

Laboratorium
Przetwarzania Sygnałów

Ćwiczenie 6

„Filtracja 2D”

Opracowali:

- dr inż. Krzysztof Mikołajczyk
- dr inż. Beata Leśniak-Plewińska

Zakład Inżynierii Biomedycznej
Instytut Metrologii i Inżynierii Biomedycznej
Wydział Mechatroniki Politechniki Warszawskiej

Warszawa, 2009



1. Cel ćwiczenia.

W ramach ćwiczenia studenci zapoznają się z podstawami filtracji liniowej (filtr uśredniający, filtr krawędziowy) i nieliniowej (filtr medianowy) obrazów w dziedzinie przestrzeni i częstotliwości.

2. Wymagane wiadomości.

1. Filtracja obrazów w dziedzinie przestrzeni: splot (konwolucja) i jego właściwości; podstawowe rodzaje filtrów: filtr uśredniający (gładzący), filtr krawędziowy, filtr medianowy.
2. Filtracja obrazów w dziedzinie częstotliwości przestrzennej: transformata Fouriera i jej właściwości.

3. Literatura

1. W. Malina i M. Siemiatacz „Cyfrowe przetwarzanie obrazów”, Akademicka Oficyna Wydawnicza EXIT, Warszawa, 2008
2. R. Tadeusiewicz i P. Korohoda „Komputerowa analiza i przetwarzanie obrazów”, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków, 1997 (http://winntbg.bg.agh.edu.pl/skrypty2/0098/komputerowa_analiza.pdf)
3. R.C. Gonzalez i R.E. Woods “Digital Image Processing Using MATLAB”, Prentice Hall, 2004

4. Spis funkcji i instrukcji przydatnych podczas realizacji ćwiczenia.

Opisane w tym punkcie funkcje nie są standardowymi wbudowanymi funkcjami MATLAB’a i zostały stworzone na potrzeby Laboratorium PTS.

O ile w trakcie realizacji ćwiczenia zajdzie potrzeba skorzystania ze standardowej funkcji (polecenia) lub instrukcji MATLAB’a, należy się z nią zapoznać wykorzystując pomoc MATLAB’a za pomocą polecenia `help` lub `doc`, np. `help sin` lub `doc sin`.



```
h = pokaz_obraz (X, 'NazwaParametru', WartoscParametru)
```

Funkcja służy do tworzenia ilustracji danych 2D.

Parametrami wyjściowymi funkcji są:

1. **h** – uchwyt do obiektu graficznego,

Parametrami wejściowymi funkcji są:

1. **X** – dane obrazowe (macierz lub skalar):
2. '**NazwaParametru**', **WartoscParametru** - nazwa i wartość parametru określającego wybrane właściwości obiektu graficznego. Parametry opisano w poniższej tabeli.

<i>Nazwa parametru</i>	<i>Opis i wartości parametru</i>
' mapa '	Macierz liczbowa określająca paletę barw. Standardowe palety barw dostępne w Matlab'ie: hsv, jet, hot, gray, cool, bone, cooper, pink, prism, flag . Wartość domyślna: gray .
' typ '	Łańcuch znakowy określający rodzaj skalowania wartości elementów obrazu; ' skalowany ' - tworzy ilustrację, w której wartości elementów macierzy (pikseli) są przeskalowane tak, aby wykorzystał pełen zakres aktualnej palety barw; ' nieskalowany ' - brak skalowania. Wartość domyślna: ' skalowany '

Przykłady:

```
h = pokaz_obraz(lena, 'mapa', gray, 'typ', 'nieskalowany');
```

Powyższa komenda utworzy ilustrację 2D danych zawartych w zmiennej **lena**. Paletą barw będą odcienie szarości. Dane nie będą skalowane i zwrócony zostanie uchwyt do utworzonego obiektu graficznego (ilustracji) i przypisze go zmiennej **h**.

```
pokaz_obraz(lena);
```

Powyższa komenda utworzy ilustrację 2D danych zawartych w zmiennej **lena**. Paleta barw i rodzaj skalowania będą miały wartości domyślne.

