

Wydział Mechaniczny Politechniki Białostockiej

SPRAWOZDANIE Z PRZEDMIOTU

Systemy sterowania robotów

Kod przedmiotu: **MYAR2S01005M**

Temat: Barwa w obrazach cyfrowych.

Imię i nazwisko: Janusz Chmaruk, Jakub Dacewicz

Kierunek studiów: Automatyka i Robotyka

Specjalność: -

Semestr: I

Rok akademicki: 2022/2023

Data wykonania pracy: 28.05.2023

.....
podpis studenta

Weryfikacja efektów kształcenia:

EK1

EK5

EK2

EK6

EK3

EK7

EK4

EK8

Uwagi prowadzącego:

Ocena sumaryczna:

.....

podpis prowadzącego

Spis treści

1 Cel ćwiczenia	2
2 Kod	2
3 Wyniki	6
4 Wnioski	8

1 Cel ćwiczenia

Celem ćwiczenia jest przybliżenie możliwości wykorzystywania biblioteki Image Processing Toolbox (IPT) do przetwarzania i analizy informacji o barwie zawartej w obrazie cyfrowym. Zakres laboratorium obejmuje: konwertowanie obrazów cyfrowych do innej przestrzeni barw (opisu w różnych modelach barwnych), pseudokolorowanie, praktyczne wykorzystanie wybranych metod i technik analizy barwnej obrazów cyfrowych.

