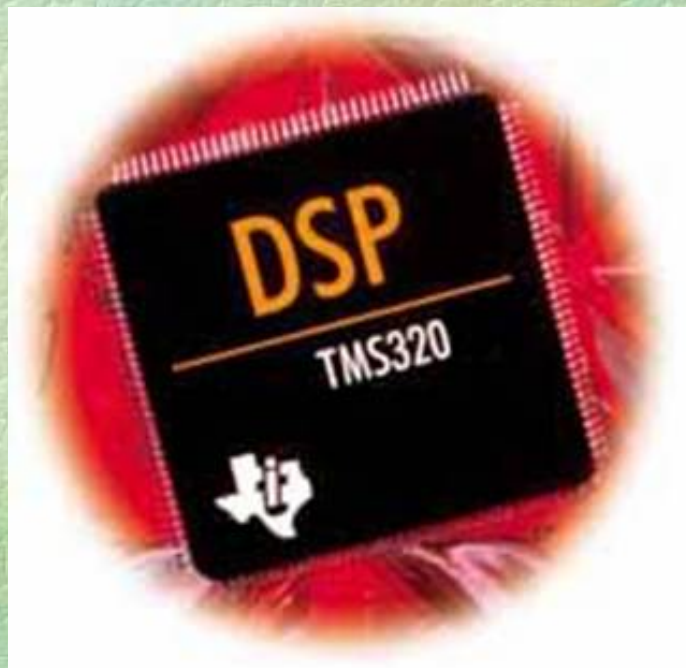


Podstawy teorii sygnałów

Splot i korelacja



wykład №8

Splot i korelacja są podstawowymi pojęciami przetwarzania sygnałów.

Splot jest bazową operacją dla filtracji cyfrowej, pozwalającej na zwiększenie stosunku mocy sygnału do mocy zakłóceń.

Korelacja pozwala na porównanie sygnału z przebiegiem odniesienia (wzorcem). Redukuje wpływ składowych losowych oraz pomaga wykryć składowe sygnału podobne do wzorca.

SPLIT KOLOWY

$$y_m = \sum_{n=0}^{N-1} x_n h_{m-n}, \quad m = \overline{0, N-1}$$



$$\mathbf{Y} = \mathbf{H}_N \mathbf{X}$$

$$\mathbf{Y} = [y_0, y_1, \dots, y_{N-1}]^T$$

$$\mathbf{X} = [x_0, x_1, \dots, x_{N-1}]^T$$

$$\mathbf{H} = [h_0, h_1, \dots, h_{N-1}]^T$$

$$\mathbf{H}_N = \begin{bmatrix} h_0 & h_{N-1} & h_{N-2} & \dots & h_1 \\ h_1 & h_0 & h_{N-1} & \dots & h_2 \\ h_2 & h_1 & h_0 & \dots & h_3 \\ \dots & \dots & \dots & \dots & \dots \\ h_{N-1} & h_{N-2} & h_{N-3} & \dots & h_0 \end{bmatrix}$$

Przykład wyznaczenia splotu kolowego

X=

1	2	4	8
---	---	---	---

H=

2	3	4	5
---	---	---	---

1	2	4	8
2	5	4	3

$$y(0) = 1 \cdot 2 + 2 \cdot 5 + 4 \cdot 4 + 8 \cdot 3 = 52$$

1	2	4	8
3	2	5	4

$$y(1) = 1 \cdot 3 + 2 \cdot 2 + 4 \cdot 5 + 8 \cdot 4 = 59$$

Y=

52	59	58	41
----	----	----	----

1	2	4	8
4	3	2	5

$$y(2) = 1 \cdot 4 + 2 \cdot 3 + 4 \cdot 2 + 8 \cdot 5 = 58$$

1	2	4	8
5	4	3	2

$$y(3) = 1 \cdot 5 + 2 \cdot 4 + 4 \cdot 3 + 8 \cdot 2 = 41$$

KORELACJA KOŁOWA

$$y_m = \sum_{n=0}^{N-1} h_{m+n} \cdot x_n, \quad m = \overline{0, N-1}$$

$$h_{m+n} = h_n, \quad \text{dla } n+m \geq N$$



$$\mathbf{Y} = \mathbf{G}_N \mathbf{X}$$

$$\mathbf{Y} = [y_0, y_1, \dots, y_{N-1}]^T$$

$$\mathbf{X} = [x_0, x_1, \dots, x_{N-1}]^T$$

$$\mathbf{H} = [h_0, h_1, \dots, h_{N-1}]^T$$

$$\mathbf{G}_N = \begin{bmatrix} g_0 & g_1 & g_2 & \cdots & g_{N-1} \\ g_{N-1} & g_0 & g_1 & \cdots & g_{N-2} \\ g_{N-2} & g_{N-1} & g_0 & \cdots & g_{N-3} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ g_1 & g_2 & g_3 & \cdots & g_0 \end{bmatrix}$$

