

Przetwarzanie obrazów

- Handout

SZ 2024/2025

Wydział Matematyki, Fizyki i Informatyki



Dr.-Ing. A. Rowińska-Schwarzsteller
agnieszka.rowinska-schwarzsteller@ug.edu.pl

29. September 2024

Spis treści

1 Wstęp	3
1.1 Wprowadzenie do cyfrowego przetwarzania obrazów	3
1.1.1 Image Processing	3
1.1.2 Image Analysis	4
1.1.3 Przykład: profesjonalna analiza obrazu w przemyśle	4
1.2 Literatura	5
1.3 Software	6
1.4 Kursy online	6
2 Zmysł wzroku	7
2.1 Widzenie	7
2.2 Anatomia oka i jej konsekwencje	7
2.2.1 Oko a kamera	7
2.2.2 Anatomia oka (siatkówka oka): reakcja na światło	7
2.2.3 Teorie dotyczące widzenia barwnego	9
2.2.4 Kolorymetria (barwometria)	11
2.2.5 CIE XYZ	13
2.2.6 CIELab	17
2.2.7 Model przestrzeni barw RGB	20
2.2.8 Inne modele przestrzeni barw	21
2.3 Widzenie jako filtrowanie obrazu	24
2.4 Przetwarzanie obrazu: ekstrakcja obiektów	24
2.5 Przetwarzanie obrazów przez człowieka - zagadnienia	25
2.5.1 Złudzenia /iluzje optyczne	26
2.6 Związek między widzeniem a maszynowym przetwarzaniem obrazu	27
3 Odtwarzanie obrazów	29
3.1 Reprezentacja obrazu w postaci cyfrowej	29
3.1.1 Obrazy w skali szarości	30
3.1.2 Obrazy RGB	30
3.2 Skalowanie obrazów	35
3.2.1 Powtórzenie pikseli (<i>Nearest neighbor</i>)	36
3.2.2 Interpolacja dwuliniowa (<i>Bilinear filtering</i>)	38
3.2.3 Interpolacja dwusześcienna	39
3.2.4 Inne algorytmy skalowania	39
3.3 Dithering	42
3.3.1 Progowanie (<i>Thresholding</i>)	43

3.3.2	Algorytm Floyda-Steinberga	43
3.3.3	Raster (<i>Halftone</i>)	46
3.3.4	Progowanie progiem zmiennym (<i>Ordered dithering</i>)	46

1

Wstęp

1.1 Wprowadzenie do cyfrowego przetwarzania obrazów

Obszary zastosowań cyfrowego przetwarzania obrazów:

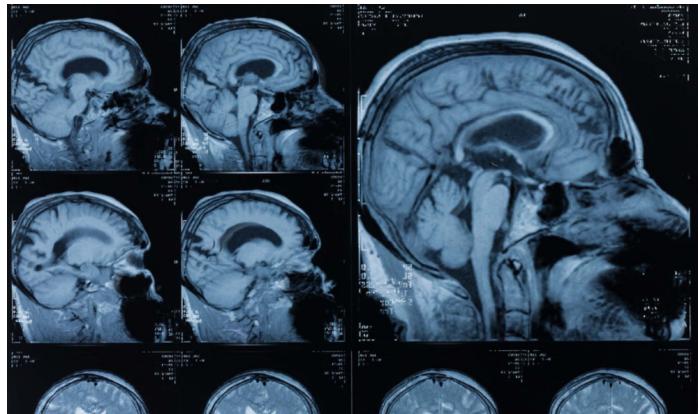
- A. Image Processing

Przetwarzanie obrazu: Obraz \rightarrow Obraz

- B. Image Analysis/Computer Vision

Analiza obrazu/Widzenie komputerowe: Obraz \rightarrow Informacja

1.1.1 Image Processing

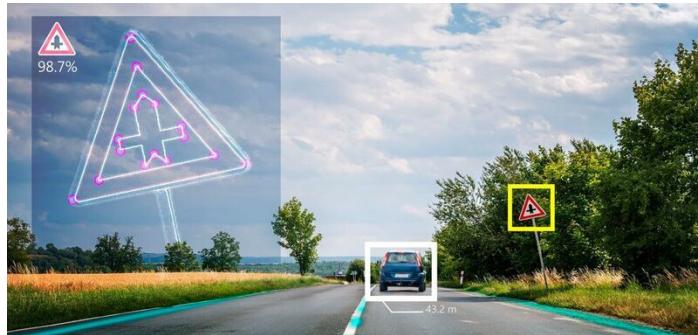


Grafika 1.1: Obrazowanie metodą rezonansu magnetycznego (MRI, magnetic resonance imaging). Źródło: "Kopf MRT zur Entdeckung von Hirnembolien", C.Völklingen

- Procedury obrazowania medycznego
- Redukcja szumów, rozmycia ruchu i zniekształceń obrazu

- Kompresja danych obrazu (transmisja, przechowywanie)
- Filtrowanie
- Teledetekcja (ocena obrazów wielospektralnych, cyfrowe modele terenu)
- Poprawa kontrastu

1.1.2 Image Analysis



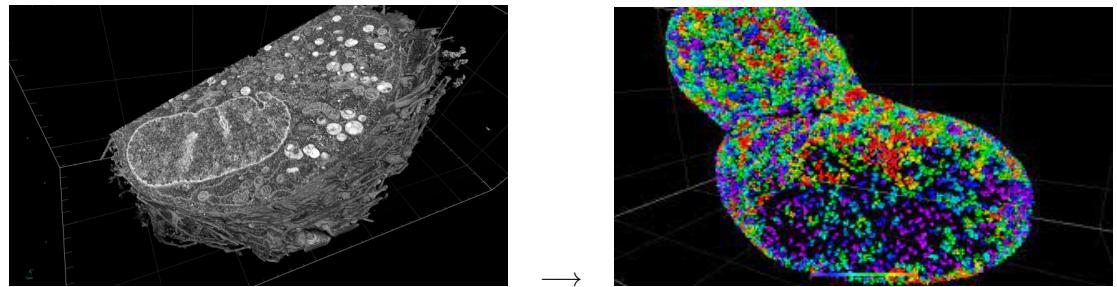
Grafika 1.2: Wspomaganie kierowcy

- Kontrola jakości
- Czytanie tekstów pisanych odręcznie (systemy sortowania listów)
- Robotyka, automatyka, sterowanie
- Produkcja (montaż, sortowanie)
- Systemy nadzoru (obszary niebezpieczne, lotnisko)
- Zautomatyzowana ocena obrazów medycznych (wykrywanie nowotworu)
- Prognoza pogody
- Kryminalistyka (odciski palców)
- Archeologia
- Zastosowania wojskowe
- Kontrola dostępu
- Wspomaganie kierowcy (wnętrze, wykrywanie obiektów)

1.1.3 Przykład: profesjonalna analiza obrazu w przemyśle

Zastosowanie analizy obrazu w objętościowej mikroskopii elektronowej (Volume Electron Microscopy):

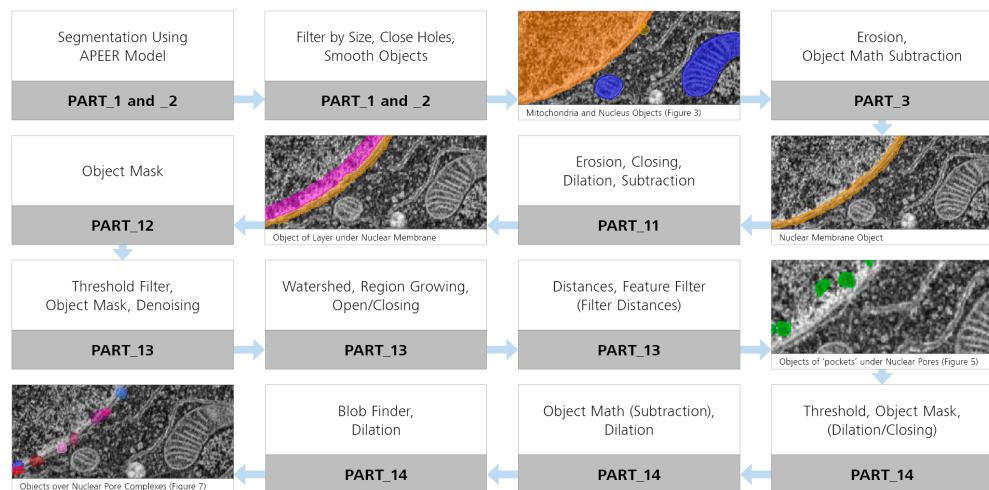
Pomiar i rozkład mitochondriów, jądra i obszarów porów jądrowych (grafika 1.3)
Software ZEISS arivis Pro



Grafika 1.3: Przetwarzanie obrazu: linia komórkowa HeLa - badanie biologii komórek nowotworowych. Źródło: materiały Zeiss

Proces przetwarzania obrazu

→ (grafika 1.4)



Grafika 1.4: Image Analysis Pipeline (ZEISS arivis Pro). Źródło: materiały Zeiss

1.2 Literatura

- *Signals And Systems, Oppenheim, Alan V., and A. S. Willsky -Lecture Notes*
<https://ocw.mit.edu/courses/res-6-007-signals-and-systems-spring-2011/pages/lecture-notes/>
- *Digital Image Processing - Books and Materials by Wilhelm Burger & Mark Burge*
<https://imagingbook.com/materials/>
- *Digital Image Processing, R. Gonzalez, Z. Faisal (2019)*
<https://www.researchgate.net/>

- *Basics of Image Processing — Vincent Mazet (Université de Strasbourg)*
<https://vincmazet.github.io/bip/index.html>

1.3 Software

- ImageJ
 - darmowe narzędzie do edycji obrazów w Javie
 - możliwość łatwej rozbudowy o nowe funkcjonalności
 - dostępny na stronie <https://ij.imagej.io/>
- Scilab + Image-Toolbox
 - darmowy język programowania, zorientowany macierzowo
 - liczne Toolboxy, w tym do przetwarzanie obrazu
 - dostępny na www.scilab.org
- MATLAB
 - profesjonalne środowisko programistyczne do ogólnych obliczeń naukowych
 - model dla wielu klonów (np. Scilab)
 - wiele narzędzi, w tym do zaawansowane narzędzia przetwarzanie obrazu
- Photoshop
 - komercyjny, profesjonalny program do edycji obrazów

1.4 Kursy online

- Hypermedia Image Processing Reference (HIPR) - Department of Artificial Intelligence in the University of Edinburgh
https://homepages.inf.ed.ac.uk/rbf/HIPR2/hipr_top.htm
- Basics of Image Processing - Vincent Mazet (Université de Strasbourg)
<https://vincmazet.github.io/bip/index.html>
- przetwarzanie obrazów w obszarze biologii i medycyny <https://bioimagebook.github.io/index.html>

2

Zmysł wzroku

2.1 Widzenie

- Oko ludzkie jako przykład systemu kamer
 - Anatomia i fizjologia oka
- Widzenie przez człowieka to przykład systemu przetwarzania obrazów (grafika 2.1)
- Iluzje optyczne pozwalają na wgląd w proces przetwarzania obrazów w mózgu
 - Ogromna wydajność ludzkiego układu wzrokowego ilustruje trudność i wymagania dla komputerowych systemów przetwarzania obrazów
 - ale także przedstawia sugestie dotyczące nowych rozwiązań

2.2 Anatomia oka i jej konsekwencje

2.2.1 Oko a kamera

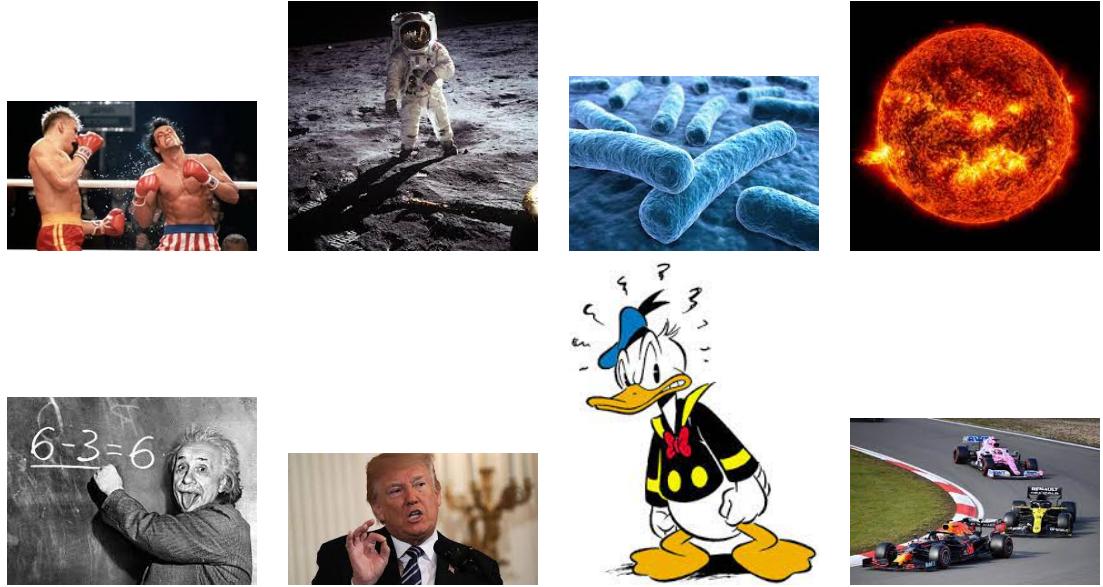
Rogówka, żrenica i soczewka oka, podobnie jak obiektyw aparatu, odpowiadają za ostrość obrazu (grafika 2.2).