2 Kod

```
1 % Wyczyszczenie środowiska
2 clc
3 clear all
4 close all
5
6 % Wczytanie obrazu
7 [obrazRGB] = imread('picture1.jpg');
8
9 % Wydzielenie składowych R, G, B
10 obrazR = obrazRGB(:,:,1);
11 obrazG = obrazRGB(:,:,2);
12 obrazB = obrazRGB(:,:,3);
13
14 % Pobranie rozmiarów obrazu
15 [w, k, ~] = size(obrazRGB);
16
17 % Wyświetlenie oryginalnego obrazu
18 figure(1)
19 imshow(obrazRGB)
20 title('Obraz RGB')
21
22 % Utworzenie palet kolorów dla
23 % składowych R, G, B
24 r=0:1:255;
25 rr=r';
26 z=zeros(256,3);
27 z(:,1)=rr;
28 red=z/255;
29
30 r2=0:1:255;
31 rr2=r2';
32 z2=zeros(256,3);
33 z2(:,2)=rr2;
34 green=z2/255;
35 r3=0:1:255;
36
37 rr3=r3';
38 z3=zeros(256,3);
39 z3(:,3)=rr3;
40 blue=z3/255;
41
42 % Wartość nasycenia do osiągnięcia (Z)
43 Z = 90;
44
45 % Zmniejszenie nasycenia dla
46 % poszczególnych kanałów barwy
47 obrazR_redukowany = obrazR * (Z/255);
48 obrazG_redukowany = obrazG * (Z/255);
49 obrazB_redukowany = obrazB * (Z/255);
50
51 % Wygenerowanie wykresów 3D dla
52 % składowych kolorów
53 figure(2)
54 mesh(1:k, w:-1:1, obrazR_redukowany,
55 'FaceColor', 'texturemap', 'EdgeColor',
56 'none', 'CdataMapping', 'direct');
57 colormap(red)
58 view([-15 70]);
59 colorbar('vert');
60 title('Wykres przestrzenny - kanał
61 czerwony (zredukowany)');
62 xlabel('Kolumna'); ylabel('Wiersz');
63 zlabel('Natężenie');
64
65 figure(3)
66 surf(1:k, w:-1:1, obrazG_redukowany,
67 'FaceColor', 'texturemap', 'EdgeColor',
68 'none', 'CdataMapping', 'direct');
69 colormap(green)
70 view([-15 70]);
71 colorbar('vert');
72 title('Wykres przestrzenny - kanał
73 zielony (zredukowany)');
74 xlabel('Kolumna'); ylabel('Wiersz');
75 zlabel('Natężenie');
76
77 %% 2. Dla jednego, wybranego
78 % obrazu RGB dokonać konwersji do 3
79 % wybranych innych
80 % reprezentacji barwnych. Wyniki w
81 % postaci obrazów umieścić w sprawozdaniu.
82 % Scharakteryzować
83 % dla każdego modelu barwnego rodzaj
84 % pozyskiwanej informacji barwnej.
```

```

77
78 % Konwersja na przestrzeń barwną HSV
79 obrazHSV = rgb2hsv(obrazRGB);
80
81 % Konwersja na przestrzeń barwną Lab
82 obrazLab = rgb2lab(obrazRGB);
83
84 % Konwersja na przestrzeń barwną YCbCr
85 obrazYCbCr = rgb2ycbcr(obrazRGB);
86
87 % Wyświetlanie obrazów w trzech różnych
     reprezentacjach barwnych
88 figure(5)
89 imshow(obrazRGB)
90 title('Obraz RGB')
91
92 figure(6)
93 imshow(obrazHSV)
94 title('Obraz w przestrzeni barwnej HSV')
95 xlabel('Odcień'); ylabel('Nasycenie');
96 zlabel('Wartość')
97
98 figure(7)
99 imshow(obrazLab)
100 title('Obraz w przestrzeni barwnej Lab')
101 xlabel('Jasność'); ylabel('a');
102 zlabel('b')
103
104 figure(8)
105 imshow(obrazYCbCr)
106 title('Obraz w przestrzeni barwnej
     YCbCr')
107 xlabel('Luminancja');
108 ylabel('Chrominancja Cb');
109 zlabel('Chrominancja Cr')
110
111 %% 3.      Dla dowolnego obrazu
112 %       barwnego:
113 %a)      dokonać konwersji do obrazu w
114 odcieniach szarości (rgb2gray)
115
116 % Konwersja na obraz w odcieniach
117 szarości
118 obrazGray = rgb2gray(obrazRGB);
119
120 % Wyświetlanie obrazów
121 figure(9)
122 imshow(obrazRGB)
123 title('Obraz barwny')
124
125 figure(10)
126 imshow(obrazGray)
127 title('Obraz w odcieniach szarości')
128
129 %% b)      wyświetlić profil barwny
130 %       RGB dla wybranego obiektu/fragmentu
131 obraz;
132
133 %% Wyświetlanie obrazu
134 figure(11)
135 imshow(obraz)
136 title('Obraz oryginalny')
137
138 %% Wybranie obszaru zainteresowania
139 h = imrect; % Narzędzie do wyboru
140 obszaru prostokątnego
141 position = wait(h);
142 x = round(position(1));
143 y = round(position(2));
144 width = round(position(3));
145 height = round(position(4));
146
147 %% Wyodrębnienie obszaru zainteresowania
148 obszarZainteresowania =
149 obrazRGB(y:y+height-1, x:x+width-1, :);
150
151 %% Obliczenie średnich wartości dla
152 %       każdego kanału barwnego
153 sredniaR =
154 mean(mean(obszarZainteresowania(:,:,1)));
155 sredniaG =
156 mean(mean(obszarZainteresowania(:,:,2)));
157 sredniaB =
158 mean(mean(obszarZainteresowania(:,:,3)));
159
160 %% Wyświetlanie profilu barwnego RGB
161 figure(12)
162 plot([1, 2, 3], [sredniaR, sredniaG,
163 sredniaB], 'o-', 'LineWidth', 2)
164 title('Profil barwny RGB dla wybranego
     obiektu')
165 xlabel('Kanał barwny')
166 ylabel('Średnia wartość piksela')
167 set(gca, 'XTick', [1, 2, 3],
168 'XTickLabel', {'R', 'G', 'B'})
169 grid on
170
171 %% c)      zmodyfikować wybrany kanał
172 %       barwny (np. poprzez odjęcie, dodanie,
173 mnożenie itd.
174 % elementów w macierzy), a następnie
175 %       za pomocą funkcji cat scalić
176 %       macierze; wyświetlić obraz;
177
178 %% Wybrany kanał barwny (np. czerwony - R)
179 kanal = obrazRGB(:,:,1);
180
181 %% Modyfikacja kanału barwnego (np.
182 %       dodanie wartości)
183 modyfikacja = kanal + 150;
184
185 %% Scalenie macierzy
186 obrazModyfikowany = cat(3, modyfikacja,
187 obrazRGB(:,:,2), obrazRGB(:,:,3));
188
189 %% Wyświetlanie obrazu
190 figure(13)
191 imshow(obraz)
192 title('Obraz modyfikowany')

