

Laboratorium

Przetwarzanie obrazów i systemy wizyjne

Ćwiczenie 3

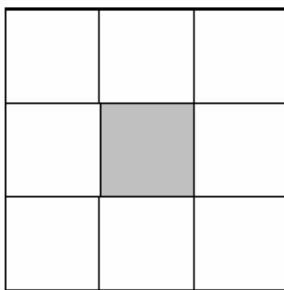
Operacje kontekstowe na obrazach

Operacje kontekstowe (filtry)

Operacje polegają na modyfikacji poszczególnych elementów obrazu w zależności od stanu ich samych i stanu ich otoczenia. Ze względu na kontekstowość mogą zajmować dużo czasu, ale algorytmy są proste i regularne a ponadto mogą być wykonywane na wszystkich punktach obrazu jednocześnie.

Filtracja

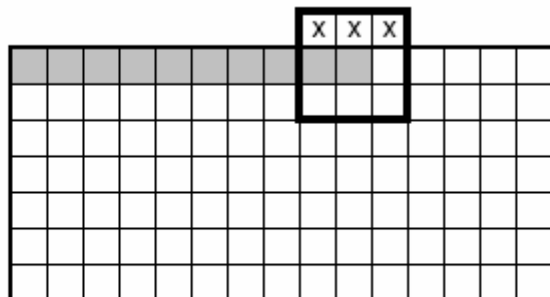
Filtry wykorzystywane do analizy obrazów zakładają, że wykonywane na obrazie operacje będą kontekstowe. Oznacza to, że dla wyznaczenia jednego punktu obrazu wynikowego trzeba dokonać określonych obliczeń na wielu punktach obrazu źródłowego. Zadanie polega na wyznaczeniu wartości funkcji, której argumentami są wartości piksela o tym samym położeniu na obrazie źródłowym oraz wartości pikseli z jego otoczenia K , które może mieć różną formę ale najczęściej utożsamiane jest z kwadratowym oknem otaczającym symetrycznie aktualnie przetwarzany punkt obrazu



Struktura „kontekstu” K wykorzystywanego typowo podczas filtracji obrazów.

Filtrowanie - przeszkody

Z powodu kontekstowości wykonywanych operacji filtracja z reguły nie może dotyczyć pikseli znajdujących się bezpośrednio na brzegu obrazu. Wieloargumentowa funkcja będąca matematycznym zapisem reguły działania filtru nie będzie posiadała wartości punktów oznaczonych symbolem x .



. Niemożność wykonania kontekstowych filtracji dla punktów położonych przy brzegu obrazu.

Filtry – praktyczne zastosowania

Filtry wykorzystywane są zazwyczaj do realizacji następujących celów:

- Stłumienie w obrazie niepożądanego szumu. Przy braku kontekstowych przesłanek na temat istoty szumów realizujący tę funkcję filtr działa na zasadzie lokalnych średnich. Każdemu z punktów obrazu przypisywana jest średnia wartości jego otoczenia.
- Wzmocnienie w obrazie pewnych elementów zgodnych z posiadanym wzorcem. W tym przypadku dany punkt zostanie wzmocniony w stopniu zależnym od stopnia spełnienia przez jego otoczenie określonych warunków.
- Usunięcie określonych wad z obrazu. Na przykład: usunięcie wad powstałych w wyniku zarysowania kliszy fotograficznej.
- Poprawa obrazu o złej jakości technicznej; np.: obrazów nieostrych, poruszonych lub o niewielkim kontraście.
- Rekonstrukcja obrazu, który uległ częściowemu zniszczeniu; np.: rekonstrukcja materiałów fotograficznych, które przez długi czas podlegały działaniu niekorzystnych warunków atmosferycznych

UWAGA: Bardzo istotne jest posiadanie wiedzy co do istoty cech obrazu, które należy osłabić lub wzmocnić. Pomocna jest wiedza odnośnie genezy powstawania tych cech.