2.2.2 Anatomia oka (siatkówka oka): reakcja na światło

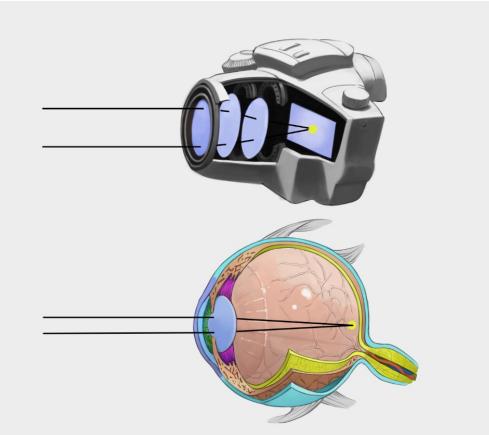
Pręciki – receptory odpowiadające za postrzeganie kształtów i ruchu.
Czopki (dawniej słupki) – receptory absorbujące światło (fotoreceptory) w różnych zakresach długości fali i umożliwiające widzenie kolorów
→ grafika 2.3

Pręciki

Pręciki są optymalnie stymulowane niebiesko-zielonym światłem ($\lambda = 500nm$) i przekazują percepcję odcieni szarości. → grafika 2.4



Grafika 2.1: Przykłady percepcji obrazu. Źródła grafik: sieć

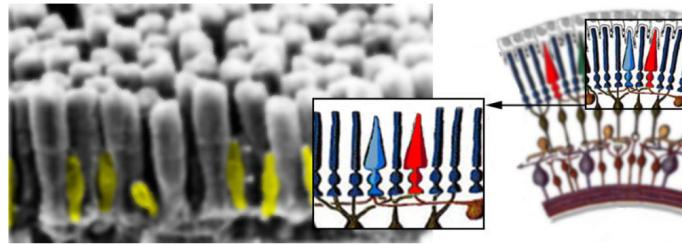


Grafika 2.2: Oko i kamera. Źródło: SchuBu Systems GmbH /Fabian Krenn

Fotoreceptory siatkówki oka

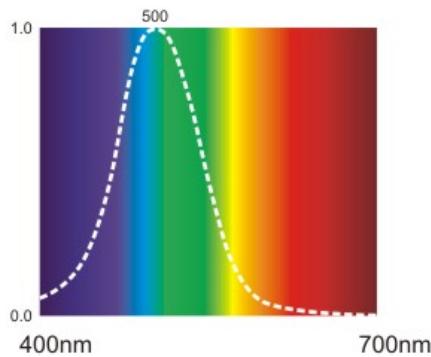
Ludzkie oko zawiera trzy rodzaje czopków, z których każdy ma inną charakterystykę:

- czopki S (short wavelength) są najbardziej czułe na promieniowanie o długości fali $\lambda = 420nm$ (wrażenie barwy niebieskiej),
- czopki M (medium wavelength) są najbardziej czułe na promieniowanie o długości fali $\lambda = 540nm$ (wrażenie barwy zielonej),
- czopki L (long wavelength) są najbardziej czułe na promieniowanie o długości fali $\lambda = 560nm$ (wrażenie barwy czerwonej/zółtej)



Grafika 2.3: Pręciki (duże) i czopki (małe i zabarwione na żółto) na siatkówce rysia - mikrofotografia elektronowa.

Źródło: <http://www.sinnesphysiologie.de/hvsinne/auge/rodcone.htm>



Grafika 2.4: Znormalizowana krzywa czułości przęcików.

Źródło: <http://www.dma.ufg.ac.at>

Znormalizowane krzywe absorpcji (krzywe czułości) trzech typów czopków opisują względne wzbudzenie receptorów padającym światłem o określonej długości fali. Najwyższej wartości wzbudzenia przypisuje się 100% lub 1.0 (= maksymalne pobudzenie), a wszystkim innym długościom fali przypisuje się odpowiednie wartości procentowe. → grafika 2.5 a)

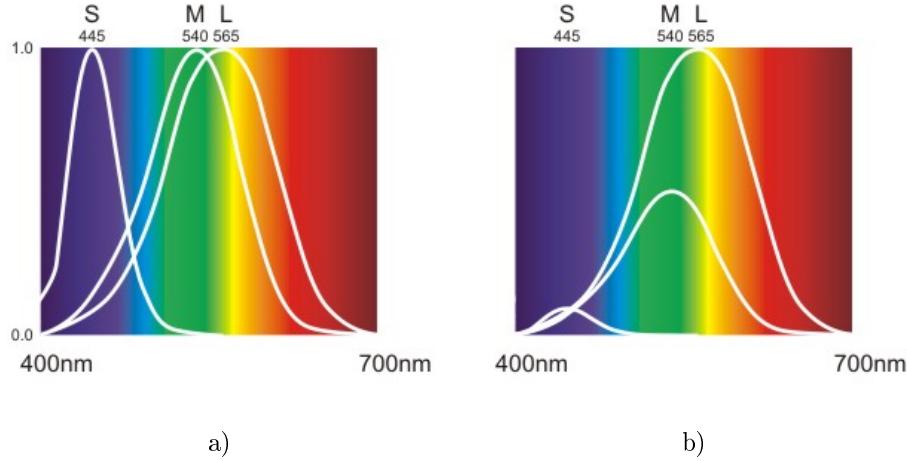
Należy zauważyć, że krzywe te nie umożliwiają porównania czułości bezwzględnej trzech typów czopków. Dla porównania ważne jest, że w przeliczeniu na całkowitą liczbę czopów jest 63% czopków L, 31% czopków M i tylko 6% czopów S. W rezultacie otrzymujemy się ważone krzywe porównawcze → grafika 2.5 b).

2.2.3 Teorie dotyczące widzenia barwnego

Teoria trójchromatyczna

Teoria Younga-Helmholtza (trójskładnikowa teoria widzenia barwnego):

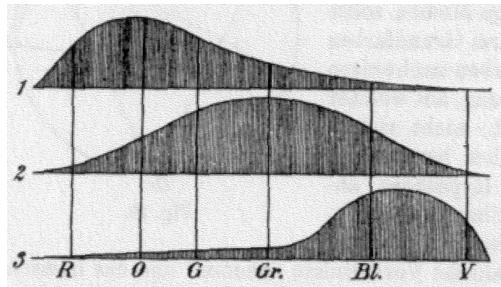
- widzenie barwne dzięki obecności w siatkówce oka człowieka trzech różnych fotoreceptorów, absorbujących światło widzialne w różnych zakresach długości fali promieniowania elektromagnetycznego.



Grafika 2.5: Czopki ludzkiego oka: a) względna absorbcja światła b) względna absorbcja światła ważona zgodnie z rozkładem procentowym Źródło: <http://www.dma.ufg.ac.at>

- teoria sformułowana przez Thomasa Younga w roku 1802 i rozbudowana w roku 1852 przez Hermanna Helmholtza.

→ grafika 2.6



Grafika 2.6: Thomas Young i Hermann Helmholtz postulowali trzy typy włókien nerwowych w oku
Źródło: Hermann Helmholtz: Gesammelte Schriften III.2, Olms-Weidmann, 2003, Fig. 21.

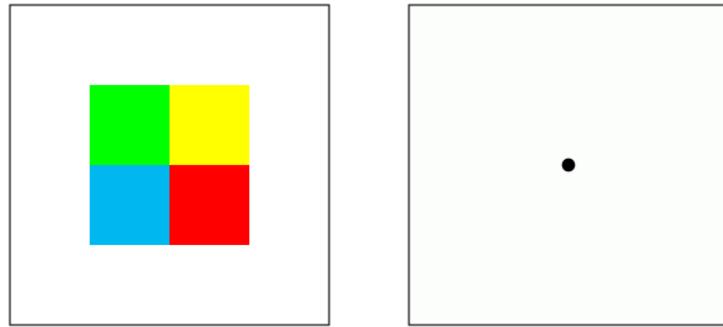
Teoria farb przeciwnych

Teoria Heringa, teoria oponentów, teoria przeciwkolorów:

- opublikowana w latach 1874 (1878) przez Ewalda Heringa jako alternatywa dla teorii Younga i Helmholtza

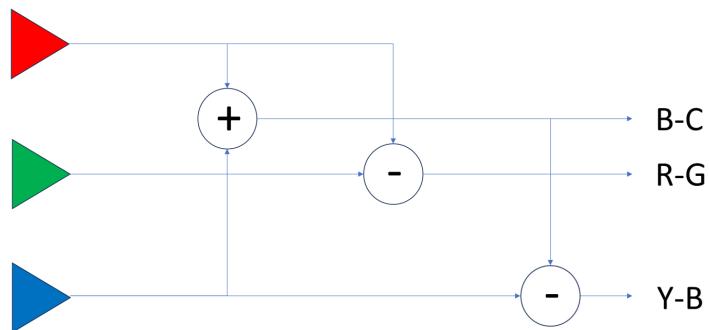
- miała tłumaczyć fenomeny w procesie widzenia, które nie dają się wytłumaczyć teorią trójchromatyczną, np.:

– kontrast następczy (powidok, obraz następczy) —> grafika 2.7



Grafika 2.7: Kontrast następczy

- nie można sobie wyobrazić mieszanych kolorów „żółto-niebieskiego” lub „czerwonawozielonego”.
- Hering postulował, że w widzeniu kolorów współdziałają trzy przeciwnostawne procesy: proces czerwono-zielony, proces żółto-niebieski i proces biało-czarny —> grafika 2.8



Grafika 2.8: Teoria farb przeciwnych

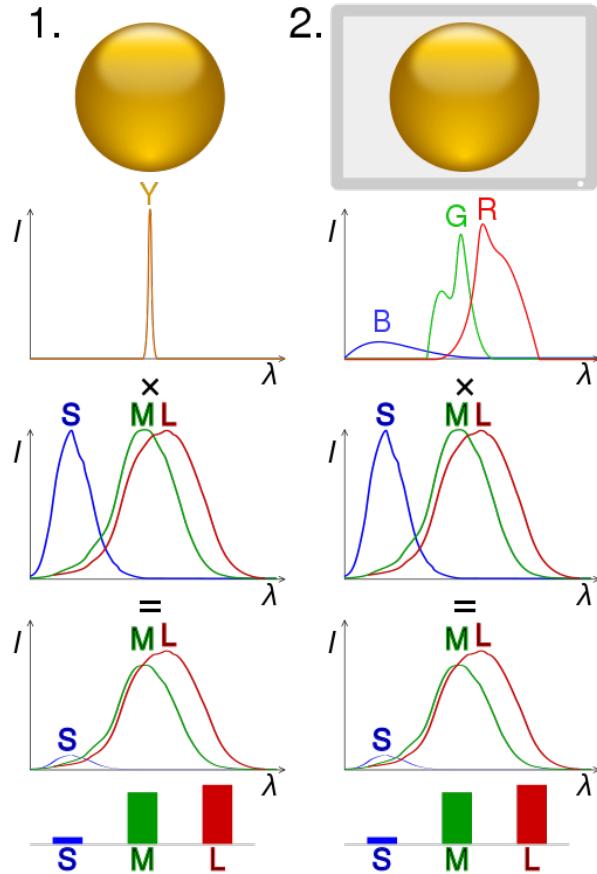
Obecnie uważa się, że obie teorie stanowią dwa etapy analizy kolorów w procesie widzenia.

2.2.4 Kolorimetria (barwometria)

—> ilościowy opis i charakterystyka barw postrzeganych przez człowieka

Metamery

Istnieją różne widma (mieszanki światła różnych długości fal, rozkłady intensywności), które powodują tę samą percepcję kolorów. Takie widma nazywane są metamerami (grafika 2.9). Ogólne wnioski:



Grafika 2.9: 1. kula oświetlona światłem monochromatycznym: mnożenie widma przez krzywe czułości czopków daje odpowiedź dla każdego typu czopków. 2 symulacja sceny z niebieskimi, zielonymi i czerwonymi diodami LED.

Źródło: <https://en.wikipedia.org/wiki/Metamerism>

- Możliwe jest, że inny skład widmowy światła będzie postrzegany jako dokładnie taki sam (metamery).
- Możliwe jest, że dwie kolorowe powierzchnie wyglądają tak samo w jednym rodzaju światła (np. świetle dziennym), ale prowadzą do innego postrzegania kolorów w innym rodzaju światła (np. świetle sztucznym) i dlatego wyglądają inaczej.

