MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE

NATIONAL TECHNICAL UNIVERSITY
"KHARKIV POLYTECHNICAL INSTITUTE"

Department of Computer Engineering and Programming

O.A. Povoroznyuk
A. E. Filatova

Methodological instructions for practical work on the course

SIGNAL AND IMAGE PROCESSING

Approved by the editorial and publishing
department university,
protocol No 1. dated   15.02.2024

Kharkiv – 2024

УДК 004.3(075)
ББК 32.973-02я7
P 42

Reviewer: G.. K. Kozhevnikov, Candidate of technical sciences, professor of the department "Multimedia and Internet technologies and systems" NTU "KhPI"

O.A. Povoroznyuk
P42 Methodological instructions for practical work on the course “Signal and Image Processing” for students of speciality 123 "Computer Engineering”/ O.A. Povoroznyuk, A. E. Filatova . – Kharkiv: NTU “KhPI”, 2024 – 94 p.

This edition provides methodological instructions for practical works 1-9, which include mathematical basics of signal processing (Fourier transforms, convolutions, z-transform) and image processing (two-dimensional Fourier transform, element-by-element transforms, digital filtering and wavelet transforms). Theoretical information and calculations are accompanied by examples.

Intended for students of all undergraduate majors in Computer Engineering 123.

Fig. 15 Tab. 13. Lit. 10 names.

УДК 004.3(075)
ББК 32.973-02я7

CONTENT

INTRODUCTION

The intensive development of digital technologies contributed to them implementation in such technological fields as digital television, biomedicine, digital mobile communication, digital audio and video recording, telecommunications, etc. Methods of digital signal processing are the basis of many of the latest digital developments and various applications of additions, including in the field of computing.

The purpose of the workshop of the course "Signal and Image Processing" is to consolidate practical skills of application of methods and algorithms of digital processing of signals and images. Workshop includes mathematical basics of signal processing (Fourier transforms and Fourier series, convolutions, z-transform) and image processing (two-dimensional Fourier transform, element-by-element transforms, digital filtering and wavelet transforms).

Workshop contains descriptions of 9 two-hour practical works. The structure and content of the workshop correspond to the work program of the course "Signal and Image Processing", lectures and practical classes of which are currently held at the Department of Computer Engineering and Programming NTU "KhPI".

Each student receives an individual task in accordance with the number in the journal and prepares a report of practical work.

Completing practical work involves preliminary study of the relevant section of the course and methodological instructions for this work. To be allowed to perform practical work, the student need to know the basics of math analysis (derivatives, integrals, matrices, complex numbers etc.), the basics of physics as well as being able to use the MATLAB program package to solve practical problems.

The work is considered passed if the student has completed the individual task with all the necessary calculations, written a report and answered theoretical questions

Practical task №1.
Fourier series of continuous signals

If the signal $f(x)$ is defined on the interval $[a;b]$, then its decomposition into the Fourier series is determined by expression (1.1),

$$f(x)=\frac{a_0}{2}+\sum_{k=1}^{+\infty} a_k \cos\frac{k\pi x}{L}+\sum_{k=1}^{+\infty} b_k \sin\frac{k\pi x}{L}, \tag{1.1}$$

where $L=\dfrac{b-a}{2}$, and the coefficients are calculated as follows:

$$a_0=\frac{1}{L}\int_a^b f(x)dx\ ;\ a_k=\frac{1}{L}\int_a^b f(x)\cos\frac{k\pi x}{L}dx\ ;\ b_k=\frac{1}{L}\int_a^b f(x)\sin\frac{k\pi x}{L}dx.$$

Examples

Example №1. Find the decomposition in the Fourier series:

$$f(x)=\begin{cases} A, & 0\le x\le L; \\ 0, & L<x\le 2L. \end{cases}$$

The solution.

Determine the coefficients of decomposition:

$$a_0=\frac{1}{L}\int_a^b f(x)dx=\frac{1}{L}\int_0^L Adx=\frac{A}{L}x\Big|_0^L=A\ ;$$

$$a_k = \frac{1}{L}\int_a^b f(x)\cos\frac{k\pi x}{L}dx = \frac{1}{L}\int_0^L A\cos\frac{k\pi x}{L}dx = \frac{A}{L}\left[\frac{L}{k\pi}\sin\frac{k\pi x}{L}\right]\Bigg|_0^L =$$
$$= \frac{A}{k\pi}\left(\sin k\pi - \sin 0\right) = 0;$$

$$b_k = \frac{1}{L}\int_a^b f(x)\sin\frac{k\pi x}{L}dx = \frac{1}{L}\int_0^L A\sin\frac{k\pi x}{L}dx = \frac{A}{L}\left[-\frac{L}{k\pi}\cos\frac{k\pi x}{L}\right]\Bigg|_0^L =$$
$$= \frac{A}{k\pi}\left(-\cos k\pi + \cos 0\right) = \frac{A}{k\pi}\left[1-(-1)^k\right] = \frac{A}{k\pi}\left[1+(-1)^{k+1}\right].$$

You may notice that for even $k = 2n$, $n = 1, 2, 3, \ldots$

$$b_{2n} = \frac{A}{2n\pi}\left[1+(-1)^{2n+1}\right] = 0\,.$$

For odd $k = 2n-1$, $n = 1, 2, 3, \ldots$

$$b_{2n-1} = \frac{A}{(2n-1)\pi}\left[1+(-1)^{2n}\right] = \frac{2A}{(2n-1)\pi}\,.$$

Thus, the Fourier series decomposition takes the form:

$$f(x) = \frac{A}{2} + \frac{2A}{\pi}\sum_{n=1}^{+\infty}\frac{1}{2n-1}\sin\left(\frac{2n-1}{L}\pi x\right).$$

Example №2. Find the decomposition in the Fourier series:

$$f(x) = \begin{cases} 0, & -1 \le x \le 0; \\ x, & 0 < x \le 1. \end{cases}$$

The solution.

Find the half-period $L = (1-(-1))/2 = 1$.

Determine the coefficients of decomposition:

$$a_0 = \frac{1}{L}\int_a^b f(x)dx = \int_{-1}^{1} f(x)dx = \int_0^1 xdx = \left.\frac{x^2}{2}\right|_0^1 = \frac{1}{2};$$

$$a_k = \frac{1}{L}\int_a^b f(x)\cos\frac{k\pi x}{L}dx = \int_0^1 x\cos k\pi x dx.$$

To find an integral of the product of functions, we use the method of integration in parts:

$$\int_a^b U(x)dV(x) = U(x)V(x)\Big|_a^b - \int_a^b V(x)dU(x).$$

Then

$$a_k = \int_0^1 x\cos k\pi x dx = \left|\begin{array}{l} U(x) = x; dU = dx; \\ dV(x) = \cos k\pi x dx; \\ V(x) = \dfrac{1}{k\pi}\sin k\pi x \end{array}\right| = \left[\frac{1}{k\pi}x\sin k\pi x\right]\Bigg|_0^1 -$$

$$-\frac{1}{k\pi}\int_0^1 \sin k\pi x dx = \underbrace{\frac{1}{k\pi}\sin k\pi}_{=0} + \left.\frac{1}{k^2\pi^2}\cos k\pi\right|_0^1 = \frac{1}{k^2\pi^2}\left[\underbrace{\cos k\pi}_{=(-1)^k} - \underbrace{\cos 0}_{=1}\right] =$$

$$= \frac{1}{k^2\pi^2}\left[(-1)^k - 1\right] = \frac{(-1)^k - 1}{k^2\pi^2};$$

$$b_k = \frac{1}{L}\int_a^b f(x)\sin\frac{k\pi x}{L}dx = \int_0^1 x\sin k\pi x dx == \left|\begin{array}{l} U(x) = x; dU = dx; \\ dV(x) = \sin k\pi x dx; \\ V(x) = -\dfrac{1}{k\pi}\cos k\pi x \end{array}\right| =$$

$$= \left[-\frac{1}{k\pi}x\cos k\pi x\right]\Bigg|_0^1 + \frac{1}{k\pi}\int_0^1 \cos k\pi x dx = -\frac{1}{k\pi}\left[\underbrace{\cos k\pi}_{=(-1)^k} - \underbrace{0\cdot\cos 0}_{=0}\right] +$$

$$+\frac{1}{k^2\pi^2}\sin k\pi x\Bigg|_0^1 = \frac{-(-1)^k}{k\pi} + \frac{1}{k^2\pi^2}\left[\underbrace{\sin k\pi}_{=0} - \underbrace{\sin 0}_{=0}\right] = \frac{(-1)^{k+1}}{k\pi}.$$

Obviously, the Fourier series decomposition looks like this:

$$f(x) = \frac{1}{4} + \sum_{k=1}^{+\infty}\frac{(-1)^k - 1}{k^2\pi^2}\cos k\pi x + \sum_{k=1}^{+\infty}\frac{(-1)^{k+1}}{k\pi}\sin k\pi x .$$

Tasks №1 for independent work

Find the Fourier series of the signal $f(x)=\begin{cases} a_1, & a \le x \le a_1 \\ x, & a_1 < x \le b \end{cases}$ shown in Fig. 1.1. The signal parameters must be selected according to the number from table. 1.1.

