

Data oddania: _____

Ocena: _____

Jakub Antosik 206788

Andrzej Lisowski 206087

Zadanie 1: Szkielet aplikacji do przetwarzania i analizy obrazów, operacje podstawowe, usuwanie szumu, modyfikacje histogramu, filtracja liniowa i nieliniowa, splot.

1. Cel

Celem zadania było zapoznanie się z metodami analizy i przetwarzania obrazów. W części implementacyjnej należało stworzyć program w wybranym przez siebie języku programowania, który będzie w stanie przeprowadzić różne operacje na obrazie. Pełen spis funkcjonalności zostanie przedstawiony w sekcji *Wprowadzenie*.

2. Wprowadzenie

Obraz w pamięci komputera jest reprezentowany przez macierz pikseli. Sam piksel jest zaś najmniejszym elementem obrazu, mogącym przyjmować różne wartości liczb naturalnych:

- 0 - 7 dla obrazów 1-bitowych
 - 0 - 255 dla obrazów 8-bitowych (odcienie szarości)
 - 0 - 16777215 dla obrazów 24-bitowych (po 8 bitów na każdy kolor RGB)
- Poniżej przedstawione zostały teoretyczne podstawy transformacji, którym poddane zostały testowe dane.

2.1. Podstawowe operacje przetwarzania obrazu

Sekcja ta opisuje podstawowe operacje przetwarzania obrazów, często wykorzystywane w codziennym życiu.

2.1.1. Zmiana jasności

Zmiana jasności obrazu polega na dodaniu do wartości każdego piksela pewnej stałej liczby k . Jeżeli stała jest dodatnia, mówimy o zwiększaniu jasności, jeżeli jest ujemna - jasność jest zmniejszana. W momencie, w którym wynik przekroczy wartości brzegowe piksela, przypisuje się mu p_{\min} lub p_{\max} zgodnie z poniższym wzorem:

$$p(i) = \begin{cases} p_{\min} & \text{jeżeli } i + k < 0 \\ i + k & \text{jeżeli } p_{\min} \leq i + k \leq p_{\max} \\ p_{\max} & \text{jeżeli } i + k > p_{\max} \end{cases}$$

gdzie:

$p(i)$ - wartość piksela po zmianie jasności,

i - wartość piksela przed zmianą jasności,

p_{\min} - minimalna wartość piksela, $p_{\min} = 0$,

p_{\max} - maksymalna wartość piksela,

k - zmiana jasności.

2.1.2. Zmiana kontrastu

Zmiana kontrastu obrazu polega na zwiększeniu jasności jasnych pikseli przy jednoczesnym zmniejszeniu jasności ciemnych pikseli. Ciemne piksele należą do przedziału $\langle 0, \frac{p_{\max}}{2} \rangle$, zaś jasne do przedziału $\langle \frac{p_{\max}}{2}, p_{\max} \rangle$.

Wzór wygląda następująco:

$$p(i) = \begin{cases} i \div k & \text{jeżeli } 0 \leq i < \frac{p_{\max}}{2} \\ i * k & \text{jeżeli } \frac{p_{\max}}{2} \leq i \leq p_{\max} \\ p_{\max} & \text{jeżeli } i * k > p_{\max} \end{cases}$$

gdzie:

$p(i)$ - wartość piksela po zmianie kontrastu,

i - wartość piksela przed zmianą kontrastu,

p_{\max} - maksymalna wartość piksela,

k - współczynnik zmiany kontrastu.

2.1.3. Wyznaczenie negatywu

Negatyw jest przedstawieniem pikseli obrazu jako różnicy wartości maksymalnej i obecnej:

$$p(i) = p_{\max} - i$$

gdzie:

$p(i)$ - wartość piksela po wyznaczeniu negatywu,

i - wartość piksela przed zmianą kontrastu,

p_{max} - maksymalna wartość piksela.

2.2. Podstawowe filtry

Poniższa sekcja opisuje 2 podstawowe filtry, które podległy analizie - filtr ze średnią arytmetyczną oraz filtr medianowy.

2.2.1. Filtr ze średnią arytmetyczną

Filtr ze średnią arytmetyczną jest wykorzystywany w operacjach odszumiania obrazu. Algorytm polega na przypisaniu do nowej wartości pikseli średniej arytmetycznej badanego elementu oraz jego sąsiedztwa s . Sąsiedztwo może przyjmować wartości potęg kolejnych liczb nieparzystych większych od 1:

$$s \in \{9, 25, 49, \dots\}$$
$$p(i) = \frac{\sum_{x=-n}^n \sum_{y=-n}^n i_{x,y}}{s}$$

gdzie:

$p(i)$ - wartość piksela po nałożeniu filtru,

i - wartość piksela przed nałożeniem filtru,

s - maska filtru,

n - rozpiętość maski filtru, obliczana ze wzoru:

$$n = \frac{\sqrt{s} - 1}{2}$$

x - współrzędna x piksela na obrazie,

y - współrzędna y piksela na obrazie.

Należy pamiętać, że piksele poddawane filtracji nie mogą być elementami brzegowymi obrazu, więc:

$$i_x + n \leq x_{max} \wedge i_x - n \geq x_{min} \wedge i_y + n \leq y_{max} \wedge i_y - n \geq y_{min}$$

gdzie:

i - wartość piksela przed zmianą kontrastu,

x - współrzędna x piksela na obrazie,

y - współrzędna y piksela na obrazie,

x_{max} - maksymalna wartość współrzędnej x na obrazie,

x_{min} - minimalna wartość współrzędnej x na obrazie, $x_{min} = 0$,

y_{max} - maksymalna wartość współrzędnej y na obrazie,

y_{min} - minimalna wartość współrzędnej y na obrazie, $y_{min} = 0$.

2.2.2. Filtr medianowy

Filtr medianowy jest bardzo podobny do filtru ze średnią arytmetyczną. W tym przypadku jednak, nowa wartość piksela jest medianą badanego elementu oraz jego sąsiedztwa s . Sąsiedztwo może przyjmować wartości potęg kolejnych liczb nieparzystych większych od 1:

$$s \in \{9, 25, 49, \dots\}$$
$$p(i) = M_s$$

gdzie:

$p(i)$ - wartość piksela po nałożeniu filtru,

M - mediana,

s - maska filtru.

Należy pamiętać, że piksele poddawane filtracji nie mogą być elementami brzegowymi obrazu, więc:

$$i_x + n \leq x_{max} \wedge i_x - n \geq x_{min} \wedge i_y + n \leq y_{max} \wedge i_y - n \geq y_{min}$$

gdzie:

i - wartość piksela przed zmianą kontrastu,

n - rozpiętość maski filtru, obliczana ze wzoru:

$$n = \frac{\sqrt{s} - 1}{2}$$

x - współrzędna x piksela na obrazie,

y - współrzędna y piksela na obrazie,

x_{max} - maksymalna wartość współrzędnej x na obrazie,

x_{min} - minimalna wartość współrzędnej x na obrazie, $x_{min} = 0$,

y_{max} - maksymalna wartość współrzędnej y na obrazie,

y_{min} - minimalna wartość współrzędnej y na obrazie, $y_{min} = 0$.

2.3. Modyfikacje obrazu w oparciu o histogram

Histogram umożliwia przedstawienie rozkładu pikseli o określonych wartościach na wykresie. Poniższa sekcja prezentuje modyfikacje obrazu na podstawie jego histogramu.

2.3.1. Jednostajna wyjściowa gęstość prawdopodobieństwa

Obliczana jest ze wzoru:

$$g(f) = g_{min} + (g_{max} - g_{min}) \frac{1}{N} \sum_{m=0}^f H(m)$$

gdzie:

f - wartość rozważanego kanału przed modyfikacją,

g - wartość rozważanego kanału po modyfikacji,

g_{min} - pożądana minimalna wartość przetwarzanego kanału,

g_{max} - pożądana maksymalna wartość przetwarzanego kanału,
 N - suma pikseli na obrazie,
 $H(m)$ - wartość histogramu dla wartości m kanału.

