

# **Wprowadzenie do grafiki maszynowej(LAB-5)**

Paweł Idzikowski, nr.albumu: 141080

27 listopada 2018

# **Spis treści**

<b>1</b>	<b>Informacje wstępne</b>	<b>3</b>
<b>2</b>	<b>Zadanie 1</b>	<b>4</b>
2.1	Wnioski . . . . .	5
2.1.1	Odejmowanie obrazów . . . . .	5
2.1.2	Dodawanie obrazów . . . . .	7
2.1.3	Dzielenie obrazów . . . . .	9
2.1.4	Operacja logiczna - AND . . . . .	12
2.1.5	Operacja logiczna - OR . . . . .	14
2.1.6	Operacja logiczna - XOR . . . . .	16
2.1.7	Operacja logiczna - NAND . . . . .	18
2.1.8	Operacja logiczna - NOR . . . . .	20
2.1.9	Operacja logiczna - XNOR . . . . .	22
<b>3</b>	<b>Zadanie 2</b>	<b>24</b>
3.1	Wnioski . . . . .	25
3.1.1	Metoda wbudowana - imabsdiff . . . . .	25
3.1.2	Metoda odejmowania . . . . .	27
<b>4</b>	<b>Zadanie 3</b>	<b>29</b>
4.1	Wnioski . . . . .	30

## 1 Informacje wstępne

Proszę nie zwracać uwagi na liczbę stron dokumentacji. Sprawozdanie zawiera dużą liczbę obrazków. Chciałem aby przy każdym przekształceniu były dostępne blisko oryginalne obrazki aby móc porównać sobie wyniki i nie jeździć na np. początek sprawozdania. Fotografie, które wykorzystałem w sprawozdaniu:



Rysunek 1: Call of Duty: Modern Warfare 2(gra)



Rysunek 2: Szeregowiec Ryan(film)



Rysunek 3: Dunkierka(film)



Rysunek 4: Furia(film) - wrzucam w tym miejscu tylko jedną fotografię ponieważ musiałem zredukować rozmiar pliku PDF!



Rysunek 5: The Last of Us(gra)

## 2 Zadanie 1

### Treść zadania:

Wybierz dwie pary obrazów, najlepiej podobnych do siebie (w parze) i wykonaj operację odejmowania, dzielenia, dodawania, a następnie przeprowadź operacje logiczne and, or i xor.

### Rozwiązańe:

```
// Zadanie1
// wczytaj obrazki , PARA1
obrazek1=imread(fullfile("C:/Users/Idzik/Desktop/Grafika maszynowa/SPR05/
23.11 [spr]/sprawozdanie/" + 'zad1-paral-1.jpg'));
obrazek2=imread(fullfile("C:/Users/Idzik/Desktop/Grafika maszynowa/SPR05/
23.11 [spr]/sprawozdanie/" + 'zad1-paral-2.jpg'));

// wczytaj obrazki , PARA2
//obrazek1=imread(fullfile("C:/Users/Idzik/Desktop/Grafika maszynowa/SPR05/
23.11 [spr]/sprawozdanie/" + 'zad1-para2-1.jpg'));
//obrazek2=imread(fullfile("C:/Users/Idzik/Desktop/Grafika maszynowa/SPR05/
23.11 [spr]/sprawozdanie/" + 'zad1-para2-2.jpg'));

// ***** ODEJMOWANIE *****
// jezeli obrazek sa tego samego rozmiaru imsubtract odejmuje
// imsubtract(im1,im2);
// element z im2 z korespondujacym mu elementem z im1
// scf(0); imshow(obrazek1); xtitle("Oryginalny obrazek 1");
// scf(1); imshow(obrazek2); xtitle("Oryginalny obrazek 2");
// scf(2); imshow(imsubtract(obrazek2,obrazek1)); xtitle("Wynik obrazek1 - obrazek2");
// scf(3); imshow(imsubtract(obrazek1,obrazek2)); xtitle("Wynik obrazek2 - obrazek1");

// ***** DODAWANIE *****
// scf(0); imshow(imadd(obrazek2,obrazek1)); xtitle("Efekt po dodaniu");

// ***** DZIELENIE *****
// scf(0); imshow(imdivide(obrazek2,obrazek1)); xtitle("Dzielenie obrazek2/obrazek1");
// scf(1); imshow(imdivide(obrazek1,obrazek2)); xtitle("Dzielenie obrazek1/obrazek2");

// ***** MNOZENIE *****
scf(0); imshow(immultiply(obrazek2,obrazek1)); xtitle("Efekt po wymnozeniu");

// ***** AND *****
//nowy=obrazek1 & obrazek2; xtitle("Operacja logiczna AND");
// scf(0); imshow(nowy);

// ***** OR *****
//nowy=obrazek1 | obrazek2;
// scf(0); imshow(nowy); xtitle("Operacja logiczna OR");

// ***** XOR *****
// (A & (~B)) | ((~A) & B)
//nowy= (obrazek1 & (~obrazek2)) | ((~obrazek1) & obrazek2);
// scf(0); imshow(nowy); xtitle("Operacja logiczna XOR");

// ***** NAND *****
// ~ (A & B)
//nowy= ~(obrazek1 & obrazek2); xtitle("Operacja logiczna NAND");
// scf(0); imshow(nowy);

// ***** NOR *****
// ~ (A | B)
//nowy= ~(obrazek1 | obrazek2); xtitle("Operacja logiczna NOR");
// scf(0); imshow(nowy);

// ***** XNOR *****
// (A & B) | ((~A) & (~B))
//nowy= (obrazek1 & obrazek2) | ((~obrazek1) & (~obrazek2)); xtitle("Operacja logiczna XNOR");
// scf(0); imshow(nowy);

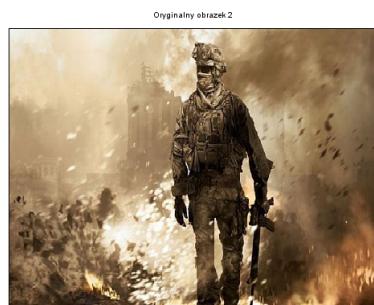
// Ciekawa funkcja jest imfuse - testujemy to
//y=imfuse(obrazek1,obrazek2,'max'); xtitle("Obrazek po funkcji imfuse");
//imshow(y);
```