Przykład wyznaczenia korelacji kolowejj

$X =$

1	2	4	8
---	---	---	---

$H =$

2	3	4	5
---	---	---	---

1	2	4	8
2	3	4	5

1	2	4	8
5	2	3	4

1	2	4	8
4	5	2	3

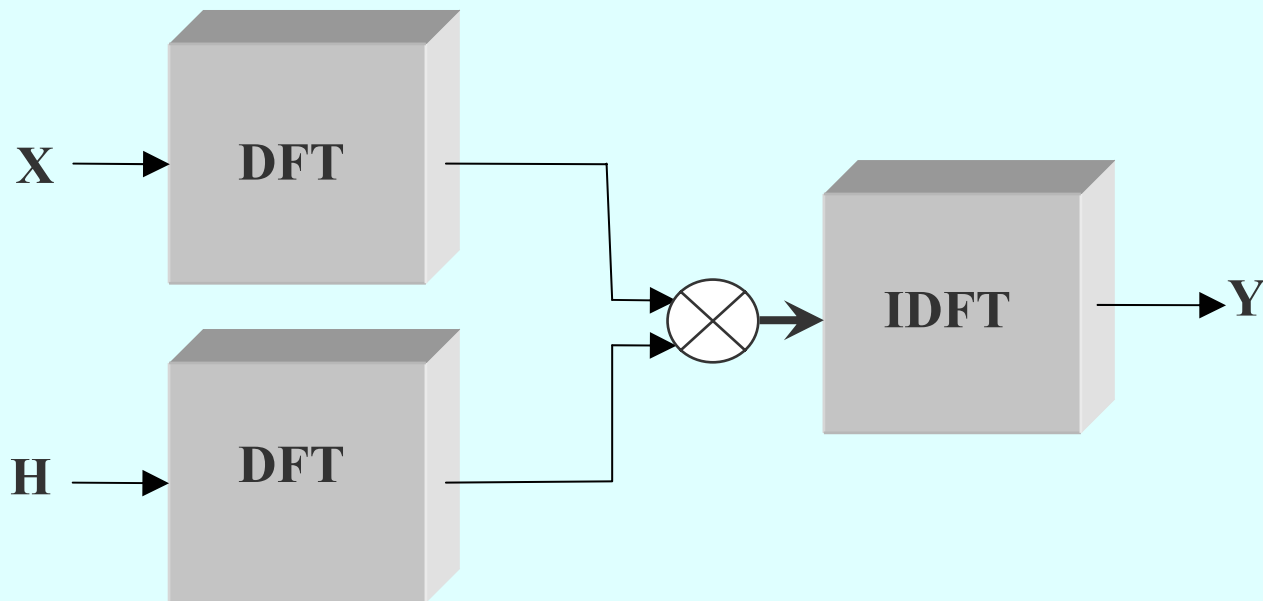
1	2	4	8
3	4	5	2

$Y =$

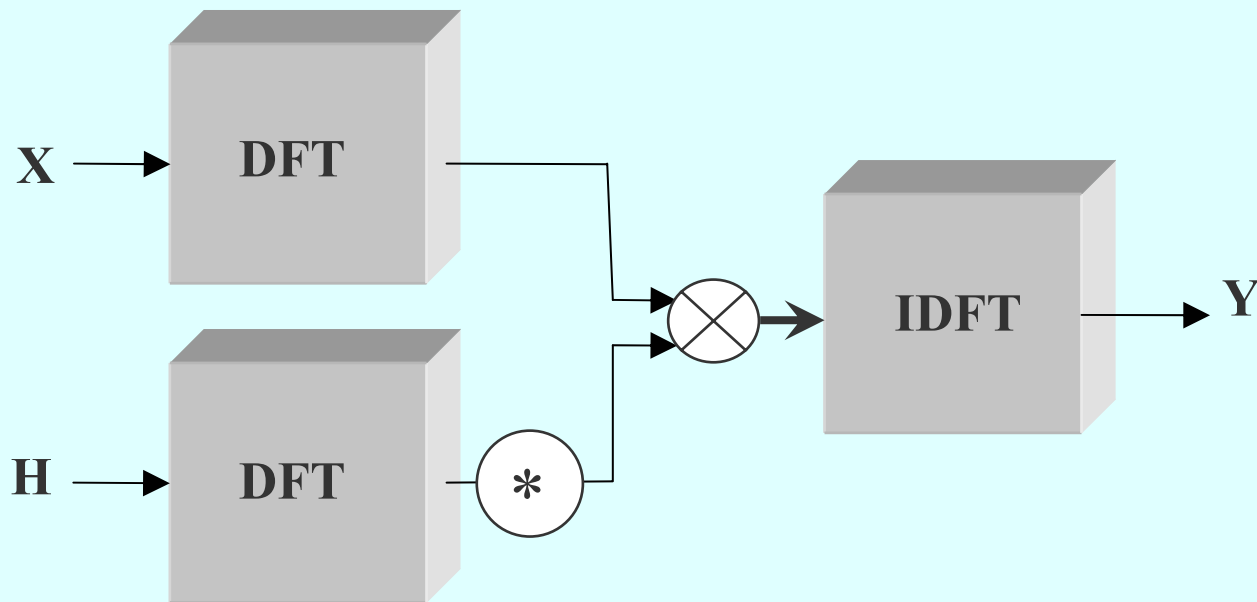
64	53	46	47
----	----	----	----

Obliczenie splotu kolowego za pomocą DFT

Splot w dziedzinie czasu może być zastąpiony przez mnożenie w dziedzinie częstotliwości!



Obliczenie korelacji kolowej za pomocą DFT



SPLIT LINIOWY

Splot liniowy dwóch ciągów skończonych jest zdefiniowany następująco:

$$\mathbf{X} = [x_0, x_1, \dots, x_{N-1}]^T$$

$$\mathbf{H} = [h_0, h_1, \dots, h_{M-1}]^T$$

$$y_l = \sum_{n=0}^l h_{l-n} \cdot x_n$$



$$h_{l-n} = 0$$

$$l - n < 0$$

$$l = 0, \dots, N + M - 2$$

$$\mathbf{Y} = [y_0, y_1, \dots, y_{N+M-2}]^T$$



$$(N + M - 1) \times 1$$

Splot sygnału z odpowiedzią impulsową można przedstawić opisowo w sposób następujący.

Ciąg odpowiedzi impulsowej należy odwrócić **“tył na przód”** i stopniowo przesuwając nad ciągiem sygnału wejściowego. W każdym położeniu wyznacza się iloczyny tych elementów obu ciągów, które znalazły się jeden nad drugim.

Zsumowanie tych iloczynów daje wynik splotu dla pojedynczej wartości indeksu sygnału wyjściowego.

Przykład wyznaczenia splotu liniowego

$$\mathbf{X} = \begin{bmatrix} 1 & 2 & 4 & 8 \end{bmatrix}$$

			1	2	4	8
5	4	3	2			

$$y(0) = 1 \cdot 2 = 2$$

			1	2	4	8
5	4	3	2			

$$y(1) = 1 \cdot 3 + 2 \cdot 2 = 7$$

			1	2	4	8
5	4	3	2			

$$y(2) = 1 \cdot 4 + 2 \cdot 3 + 4 \cdot 2 = 18$$

			1	2	4	8
5	4	3	2			

$$y(3) = 1 \cdot 5 + 2 \cdot 4 + 8 \cdot 2 = 41$$

$$\mathbf{H} = \begin{bmatrix} 2 & 3 & 4 & 5 \end{bmatrix}$$

1	2	4	8	
	5	4	3	2

$$y(4) = 2 \cdot 5 + 4 \cdot 4 + 8 \cdot 3 = 50$$

1	2	4	8		
		5	4	3	2

$$y(5) = 4 \cdot 5 + 8 \cdot 4 = 52$$

1	2	4	8			
			5	4	3	2

$$y(6) = 8 \cdot 5 = 40$$

$$\mathbf{Y} = \begin{bmatrix} 2 & 7 & 18 & 41 & 50 & 52 & 40 \end{bmatrix}$$

$$\mathbf{Y}_{(2N-1) \times 1} = \mathbf{H}_{(2N-1) \times N} \cdot \mathbf{X}_{N \times 1}$$

$$\mathbf{Y}_{(2N-1) \times 1} = [y_0, y_1, \dots, y_{2N-2}]^T$$

$$\mathbf{X} = [x_0, x_1, \dots, x_{N-1}]^T$$

$$\mathbf{H}_{(2N-1) \times N} = \begin{bmatrix} h_0 & & & & \\ h_1 & h_2 & & & \\ h_2 & h_1 & h_0 & & \\ \dots & \dots & \dots & \dots & \dots \\ h_{N-1} & h_{N-2} & h_{N-3} & \dots & h_0 \\ & h_{N-1} & h_{N-2} & \dots & h_1 \\ \dots & \dots & \dots & \dots & \dots \\ & & & h_{N-1} & h_{N-2} \\ & & & & h_{N-1} \end{bmatrix}$$

KORELACJA LINIOWA

$$\mathbf{Y}_{(2N-1) \times 1} = \mathbf{G}_{(2N-1) \times N} \cdot \mathbf{X}_{N \times 1}$$

$$\mathbf{Y}_{(2N-1) \times 1} = [y_0, y_1, \dots, y_{2N-2}]^T$$

$$\mathbf{X} = [x_0, x_1, \dots, x_{N-1}]^T$$

$$\mathbf{G}_{(2N-1) \times N} = \begin{bmatrix} g_{N-1} & & & & \\ g_{N-2} & g_{N-1} & & & \\ g_{N-3} & g_{N-2} & g_{N-1} & & \\ \dots & \dots & \dots & \dots & \dots \\ g_0 & g_1 & g_2 & \dots & g_{N-1} \\ & g_0 & g_1 & \dots & g_{N-2} \\ \dots & \dots & \dots & \dots & \dots \\ & & & g_0 & g_1 \\ & & & & g_0 \end{bmatrix}$$

Korelację sygnału z przebiegiem odniesienia można przedstawić opisowo w sposób następujący.

Ciąg przebiegu odniesienia uporządkowany według kolejności naturalnej należy rozmieścić pod sygnałem i stopniowo przesuwając nad ciągiem sygnału wejściowego.

W każdym położeniu wyznacza się iloczyny tych elementów obu ciągów, które znalazły się jeden nad drugim.

Zsumowanie tych iloczynów daje wynik splotu dla pojedynczej wartości indeksu sygnału wyjściowego.