2.3.2. Wyjściowa gęstość prawdopodobieństwa o postaci wykładniczej

Obliczana jest ze wzoru:

$$g(f) = g_{min} - \frac{1}{\alpha} \ln(1 - \frac{1}{N} \sum_{m=0}^f H(m))$$

gdzie:

f - wartość rozważanego kanału przed modyfikacją,
 g - wartość rozważanego kanału po modyfikacji,
 g_{min} - pożądana minimalna wartość przetwarzanego kanału,
 α - współczynnik obliczany ze wzoru: TODO,
 N - suma pikseli na obrazie,
 $H(m)$ - wartość histogramu dla wartości m kanału.

2.3.3. Wyjściowa gęstość prawdopodobieństwa podana wzorem Raleigha

Obliczana jest ze wzoru:

$$g(f) = g_{min} + (2\alpha^2 \ln(\frac{1}{N} \sum_{m=0}^f H(m)))^{-1/2}$$

gdzie:

f - wartość rozważanego kanału przed modyfikacją,
 g - wartość rozważanego kanału po modyfikacji,
 g_{min} - pożądana minimalna wartość przetwarzanego kanału,
 α - współczynnik obliczany ze wzoru: TODO,
 N - suma pikseli na obrazie,
 $H(m)$ - wartość histogramu dla wartości m kanału.

2.3.4. Wyjściowa gęstość prawdopodobieństwa określona przez potęgę 2/3

Obliczana jest ze wzoru:

$$g(f) = (g_{min}^{\frac{1}{3}} + (g_{max}^{\frac{1}{3}} - g_{min}^{\frac{1}{3}}) \frac{1}{N} \sum_{m=0}^f H(m))^3$$

gdzie:

f - wartość rozważanego kanału przed modyfikacją,
 g - wartość rozważanego kanału po modyfikacji,
 g_{min} - pożądana minimalna wartość przetwarzanego kanału,

g_{max} - pożądana maksymalna wartość przetwarzanego kanału,
 N - suma pikseli na obrazie,
 $H(m)$ - wartość histogramu dla wartości m kanału.

2.3.5. Wyjściowa gęstość prawdopodobieństwa o postaci hiperbolicznej

Obliczana jest ze wzoru:

$$g(f) = g_{min} \left(\frac{g_{max}}{g_{min}} \right)^{\frac{1}{N} \sum_{m=0}^f H(m)}$$

gdzie:

f - wartość rozważanego kanału przed modyfikacją,
 g - wartość rozważanego kanału po modyfikacji,
 g_{min} - pożądana minimalna wartość przetwarzanego kanału,
 g_{max} - pożądana maksymalna wartość przetwarzanego kanału,
 N - suma pikseli na obrazie,
 $H(m)$ - wartość histogramu dla wartości m kanału.

2.4. Filtracja liniowa oparta o splot

Poniższa sekcja przedstawia zbadane przekształcenia filtracji liniowej. Polega ona na modyfikacji pikseli obrazu poprzez nałożenie na nie maski filtrującej. Wzór przedstawia się następująco:

$$g(p, q) = \sum_{i=-M}^M \sum_{j=-M}^M h(i, j) x(p+i, q+j), p = M, 2, \dots, P-M-1, q = M, 2, \dots, Q-M-1$$

gdzie:

$x(p, q)$ - wartość rozważanego kanału przed modyfikacją,
 $g(p, q)$ - wartość rozważanego kanału po modyfikacji,
 $h(i, j)$ - wartość maski filtru.

2.4.1. Filtr dolnoprzepustowy

Filtry dolnoprzepustowe są wykorzystywane do usuwania elementów o wysokiej częstotliwości. Dążą one to uśrednienia wartości sąsiadujących pikselów. Wykorzystane maski:

$$\frac{1}{9} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad \frac{1}{10} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad \frac{1}{16} \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$$

2.4.2. Wyostrażanie krawędzi

Poniższe maski zostały użyte do wyostrażenia krawędzi:

$$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix} \quad \begin{pmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{pmatrix} \quad \begin{pmatrix} 1 & -2 & 1 \\ -2 & 5 & -2 \\ 1 & -2 & 1 \end{pmatrix}$$

2.4.3. Wydobywanie szczegółów z tła: N, NE, E, SE

Wydobycie szczegółów z tła w kierunkach: północnym, północno-wschodnim, wschodnim i południowo-wschodnim zostało przebadane następującymi maskami:

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{pmatrix} \quad \begin{pmatrix} 1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & -1 & 1 \end{pmatrix} \quad \begin{pmatrix} -1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & 1 & 1 \end{pmatrix} \quad \begin{pmatrix} -1 & -1 & 1 \\ -1 & -2 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

2.4.4. Wydobywanie szczegółów z tła: S, SW, W, NW

Do wydobywania szczegółów z tła w kierunkach: południowym, południowo-zachodnim, zachodnim i północno-zachodnim zostało wykorzystane następujące maski:

$$\begin{pmatrix} -1 & -1 & -1 \\ 1 & -2 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad \begin{pmatrix} 1 & -1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & 1 \end{pmatrix} \quad \begin{pmatrix} 1 & 1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & -1 \end{pmatrix} \quad \begin{pmatrix} 1 & 1 & 1 \\ 1 & -2 & -1 \\ 1 & -1 & -1 \end{pmatrix}$$

2.4.5. Wydobywanie szczegółów z tła bez zdefiniowanego kierunku (laplasjan)

Wydobycie szczegółów z tła bez zdefiniowanego kierunku jest możliwe przy nałożeniu następujących masek:

$$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix} \quad \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix} \quad \begin{pmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{pmatrix}$$

2.4.6. Identyfikowanie linii

Identyfikowanie linii na obrazie zostało zrealizowane po nałożeniu masek:

$$\begin{pmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{pmatrix} \quad \begin{pmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{pmatrix} \quad \begin{pmatrix} -1 & -1 & 2 \\ -1 & 2 & -1 \\ 2 & -1 & -1 \end{pmatrix} \quad \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix}$$

2.5. Filtracja nieliniowa

Poniższa sekcja przedstawia zbadane przekształcenia filtracji nieliniowej.

2.5.1. Operator Robertsa (Wariant I)

Wzór operatora jest następujący:

$$g(x, y) = ((f(x, y) - f(x + 1, y + 1))^2 + (f(x, y + 1) - f(x + 1, y))^2)^{\frac{1}{2}}$$

gdzie:

$g(x, y)$ - wartość rozważanego piksela po modyfikacji,

$f(x, y)$ - wartość rozważanego piksela przed modyfikacją.

2.5.2. Operator Robertsa (Wariant II)

Wzór operatora jest następujący:

$$g(x, y) = |f(x, y) - f(x + 1, y + 1)| + |f(x, y + 1) - f(x + 1, y)|$$

gdzie:

$g(x, y)$ - wartość rozważanego piksela po modyfikacji,

$f(x, y)$ - wartość rozważanego piksela przed modyfikacją.

2.5.3. Operator Sobela

Wzór operatora jest następujący:

$$g(x, y) = \sqrt{S_x^2 + S_y^2}$$

$$S_x = (a_2 + 2a_3 + a_4) - (a_0 + 2a_7 + a_6)$$

$$S_y = (a_0 + 2a_1 + a_2) - (a_6 + 2a_5 + a_4)$$

gdzie:

$g(x, y)$ - wartość rozważanego piksela po modyfikacji,

$a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7$, - wzory na współczynniki podane pod koniec sekcji.