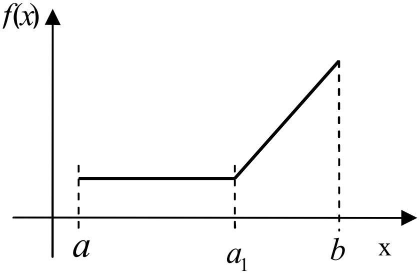


Figure 1.1 - Non-periodic signal specified in the interval $[a;b]$

Table 1.1 - Signal parameters (N - student number according to the journal)

N	a	a_1	b	N	a	a_1	b	N	a	a_1	b
1	-9	-5	0	**11**	-7	-4	-1	**21**	-5	-3	0
2	-8	-4	1	**12**	-6	-3	0	**22**	-4	-2	1
3	-7	-3	2	**13**	-5	-2	1	**23**	-3	-1	2
4	-6	-2	3	**14**	-4	-1	2	**24**	-2	1	3
5	-5	-1	4	**15**	-3	1	3	**25**	-1	1	4
6	-4	1	5	**16**	-2	1	4	**26**	0	2	5
7	-3	1	6	**17**	-1	2	5	**27**	1	3	6
8	-2	2	7	**18**	0	3	6	**28**	2	4	7
9	-1	3	8	**19**	1	4	7	**29**	3	5	8
10	0	4	9	**20**	2	5	8	**30**	4	6	9

$$a_0 = \frac{1}{L}\int_a^b f(x)dx = \frac{1}{L}\left[\int_a^{a_1} a_1 dx + \int_{a_1}^b x dx\right] = \frac{1}{L}\left[a_1 x\Big|_a^{a_1} + \frac{x^2}{2}\Big|_{a_1}^b\right] = \frac{1}{L}\left[a_1^2 - a_1 a + \frac{b^2}{2} - \frac{a_1^2}{2}\right] =$$

$$= \frac{1}{2L}\left[a_1^2 - 2a_1 a + b^2\right]$$

$$a_n = \frac{1}{L}\int_a^b f(x)\cos\frac{n\pi x}{L}dx = \frac{1}{L}\left[\int_a^{a_1} a_1\cos\frac{n\pi x}{L}dx + \int_{a_1}^b x\cos\frac{n\pi x}{L}dx\right] =$$

$$= \frac{1}{L}\left[a_1\frac{L}{n\pi}\sin\frac{n\pi x}{L}\Big|_a^{a_1} + \frac{L}{n\pi}x\sin\frac{n\pi x}{L}\Big|_{a_1}^b - \frac{L}{n\pi}\int_{a_1}^b \sin\frac{n\pi x}{L}dx\right] =$$

$$= \frac{1}{L}\frac{L}{n\pi}\left[a_1\left(\sin\frac{n\pi a_1}{L} - \sin\frac{n\pi a}{L}\right) + b\sin\frac{n\pi b}{L} - a_1\sin\frac{n\pi a_1}{L} + \frac{L}{n\pi}\cos\frac{n\pi x}{L}\Big|_{a_1}^b\right] =$$

$$= \frac{1}{n\pi}\left[-a_1\sin\frac{n\pi a}{L} + b\sin\frac{n\pi b}{L} + \frac{L}{n\pi}\left(\cos\frac{n\pi b}{L} - \cos\frac{n\pi a_1}{L}\right)\right]$$

$$b_n = \frac{1}{L}\int_a^b f(x)\sin\frac{n\pi x}{L}dx = \frac{1}{L}\left[\int_a^{a_1} a_1\sin\frac{n\pi x}{L}dx + \int_{a_1}^b x\sin\frac{n\pi x}{L}dx\right] =$$

$$= \frac{1}{L}\left[-a_1\frac{L}{n\pi}\cos\frac{n\pi x}{L}\Big|_a^{a_1} - \frac{L}{n\pi}x\cos\frac{n\pi x}{L}\Big|_{a_1}^b + \frac{L}{n\pi}\int_{a_1}^b \cos\frac{n\pi x}{L}dx\right] =$$

$$= \frac{1}{L}\frac{L}{n\pi}\left[-a_1\left(\cos\frac{n\pi a_1}{L} - \cos\frac{n\pi a}{L}\right) - b\cos\frac{n\pi b}{L} + a_1\cos\frac{n\pi a_1}{L} + \frac{L}{n\pi}\sin\frac{n\pi x}{L}\Big|_{a_1}^b\right] =$$

$$= \frac{1}{n\pi}\left[a_1\cos\frac{n\pi a}{L} - b\cos\frac{n\pi b}{L} + \frac{L}{n\pi}\left(\sin\frac{n\pi b}{L} - \sin\frac{n\pi a_1}{L}\right)\right]$$

Practical task № 2.
Discrete Fourier Transform

The expression for the direct discrete Fourier transform (DFT) is:

$$X_n(k) = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi kn}{N}}, \tag{2.1}$$

where the factor $e^{-j\frac{2\pi}{N}}$ is called the *rotation factor* and is denoted W_N. The value $X[k]$ is the frequency output of the DFT at the k-th point of the spectrum, where k is in the range from 0 to $N-1$. The value $x[n]$ is the n-th count in the time domain, where n is also in the range from 0 to $N-1$.

The expression for the inverse discrete Fourier transform (IDFT) has the following form:

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j\frac{2\pi}{N}kn}. \tag{2.2}$$

Substitution of rotation factors W_N in (2.1) and (2.2) leads to the definition of DFT and IDFT, respectively, in this form:

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{kn};$$

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] W_N^{-kn}.$$

The obtained expressions allow to convert the time domain of the discrete periodic signal to the frequency and back.

Examples

Example № 1. Calculate the spectrum of the discrete signal $x[n]=(a;0;0;0;0)$.

The solution.

We use expression (2.1) to calculate the spectrum of a discrete non-periodic signal, in which we substitute the value $x[n]$ of a given signal

$$X[k]=\sum_{n=0}^{N-1} x[n]e^{-j\frac{2\pi}{N}kn}=ae^{\frac{-j2\pi}{N}k0}=a.$$

Example № 2. The signal $x[n]$ has the following samples: $x[n]=(1;-1)$. This signal is shown in Fig. 2.1. It is necessary to calculate the first two harmonics of the spectrum of this signal.

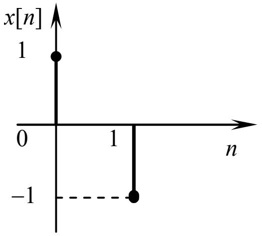


Figure 2.1 - Graph of the input signal

The solution.

There are two ($N=2$) samples. To find complex quantities, it is necessary to calculate the value of the DFT equation for $k=0,1$.

When $k=0$ we get:

$$X[0]=\sum_{n=0}^{1} x[n]\,e^{-j\frac{2\pi}{2}0n}=\sum_{n=0}^{1} x[n]\,e^{-j0}=\sum_{n=0}^{1} x[n];$$

$$X[0]=x[0]+x[1]=1-1=0\ .$$

When $k=1$ we get:

$$X[1]=\sum_{n=0}^{1} x[n]\,e^{-j\frac{2\pi}{2}1n}=\sum_{n=0}^{1} x[n]e^{-j\pi n}=x[0]e^{-j\pi 0}+x[1]e^{-\mathrm{j}\pi 1},$$

$$X[1]=1\,e^{-j0}+(-1)\ e^{-j\pi}=1+(-1)(\cos(\pi)-j\sin(\pi))=1+1=2\ .$$

Therefore, the first two harmonics of the spectrum are equal $X[k]=(0;2)$.