Konwolucja dyskretna

W komputerowej analizie obrazu dziedzina funkcji $L(m,n)$ jest dwuwymiarowa i nieciągła gdyż taka jest reprezentacja obrazu. Konwolucję dla dwuwymiarowego dyskretnego obrazu można zapisać:

$$L'(m,n) = (w \times L)(m,n) = \sum_{i,j \in K} L(m-i, n-j) w(i,j)$$

Filtry definiuje się jako tablice współczynników $w(i,j)$. Współczynniki $w(i,j)$ wraz z odpowiednimi elementami obrazu $L(m-i, n-j)$ znajdującego się w oknie K rozłożonym wokół punktu o współrzędnych m,n służą łącznie do obliczania wartości funkcji $L'(m,n)$ w danym punkcie na obrazie wynikowym.

Problem normalizacji

Współczynniki $w(i,j)$ wybiera się zwykle aby były liczbami całkowitymi bo dla zmiennoprzecinkowych wartości operacje matematyczne trwają dłużej. (dla obrazu 512×512 i filtru 9-cio elementowego należy w normalnych warunkach wykonać 2 359 296 dodawań i mnożeń) Po takiej operacji konwolucji nie będzie zazwyczaj spełniony warunek normalizacji $L'(m,n) \in [0, 2B-1]$. Dla filtrów gdzie $w(i,j) \geq 0$ technika normalizacji dana jest wzorem:

$$L'(m,n) = \frac{1}{\sum_{(i,j) \in K} w(i,j)} \sum_{(i,j) \in K} L(m-i, n-j) w(i,j)$$

Dla filtrów gdzie $w(i,j)$ są zarówno dodatnie jak i ujemne normalizacja dana jest wzorem:

$$L''(m,n) = \frac{L'(m,n) - \min L'(m,n)}{\max L'(m,n) - \min L'(m,n)} 2^B$$

Kształt filtra

Rozmiar otoczenia K zależy od projektanta. Im większy rozmiar okna tym bardziej radykalne działanie filtra. Najczęściej przyjmuje się filtry 3×3 , które ogólnie zapisujemy:

$$\begin{bmatrix} w(1,1) & w(1,0) & w(1,-1) \\ w(0,1) & w(0,0) & w(0,-1) \\ w(-1,1) & w(-1,0) & w(-1,-1) \end{bmatrix}$$

lub

$$\begin{bmatrix} w_1 & w_2 & w_3 \\ w_4 & w_5 & w_6 \\ w_7 & w_8 & w_9 \end{bmatrix}$$

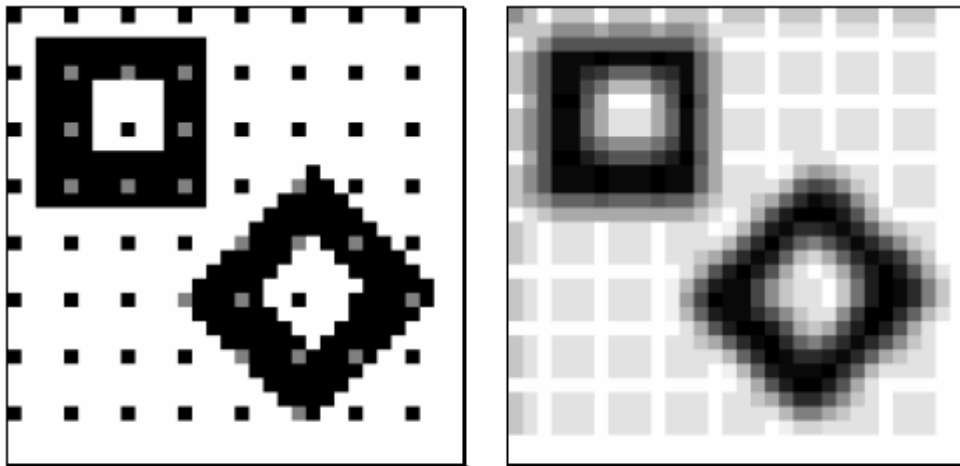
Punkt docelowy $L'(m,n)$ może być ustawiony w środku okna filtracji, a konwolucja będzie wyrażona wzorem:

$$\begin{aligned}
 L'(m,n) = & w_1 L(m-1,n-1) + w_2 L(m-1,n) + w_3 L(m-1,n+1) \\
 & + w_4 L(m,n-1) + w_5 L(m,n) + w_6 L(m,n+1) \\
 & + w_7 L(m+1,n-1) + w_8 L(m+1,n) + w_9 L(m+1,n+1)
 \end{aligned}$$

Filtry dolnoprzepustowe

Typowe zastosowanie filtrów polega na usuwaniu zakłóceń obrazu. Można do tego wykorzystać najprostszy filtr uśredniający:

1	1	1
1	1	1
1	1	1



Obraz z zakłóceniami (po lewej stronie) i efekt jego filtracji z pomocą filtru konwolucyjnego.

Drobne zakłócenia obrazu znikają (są rozmywane). Filtr powoduje rozmycie konturów obiektów i pogorszenie rozpoznawalności ich kształtów.



. Efekt filtracji konwolucyjnej dla niezakłóconego obrazu naturalnego.

W celu zmniejszenia negatywnych skutków filtracji konwolucyjnej stosuje się filtry uśredniające wartości pikseli wewnątrz rozważanego obszaru w sposób ważony:

1	1	1
1	2	1
1	1	1

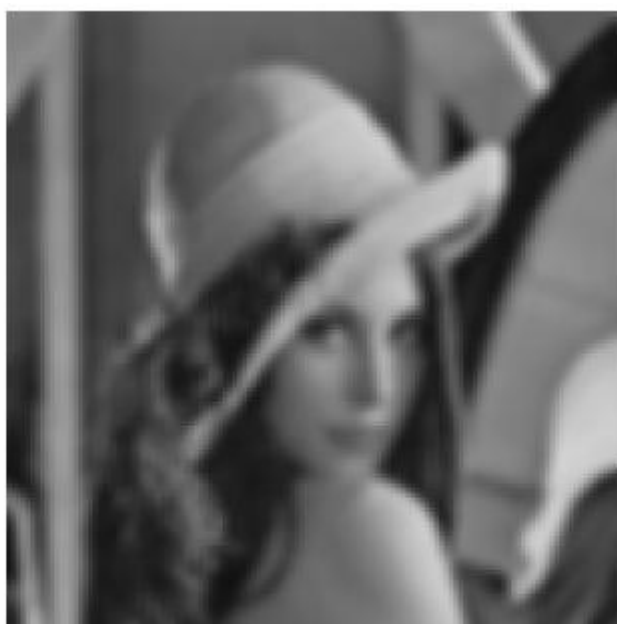
1	2	1
2	4	2
1	2	1



Obraz „Lena” poddany filtracji z uwydatnionym punktem centralnym w macierzy konwolucji.

Można stosować filtrację maskami pozbawionymi elementu centralnego:

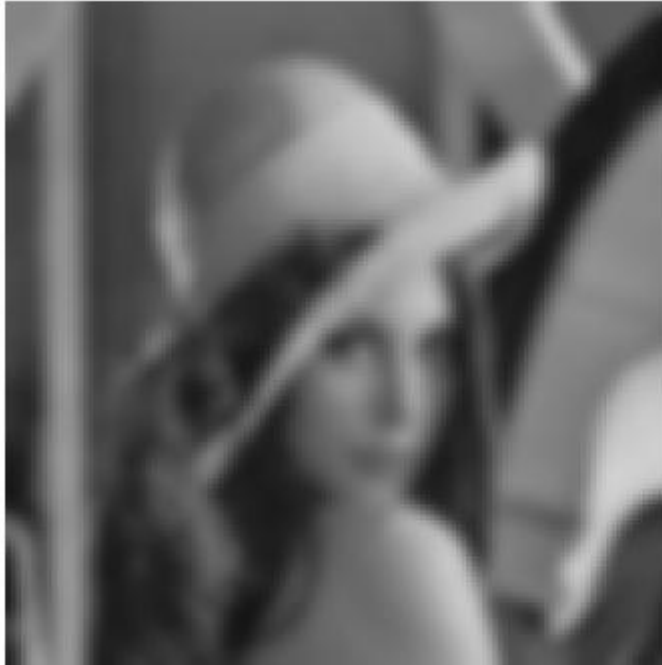
1	1	1
1	0	1
1	1	1



Obraz „Lena” poddany filtracji z pominiętym punktem centralnym w macierzy konwolucji.

Zwiększenie rozmiaru maski

Powoduje bardziej radykalne zmiany w obrazie Zwiększają koszt obliczeniowy. Dla rysunku 512x512:3x3 -> ok. 2,4 mln operacji 5x5 -> ok. 6,5 mln operacji 7x7 -> ok. 16,5 mln operacji. Na ogół unika się większych rozmiarów masek chyba, że trzeba usunąć bardzo dokuczliwe zakłócenia w obrazie.



Obraz „Lena” poddany filtracji za pomocą macierzy konwolucji o rozmiarach 5 x 5.

Filtry górnoprzepustowe-gradienty

Filtry górnoprzepustowe mogą służyć do wydobywania z obrazu składników odpowiedzialnych za szybkie zmiany jasności, a więc konturów, krawędzi, drobnych elementów faktury, itp. Popularnie mówi się, że filtry górnoprzepustowe dokonują wyostrenia obrazu –rozumianych jako uwypuklenie krawędzi w obrazie. Krawędź jest to linia (czasami prosta) oddzielająca obszary o różnej jasności L1, L2. Prosty model matematyczny krawędzi ma postać:

$$i(z) = \begin{cases} 1; z > 0 \\ \frac{1}{2}; z = 0 \\ 0; z < 0 \end{cases}$$

Filtr górnoprzepustowy-przykład

Dla przykładowego gradientu Robertsa macierz współczynników $w(i,j)$ będzie miała postać:

-1	0
0	1

lub

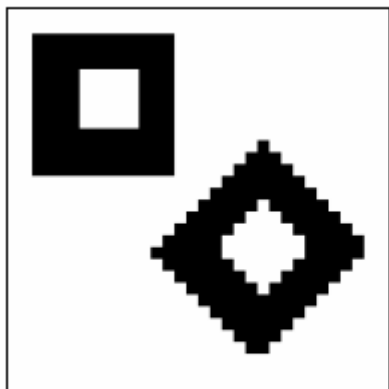
0	0	0
-1	0	0
0	1	0

Ponieważ gradient Roberta generuje ujemne i dodatnie wartości pikseli należy albo dokonać skalowania, albo brać pod uwagę wartość bezwzględną pikseli.

Przy skalowaniu tło z reguły staje się szare, piksele dodatnie ciemne, a piksele ujemne jasne.

Przy modułach efekty są rozmywane i ukrywana jest informacja które piksele były dodatnie a które ujemne.

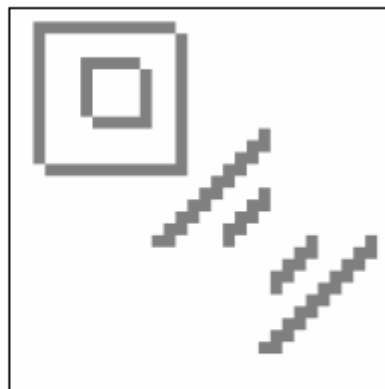
a)



b)



c)



. Filtracja obrazu gradientem Roberta. a) obraz poddawany filtracji, b) obraz po wykonaniu filtracji, c) obraz po wzięciu modułu wartości pikseli po wykonaniu filtracji.