```

```

170
171 figure(14)
172 imshow(obrazModyfikowany)
173 title('Obraz po modyfikacji')
174
175 %% d)      dokonać pseudokolorowania
176 %% dla 2 przypadki dla 2 wartości: 64, 255
177 %% kolorów;
178
179 %% Konwersja obrazu do obrazu w
180 %% odcieniach szarości
181 obrazSzary = rgb2gray(obrazRGB);
182
183 %% Pseudokolorowanie dla 64 kolorów
184 mapaKolorow64 = parula(64); % Wybór mapy
185 %% kolorów (np. parula)
186 obrazPseudokolor64 = ind2rgb(obrazSzary,
187 mapaKolorow64);
188
189 %% Pseudokolorowanie dla 255 kolorów
190 mapaKolorow255 = parula(255); % Wybór
191 %% mapy kolorów (np. parula)
192 obrazPseudokolor255 =
193 ind2rgb(obrazSzary, mapaKolorow255);
194
195 %% Wyświetlanie obrazów
196 figure(15)
197 subplot(2, 3, 1)
198 imshow(obrazSzary)
199 title('Obraz w odcieniach szarości')
200
201 subplot(2, 3, 2)
202 imshow(obrazPseudokolor64)
203 title('Pseudokolorowanie - 64 kolory
204 (parula)')
205
206 subplot(2, 3, 3)
207 imshow(obrazPseudokolor255)
208 title('Pseudokolorowanie - 255 kolorów
209 (parula)')
210
211 subplot(2, 3, 4)
212 imshow(obrazPseudokolor64)
213 title('Pseudokolorowanie - 64 kolory
214 (jet)')
215
216 subplot(2, 3, 5)
217 imshow(obrazPseudokolor255)
218 title('Pseudokolorowanie - 255 kolorów
219 (jet)')
220
221 subplot(2, 3, 6)
222 imshow(obrazRGB)
223 title('Obraz w odcieniach RGB')
224
225 %% e)      * wyodrębnić na obrazie
226 %% wybrany kolor.
227
228 %% Konwersja obrazu do obrazu w
229 %% odcieniach szarości
230
231 obrazSzary = rgb2gray(obrazRGB);
232
233 %% Wyodrębnienie koloru czerwonego
234 progR = 150; % Dolny próg dla składowej
235 R
236 progG = 0;   % Dolny próg dla składowej
237 G
238 progB = 0;   % Dolny próg dla składowej
239 B
240
241 %% Utworzenie maski dla koloru czerwonego
242 maskaCzerwony = (obrazRGB(:,:,1) >
243 progR) & (obrazRGB(:,:,2) > progG) &
244 (obrazRGB(:,:,3) > progB);
245
246 %% Wyodrębnienie koloru czerwonego na
247 %% obrazie
248 obrazWyodrebniony = obrazRGB;
249 obrazWyodrebniony(repmat(~maskaCzerwony,[1,1,3])) =
250 0;
251
252 %% Wyświetlanie obrazów
253 figure(16)
254 subplot(1, 2, 1)
255 imshow(obrazRGB)
256 title('Obraz oryginalny')
257
258 subplot(1, 2, 2)
259 imshow(obrazWyodrebniony)
260 title('Wyodrębniony kolor czerwony')
261
262 %% 4.      *Zmodyfikować kontrast
263 %% dowolnego obrazu (funkcja imadjust).
264 %% Sprawdzić różne wartości parametrów
265 %% tej funkcji. Dla wybranego przypadku
266 %% porównać
267 %% histogramy barw przed korekcją i po
268 %% korekcji gamma.
269
270 %% Konwersja obrazu do obrazu w
271 %% odcieniach szarości
272 obrazSzary = rgb2gray(obrazRGB);
273
274 %% Wyświetlenie obrazu przed regulacją
275 %% kontrastu
276 figure(17)
277 subplot(1, 3, 1)
278 imshow(obrazSzary)
279 title('Obraz przed regulacją kontrastu')
280
281 %% Regulacja kontrastu dla różnych
282 %% wartości parametrów
283 low_in = 0.2; % Dolne ograniczenie dla
284 %% wartości wejściowych
285 high_in = 0.8; % Górnne ograniczenie dla
286 %% wartości wejściowych
287 low_out = 0;   % Dolne ograniczenie dla
288 %% wartości wyjściowych
289 high_out = 1;  % Górnne ograniczenie dla
290 %% wartości wyjściowych