## 2.1 Wnioski

### 2.1.1 Odejmowanie obrazów



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt *obrazek2 – obrazek1*



(d) efekt *obrazek1 – obrazek2*

Rysunek 6: Para 1



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt *obrazek2 – obrazek1*



(d) efekt *obrazek1 – obrazek2*

Rysunek 7: Para 2

Odejmowanie obrazków jest (patrząc na wszystkie wykonane operacje) jedną z ciekawszych operacji obok dodawania, mnożenia i dzielenia. Pierwsza para obrazków dotycząca gry Call of Duty: Modern Warfare 2, otrzymała fajne efekty (podpunkty c i d w [Rysunek6](#)), czego nie można powiedzieć o drugiej parze dotyczącej filmu Szeregowiec Ryan... Tam (według mnie) operacja imsubtract nie dała dobrego efektu - trudno czegokolwiek się dopatrzyć. Podejrzewam, że obrazki z gry Call Of Duty zawierają mniejszą paletę wykorzystanych kolorów (mamy w pierwszej parze tak naprawdę różne odcienie żółtego) stąd efekt odejmowania jest bardziej 'efektowny'.

### 2.1.2 Dodawanie obrazów



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt *obrazek1 + obrazek2*

Rysunek 8: Para 1



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt *obrazek1 + obrazek2*

Rysunek 9: Para 2

W przypadku dodania do siebie obrazków sytuacja jest podobna jak w odejmowaniu. Gama kolorów obrazku z Call of Duty jest mniejsza i dodawanie daje ciekawy efekt, jakby były 'dwie wizje' żołnierza (podpunkt c [Rysunek 8](#)) - coś w tym stylu. Dodawanie w przypadku drugiej pary obrazków daje wynik, który w oddaleniu przypomina mi zdjęcie do gazety albo zdjęcie nadające się na okładkę filmu (podpunkt c [Rysunek 9](#)) - rezultat przypomina nałożenie na siebie kilku klatek z filmu.

### 2.1.3 Dzielenie obrazów



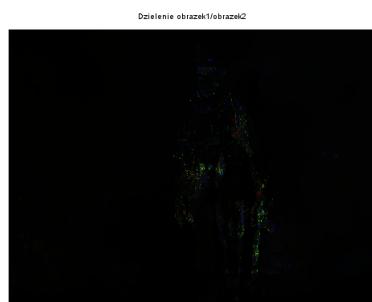
(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt *obrazek2/obrazek1*