<u>Example № 3.</u> Calculate the DFT of a discrete periodic signal given by three samples $x[n]=(0;1;2)$.

The solution.

$$X[0]=x[0]e^{-j\frac{2\pi}{3}0\cdot 0}+x[1]e^{-j\frac{2\pi}{3}1\cdot 0}+x[2]e^{-j\frac{2\pi}{3}2\cdot 0}=0+1+2=3\,;$$

$$X[1]=x[0]e^{-j\frac{2\pi}{3}0\cdot 1\cdot}+x[1]e^{-j\frac{2\pi}{3}1\cdot 1}+x[2]e^{-j\frac{2\pi}{3}2\cdot 1}=0+1e^{-j\frac{2\pi}{3}}+2e^{-j\frac{4\pi}{3}};$$

$$X[2]=x[0]e^{-j\frac{2\pi}{3}0\cdot 2\cdot}+x[1]e^{-j\frac{2\pi}{3}1\cdot 2}+x[2]e^{-j\frac{2\pi}{3}2\cdot 2}=0e^{-j0^\circ}+1e^{-j\frac{4\pi}{3}}+2e^{-j\frac{8\pi}{3}}.$$

Because

$$e^{-j\frac{2\pi}{3}}=e^{-j\frac{8\pi}{3}}=\frac{-1}{2}-j\frac{\sqrt{3}}{2},$$

$$e^{-j\frac{4\pi}{3}}=\frac{-1}{2}+j\frac{\sqrt{3}}{2},$$

then $X[1]=\dfrac{-3+j\sqrt{3}}{2}$ and $X[2]=\dfrac{-3-j\sqrt{3}}{2}$.

Graphs of the given discrete periodic signal $x[n]$ and the calculated discrete periodic spectrum of amplitudes $X[k]$ are given in fig. 2.2 (amplitude A of a complex number $X=a+jb$ is calculated by expression $A=\sqrt{a^2+b^2}$).

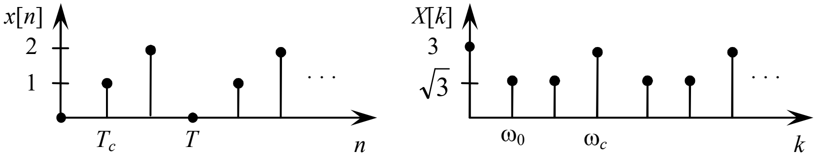


Figure 2.2 - Graph of the discrete signal $x[n]$
and its spectrum of amplitudes $X[k]$

<u>Example № 4.</u> Calculate the values of the discrete signal $x[n]$, if the DFT of this signal is given in the following form $X[k]=(0;1;0;1)$.

The solution.

$$x[0]=\frac{1}{4}(X[0]e^{j0^\circ}+X[1]e^{j0^\circ}+X[2]e^{j0^\circ}+X[3]e^{j0^\circ})=0,5\,;$$

$$x[1]=\frac{1}{4}(X[0]e^{j0^\circ}+X[1]e^{j\frac{\pi}{2}}+X[2]e^{j\pi}+X[3]e^{j\frac{6\pi}{4}})=\frac{1}{4}(0+j1+0-j1)=0;$$

$$x[2]=\frac{1}{4}(X[0]e^{j0^\circ}+X[1]e^{j\pi}+X[2]e^{j2\pi}+X[3]e^{j3\pi})=\frac{1}{4}(0-1+0-1)=-0,5;$$

$$x[3]=\frac{1}{4}(X[0]e^{j0^\circ}+X[1]e^{j\frac{3\pi}{2}}+X[2]e^{j3\pi}+X[3]e^{j\frac{9\pi}{2}})=\frac{1}{4}(0-j+0+j)=0\,.$$

The graph of the sequence $x[n]=(0,5;0;-0,5;0)$ is shown in Fig. 2.3. The graph shows that the signal $x[n]$ is discrete and periodic.

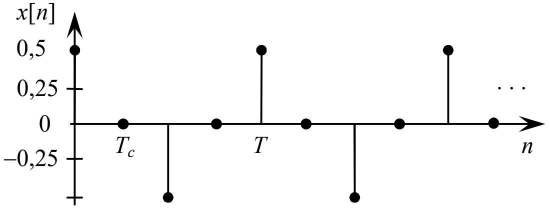


Figure 2.3 - Graph of the signal recovered by the formula IDFT

Tasks № 2 for independent work

1) Calculate the DFT of the discrete signal given by its samples. Plot graphs of signal and amplitude spectrum.

2) Calculate the value of the discrete signal based on the available DFT. Plot graphs of the amplitude spectrum and signal.

Initial data for tasks 1 and 2, see in table. 2.1.

Table 2.1 - Task options (N - student number in the journal)

N	Discrete signal (for task 1)	Spectrum (for task 2)
1	$x[n]=(-2;-1;1;2)$	$X[k]=(-2;-1;1;2)$
2	$x[n]=(-2;1;-1;2)$	$X[k]=(-2;1;-1;2)$
3	$x[n]=(-2;1;2;-1)$	$X[k]=(-2;1;2;-1)$
4	$x[n]=(-2;-1;2;1)$	$X[k]=(-2;-1;2;1)$
5	$x[n]=(1;2;3;4)$	$X[k]=(1;2;3;4)$
6	$x[n]=(1;-2;3;-4)$	$X[k]=(1;-2;3;-4)$
7	$x[n]=(1;-2;-3;4)$	$X[k]=(1;-2;-3;4)$
8	$x[n]=(-3;-1;0;2)$	$X[k]=(-3;-1;0;2)$
9	$x[n]=(-3;0;1;3)$	$X[k]=(-3;0;1;3)$
10	$x[n]=(-3;0;0;3)$	$X[k]=(-3;0;0;3)$
11	$x[n]=(0;2;2;0)$	$X[k]=(0;2;2;0)$
12	$x[n]=(2;0;3;0)$	$X[k]=(2;0;3;0)$
13	$x[n]=(2;0;1;0)$	$X[k]=(2;0;1;0)$
14	$x[n]=(-2;0;3;0)$	$X[k]=(-2;0;3;0)$
15	$x[n]=(2;0;-3;0)$	$X[k]=(2;0;-3;0)$

Practical task № 3.
Fast Fourier Transform

FFT is the generic name for any method for reducing the compuational complexity of a DFT. There are many different methods for reducing complexity.

The most common algorithms are radix 2 FFT.

A radix 2 FFT splits the complete DFT calculation into a combination of 2-point DFTs. Each 2-point DFT contains a basic multiply-accumulate operation called a “butterfly”.

There are *time decimation* and *frequency decimation* FFT algorthms.

3.1. Time decimation FFT algorthm

In Fig. 3.1 shows the basic operation for FFT with time decimation, and in Fig. 3.2 is an example for a 4-point FFT.

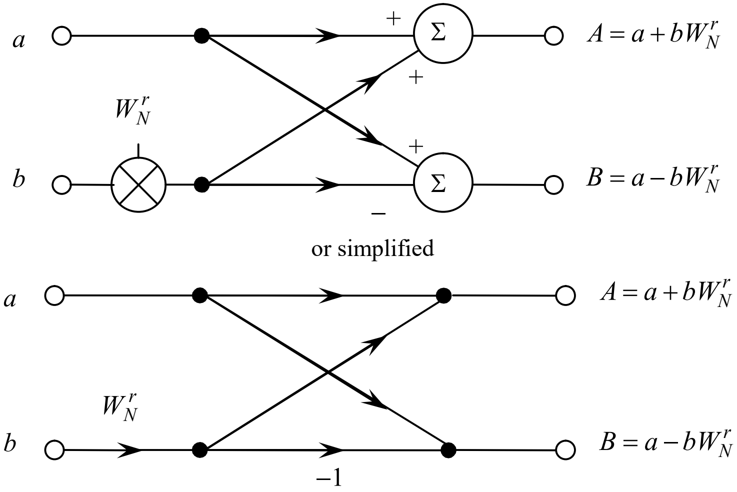


Figure 3.1 - Basic operation in the FFT algorithm with decimation in time

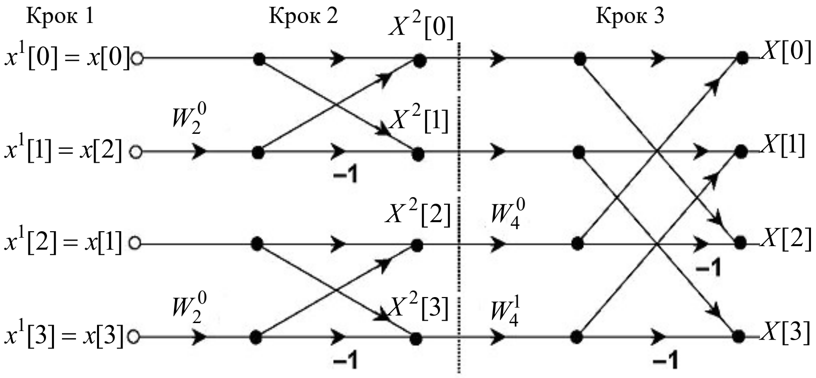


Figure 3.2 - Algorithm for 4-point FFT with decimation in time

An example of a bit reversal algorithm used to implement decimation in time is shown in Table 3.1.

