

# **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
36 r3=0:1:255;
37 rr3=r3';
38
39 z3=zeros(256,3);
40 z3(:,3)=rr3;
41 blue=z3/255;
42
43 % Wartość nasycenia do osiągnięcia (Z)
44 Z = 90;
45
46 % Zmniejszenie nasycenia dla
47 % poszczególnych kanałów barwy
48 obrazR_redukowany = obrazR * (Z/255);
49 obrazG_redukowany = obrazG * (Z/255);
50 obrazB_redukowany = obrazB * (Z/255);
51
52 % Wygenerowanie wykresów 3D dla
53 % składowych kolorów
54 figure(2)
55 mesh(1:k, w:-1:1, obrazR_redukowany,
56 'FaceColor', 'texturemap', 'EdgeColor',
57 'none', 'CdataMapping', 'direct');
58 colormap(red)
59 view([-15 70]);
60 colorbar('vert');
61 title('Wykres przestrzenny - kanał
62 czerwony (zredukowany)');
63 xlabel('Kolumna'); ylabel('Wiersz');
64 zlabel('Natężenie');
65
66 figure(3)
67 surf(1:k, w:-1:1, obrazG_redukowany,
68 'FaceColor', 'texturemap', 'EdgeColor',
69 'none', 'CdataMapping', 'direct');
70 colormap(green)
71 view([-15 70]);
72 colorbar('vert');
73 title('Wykres przestrzenny - kanał
74 zielony (zredukowany)');
75 xlabel('Kolumna'); ylabel('Wiersz');
76 zlabel('Natężenie');
77
78 %% 2. Dla jednego, wybranego
79 % obrazu RGB dokonać konwersji do 3
80 % wybranych innych
81 % reprezentacji barwnych. Wyniki w
82 % postaci obrazów umieścić w sprawozdaniu.
83 Scharakteryzować
84 % dla każdego modelu barwnego rodzaj
85 % pozyskiwanej informacji barwnej.
86
87 % 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
88 % reprezentacjach barwnych
89 figure(5)
90 imshow(obrazRGB)
91 title('Obraz RGB')
92
93 figure(6)
94 imshow(obrazHSV)
95 title('Obraz w przestrzeni barwnej HSV')
96 xlabel('Odcień'); ylabel('Nasyście');
97 zlabel('Wartość')
98
99 figure(7)
100 imshow(obrazLab)
101 title('Obraz w przestrzeni barwnej Lab')
102 xlabel('Jasność'); ylabel('a');
103 zlabel('b')
104
105 figure(8)
106 imshow(obrazYCbCr)
107 title('Obraz w przestrzeni barwnej
108 YCbCr')
109 xlabel('Luminancja');
110 ylabel('Chrominancja Cb');
111 zlabel('Chrominancja Cr')
112
113 %% 3. Dla dowolnego obraz
114 %a) dokonać konwersji do obrazu w
115 %odcieniach szarości (rgb2gray)
116
117 % Konwersja na obraz w odcieniach
118 %szarości
119 obrazGray = rgb2gray(obrazRGB);
120
121 % Wyświetlanie obrazów
122 figure(9)
123 imshow(obrazRGB)
124 title('Obraz barwny')
125
126 figure(10)
127 imshow(obrazGray)
128 title('Obraz w odcieniach szarości')
129
130 %% b) wyświetlić profil barwny
131 %RGB dla wybranego obiektu/fragmentu
132 %obrazu;
133
134 % Wyświetlanie obrazu
135 figure(11)
136 imshow(obrazRGB)
137 title('Obraz barwny')
138
139 % Wybór obszaru zainteresowania
140 h = imrect; % Narzędzie do wyboru
141 %obszaru prostokątnego
142 position = wait(h);
143 x = round(position(1));
144 y = round(position(2));
145 width = round(position(3));
146 height = round(position(4));
147
148 % Wyodrębnienie obszaru zainteresowania
149 obszarZainteresowania =
150 obrazRGB(y:y+height-1, x:x+width-1, :);
151
152 % Obliczenie średnich wartości dla
153 %każdego kanału barwnego
154 sredniaR =
155 mean(mean(obszarZainteresowania(:,:,1)));
156 sredniaG =
157 mean(mean(obszarZainteresowania(:,:,2)));
158 sredniaB =
159 mean(mean(obszarZainteresowania(:,:,3)));
160
161 % Wyświetlanie profilu barwnego RGB
162 figure(12)
163 plot([1, 2, 3], [sredniaR, sredniaG,
164 sredniaB], 'o-', 'LineWidth', 2)
165 title('Profil barwny RGB dla wybranego
166 obiektu')
167 xlabel('Kanał barwny')
168 ylabel('Średnia wartość piksela')
169 set(gca, 'XTick', [1, 2, 3],
170 'XTickLabel', {'R', 'G', 'B'})
171 grid on
172
173 %% c) zmodyfikować wybrany kanał
174 %barwny (np. poprzez odjęcie, dodanie,
175 %mnożenie itd.
176 % elementów w macierzy), a następnie
177 %za pomocą funkcji cat scalić
178 %macierze; wyświetlić obraz;
179
180 %% Wybrany kanał barwny (np. czerwony -
181 %R)
182 kanal = obrazRGB(:,:,1);
183
184 %% Modyfikacja kanału barwnego (np.
185 %dodanie wartości)
186 modyfikacja = kanal + 150;
187
188 %% Scalenie macierzy
189 obrazModyfikowany = cat(3, modyfikacja,
190 obrazRGB(:,:,2), obrazRGB(:,:,3));
191
192 % Wyświetlanie obrazu
193 figure(13)
194 imshow(obrazRGB)
195 title('Obraz oryginalny')
196
197 figure(14)
198 imshow(obrazModyfikowany)
199 title('Obraz po modyfikacji')

```