(d) efekt *obrazek1/obrazek2*

Rysunek 10: Para 1



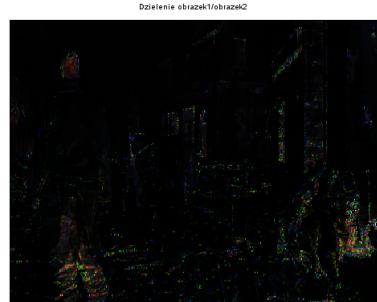
(a) efekt *obrazek2/obrazek1*



(b) Oryginalny obrazek 2



(c) efekt *obrazek2/obrazek1*



(d) efekt *obrazek1/obrazek2*

Rysunek 11: Para 2

	Var - obrazek1	Var - obrazek2	1	2	3	4	5	6	7	8	9	10	11
1	192	192	191	191	191	191	192	194	196	195	194	194	
2	190	190	190	190	191	191	191	192	193	194	194	194	
3	189	189	190	191	192	192	193	193	193	193	192	193	
4	189	188	189	190	192	193	193	193	193	191	191	191	
5	189	188	188	188	190	190	190	190	190	190	189	190	
6	191	190	189	189	189	188	188	188	187	189	189	189	
7	189	190	190	190	189	188	187	187	189	189	190		
8	186	187	189	190	189	188	188	187	187	189	189	190	
9	187	194	186	188	188	186	194	194	187	189	186	191	
10	190	188	185	189	191	187	188	186	186	184	190	192	
11	184	187	193	190	187	189	189	193	191	185	187		
12	189	183	184	187	188	187	183	186	182	190	185		
13	200	202	195	188	186	188	191	187	188	190	190		
14	150	170	184	189	189	190	197	195	194	193	191		
15	97	95	104	114	118	117	120	131	146	156	199		
16	80	91	105	92	90	104	96	91	87	90	170		
17	95	65	99	121	107	113	119	118	117	100	129		
18	102	103	69	80	105	89	87	97	116	103	104		
19	78	109	120	89	79	90	86	86	90	99	81		
20	98	95	113	121	101	99	108	98	97	90	94		

(a) Macierz obrazka 1

	Var - obrazek1	Var - obrazek2	1	2	3	4	5	6	7	8	9	10	11
1	41	153	150	153	146	148	149	149	150	155	155	150	
2	39	166	162	156	151	150	147	145	146	153	149		
3	27	153	138	127	134	138	139	135	132	143	143		
4	48	166	150	144	146	143	147	147	147	150	150		
5	29	138	136	149	145	131	131	132	148	149	148		
6	43	146	153	171	151	133	133	144	147	150	152		
7	30	138	148	159	128	120	126	149	155	153	150		
8	36	136	134	145	123	127	116	125	132	135	141		
9	38	155	149	154	145	143	157	156	157	165	166		
10	28	134	129	126	126	123	137	132	134	141	136		
11	14	102	94	92	94	93	101	99	113	121	121		
12	15	89	80	90	86	88	90	100	99	104	110		
13	18	82	73	82	78	83	90	106	100	103	114		
14	10	79	81	83	81	80	85	87	95	95	101		
15	19	82	76	72	69	75	89	90	88	92	100		
16	17	81	73	82	77	90	101	103	107	109	113		
17	14	79	78	86	84	88	106	106	112	116	119		
18	15	79	75	82	83	87	104	107	111	114	118		
19	12	81	75	80	83	89	102	107	109	115	119		
20	9	79	74	79	86	91	98	104	110	118	122		

(b) Macierz obrazka 2

Rysunek 12: Macierze obrazów

Operacja dzielenia w pierwszej parze szału nie robi([Rysunek 10](#)). Na obrazku(c) nic ciekawego się nie dzieje. Zwróćmy jednak uwagę na wynik podzielenia obrazka 1(a) przez obrazek 2(b). Otrzymaliśmy coś w stylu wykrywania krawędzi. Dosyć dobrze zostało odwzorowany żołnierz ale najbardziej rzuca się w oczy rekonstrukcja karabinu.

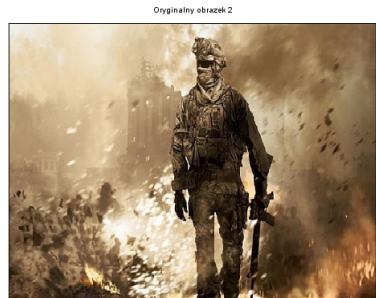
Warto w przypadku tej operacji skupić się na drugiej parze obrazków([Rysunek 11](#)). Tak jak wcześniej wspominaliśmy, obrazki pary drugiej są wykonane w wiele barwach przez co w operacji dzielenia otrzymaliśmy ciekawe obrazki, które mógłbym zatytuować jako 'miasto w nocy'(szczególnie obrazek z podpunktu d).

Jeżeli przyjrzymy się na macierze obrazków dotyczących filmu szeregowca Ryana(znajdują się niżej, [Rysunek 12](#)) to zauważymy, że w przypadku podzielenia obrazku1 przez obrazek2 wynik będzie większy niż 1 i otrzymujemy więcej ciekawych kolorów. W przypadku operacji odwrotnej, czyli obrazek2/obrazek1 dostalibyśmy więcej liczby w przedziale 0-0,999 stąd w efekcie nasz rysunek(podpunkt c) ma dużo kolorów bliskich czerni.