The decimal index n is represented by its binary equivalent, then the binary digits are reversed and converted back to a decimal number.

Table 3.1 - Bit reversal in the time decimation algorithm

Decimal number	0	1	2	3	4	5	6	7
Binary equivalent	000	001	010	011	100	101	110	111
Binary equivalent with reversal	000	100	010	110	001	101	011	111
Decimal equivalent	0	4	2	6	1	5	3	7

3.2. Frequency decimation FFT algorthm

In Fig. 3.3 shows the basic operation for FFT with frequency decimation, and in Fig. 3.4 is an example for a 4-point FFT.

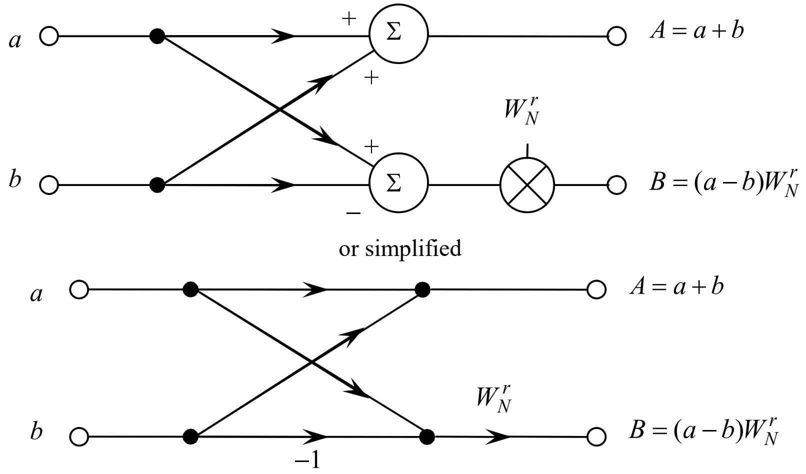


Figure 3.3 - Basic operation in the FFT algorithm with decimation in frequency

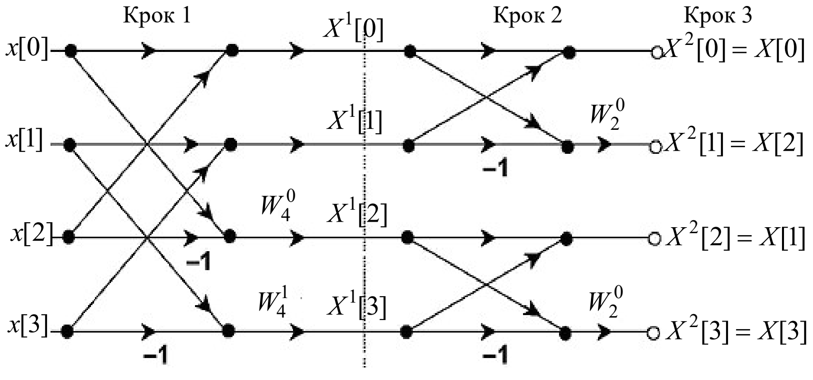


Figure 3.4 - Algorithm for 4-point FFT with decimation in frequency

3.3. Frequency decimation IFFT algorthm

In Fig. 3.5 is an example for a 4-point inverse FFT (IFFT) with decimation in frequency

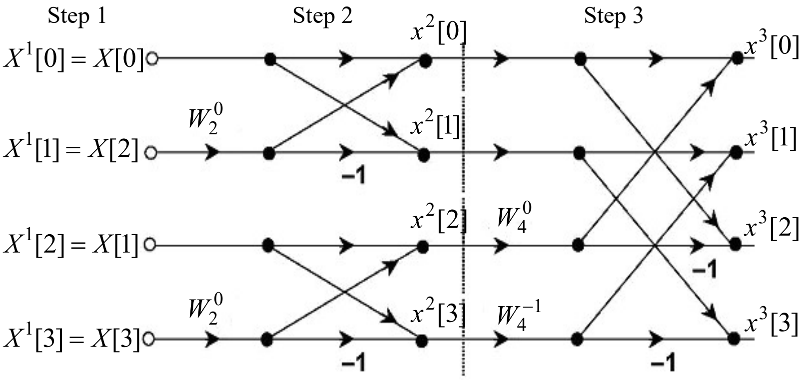


Figure 3.5 - Algorithm for 4-point IFFT with decimation in frequency

Examples

Example № 1.

Compare the spectra of the following signals $x[n]=(1;0;-1)$ and $x[n]=(1;0;-1;0)$.

The solution.

Calculate the spectrum of the discrete signal $x[n]=(1;0;-1)$.

For $k=0$:

$$X[0]=\sum_{n=0}^{2}x[n]\,e^{-j\frac{2\pi}{3}0n}=\sum_{n=0}^{2}x[n], \text{ отже}$$

$$X[0]=x[0]+x[1]+x[2]=1+0-1=0.$$

For $k=1$:

$$X[1]=\sum_{n=0}^{2}x[n]\,e^{-j\frac{2\pi}{3}1n}=x[0]e^{-j\frac{2\pi}{3}0}+x[1]e^{-j\frac{2\pi}{3}1}+x[2]e^{-j\frac{2\pi}{3}2},$$

$$X[1] = 1e^{-j0} + 0\,e^{-j\frac{2\pi}{3}} - 1e^{-j\frac{4\pi}{3}} = 1 - \frac{-1+j\sqrt{3}}{2} = \frac{3-j\sqrt{3}}{2}\,.$$

For $k = 2$:

$$X[2] = \sum_{n=0}^{2} x[n]\,e^{-j\frac{2\pi}{3}2n} = x[0]e^{-j\frac{2\pi}{3}0} + x[1]e^{-j\frac{2\pi}{3}2} + x[2]e^{-j\frac{2\pi}{3}4}\,,$$

$$X[2] = 1e^{-j0} + 0\,e^{-j\frac{4\pi}{3}} - 1e^{-j\frac{8\pi}{3}} = 1 - \frac{-1-j\sqrt{3}}{2} = \frac{3+j\sqrt{3}}{2}\,.$$

Thus, the first three harmonics of the spectrum are: $X[k] = (0;\frac{3-j\sqrt{3}}{2};\frac{3+j\sqrt{3}}{2})$.

Add one sample equal to 0 to the output signal and calculate the spectrum of the received signal $x[n] = (1;0;-1;0)$.