2.5.4. Operator Kirsha

Wzór operatora jest następujący:

$$g(x, y) = \max(1, \max_{i=0, \dots, 7} |5K_{1,i} - 3K_{2,i}|)$$

$$K_{1,i} = a_1 + a_{i+1} + a_{i+2}$$

$$K_{2,i} = a_{i+3} + a_{i+4} + a_{i+5} + a_{i+6} + a_{i+7}$$

gdzie:

$g(x, y)$ - wartość rozważanego piksela po modyfikacji,

a_i - wzory na współczynniki podane pod koniec sekcji,
operacje dodawania indeksów są operacjami modulo 8.

2.5.5. Operator Rosenfelda

Wzór operatora jest następujący:

$$g(x, y) = \frac{1}{R} \left(\sum_{i=1}^R f(x+i-1, y) - \sum_{i=1}^R f(x-i, y) \right)$$

gdzie:

$g(x, y)$ - wartość rozważanego piksela po modyfikacji,

$f(x, y)$ - wartość rozważanego piksela przed modyfikacją,

R - współczynnik Rosenfelda, podawany na wejściu przez użytkownika.

2.5.6. Operator Uolisa

Wzór operatora jest następujący:

$$g(x, y) = \frac{1}{4} \log\left(\frac{f(x, y)^4}{a_1 a_3 a_5 a_7}\right)$$

gdzie:

$g(x, y)$ - wartość rozważanego piksela po modyfikacji,

$f(x, y)$ - wartość rozważanego piksela przed modyfikacją,

a_1, a_3, a_5, a_7 , - wzory na współczynniki podane pod koniec sekcji.

2.5.7. Wzory na współczynniki a

$$a_0 = f(x - 1, y - 1)$$

$$a_1 = f(x, y - 1)$$

$$a_2 = f(x + 1, y - 1)$$

$$a_3 = f(x + 1, y)$$

$$a_4 = f(x + 1, y + 1)$$

$$a_5 = f(x, y + 1)$$

$$a_6 = f(x - 1, y + 1)$$

$$a_7 = f(x - 1, y)$$

gdzie:

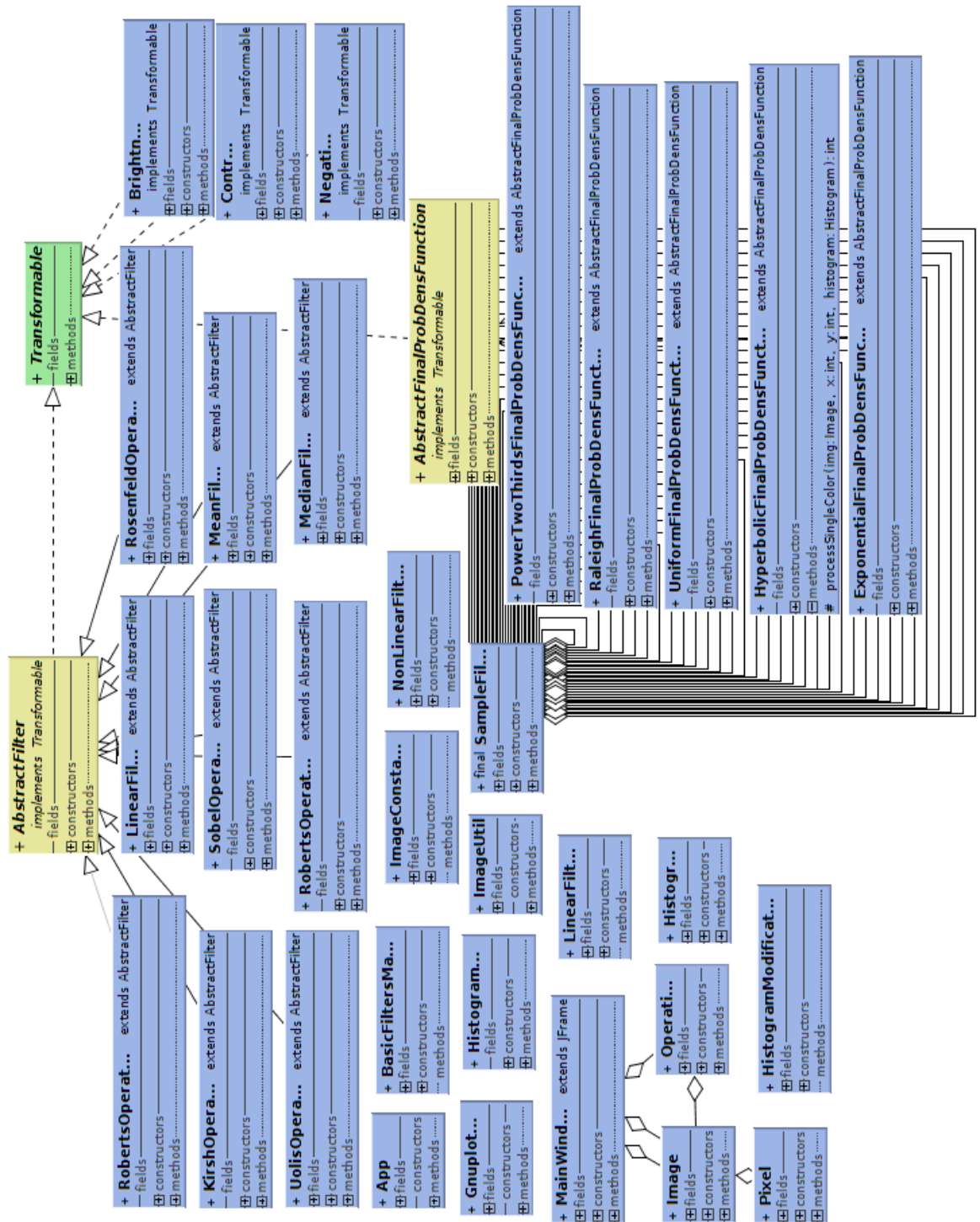
$f(x, y)$ - wartość rozważanego piksela przed modyfikacją.

3. Opis implementacji

Aplikacja została napisana w języku programowania Java. Wybór środowiska był podyktowany dużą ilością bibliotek mogących pomóc w realizacji zadania oraz osobistymi preferencjami członków zespołu. Warstwa GUI została realizowana przy użyciu standardowej biblioteki graficznej Javy - Swing. Aplikacja wykorzystuje następujące biblioteki zewnętrzne:

- *JavaPlot.jar* - wykorzystywana do tworzenia histogramów; jest to jadowa implementacja gnuplota
- *log4j-1.2.17.jar* - wykorzystywana do obsługi logowania
- *slf4j-api-1.7.12.jar* - wykorzystywana do obsługi logowania
- *slf4j-log4j12-1.7.12.jar* - wykorzystywana do obsługi logowania

Poniżej przedstawiony został diagram UML klas znajdujących się w programie. Aby zachować czytelność, nie zamieszczono połączeń pomiędzy wywołaniami obiektów, a także ukryto informacje na temat metod i parametrów.