Filtracja obrazu „Lena” gradientem Robertsa (pokazano obraz po wykonaniu filtracji; przeskalowany - po lewej oraz w postaci modułu - po prawej).

Kierunkowość gradientu Robertsa

Gradient Robertsa ma charakter kierunkowy. Jest to kąt 45° i zależy od implementacji maski.

-1	0
0	1

Można to sprawdzić używając gradientu Robertsa dla komplementarnej maski:

0	-1
1	0

lub

0	0	0
0	0	-1
0	1	0



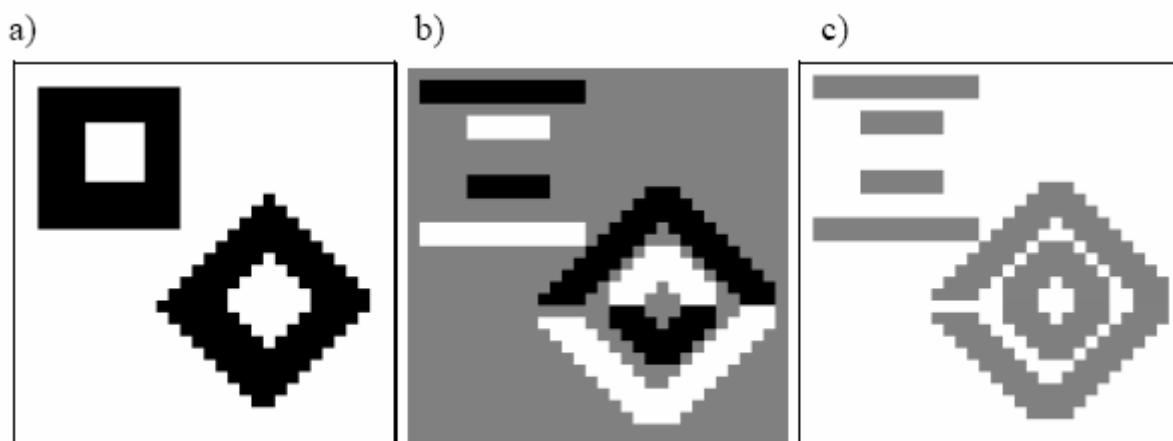
. Filtracja obrazu „Lena” gradientem Roberta o przeciwnej orientacji.

Maska Prewitta

Gradient Roberta można w dość naturalny sposób uogólnić na maski trójwymiarowe. Maską:

-1	-1	-1
0	0	0
1	1	1

po dokonaniu operacji konwolucji dla obrazu sztucznego daje następujący rezultat:



. Sztuczny obraz filtrowany górnoprzepustowo z użyciem poziomej maski Prewitta.

Dla maski Prewitta wzmocnieniu ulegają linie zbliżone do horyzontalnych i wertykalnych



Obraz „Lena” filtrowany górnoprzepustowo z użyciem poziomej maski Prewitta.



Obraz „Lena” filtrowany górnoprzepustowo z użyciem pionowej maski Prewitta.

Przy filtracjach gradientowych można wzmocnić wpływ bezpośrednio najbliższego otoczenia piksela (maska Sobela).

Pozioma maska Sobela

-1	-2	-1
0	0	0
1	2	1



Obraz „Lena” filtrowany górnoprzepustowo z użyciem poziomej maski Sobela.