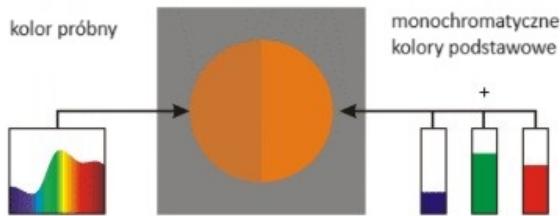
2.2.5 CIE XYZ

W 1931 roku Międzynarodowa Komisja Oświetleniowa (Commission Internationale de l'Eclairage, CIE) opracowała po raz pierwszy standardowy system kolorymetrii do opisu i klasyfikacji wszystkich dostrzegalnych kolorów, który opiera się na eksperymetach percepcyjnych w zakresie dopasowywania kolorów.

Eksperiment dopasowywania kolorów

→ grafika 2.10

W eksperymetach CIE obserwatorom zaprezentowano oświetlone kolorowe obsza-



Grafika 2.10: Eksperiment dopasowania kolorów (CIE).

Źródło: <http://www.dma.ufg.ac.at/app/link/Grundlagen3AAllgemeine/module/16457>

ry na neutralnym tle. Lewą połowę kolorowego koła oświetlono próbką koloru (albo światłem monochromatycznym, albo dowolną mieszaniną światła). Badani mieli za zadanie stworzyć metameryczną mieszaninę światła w prawej połowie kolorowego koła z trzech precyzyjnie określonych monochromatycznych kolorów podstawowych, która dokładnie odpowiadała próbce koloru.

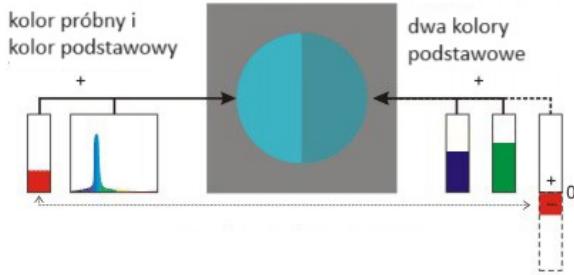
Eksperymenty wykazały, że nie wszystkie kolory widmowe i mieszaniny kolorów można mieszać bezpośrednio z wybranych kolorów podstawowych. Czasami w celu dopasowania koloru konieczne było zmieszanie próbki koloru z jednym z trzech kolorów podstawowych, aby uzyskać dopasowanie z mieszaniną pozostałych dwóch kolorów podstawowych. Oznacza to, że niektóre kolory można mieszać z trzema kolorów podstawowych tylko wtedy, gdy jeden z kolorów podstawowych ma „udział ujemny”.

→ grafika 2.11

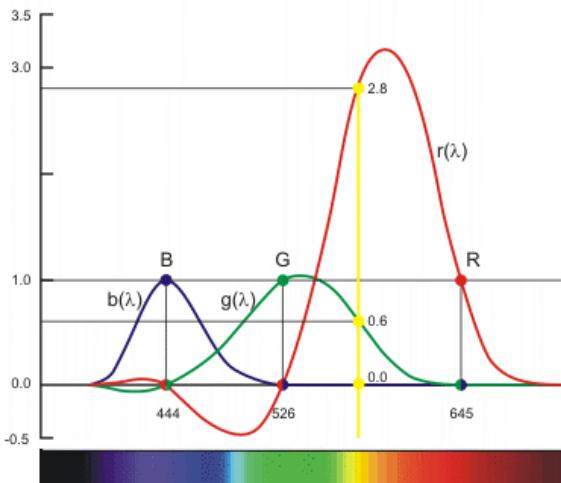
Funkcje wartości widmowych (*Color Matching Functions CMF*)

Eksperymenty z dopasowywaniem kolorów dla wszystkich kolorów widmowych o tej samej intensywności, w oparciu o (dowolnie) wybrane kolory podstawowe R ($\lambda = 645nm$), G ($\lambda = 526nm$) i B ($\lambda = 444nm$) daje trzy funkcje wartości widmowych $r(\lambda)$, $g(\lambda)$ i $b(\lambda)$. Funkcje wartości widmowych pokazują udział odpowiednich kolorów podstawowych w mieszaninie dla każdego koloru widmowego, przy czym wybrane natężenie początkowe przyjmuje się jako wartość odniesienia (= 1,0) dla kolorów podstawowych.

→ grafika 2.12



Grafika 2.11: Ujemna proporcja koloru podstawowego podczas dopasowywania kolorów.
Źródło: <http://www.dma.ufg.ac.at/app/link/Grundlagen3AAAllgemeine/module/16457>



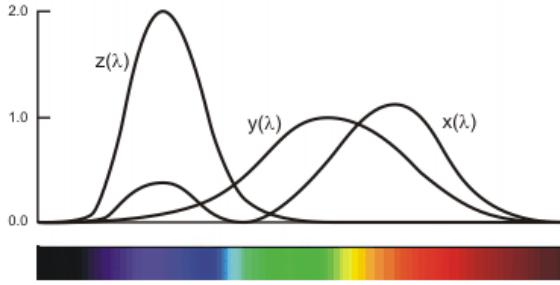
Grafika 2.12: Funkcje wartości widmowych CIE 1964.
Źródło: <http://www.dma.ufg.ac.at>

Przykład: $\lambda = 500$

$$(R, G, B) = (-0.07173, \quad 0.08536, \quad 0.04776)$$

Trójchromatyczne składowe widmowe

Aby otrzymać nieujemne funkcje wartości widmowych z funkcji wartości widmowych $r(\lambda)$, $g(\lambda)$ i $b(\lambda)$ poprzez transformację współrzędnych wprowadzone zostały trójchromatyczne składowe widmowe $x(\lambda)$, $y(\lambda)$ i $z(\lambda)$. $x(\lambda)$, $y(\lambda)$ i $z(\lambda)$ opierają się na abstrakcyjnych kolorach podstawowych X , Y i Z , które nie odpowiadają żadnym rzeczywistym kolorom. — grafika 2.13



Grafika 2.13: Trójchromatyczne składowe widmowe CIE 1964.
Źródło: <http://www.dma.ufg.ac.at>

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 2.768892 & 1.751748 & 1.130160 \\ 1 & 4.590700 & 0.060100 \\ 0 & 0.056508 & 5.594292 \end{pmatrix} \cdot \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

Przekształcenia zostały tak dobrane, że

- funkcja wartości widmowej $y(\lambda)$ odpowiada krzywej percepcji jasności w widzeniu fotopowym (widzenie prążkami), a współrzędna Y opisywanego koloru opisuje jednocześnie jasność (luminancję) koloru,
- przy równej mieszaninie kolorów podstawowych X , Y i Z uzyskuje się barwę światła białą.

Przykład: $\lambda = 500$

$$\begin{aligned} (R, G, B) &= (-0.07173, 0.08536, 0.04776) \\ (X, Y, Z) &= (0.0049, 0.3230, 0.2720) \end{aligned}$$

Trójkąt barw

Normalizacja wartości (X, Y, Z) prowadzi do diagramu xy (standardowa tabela kolorów CIE)

$$\begin{aligned} x &= \frac{X}{X + Y + Z} \\ y &= \frac{Y}{X + Y + Z} \end{aligned}$$

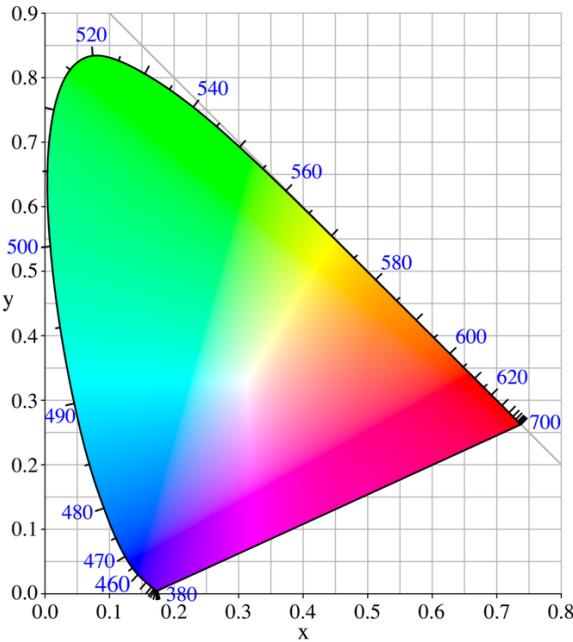
Przykład: $\lambda = 500$

$$\begin{aligned} (R, G, B) &= (-0.07173, 0.08536, 0.04776) \\ (X, Y, Z) &= (0.0049, 0.3230, 0.2720) \\ (x, y) &= (0.0082, 0.5384) \end{aligned}$$

Każdemu kolorowi przypisane jest jednoznaczne miejsce w trójkącie barw CIE. Kolor biały ma współrzędne $(\frac{1}{3}, \frac{1}{3})$.

→ grafika 2.14

Podobszar kolorów, który można technicznie przedstawić za pomocą RGB, to obszar trójkątny z lokalizacjami kolorów 3 źródeł światła jako punktami narożnymi. Przy



Grafika 2.14: Trójkąt barw CIE XYZ. Źródło: wikipedia

każdym wyborze źródła światła istnieją w przyrodzie kolory, które leżą poza trójkątem i dlatego w zasadzie nie można ich przedstawić.

Przykład

- Wszystkie kolory, które można zmieszać z dwóch kolorów widmowych o długości fali 420 nm i 530 nm, leżą na linii łączącej ich lokalizacje w trójkącie barw.
→ grafika 2.15
- Wszystkie kolory, które można mieszać z kolorów *R*, *G* i *B* (rogi trójkąta), leżą w pokazanym trójkącie. → grafika 2.16

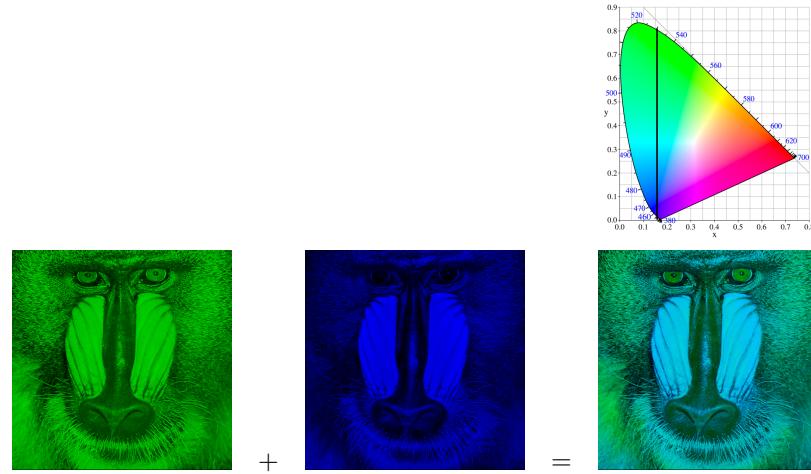
Przykład: filtr RGB wyświetlacza LCD

→ grafika 2.17.

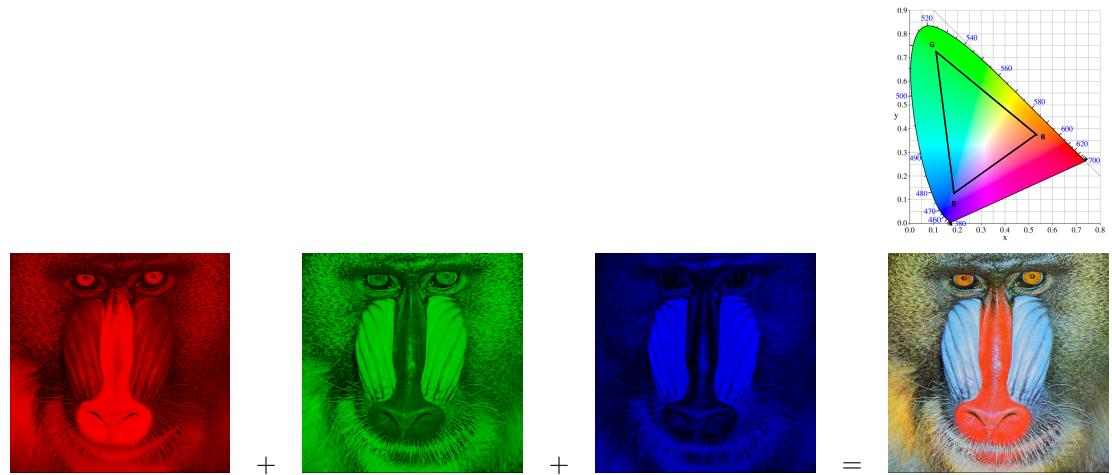
Przykład: filtry kolorów w aparatach cyfrowych

W aparatach cyfrowych stosowane są dwa typy filtrów:

- podstawowy filtr kolorów *RGB* składający się z trzech kolorów (czerwony, zielony i niebieski) (grafika 2.18)
- filtr kolorów uzupełniających *CMYG* składający się z czterech kolorów (cyjan, magenta, żółty i zielony) (grafika 2.19).



Grafika 2.15: Mieszanie barw.