4.a Dodatek

Transformacja Fouriera (TF)

Transformacja Fouriera (TF) jest transformacją całkową z dziedziny czasu w dziedzinę częstotliwości. Transformata Fouriera jest wynikiem transformacji Fouriera. Transformacja jest operacją (operatorem) na funkcji wejściowej (operand), dającą w wyniku transformatę (funkcja wyjściowa). TF rozkłada funkcję na szereg funkcji okresowych (sinus i/lub cosinus) tak, że uzyskana transformata zawiera informację o wartościach amplitudy i fazy poszczególnych harmonicznym składających się na pierwotną funkcję wejściową. Analizę tego typu nazywa się analizą spektralną lub widmową.

W dziedzinie przetwarzania sygnałów jednowymiarowych TF jest fundamentalnym narzędziem, szczególnie przydatnym w analizie procesów zależnych od czasu, a zwłaszcza analizie drgań (akustyce, analizie sygnału mowy, sygnałów radiowych czy telekomunikacyjnych). Dla sygnałów cyfrowych stosowana jest jej zmodyfikowana wersja zwana Dyskretną Transformacją Fouriera (DTF).

Ze względu na różną specyfikę zmysłu wzroku i słuchu analiza i przetwarzanie obrazów w dziedzinie częstotliwości jest rzadziej stosowane niż analiza i przetwarzanie w dziedzinie częstotliwości sygnałów akustycznych. W analizie obrazów z TF korzysta się najczęściej do:

- detekcji tych cech obrazu, które są łatwiejsze do analizy w dziedzinie częstotliwości przestrzennej (widmie - transformacie Fouriera obrazu) niż w dziedzinie przestrzeni,
- szybkiego obliczania splotu,
- wzmocnienia lub usunięcia pewnych składowych widma obrazu (odpowiadające np. szumom), a następnie przez tzw. odwrotną transformację Fouriera, odtworzenie skorygowanego obrazu.

Z matematycznego punktu widzenia zastosowanie DTF do obrazów cyfrowych jest poszerzeniem odpowiednich zależności do dwóch wymiarów.

Obraz cyfrowy można traktować jako uporządkowany i ponumerowany dwuwymiarowy zbiór L liczb rzeczywistych (ciąg dwuwymiarowy o wartościach rzeczywistych) określony wzorem (1):

$$L = L(m, n) \in \mathbb{R} : m = 0, 1, \dots, M-1 ; n = 0, 1, \dots, N-1 \quad (1)$$

DTF dla obrazu cyfrowego określa równanie (2):

$$F(i, k) = \beta_L \cdot \sum_{m=0}^{M-1} \left[\sum_{n=0}^{N-1} L(m, n) \cdot \exp\left(\frac{-j \cdot 2 \cdot \pi \cdot m \cdot i}{M}\right) \right] \cdot \exp\left(\frac{-j \cdot 2 \cdot \pi \cdot n \cdot k}{N}\right) \quad (2)$$

$$\text{dla } i = 0, 1, 2, \dots, M-1 ; k = 0, 1, 2, \dots, N-1$$

Operacja dwuwymiarowej odwrotnej DTF jest zdefiniowana wzorem (3):

$$L(m, n) = \beta_F \cdot \sum_{i=0}^{M-1} \left[\sum_{k=0}^{N-1} F(i, k) \cdot \exp\left(\frac{j \cdot 2 \cdot \pi \cdot m \cdot i}{M}\right) \right] \cdot \exp\left(\frac{j \cdot 2 \cdot \pi \cdot n \cdot k}{N}\right) \quad (3)$$

$$\text{dla } m = 0, 1, 2, \dots, M-1 ; n = 0, 1, 2, \dots, N-1$$

Aby operacje określone równaniami (2) i (3) były wzajemnymi odwrotnościami musi zachodzić zależność:

$$\beta_L \cdot \beta_F = \frac{1}{M \cdot N} \quad (4)$$

Jeśli przyjmiemy:

$$\beta_L = 1; \beta_F = \frac{1}{M \cdot N} \quad (5)$$

w dziedzinie częstotliwości element o indeksach (0,0) będzie sumą wszystkich wartości dwuwymiarowego ciągu L (obrazu):

$$F(0,0) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} L(m, n) \quad (6)$$