4. Materiały i metody

Testowy zbiór przekształczanych obrazów można podzielić na 3 główne grupy:

- obrazy 1-bitowe
- obrazy 8-bitowe (w skali szarości)
- obrazy 24-bitowe (RGB)



(a) Boat



(b) Girl

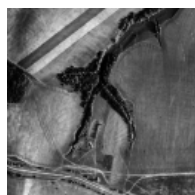


(c) Lena

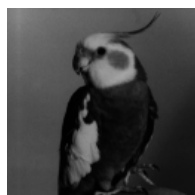


(d) Mandril

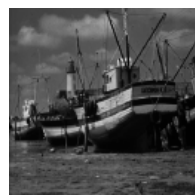
Rysunek 1: Testowe obrazy 1-bitowe



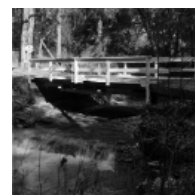
(a) Aero



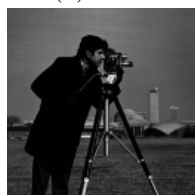
(b) Bird



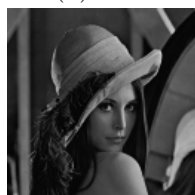
(c) Boat



(d) Bridge



(e) Camera



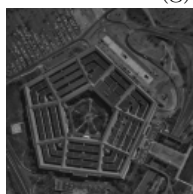
(f) Lena



(g) Mandril



(h) Messer



(i) Pentagon

Rysunek 2: Testowe obrazy 8-bitowe



Rysunek 3: Testowe obrazy Lena 8-bitowe zaszumione



Rysunek 4: Testowe obrazy 24-bitowe



Rysunek 5: Testowe obrazy Lena 24-bitowe zaszumione

Badania zostały zrealizowane przy pomocy stworzonej aplikacji. Użytkownik definiował wejściowy obraz, który potem był poddawany wybranemu przez niego przekształceniu. Wyniki były zapisywane na dysku. Szczegółowa konfiguracja współczynników podczas transformacji zostanie podany w sekcji *Wyniki*.

5. Wyniki

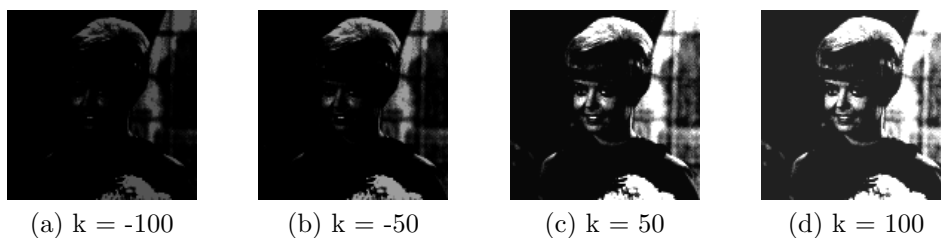
Poniżej przedstawione zostały efekty przeprowadzonych badań. Przeanalizowano wybrane obrazy 1-, 8- i 24-bitowe.

5.1. Podstawowe operacje przetwarzania obrazu

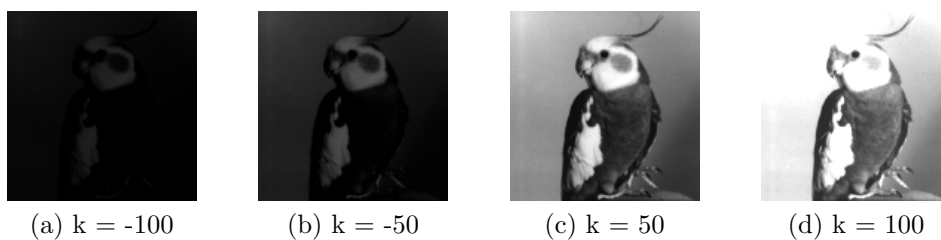
Sekcja przedstawia wyniki podstawowego przetwarzania obrazów - zmiany jasności, kontrastu oraz wyznaczenia negatywu.

5.1.1. Zmiana jasności

Poniżej przedstawione zostały wyniki zmiany jasności dla 3 wybranych obrazów.



Rysunek 6: Zmiana jasności w obrazie Girl 1-bitowym



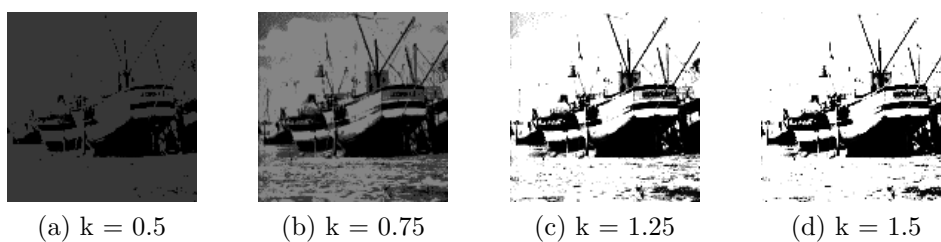
Rysunek 7: Zmiana jasności w obrazie Bird 8-bitowym



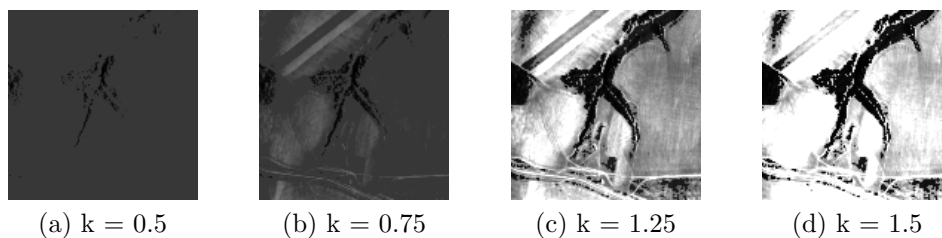
Rysunek 8: Zmiana jasności w obrazie Lena 24-bitowym

5.1.2. Zmiana kontrastu

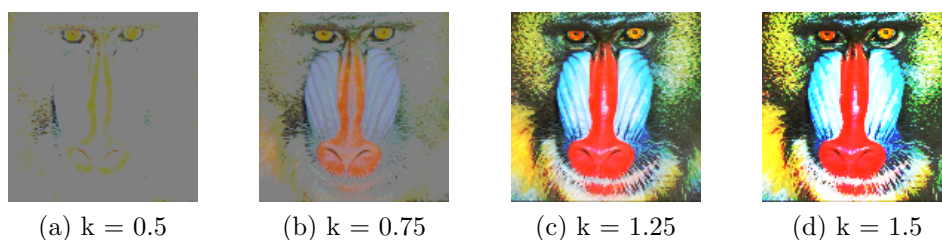
Poniżej przedstawione zostały wyniki zmiany kontrastu dla 3 wybranych obrazów.



Rysunek 9: Zmiana kontrastu w obrazie Boat 1-bitowym



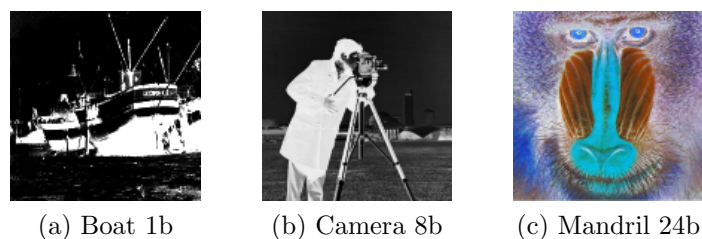
Rysunek 10: Zmiana kontrastu w obrazie Aero 8-bitowym



Rysunek 11: Zmiana kontrastu w obrazie Mandril 24-bitowym

5.1.3. Wyznaczenie negatywu

Poniżej przedstawione zostały wyniki wyznaczenia negatywu dla 3 wybranych obrazów.



Rysunek 12: Negatyw wybranych obrazów

5.2. Podstawowe filtry

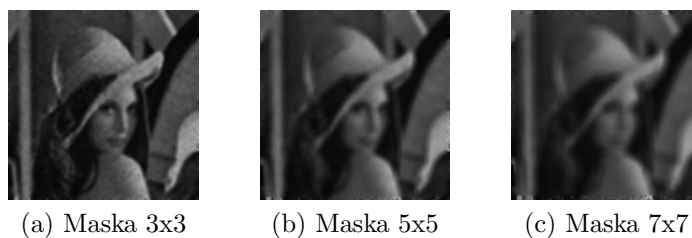
Sekcja przedstawia wyniki nałożenia na obrazy podstawowych filtrów - ze średnią arytmetyczną i medianowego.

5.2.1. Filtr ze średnią arytmetyczną

Poniżej przedstawione zostały wyniki nałożenia filtru ze średnią arytmetyczną na obraz Lena 8- i 24-bitowy dla szumów:

- Impulsowy 3
- Jednolity 3
- Normalny 3

W sprawozdaniu przedstawiono wyniki dla trzech masek: 3x3, 5x5 i 7x7.