Przykład wyznaczenia korelacji liniowej

$X =$

1	2	4	8
---	---	---	---

$G =$

2	3	4	5
---	---	---	---

			1	2	4	8
2	3	4	5			

1	2	4	8			
	2	3	4	5		

			1	2	4	8
2	3	4	5			

1	2	4	8			
		2	3	4	5	

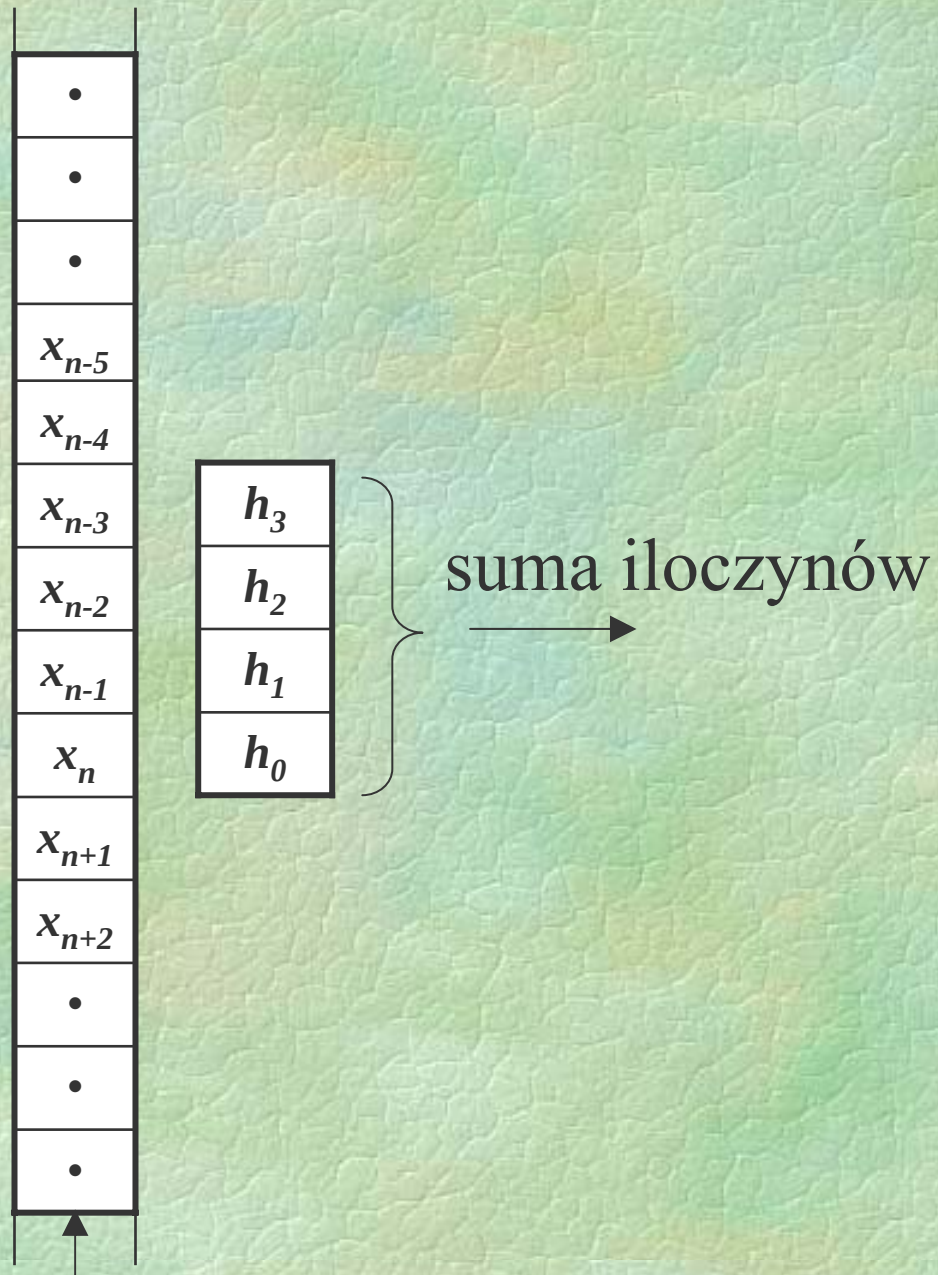
			1	2	4	8
2	3	4	5			

1	2	4	8			
			2	3	4	5

1	2	4	8
2	3	4	5

$Y =$

5	14	31	64	48	32	16
---	----	----	----	----	----	----



Wyznaczenie splotu liniowego za pomocą splotu kołowego

Algorytm wyznaczenia splotu liniowego za pomocą splotu kołowego

1. Sekwencje próbek sygnału oraz odpowiedź impulsowa filtru FIR uzupełniają się zerami.

Długość sekwencji po uzupełnieniu jest równa **sumie** sekwencji pierwotnych pomniejszonej o 1, czyli $N+M-1$.

2. Nad poszerzonymi w taki sposób sekwencjami wykonyje się operacja splotu kołowego za pomocą metody klasycznej wcześniej rozpatrzonej.

Przykład wyznaczenia splotu liniowego za pomocą splotu kołowego

$$X = \begin{bmatrix} 1 & 2 & 4 & 8 & 0 & 0 & 0 \end{bmatrix}$$

1	2	4	8	0	0	0
2	0	0	0	5	4	3

$$y(0) = 1 \cdot 2 = 2$$

1	2	4	8	0	0	0
3	2	0	0	0	5	4

$$y(1) = 1 \cdot 3 + 2 \cdot 2 = 7$$

1	2	4	8	0	0	0
4	3	2	0	0	0	5

$$y(2) = 1 \cdot 4 + 2 \cdot 3 + 4 \cdot 2 = 18$$

1	2	4	8	0	0	0
5	4	3	2	0	0	0

$$y(3) = 1 \cdot 5 + 2 \cdot 4 + 4 \cdot 3 + 8 \cdot 2 = 41$$

$$H = \begin{bmatrix} 2 & 3 & 4 & 5 & 0 & 0 & 0 \end{bmatrix}$$

1	2	4	8	0	0	0
0	5	4	3	2	0	0

$$y(4) = 2 \cdot 5 + 4 \cdot 4 + 8 \cdot 3 = 50$$

1	2	4	8	0	0	0
0	0	5	4	3	2	0

$$y(5) = 4 \cdot 5 + 8 \cdot 4 = 52$$

1	2	4	8	0	0	0
0	0	0	5	4	3	2

$$y(6) = 8 \cdot 5 = 40$$

$$Y = \begin{bmatrix} 2 & 7 & 18 & 41 & 50 & 52 & 40 \end{bmatrix}$$

**Realizacja splotu
 liniowego za pomocą
splotu kolowego
z wykorzystaniem FFT

(Filtracja długich ciągów)**

W dotychczasowych rozważaniach przyjmowano, że przez filtr **FIR** rozumie się jego odpowiedź impulsową, bezpośrednio stosowaną do wyznaczania splotu liniowego.

Charakterystyka widmowa filtru stanowiła jedynie wstępne założenie projektowe.

W celu uproszczenia realizacji splotu liniowego dla długich sekwencji danych zostały zaproponowane metody wykorzystujące segmentację danych. Główne z nich to „**Overlap-Save**” oraz „**Overlap-Add**”.

Należy wspomnieć, że te metody wykorzystywane są w większości przypadków dla bardzo długich sekwencji danych, lub w szczególności dla nieskończonych sekwencji danych.

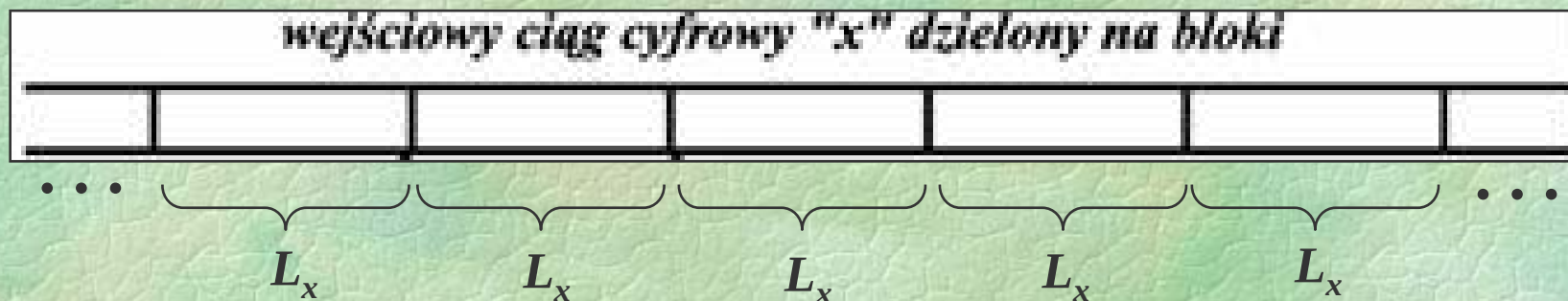
W literaturze metody te były implementowane przede wszystkim przy realizacji splotu za pomocą transformaty FFT.