#### 2.1.4 Operacja logiczna - AND



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt *obrazek1 & obrazek2*

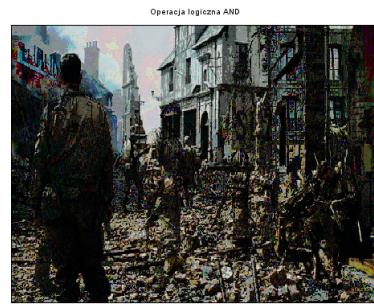
Rysunek 13: Para 1



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt *obrazek1 & obrazek2*

Rysunek 14: Para 2

Pierwsza operacja logiczna AND miło zaskoczyła - efekty w przypadku obu par fotografii są zadowalające. Obrazki z podpunktu (c - rysunki 15 i 16) sprawiają wrażenie jakby były namalowane ręcznie...

### 2.1.5 Operacja logiczna - OR



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt  $obrazek1 \parallel obrazek2$

Rysunek 15: Para 1



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt  $obrazek1 \parallel obrazek2$

Rysunek 16: Para 2

Operacja logiczna OR z kolei nie daje żadnych ciekawych efektów na których warto 'zawiesić' oko. Rezultaty wyglądają jakby grafiki przygotowała osoba, która dopiero zaczyna zabawę z edycją grafik.

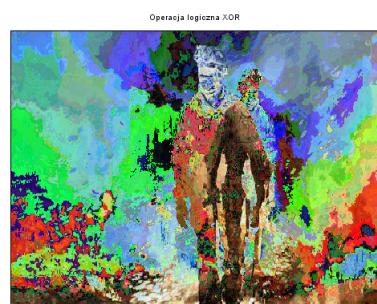
### 2.1.6 Operacja logiczna - XOR



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt  $obrazek1 \oplus obrazek2$

Rysunek 17: Para 1



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt  $obrazek1 \oplus obrazek2$

Rysunek 18: Para 2

Operacja XOR zwróciła moją uwagę szczególnie w przypadku pary pierwszej zdjęć([Rysunek 19](#)). Jak wspominaliśmy wcześniej, ta para zdjęć nie ma za dużo barw a jedynie różne odcienie koloru żółtego.. Operacja XOR między tymi fotografiemi spowodowała, że dostaliśmy informację w postaci barwy kolorów jaki poziom jasności ma dana barwa. Widzimy, że np. na rysunku z podpunktu (a, [Rysunek 19](#)), gdzie w prawym dolnym rogu mamy mocny kolor płomieni otrzymujemy w wyniku(c, [Rysunek 19](#)) barwę czerwoną. Jest to bardzo ciekawe zachowanie. Nie można tego powiedzieć w przypadku drugiej pary. Po raz kolejny duża różnorodność kolorów spowodowała, że XOR nie dał żadnych ciekawych efektów.

### 2.1.7 Operacja logiczna - NAND



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt  $obrazek1 \neg \& obrazek2$

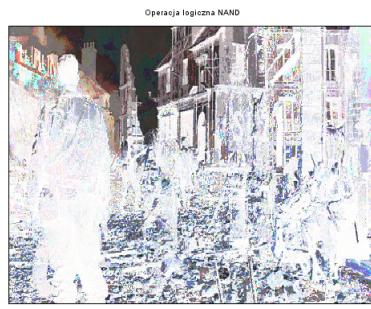
Rysunek 19: Para 1



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt  $obrazek1 \neg \& obrazek2$

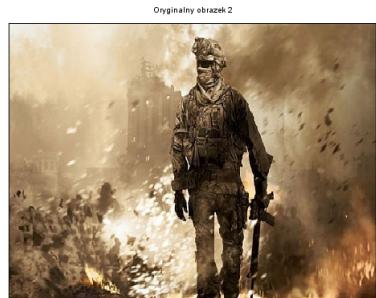
Rysunek 20: Para 2

Połączenie obrazków za pomocą operacji NAND daje w obu przypadkach ciekawe efekty wizualne. Mógłbym pokusić się tu nawet o stwierdzenie, że są to efekty w stylu nadania gradientu dla obrazów i potem 'wypikselowania' ich(znowu tutaj kieruję się doświadczeniem z przeglądarkowego photoshopa zwanego pixlr, który swoją drogą jest fajnym edytorem grafiki...(reklama)).

### 2.1.8 Operacja logiczna - NOR



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt  $obrazek1 \neg\parallel obrazek2$

Rysunek 21: Para 1



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt  $obrazek1 \neg \parallel obrazek2$

Rysunek 22: Para 2

Historia taka sama jak w przypadku NAND'a. Operacja NOR także zasługuje na miejsce w zakładce 'efektów' do zdjęć. Co tu dużo mówić.. 'jest efekciarska' i w połączeniu z kombinacjami innych operacji może dać fajne wyniki :).