```

```

259
260 obrazSkorygowany1 = imadjust(obrazSzary,
261 [low_in high_in], [low_out high_out]);
262
263 low_in = 0.4; % Dolne ograniczenie dla
264 wartości wejściowych
265 high_in = 0.6; % Górnne ograniczenie dla
266 wartości wejściowych
267
268 obrazSkorygowany2 = imadjust(obrazSzary,
269 [low_in high_in], [low_out high_out]);
270
271 % Wyświetlenie obrazów po regulacji
272 kontrastu
273 subplot(1, 3, 2)
274 imshow(obrazSkorygowany1)
275 title('Obrazy po regulacji kontrastu1')
276
277 subplot(1, 3, 3)
278 imshow(obrazSkorygowany2)
279 title('Obrazy po regulacji kontrastu2')
280
281 % Wyświetlenie histogramu przed korekcją
282 figure(18)
283 subplot(1, 2, 1)
284 imhist(obrazSzary)
285 title('Histogram przed korekcją')
286
287 % Korekcja gamma
288 gamma = 0.5; % Wartość gamma (dostosuj
289 wg potrzeb)
290 obrazSkorygowany = imadjust(obrazSzary,
291 [], [], gamma);
292
293 % Wyświetlenie histogramu po korekcji
294 subplot(1, 2, 2)
295 imhist(obrazSkorygowany)
296 title('Histogram po korekcji gamma')
297
298 % Wyświetlenie obrazów
299 figure(19)
300 subplot(1, 2, 1)
301 imshow(obrazSzary)
302 title('Obraz przed korekcją')
303
304 subplot(1, 2, 2)
305 imshow(obrazSkorygowany)
306 title('Obraz po korekcji gamma')
307
308 %% 5.      **Dokonać pomiaru różnicy
309 kolorów z wykorzystaniem funkcji deltaE
310 % oraz imcolordiff (można wykorzystać
311 obraz lub jego fragment z
312 % punktu 3 oraz jego wersję
313 zmodyfikowaną 3c).
314
315 % Obliczenie różnicy kolorów między
316 % obrazami
317 color_diff = imabsdiff(obrazRGB,
318 obrazModyfikowany);

```

```

307 color_diff2 = deltaE(obrazRGB,
308 obrazModyfikowany);
309
310 % Wyświetlenie różnicy kolorów
311 figure(20)
312 subplot(1, 2, 1)
313 imshow(color_diff);
314 title('Różnica kolorów (imabsdiff)');
315 subplot(1, 2, 2)
316 imshow(color_diff2);
317 title('Różnica kolorów (deltaE)');

```

Pierwsza część kodu dotyczy wyświetlania rozkładu składowych barw RGB we współrzędnych 3D dla obrazu dog.jpg. Początkowo, kod wczytuje obraz i wyodrębnia składowe czerwony (R), zielony (G) i niebieski (B) do osobnych zmiennych. Następnie, tworzy mapy kolorów dla każdej składowej barwy. Wartość nasycenia Z jest następnie używana do redukcji nasycenia dla każdej składowej barwy, co skutkuje obrazami z zredukowanym nasyceniem dla składowych R, G, i B. Te obrazy są następnie wyświetlane jako wykresy przestrzenne.

W drugiej części kodu, obraz RGB jest konwertowany do trzech różnych reprezentacji barwnych: HSV, Lab, i YCbCr. Te konwersje są użyteczne, ponieważ różne modele barwne mogą lepiej reprezentować pewne aspekty obrazu, takie jak percepcyjna jasność, nasycenie lub odcień. Wynikowe obrazy są następnie wyświetlane.