Rysunek 13: Wyniki nałożenia filtru ze średnią arytmetyczną na obraz Lena 8-bitowy, szum Impulsowy 3



Rysunek 14: Wyniki nałożenia filtru ze średnią arytmetyczną na obraz Lena 8-bitowy, szum Jednolity 3



Rysunek 15: Wyniki nałożenia filtru ze średnią arytmetyczną na obraz Lena 8-bitowy, szum Normalny 3



Rysunek 16: Wyniki nałożenia filtru ze średnią arytmetyczną na obraz Lena 24-bitowy, szum Impulsowy 3



Rysunek 17: Wyniki nałożenia filtru ze średnią arytmetyczną na obraz Lena 24-bitowy, szum Jednolity 3



Rysunek 18: Wyniki nałożenia filtru ze średnią arytmetyczną na obraz Lena 24-bitowy, szum Normalny 3

5.2.2. Filtr medianowy

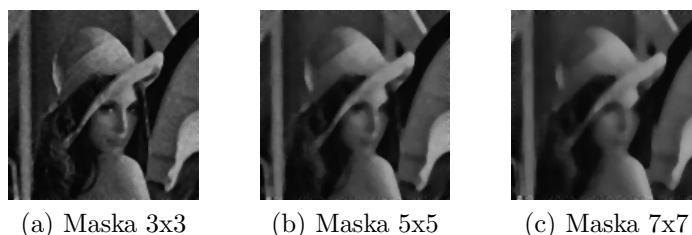
Poniżej przedstawione zostały wyniki nałożenia filtru medianowego na obraz Lena 8- i 24-bitowy dla szumów:

- Impulsowy 3
- Jednolity 3
- Normalny 3

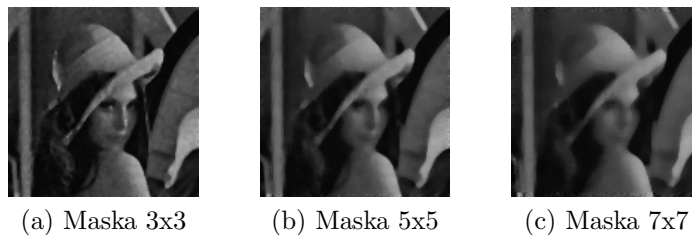
W sprawozdaniu przedstawiono wyniki dla trzech masek: 3x3, 5x5 i 7x7.



Rysunek 19: Wyniki nałożenia filtru medianowego na obraz Lena 8-bitowy, szum Impulsowy 3



Rysunek 20: Wyniki nałożenia filtru ze średnią arytmetyczną na obraz Lena 8-bitowy, szum Jednolity 3



Rysunek 21: Wyniki nałożenia filtru medianowego na obraz Lena 8-bitowy, szum Normalny 3



Rysunek 22: Wyniki nałożenia filtru medianowego na obraz Lena 24-bitowy, szum Impulsowy 3



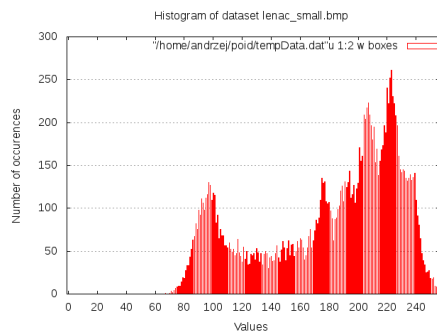
Rysunek 23: Wyniki nałożenia filtru medianowego na obraz Lena 24-bitowy, szum Jednolity 3



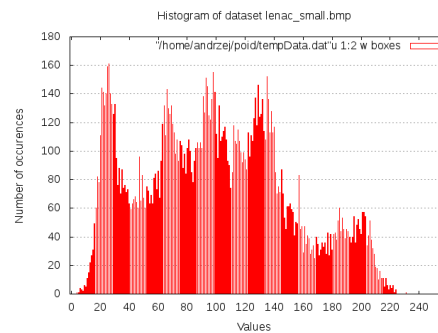
Rysunek 24: Wyniki nałożenia filtru medianowego na obraz Lena 24-bitowy, szum Normalny 3

5.3. Modyfikacje obrazu w oparciu o histogram

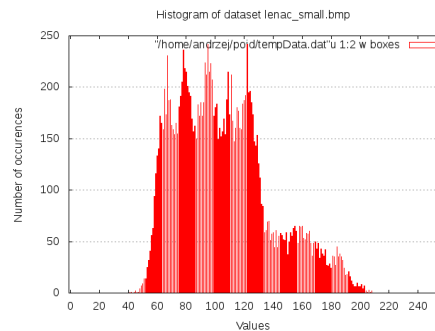
Sekcja przedstawia wyniki modyfikacji obrazów w oparciu o histogram. W celu porównania wyników, wszystkie operacje przeprowadzane były na obrazie 24-bitowym obrazie Lena. Poniżej przedstawiono histogramy kolorów obrazu oryginalnego.



(a) CZERWONY



(b) ZIELONY



(c) NIEBIESKI

Rysunek 25: Histogramy kolorów dla 24-bitowego obrazu Lena

5.3.1. Jednostajna wyjściowa gęstość prawdopodobieństwa

//TODO

5.3.2. Wyjściowa gęstość prawdopodobieństwa o postaci wykładniczej

//TODO

5.3.3. Wyjściowa gęstość prawdopodobieństwa podana wzorem Raleigha

//TODO

5.3.4. Wyjściowa gęstość prawdopodobieństwa określona przez potęgę $2/3$

//TODO

5.3.5. Wyjściowa gęstość prawdopodobieństwa o postaci hiperbolicznej

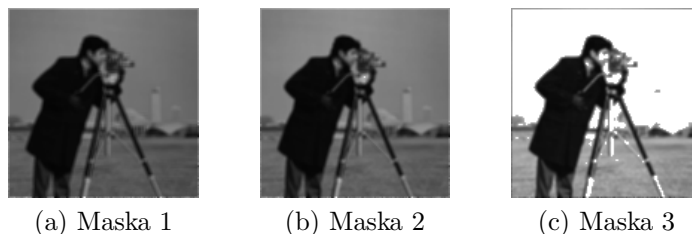
//TODO

5.4. Filtracja liniowa oparta o splot

Sekcja przedstawia wyniki nałożenia na obrazy filtrów liniowych w oparciu o splot.

5.4.1. Filtr dolnoprzepustowy

Poniżej przedstawione zostały wyniki nałożenia filtra dolnoprzepustowego o różnych maskach na wybrane obrazy 8- i 24-bitowe. Wygląd masek znajduje się w sekcji *Wprowadzenie*.



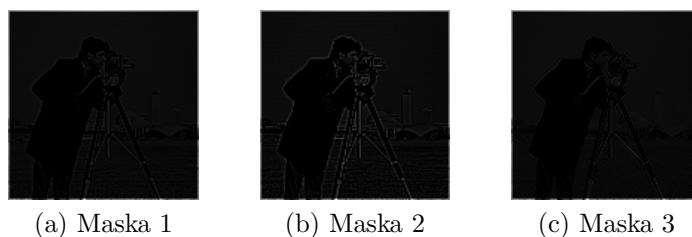
Rysunek 26: Wyniki nałożenia filtra dolnoprzepustowego o różnych maskach na obraz Camera 8-bitowy



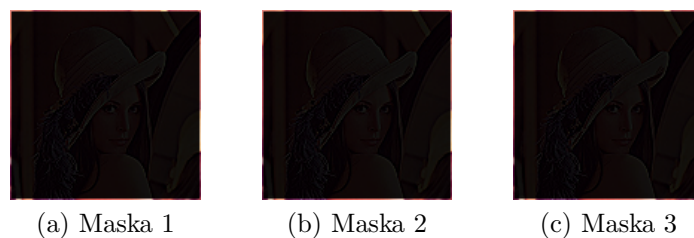
Rysunek 27: Wyniki nałożenia filtra dolnoprzepustowego o różnych maskach na obraz Lena 24-bitowy

5.4.2. Wyostrażanie krawędzi

Poniżej przedstawione zostały wyniki wyostrażania krawędzi stosując różne maski na wybranych obrazach 8- i 24-bitowych. Wygląd masek znajduje się w sekcji *Wprowadzenie*.