Dwuwymiarowy ciąg $F(i, k)$ przyjmuje wartości ze zbioru liczb zespolonych i posiada dokładnie tyle samo elementów co dwuwymiarowy ciąg $L(m, n)$ będący reprezentacją cyfrowego obrazu. Można go zinterpretować jako specyficzny obraz, łącznie z określeniem poszczególnych jego elementów jako pikseli. W celu uniknięcia nieporozumień obraz w dziedzinie transformaty Furiera określa się F -obrazami, a ich elementy F -pikselami.

Wyrażenie (2) możemy w skrócie zapisać w postaci kanonicznej (algebraicznej) (7):

$$F(i, k) = \Re(F(i, k)) + j \cdot \Im(F(i, k))$$

gdzie $\Re(F(i, k))$ – część rzeczywista $F(i, k)$; $\Im(F(i, k))$ – część urojona $F(i, k)$;
 $j = \sqrt{-1}$ – jednostka urojona
 (7)

w postaci wykładniczej (8):

$$F(i, k) = A(i, k) \cdot e^{j\phi(i, k)}$$

gdzie $A(i, k) = |F(i, k)| = \sqrt{\Re(F(i, k))^2 + \Im(F(i, k))^2}$ – moduł $F(i, k)$;
 $\phi(i, k)$ – argument $F(i, k)$ (8)

lub w postaci trygonometrycznej (9):

$$F(i, k) = A(i, k) \cdot [\cos(\phi(i, k)) + j \cdot \sin(\phi(i, k))]$$

gdzie $|F(i, k)|$ – moduł $F(i, k)$; $\cos(\phi(i, k)) = \frac{\Re(F(i, k))}{|F(i, k)|}$; $\sin(\phi(i, k)) = \frac{\Im(F(i, k))}{|F(i, k)|}$ (9)

Dwuwymiarowa transformacja Fouriera jest separowalna ze względu na wymiar. Nawiasy kwadratowe we wzorach (2) i (3) wskazują fakt, że najpierw można wyznaczyć jednowymiarowe transformaty Fouriera dla jednego wymiaru (np. kolumn), a następnie tak otrzymany obraz pośredni przetransformować dla drugiego wymiaru (wierszy). Ostatecznie, po obu etapach, obraz może zostać opisany jako suma dwuwymiarowych ciągów bazowych z odpowiednimi współczynnikami (10 lub 11):

$$L(m, n) = \sum_{i=0}^{M/2} \sum_{k=0}^{N/2} a_{i,k} \cdot L_{i,k}(m, n) \quad (10)$$

gdzie M i N – parzyste

lub

$$L(m, n) = \sum_{i=0}^{(M-1)/2} \sum_{k=0}^{(N-1)/2} a_{i,k} \cdot L_{i,k}(m, n) \quad (11)$$

gdzie M i N – nieparzyste

Każdy z ciągów składowych $L_{i,k}$ otrzymany jest z iloczynu elementarnych ciągów jednowymiarowych:

$$L_{i,k}(m, n) = l_{m,i} \cdot l_{n,k}$$

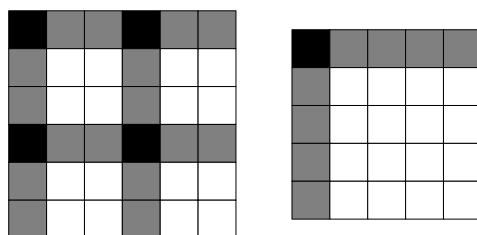
$$\text{gdzie } l_{m,i} = \cos\left(\frac{2 \cdot \pi \cdot m \cdot \Delta t}{T_i} + \phi_i\right); l_{n,k} = \cos\left(\frac{2 \cdot \pi \cdot n \cdot \Delta t}{T_k} + \phi_k\right) \quad (12)$$