Trzecia część kodu obejmuje kilka operacji na obrazie barwnym. Obraz jest najpierw konwertowany do obrazu w odcieniach szarości. Następnie, jest tworzony profil barwny RGB dla wybranego fragmentu obrazu. Wybrany kanał barwny jest modyfikowany (w tym przypadku, poprzez dodanie wartości do kanału czerwonego), a następnie obrazy są skalowane z powrotem do obrazu RGB i wyświetlane. Kolejnym krokiem jest pseudokolorowanie obrazu dla dwóch różnych liczb kolorów (64 i 255), przy użyciu dwóch różnych map kolorów (parula i jet). Na końcu, z obrazu wyodrębniany jest wybrany kolor (w tym przypadku czerwony).

Czwarta część kodu skupia się na modyfikacji kontrastu obrazu. Zaczyna się od konwersji obrazu do obrazu w odcieniach szarości. Następnie, obraz jest wyświetlany przed regulacją kontrastu. Kontrast jest regulowany dla różnych wartości parametrów za pomocą funkcji imadjust. Na końcu, obrazy po regulacji kontrastu są wyświetlane.

3 Wyniki



figure 1

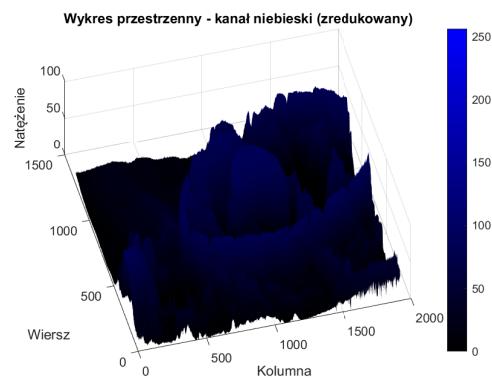


figure 4

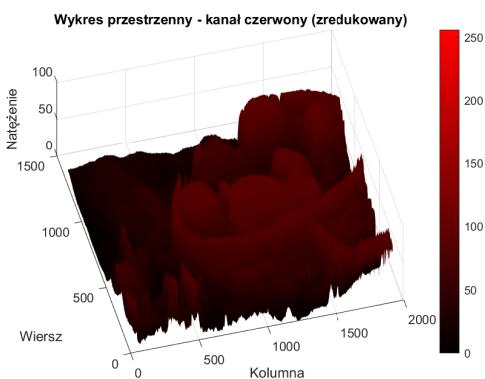


figure 2

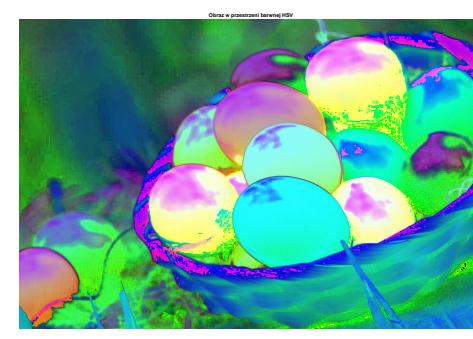


figure 6

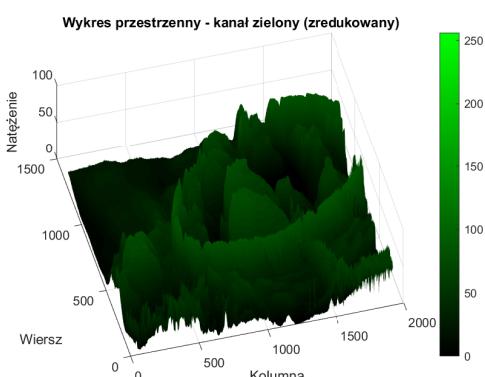


figure 3



figure 7



figure 8

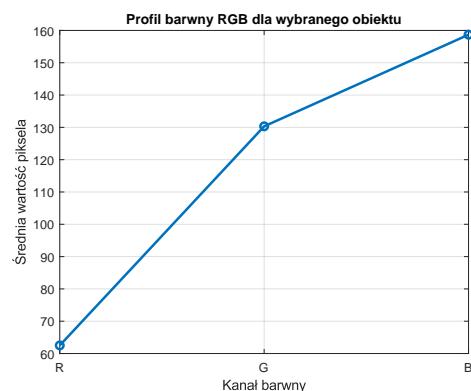


figure 12



figure 9



figure 13



figure 10



figure 14



figure 11

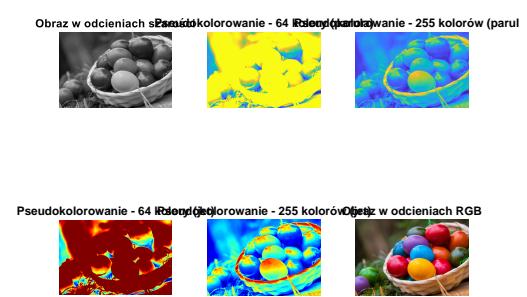


figure 15



Obraz oryginalny



Wyodrębniony kolor czerwony



Obraz przed korekcją



Obraz po korekcji gamma

figure 16

figure 19



figure 17



Różnica kolorów(imabsdiff)



Różnica kolorów(deltaE)

figure 20

4 Wnioski

Realizacja tego ćwiczenia pozwala studentowi wy ciągnąć następujące wnioski dotyczące obróbki obrazów:

1. Istnieje wiele przestrzeni barwnych, takich jak HSV, Lab, YCbCr, które oferują różne sposoby reprezentacji kolorów. Przełączanie się między tymi przestrzeniami może być przydatne w różnych zastosowaniach, takich jak analiza, przetwarzanie lub manipulacja obrazem.