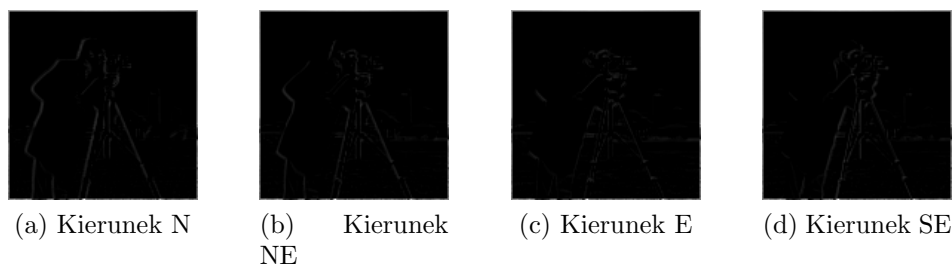
Rysunek 28: Wyniki wyostrażania krawędzi stosując różne maski na obrazie Camera 8-bitowym



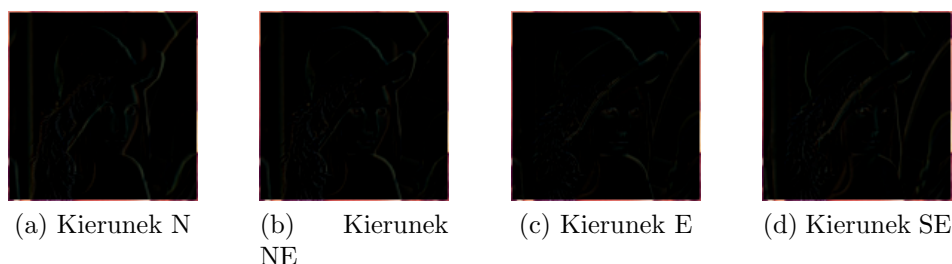
Rysunek 29: Wyniki wyostrzania krawędzi stosując różne maski na obrazie Lena 24-bitowym

5.4.3. Wydobywanie szczegółów z tła: N, NE, E, SE

Poniżej przedstawione zostały wyniki wydobywania szczegółów z tła w kierunkach N, NE, E i SE na wybranych obrazach 8- i 24-bitowych. Wygląd masek znajduje się w sekcji *Wprowadzenie*.



Rysunek 30: Wyniki wydobywania szczegółów z tła w kierunkach N, NE, E i SE na obrazie Camera 8-bitowym



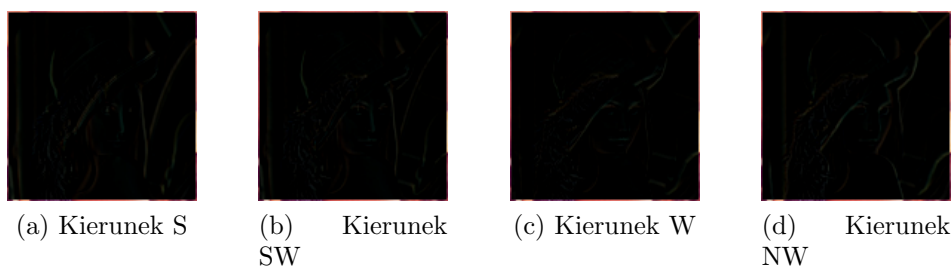
Rysunek 31: Wyniki wydobywania szczegółów z tła w kierunkach N, NE, E i SE na obrazie Lena 24-bitowym

5.4.4. Wydobywanie szczegółów z tła: S, SW, W, NW

Poniżej przedstawione zostały wyniki wydobywania szczegółów z tła w kierunkach S, SW, W, NW na wybranych obrazach 8- i 24-bitowych. Wygląd masek znajduje się w sekcji *Wprowadzenie*.



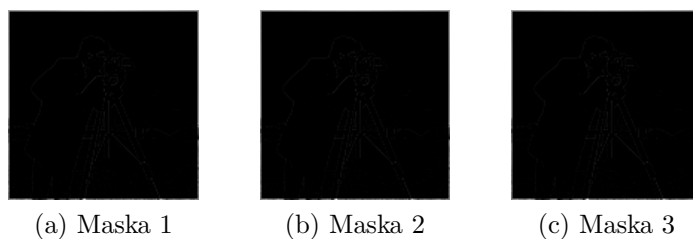
Rysunek 32: Wyniki wydobywania szczegółów z tła w kierunkach S, SW, W i NW na obrazie Camera 8-bitowym



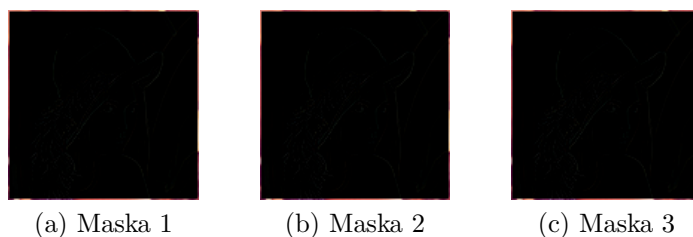
Rysunek 33: Wyniki wydobywania szczegółów z tła w kierunkach S, SW, W i NW na obrazie Lena 24-bitowym

5.4.5. Wydobywanie szczegółów z tła bez zdefiniowanego kierunku (laplasjan)

Poniżej przedstawione zostały wyniki wydobywania szczegółów z tła bez zdefiniowanego kierunku na wybranych obrazach 8- i 24-bitowych. Wygląd masek znajduje się w sekcji *Wprowadzenie*.



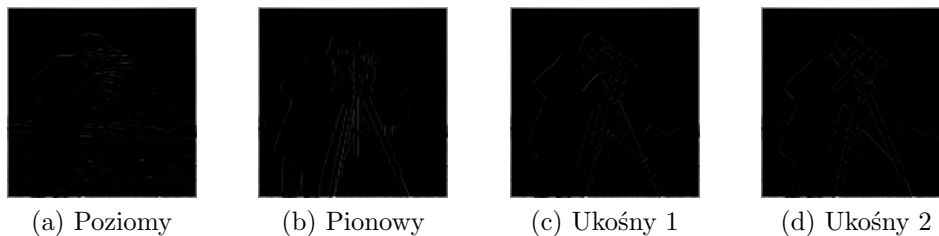
Rysunek 34: Wyniki wydobywania szczegółów z tła bez zdefiniowanego kierunku na obrazie Camera 8-bitowym



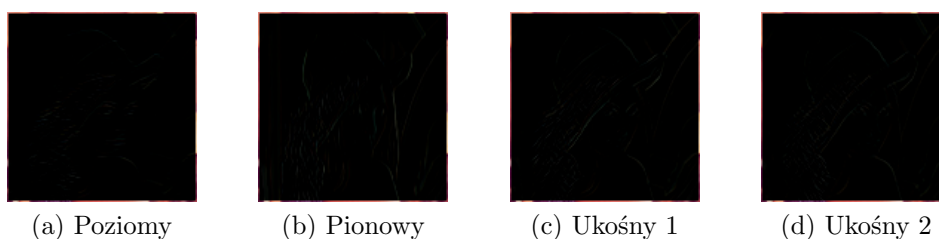
Rysunek 35: Wyniki wydobywania szczegółów z tła bez zdefiniowanego kierunku na obrazie Lena 24-bitowym

5.4.6. Identyfikowanie linii

Poniżej przedstawione zostały wyniki identyfikacji linii z wybranych obrazów 8- i 24-bitowych. Wygląd masek znajduje się w sekcji *Wprowadzenie*.