Położenie współczynnika po pełnej dwuwymiarowej DTF na dwuwymiarowej siatce indeksów jednoznacznie informuje, z jaką kombinacją ciągów jednowymiarowych należy go łączyć. Np. w punkcie $(0,0)$ znajduje wartość średnia (lub suma) – zwana często składową stałą, w punkcie $\left(0, \frac{N}{2}\right)$ (lub $\left(0, \frac{N-1}{2}\right)$) współczynnik określający udział w obrazie ciągu powstałego ze stałego ciągu wierszowego i najszybciej zmiennego ciągu kolumnowego, a w punkcie $\left(\frac{M}{2}, \frac{N}{2}\right)$ (lub $\left(\frac{M-1}{2}, \frac{N-1}{2}\right)$) umieszczony jest współczynnik związany z połączeniem najszybciej zmiennych ciągów kolumnowego i wierszowego.

Zależności pomiędzy modułami wartości transformaty $F(i,k)$ ($A(i,k)$) oraz współczynnikami $a_{i,k}$ we wzorach (10) i (11) jest następująca (na rysunku 1 pokazano rozmieszczenie F -pikseli poszczególnych typów):

$$A(i, k) = \begin{cases} a_{i,k} \cdot 1/\beta_L & \text{dla } F\text{-pikseli typu 1 (Rys.1. piksele w kolorze czarnym)} \\ 1/2 \cdot a_{i,k} \cdot 1/\beta_L & \text{dla } F\text{-pikseli typu 2 (Rys.1. piksele w kolorze szarym)} \\ 1/4 \cdot a_{i,k} \cdot 1/\beta_L & \text{dla } F\text{-pikseli typu 3 (Rys.1. piksele w kolorze białym)} \end{cases} \quad (13)$$

Sama znajomość współczynników $a_{i,k}$ równania (10) lub (11) nie wystarcza by jednoznacznie określić ciąg L . Potrzebna jest jeszcze informacja o tym jak były przesunięte w fazie obie funkcje kosinus, które służą do wygenerowania ciągów jednowymiarowych $l_{m,i}$, oraz $l_{n,k}$ (wzór 12). Faza jednego F -piksela nie może jednoznacznie określić dwóch faz kosinusów generujących ciąg bazowy. Oznaczmy przez ϕ_i fazę kosinusa generującego ciąg jednowymiarowy $l_{m,i}$, przez ϕ_k fazę kosinusa generującego ciąg jednowymiarowy $l_{n,k}$, a przez $\phi(i,k)$ fazę F -piksela o współrzędnych (i,k) (porównaj ze wzorami (8) i (9)). F -piksele typu 1 (Rys.1) powinny być zawsze rzeczywiste. F -piksele typu 2 wiążą się z dwuwymiarowymi ciągami bazowymi, które powstały z ciągu stałego (bez fazy) i z jednej kosinusoidy (którą można przesuwac fazowo). Zarówno dla F -pikseli typu 1 jak i 2 łatwo jest określić związek pomiędzy odpowiednimi fazami. Natomiast dla wyjaśnienia zależności fazowych F -pikseli typu 3 wygodniej będzie najpierw wyjaśnić symetrie zachodzące w dwuwymiarowej TF obrazu cyfrowego, który składa się z pikseli o wartościach rzeczywistych (bez części urojonej).

