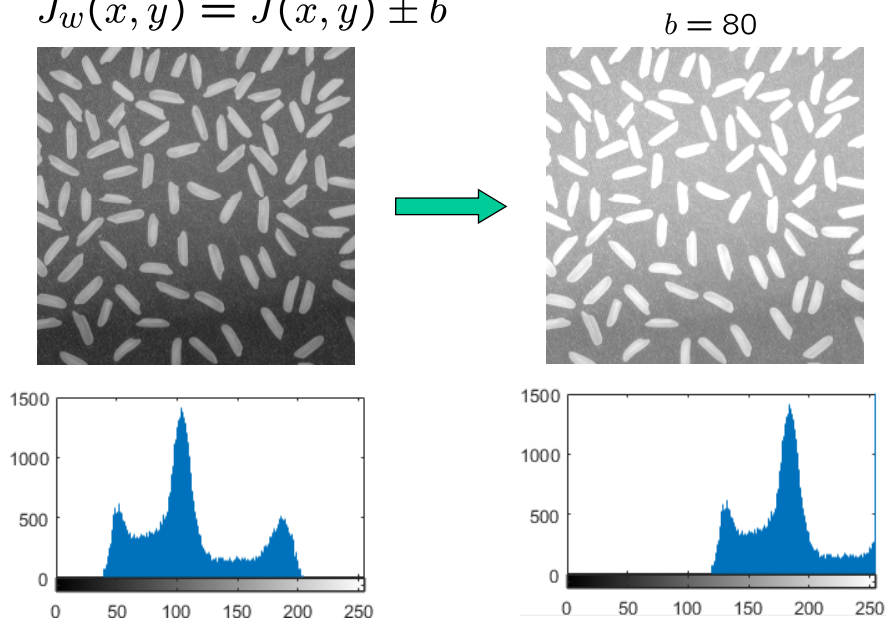


# Proste metody przetwarzania obrazu

Operacje na pikselach obrazu (operacje punktowe, bezkontekstowe)

- Operacje arytmetyczne
  - Dodanie (odjęcie) do obrazu stałej

$$J_w(x, y) = J(x, y) \pm b$$

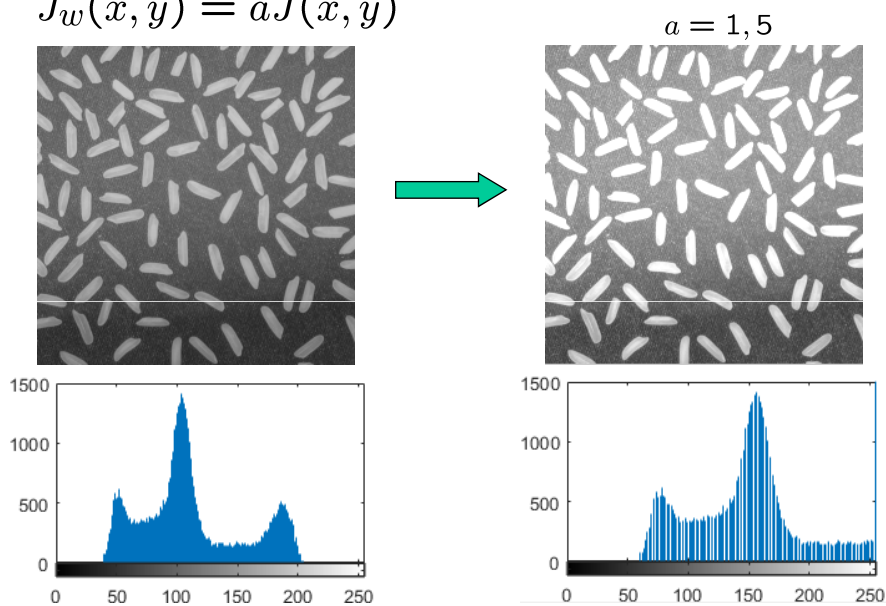


1

# Proste metody przetwarzania obrazu

- Mnożenie (dzielenie) obrazu przez stałą

$$J_w(x, y) = aJ(x, y)$$



Operacje dodawania i mnożenia są operacjami liniowymi

$$J_w(x, y) = aJ(x, y) + b$$

Należy uważać aby nie przekroczyć zakresu!