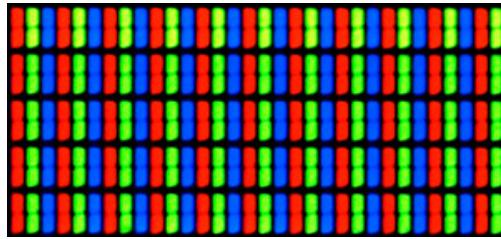


Grafika 2.16: Mieszanie barw - monitor RGB.

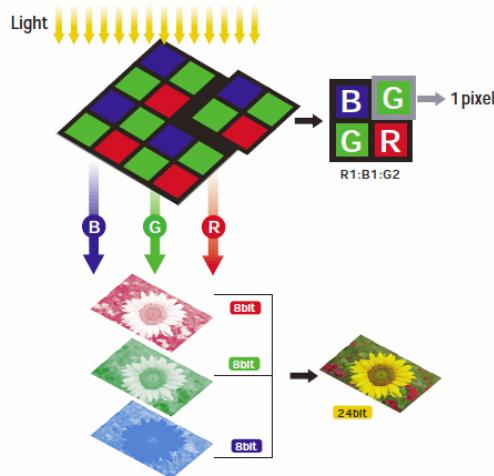
Wybór przestrzeni barw w aparatach cyfrowych → grafika 2.20

2.2.6 CIELab

- przestrzeń barw zdefiniowana jako odniesienie kolorystyczne niezależne od urządzenia
- model przestrzeni barw Lab opiera się między innymi na teorii przeciwkoloru Heringa



Grafika 2.17: Filtr RGB. Źródło: wikipedia



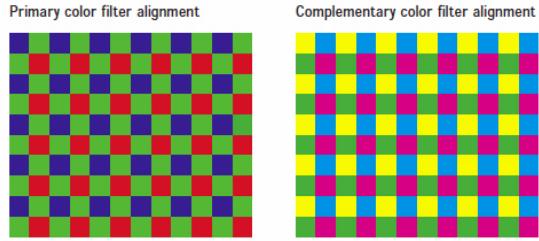
Grafika 2.18: Filtr RGB aparatu cyfrowego. Źródło: materiały firmy Panasonic

- norma ISO/CIE 11664-4 „Colorimetry – Part 4: CIE 1976 L*a*b* Colour space“

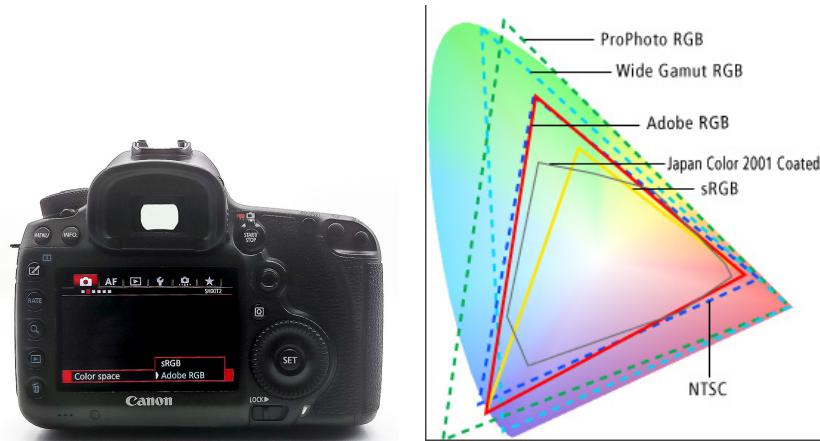
Zasadniczym wymogiem kolorymetrii dla modelu kolorów jest prawidłowe odwzorowanie podobieństw kolorów lub różnic w przestrzeni barw. Oznacza to, że odległość pomiędzy dwoma punktami barw w przestrzeni barw powinna być miarą dostrzegalnego podobieństwa lub różnicy kolorami -> ta sama odległość chrominancji odpowiada tej samej różnicy w postrzeganiu barw. Jeśli na przykład różnica między trzema kolorami zostanie postrzegana jako taka sama w porównaniu percepcyjnym, wówczas trzy punkty kolorów w modelu kolorów powinny znajdować się w tej samej odległości od siebie. Nie jest tak jednak w przypadku modelu Yxy, gdzie zielone odcienie zajmują więcej miejsca niż czerwony, pomarańczowy i żółty. W przykładzie (→ grafika 2.21) dwa połączone punkty w zielonym obszarze są postrzegane jako znacznie bardziej podobne niż pozostałe dwie pary punktów, mimo że są od siebie tak samo oddalone w trojkącie barw.

Aby rozwiązać ten problem przestrzeni barw CIE Yxy , została ona przekształcona w przestrzeń barw CIE Lab poprzez złożoną transformację matematyczną (znieksztalconie), która została wprowadzona w 1976 roku.

Jest to obecnie najbardziej znana i najczęściej używana przestrzeń kolorów oraz neutralny pod względem mediów, równomiernie odległy model opisujący kolor za pomocą



Grafika 2.19: Filtr kolorów uzupełniających CMYG aparatu cyfrowego. Źródło: materiały firmy Panasonic



Grafika 2.20: Wybór przestrzeni barw w aparatach cyfrowych.
Źródło: <https://global.canon/en/imaging/picturestyle/editor/matters04.html>

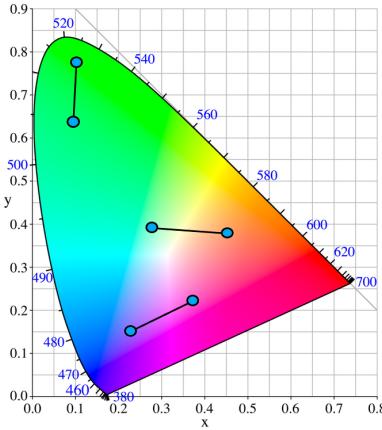
trzech wartości:

- L - współrzędna jasności (luminancja)
 L jest neutralną osią szarości, ponieważ wszystkie kolory achromatyczne (odcienie szarości) zawarte są pomiędzy punktami końcowymi: czarnym ($L = 0$) i białym ($L = 100$).
- a - wartość koloru czerwonego(+) i zielonego(-),
 $a \in [-170, 100]$
- b - to wartość żółtego(+) i niebieskiego(-),
 $b \in [-100, 150]$

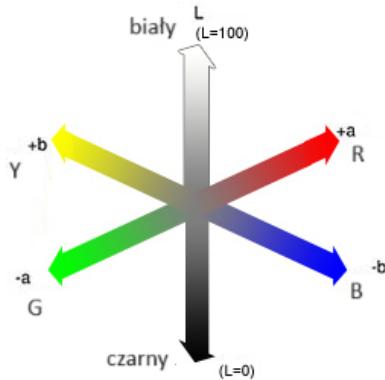
→ grafika 2.22

CIELab pozwala na porównanie podobieństw i różnic kolorów:

$$\Delta E = \sqrt{(\Delta L)^2 + (\Delta a)^2 + (\Delta b)^2}$$



Grafika 2.21: Odległości między barwami w przestrzeni CIExy.



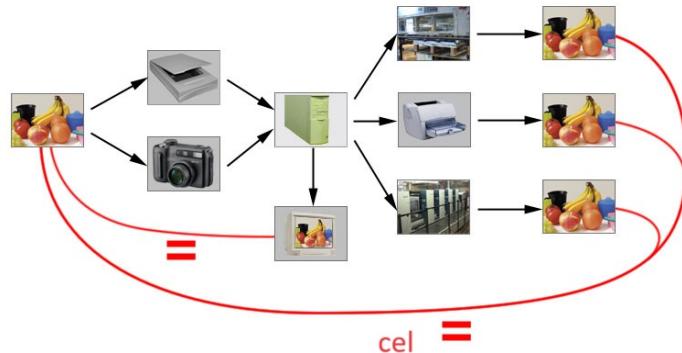
Grafika 2.22: Współrzędne barw w przestrzeni CIELab Źródło:
<http://www.dma.ufg.ac.at>

- odległość kolorów ΔE (odległość euklidesowa między dwoma kolorami) jest bezpośrednią miarą podobieństwa dwóch kolorów. Specjalista może rozpoznać różnicę kolorów $\Delta E = 1$, laik rozpoznaje około 2.5. Różnica w kolorze 6 – 7 jest nadal tolerowana, a od wartości $\Delta E = 10$ różnica jest bardzo wyraźna

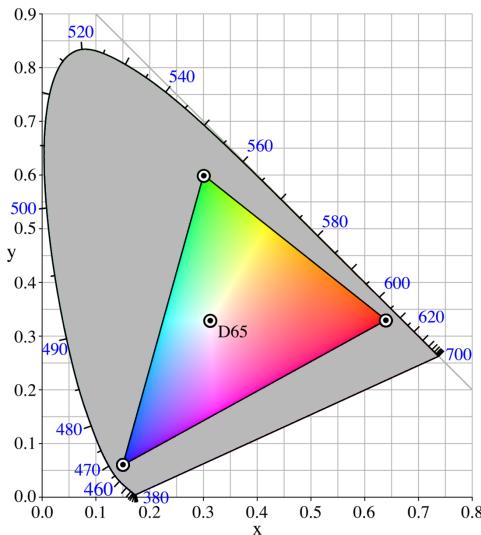
Paleta barw CIELab jest neutralną, niezależną od urządzenia technicznego paletą barw - dlatego służy jako etap pośredni poprawnemu dopasowaniu kolorów między urządzeniami (Colormanagement → grafika 2.23).

2.2.7 Model przestrzeni barw RGB

- Rzeczywisty zestaw kolorów podstawowych (np. sRGB) definiuje trójkąt kolorów wynikający z technologii produkcji wyświetlacza; tylko kolory w tym trójkącie można odtworzyć poprzez zmieszanie kolorów podstawowych. → grafika 2.24
- Z połączenia barw RGB w dowolnych kombinacjach można otrzymać szeroki



Grafika 2.23: Colormanagement. Źródło: <http://www.dma.ufg.ac.at>



Grafika 2.24: Trójkąt kolorów RGB. Źródło: sieć

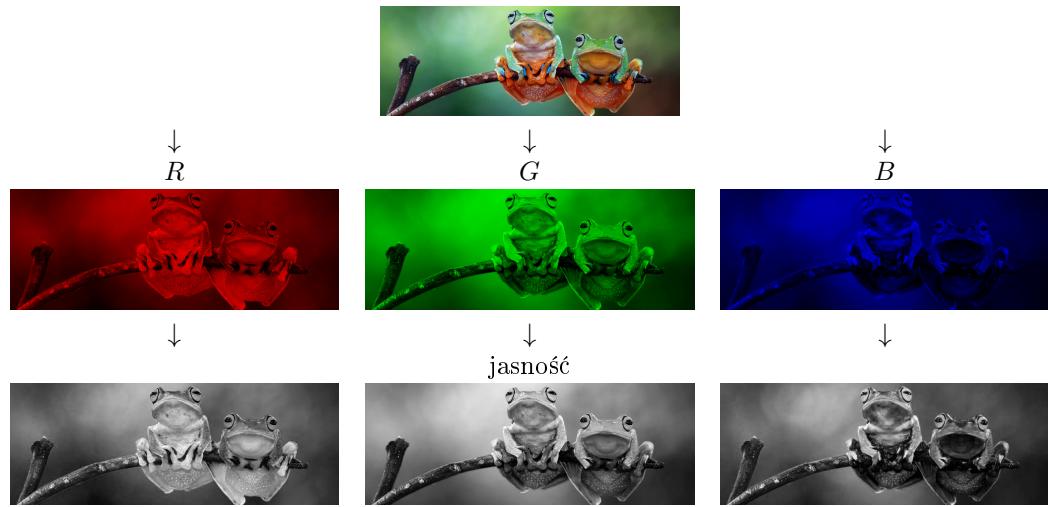
zakres barw pochodnych → grafika 2.25

2.2.8 Inne modele przestrzeni barw

Model CMYK

Model kolorów CMYK jest zależnym od urządzenia modelem subtraktynego mieszanego kolorów farb malarskich i drukarskich z podstawowych kolorów cyjan (cyan, C), magenta (magenta, M) i żółty (yellow, Y). Kolor czarny jest tradycyjnie nazywany *Key*. → grafika 2.26.

-
Przykład: → grafika 2.27



Grafika 2.25: Rozkład obrazu na kanały RGB.