For $k = 0$:

$$X[0] = \sum_{n=0}^{3} x[n]\,e^{-j\frac{2\pi}{4}0n} = \sum_{n=0}^{3} x[n]\,,\; X[0] = 1 + 0 - 1 + 0 = 0\,.$$

For $k = 1$:

$$X[1] = \sum_{n=0}^{3} x[n]\,e^{-j\frac{2\pi}{4}1n} = x[0]e^{-j\frac{\pi}{2}0} + x[1]e^{-j\frac{\pi}{2}1} + x[2]e^{-j\frac{\pi}{2}2} + x[3]e^{-j\frac{\pi}{2}3}\,,$$

$$X[1] = 1e^{-j0} + 0\,e^{-j\frac{\pi}{2}} - 1e^{-j\frac{2\pi}{2}}\,0\,e^{-j\frac{3\pi}{2}} = 1 + 1 = 2\,.$$

For $k = 2$:

$$X[2] = \sum_{n=0}^{3} x[n]\,e^{-j\frac{2\pi}{4}2n} = x[0]e^{-j\frac{\pi}{2}0} + x[1]e^{-j\frac{\pi}{2}2} + x[2]e^{-j\frac{\pi}{2}4} + x[2]e^{-j\frac{\pi}{2}6}$$

$$X[2] = 1e^{-j0} + 0\,e^{-j\pi} - 1e^{-j2\pi} + 0e^{-j3\pi} = 1 - 1 = 0\,.$$

For $k=3$:

$$X[2]=\sum_{n=0}^{3}x[n]\,e^{-j\frac{2\pi}{4}3n}=x[0]e^{-j\frac{\pi}{2}0}+x[1]e^{-j\frac{\pi}{2}3}+x[2]e^{-j\frac{\pi}{2}6}+x[2]e^{-j\frac{\pi}{2}9}$$

$$X[3]=1\,e^{-j0}+0\ e^{-j\frac{3\pi}{2}}-1e^{-j3\pi}+0e^{-j\frac{9\pi}{2}}=1+1=2\,.$$

Therefore, the first four harmonics of the spectrum are $X[k]=(0;2;0;2)$.

Thus, the spectrum contains more harmonics, and the first 3 harmonics of the signals do not match.

Example № 2.

Calculate the FFT of the discrete periodic signal given by the samples $x[n]=(-2;-1;0;1)$. Compare the results of calculations with two algorithms.

The solution.

Time decimation algorithm

Step 1. Reversal of samples

Input numbers	Binary form of the input numbers	Reverse bits	A new order of samples for the implementation of the algorithm
0	00	00	0
1	01	10	2
2	10	01	1
3	11	11	3

Thus, we obtain the following sequence:

$x^1[0]=x[0]=-2$, $x^1[1]=x[2]=0$, $x^1[2]=x[1]=-1$, $x^1[3]=x[3]=1$.

Step 2.

$X^2[0] = x^1[0] + W_2^0 x^1[1] = -2 + 0 = -2$;

$X^2[1] = x^1[0] - W_2^0 x^1[1] = -2 - 0 = -2$;

$X^2[2] = x^1[2] + W_2^0 x^1[3] = -1 + 1 = 0$;

$X^2[3] = x^1[2] - W_2^0 x^1[3] = -1 - 1 = -2$.

Step 3.

$X[0] = X^2[0] + W_4^0 X^2[2] = -2 + 0 = -2$;

$X[1] = X^2[1] + W_4^1 X^2[3] = -2 - 2e^{-j\frac{2\pi}{4}1} = -2 - 2(\cos\frac{\pi}{2} - j\sin\frac{\pi}{2}) = -2 + 2j$;

$X[2] = X^2[0] - W_4^0 X^2[2] = -2 + 0 = -2$;

$X[3] = X^2[1] - W_4^1 X^2[3] = -2 + 2e^{-j\frac{2\pi}{4}1} = -2 + 2(\cos\frac{\pi}{2} - j\sin\frac{\pi}{2}) = -2 - 2j$.

Frequency decimation algorithm

Step 1.

$X^1[0] = x[0] + x[2] = -2 + 0 = -2$;

$X^1[1] = x[1] + x[3] = -1 + 1 = 0$;

$X^1[2] = (x[0] - x[2])W_4^0 = (-2 - 0)e^{-j0} = -2$;

$X^1[3] = (x[1] - x[3])W_4^1 = (-1 - 1)e^{-j\frac{\pi}{2}} = 2j$.

Step 2.

$X^2[0] = X^1[0] + X^1[1] = -2 + 0 = -2$;

$X^2[1] = (X^1[0] - X^1[1])W_2^0 = -2 - 0 = -2$;

$X^2[2] = X^1[2] + X^1[3] = -2 + 2j$;

$X^2[3]=(X^1[2]-X^1[3])W_2^0=-2-2j$.

Step 3. Reversal

$X[0]=X^2[0]=-2$; $X[1]=X^2[2]=-2+2j$;

$X[2]=X^2[1]=-2$; $X[3]=X^2[3]=-2-2j$.

The spectra obtained by the two algorithms completely coincide.

Example № 3. Calculate the value of the input signal using IFFT based on the spectra with was obtained in example 2.

The solution.

Step 1.

$X^1[0]=X[0]=-2$; $X^1[1]=X[2]=-2$;

$X^1[2]=X[1]=-2+2j$; $X^1[3]=X[3]=-2-2j$.

Step 2.

$x^2[0]=X^1[0]+W_2^0X^1[1]=-2-2=-4$;

$x^2[1]=X^1[0]-W_2^0X^1[1]=-2+2=0$;

$x^2[2]=X^1[2]+W_2^0X^1[3]=-2+2j+(-2-2j)=-4$;

$x^2[3]=X^1[2]-W_2^0X^1[3]=-2+2j+2+2j=4j$.

Step 3.

$x^3[0]=x^2[0]+W_4^0x^2[2]=-4-4=-8$;

$x^3[1]=x^2[1]+W_4^{-1}x^2[3]=0+4je^{j\frac{\pi}{2}}=-4j(-j)=-4$;

$x^3[2]=x^2[0]-W_4^0x^2[2]=-4+4=0$;

$x^3[3]=x^2[1]-W_4^{-1}x^2[3]=0-4je^{j\frac{\pi}{2}}=4j(-j)=4$.

Step 4. Perform normalization on *N*, ie $x[n]=\frac{1}{4}x^3[n]$, $x[n]=(-2,-1,0,1)$.

Tasks № 3 for independent work

1) Calculate the FFT of the discrete signal.

2) Calculate the IFTF of a discrete signal given by its spectrum.

The input data for tasks 1 and 2 see in the table. 3.2.

Table 3.2 - Task options (*N* - student number in the journal)

N	Discrete signal (for task 1)	FFT algorithm with decimation in …	Spectrum (for task 2)	IFFT algorithm with decimation in …
1	$x[n]=(-2;-1;1;2)$	time	$X[k]=(-2;-1;1;2)$	time
2	$x[n]=(-2;1;-1;2)$	frequency	$X[k]=(-2;1;-1;2)$	frequency
3	$x[n]=(-2;1;2;-1)$	time	$X[k]=(-2;1;2;-1)$	time
4	$x[n]=(-2;-1;2;1)$	frequency	$X[k]=(-2;-1;2;1)$	frequency
5	$x[n]=(1;2;3;4)$	time	$X[k]=(1;2;3;4)$	time
6	$x[n]=(1;-2;3;-4)$	frequency	$X[k]=(1;-2;3;-4)$	frequency
7	$x[n]=(1;-2;-3;4)$	time	$X[k]=(1;-2;-3;4)$	time
8	$x[n]=(-3;-1;0;2)$	frequency	$X[k]=(-3;-1;0;2)$	frequency
9	$x[n]=(-3;0;1;3)$	time	$X[k]=(-3;0;1;3)$	time
10	$x[n]=(-3;0;0;3)$	frequency	$X[k]=(-3;0;0;3)$	frequency
11	$x[n]=(0;2;2;0)$	time	$X[k]=(0;2;2;0)$	time
12	$x[n]=(2;0;3;0)$	frequency	$X[k]=(2;0;3;0)$	frequency
13	$x[n]=(2;0;1;0)$	time	$X[k]=(2;0;1;0)$	time
14	$x[n]=(-2;0;3;0)$	frequency	$X[k]=(-2;0;3;0)$	frequency
15	$x[n]=(2;0;-3;0)$	time	$X[k]=(2;0;-3;0)$	time

Practical task № 4.
Circular and linear convolution

By definition, the circular convolution of two sequences $x[n]$ and $y[n]$ is equal to

$$g[n] = \sum_{m=0}^{N-1} x[m]y[n-m], \ n = \overline{0, N-1}. \qquad (4.1)$$

Since the sequences $x[n]$ and $y[n]$ are finite, they are considered as periodic, with a repetition period N (Fig. 4.1).