Idea tych metod polega na podziale danych wejściowych na mniejsze segmenty, a następnie rozważenie procesu obliczania splotu dla mniejszej liczby elementów (prostsze w budowie algorytmy).

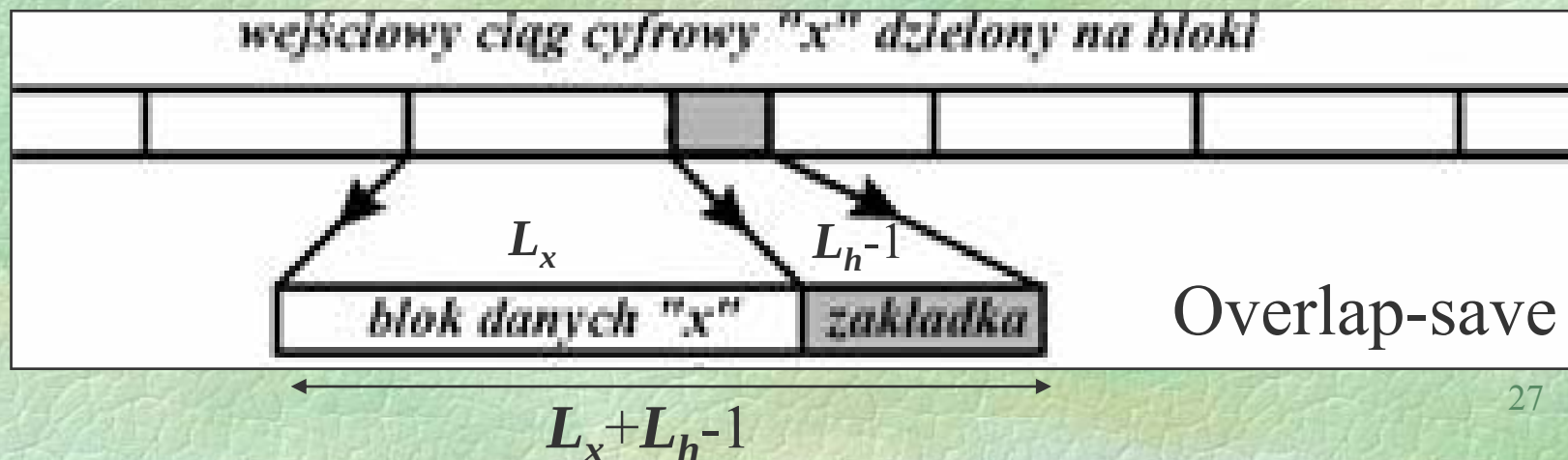
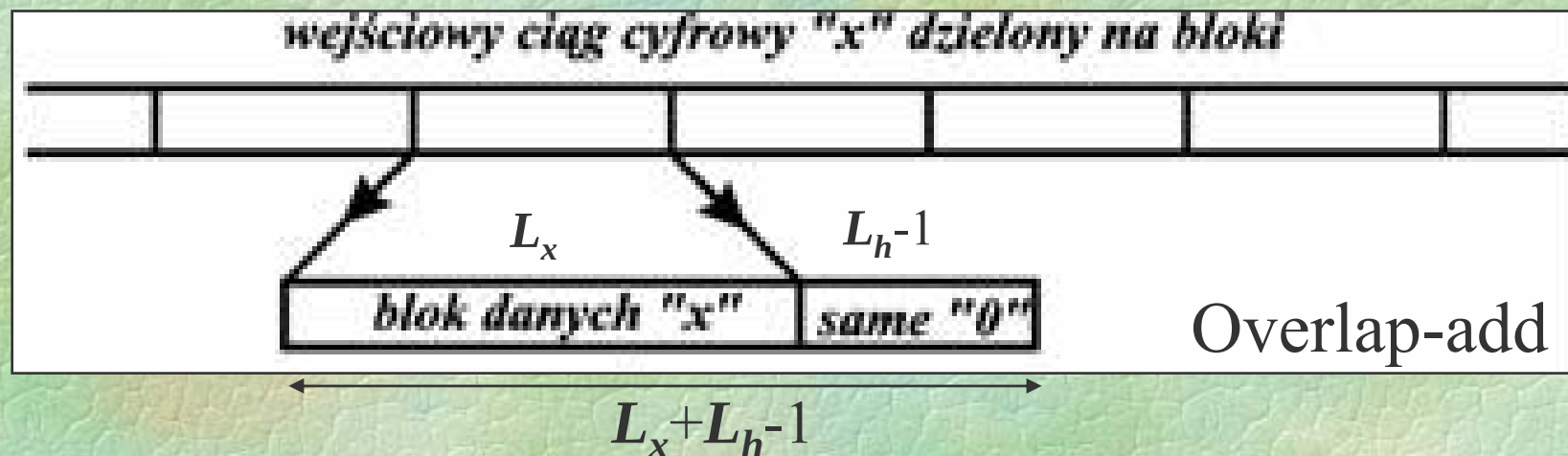
Następnie należy odpowiednio złożyć wyniki obliczeń w celu otrzymania końcowego rezultatu.

W obu metodach ciąg wejściowy dzielony jest w miarę napływania danych na bieżąco na bloki o długości L_x .

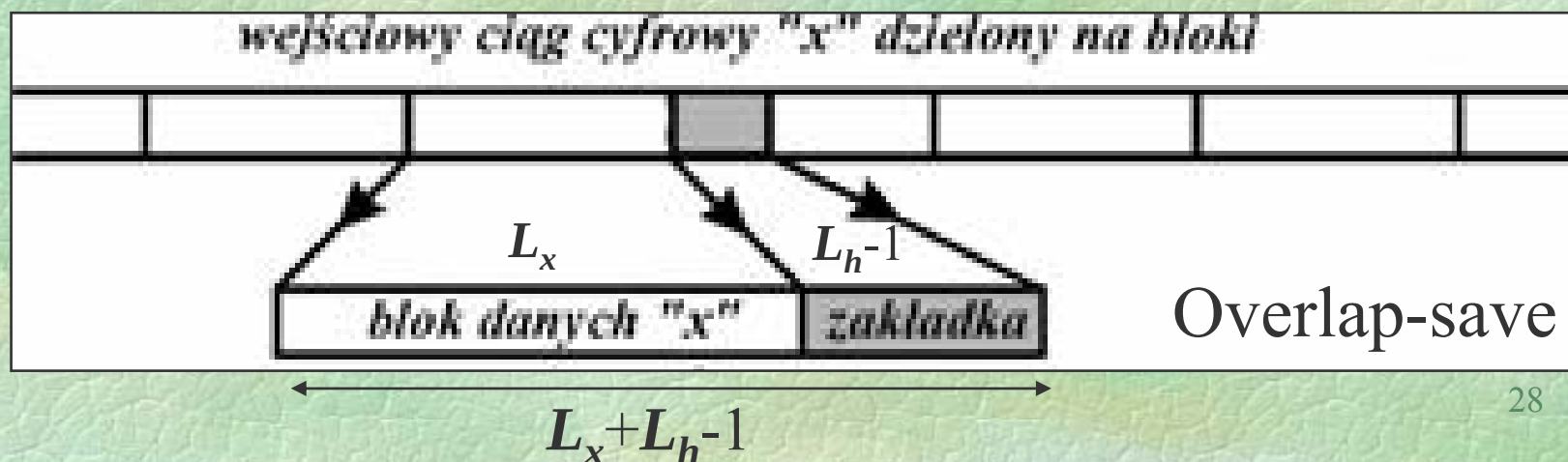
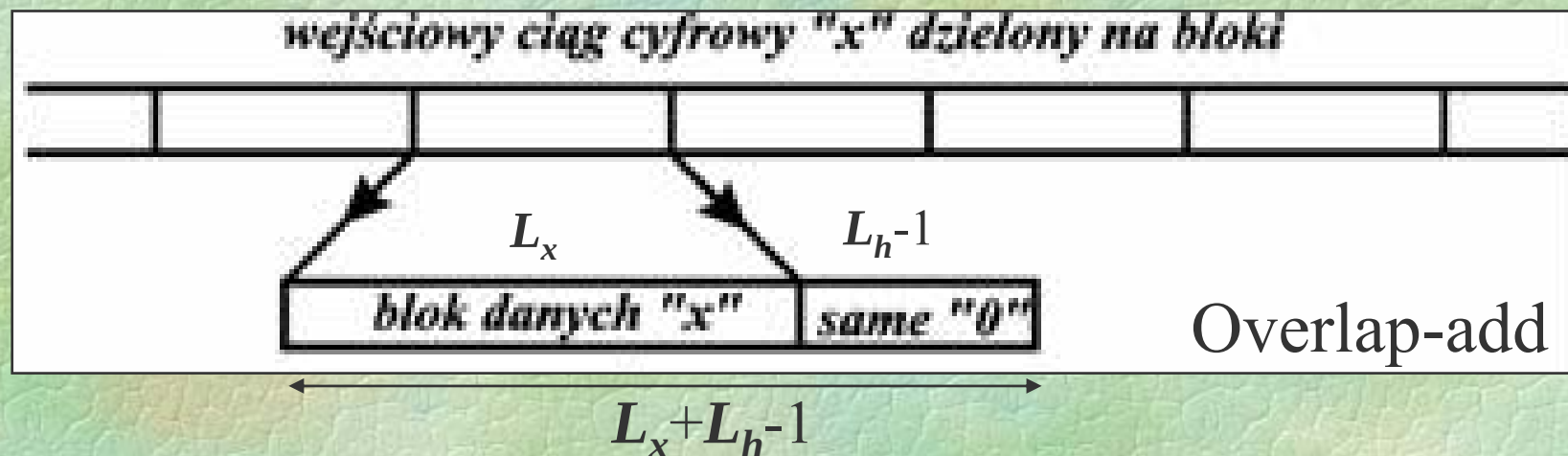
$$\mathbf{X} = [x_0, x_1, \dots, x_{N-1}]^T$$