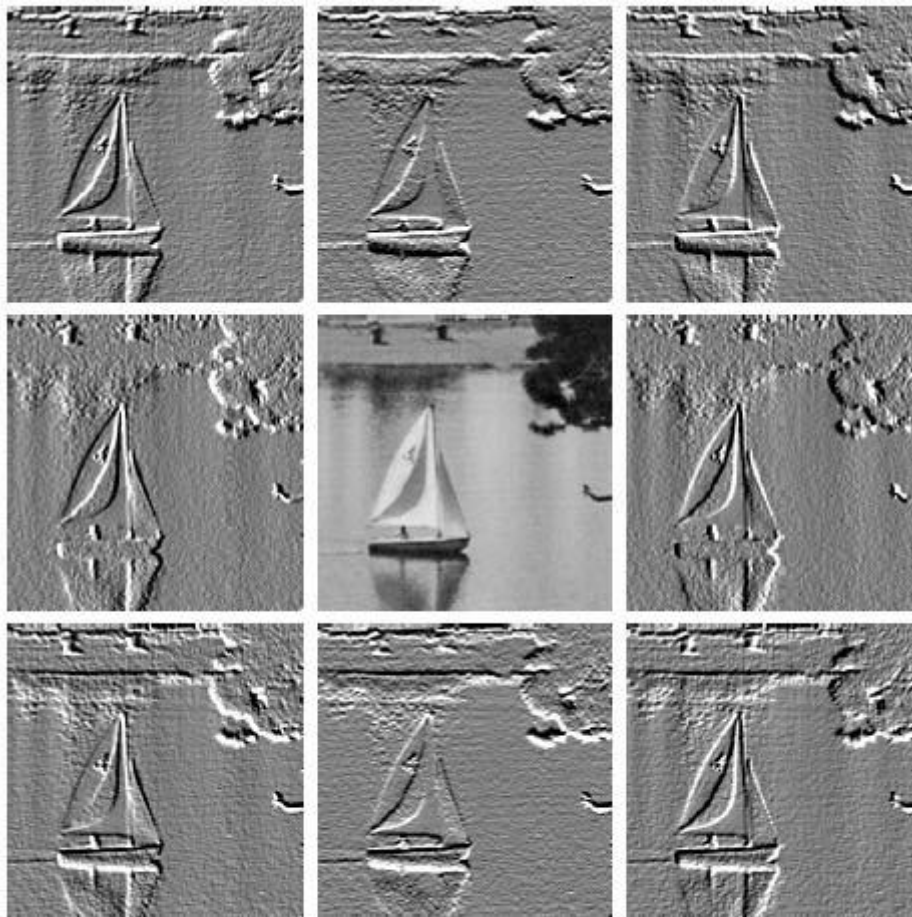
Pionowa maska Sobela:

-1	0	1
-2	0	2
-1	0	1



Obraz „Lena” filtrowany górnoprzepustowo z użyciem poziomej maski Sobela.

Maska Sobela może być obracana nie tylko o 90° ale również o 45°



Obraz żagłówki (w centrum) filtrowany górnoprzepustowo z użyciem masek Sobela o różnych orientacjach.