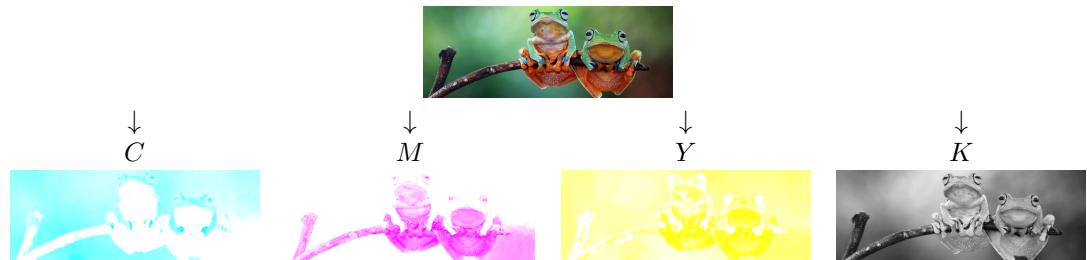


Grafika 2.26: Kolory druku: cyjan, magenta, żółty, czarny zgodnie z normą ISO 2846-1 (z CIE Lab przekonwertowane na RGB)

Model YCbCr

→ grafika 2.28

- Model przestrzeni kolorów YCbCr używany do cyfrowego przesyłania oraz przechowywania obrazów i wideo
(standard CCIR-601 do cyfrowego nagrywania obrazów i wideo, używany do obrazów JPEG, filmów MPEG, płyt DVD, Video CD i większości innych cyfrowych formatów wideo).
- Model opiera się między innymi na teorii przeciwkoloru Heringa.
- Model przestrzeni YCbCr wykorzystuje trzy typy danych:
 - Y – składową luminancji,
 - Cb – składową różnicową chrominancji Y-B, stanowiącą różnicę między luminancją a niebieskim,



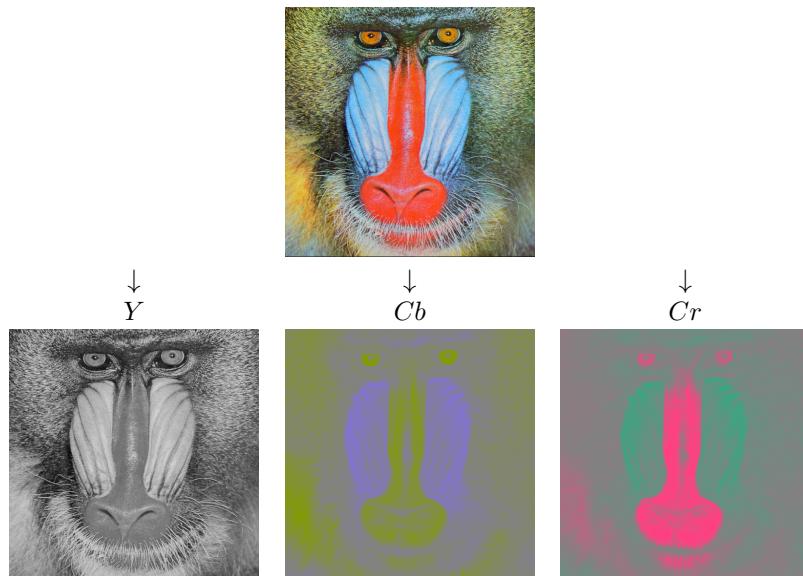
Grafika 2.27: Kanały CMYK obrazu (subtraktywne mieszanie kolorów)

- Cr – składową chrominancji Y-R, stanowiącą różnicę między luminancją a czerwonym.
- Kolor zielony jest uzyskiwany na podstawie tych trzech wartości.

Zasadniczo stosuje się schemat przeliczania zgodnie z standardem CCIR 601:

$$\begin{pmatrix} Y \\ Cb \\ Cr \end{pmatrix} \approx \begin{pmatrix} 0 \\ 128 \\ 128 \end{pmatrix} + \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.168736 & -0.331264 & 0.5 \\ 0.5 & -0.418688 & -0.081312 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

Po konwersji współrzędne YCbCr mają wartości w przedziale $\{0, 1, \dots, 255\}$.



Grafika 2.28: Model przestrzeni kolorów YCbCr

Przykład: zastosowanie w detekcji znaków znaków drogowych → grafika 2.29



Grafika 2.29: Detekcja znaków ograniczenia prędkości.

2.3 Widzenie jako filtrowanie obrazu

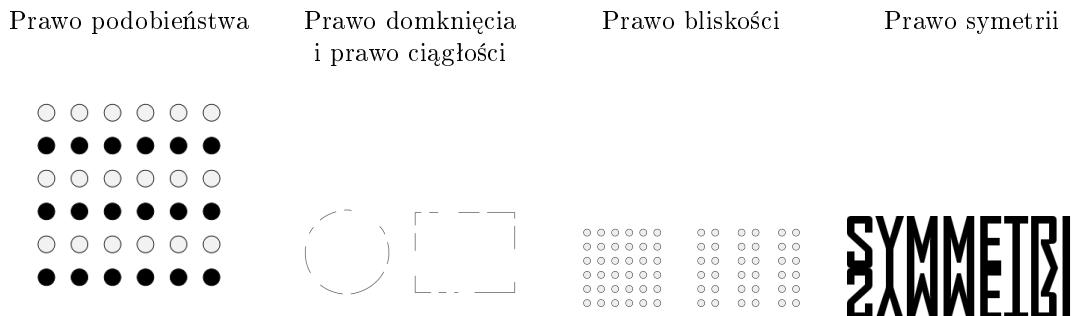
Funkcja filtra (teoria sygnałów i systemów):
przepuszczanie lub blokowanie sygnałów w określonym zakresie częstotliwości (grafika 2.30).

2.4 Przetwarzanie obrazu: ekstrakcja obiektów

→ grafika 2.31



Grafika 2.30: Filtrowanie obrazów. Źródło: sieć



Grafika 2.31: Niektóre prawa algorytmu ekstrakcji obiektów przez mózg

2.5 Przetwarzanie obrazów przez człowieka - zagadnienia

- Oko jest podobne do aparatu, ale co z dalszą obróbką i przechowywaniem?
 - Przesyłanie i przechowywanie jak w albumie fotograficznym w pełnej rozdzielczości?
- Nieświadome przetwarzanie wstępne
 - Ilość informacji strumienia obrazów w czasie rzeczywistym jest zbyt duża, aby umożliwić pełny transfer informacji i przechowywanie ich w mózgu.
 - Dlatego oko już redukuje i wstępnie przetwarza sygnały, tak że tylko ich część jest przekazywana do mózgu w celu faktycznego przetworzenia
Np. redukcja informacji poprzez sakady: Oko redukuje informacje w jednorodnych obszarach
 - Nie tylko redukcja informacji, ale także dodanie informacji w martwym punkcie
- Problemy spowodowane przetwarzaniem wstępny /redukcją informacji:

- Informacje w martwym punkcie nie są prawdziwe
Np. różne lub sprzeczne zeznania świadków
- Ludzki układ wzrokowy jest nieodpowiedni, by przechować fotograficzne szczegóły przez długi czas.

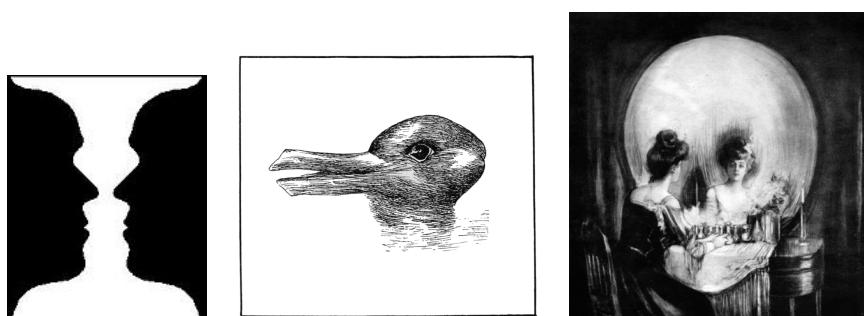
2.5.1 Złudzenia /iluzje optyczne

→ umożliwiają wgląd w procesy przetwarzania obrazu przez człowieka.

- To, co tak naprawdę widzimy, to rekonstrukcja obiektów dokonywana przez nasz umysł na podstawie danych wejściowych dostarczanych przez oczy, a nie rzeczywistego światła odbieranego przez nasze oczy.
- Badanie złudzeń optycznych pokazało, jakie nieświadome wnioski mogą zawierać się w procesie percepcyjnym.
- Inny rodzaj nieświadomego wnioskowania opiera się na postrzeganiu prawdopodobieństwa, które jest określane na podstawie częstotliwości podobnych już dokonanych spostrzeżeń.

Obrazy bistabilne

→ grafika 2.32



Grafika 2.32: Obrazy bistabilne

Możliwe wyjaśnienie:

- Mózg przechowuje modele obiektów, na które odwzorowywane są wrażenia wzrokowe.
- System wizualny preferencyjnie zakłada „ciągłość” pomiędzy zarejestrowanymi obiekty w sekwencjach.

Iluzje długości

→ grafika 2.33

Możliwe wyjaśnienie:

- Nasz mózg próbuje osadzić to, co widzimy, w przestrzeni 3D.



Grafika 2.33: Iluzje długości

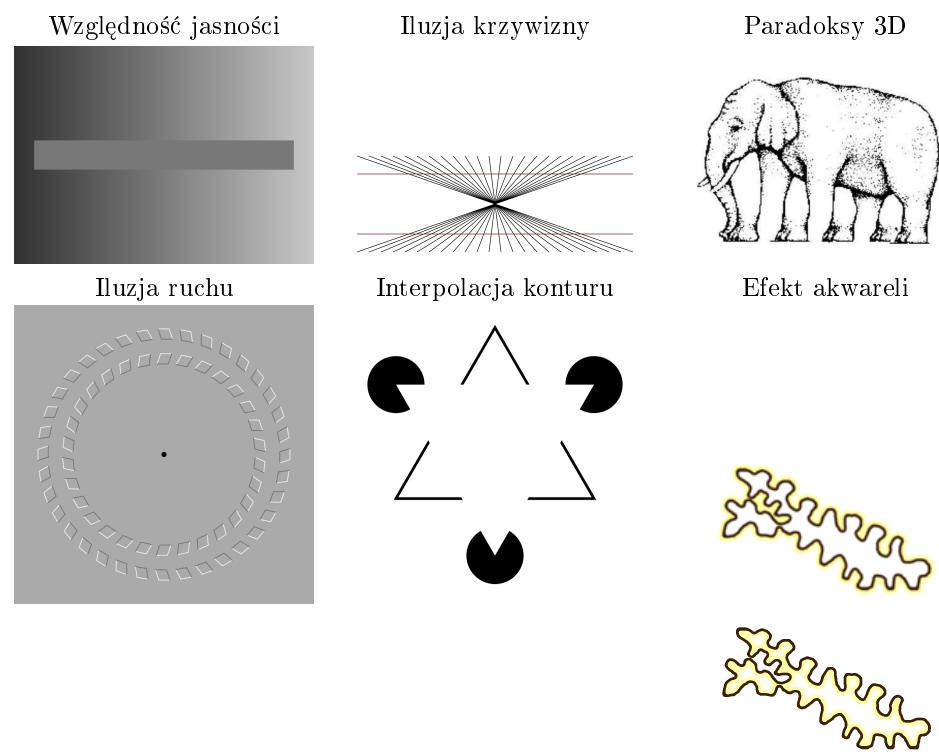
- Linie prowadzące ukośnie w górę są interpretowane jako prowadzące głębiej
→ górnego pasek jest bardziej cofnięty i dlatego jest postrzegany jako dłuższy.

Inne iluzje

→ grafika 2.34

2.6 Związek między widzeniem a maszynowym przetwarzaniem obrazu

- Ludzki wzrok może być traktowany jako system algorytmicznego przetwarzania danych
- Ludzie są słabi, jeśli chodzi o precyzyjne pytania pomiarowe: określenie wartości szarości, długości, szacowanie powierzchni.
- Ludzie są silni w rozpoznawaniu obiektów, rozpoznawaniu celu, rekonstrukcji 3D itp. nawet w przypadku brakujących części, „interpolacji granic obiektu”, rozpoznawaniu twarzy
- Widzenie maszynowe: bardzo skuteczne w prostych, dobrze zdefiniowanych zadaniach, zwłaszcza w zadaniach obiektywnych pomiarów.
- Naśladowanie odkryć fizjologicznych na potrzeby nowych podejść do przetwarzania obrazu jest owocne: piramidy obrazu, orientacja lokalna, ocena ruchu poprzez filtrowanie...