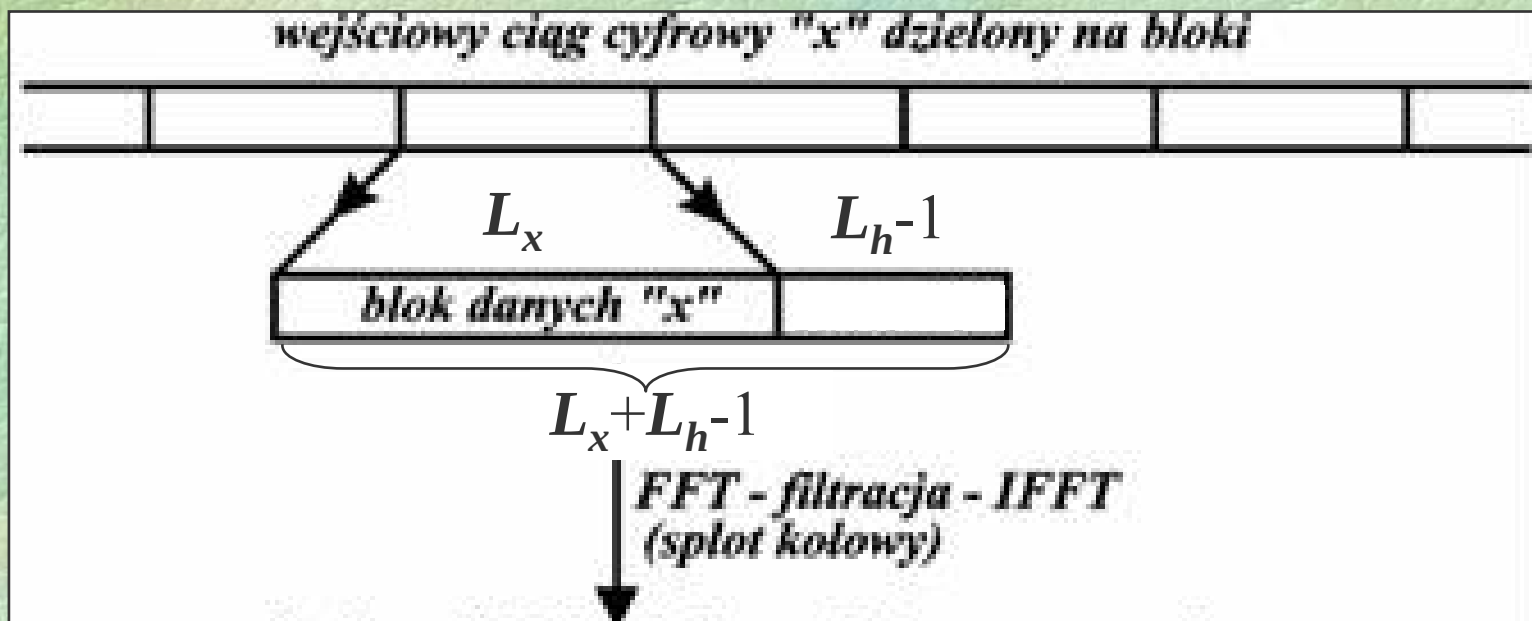
Każdy blok, w chwili jego filtracji, jest uzupełniony zerami (metoda **overlap-add**) lub początkiem kolejnego bloku (metoda **overlap-save**).



Długość tego uzupełnienia jest równa długości odpowiedzi impulsowej filtru *FIR* pomniejszonej o 1, czyli L_h-1 .



W kolejnym kroku blok o łącznej długości $L_x + L_h - 1$ poddawany jest *FFT*, po czym przeprowadzana jest filtracja za pomocą mnożenia przez transformatę *DFT* odpowiedzi impulsowej filtru i wynik filtracji poddawany jest odwrotnej transformacji - czyli *IFFT*.



Filtracja taka odpowiada splotowi kołowemu bloku danych o długości $L_x + L_h - 1$.

W celu sprowadzenia wyniku do postaci odpowiadającej fragmentowi splotu liniowego o długości L_x stosuje się następujące zabiegi:

w metodzie „**overlap-add**”:

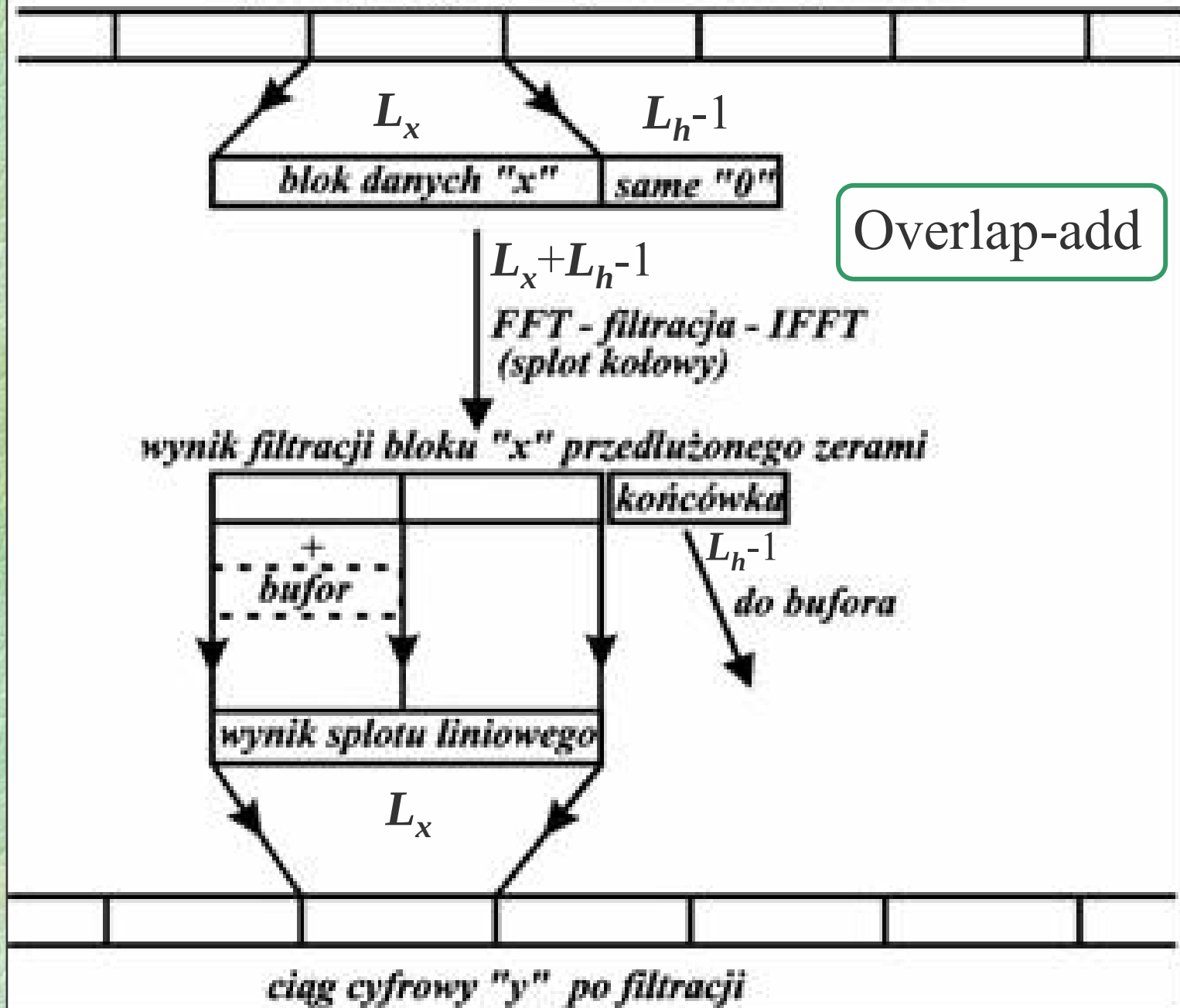
wynik splotu kołowego, o długości $L_x + L_h - 1$, dzielony jest na dwie części - pierwszą, o długości L_x i drugą, o długości $L_h - 1$.