```

174
175    %% d)      dokonać pseudokolorowania
176    %% (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    %% e)      * wyodrębnić na obrazie
222    %% wybrany kolor.
223
224    % Konwersja obrazu do obrazu w
225    % odcieniach szarości
226    obrazSzary = rgb2gray(obrazRGB);
227
228    % Wyodrębnienie koloru czerwonego
229    maskaCzerwony = (obrazRGB(:,:,1) >
230    progR) & (obrazRGB(:,:,2) > progG) &
231    (obrazRGB(:,:,3) > progB);
232
233    % Wyodrębnienie koloru czerwonego na
234    % obrazie
235    obrazWyodrebniony = obrazRGB;
236    obrazWyodrebniony(repmat(~maskaCzerwony, [1,1,3])) =
237    0;
238
239    % Wyświetlanie obrazów
240    figure(16)
241    subplot(1, 2, 1)
242    imshow(obrazRGB)
243    title('Obraz oryginalny')
244
245    subplot(1, 2, 2)
246    imshow(obrazWyodrebniony)
247    title('Wyodrębniony kolor czerwony')
248
249    %% 4.      *Zmodyfikować kontrast
250    %% dowolnego obrazu (funkcja imadjust).
251    %% Sprawdzić różne wartości parametrów
252    %% tej funkcji. Dla wybranego przypadku
253    %% porównać
254    %% histogramy barw przed korekcją i po
255    %% korekcji gamma.
256
257    % Konwersja obrazu do obrazu w
258    % odcieniach szarości
259    obrazSzary = rgb2gray(obrazRGB);
260
261    % Wyświetlenie obrazu przed regulacją
262    % kontrastu
263    figure(17)
264    subplot(1, 3, 1)
265    imshow(obrazSzary)
266    title('Obraz przed regulacją kontrastu')
267
268    % Regulacja kontrastu dla różnych
269    % wartości parametrów
270    low_in = 0.2; % Dolne ograniczenie dla
271    % wartości wejściowych
272    high_in = 0.8; % Górnne ograniczenie dla
273    % wartości wejściowych
274    low_out = 0; % Dolne ograniczenie dla
275    % wartości wyjściowych
276    high_out = 1; % Górnne ograniczenie dla
277    % wartości wyjściowych
278
279    obrazSkorygowany1 = imadjust(obrazSzary,
280    [low_in high_in], [low_out high_out]);
281
282    low_in = 0.4; % Dolne ograniczenie dla
283    % wartości wejściowych

```