Maski Sobela o różnej orientacji stosowane do splotu z rysunkiem żagłówki

2	1	0
1	0	-1
0	-1	-2

1	2	1
0	0	0
-1	-2	-1

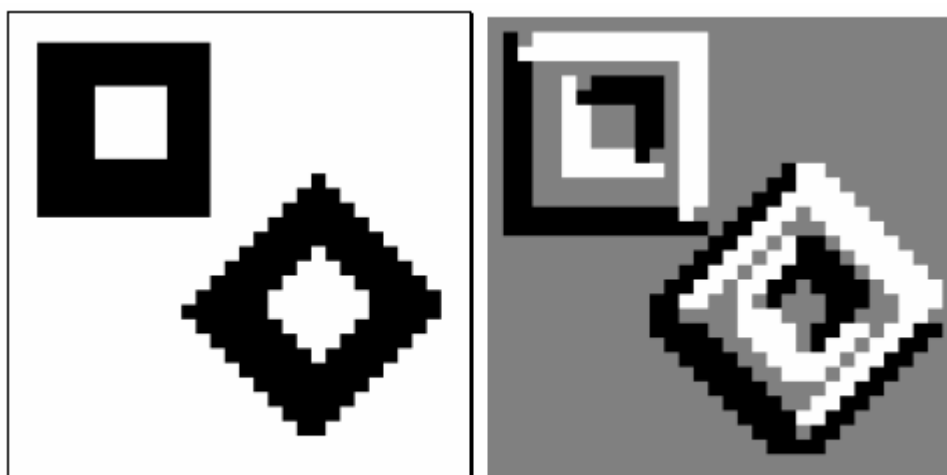
0	1	2
-1	0	1
-2	-1	0

1	0	-1
2	0	-2
1	0	-1
-1	0	1
-2	0	2
-1	0	1
0	-1	-2
1	0	-1
2	1	0
-1	-2	-1
0	0	0
1	2	1
-2	-1	0
-1	0	1
0	1	2

. Maski Sobela użyte do górnoprzepustowej filtracji obrazu na rysunku

Filtry górnoprzepustowe wykrywające narożniki

1	1	1
-1	-2	1
-1	-1	1



Splot obrazu testowego pokazanego po lewej stronie z maską wykrywającą prawy górny narożnik.



Obraz „Lena” po wykonaniu konwolucji z maską wykrywającą narożniki. Po lewej stronie obraz wynikowy skalowany, po prawej - obraz po wykonaniu inwersji i uwzględnieniu tylko bezwzględnych wartości pikseli.

Można tak ustawić maskę aby wykrywała lewy górny narożnik:

1	1	1
1	-2	-1
1	-1	-1



Obraz „Lena” po wykonaniu konwolucji z obroconą maską wykrywającą narożniki.

Istnieje wiele możliwych form w jakich mogą występować maski wykrywające narożniki:

-1	1	1
-1	-2	1
-1	1	1

1	1	1
1	-2	1
-1	-1	-1

-1	-1	1
-1	-2	1
1	1	1

-1	-1	-1
1	-2	1
1	1	1

1	-1	-1
1	-2	-1
1	1	1

1	1	-1
1	-2	-1
1	1	-1

1	1	1
1	-2	-1
1	-1	-1

Jak również

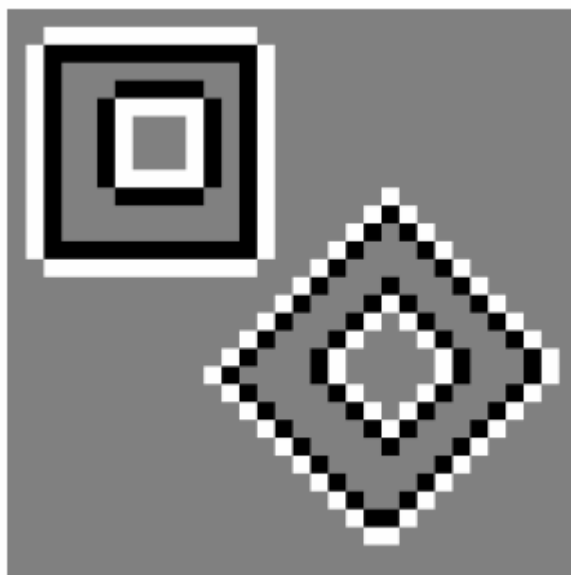
3	3	3
3	0	-5
3	-5	-5

3	-5	-5
3	0	-5
3	3	3

Laplasjany

Czasami istnieje jednak potrzeba zastosowania zmian bezkierunkowych. Dobry i prosty do uzyskania rezultat jest do osiągnięcia dzięki specjalnym maskom – laplasjanom, np.:

0	-1	0
-1	4	-1
0	-1	0



Przeskalowany wynik spłotu sztucznego obrazu testowego z maską laplasjanu.