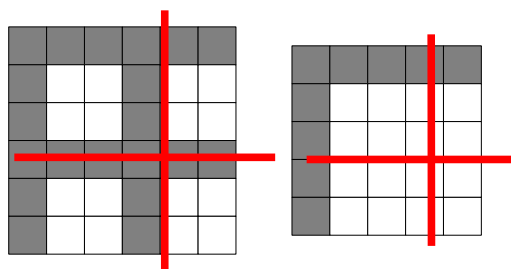


Rysunek 1: Lokalizacja F -pikseli poszczególnych typów dla wzoru 13: z lewej dla obrazu o rozmiarach parzystych (6x6), z prawej dla obrazu o rozmiarach nieparzystych (5x5); kolory pikseli: czarny – typu 1, szary – typu 2, biały – typu 3

Symetrie dwuwymiarowej TF obrazu o rzeczywistych wartościach pikseli

W przypadku gdy oryginalny ciąg składa się z elementów o wartościach wyłącznie rzeczywistych, w dziedzinie transformaty Fouriera powstają specyficzne symetrie. Zagadnienie to zostanie opisane w sposób uproszczony, bez wnikania w szczegóły analityczne.

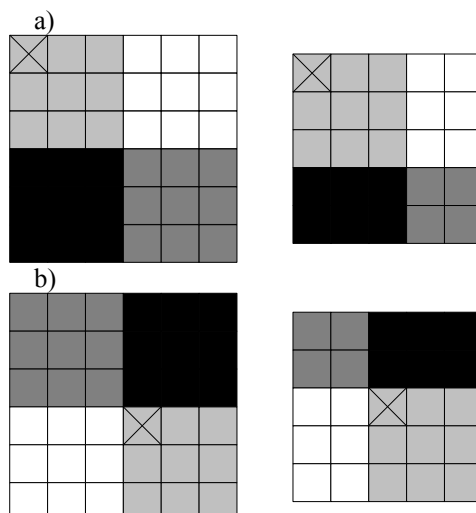
Rozważmy dwa obrazy jeden o rozmiarach parzystych np. 6x6 pikseli, drugi o rozmiarach nieparzystych np 5x5 pikseli (Rys.2).



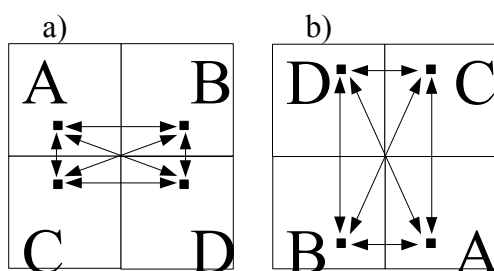
Rysunek 2: Ilustracja symetrii w dziedzinie TF: z lewej obraz o rozmiarach parzystych (6x6), z prawej obraz o rozmiarach nieparzystych (5x5)

Pogrubione czerwone linie na rysunku 2 pokazują osie symetrii amplitudy transformaty Fouriera, a punkt przecięcia tych linii wyznacza punkt symetrii dla fazy. F -piksele oznaczone kolorem szarym nie są związane z powyższymi symetriami. Są to F -piksele typu 1 lub 2 i spełniają one warunki dotyczące ciągów jednowymiarowych.

W przypadku dwuwymiarowej DTF, podobnie jak w przypadku jednowymiarowym, stosuje się zmianę organizacji wynikowej transformaty, tak aby F -piksel odpowiadający składowej stałej znajdujący się pierwotnie się w lewym górnym rogu (niezależnie od tego czy rozmiar obrazu jest parzysty, czy nieparzysty) znalazł się w środku obrazu. W przypadku obrazów o rozmiarze nieparzystym ten F -piksel znajduje się dokładnie w centrum obrazu (zakreślony – rys.3), a w przypadku obrazów o rozmiarze parzystym - lewym górnym rogu prawej dolnej ćwiartki F -obrazu (zakreślony – rys.3). Ponadto, po takim przesunięciu, na krawędziach F -obrazu znajdują się F -piksele odpowiadające największym częstotliwościom kosinusoid tworzących ciągi bazowe. Rysunek 3 demonstruje schematycznie to przesunięcie w przypadku obrazów o rozmiarach parzystych i nieparzystych. Rysunek 4 przedstawia relacje symetrii zachodzące pomiędzy F -pikselami należącymi do poszczególnych ćwiartek F -obrazu. Są to symetrie dwojakiego typu: symetria osiowa – względem danej linii poziomej lub pionowej, oraz symetria punktowa – względem punktu obrazu, wyznaczonego przez przecięcie linii symetrii poziomej i pionowej.



