

Roboty Mobilne

Metody cyfrowego przetwarzania obrazów

dr inż. Tomasz Krzeszowski

2018-05-20

Spis treści

1	Przygotowanie do laboratorium.....	3
2	Cel laboratorium.....	3
3	Przetwarzanie obrazów z wykorzystaniem oprogramowania Matlab	3
3.1	Wyświetlenie pomocy	3
3.2	Zapis, odczyt i wyświetlenie obrazu	3
3.3	Reprezentacja obrazu.....	3
3.4	Zmiana reprezentacji obrazu.....	3
3.5	Histogram	4
3.6	Zakłócanie i filtracja obrazu.....	4
3.6.1	Zakłócanie obrazu.....	4
3.6.2	Filtracja obrazu	5
3.7	Operacje morfologiczne	5
3.8	Znajdywanie krawędzi, konturów i narożników.....	6
4	Zadania do samodzielnego wykonania.....	6
5	Literatura	7

1 Przygotowanie do laboratorium

Przed zajęciami należy się zapoznać z niniejszą instrukcją, w szczególności ze składnią oraz przeznaczeniem poszczególnych poleceń.

2 Cel laboratorium

Celem laboratorium jest zapoznanie studentów z podstawowymi metodami cyfrowego przetwarzania obrazów. Podczas ćwiczeń wykorzystywany będzie pakiet Image Processing Toolbox oprogramowania Matlab.

3 Przetwarzanie obrazów z wykorzystaniem oprogramowania Matlab

3.1 Wyświetlenie pomocy

W celu wyświetlenia pomocy dla danego polecenia należy je wpisać w linii poleceń, a następnie wcisnąć przycisk F1.

3.2 Zapis, odczyt i wyświetlenie obrazu

Do odczytywania, zapisywania oraz wyświetlania obrazu wykorzystuje się odpowiednio funkcje `imread`, `imwrite` i `imshow`.

- **imread** – wczytuje plik z dysku, **składnia polecenia**:

```
I = imread('nazwa_pliku')  
[X, map] = imread('nazwa_pliku') (musi być właściwy  
format pliku)
```

gdzie `X` – mapa obrazu, `map` – mapa kolorów.

- **imwrite** – zapisuje obraz na dysku, **składnia polecenia**:

```
imwrite(I, 'nazwa_pliku')  
imwrite(X, map, 'nazwa_pliku')
```

- **imshow** – wyświetla obraz, **składnia polecenia**:

```
imshow(I)  
imshow(X, map)
```

3.3 Reprezentacja obrazu

W oprogramowaniu MATLAB obraz może być reprezentowany na 4 sposoby:

- binarnej - obraz jest reprezentowany przez macierz o wartościach 0 lub 1.
- intensywnościowej - pozwala na przechowywanie obrazów w odcieniach szarości. Każdy element macierzy to liczba określająca intensywność barwy szarej.
- indeksowej - obraz składa się z dwóch macierzy: macierzy obrazu, której elementy są wskaźnikami do mapy kolorów oraz macierzy z mapą kolorów.
- RGB - obraz jest reprezentowany przez trzy macierze, z których każda określa intensywność jednej z barw składowych (czerwonej, zielonej i niebieskiej).

3.4 Zmiana reprezentacji obrazu

- **ind2rgb** - przekształca obraz indeksowy na obraz RGB. Intensywność każdego z kolorów jest reprezentowana przy pomocy jednej macierzy.
Składnia polecenia:
`[RGB] = ind2rgb(X, map)`
- **rgb2ind** - przekształca obraz RGB na obraz indeksowy.
Składnia polecenia:
`[X, map] = rgb2ind(RGB)`
- **rgb2gray** - przekształca obraz RGB na obraz w odcieniach szarości.
Składnia polecenia:
`I = rgb2gray(RGB)`
- **ind2gray** - przekształca obraz indeksowy na obraz w odcieniach szarości.
Składnia polecenia:
`I = ind2gray(X, map)`
- **gray2ind** - przekształca obraz w odcieniach szarości na obraz indeksowy.
Składnia polecenia:
`[X, map] = gray2ind(I)`
- **im2bw** - przekształca obraz intensywnościowy, indeksowy lub RGB na obraz czarno-biały.
Składnia polecenia:
`BW = im2bw(I, level)`
`BW = im2bw(RGB, level)`
`BW = im2bw(X, map, level)`

3.5 Histogram

Do wyświetlenia histogramu stosuje się funkcję `imhist` natomiast operację wyrównania histogramu można wykonać przy pomocy funkcji `histeq`.