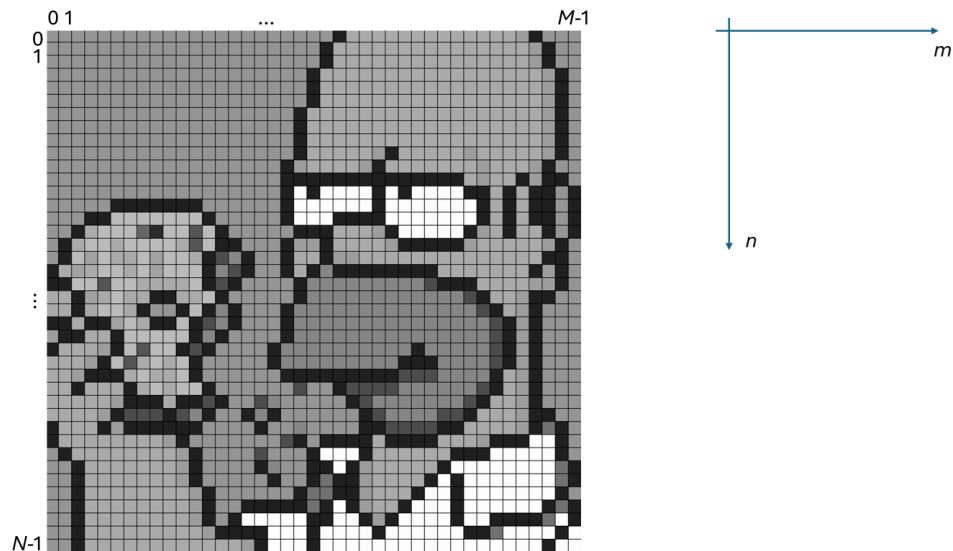


Grafika 2.34: Iluzje optyczne

3

Odtwarzanie obrazów

3.1 Reprezentacja obrazu w postaci cyfrowej



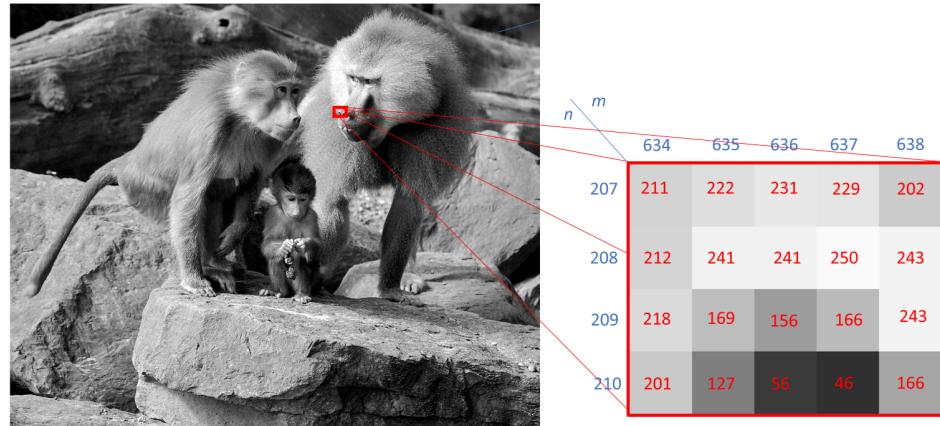
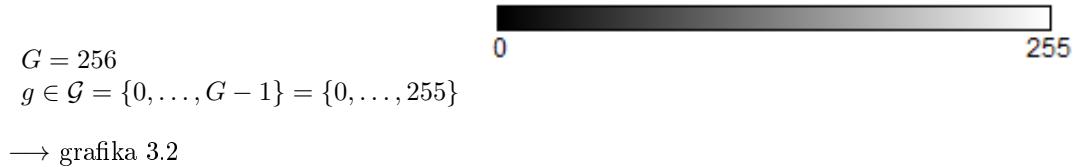
Grafika 3.1: Reprezentacja obrazu w postaci cyfrowej.

$$f : \mathcal{M} \times \mathcal{N} \rightarrow \mathcal{G}$$
$$f(m, n) = g$$

$$m \in \mathcal{M} = \{0, \dots, M - 1\}$$
$$n \in \mathcal{N} = \{0, \dots, N - 1\}$$

3.1.1 Obrazy w skali szarości

Zapis 8-bitowy



Grafika 3.2: Reprezentacja obrazu w postaci cyfrowej w 8-u bitach.

Zapis 32-bitowy

→ w specjalnych zastosowaniach (np. filtrowanie obrazu)

Przykład:

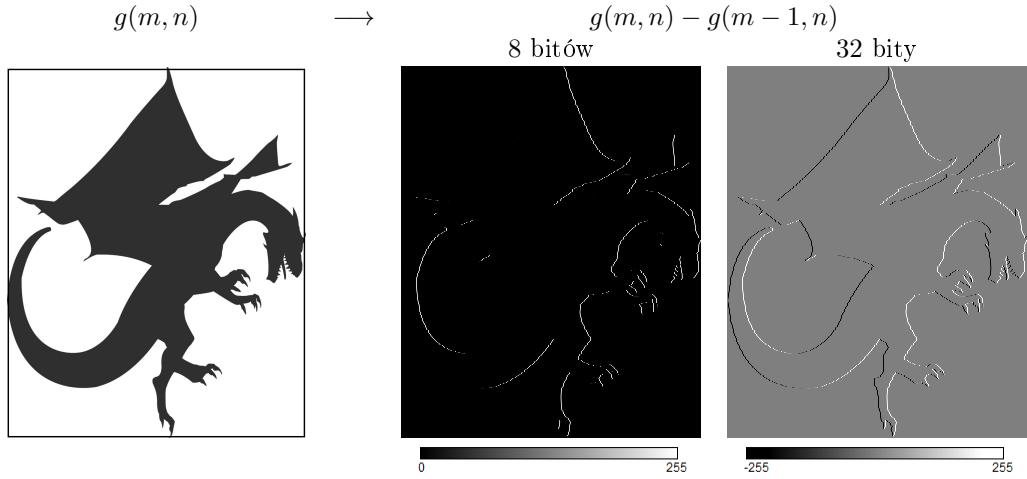
od każdej wartości szarości piksela w obrazie odejmowana jest wartość szarości lewego sąsiada

$$g(m, n) - g(m - 1, n)$$

→ grafika 3.3

3.1.2 Obrazy RGB

→ grafika 3.4



Grafika 3.3: Wyniki odejmowania $g(m, n) - g(m - 1, n)$ w reprezentacji 8 i 32-bitowej.

Zapis 24-bitowy

→ grafika 3.5

$$R, G, B \in \{0, 1, \dots, 255\}$$

$$\text{numer koloru} = 256^2 \cdot R + 256 \cdot G + B$$

→ liczba dostępnych kolorów $256^3 = 16\,777\,216$

Reprezentacja szesnastkowa barw RGB

→ grafika 3.6

Skala szarości w modelu RGB

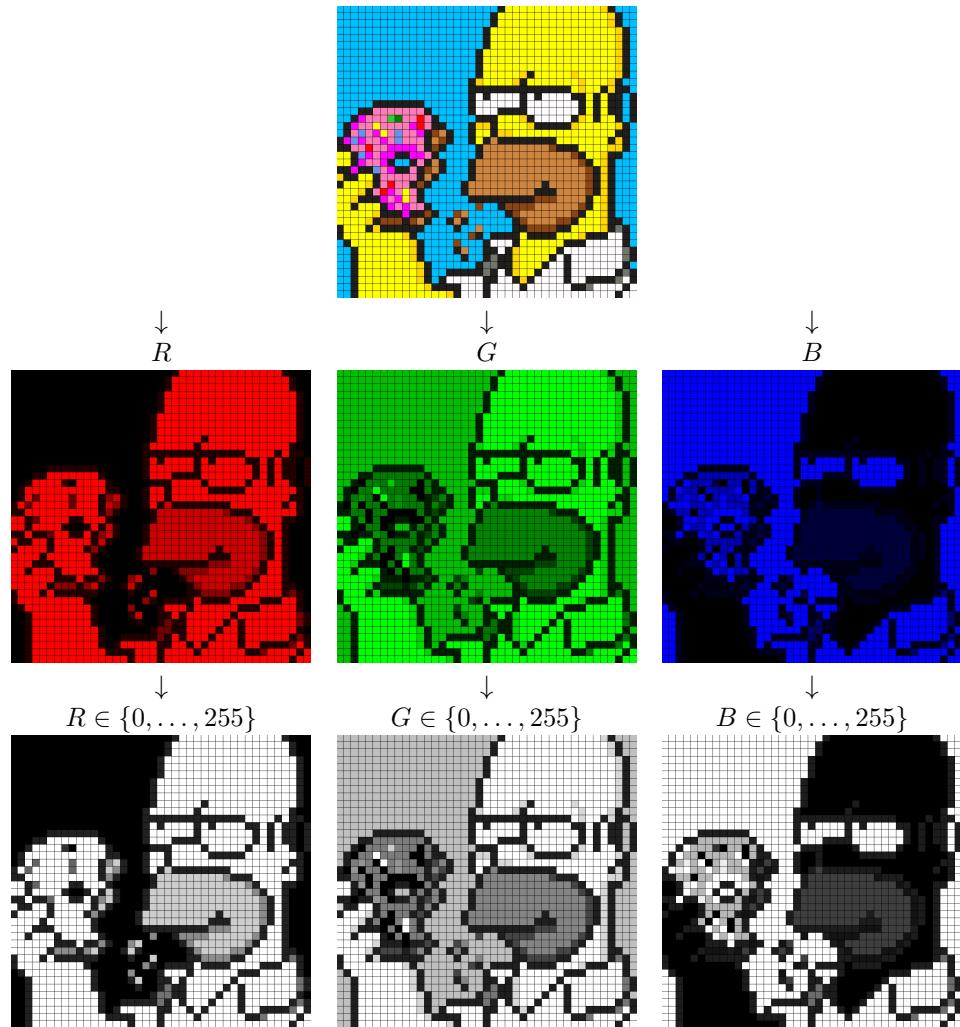
- Poziomy skali szarości osiąga się w modelu kolorów RGB poprzez połączenie wszystkich trzech wartości kolorów w równych proporcjach.
- Ustawienie wszystkich trzech wartości kolorów na 255 (255, 255, 255) odpowiada czystej bieli, a ustawienie wszystkich trzech wartości kolorów na 0 odpowiada czystej czerni.

Konwersja RGB w przetwarzaniu obrazu

W przetwarzaniu obrazu wartość szarości reprezentuje jasność lub intensywność pojedynczego piksela.

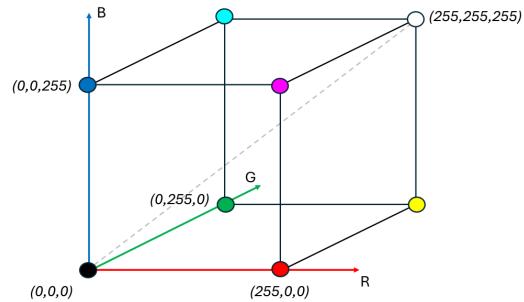
W przeliczeniu wartości koloru stosuje się wzór

$$\text{wartość szarości} = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B$$

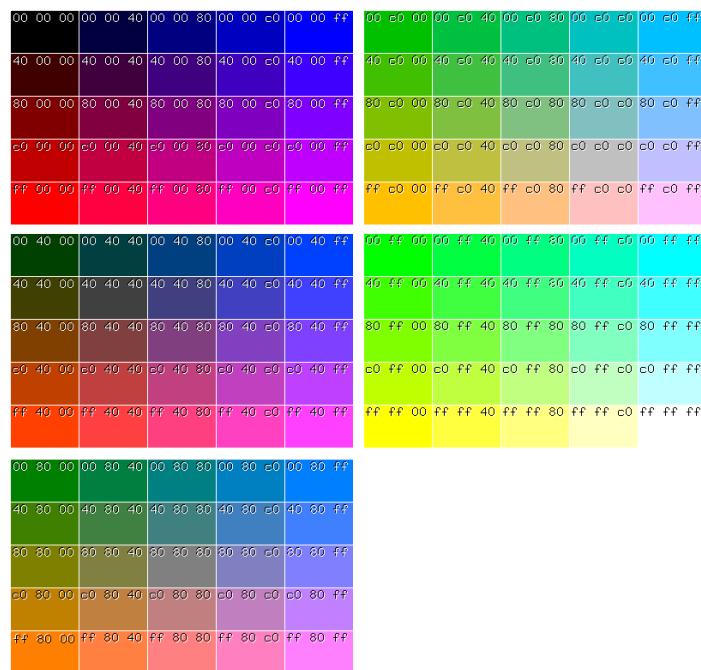


Grafika 3.4: Kanały RGB

Wynikiem działania jest wartość odzwierciedlająca jasność piksela niezależnie od kolorów. Procentowy rozkład wagi 3 kolorów jest powiązany z odpowiednią wrażliwością oczu na dany kolor. Jednak w praktyce przetwarzania obrazu rozkład ten jest czasami ignorowany i wszystkie trzy kanały kolorów mają jednakową wagę. → grafika 3.7



Grafika 3.5: Reprezentacja modelu kolorów RGB w sześciianie



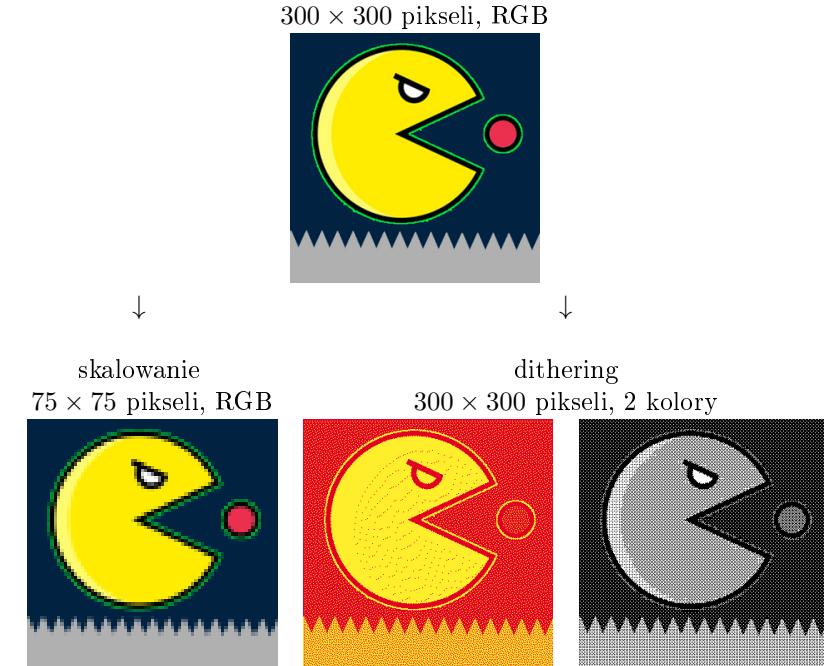
Grafika 3.6: Szesnastkowe, 8-bitowe reprezentacje RGB dla głównych 125 kolorów.
Źródło: sieć



Grafika 3.7: a) Obraz RGB, b) Obraz w kolorze szarym z ważeniem kanałów, c) Obraz w kolorze szarym bez ważenia kanałów.