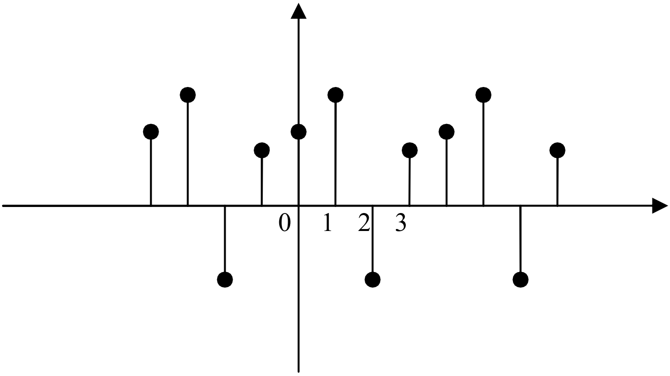


Figure 4.1. - Example of the periodicity of the final signal

Circular convolution

Let's write the expression (4.1)

$$g[0] = x[0]y[0] + x[1]y[N-1] + x[2]y[N-2] + \ldots$$
$$g[1] = x[0]y[1] + x[1]y[0] + x[2]y[N-1] + \ldots$$

…

$g[N-1] = x[0]y[N-1] + x[1]y[N-2] + x[2]y[N-3] + \dots$

This algorithm can be written in matrix form if the sequence $y[n]$ is written in matrix form **Y.**

$$\mathbf{Y} = \begin{bmatrix} y[0] & y[N-1] & y[N-2] & \dots & y[1] \\ y[1] & y[0] & y[N-1] & \cdots & y[2] \\ y[2] & y[1] & y[0] & \cdots & y[3] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ y[N-1] & y[N-2] & y[N-3] & \dots & y[0] \end{bmatrix}.$$

Then the convolution of sequences in matrix form is calculated by multiplying the matrix **Y** by the vector $\vec{x}$:

$$\vec{g} = \mathbf{Y}\vec{x}\ .$$

Linear convolution

To obtain a linear convolution of the final sequences $x[n]$ and $y[n]$ it is necessary to add zeros to the end of the sequences so that the length of each sequence was $N = N_1 + N_2 - 1$, where N_1, N_2 - the length of the sequences $x[n]$ and $y[n]$, accordingly. Then the convolution is calculated by expression (3.1).

Discrete Fourier transform properties

If $X[k]$ and $Y[k]$ are the DFTs of the sequences $x[n]$ and $y[n]$, accordingly, then the DFT of the sequence $g[n]$ is equal to

$$G[k] = X[k]Y[k].$$

Examples

Example № 1.

Find the circular and linear convolution of the sequences $x[n]=(1,2,-1,1)$ and $y[n]=(1,2,3,4)$

The solution.

Find the circular convolution. In the time domain (in matrix form) we will have :

$$\begin{bmatrix} 1 & 4 & 3 & 2 \\ 2 & 1 & 4 & 3 \\ 3 & 2 & 1 & 4 \\ 4 & 3 & 2 & 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ 2 \\ -1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1+8-3+2 \\ 2+2-4+3 \\ 3+4-1+4 \\ 4+6-3+1 \end{bmatrix} = \begin{bmatrix} 8 \\ 3 \\ 10 \\ 9 \end{bmatrix}.$$

The result was a circular convolution $g[n]=(8,3,10,9)$.

Let's check the result with the help of DFT. Find the DFT of the sequence $x=(1,2,-1,1)$:

$X[0]=1+2-1+1=3$;

$$X[1]=1e^{0}+2e^{-j\frac{2\pi}{4}1}-1e^{-j\frac{2\pi}{4}2}+1e^{-j\frac{2\pi}{4}3}=1-2j+1+j=2-j;$$

$$X[2]=1e^{0}+2e^{-j\frac{2\pi}{4}2}-1e^{-j\frac{2\pi}{4}4}+1e^{-j\frac{2\pi}{4}6}=1-2-1-1=-3;$$

$$X[3]=1e^{0}+2e^{-j\frac{2\pi}{4}3}-1e^{-j\frac{2\pi}{4}6}+1e^{-j\frac{2\pi}{4}9}=1+2j+1-j=2+j.$$

Find the DFT of the sequence $y=(1,2,3,4)$:

$Y[0]=1+2+3+4=10$;

$$Y[1]=1e^{0}+2e^{-j\frac{2\pi}{4}1}+3e^{-j\frac{2\pi}{4}2}+4e^{-j\frac{2\pi}{4}3}=1-2j-3+4j=-2+2j;$$

$$Y[2]=1e^{0}+2e^{-j\frac{2\pi}{4}2}+3e^{-j\frac{2\pi}{4}4}+4e^{-j\frac{2\pi}{4}6}=1-2+3-4=-2;$$

$$X[3] = 1e^0 + 2e^{-j\frac{2\pi}{4}3} + 3e^{-j\frac{2\pi}{4}6} + 4e^{-j\frac{2\pi}{4}9} = 1 + 2j - 3 - 4j = -2 - 2j\ .$$

Find the product $X[k]$ and $Y[k]$:

$G[0] = X[0]Y[0] = 10 \times 3 = 30$;

$G[1] = X[1]Y[1] = (2 - j)(-2 + 2j) = -4 + 2j + 4j + 2 = -2 + 6j$;

$G[2] = X[2]Y[2] = (-3)(-2) = 6$;

$G[3] = X[3]Y[3] = (2 + j)(-2 - 2j) = -4 - 2j - 4j + 2 = -2 - 6j$.

Calculate the IDFT of the sequence $G[k]$:

$$g[0] = \frac{1}{4}(30 - 2 + 6j + 6 - 2 - 6j) = 32/4 = 8\ ;$$

$$g[1] = \frac{1}{4}(30e^0 + (-2 + 6j)e^{j\frac{2\pi}{4}1} + 6e^{j\frac{2\pi}{4}2} + (-2 - 6j)e^{j\frac{2\pi}{4}3}) =$$

$$= \frac{1}{4}(30 + (-2 + 6j)j + 6(-1) + (-2 - 6j)(-j)) = 12/4 = 3;$$

$$g[2] = \frac{1}{4}(30e^0 + (-2 + 6j)e^{j\frac{2\pi}{4}2} + 6e^{j\frac{2\pi}{4}4} + (-2 - 6j)e^{j\frac{2\pi}{4}6}) =$$

$$= \frac{1}{4}(30 + (-2 + 6j)(-1) + 6 + (-2 - 6j)(-j)) = 40/4 = 10;$$

$$g[3] = \frac{1}{4}(30e^0 + (-2 + 6j)e^{j\frac{2\pi}{4}3} + 6e^{j\frac{2\pi}{4}6} + (-2 - 6j)e^{j\frac{2\pi}{4}9}) =$$

$$= \frac{1}{4}(30 + (-2 + 6j)(-j) + 6(-1) + (-2 - 6j)j) = 36/4 = 9.$$

The result coincided with the result of the circular convolution calculated by the matrix method.

Calculate the linear convolution. Add the input sequences with zeros so that the length of each sequence is equal $N = N_1 + N_2 - 1 = 4 + 4 - 1 = 7$, ie it is necessary to add 3 zeros so that the input sequence has a length equal to 7.

In the time domain (in matrix form) we will have:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 4 & 3 & 2 \\ 2 & 1 & 0 & 0 & 0 & 4 & 3 \\ 3 & 2 & 1 & 0 & 0 & 0 & 4 \\ 4 & 3 & 2 & 1 & 0 & 0 & 0 \\ 0 & 4 & 3 & 2 & 1 & 0 & 0 \\ 0 & 0 & 4 & 3 & 2 & 1 & 0 \\ 0 & 0 & 0 & 4 & 3 & 2 & 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ 2 \\ -1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 2+2 \\ 3+4-1 \\ 4+6-2+1 \\ 6-3+2 \\ -4+3 \\ 4 \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \\ 6 \\ 9 \\ 7 \\ -1 \\ 4 \end{bmatrix}.$$

The result was a linear convolution $g[n] = (1,4,6,9,7,–1,4)$. Obviously, the linear convolution does not coincide with the circular one.

Example №2.

Find the linear convolution of two sequences $x[n]$=(3,–1,2,4) and $y[n]$=(3,–2,–1).

The solution.

The length of the linear convolution is equal to $N = 4 + 3 - 1 = 6$. So, you need to add 2 zeros to the sequence $x[n]$, and 3 zeros to the sequence $y[n]$.