- **imhist** - wyświetla histogram obrazu.
Składnia polecenia:
`imhist(I)`
`imhist(X, map)`
- **histeq** - wykonanie operacji wyrównania histogramu.
Składnia polecenia:
`J = histeq(I)`
`newmap = histeq(X, map)`

3.6 Zakłócanie i filtracja obrazu

3.6.1 Zakłócanie obrazu

Do zakłócania obrazu służy funkcja `imnoise`

- **imnoise** - funkcja pozwala na wprowadzenie zakłóceń na obrazie.
Składnia polecenia:
`J = imnoise(I, 'type')`
`J = imnoise(I, 'type', arguments)`
gdzie 'type', określa typ wprowadzanego szumu. Dostępne są następujące opcje:

`gaussian` – dla białego szumu Gaussa dla stałej wartości średniej i wariancji;
`localvar` – dla białego szumu Gaussa z wartością średnią równą zero oraz wariancją zależną od intensywności;
`salt & pepper` – dla szumu typu "sól i pieprz";
`poisson` – dla szum Poissona;
`speckle` – dla szumu "speckle".

natomiast parametr `arguments` określa parametry danego szumu (patrz dokumentacja).

3.6.2 Filtracja obrazu

- **`imfilter`** - spłot obrazu z filtrem

Składnia polecenia:

`J = imfilter(I, F)`

gdzie `F` to macierz na podstawie której odbywa się filtracja. Przykładowe filtry dolnoprzepustowe:

$$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \frac{1}{10} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \frac{1}{12} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 4 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Przykładowe filtry górnoprzepustowe:

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}, \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}, \begin{bmatrix} 1 & -2 & 1 \\ -2 & 5 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

- **`medfilt2`** - filtracja medianowa

Składnia polecenia:

`B = medfilt2(A, [m n])`

`B = medfilt2(A)`

3.7 Operacje morfologiczne

Operacje morfologiczne są realizowane na obrazach binarnych. Do utworzenia elementu strukturalnego (SE) należy użyć funkcji `strel`

- **`imdilate`** - operacja dylatacji morfologicznej

Składnia polecenia:

`J = imdilate(I, SE)`

- **`imerode`** - operacja erozji morfologicznej

Składnia polecenia:

`J = imerode(I, SE)`

- **`imclose`** - operacja zamknięcia morfologicznego

Składnia polecenia:

`J = imclose(I, SE)`

- **`imopen`** - operacja otwarcia morfologicznego

Składnia polecenia:

```
J = imopen(I, SE)
```

3.8 Znajdywanie krawędzi, konturów i narożników

Do ekstrakcji krawędzi wykorzystuje się funkcję `edge`. Funkcja działa na obrazach w odcieniach szarości.

- **edge** - ekstrakcja krawędzi

Składnia polecenia:

```
BW = edge(I, 'method')  
BW = edge(I, 'method', thresh)  
BW = edge(I, 'method', thresh, direction)
```

gdzie `thresh` to próg obcięcia, `direction` definiuje kierunek wykrywanych krawędzi, a `method` określa wykorzystywaną metodę. Parametr `method` może przyjmować następujące wartości:

```
sobel – operator Sobela ;  
prewitt – operator Prewitta;  
roberts – operator Roberts'a;  
log – operator Laplace'a;  
canny – algorytm Canny'ego.
```

- **imcontour** - wyznaczenie konturów na obrazie (obraz powinien być w odcieniach szarości)

Składnia polecenia:

```
imcontour(I)
```

- **corner** - znajdowanie narożników

Składnia polecenia:

```
C = corner(I)  
C = corner(I, 'method')
```

gdzie `method` określa algorytm wykorzystywany do detekcji narożników i może przyjmować następujące wartości:

```
Harris – detektor Harris'a;  
MinimumEigenvalue – metoda Shi & Tomasi's.
```

4 Zadania do samodzielnego wykonania

Z wykonywanych zadań laboratoryjnych należy przygotować sprawozdanie. Sprawozdanie powinno zawierać:

- a) stronę tytułową;
- b) wywoływane polecenia wraz z wynikiem ich działania;
- c) wnioski.

Sprawozdanie należy przestać za pomocą aplikacji internetowej znajdującej się pod adresem:
<http://sprawozdania.kia.prz.edu.pl/>

Na stronie należy założyć konto i zapisać się na odpowiednie zajęcia.

Zadania do wykonania:

1. Wczytać, zapisać oraz wyświetlić przykładowe obrazy.
2. Dokonać konwersji pomiędzy różnymi typami reprezentacji obrazów.
3. Wyświetlić histogram dla przykładowych obrazów i dokonać wyrównania histogramu.
4. Dodać szum do przykładowych obrazów, a następnie przetestować różne metody filtracji (w metodach dodających szum i metodach filtracji należy uwzględnić różne parametry).
5. Przetestować metody morfologiczne na obrazach binarnych.
6. Przeprowadzić dostępne metod ekstrakcji krawędzi, konturu i narożników.

5 Literatura

1. Richard Szeliski: „Computer Vision: Algorithms and Applications”. Springer, New York, 2010
2. M. Wysocki, J. Marnik, T. Kapuściński, „Wizja komputerowa. Materiały pomocnicze.” Oficyna Wydawnicza Politechniki Rzeszowskiej, Rzeszów, 2004