Do kolejnych początkowych $L_x - 1$ elementów pierwszej części dodawane są kolejne elementy zapamiętane w buforze, znajdujące się tam w wyniku przetwarzania poprzedniego bloku.

Z kolei druga część - właśnie o długości $L_h - 1$ - jest wpisywana jako nowa zawartość bufora, do wykorzystania przy filtracji kolejnego bloku.

Cała, tak zmodyfikowana, pierwsza część o długości L_x stanowi odpowiedni fragment splotu liniowego.

wejściowy ciąg cyfrowy "x" dzielony na bloki



w metodzie „*overlap-save*”:

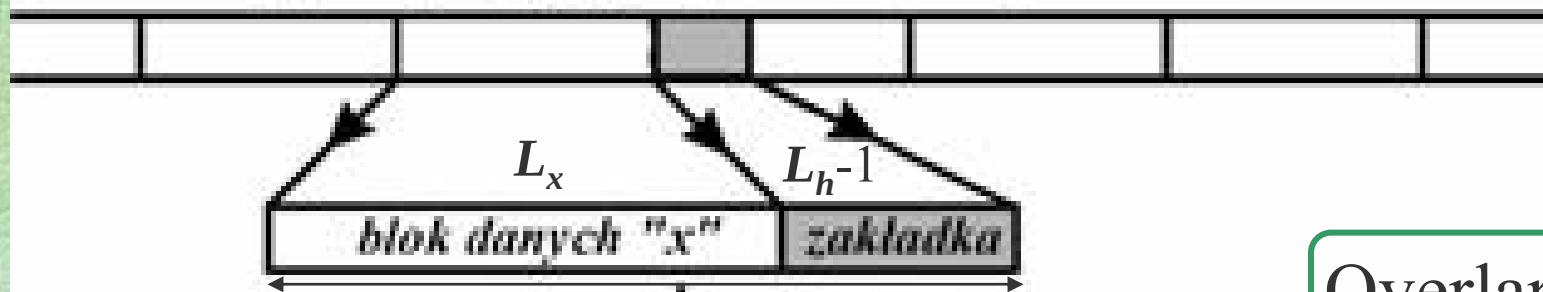


Wynik splotu kołowego, o długości $L_x + L_h - 1$, jest dzielony na dwie części.



W tym przypadku pierwsza część ma długość $L_h - 1$ i jest po prostu odrzucana, natomiast jako odpowiedni fragment splotu liniowego pozostawia się część drugą, o długości L_x .