### 2.1.9 Operacja logiczna - XNOR



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt  $obrazek1 \neg \oplus obrazek2$

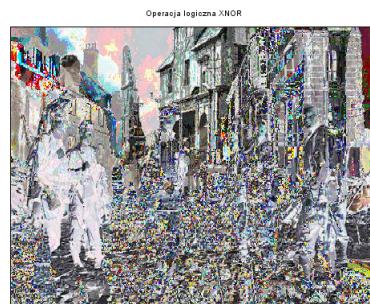
Rysunek 23: Para 1



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt  $obrazek1 \neg \oplus obrazek2$

Rysunek 24: Para 2

Ostatnia operacja, którą przetestowałem na obrazkach czyli XNOR daje rezultaty jak jego 'kolega po fachu' czyli XOR tylko w przypadku pierwszej pary(z gry Call of duty) używa innych barw(ciemniejszych) natomiast obrazki dotyczące filmu Szeregowiec Ryan nadal prezentują się słabo..

### 3 Zadanie 2

#### Treść zadania:

Gdybyś chciał/chciała automatycznie wyszukać różnicę w obrazach (na przykład przy obróbce filmu z kamery) jakiej metody byś użył/użyła? Zastosuj ją na 2 parach podobnych do siebie obrazów i wyświetl rezultat.

#### Rozwiążanie:

```
// Zadanie2
// wczytaj obrazki(1)
//obrazek1=imread(fullfile("C:/Users/Idzik/Desktop/Grafika maszynowa/SPR05/
23.11 [spr]/sprawozdanie/" + 'zad2para1-1.png'));
//obrazek2=imread(fullfile("C:/Users/Idzik/Desktop/Grafika maszynowa/SPR05/
23.11 [spr]/sprawozdanie/" + 'zad2para1-2.png'));

// 2
obrazek1=imread(fullfile("C:/Users/Idzik/Desktop/Grafika maszynowa/SPR05/
23.11 [spr]/sprawozdanie/" + 'zad2para2-1.png'));
obrazek2=imread(fullfile("C:/Users/Idzik/Desktop/Grafika maszynowa/SPR05/
23.11 [spr]/sprawozdanie/" + 'zad2para2-2.png'));

// testy
//scf(0); imshow(obrazek1);
//scf(1); imshow(obrazek2);

// modull wykrywajacy ruch
//imshow(imabsdiff(obrazek1,obrazek2)); xtitle("Metoda predefiniowana imabsdiff");

// metoda odejmowania jednego od drugiego
//scf(0); imshow(imsubtract(obrazek1,obrazek2)); xtitle("Obrazek2 - obrazek1");
//scf(1); imshow(imsubtract(obrazek2,obrazek1)); xtitle("Obrazek1 - obrazek2");

// METODA 2 - dosyć dobrze pokazuje różnice jednak ciekwo by było
// odróżnić kto jest człowiekiem gdybysmy mieli dodatkowo inne obiekty
// na obrazkach
binarny1 = im2bw(obrazek1, 0.3);
binarny2 = im2bw(obrazek2, 0.3);
//imshow(imabsdiff(binarny1, binarny2));
```

### 3.1 Wnioski

#### 3.1.1 Metoda wbudowana - imabsdiff



(a) Oryginalny obrazek 1



(b) Oryginalny obrazek 2



(c) efekt *imabsdiff*



(d) Oryginalny obrazek 1



(e) Oryginalny obrazek 2



(f) efekt *imabsdiff*

Metoda wbudowana jest moją ulubioną metodą do tego aby sprawdzić różnice w klatkach i to nie dlatego, że jest wbudowana... Predefiniowana funkcja porównuje fotografie i w rezultacie otrzymujemy informacje o tym co się zmieniło. W przypadku pierwszej pary(obrazków z filmu Dunkierka), metoda dobrze 'zauważała', że nagle z lewej strony pojawił się żołnierz, którego nie było, drugi od lewej zmienił kierunek patrzenia, ten z prawej strony - zmienił pozycje i zza jego pleców pokazał się kolejny żołnierz. Wszystko to zostało zarejestrowane przez wbudowaną metodę i powiem szczerze, że jestem zaskoczony z otrzymanych rezulatów.

Jeżeli chodzi o drugą parę obrazków(obrazki z filmu Furia) bardzo podoba mi się to, że metoda wyszczególniła tylko miejsce, które się zmieniło czyli środek kadru(ale może to tylko przypadek..?)! Odnosiwał dobre zmiany w kadrach czyli, że na drugim planie pojawił się żołnierz i dowódca zmienił pozycję swojej głowy. Pozostałe rzeczy czyli tło i tyły głowy ze skrajów obrazka nie zmieniły się co nie zostało przez niego wyszczególnione!