2

## Proste metody przetwarzania obrazu

- Dodanie (odejmowanie) obrazów jako odpowiednie operacje na macierzach

$$J_w(x, y) = J_1(x, y) \pm J_2(x, y)$$

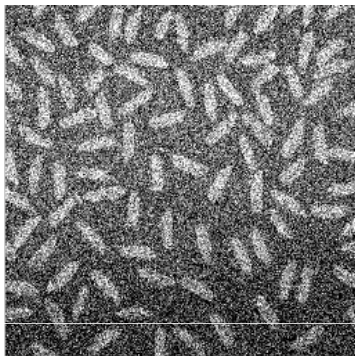


W przypadku przekroczenia zakresu należy stosować arytmetykę przepelnieniową lub normalizację!

3

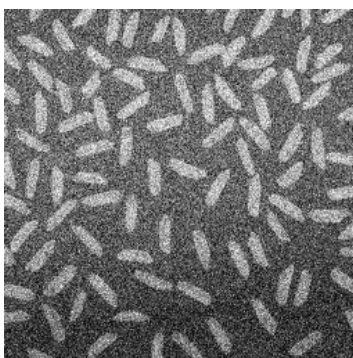
## Proste metody przetwarzania obrazu

- Zastosowanie dodawania obrazów do redukcji szumów

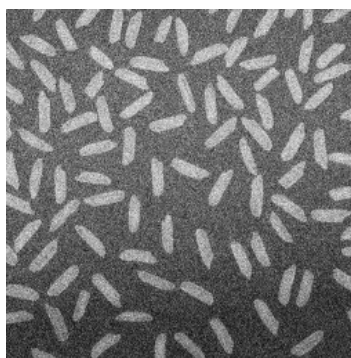


```
In=imnoise(i, 'gaussian', 0, .05)
```

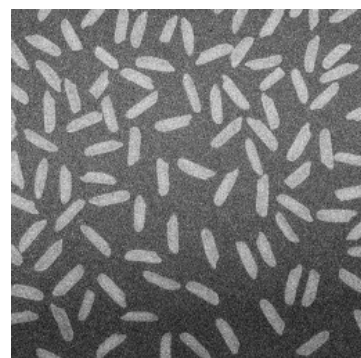
$$J_w(x, y) = \sum_{i=1}^N J_i(x, y) / N$$



$N = 3$



$N = 10$



$N = 20$

4

## Proste metody przetwarzania obrazu

- Mieszanie dwóch obrazów (uogólnienie dodawania)

$$J_w(x, y) = \alpha J_1(x, y) + (1 - \alpha) J_2(x, y) \quad \alpha \in [0, 1]$$

$\alpha = 0,5$



- Mieszanie  $N$  obrazów

$$J_w(x, y) = \alpha_1 J_1(x, y) + \alpha_2 J_2(x, y) + \dots + \alpha_N J_N(x, y)$$

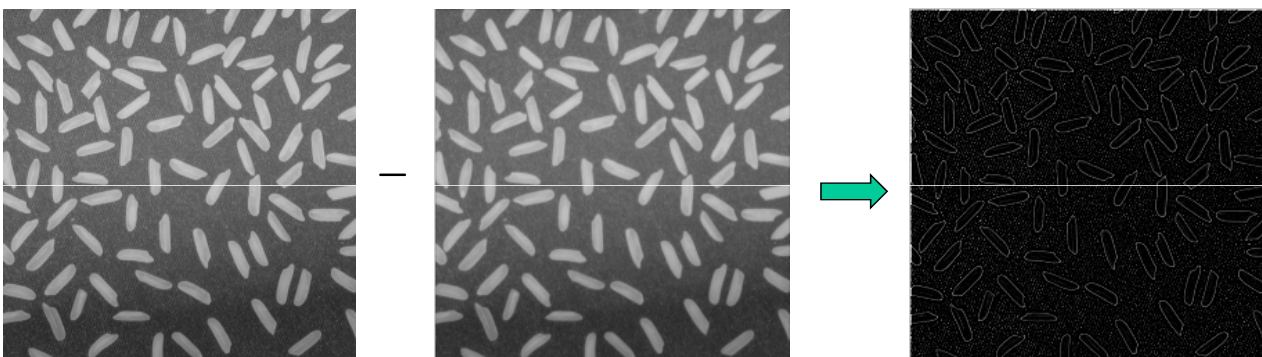
przy czym  $\sum_{i=1}^N \alpha_i = 1$

5

## Proste metody przetwarzania obrazu

- Odejmowanie obrazów – eksponowanie różnic pomiędzy obrazami (obraz różnicowy)