Rysunek 3: Przesunięcia poszczególny obszarów F -obrazów - z lewej F -obraz o rozmiarze parzystym 6×6 , z prawej o rozmiarze nieparzystym 5×5 ; a) F -obrazy przed przesunięciem, b) F -obrazy o przesunięciu



Rysunek 4: Ilustracja symetrii zachodzących w uproszczonej (z pominięciem F -pikseli) interpretacji F -obrazu, linie ze strzałkami oznaczają zachodzące rodzaje symetrii, a) F -obraz przed przesunięciem, b) F -obraz po przesunięciu

Wprowadźmy następujące oznaczenia, określające wszystkie F -piksele obrazu w odniesieniu do obszaru A (rys. 4):

- $F_A(i, k)$ - F -piksel należący do obszaru A o położeniu wyznaczonym przez indeksy (i, k) ;
- $F_{B/A}(i, k)$ - F -piksel należący do obszaru B o położeniu symetrycznym do $F_A(i, k)$ względem osi pionowej;
- $F_{C/A}(i, k)$ - F -piksel należący do obszaru C o położeniu symetrycznym do $F_A(i, k)$ względem osi poziomej;
- $F_{D/A}(i, k)$ - F -piksel należący do obszaru D o położeniu symetrycznym do $F_A(i, k)$ względem środka obrazu.

Jeśli $A_{B/A}(i, k)$ oznacza amplitudę F -piksela należącego do obszaru B i będącego w odpowiedniej symetrii względem F -piksela obszaru A o indeksach (i, k) i $\varphi_{B/A}(i, k)$ oznacza fazę tego samego F -piksela należącego do obszaru B , to:

$$\begin{aligned} A_A(i, k) &= A_{B/A}(i, k) \\ A_{B/A}(i, k) &= A_{C/A}(i, k) \\ A_{C/A}(i, k) &= A_{D/A}(i, k) \end{aligned} \quad (14)$$

Zatem wszystkie cztery F -piksele wyszczególnione na rysunku 4 posiadają tę samą amplitudę. Relacje fazowe są opisane następującymi zależnościami:

$$\begin{aligned}\phi_A(i, k) &= -\phi_{D/A}(i, k) \\ \phi_{B/A}(i, k) &= -\phi_{C/A}(i, k)\end{aligned}\quad (15)$$

Nie ma związku pomiędzy fazami: $\phi_A(i, k)$ i $\phi_{B/A}(i, k)$ ani pomiędzy $\phi_{C/A}(i, k)$ i $\phi_{D/A}(i, k)$. Zatem relacje fazowe (wzór 14) dotyczą jedynie symetrii punktowej względem środka F -obrazu.

Ponadto, jeśli dany F -piksel typu 3 należy do obszaru A , to jego faza jest powiązana fazami kosinusów generujących (wzór 12) zgodnie ze wzorem:

$$\phi(i, k) = \phi_i + \phi_k \quad (16)$$

Uwzględniając wzór 16 można uzyskać wzory na dwie szukane fazy jednowymiarowe:

$$\begin{aligned}\phi_A(i, k) &= \phi_i + \phi_k \\ \phi_{B/A}(i, k) &= \phi_i - \phi_k\end{aligned}\quad (17)$$

5. Przebieg ćwiczenia

Przed przystąpieniem do realizacji ćwiczenia należy poprosić prowadzącego o wskazanie folderu zawierającego dane niezbędne do realizacji ćwiczenia.

Oznaczenia użyte w tekście:

sygnał – nazwa funkcji, która powinna zostać użyta w celu rozwiązania danego problemu,

s1 – nazwa zmiennej,

`s1a=s1(1:end/4)` – polecenie Matlab'a, które należy wpisać w okno komend (ang.: command window).

$s(t) = A \cdot \sin(2 \cdot \pi \cdot f \cdot t)$ - opis w notacji matematycznej.