### 3.1.2 Metoda odejmowania



(g) Oryginalny obrazek 1



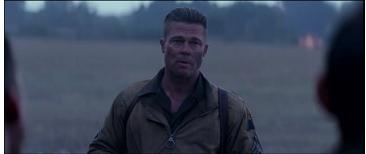
(h) Oryginalny obrazek 2



(i) Efekt po operacji:  
 $obrazek2 - obrazek1$



(j) Efekt po operacji:  
 $obrazek1 - obrazek2$



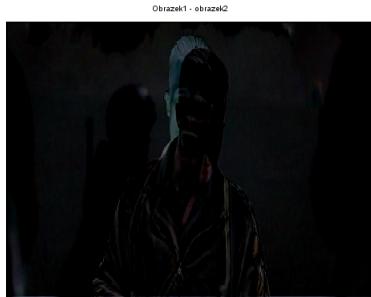
(k) Oryginalny obrazek 1



(l) Oryginalny obrazek 2



(m) Efekt po operacji:  
*obrazek2 - obrazek1*



(n) Efekt po operacji:  
*obrazek1 - obrazek2*

W trakcie sprawdzania jaka metoda uzyskiwania zmian w obrazach daje najlepsze efekty przetestowałem również tą prostą metodę odjęcia jednego obrazka od drugiego. Mimo, że widać iż metoda wbudowana imabsdiff czerpie z odejmowania swoje działanie to w przypadku 'surowego odjęcia' efekt nie jest już taki zaskakujący - w przypadku np. pary drugiej(kadrów z filmu Furia - podpunkty: k,l,m,n) efekty są słabo czytelne a robiąc tak idealne klatki z filmu spodziewałem się lepszych wyników. W parze pierwszej(kadry z filmu Dunkierka) widzimy, że operacja ta pokazała tylko, ze zmieniło się położenie żołnierza, który stoi bokiem do kadru z prawej strony natomiast tego znajdującego się na drugim planie nie wzięła pod uwagę. Wrzucam tą operację raczej jako taki wynik podczas sprawdzania metod i szukania właściwej.. Nie podobały mi się efekty tej metody i raczej nie wziąłbym jej pod uwagę. Na marginesie mogę dodać, że próbowałem jeszcze sposobu żeby przekonwertować w pierwszej kolejności zdjęcie do postaci binarnej a dopiero potem wykonać operacje odejmowania ale efekt również nie był jakiś 'szalowy'.... A więc ostatecznie mogę śmiało powiedzieć, że osobiście skorzystałbym z wbudowanej metody imabsdiff.

## 4 Zadanie 3

### Treść zadania:

Wybierz sobie jeden obraz, następnie przeprowadź operację negacji, wyciągnij maskę R, G, B. Jakie są efekty? Co jeszcze możemy zrobić z jednym obrazem aby osiągnąć ciekawy efekt?