$$J_w(x, y) = J_1(x, y) - J_2(x, y)$$



$J_1(x, y)$

$J_2(x, y)$

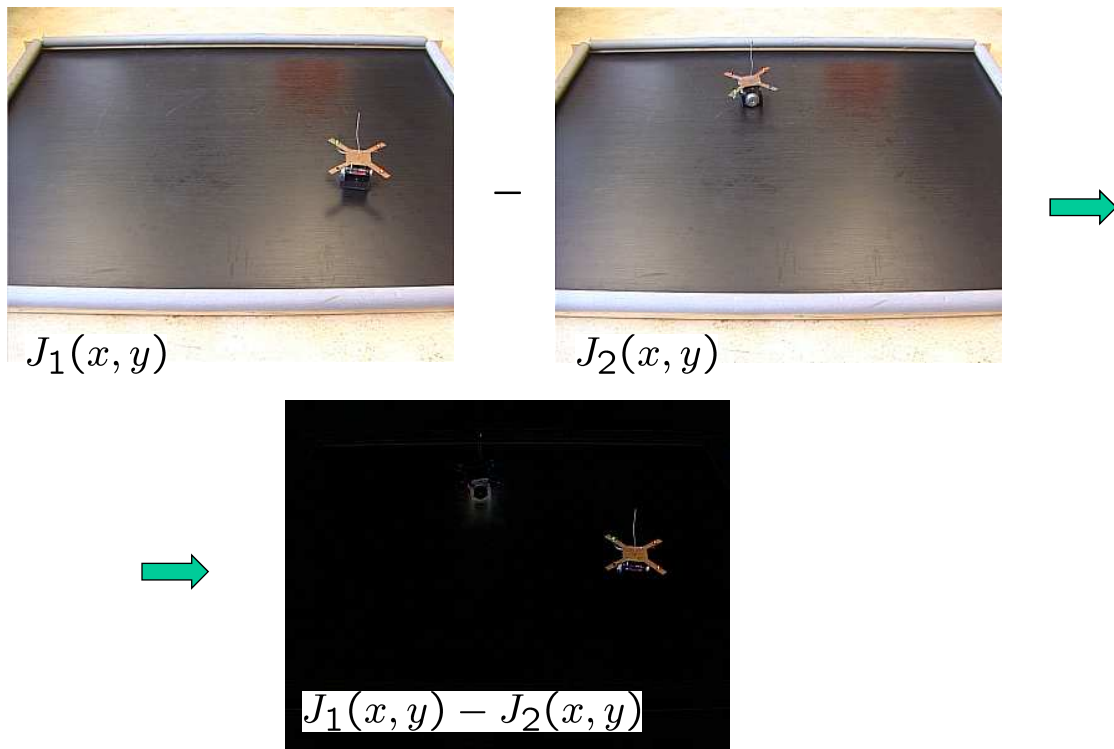
$J_w(x, y) * 10$

`J2=imfilter(J1,fspecial('gaussian'));`

6

# Proste metody przetwarzania obrazu

- Obraz różnicowy do wykrywania obiektów



7

# Proste metody przetwarzania obrazu

- Operacje punktowe nieliniowe

- Potęgowanie

$$J_w(x, y) = J(x, y)^\alpha, \quad \alpha > 0$$

Operacja potęgowania z normalizacją

$$J_w(x, y) = 255 \cdot \left( \frac{J(x, y)}{J_{max}} \right)^\alpha$$

- Dla całkowitych wartości wykładnika i większych od jedności potęgowanie można zrealizować przez mnożenie obrazów
- Dla  $\alpha > 1$  uzyskuje się przyciemnienie obrazu z większym zróżnicowaniem jasnych pikseli
- Dla  $\alpha < 1$  uzyskuje się rozjaśnienie obrazu z większym zwiększeniem kontrastu dla ciemnych części obrazu

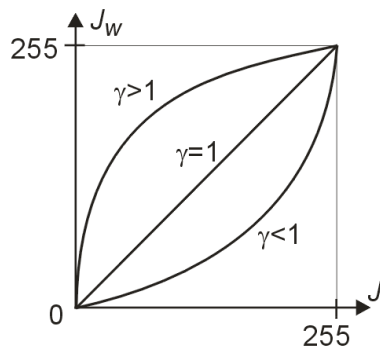
Korekcja Gamma – zmiana poziomów wywodząca się z korekcji nieliniowości układów lamp analizujących oraz kineskopów w systemach telewizyj analogowej (dla systemu PAL  $\gamma = 2,8$ ).