1. Filtracja dolnoprzepustowa – filtr uśredniający

- Z pliku 'pruski_mur.mat' załadować do przestrzeni roboczej obraz **pruski_mur**.
- Wyświetl obraz **pruski_mur** (**pokaz_obraz**, dobierz wartość parametru: 'typ', 'mapa' - wartość domyślna).
- Wykonaj filtrację obrazu **pruski_mur** (**imfilter** - standardowa funkcja modułu Image Processing pakietu MATLAB) dla następujących jąder (masek) filtru (jeśli trzeba znormalizuj maskę filtru):

$$\blacksquare \quad h1 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix},$$

$$\blacksquare \quad h2 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 4 & 1 \\ 1 & 1 & 1 \end{bmatrix},$$

$$\blacksquare \quad h3 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix},$$

$$\blacksquare \quad h4 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Zilustruj wyniki filtracji (**pokaz_obraz**, dobierz wartość parametru: 'typ', 'mapa' - wartość domyślna).

Wyjaśnij działanie czterech filtrów p. 1.c i różnice w uzyskanych w wyniku ich działania obrazach.

- d. Dodaj do obrazu **pruski_mur** szum gaussowski (*imnoise* - standardowa funkcja modułu Image Processing pakietu MATLAB, zastosuj domyślne wartości parametrów: 'm' i 'v').

Zilustruj wyniki filtracji (*pokaz_obraz*, dobierz wartość parametru: 'typ', 'mapa' - wartość domyślna).

- e. Wykonaj filtrację zaszumionego obrazu **pruski_mur** dla jąder filtru z p. 1.c (*imfilter* - standardowa funkcja modułu Image Processing pakietu MATLAB) (jeśli trzeba znormalizuj maskę filtru).

Zilustruj wyniki filtracji (*pokaz_obraz*, dobierz wartość parametru: 'typ', 'mapa' - wartość domyślna).

Co osiągnięto za pomocą zastosowanych filtracji? Kosztem czego?

- f. Dodaj do obrazu **pruski_mur** szum impulsowy (typu 'salt & pepper') (*imnoise* - standardowa funkcja modułu Image Processing pakietu MATLAB, zastosuj domyślną wartość parametru 'd').

Zilustruj wyniki filtracji (*pokaz_obraz*, dobierz wartość parametru: 'typ', 'mapa' - wartość domyślna).

- g. Wykonaj filtrację zaszumionego obrazu **pruski_mur** dla jąder filtru z p. 1.c (*imfilter* - standardowa funkcja modułu Image Processing pakietu MATLAB) (jeśli trzeba znormalizuj maskę filtru).

Zilustruj wyniki filtracji (*pokaz_obraz*, dobierz wartość parametru: 'typ', 'mapa' - wartość domyślna).

Porównaj skuteczność filtracji dla obydwu rodzajów szumu.

2. Filtracja nieliniowa - filtr medianowy

- a. Poddaj obraz **pruski_mur** filtracji medianowej dla następujących rozmiarów jąder filtru medianowego (*medfilt2* - standardowa funkcja modułu Image Processing pakietu MATLAB):

- 3 x 3;
- 5 x 5;
- 11 x 11;
- 21 x 21;

Zilustruj wyniki filtracji (*pokaz_obraz*, dobierz wartość parametru: 'typ', 'mapa' - wartość domyślna).

Jakie są efekty działania filtru medianowego?

- b. Poddaj filtracji medianowej obrazy **pruski_mur** zaszumione szumem gausowskim (pp. 1.d) i szumem impulsowym (pp. 1.f) (*medfilt2* - standardowa

funkcja modułu Image Processing pakietu MATLAB, samodzielnie dobierz optymalny rozmiar filtru).

Zilustruj wyniki filtracji (*pokaz_obraz*, dobierz wartość parametru: 'typ', 'mapa' - wartość domyślna).

Porównaj skuteczność filtracji medianowej dla obydwu rodzajów szumu z filtracją przeprowadzoną w pp. 1.f oraz 1.i.