### Rozwiążanie:

```
// Zadanie3

// wczytaj obrazek
obrazek=imread(fullfile('C:/Users/Idzik/Desktop/Grafika maszynowa/SPR05/
23.11 [spr]/sprawozdanie/' + 'zad3.jpg'));

// oryginalnie
im=obrazek;

// subplot(2,2,1); imhist(im(:,:,1), 40, 'red'); xtitle("Kanal R");
// subplot(2,2,2); imhist(im(:,:,2), 40, 'green'); xtitle("Kanal G");
// subplot(2,2,3); imhist(im(:,:,3), 40, 'blue'); xtitle("Kanal B");
// subplot(2,2,4); imshow(im); xtitle("Zrodlo:");

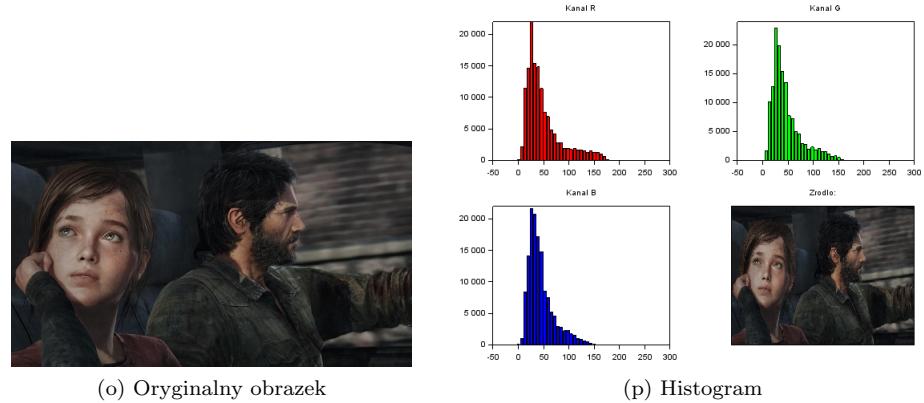
// negujmy
img=~obrazek;
// subplot(2,2,1); imhist(img(:,:,1), 40, 'red'); xtitle("Kanal R");
// subplot(2,2,2); imhist(img(:,:,2), 40, 'green'); xtitle("Kanal G");
// subplot(2,2,3); imhist(img(:,:,3), 40, 'blue'); xtitle("Kanal B");
// subplot(2,2,4); imshow(img); xtitle("Zrodlo:");

// wyciągamy maski R G B
// scf(1); imshow(img(:,:,1)); xtitle("Kanal R");
// scf(2); imshow(img(:,:,2)); xtitle("Kanal G");
// scf(3); imshow(img(:,:,3)); xtitle("Kanal B");

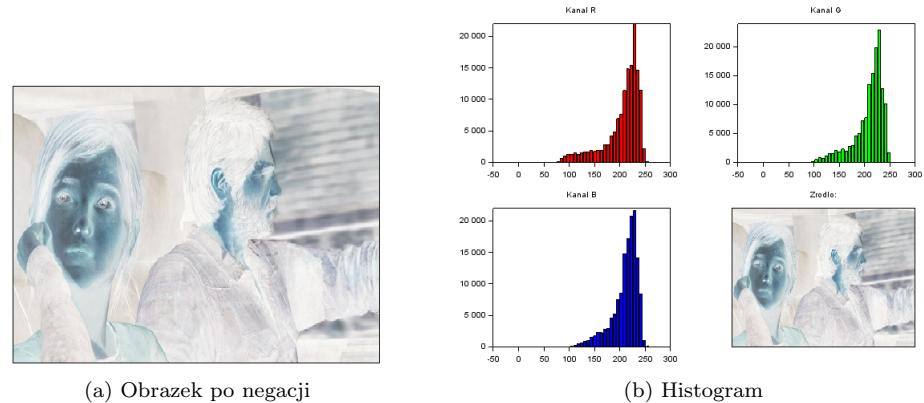
// Ciekawie gdy... *-1
img_test=obrazek*-1;
// subplot(2,2,1); imhist(img_test(:,:,1), 40, 'red'); xtitle("Kanal R");
// subplot(2,2,2); imhist(img_test(:,:,2), 40, 'green'); xtitle("Kanal G");
// subplot(2,2,3); imhist(img_test(:,:,3), 40, 'blue'); xtitle("Kanal B");
// subplot(2,2,4); imshow(img-test); xtitle("Zrodlo(przemnozenie przez -1):");

// cos tam za funkcja piszaca HI na ekranie
// i = dlugosc
// j = wysokosc
// for i=50:60
//     for j=50:100;
//         // rysuje H
//         im(j,i)=im(j,i)*-100;
//         im(j,i+50)=im(j,i+50)+5000;
//         im(i+25,j)=im(i+25,j)+5000;
// 
//         // rysuje I
//         im(j,i+100)=im(j,i+100)*-100;
//         imshow(im);
//     end
// end
```

## 4.1 Wnioski

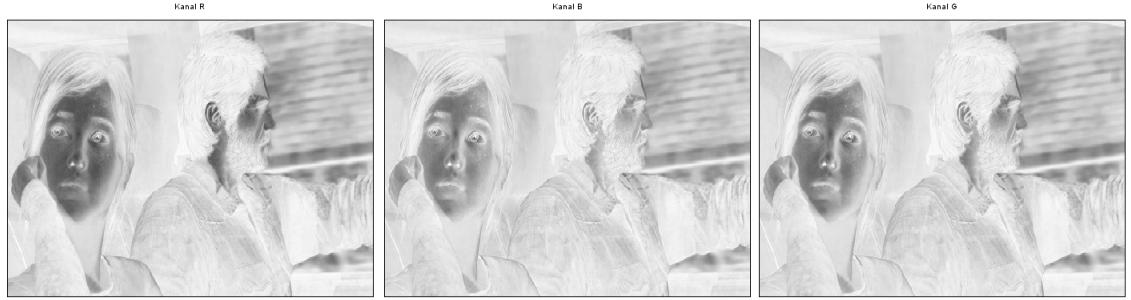


Rysunek 25: Informacje o oryginalnym obrazku



Rysunek 26: Informacje o obrazku po negacji

Jak przyjrzymy się histogramom obu obrazków to zauważymy, że rzeczywiście słupki każdego z kanałów odwróciły się (rysunek 28) po wykonaniu negacji.