In the time domain (in matrix form) we will have:

$$\begin{bmatrix} 3 & 0 & 0 & 0 & -1 & -2 \\ -2 & 3 & 0 & 0 & 0 & -1 \\ -1 & -2 & 3 & 0 & 0 & 0 \\ 0 & -1 & -2 & 3 & 0 & 0 \\ 0 & 0 & -1 & -2 & 3 & 0 \\ 0 & 0 & 0 & -1 & -2 & 3 \end{bmatrix} \times \begin{bmatrix} 3 \\ -1 \\ 2 \\ 4 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 9 \\ -9 \\ 5 \\ 9 \\ -10 \\ -4 \end{bmatrix}$$

The result was a linear convolution $g[n] = (9,–9,5,9,–10,–4)$

Tasks № 4 for independent work

Calculate circular and linear convolution of sequences (matrix method). See input data in the table. 4.1.

Table 4.1 - Variants of tasks (N - number of the student on the journal)

N	$x[n]$	$y[n]$	N	$x[n]$	$y[n]$
1	(–2;–1;1;2)	(2;3;1;2)	16	(–2;–3;1;2)	(–1;2;1;4)
2	(–2;1;–1;2)	(2;–3;1;2)	17	(–2;1;–1;4)	(1;–2;1;4)
3	(–2;1;2;–1)	(2;3;–1;2)	18	(–4;1;2;–1)	(1;2;–1;4)
4	(–2;–1;2;1)	(2;3;1;–2)	19	(–2;–1;5;1)	(1;2;1;–4)
5	(1;2;3;4)	(1;3;5;2)	20	(1;2;3;4)	(2;0;–3;0)
6	(1;–2;3;–4)	(–1;3;5;2)	21	(1;–2;3;–4)	(–2;–1;5;1)
7	(1;–2;–3;4)	(1;–3;5;2)	22	(1;–2;–3;4)	(–2;1;–5;1)
8	(–3;–1;0;2)	(1;3;–5;2)	23	(–3;–1;0;2)	(2;1;5;1)
9	(–3;0;1;3)	(1;3;5;–2)	24	(–3;0;1;3)	(2;1;5;–1)
10	(–3;0;0;3)	(5;2;1;3)	25	(–3;0;0;3)	(–1;2;1;4)
11	(0;2;2;0)	(–5;2;1;3)	26	(0;2;2;0)	(5;2;1;–3)
12	(2;0;3;0)	(5;–2;1;3)	27	(2;0;3;0)	(5;–2;1;3)
13	(2;0;3;0)	(5;2;–1;3)	28	(2;0;3;0)	(1;2;–1;4)
14	(–2;0;3;0)	(5;2;1;–3)	29	(–2;0;3;0)	(1;2;1;4)
15	(2;0;–3;0)	(1;2;1;4)	30	(2;0;–3;0)	(1;2;1;–4)

Practical task № 5.
Z- transform

Direct z-transform

A **z-transform** is used to analyze discrete signals and systems.

We define a z-transform that is valid for all n:

$$X(z)=\sum_{n=-\infty}^{\infty}x[n]z^{-n}\ , \tag{5.1}$$

If $x[n]=0$ where $n<0$, then expression (5.1) is reduced to the so-called **one-way z-transform**:

$$X(z)=\sum_{n=0}^{\infty}x[n]z^{-n}\ .$$

The z-transform is a power series with an infinite number of terms, so it may not converge for all z values.

The region in which the z-transform converges is called the **region of convergence**, and in this region the values $X(z)$ are finite.

The z values for which $X(z)=\infty$ are called the **poles** of the function $X(z)$. The z values for which $X(z)=0$ are called **zeros of function** $X(z)$.

Inverse z-transform

Inverse z-transform allows you to reconstruct a discrete signal from its z-image. Symbolically, the inverse z-transform can be defined as

$$x[n]=Z^{-1}\left[X(z)\right].$$

In practice, $X(z)$ is often expressed through the ratio of two polynomials from z^{-n}, for example:

$$X(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \ldots + b_N z^{-N}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \ldots + a_M z^{-M}} .$$

In that case, the inverse z-transform can be found using one of many methods. The most common ones are:

1) the method of expansion in a power series;
2) the method of decomposition into elementary fractions;
3) the method of calculations.

Examples

Example № 1.

Find the z-transform and the convergence region of all discrete time sequences shown in Figs. 7.1.

The solution.

1) The sequence in Fig. 5.1a has a finite length and is not equal to 0 when $n < 0$. The value of the sequence are $x[-6]=0$, $x[-5]=1$, $x[-4]=3$, $x[-3]=5$, $x[-2]=3$, $x[-1]=1$, $x[0]=0$. Then the z-image looks like this:

$$X_1(z) = \sum_{n=-\infty}^{\infty} x[n] z^{-n} = z^5 + 3z^4 + 5z^3 + 3z^2 + z .$$

Obviously, $X(z) = \infty$ when $z = \infty$. Therefore, the region of convergence will be the whole plane z, except $z = \infty$.

2) The sequence in Fig. 5.1b is bilateral and has a finite length. The value of the sequence are $x[-3]=0$, $x[-2]=1$, $x[-1]=3$, $x[0]=5$, $x[1]=3$, $x[2]=1$, $x[3]=0$. Then

the z-image of the sequence has the following form:

$$X_2(z) = \sum_{n=-\infty}^{\infty} x[n]z^{-n} = z^2 + 3z + 5 + 3z^{-1} + z^{-2}.$$

Obviously, $X(z) = \infty$ when $z = 0$ or $z = \infty$. Therefore, the region of convergence does not include only the points $z = 0$ and $z = \infty$.

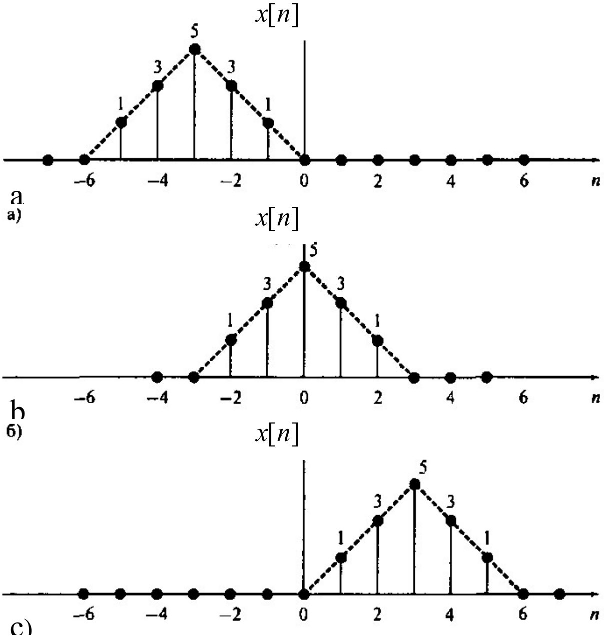


Figure 5.1 - Discrete time sequences

3) The sequence in Fig. 5.1b has a finite length and is equal to 0 when $n < 0$, ie the signal is causal. Sequence values are

$x[0]=0$, $x[1]=1$, $x[2]=3$, $x[3]=5$, $x[4]=3$, $x[5]=1$, $x[6]=0$. Then the z-image will look like:

$$X_3(z)=\sum_{n=-\infty}^{\infty} x[n]z^{-n}=z^{-1}+3z^{-2}+5z^{-3}+3z^{-4}+z^{-5}.$$

In the case $X(z)=\infty$ when $z=0$. Therefore, only the point $z=0$ does not enter the region of convergence.

Example №2. The following z-image of the signal is given:

$$X(z)=\frac{1+3z^{-1}+z^{-2}}{1-z^{-1}-2z^{-2}}\ .$$

It is necessary to find the time sequence by the method of decomposition into a power series.

The solution.

Solve the problem by the method of power series.

If a z-image of a causal signal is given, then it can be decomposed into an infinite series with respect to z^{-1} or z by division into a column (sometimes such division is called synthetic division). In this method, the numerator and denominator of the function *X(z)* are first expressed through a decreasing exponent z, or through an increasing exponent z^{-1} , and then, by dividing by a column, there is a quotient .

Consider the method by example.