```

221    progG = 0;    % Dolny próg dla składowej
222    G
223    progB = 0;    % Dolny próg dla składowej
224    B
225
226    % Utworzenie maski dla koloru czerwonego
227    maskaCzerwony = (obrazRGB(:,:,1) >
228    progR) & (obrazRGB(:,:,2) > progG) &
229    (obrazRGB(:,:,3) > progB);
230
231    % Wyodrębnienie koloru czerwonego na
232    % obrazie
233    obrazWyodrebniony = obrazRGB;
234    obrazWyodrebniony(repmat(~maskaCzerwony, [1,1,3])) =
235    0;
236
237    % Wyświetlanie obrazów
238    figure(16)
239    subplot(1, 2, 1)
240    imshow(obrazRGB)
241    title('Obraz oryginalny')
242
243    subplot(1, 2, 2)
244    imshow(obrazWyodrebniony)
245    title('Wyodrębniony kolor czerwony')
246
247    %% 4.      *Zmodyfikować kontrast
248    %% dowolnego obrazu (funkcja imadjust).
249    %% Sprawdzić różne wartości parametrów
250    %% tej funkcji. Dla wybranego przypadku
251    %% porównać
252    %% histogramy barw przed korekcją i po
253    %% korekcji gamma.
254
255    % Konwersja obrazu do obrazu w
256    % odcieniach szarości
257    obrazSzary = rgb2gray(obrazRGB);
258
259    % Wyświetlenie obrazu przed regulacją
260    % kontrastu
261    figure(17)
262    subplot(1, 3, 1)
263    imshow(obrazSzary)
264    title('Obraz przed regulacją kontrastu')
265
266    % Regulacja kontrastu dla różnych
267    % wartości parametrów
268    low_in = 0.2; % Dolne ograniczenie dla
269    % wartości wejściowych
270    high_in = 0.8; % Górnne ograniczenie dla
271    % wartości wejściowych
272    low_out = 0; % Dolne ograniczenie dla
273    % wartości wyjściowych
274    high_out = 1; % Górnne ograniczenie dla
275    % wartości wyjściowych
276
277    obrazSkorygowany1 = imadjust(obrazSzary,
278    [low_in high_in], [low_out high_out]);
279
280    low_in = 0.4; % Dolne ograniczenie dla
281    % wartości wejściowych

```

```

263 high_in = 0.6; % Górnne ograniczenie dla
264 % wartości wejściowych
265
266 obrazSkorygowany2 = imadjust(obrazSzary,
267 [low_in high_in], [low_out high_out]);
268
269 % Wyświetlenie obrazów po regulacji
270 % kontrastu
271 subplot(1, 3, 2)
272 imshow(obrazSkorygowany1)
273 title('Obrazy po regulacji kontrastu1')
274
275 subplot(1, 3, 3)
276 imshow(obrazSkorygowany2)
277 title('Obrazy po regulacji kontrastu2')
278
279 % Wyświetlenie histogramu przed korekcją
280 figure(18)
281 subplot(1, 2, 1)
282 imhist(obrazSzary)
283 title('Histogram przed korekcją')
284
285 % Korekcja gamma
286 gamma = 0.5; % Wartość gamma (dostosuj
287 % wg potrzeb)
288 obrazSkorygowany = imadjust(obrazSzary,
289 [], [], gamma);
290
291 % Wyświetlenie histogramu po korekcji
292 subplot(1, 2, 2)
293 imhist(obrazSkorygowany)
294 title('Histogram po korekcji gamma')
295
296 % Wyświetlenie obrazów
297 figure(19)
298 subplot(1, 2, 1)
299 imshow(obrazSzary)
300 title('Obraz przed korekcją')
301
302 subplot(1, 2, 2)
303 imshow(obrazSkorygowany)
304 title('Obraz po korekcji gamma')
305
306 %% 5.      **Dokonać pomiaru różnicy
307 % kolorów z wykorzystaniem funkcji deltaE
308 % oraz imcolordiff (można wykorzystać
309 % obraz lub jego fragment z
310 % punktu 3 oraz jego wersję
311 % zmodyfikowaną 3c).
312
313 % Obliczenie różnicy kolorów między
314 % obrazami
315 color_diff = imabsdiff(obrazRGB,
316 obrazModyfikowany);
317 color_diff2 = deltaE(obrazRGB,
318 obrazModyfikowany);
319
320 % Wyświetlenie różnicy kolorów
321 figure(20)
322 subplot(1, 2, 1)
323 imshow(color_diff);

```

```

313 title('Różnica kolorów (imabsdiff)');
314
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