Metody odtwarzania obrazów, gdy wymagana jest zmiana rozdzielczości i/lub palety barw

- zmiana rozdzielczości obrazu
→ skalowanie obrazu
 - zmiana palety barw (zmniejszenie liczby używanych kolorów)
→ dithering
- grafika 3.8



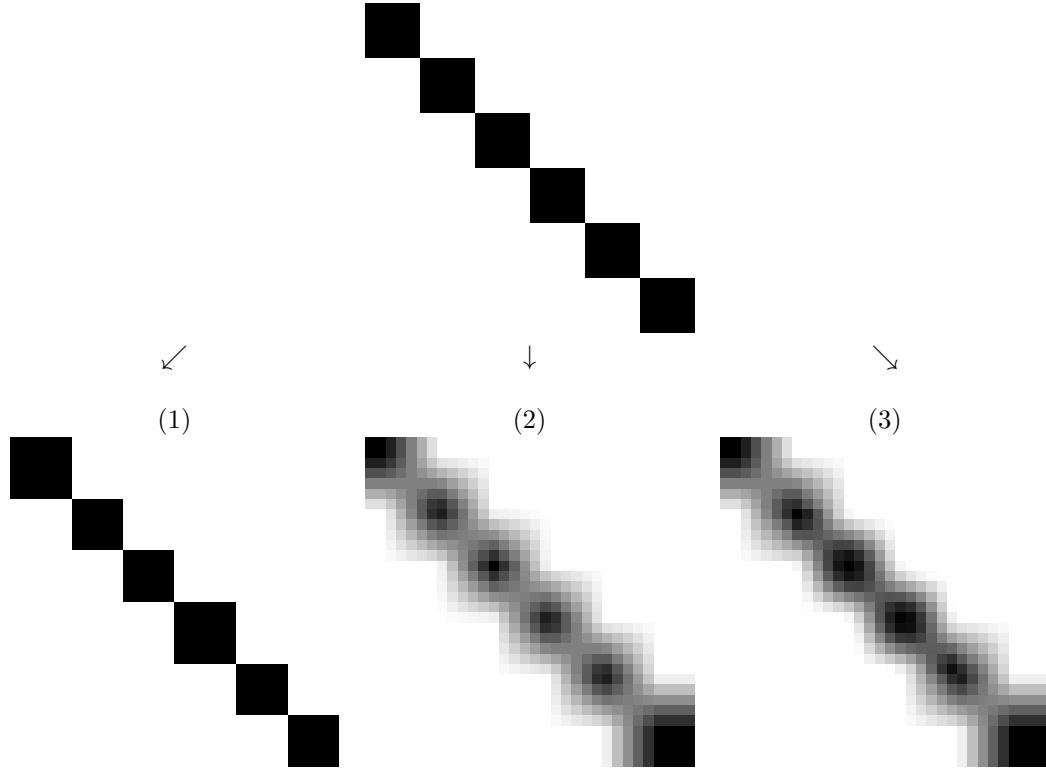
Grafika 3.8: Odtwarzanie obrazu ze zmianą rozdzielczości lub palety barw.

3.2 Skalowanie obrazów

Skalowanie zmienia rozdzielczość obrazu. Oznacza to, że z danej grafiki rastrowej tworzony jest nowy obraz o większej lub mniejszej liczbie pikseli. Zwiększenie liczby pikseli (upsampling) wiąże się zwykle z widoczną utratą jakości. Efekty zmniejszenia liczby pikseli w obrazie zostaną omówione w rozdziale "Digitalizacja obrazów". → grafika 3.9

Skalowanie z filtrem rekonstrukcyjnym

Programy do edycji obrazu zazwyczaj oferują kilka metod skalowania. Najczęściej obsługiwane metody – powtarzanie pikseli, interpolacja dwuliniowa i dwusześcienna – skalują obraz za pomocą filtra rekonstrukcyjnego.



Grafika 3.9: Skalowanie obrazu 6×6 do wymiarów 32×32 .

- (1) powtarzanie pikseli
- (2) interpolacja dwuliniowa
- (3) interpolacja dwusześcienna

Skalowanie można zobrazować poprzez umieszczenie siatki pikseli obrazu wyjściowego na siatce pikseli obrazu wejściowego. Każdemu pikselowi obrazu wyjściowego przypisana jest wartość koloru obliczana na podstawie pobliskich pikseli obrazu wejściowego. Zastosowany filtr rekonstrukcji określa, które piksele obrazu wejściowego zostaną użyte do obliczeń i jak ważone są ich wartości kolorów. → grafika 3.10

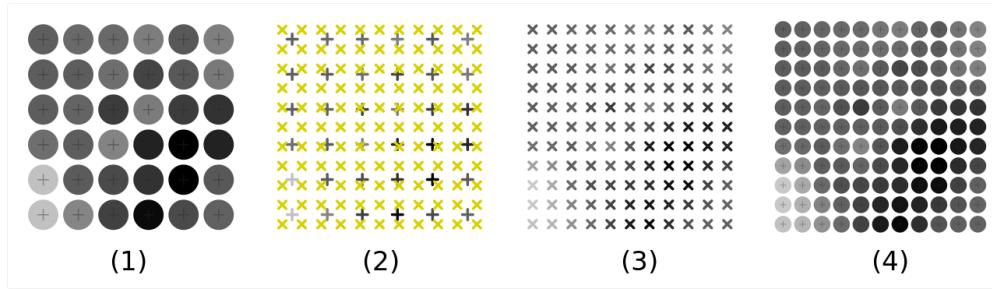
Podczas skalowania na każdym pikselu obrazu wyjściowego umieszczany jest dwuwymiarowy filtr rekonstrukcyjny. Wartość koloru oblicza się jako ważoną sumę wartości kolorów pikseli obrazu wejściowego w obszarze filtra rekonstrukcyjnego. → grafika 3.11

3.2.1 Powtórzenie pikseli (*Nearest neighbor*)

→ grafika 3.12

W przypadku powtarzania pikseli każdemu pikselowi obrazu wyjściowego przypisuje się wartość koloru najbliższego piksela obrazu wejściowego. Zmniejszanie obrazów przy użyciu tej metody może skutkować silnymi efektami aliasingu.

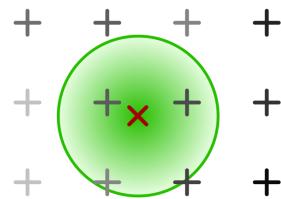
Podczas powiększania przy użyciu powtarzania pikseli pojawia się blokowy obraz „pik-



Grafika 3.10: Skalowanie obrazu o wymiarach 6×6 pikseli do rozmiarów 11×11 pikseli:
 (1) obraz wejściowy (piksele są tutaj pokazane jako koła)

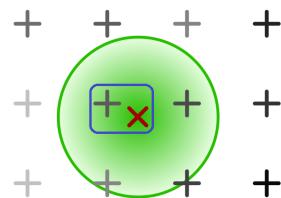
- (2) Siatka pikseli obrazu wyjściowego (żółte krzyżyki) nałożona na obraz wejściowy.
- (3) Wartości kolorów obrazu wyjściowego są obliczane na podstawie pobliskich pikseli obrazu wejściowego.
- (4) Obraz wyjściowy.

Źródło: [https://de.wikipedia.org/wiki/Skalierung_\(Computergrafik\)](https://de.wikipedia.org/wiki/Skalierung_(Computergrafik))



Grafika 3.11: Radialny filtr rekonstrukcyjny.

Źródło: [https://de.wikipedia.org/wiki/Skalierung_\(Computergrafik\)](https://de.wikipedia.org/wiki/Skalierung_(Computergrafik))



Grafika 3.12: Nearest neighbor.

Źródło: [https://de.wikipedia.org/wiki/Skalierung_\(Computergrafik\)](https://de.wikipedia.org/wiki/Skalierung_(Computergrafik)) (edytowane)

selowany". → grafika 3.13

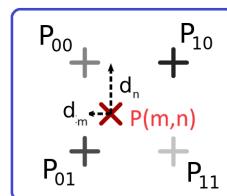


Grafika 3.13: Przykład: powtórzenie pikseli.

3.2.2 Interpolacja dwuliniowa (*Bilinear filtering*)

→ grafika 3.14

Wartość koloru piksela wyjściowego $P(m, n)$ oblicza się ze wzoru:



Grafika 3.14: Interpolacja dwuliniowa.

Źródło: [https://de.wikipedia.org/wiki/Skalierung_\(Computergrafik\)](https://de.wikipedia.org/wiki/Skalierung_(Computergrafik)) (edytowane)

$$\begin{aligned} Q_0 &= (1 - d_m)P_{00} + d_mP_{10} \\ Q_1 &= (1 - d_m)P_{01} + d_mP_{11} \\ P(m, n) &= (1 - d_n)Q_0 + d_nQ_1 \end{aligned}$$

Przykład: grafika 3.15



Grafika 3.15: Przykład: interpolacja dwuliniowa.

3.2.3 Interpolacja dwusześcienna

W przypadku interpolacji dwusześciennej wartość koloru obrazu wyjściowego jest interpolowana z sąsiednich wartości kolorów obrazu wejściowego za pomocą splajnów sześciennych. Istnieje kilka typowych splajnów sześciennych o różnych właściwościach, dlatego oznaczenie „interpolacja dwusześcienna” jest niejednoznaczne.

Przykład: grafika 3.16

3.2.4 Inne algorytmy skalowania

Zniekształcenie obrazu zależne od treści

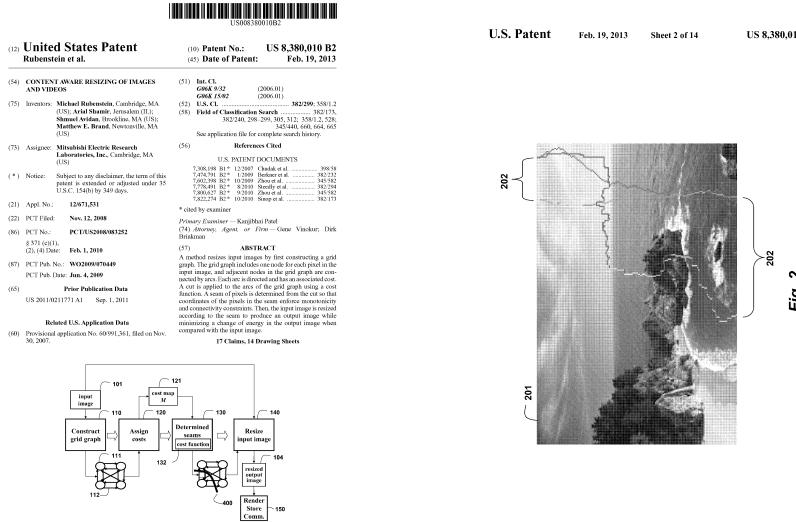
→ algorytm tworzy ocenę usuwalności dla wszystkich obszarów/pikseli obrazu, a następnie wykorzystuje ją do wyszukiwania ścieżek dla możliwie nieistotnych pikseli, które biegą prostopadle do kierunku kompresji lub rozciągania, a następnie wstawia lub usuwa piksele w tych szwach (*seams* → *seam carving*). → grafika 3.17
Zastosowanie: np. Photoshop CS4 (Funktion "Content Aware Scaling").

Skalowanie grafiki pikselowej (Pixel-Art)

Istnieją specjalne algorytmy umożliwiające powiększanie obrazów pikselowych o twardych krawędziach, które zapewniają lepsze wyniki w przypadku tego typu obrazów niż metody opisane powyżej. → grafika 3.18



Grafika 3.16: Przykład: interpolacja dwusześcienna.



