

02. Filtracja w domenie czasu

1 Filtracja liniowa

Filtracja liniowa w domenie czasu jest to spłot sygnału wraz z maską filtra. Rozmiar maski na ogół jest liczbą nieparzystą, gdzie wartość sumomnożenia przypisywana jest elementowi centralnemu. W zależności od sumy elementów maski wyróżniamy:

- filtrację dolnoprzepustową (uśredniającą), gdzie suma elementów maski równa się **1** oraz wszystkie elementy maski są nieujemne, np.:
LP1=[0.2 0.2 0.2 0.2 0.2]; LP2=[0.25 0.5 0.25]; sygnał Gaussa;
- filtrację górnoprzepustową, gdzie suma elementów maski równa się **0**, np.:
HP1=[-1 0 1]; czy HP2=[-1 2 -1];

W pakiecie MathWorks MatLAB spłot sygnałów realizowany jest poprzez funkcję

```
z = conv(x, y, 'shape');
```

gdzie parametr 'shape' przyjmuje następujące wartości:

- **full** - zwraca całą długość wektora spłotu (wartość domyślna):
 $\text{length}(z) = \text{length}(x) + \text{length}(y) - 1;$
- **same** - wektor spłotu ma rozmiar pierwszego argumentu:
 $\text{length}(z) = \text{length}(x);$
- **valid** - zwraca tylko tę część wektora spłotu, która została policzona bez wykorzystania wartości zerowych krótszego sygnału dodanych na brzegach:
 $\text{length}(z) = \max(\text{length}(x) - \max(0, \text{length}(y) - 1), 0);$

2 Filtracja nieliniowa

Istnieje również cała gama filtrów, których działanie zależy od np. uporządkowania obrazu. Są to np. filtry medianowe, maksymalizujące, minimalizujące czy entropii, które przypisują filtrowanemu punktowi odpowiednio medianę, maksimum, minimum i entropię z obszaru pokrytego maską. Większość z tych filtrów 2D umożliwia filtrację sygnałów 1D. W tym celu należy ustawić wartość szerokości (dla sygnałów pionowych) lub wysokości (dla sygnałów poziomych) maski na 1 lub stworzyć pionowy/poziomy wektor złożony z wartości 0/1. W pakiecie MatLAB zostały zdefiniowane następujące metody filtracji:

- **medfilt1(sygnał, rozmiar)**; - filtracja medianowa 1D;
- **medfilt2(obraz, [rozmiar], 'brzeg')**; - filtracja medianowa 2D ;
- **ordfilt2(obraz, nr, zasięg)**; - zwraca wartość znajdującą się na pozycji **nr** w uporządkowanym wektorze (1 - minimum, wielkość maski - maksimum);
- **rangefilt(obraz, zasięg)** - zwraca różnicę pomiędzy maksimum i minimum intensywności dla pikseli otoczenia zdefiniowanych w tablicy *zasięg*, np. ones(3);
- **entropyfilt(obraz, zasięg)**; - zwraca wartość entropii *E* w blokach o zadanym rozmiarze (domyślnie 9x9 pikseli), obliczoną wg wzoru:

$$E = - \sum_{n=0}^N \log p(n) \cdot p(n) \quad (1)$$

N - ilość poziomów intensywności

p(n) - prawdopodobieństwo wystąpienia piksela o intensywności *n*

- `stdfilt(obraz, zasięg)` - dla każdego piksela zwraca wartość odchylenia standardowego z otoczenia.

3 Filtracja adaptatywna (Wienera)

Osobną kategorią filtrów nieliniowych są filtry adaptacyjne. Są to filtry, których charakterystyka zmienia się w zależności od analizowanego obszaru. W celu usuwania szumu działają one dwuetapowo:

1. Dla każdego punktu i jego otoczenia obliczamy wartość parametru, który kwalifikuje dany punkt jako należący lub nie do krawędzi. W przypadku filtru Wienera jest to wariancja w obszarze obrazu pokrytym maską.
2. jeżeli dany punkt został zakwalifikowany jako nie należący do krawędzi, zostaje on poddany silnemu uśrednieniu. W przeciwnym wypadku jego wartość pozostaje bez zmian lub poddany zostaje uśrednieniu o niewielkiej mocy.

Do filtrów tego typu zaliczamy filtr Wienera: `wiener2(obraz, [rozmiar])`.

4 Filtr Savitzky-Golay

Metoda Savitzky-Golay polega na lokalnym wygładzaniu sygnału przy użyciu wielomianów określonego stopnia. Do filtracji służy polecenie `sgolayfilt(sygnał, stopień, rozmiar)`, gdzie rozmiar musi być nieparzystą, dodatnią liczbą całkowitą.

5 Zadania

1. Ściągnij i wczytaj sygnał `2022_corr_01.txt`. Dokonaj na nim filtracji uśredniającej w oknach o rozmiarze: 3, 5, 9 próbek oraz 1%, 2%, i 5% szerokości sygnału (zaokrąglając do najbliższej liczby nieparzystej).
2. Stwórz uśredniającą maskę Gaussa o zadanej długości i odchyleniu. Zbadaj wpływ obu tych parametrów na jakość sygnału.
3. Stwórz sygnał harmoniczny $f(t)$ ($A_1 = 5$, $f_1 = 2Hz$) dla czasu $T \in < 0, 10 > s$, $F_S = 100Hz$. Następnie dodaj do niego (osobno):
 - (a) sygnał harmoniczny ($A_2 = 0.5$, $f_2 = 60Hz$)
 - (b) szum gaussowski o odchyleniu ($sr = 0$, $\sigma = 0.25$)
 - (c) szum impulsowy ($amp = 2$, $Prob. = 5\%$);Dokonaj odsumienia sygnału $f(t)$. Przetestuj różne rodzaje filtracji odsumiającej.
4. Policzyć pierwszą i drugą pochodną numeryczną dla sygnału `corr_1.txt`