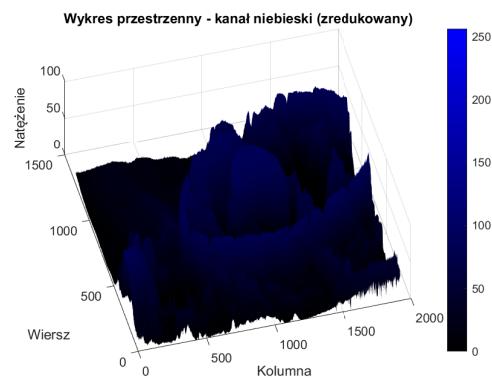


figure 4

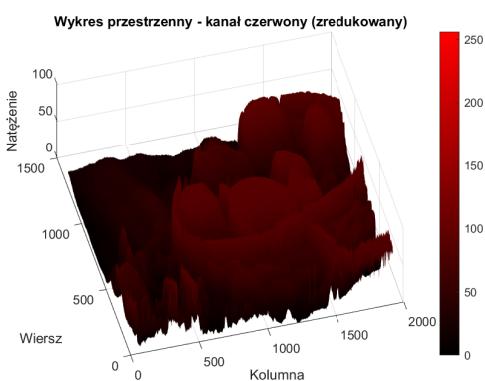


figure 2

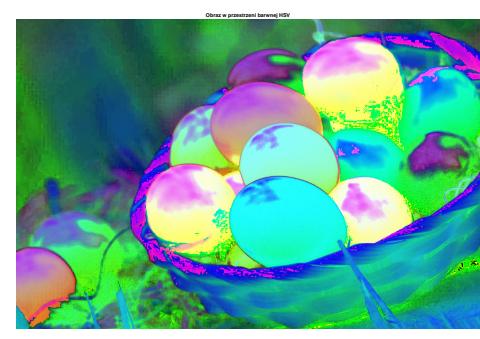


figure 6

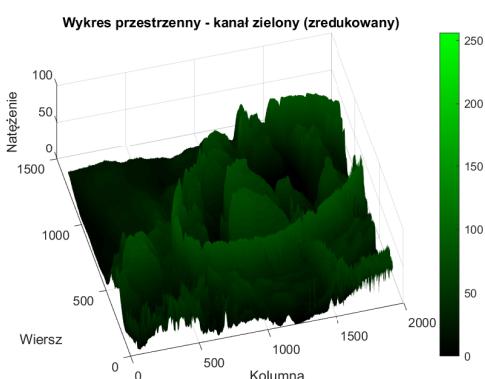


figure 3



figure 7



figure 8

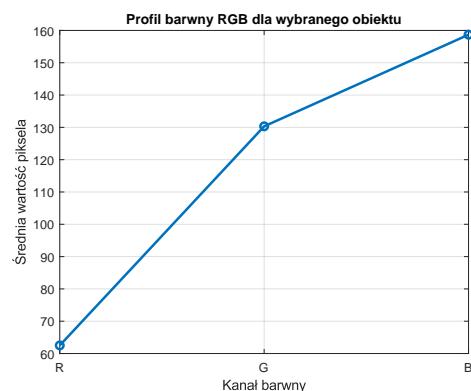


figure 12



figure 9



figure 13



figure 10



figure 14



figure 11

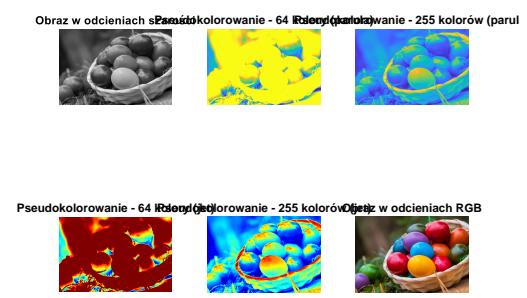


figure 15



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.

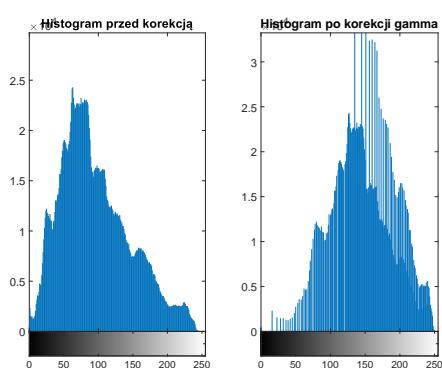


figure 18

5. Pseudokolorowanie obrazów pozwala na przy-pisanie kolorów do wartości odcienni 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.