- Korekcję gamma realizuje się przez potęgowanie, przy czym  $\alpha = 1/\gamma$
- Celem jest zazwyczaj rozjaśnienie ciemnych obszarów obrazu

8

# Proste metody przetwarzania obrazu

Krzywe korekcji w zależności od współczynnika  $\gamma$



$\gamma = 2$

Uwaga: w Matlabie  $\alpha = \gamma$

# Proste metody przetwarzania obrazu

- Logarytmowanie

$$J_w(x, y) = \log(1 + J(x, y))$$

W przypadku normalizacji

$$J_w(x, y) = 255 \cdot \frac{\log(1+J(x,y))}{\log(1+J_{max})}$$

Operacja logarytmowania powoduje silne rozjaśnienie i zróżnicowanie najciemniejszych obszarów obrazu

- Maskowanie obrazów poprzez operację mnożenia

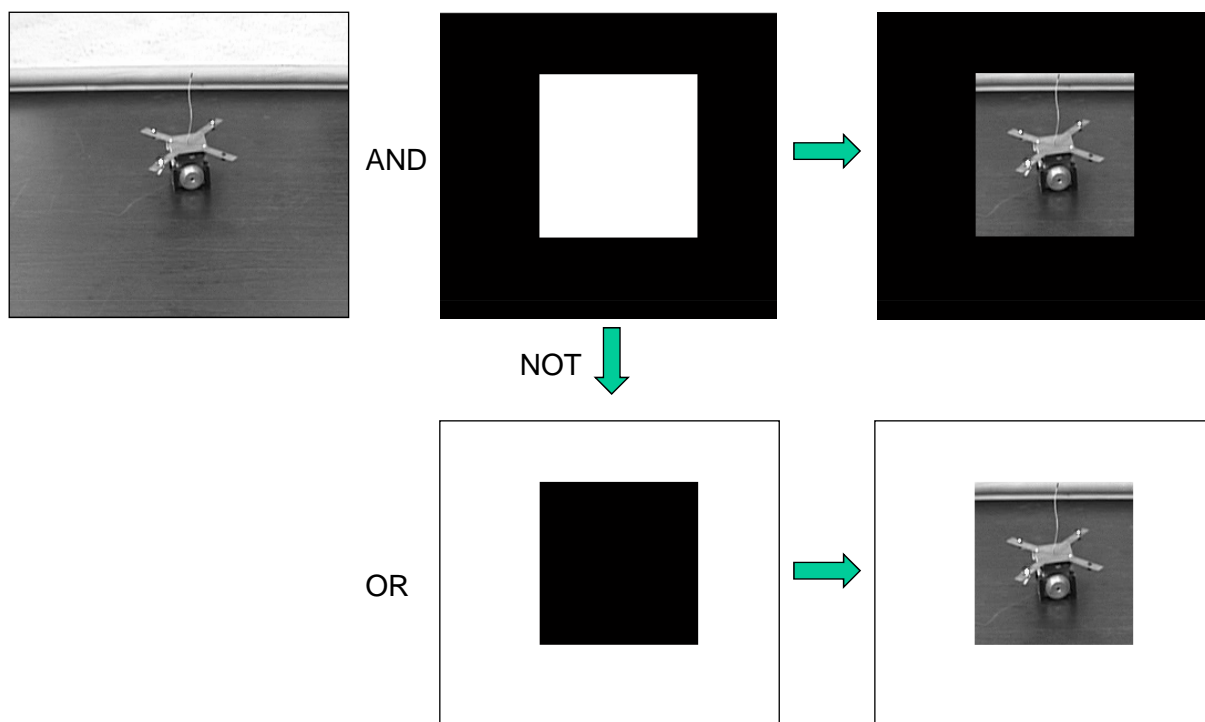
$$J_w(x, y) = J(x, y) \cdot M(x, y)$$

Maskowanie umożliwia wyróżnienie fragmentu obrazu oraz przeprowadzenie na takim podobrazie właściwych operacji.

$$\begin{bmatrix} 3 & 4 & 4 & 3 \\ 5 & 5 & 4 & 3 \\ 3 & 3 & 0 & 2 \\ 1 & 2 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 3 \\ 0 & 5 & 4 & 3 \\ 0 & 3 & 0 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# Proste metody przetwarzania obrazu