3. Filtracja górnoprzepustowa – filtr krawędziowy

- Z pliku 'kolo.mat' załadować do przestrzeni roboczej obraz **kolo**.
- Wykonaj filtrację obrazu **kolo** (*imfilter* - standardowa funkcja modułu Image Processing pakietu MATLAB) dla następujących jąder filtru (Prewitt'a) (jeśli trzeba dokonaj rzutowania typu zmiennej zawierającej filtrowany obraz):

$$\bullet \quad hp1 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix},$$

$$\bullet \quad hp2 = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -1 \end{bmatrix},$$

$$\bullet \quad hp3 = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}.$$

Zilustruj wyniki filtracji (*pokaz_obraz*, dobierz wartość parametru: 'typ', 'mapa' - wartość domyślna).

Wyjaśnij działanie trzech powyższych filtrów i różnice w uzyskanych obrazach wynikowych.

4. Filtracja w dziedzinie częstotliwości

- Wyznacz widmo amplitudowe obrazu **pruski_mur** (*fft2* - standardowa funkcja modułu Image Processing pakietu MATLAB, *abs* - standardowa funkcja pakietu MATLAB).

Zilustruj widmo amplitudowe (w celu uwidocznienia słabszych składowych zastosuj skalę logarytmiczną - użyj funkcji logarytmicznej) (*log* - standardowa funkcja pakietu MATLAB, *pokaz_obraz*, dobierz wartość parametru: 'typ', 'mapa' - wartość domyślna).

- Przekształć widmo obrazu **pruski_mur** tak, aby element odpowiadający składowej stałej znajdował się w „środku” wyznaczonego widma (*fftshift* - standardowa funkcja modułu Image Processing pakietu MATLAB) i zilustruj je (*pokaz_obraz*, dobierz wartość parametru: 'typ', 'mapa' - wartość domyślna).

- c. Wyznacz charakterystykę amplitudowo-częstotliwościową (A-F) jednego z wcześniej użytych jąder filtru, np. jądra `h4` (p. 1.c) (rozmiar charakterystyki A-F musi być zgodny z rozmiarem obrazu `pruski_mur`) (`freqz2` - standardowa funkcja modułu Image Processing pakietu MATLAB)

Zilustruj charakterystykę A-F (`pokaz obraz`, dobierz wartość parametru: `'typ'`, `'mapa'` - wartość domyślna).

- d. Przekształć charakterystykę A-F tak, aby element odpowiadający składowej stałej znajdował się w lewym górnym rogu (`ifftshift` - standardowa funkcja modułu Image Processing pakietu MATLAB) i zilustruj ją (`pokaz obraz`, dobierz wartość parametru: `'typ'`, `'mapa'` - wartość domyślna).
- e. Dokonaj filtracji obrazu `pruski_mur` w dziedzinie częstotliwości wykorzystując właściwości transformaty Fouriera (TF) splotu:

$$TF(X * Y) = TF(X) \cdot TF(Y)$$

- f. Wyznacz obraz po filtracji (`ifft2` - standardowa funkcja modułu Image Processing pakietu MATLAB, `abs` - standardowa funkcja pakietu MATLAB) i zilustruj go (`pokaz obraz`, dobierz wartość parametru: `'typ'`, `'mapa'` - wartość domyślna).

Porównaj wyniki filtracji obrazu pruski mur w dziedzinie przestrzeni i w dziedzinie częstotliwości (z wykorzystaniem tego samego jądra filtru).

Sprawozdanie

Ćwiczenie 6 „Filtracja 2D”

L.p.	Imię i nazwisko	Grupa	Data
1			
2			
3			

Punkt ćwiczenia		Liczba punktów	Uzyskana liczba punktów	Uwagi prowadzącego
1	a) - d)	2		
	d) - e)	1		
	f) - g)	1		
2		2		
3		1		
4	a) - b)	1		
	c) - d)	1		
	e) - f)	1		