Rysunek 36: Wyniki identyfikacji linii na obrazie Camera 8-bitowym



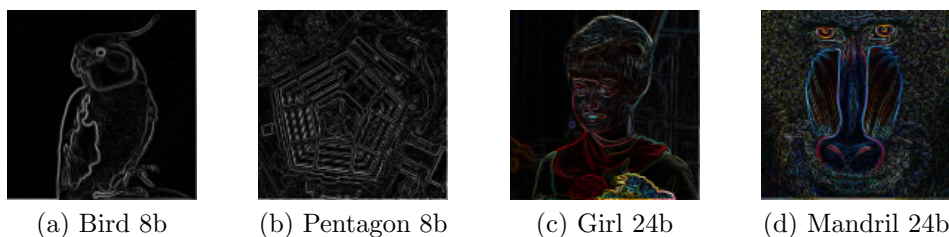
Rysunek 37: Wyniki identyfikacji linii na obrazie Lena 24-bitowym

5.5. Filtracja nieliniowa

Sekcja przedstawia wyniki nałożenia na obrazy filtrów nieliniowych.

5.5.1. Operator Robertsa (Wariant I)

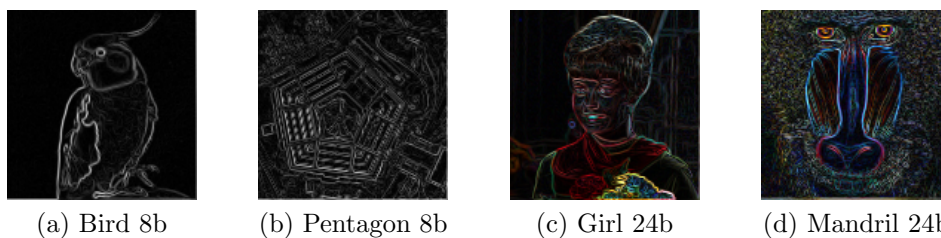
Poniżej przedstawione zostały wyniki zastosowania operatora Robertsa w I wariacie na wybranych obrazach 8- i 24-bitowych.



Rysunek 38: Wyniki zastosowania operatora Robertsa w I wariacie na wybrane obrazy 8- i 24-bitowe

5.5.2. Operator Robertsa (Wariant II)

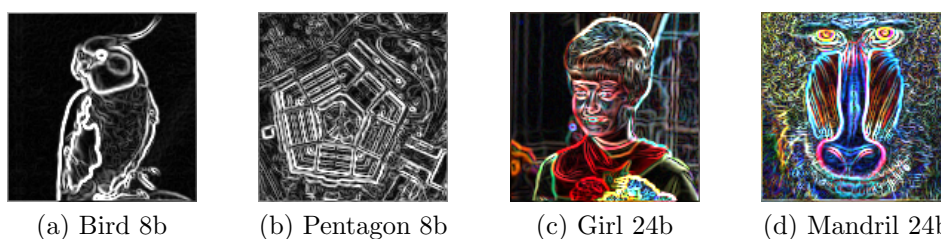
Poniżej przedstawione zostały wyniki zastosowania operatora Robertsa w II wariacie na wybranych obrazach 8- i 24-bitowych.



Rysunek 39: Wyniki zastosowania operatora Robertsa w I wariancie na wybrane obrazy 8- i 24-bitowe

5.5.3. Operator Sobela

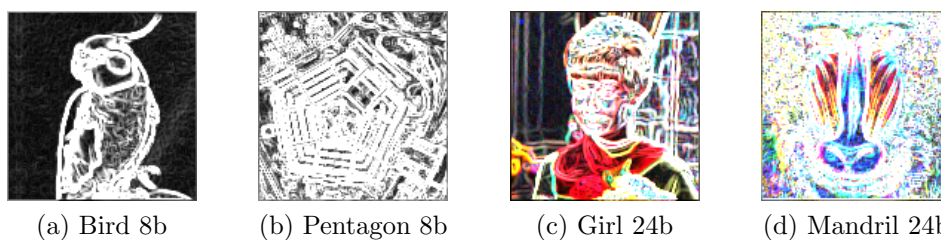
Poniżej przedstawione zostały wyniki zastosowania operatora Sobela na wybranych obrazach 8- i 24-bitowych.



Rysunek 40: Wyniki zastosowania operatora Robertsa w I wariancie na wybrane obrazy 8- i 24-bitowe

5.5.4. Operator Kirsha

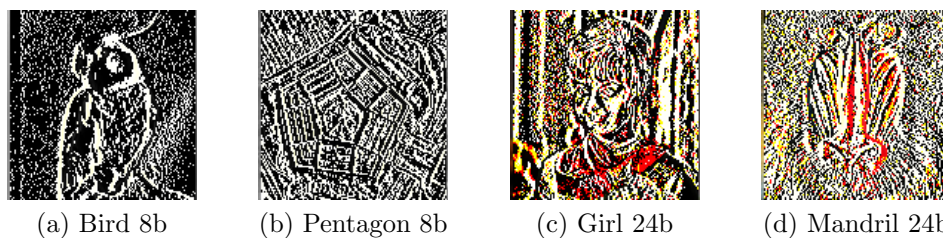
Poniżej przedstawione zostały wyniki zastosowania operatora Sobela na wybranych obrazach 8- i 24-bitowych.



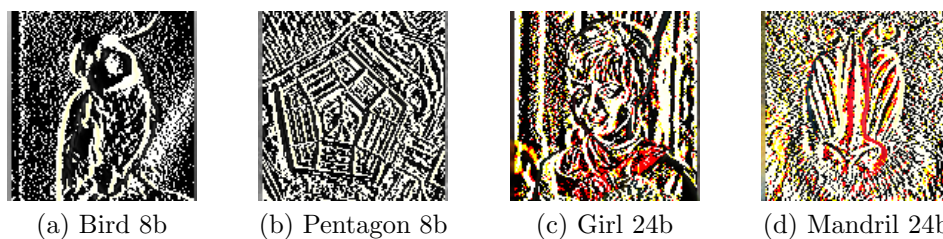
Rysunek 41: Wyniki zastosowania operatora Kirsha na wybrane obrazy 8- i 24-bitowe

5.5.5. Operator Rosenfelda

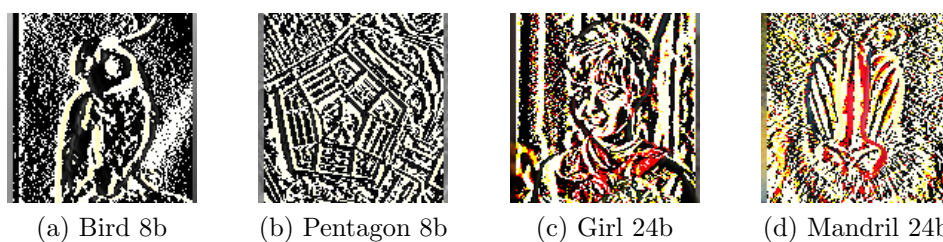
Poniżej przedstawione zostały wyniki zastosowania operatora Rosenfelda na wybranych obrazach 8- i 24-bitowych, w zależności od różnych parametrów R.