- Przykłady maskowania za pomocą operacji logicznych AND i OR



11

# Proste metody przetwarzania obrazu

- Operacje geometryczne

- Przesuwanie punktów obrazu

$$x' = x_o + x_i$$

$(x_o, y_o)$  - współrzędne początkowe pikseli

$$y' = y_o + y_i$$

$(x_i, y_i)$  - wartości przesunięcia współrzędnych pikseli

$(x', y')$  - nowe wartości współrzędnych pikseli po przesunięciu

- Skalowanie obrazu

$$x' = x_o \cdot x_s$$

$x_s, y_s$  - wartości współczynników skalowania

$$y' = y_o \cdot y_s$$

- Obracanie obrazu

$$x' = x_o \cdot \cos(\theta) - y_o \cdot \sin(\theta)$$

$\theta$  - kąt obrotu

$$y' = x_o \cdot \sin(\theta) + y_o \cdot \cos(\theta)$$

Po operacji skalowania oraz obrotu zazwyczaj należy przeprowadzić próbkowanie obrazu korzystając z metod interpolacji dla nieistniejących pikseli (najbliższego sąsiedztwa ang. *nearest neighbor*, dwuliniowa ang. *bilinear*, dwukwadratowa ang. *bicubic*)

12

# Proste metody przetwarzania obrazu

- Interpolacja metodą najbliższego sąsiedztwa

- każdy piksel obrazu wynikowego przyjmuje niezmodyfikowaną wartość piksela obrazu wejściowego położonego najbliżej aktualnie rozpatrywanego punktu
- w przypadku powiększania kilka punktów obrazu wynikowego może przyjąć wartość tego samego punktu z obrazu wejściowego, przy pomniejszaniu niektóre punkty obrazu, które nie mają odpowiedników w obrazie wyjściowym są tracone

$$x_s = M_{src}/M_{dst}$$

$$y_s = N_{src}/N_{dst}$$

Dla punktu obrazu wyjściowego o współrzędnych  $(i, j)$   
współrzędne punktu obrazu wejściowego:

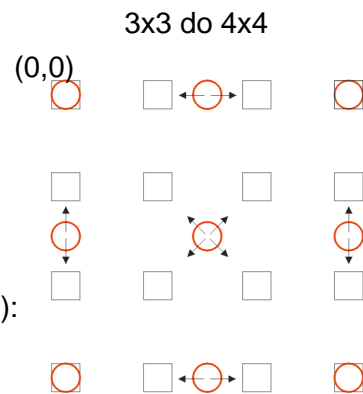
$$x = i \cdot x_s$$

$$y = j \cdot y_s$$

Przykład dla obrazu 3x3 do 4x4 dla punktu wynikowego  $(1,3)$ :

$$x = i \cdot x_s = 1 \cdot 3/4 = 0,75 \approx 1$$

$$y = j \cdot y_s = 3 \cdot 3/4 = 2,25 \approx 2$$



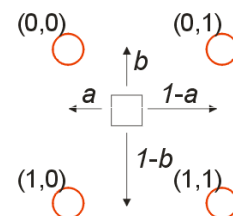
13

# Proste metody przetwarzania obrazu

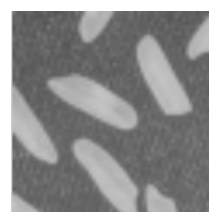
- Interpolacja dwuliniowa

- każdy piksel obrazu wynikowego przyjmuje wartość na podstawie wartości czterech sąsiednich punktów obrazu wejściowego
- w pierwszym kroku wyznacza się współrzędne punktu wynikowego w obrazie wejściowym (podobnie jak w metodzie najbliższego sąsiada)
- następnie przeprowadza się interpolacje w dwóch kierunkach

$$f(i, j) = (1 - b)[(1 - a)f(x, y) + af(x + 1, y)] + b[(1 - a)f(x, y + 1) + af(x + 1, y + 1)]$$



x2.5



najbliższe sąsiedztwo

interpolacja dwuliniowa

14

# Proste metody przetwarzania obrazu

- Histogram obrazu

- Histogram przedstawia w sposób graficzny rozkład częstości występowania punktów o jednakowej jasności (barwie) w obrazie.

$J_0, J_1, J_2, \dots, J_{L-1}$  - oznaczają możliwe wartości jasności pikseli,  $L$  jest liczbą dostępnych poziomów jasności