2. Konwersja obrazu RGB na odcienie szarości (grayscale) jest jednym ze sposobów redukcji informacji barwnych, pozwalającym skupić się na strukturze i jasności obrazu.
3. Profil barwny RGB dla wybranego obiektu-/fragmentu obrazu umożliwia analizę średnich wartości pikseli dla poszczególnych kanałów barwnych. Może być używany do charakteryzacji barwy danego obiektu i porównywania go z innymi obiektyami.
4. Modyfikacja kanałów barwnych, takich jak dodawanie, odejmowanie, mnożenie itp., pozwala na manipulację kolorami obrazu i uzyskanie różnych efektów wizualnych.

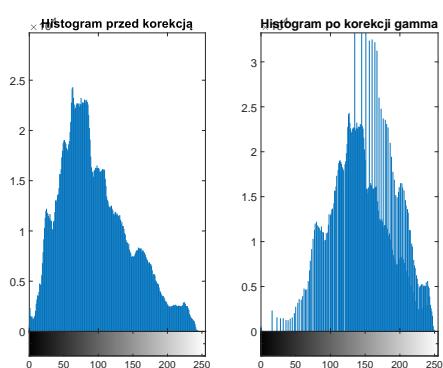


figure 18

5. Pseudokolorowanie obrazów pozwala na przy-pisanie kolorów do wartości odcieni szarości, co może ułatwić percepcję różnic w intensyw-nościach pikseli.
6. Wyodrębnianie określonych kolorów na obrazie może być użyteczne w analizie obrazów i wy-odrębnianiu obiektów o konkretnych cechach barwnych.
7. Regulacja kontrastu obrazu może poprawić jego jakość wizualną i uwypuklić szczegóły w obrazie. Różne wartości parametrów regulacji kontrastu mogą prowadzić do różnych efektów wizualnych.
8. Pomiar różnicy kolorów z wykorzystaniem funkcji deltaE i imcolordiff pozwala na analizę różnic pomiędzy dwoma obrazami lub ich frag-mentami. Może być przydatne w analizie kolorystycznej, porównywaniu obrazów lub iden-tyfikacji różnic między nimi.

Ogólnie rzecz biorąc, to ćwiczenie pozwala studen-towi na zdobycie praktycznego doświadczenia w obróbce obrazów, eksploracji różnych przestrzeni barwnych, regulacji kontrastu, wyodrębnianiu kolo-rów i analizie różnic kolorystycznych. Jest to cenne w zrozumieniu podstawowych technik prze-twarzania obrazów i ich zastosowań w różnych dzie-dzinach.