Rysunek 27: Od lewej: kanał R, kanał B, kanał G

	Var - img	Var - img_test	1	2	3	4	5	6	7	8	9	10
1	233	233	233	232	232	233	233	234	234	234	234	234
2	233	232	232	232	232	232	232	232	232	234	233	233
3	233	233	233	232	232	232	232	232	232	233	233	233
4	236	235	235	234	234	234	234	234	234	233	233	233
5	237	237	236	236	235	235	235	235	235	233	234	234
6	237	237	237	237	236	236	236	236	236	236	236	236
7	238	238	238	238	238	238	238	237	237	238	238	238
8	240	240	240	240	240	240	240	239	239	238	238	238
9	239	240	241	241	241	241	241	241	242	241	241	241
10	242	242	242	242	242	242	242	242	242	242	243	242
11	243	242	241	242	243	243	243	242	242	243	242	242
12	240	240	240	241	242	242	242	242	242	243	242	242
13	240	240	241	242	242	242	242	242	242	243	244	243
14	243	243	244	244	243	243	243	243	244	244	244	244
15	245	245	245	245	245	245	244	244	244	244	244	244
16	245	244	244	244	245	245	245	245	244	245	245	245
17	245	245	244	244	244	245	245	245	245	245	245	246
18	243	243	243	242	242	243	243	244	244	245	245	245
19	241	241	241	241	240	240	240	241	242	243	242	242
20	239	240	240	239	238	238	238	239	241	240	238	238

(a) Macierz po negacji

	Var - img	Var - img_test	1	2	3	4	5	6	7	8	9	10
1	234	234	234	233	233	234	234	235	235	235	235	235
2	234	233	233	233	233	233	233	233	233	235	234	234
3	234	234	234	233	233	233	233	233	233	234	234	234
4	237	236	236	235	235	235	235	235	235	234	234	234
5	238	238	237	237	236	236	236	236	236	234	235	235
6	238	238	238	238	237	237	237	237	237	237	237	237
7	239	239	239	239	239	239	239	238	238	239	239	239
8	241	241	241	241	241	241	241	240	240	239	239	239
9	240	241	242	242	242	242	242	242	243	242	242	242
10	243	243	243	243	243	243	243	243	243	244	243	243
11	244	243	242	243	244	244	244	243	243	244	243	243
12	241	241	241	242	243	243	243	243	243	244	243	243
13	241	241	241	242	243	243	243	243	244	245	244	244
14	244	244	244	245	245	244	244	244	245	245	245	245
15	246	246	246	246	246	246	245	245	245	245	245	245
16	246	245	245	245	245	246	246	246	246	246	246	246
17	246	246	245	245	245	245	246	246	247	246	247	247
18	244	244	244	243	244	244	244	245	245	246	246	246
19	242	242	242	242	241	241	242	242	243	244	243	243
20	240	241	241	240	239	239	239	240	242	241	239	239

(b) Macierz po przemnożeniu przez wartość -1

Rysunek 28: Macierze obrazków

Przy negacji chciałem przetestować jak zachowa się przemnożenie macierzy obrazka przez wartość -1, którą testowałem we wcześniejszym sprawozdaniu i miałem wrażenie, że wyniki negacji jest taki sam. Rzeczywiście, otrzymałem wizualnie ten sam wynik jednak czy aby na pewno? Patrząc na macierze obu obrazków(i tego po negacji i tego po przemnożeniu) możemy jasno stwierdzić

spoglądając na np. wytyczone miejsca, że wartości różnią się o wartość 1.

Przykładowo jeżeli weźmiemy sobie na tapete jakieś konkretne punkty:

$$poNegacji(5, 3) = 236$$

$$poMnozeniu(5, 3) = 237$$

albo inne jeszcze porównanie

$$poNegacji(7, 4) = 238$$

$$poMnozeniu(7, 4) = 239$$

Traktuję to jako lekkie przekłamanie, gdyż wizualnie nie widzę różnic w efekcie zanegowania obrazku i przemnożenia go przez wartość -1, zresztą proszę spojrzeć poniżej na wycinki. Ja nie widzę różnic między nimi.



(a) Obrazek po negacji



(b) Obrazek po pomnożeniu przez wartość -1

Rysunek 29: Porównanie negacji i mnożenia

Jeżeli chodzi o kwestię tego czy są jeszcze jakieś rzeczy aby stworzyć ciekawe wizualne efekty na zdjęciach? W trakcie opracowywania sprawozdania natknęliśmy się już na operacje matematyczne i logiczne. Wydaje mi się, że jedynie funkcje z jakąś inwencją mogą zrobić coś 'szałowego' na obrazie. Zaprezentowana została na ćwiczeniach pętla, która powodowała wygenerowanie szumu na zdjęciu. Od siebie zawarłem w niniejszym sprawozdaniu(i kodzie) pętlę, która wykonuje napis HI na fotografii. Efekt widoczny jest na poniższym wycinku.



Rysunek 30: Możemy pisać po fotografii...