$$h(J_i) = n_i, \quad i = 0, 1, 2, \dots, L - 1$$

Obliczanie składowych histogramu

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

$$g_i(x, y) = 1 \quad \text{dla} \quad J(x, y) = i$$

$$g_i(x, y) = 0 \quad \text{dla} \quad J(x, y) \neq i$$

$n_i$  - jest liczbą pikseli o wartości jasności  $J_i$

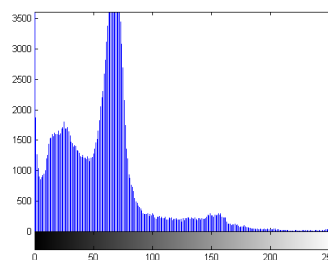
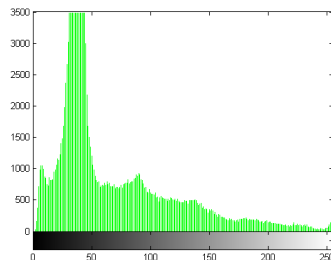
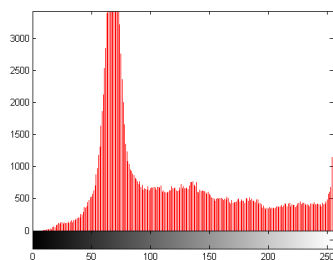
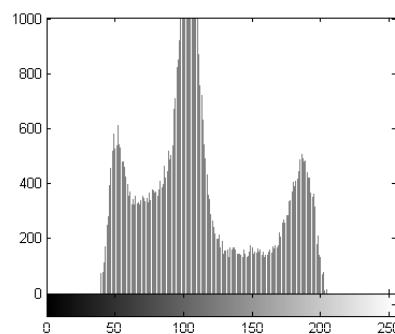
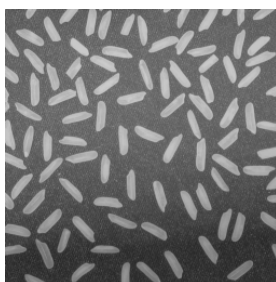
W przypadku normalizacji

$$h(J_i) = \frac{n_i}{n}, \quad n = M \cdot N$$

15

# Proste metody przetwarzania obrazu

- Przykłady obrazów i ich histogramów



16



## Proste metody przetwarzania obrazu

- Wyrównanie histogramu inaczej spłaszczenie (ang. *Equalisation*)
  - Polega na transformacji funkcji  $h(J_i)$  w taki sposób, by nowe jej rzędne były maksymalnie wyrównane

Prawdopodobieństwo wystąpienia na obrazie jasności o poziomie  $r_k$

$$p(r_k) = \frac{n_k}{n}, \quad 0 \leq p(r_k) \leq 1, \quad k = 0, 1, \dots, L - 1$$

$n_k$  - liczba pikseli danej jasności

$n$  - całkowita liczba pikseli na obrazie

Funkcja transformacji wyrównania histogramu (histogram skumulowany)

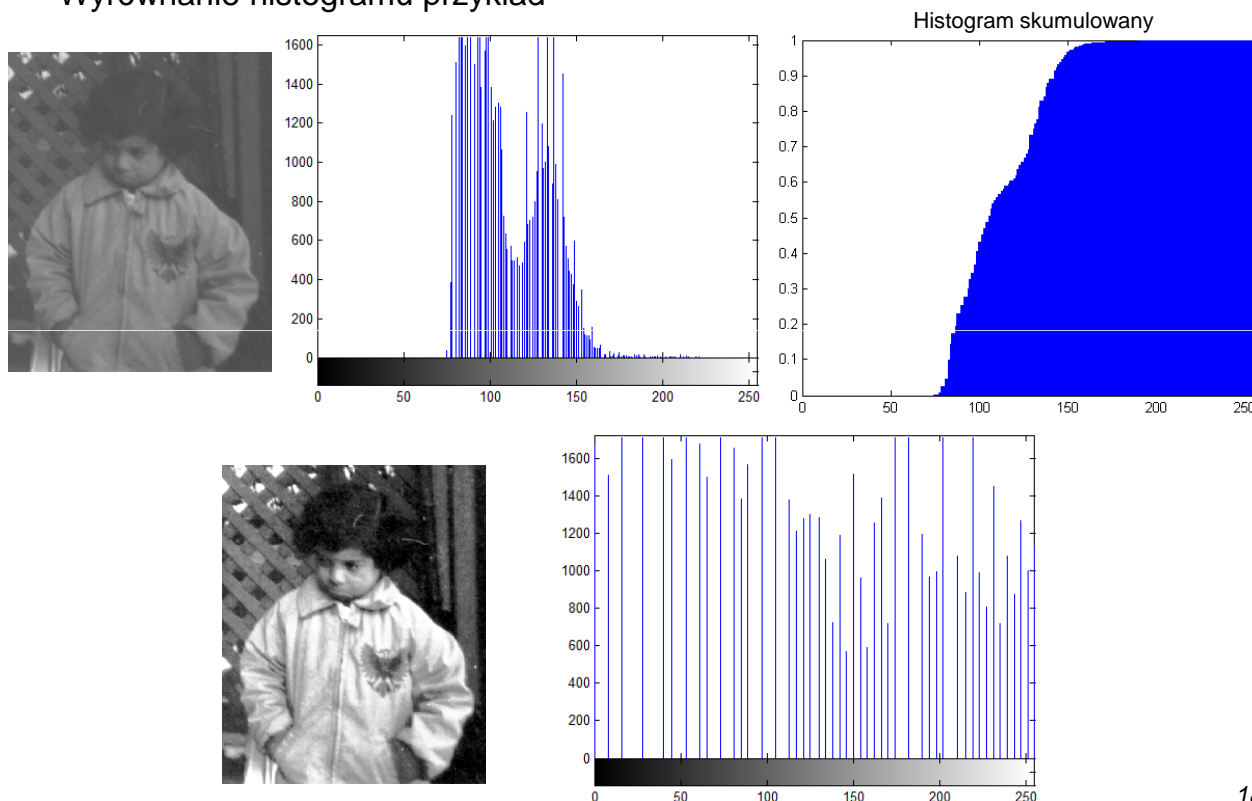
$$s_k = T(r_k) = \sum_{j=0}^k p_r(r_j), \quad k = 0, 1, 2, \dots, L - 1$$