wejściowy ciąg cyfrowy "x" dzielony na bloki

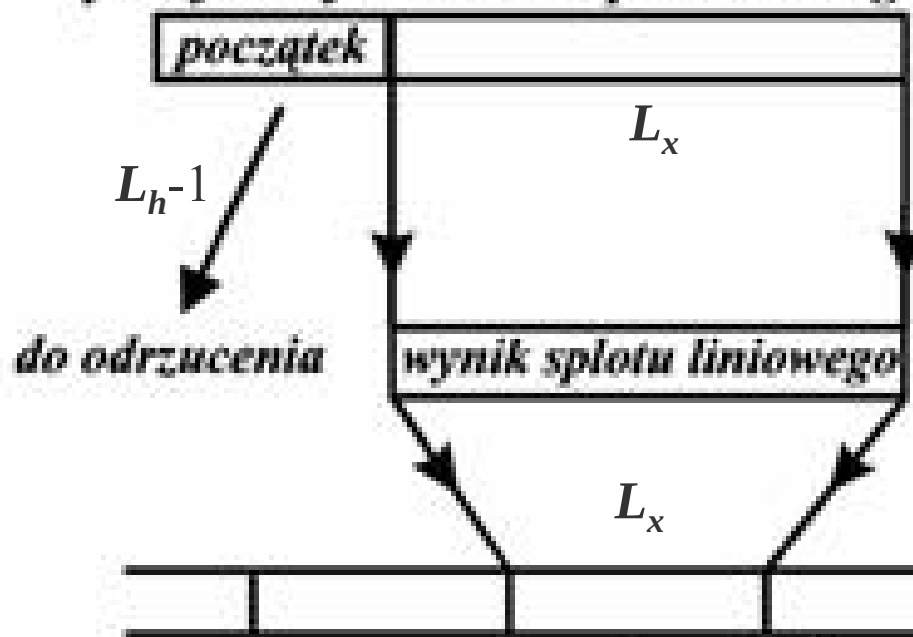


Overlap-save

$$L_x + L_h - 1$$

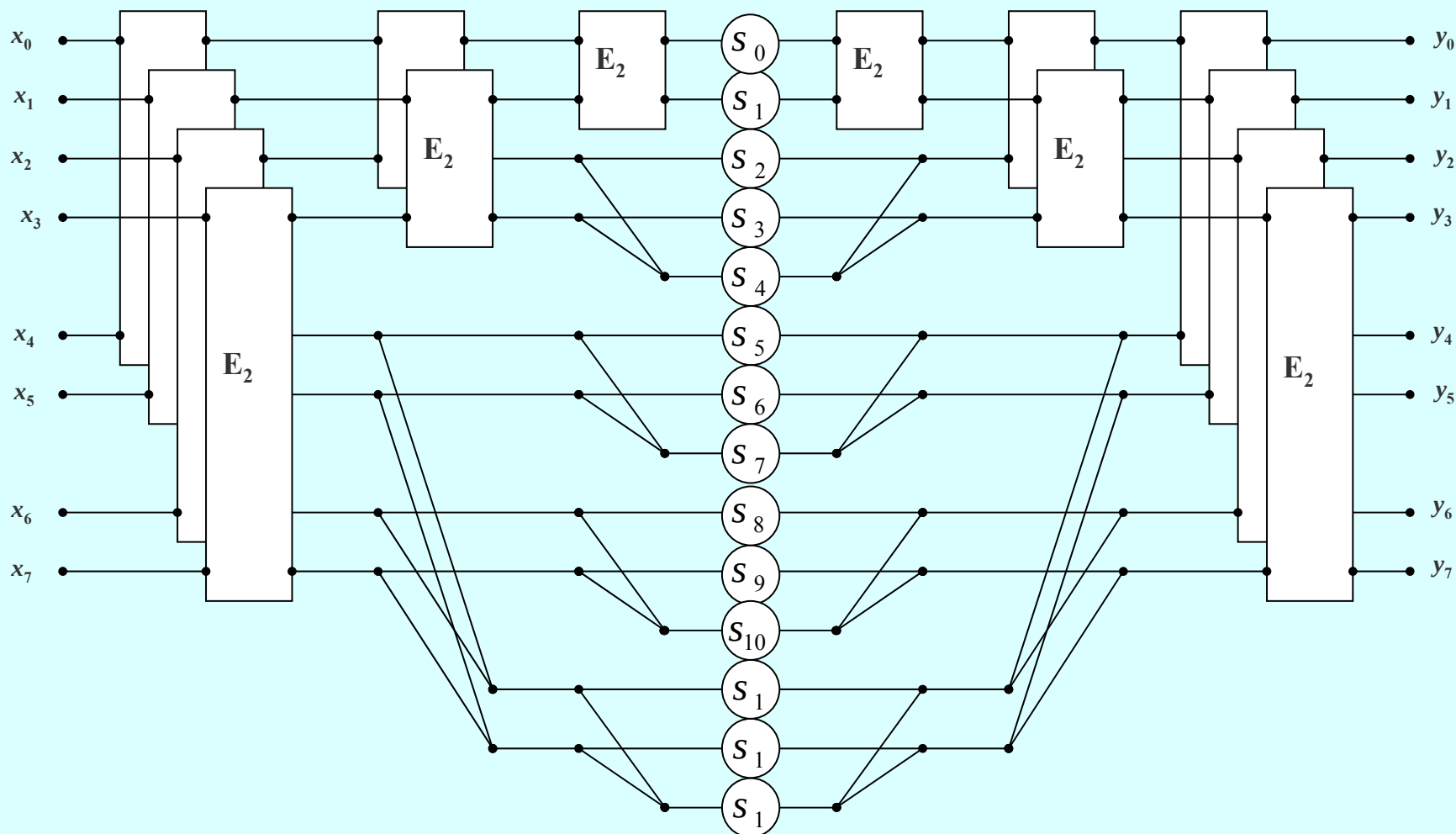
FFT - filtracja - IFFT
(splot kołowy)

wynik filtracji bloku "x" przedłużonego "zakładką"