Rysunek 42: Wyniki zastosowania operatora Rosenfelda na wybranych obrazach 8- i 24-bitowych, $R = 2$



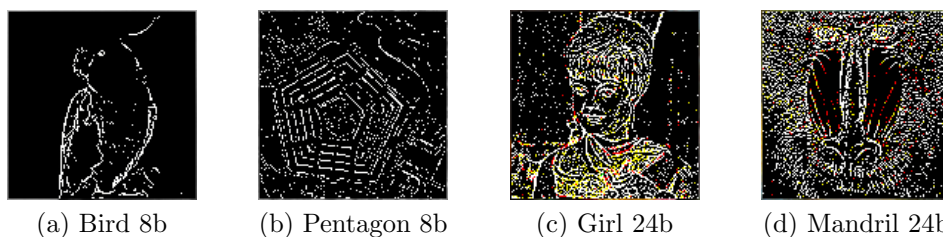
Rysunek 43: Wyniki zastosowania operatora Rosenfelda na wybranych obrazach 8- i 24-bitowych, $R = 3$



Rysunek 44: Wyniki zastosowania operatora Rosenfelda na wybranych obrazach 8- i 24-bitowych, $R = 4$

5.5.6. Operator Uolisa

Poniżej przedstawione zostały wyniki zastosowania operatora Uolisa na wybranych obrazach 8- i 24-bitowych.



Rysunek 45: Wyniki zastosowania operatora Uolisa na wybranych obrazach 8- i 24-bitowych

6. Dyskusja

Poniższa sekcja prezentuje interpretację uzyskanych wyników oraz wnioski. Opisano również napotkane problemy oraz możliwe sposoby ich rozwiązania.

6.1. Podstawowe operacje przetwarzania obrazu

Przetworzenie obrazów podstawowymi operacjami przyniosło spodziewane rezultaty.

6.2. Podstawowe filtry

Filtr ze średnią arytmetyczną umożliwia dokładne usunięcie szumu dopiero przy nałożeniu maski 5x5 (rys. 13-18b). Niestety, powoduje to również bardzo widoczne rozmycie obrazu.

Filtr medianowy spełnia swoją rolę o wiele lepiej. Nadal widać różnicę pomiędzy obrazem oryginalnym, a odszumionym lecz porównując tę metodę z poprzednią (rys. 13-18 w porównaniu do rys. 19-24), widać wyraźną poprawę. Podobnie jak w przypadku filtru ze średnią arytmetyczną, po zastosowaniu większych masek, obraz staje się bardzo rozmazany.

6.3. Modyfikacje obrazu w oparciu o histogram

//TODO

6.4. Filtracja liniowa oparta o splot

Algorytm analizowanych filtrów polega na nałożeniu odpowiedniej maski na piksele oraz ich sąsiedztwo. Normalizacja wyników, w celu dokładniejszego wyświetlenia krawędzi, została zrealizowana poprzez dodanie współczynnika wynoszącego 0.7. Mnożone są przez niego wartości wszystkich pikseli wynikowych, jeżeli są mniejsze od maksymalnej.

Filtr dolnoprzepustowy tłumi elementy o wysokiej częstotliwości i wygładza wejściowy obraz. Zarówno rysunek 8-bitowy (rys. 26), jak i 24-bitowy (rys. 27) dobrze prezentują to zachowanie.

Wyostrenie krawędzi dla obrazów w skali szarości przynosi najlepsze efekty przy zastosowaniu maski 2 (rys. 28b). Dla próbek w skali RGB (rys. 29), wyniki są porównywalne.

Wydobywanie szczegółów z tła przynosi, zgodnie z oczekiwaniami, różne rezultaty przy analizie odmiennych obrazów. Wyraźnie można dostrzec korelację pomiędzy konturami elementów na obrazie a zastosowanym kierunkiem filtru (rys. 30-33).

Wydobycie szczegółów z tła bez zdefiniowanego kierunku okazało się skomplikowanym zadaniem w przypadku testowych obrazów Camera i Lena. Algo-

rytm zwrócił niesatysfakcjonujący wynik, odnajdując jedynie część konturów.

Identyfikowanie linii, podobnie jak wydobywanie szczegółów z tła z zastosowanie konkretnego kierunku poszukiwań, jest ściśle związane z analizowanym obrazem. Widać to między innymi po identyfikacji pionowej obrazu Camera (rys. 36b) oraz ukośnej 1 próbki Lena (rys. 37c).

W celu zoptymalizowania filtrów z powyższej sekcji pod względem czasowym i pamięciowym, można zastosować następujące rozwiązania:

- w przypadku masek posiadających zera, można zarezerwować mniejszą ilość pamięci na jej przechowywanie, pomijając te elementy; podczas działania programu, program nie powinien wykonywać podwójnego foru na pikselu (aby obliczyć wartości z całej maski $n \times n$), lecz wybierać tych sąsiadów, którzy faktycznie mają wpływ na finalny wynik
- podobnie jak w poprzednim punkcie można postąpić z elementami maski o wartościach 1 lub -1; zamiast wymnażać je przez wartość piksela i dopiero po tej operacji dodawać go do sumy, można pominąć zbędny krok

6.5. Filtracja nieliniowa

Filtracja nieliniowa umożliwia znalezienie krawędzi elementów, bez względu na ich położenie względem osi XOY.

Operator Robertsa, a zwłaszcza jego drugi wariant, bardzo dokładnie pokazuje wszystkie wyraźne, lokalne zmiany w natężeniu barw. Widać to szczególnie dobrze na rysunkach przedstawiających ptaka (rys. 39a) oraz dziewczynę (rys. 39c).

Operator Sobela również dobrze sobie radzi z odnajdywaniem krawędzi, jednakże oznacza je o wiele grubszymi liniami. Ponieważ bierze pod uwagę większe sąsiedztwo pikseli niż operator Robertsa, dokładnie odwzorowuje kontury elementów pod warunkiem, że nie są one położone blisko siebie (rys. 40a i rys. 40b). W momencie w którym obraz na wejściu nie ma wyraźnych krawędzi znacznie oddalonych od siebie (rys. 40d), wynik jest niesatysfakcjonujący.

Zastosowanie operatora Kirsha przynosi efekty podobne do operatora Sobela, jednakże jeszcze trudniej można się doszukać oryginalnych konturów. Dobrze obrazuje to przykład Pentagonu (rys. 41b) oraz mandryla (rys. 41d). Ze względu na dużą jasność wszystkich wyników, dane wyjściowe możnaby poddać normalizacji.

Operator Rosenfelda odnajduje krawędzi (rys. 42b) lecz tworzy niepotrzebny szum tła (rys. 42a). Zwiększanie parametru R powoduje pogrubianie wynikowych linii.

Wynik przekształcenia operatorem Uolisa został znormalizowany poprzez pomnożenie wartości każdego piksela przez liczbę 10. Bez tej operacji rezul-

tat był nieczytelny. Po transformacji, wynik prezentuje najbardziej wyraziste krawędzie (rys. 45).

7. Wnioski

Podsumowując, do usuwania szumów nie powinno się stosować filtru ze średnią arytmetyczną, ponieważ filtr medianiowy daje lepsze rezultaty. Trzeba jednak pamiętać, aby nie nałożyć zbyt dużej maski, ponieważ otrzymany obraz może się okazać zbyt rozmazany.

Filtracja liniowa oparta o splot potrafi dać bardzo dobre rezultaty wyszukiwania krawędzi pod warunkiem, że szukamy ich w konkretnych kierunkach.

Filtracja nieliniowa daje o wiele lepsze efekty. Szczególnie operator Roberta w wariancie 2 oraz operator Sobela pozwalają na bardzo dokładne wyróżnienie wszystkich krawędzi.

Literatura

- [1] http://ftims.edu.p.lodz.pl/pluginfile.php/18220/mod_resource/content/1/Zadanie1.pdf, 2015
- [2] http://ics.p.lodz.pl/tomczyk/available/po_en/second.html, 2015
- [3] <https://en.wikibooks.org/wiki/LaTeX/Mathematics>, 2015
- [4] https://pl.wikipedia.org/wiki/Filtracja_obraz%C3%B3w, 2015