- Wyrównanie histogramu zwiększa kontrast obrazu (czasem nienaturalnie) i lepiej wykorzystuje dostępny zakres poziomów jasności

17

## Proste metody przetwarzania obrazu

- Wyrównanie histogramu przykład

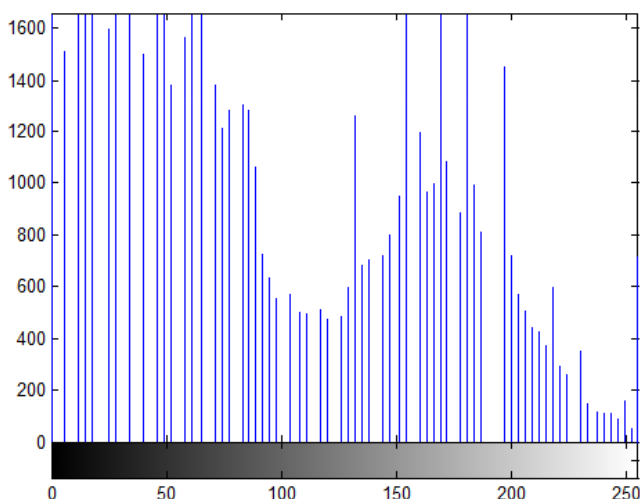


18

## Proste metody przetwarzania obrazu

- Rozszerzanie zakresu jasności, rozciągnięcie histogramu (ang. *Stretching*)
  - Niech zakres jasności zawiera się między  $J_{min}$  i  $J_{max}$ . Operację liniowego rozszerzania jasności na cały zakres wyraża się wzorem

$$J_w(x, y) = \frac{255}{J_{max} - J_{min}} (J(x, y) - J_{min})$$

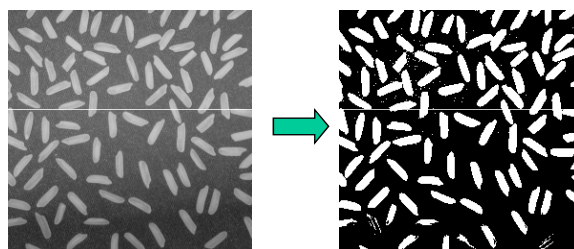


19

## Proste metody przetwarzania obrazu

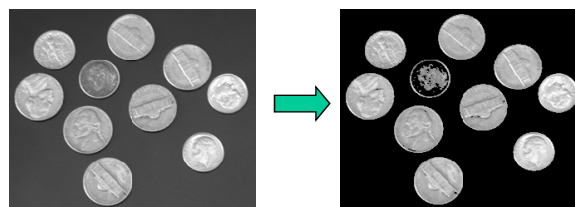
- Progowanie obrazu
  - Podział obrazu na obszary spełniające pewne (określone z góry) kryterium nazywa się *segmentacją*. Jest to jedno z podstawowych zadań analizy obrazu. Pojęcie obszaru jest używane do określenia spójnej grupy punktów obrazu mających wspólną cechę.
  - Najprostsza metoda progowania to binaryzacja obrazu z jedną wartością progu  $t$

$$J_w(x, y) = \begin{cases} 1, & J(x, y) > t \\ 0, & J(x, y) \leq t \end{cases}$$



- Metoda tzw. pseudoprogowania, pozostawia wyodrębniony obszar w obrazie

$$J_w(x, y) = \begin{cases} J(x, y), & J(x, y) > t \\ 0, & J(x, y) \leq t \end{cases}$$



