

Przykładowym urządzeniem służącym do prezentacji obrazów wektorowych jest **ploter**.

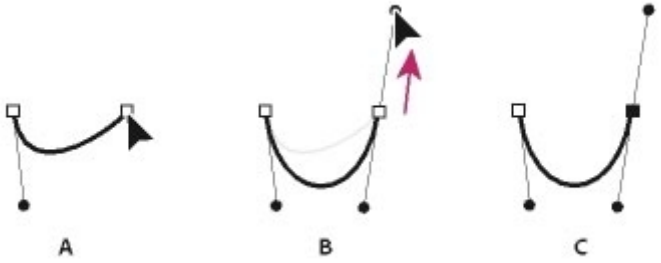


- Obraz opisany jest *nie* za pomocą kolorów równomiernie rozłożonych pikseli, ale za pomocą **informacji o obiektach** które go tworzą
- Przy zapisywaniu obrazu wektorowego, wystarczy że zapamiętamy jakie **kształty** tworzą obraz
  - System graficzny dokona **rasteryzacji** obrazu w procesie ***renderowania***
- Rozmiar pliku, jak i szybkość jego rasteryzacji **zależą od liczby obiektów** tworzących obraz
  - Wyświetlanie skomplikowanych obrazów wektorowych jest bardzo kosztowne – przykład: szczegółowe plany miast w formacie *Adobe PDF*

Źródła obrazów:
Wikipedia
PEPSI, Iconfinder, TFL, Hed Kandi

OBRAZ WEKTOROWY

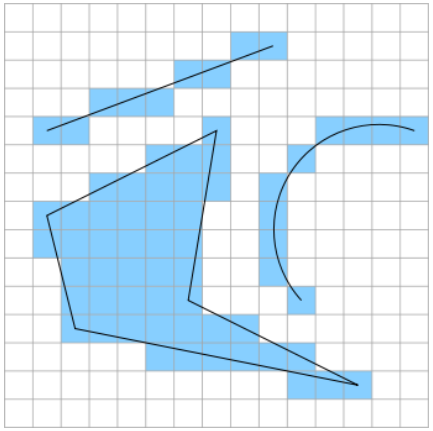
- Obraz opisywany jest za pomocą prostych, parametrycznych figur geometrycznych, tzw. **prymitywów**:
  - Linie
  - Krzywe
  - Wielokąty
  - Okręgi
  - Tekst
- Figury te mogą być później **dowolnie i niezależnie modyfikowane**



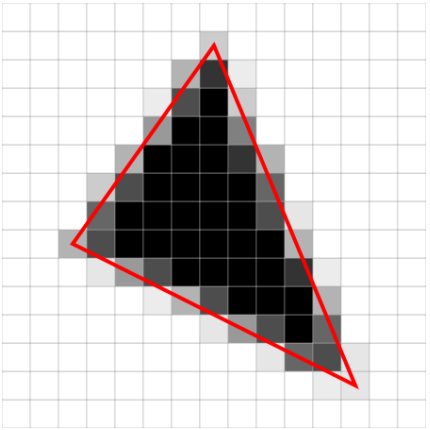
Źródło obrazu:
http://cpregier.wikispaces.com

OBRAZ WEKTOROWY

- Przed wyświetleniem musi zostać poddana **rasteryzacji**
  - Musi zostać utworzona **bitmapa** w oczekiwanym rozmiarze, uwzględniając skalowanie
  - **Kosztowny** proces dla obrazów złożonych z dużej liczby obiektów
  - Pożądane jest użycie technik zmniejszających problem *aliasingu* (ostrych krawędzi), tzw. ***anti-aliasing***



bez antialiasingu



antialiasing

Źródło obrazów:
http://www.wikipedia.org

OBRAZ WEKTOROWY

- **Cechy charakterystyczne**:
  - **Transformacje geometryczne bez utraty szczegółowości**
  - **Kosztowny w wyświetlaniu** z uwagi na proces rasteryzacji, szczególnie w przypadku bardzo złożonych obrazów
  - Ilość niezbędnej pamięci **zależna od złożoności obrazu**
  - Umożliwia **edycję obiektów zachowując topologię**
  - **Nie jest odpowiednia** do reprezentacji obrazów realistycznych, zaszumionych, cechujących się zmianami o wysokiej częstotliwości, jak np. zdjęć (ogromna liczba obiektów, konieczność wcześniejszej wektoryzacji bez wiedzy o topologii sceny)
  - Zwykle świetnie sprawdza się do rysunków, ilustracji, diagramów, map

Zachodniopomorska Szkoła Biznesu w Szczecinie

Bartosz Bazyluk

GRAFIKA KOMPUTEROWA

Obraz. Reprezentacja, przechowywanie, kompresja.

Grafika Komputerowa, Informatyka, I Rok