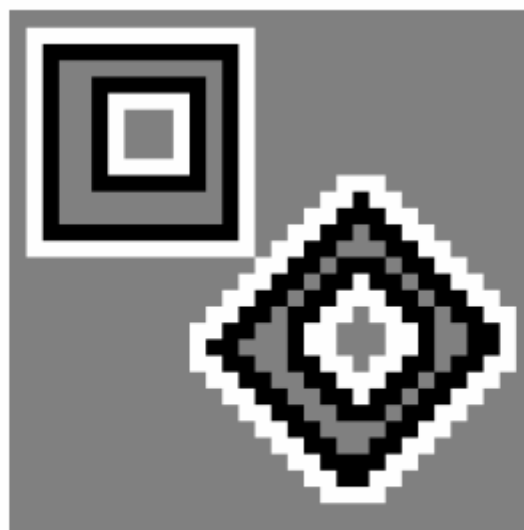
Zastosowanie maski laplasjanu uwypukla i podkreśla wszelkie linie i krawędzie w obrazie



Obraz „Lena” po zastosowaniu laplasjanu. Po lewej obraz skalowany, po prawej - moduł wartości pikseli.

Laplasjan może być też zdefiniowany za pomocą nieco bardziej złożonej maski:

-1	-1	-1
-1	8	-1
-1	-1	-1



Przeskalowany wynik splotu sztucznego obrazu testowego z maską rozbudowanego laplasjanu.



Obraz „Lena” po zastosowaniu rozbudowanego laplasjanu.

Inne sposoby definicji laplasjanu pokazują podane poniżej maski:

1	-2	1
-2	4	-2
1	-2	1

-1	-1	-1
-1	9	-1
-1	-1	-1

0	-1	0
-1	5	-1
0	-1	0

1	-2	1
-2	5	-2
1	-2	1

Maska może mieć również rozmiary 5x5

0,25	0,5	0,5	0,5	0,25
0,5	-0,25	-1	-0,25	0,5
0,5	-1	-2	-1	0,5
0,5	-0,25	-1	-0,25	0,5
0,25	0,5	0,5	0,5	0,25

Filtracja w module Scipy

Wybrane funkcje do filtracji obrazów z modułu `scipy.ndimage`

`gaussian_filter(input, sigma[, order, ...])` Multidimensional Gaussian filter.

`gaussian_filter1d(input, sigma[, axis, ...])` 1-D Gaussian filter.

`gaussian_gradient_magnitude(input, sigma[, ...])` Multidimensional gradient magnitude using Gaussian derivatives.

`gaussian_laplace(input, sigma[, output, ...])` Multidimensional Laplace filter using Gaussian second derivatives.

`generic_filter(input, function[, size, ...])` Calculate a multidimensional filter using the given function.

`generic_gradient_magnitude(input, derivative)` Gradient magnitude using a provided gradient function.

`generic_laplace(input, derivative2[, ...])` N-D Laplace filter using a provided second derivative function.

`laplace(input[, output, mode, cval])` N-D Laplace filter based on approximate second derivatives.

`prewitt(input[, axis, output, mode, cval])` Calculate a Prewitt filter.

`rank_filter(input, rank[, size, footprint, ...])` Calculate a multidimensional rank filter.

`sobel(input[, axis, output, mode, cval])` Calculate a Sobel filter.

`uniform_filter(input[, size, output, mode, ...])` Multidimensional uniform filter.

Zadania

1. Utworzyć własną funkcję realizującą filtrację obrazów znajdujących się w tablicy, dla dowolnego rozmiaru maski i różnych metod obliczeń na brzegach obrazów.

2. Wykonać filtrację z wykorzystaniem filtrów pokazanych w instrukcji.
3. Znaleźć odpowiedniki filtrów pokazanych w instrukcji w bibliotece scikits-image filters (<https://scikit-image.org/docs/stable/api/skimage.filters.html>) i pokazać ich użycie.
4. Przeanalizować wybrane funkcje z modułu scikit-image do filtracji obrazów oraz pokazać przykłady ich użycia.