20

## Proste metody przetwarzania obrazu

- Metoda progowania z podwójnym ograniczeniem, pozwala wyodrębnić obszary o jasności leżącej w pewnym przedziale

$$J_w(x, y) = \begin{cases} 0, & J(x, y) \leq t_1 \\ 1, & t_1 < J(x, y) \leq t_2 \\ 0, & J(x, y) > t_2 \end{cases}$$

- Progowanie wielokryterialne może generować obrazy złożone z wielu obszarów o różnych poziomach szarości

$$J_w(x, y) = \begin{cases} 1, & J(x, y) \in D_1 \\ 2, & J(x, y) \in D_2 \\ \dots, & \dots \\ n, & J(x, y) \in D_n \\ 0, & \text{w pozostałych przypadkach} \end{cases}$$

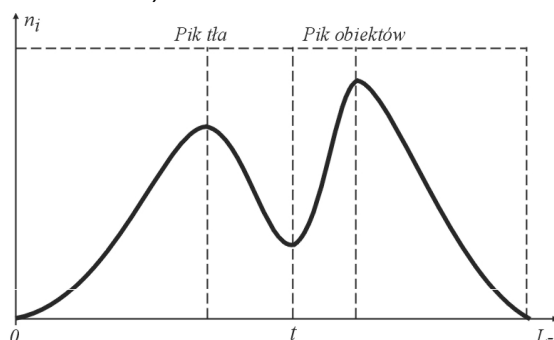
- Progowanie globalne – wartość progowa taka sama dla całego obrazu
- Progowanie lokalne – próg dynamiczny, przyjmuje wartości w zależności od współrzędnych obrazowych

$$J_w(x, y) = \begin{cases} J(x, y), & J(x, y) > t(x, y) \\ 0, & J(x, y) \leq t(x, y) \end{cases}$$

21

## Proste metody przetwarzania obrazu

- Wybór progów dla obrazu, którego histogram ma rozkład bimodalny (metoda nie gwarantuje spójności obszarów)



- Wyznaczanie progów na podstawie statystyki obrazu
  - Wyznaczenie maksymalnego modułu gradientu  $G(x, y)$  jasności dla każdego punktu

$$G_x(x, y) = J(x + 1, y) - J(x - 1, y),$$

$$G_y(x, y) = J(x, y + 1) - J(x, y - 1)$$

$$G(x, y) = \max\{|G_x(x, y)|, |G_y(x, y)|\}$$

- Obliczenie wartości progów  $t = \frac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} J(x, y) G(x, y)}{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} G(x, y)}$

22

# Proste metody przetwarzania obrazu

- Tablice LUT (ang. *Look Up Table*)
  - Tablica LUT to tablica w której zapisano wartości funkcji transformacji
  - Obraz indeksowany jako przykład wykorzystania tablicy LUT

Mapa kolorów  
Tablica LUT

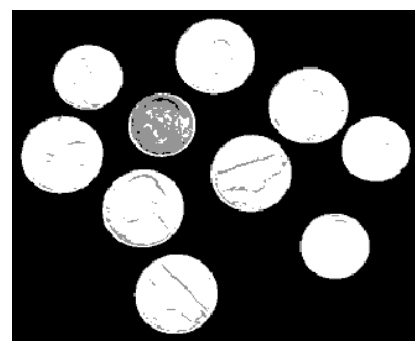
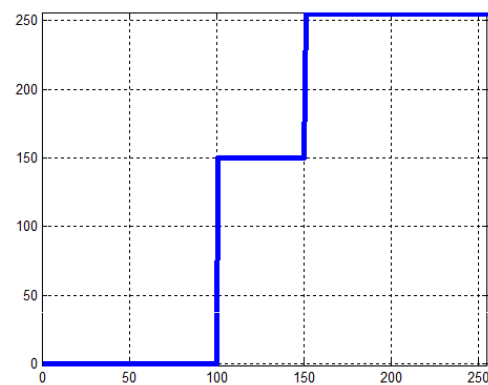
0	0	0
0.0627	0.0627	0.0314
0.2902	0.0314	0
0	0	1.0000
0.2902	0.0627	0.0627
0.3882	0.0314	0.0941
0.4510	0.0627	0
0.2588	0.1608	0.0627

23

# Proste metody przetwarzania obrazu

- Przykład wykorzystania tablicy LUT do binaryzacji wielokryterialnej

0	0
...	...
99	0
100	150
...	...
149	150
150	255
...	...
255	255



24