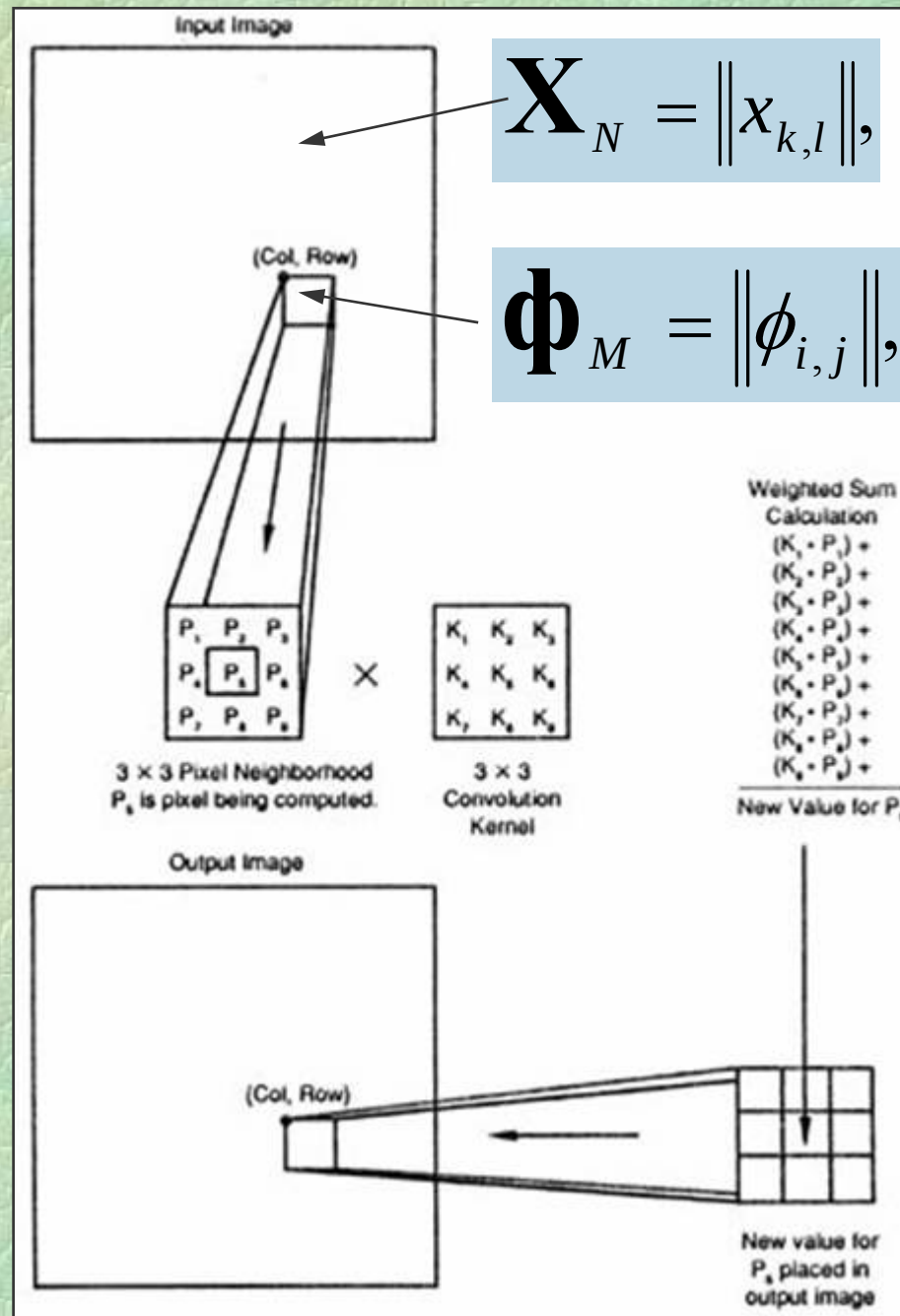
ciąg cyfrowy "y" po filtracji

Splot cykliczny rzędu $N=8$



14 operacji mnożenia, 46 operacji dodawania

SPLOT DWUWYMIAROWY



$$\mathbf{X}_N = \left\| x_{k,l} \right\|,$$

$$\boldsymbol{\Phi}_M = \left\| \phi_{i,j} \right\|,$$

$$k, l = \overline{0, N-1}$$

$$i, j = \overline{0, M-1}, M < N;$$

$$y_{k,l} = \sum_i \sum_j x_{k+l, l+j} \cdot \phi_{i,j}$$

SPATIAL CONVOLUTION

Input Digital Image

9	5	7
4	6	2
6	7	9

Target Pixel

	12	

Output Grey Value
of Target Pixel is
Sum of Products

Convolution Mask

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

Mask Overlay on
Target Pixel

0 x ₉	-1 x ₅	0 x ₇
-1 x ₄	5 x ₆	-1 x ₂
0 x ₆	-1 x ₇	0 x ₉

Multiply Mask Times
Pixel Intensities

0	-5	0
-4	30	-2
0	-7	0

$$\begin{array}{r} -5 \\ -2 \\ -7 \\ -4 \\ +30 \\ \hline 12 \end{array}$$

Low-Pass Filtering

2	4	5	4	2
4	9	12	9	4
5	12	15	12	5
4	9	12	9	4
2	4	5	4	2

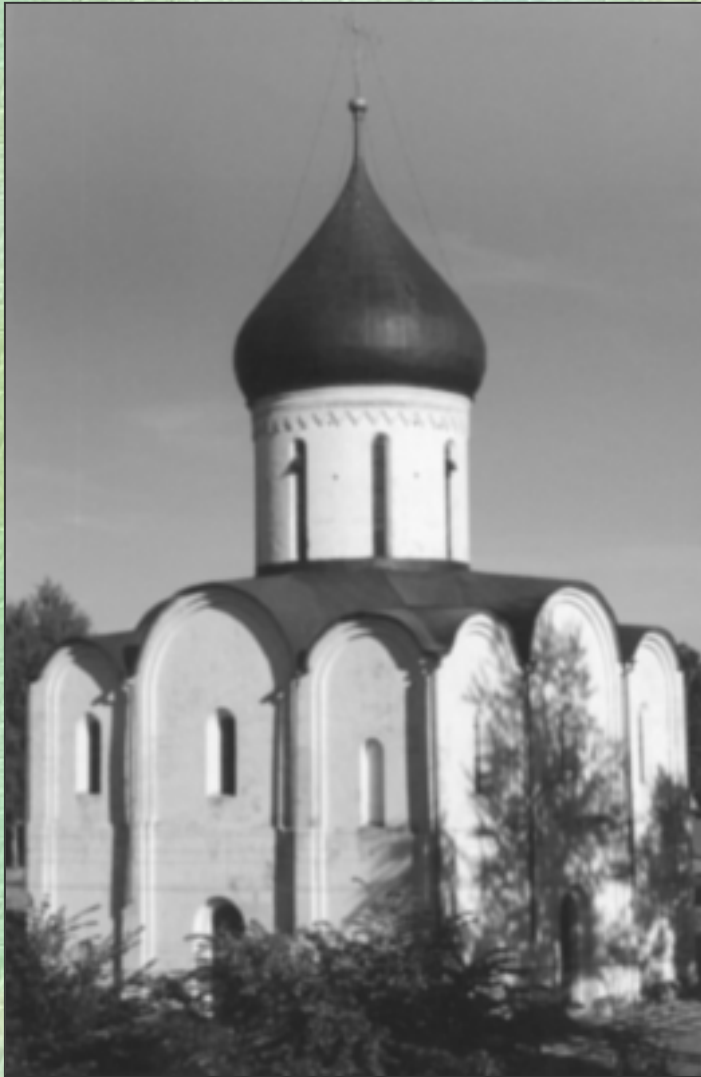
Gaussian mask

High-Pass Filtering

0	-1	0	-1	-1	-1
-1	4	-1	-1	8	-1
0	-1	0	-1	-1	-1

Laplacian masks

Rozfokusowanie (blur)



$$\frac{1}{6} \begin{bmatrix} & & 1 \\ 1 & 2 & 1 \\ & 1 & \end{bmatrix}$$

Rozfokusowanie (Cz.d)



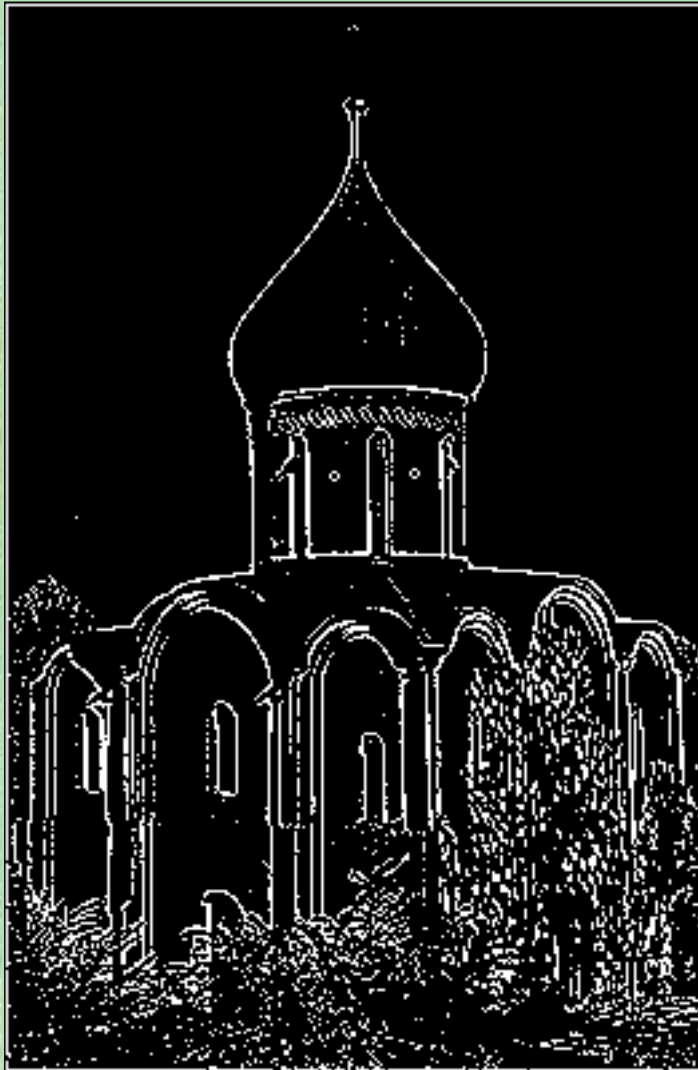
$$\frac{1}{74} \begin{bmatrix} 1 & 2 & 3 & 2 & 1 \\ 2 & 4 & 5 & 4 & 2 \\ 3 & 5 & 6 & 5 & 3 \\ 2 & 4 & 5 & 4 & 2 \\ 1 & 2 & 3 & 2 & 1 \end{bmatrix}$$

Podniesienie ostrości



$$\begin{bmatrix} & -1 & \\ -1 & 5 & -1 \\ & -1 & \end{bmatrix}$$

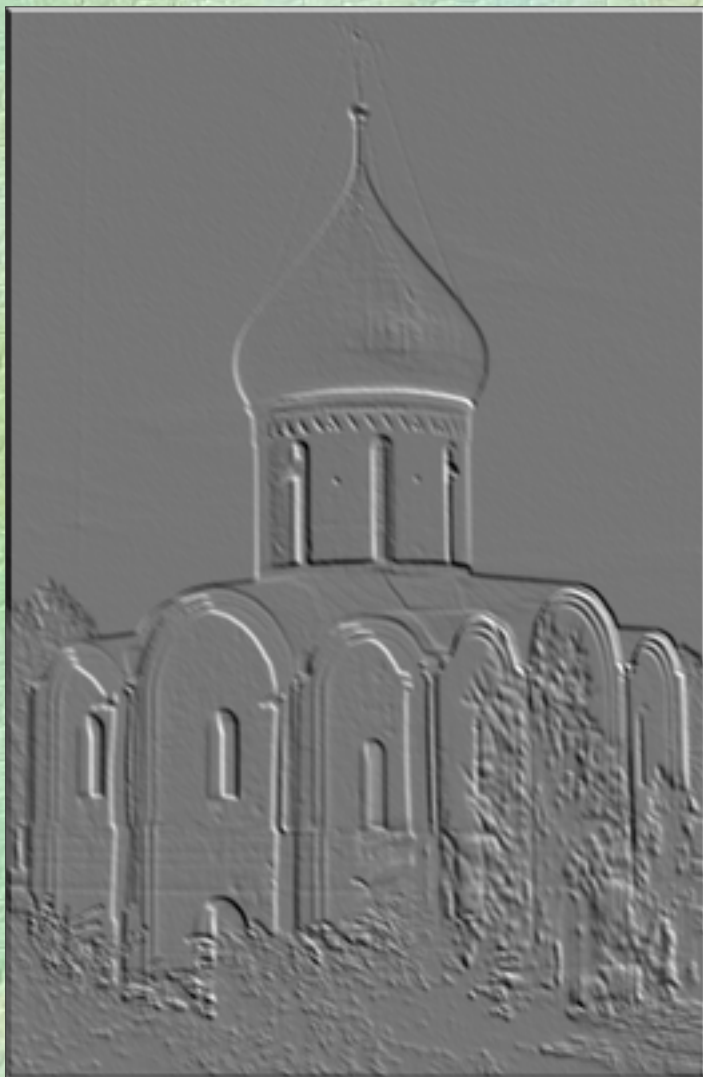
Wydzielenie krawędzi



$$\begin{bmatrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix}$$

+ próg

Wytlaczanie



$$\begin{bmatrix} & 1 & \\ -1 & 0 & 1 \\ & -1 & \end{bmatrix}$$

+ przesunięcie
jaskrawości



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

Prewitt masks

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

Sobel masks

0	-1	0	-1	-1	-1
-1	4	-1	-1	8	-1
0	-1	0	-1	-1	-1

Laplacian masks



Dziękuję za uwagę