Grafika 3.17: Zniekształcenie obrazu zależne od treści. Patent US8380010B2

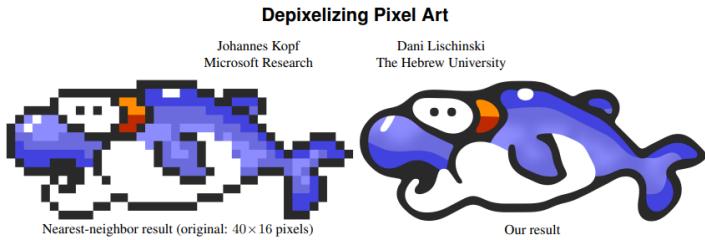


Figure 1: Native upsampling of pixel art images leads to unsatisfactory results. Our algorithm extracts a smooth, resolution-independent vector representation from the image, which is suitable for high-resolution display devices. (Input image © Nintendo Co., Ltd.).

Abstract

We describe a novel algorithm for extracting a resolution-independent vector representation from *pixel art* images, which enables magnifying the results by an arbitrary amount without image degradation. Our algorithm resolves pixel-scale features in the input and converts them into regions with smoothly varying shading that are crisply separated by piecewise-smooth contour curves.

In the original image, pixels are represented on a coarse grid but higher resolution artwork. For this reason, classical pixel art is usually marked by an economy of means, minimalism, and inherent modesty, which some say is lost in modern computer graphics. The best pixel art from the golden age of video games are masterpieces, many of which have become cultural icons that are instantly recognized by a whole generation, e.g. “Space Invaders” or the 3-color Super Mario Bros. sprite. These video games continue to be enjoyed today, thanks to numerous emulators that were developed to

Grafika 3.18: Fragment artykułu J. Kopf, D. Lischinski: Depixelizing Pixel Art. ACM Transactions on Graphics 30

3.3 Dithering

Dithering (*to dither*, rozsiewanie, rozpraszać): technika stosowana w grafice komputerowej, mająca na celu stworzenie iluzji większej głębi kolorów w obrazach, które ze względu na ograniczenia techniczne muszą być renderowane ze zmniejszoną głębią kolorów (lub czarno-białe).

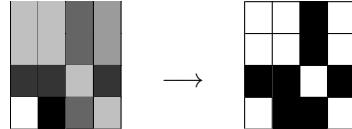
- Najczęstszym zastosowaniem ditheringu jest redukcja kolorów. (→ grafika 3.19)



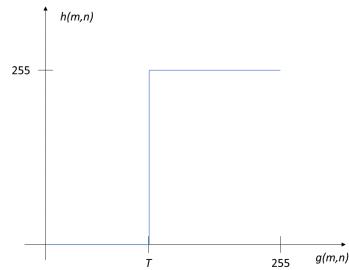
Grafika 3.19: Dithering

- W obrazie wynikowym brakujące kolory są odtwarzane przy użyciu określonego układu pikseli dostępnych kolorów, co pozwala uniknąć ostrzych przejść między kolorami. Ludzkie oko postrzega dithering jako mieszaninę poszczególnych kolorów.

3.3.1 Progowanie (*Thresholding*)



Wartości szarości $g(m, n)$ powyżej progu T są wyświetlane jako białe, poniżej progu jako czarne:



`if $g(m, n) < T$ then $h(m, n) = 0$ (black) else $h(m, n) = 255$ (white)`

→ grafika 3.20



Grafika 3.20: Progowanie obrazu

3.3.2 Algorytm Floyda-Steinberga

→ dyfuzja błędów (*error diffusion*), dostosowywanie obrazu do formatu zapisu lub medium o ograniczonej paletie barwnej (pliki w formacie GIF, niektóre drukarki atramentowe):

Błąd powstający podczas kwantyzacji (różnica między wartością początkową a wartością skwantowaną, tzn. wartością z dostępnej palety kolorów) każdego piksela jest rozdzielany na sąsiednie piksele według ustalonego schematu. Algorytm skanuje obraz

od lewej do prawej, od góry do dołu, kwantyzując wartości pikseli jeden po drugim. Za każdym razem błąd kwantyzacji jest przenoszony na sąsiednie piksele, nie wpływając przy tym na piksele, które zostały już skwantowane.

	P	7/16
3/16	5/16	1/16

Algorytm (w pseudokodzie):

```

for each y from top to bottom
    for each x from left to right
        oldpixel := pixel[x][y]
        newpixel := find_closest_palette_color (oldpixel)
        pixel[x][y] := newpixel
        quant_error := oldpixel - newpixel
        pixel[x+1][y] := pixel[x+1][y] + quant_error * 7 / 16
        pixel[x-1][y+1] := pixel[x-1][y+1] + quant_error * 3 / 16
        pixel[x][y+1] := pixel[x][y+1] + quant_error * 5 / 16
        pixel[x+1][y+1] := pixel[x+1][y+1] + quant_error * 1 / 16
    
```

Przykład:

redukcja palety szarości → grafika 3.21

Dany obraz $g(m, n) \in \{0, 1, 2, \dots, 255\}$

1. progowanie (rozdział 3.3.1) → $h(m, n)$

2. błąd kwantyzacji δ

$$\delta(m, n) = g(m, n) - h(m, n)$$

3. rozkład błędu na sąsiednie piksele

$$\begin{aligned}
 g(m+1, n) &:= g(m+1, n) + \frac{7}{16}\delta \\
 g(m+1, n+1) &:= g(m+1, n+1) + \frac{1}{16}\delta \\
 g(m, n+1) &:= g(m, n+1) + \frac{5}{16}\delta \\
 g(m-1, n+1) &:= g(m-1, n+1) + \frac{3}{16}\delta
 \end{aligned}$$



Grafika 3.21: Redukcja palety szarości z wykorzystaniem algorytmu Floyd-Steinberga.

3.3.3 Raster (*Halftone*)

- symulacja szarości (intensywności) poprzez użycie (czarnych) kropek o różnej wielkości lub odstępach
- grafika 3.22

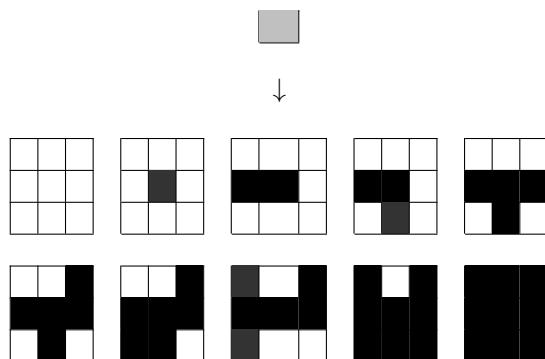


Grafika 3.22: Rasterowanie obrazu. Źródło: The Canadian Illustrated News, October 30th 1869

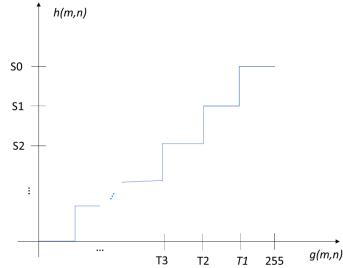
3.3.4 Progowanie progiem zmiennym (*Ordered dithering*)

Próg zmienny

- piksel zostaje zastąpiony przez 3×3 piksele (lub 4×4 itd.)



- grupa $n \times n$ może reprezentować $n^2 + 1$ poziomów szarości S (jasności)
 \longrightarrow kwantyzacja obrazu do $n^2 + 1$ poziomów szarości



- macierz ditheringu dla grupy 3×3
 10 poziomów szarości S_0, S_1, \dots, S_9 , $S_i > S_j$,
 np. $S_0 = 255$ (biały), $S_9 = 0$ (czarny)

$$\begin{pmatrix} 6 & 8 & 4 \\ 1 & 0 & 3 \\ 5 & 2 & 7 \end{pmatrix}$$

- aby przedstawić poziom szarości S_k , na kolor czarny ustawiane są wszystkie piksele, których wartość jest mniejsza niż k

Przykład:

redukcja palety szarości metodą zmiennego progu \longrightarrow grafika 3.23

Metody wieloprogowe

\longrightarrow nie zwiększa pierwotnego obrazu (grafika 3.24)

Przykład:

redukcja palety szarości \longrightarrow grafika ??

- dana macierz ditheringu 4×4

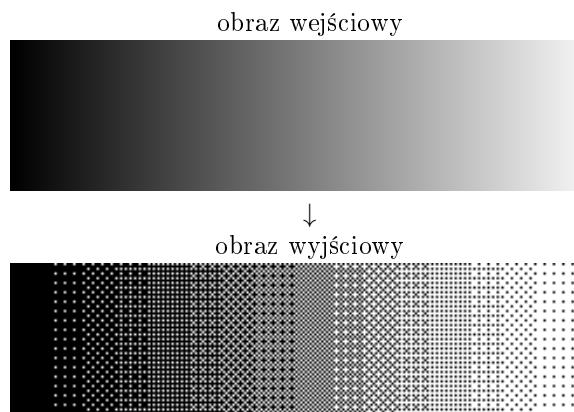
$$M = \begin{pmatrix} 0 & 128 & 32 & 160 \\ 192 & 64 & 224 & 96 \\ 48 & 176 & 16 & 144 \\ 240 & 112 & 208 & 80 \end{pmatrix}$$

- obraz wyjściowy $h(m, n)$

$$h(m, n) = \begin{cases} 0 \text{ (black)} & : g(m, n) < M(m \mod 4, n \mod 4) \\ 255 \text{ (white)} & : g(m, n) \geq M(m \mod 4, n \mod 4) \end{cases}$$



Grafika 3.23: Redukcja palety szarości z wykorzystaniem zmennego progu.



Grafika 3.24: Przykładowe 17 wzorów macierzy ditheringu 4×4 , gdy jest używana tylko z dwoma kolorami, czarnym i białym

Bayer matrix

→ algorytm używany do wyświetlania ciągłego obrazu na wyświetlaczu o mniejszej głębi kolorów (np. Microsoft Windows używa go w 16-kolorowych trybach graficznych).

S



Grafika 3.25: Redukcja palety szarości z wykorzystaniem metody wieloprogowej (macierzy Bayera).

- Mapa progowa dana rekurencyjnie (rozmiar mapy progowej jest zazwyczaj potęgą liczby 2):

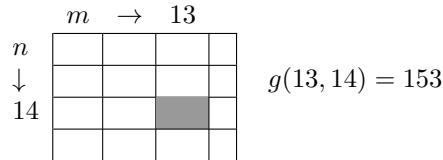
$$\begin{aligned} M_1 &= 0 \\ M_{2k} &= \frac{1}{(2k)^2} \begin{pmatrix} (2k)^2 \cdot M_k & (2k)^2 \cdot M_k + 2 \\ (2k)^2 \cdot M_k + 3 & (2k)^2 \cdot M_k + 1 \end{pmatrix} : k > 0 \end{aligned}$$

Algorytm (w pseudokodzie): g : wartość koloru wejściowego h : wartość koloru wyjściowego r : zakres wartości kolorów wyjściowych M : Bayer matrix o rozmiarze $k \times k$ (n, m) : pozycja piksela

```
h=nearest_palette_color(g + r · (M(n mod k, m mod k) - 0.5))
```

Przykład $k = 4$

$$M_4 = \frac{1}{16} \begin{pmatrix} 0 & 8 & 2 & 10 \\ 12 & 4 & 14 & 6 \\ 3 & 11 & 1 & 9 \\ 15 & 7 & 13 & 5 \end{pmatrix}$$

Obraz wejściowy $g(m, n)$ w skali szarości:Obraz wyjściowy $h(m, n)$:

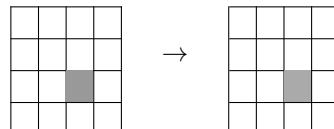
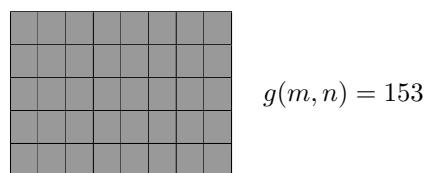
$$\begin{aligned} h(13, 14) &= \text{nearest_palette_color}(g(13, 14) + r \cdot (M(13 \bmod 4, 14 \bmod 4) - 0.5)) \\ &= \text{nearest_palette_color}(g(13, 14) + r \cdot (M(1, 2) - 0.5)) \\ &= \text{nearest_palette_color}\left(g(13, 14) + r \cdot \frac{3}{16}\right) \end{aligned}$$

- paleta kolorów wyjściowych: $\{0, 255\}$ ($r = 256$)

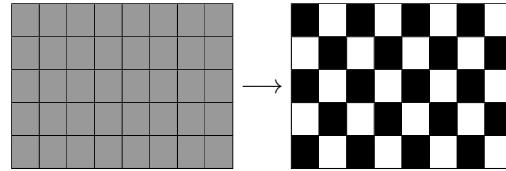
$$h(13, 14) = \text{nearest_palette_color}\left(153 + 256 \cdot \frac{3}{16}\right) = 255$$

- paleta kolorów wyjściowych: $\{0, 85, 170, 255\}$ ($r = 256$)

$$h(13, 14) = \text{nearest_palette_color}\left(153 + 256 \cdot \frac{3}{16}\right) = 170$$

Obraz wejściowy $g(m, n)$ w skali szarości:

- paleta kolorów wyjściowych: $\{0, 255\}$ ($r = 256$)



- paleta kolorów wyjściowych: $\{0, 85, 170, 255\}$ ($r = 256$)