The numerator and denominator of $X(z)$ are given in the form of polynomials with increasing degree z^{-1}, therefore, to obtain the original function, we perform the usual division into columns:

$$
\begin{array}{l|l}
1+3z^{-1}+z^{-2} & \underline{1-z^{-1}-2z^{-2}} \\
\underline{1-z^{-1}-2z^{-2}} & 1+4z^{-1}+7z^{-2}+15z^{-3}+\ldots \\
\quad 4z^{-1}+3z^{-2} & \\
\quad \underline{4z^{-1}-4z^{-2}-8z^{-3}} & \\
\qquad\quad 7z^{-2}+8z^{-3} & \\
\qquad\quad \underline{7z^{-2}-7z^{-3}-14z^{-4}} & \\
\qquad\qquad\quad 15z^{-3}+14z^{-4} & \\
\qquad\qquad\quad \underline{15z^{-3}-11z^{-4}-22z^{-5}} & \\
\qquad\qquad\qquad\quad 25z^{-4}+22z^{-5} &
\end{array}
$$

Then the z-image decomposes into a power series:

$$X(z)=1+4z^{-1}+7z^{-2}+15z^{-3}+\ldots$$

You can now directly write the inverse z-transform:

$x[0]=1$, $x[1]=4$, $x[2]=7$, $x[3]=15$ ...

Example № 3. The following z-image of the signal is given:

$$X(z)=\frac{1+3z^{-1}+z^{-2}}{1-z^{-1}-2z^{-2}}\ .$$

It is necessary to find the time sequence by the method of decomposition into elementary fractions.

The solution.

In this method, the z-image is first decomposed into the sum of simple fractions. Then there are the tables of the inverse z-transformation of each elementary fraction. These images are summed, and a general inverse z-transform is obtained.

Table 5.1 - Examples of z-images of some common sequences

Sequence of discrete time $x[n]$ for $n \geq 0$	z-image $X(z)$	Region of convergence $X(z)$
$k\delta[n]$	K	everywhere
k	$\frac{kz}{z-1}$	$\lvert z \rvert > 1$
kn	$\frac{kz}{(z-1)^2}$	$\lvert z \rvert > 1$
kn^2	$\frac{kz(z+1)}{(z-1)^3}$	$\lvert z \rvert > 1$
ke^{-an}	$\frac{kz}{z-e^{-a}}$	$\lvert z \rvert > e^{-a}$
kne^{-an}	$\frac{kze^{-a}}{(z-e^{-a})^2}$	$\lvert z \rvert > e^{-a}$
ka^n	$\frac{kz}{z-a}$	$\lvert z \rvert > a$
kna^n	$\frac{kaz}{(z-a)^2}$	$\lvert z \rvert > a$

If the poles of the function $X(z)$ are of the first order and $N = M$, then we can decompose $X(z)$ as follows:

$$\begin{aligned} X(z) &= B_0 + C_1 \frac{z}{z-p_1} + C_2 \frac{z}{z-p_2} + \ldots + C_M \frac{z}{z-p_M} = \\ &= B_0 + \sum_{k=1}^{M} C_k \frac{z}{z-p_k}, \end{aligned} \tag{5.2}$$

Where p_k - the poles of the function $X(z)$; C_k - coefficients of elementary fractions (surplus function C_k), and the coefficient B_0 is:

$$B_0 = b_N / a_M . \qquad (5.3)$$

The coefficient C_k associated with the pole p_k can be found by multiplying the right and left parts of equation (5.2) by $(z - p_k)/z$ and then replacing $z = p_k$:

$$C_k = \frac{X(z)}{z}(z - p_k)\bigg|_{z=p_k} . \qquad (5.4)$$

Let's find the poles of the function $X(z)$. To do this, equate the denominator to zero and find the roots of the equation: $z^2 - z - 2 = 0$. The roots of the quadratic equation are calculated by the following expression: $p_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$.

Therefore, $p_1 = \frac{1 + \sqrt{1+8}}{2} = 2$, $p_2 = \frac{1 - \sqrt{1+8}}{2} = -1$.

The function $X(z)$ has 2 first order poles at points $z = 2$ and $z = -1$.

Then the function $X(z)$ can be written as follows

$$X(z) = \frac{z^2 + 3z + 1}{(z-2)(z+1)} .$$

According to expression (5.3) $B_0 = 1/(-2) = -0.5$.

Find the subtraction of the function $X(z)$ by expression (5.4):

$$C_1 = \frac{X(z)}{z}(z-p_1)\bigg|_{z=p_1} = \frac{z^2+3z+1}{z(z-2)(z+1)}(z-2)\bigg|_{z=2} = \frac{z^2+3z+1}{z(z+1)}\bigg|_{z=2} =$$

$$= \frac{4+6+1}{2(2+1)} = \frac{11}{6};$$

$$C_2 = \frac{X(z)}{z}(z-p_2)\bigg|_{z=p_2} = \frac{z^2+3z+1}{z(z-2)(z+1)}(z+1)\bigg|_{z=-1} = \frac{z^2+3z+1}{z(z-2)}\bigg|_{z=-1} =$$

$$= \frac{1-3+1}{-1(-1-2)} = -\frac{1}{3}.$$

Therefore, according to expression (5.2), the function $X(z)$ has the form:

$$X(z) = B_0 + C_1\frac{z}{z-p_1} + C_2\frac{z}{z-p_2} = -0.5 + \frac{11}{6}\frac{z}{z-2} - \frac{1}{3}\frac{z}{z+1}.$$

According to table 5.1 we have:

$Z^{-1}[-0.5] = -0.5\delta[n]$;

$Z^{-1}\left[\frac{11}{6}\frac{z}{z-2}\right] = \frac{11}{6}2^n$;

$Z^{-1}\left[-\frac{1}{3}\frac{z}{z+1}\right] = -\frac{1}{3}(-1)^n$;

Thus, the original function has the form

$$x[n] = -0.5\delta[n] + \frac{11}{6}2^n - \frac{1}{3}(-1)^n,\ n \ge 0.$$

Let us check the obtained result with the result obtained by the power series decomposition method (see example №2). Let's calculate some first values of function $x[n]$:

$$x[0] = -0.5 + \frac{11}{6} - \frac{1}{3} = -0.5 + 1.5 = 1\,;$$

$$x[1] = \frac{11}{6}2 - \frac{1}{3}(-1) = \frac{12}{3} = 4\,;$$

$$x[2] = \frac{11}{6}2^2 - \frac{1}{3}(-1)^2 = \frac{21}{3} = 7\,;$$

$$x[3] = \frac{11}{6}2^3 - \frac{1}{3}(-1)^3 = \frac{45}{3} = 15\,.$$

Tasks № 5 for independent work

1) Find the z-transform and the convergence region of the sequence $x[n]$. See input data in the table 5.2 (N - student's number in the journal).

Table 5.2 - Task options for task 1

N	$x[n]$	n	N	$x[n]$	n
1	(−2;−1;1;2)	−1,0,1,2	16	(−2;−3;1;2)	6,7,8,9
2	(−2;1;−1;2)	−2,−1,0,1	17	(−2;1;−1;4)	4,5,6,7
3	(−2;1;2;−1)	−5,−4,−3,−2	18	(−4;1;2;−1)	−7,−6,−5,−4
4	(−2;−1;2;1)	0,1,2,3	19	(−2;−1;5;1)	13,14,15,16
5	(1;2;3;4)	−3,−2,−1,0	20	(1;2;3;4)	−9,−8,−7,−6
6	(1;−2;3;−4)	1,2,3,4	21	(1;−2;3;−4)	8,9,10,11
7	(1;−2;−3;4)	−4,−3,−2,−1	22	(1;−2;−3;4)	15,16,17,18
8	(−3;−1;0;2)	2,3,4,5	23	(−3;−1;0;2)	11,12,13,14
9	(−3;0;1;3)	5,6,7,8	24	(−3;0;1;3)	−11,−10,−9,−8
10	(−3;0;0;3)	−6,−5,−4,−3	25	(−3;0;0;3)	9,10,11,12
11	(0;2;2;0)	10,11,12,13	26	(0;2;2;0)	−13,−12,−11,−10
12	(2;0;3;0)	−8,−7,−6,−5	27	(2;0;3;0)	12,13,14,15
13	(2;0;3;0)	7,8,9,10	28	(2;0;3;0)	−12,−11,−10,−9
14	(−2;0;3;0)	$n = 3,4,5,6$	29	(−2;0;3;0)	−14,−13,−12,−11
15	(2;0;−3;0)	−10,−9,−8,−7	30	(2;0;−3;0)	14,15,16,17

2) Calculate the inverse z-transform for a given function $X(z)$ by the method of decomposition into a power series. See input data in the table 5.3.

3) Calculate the inverse z-transform for a given function $X(z)$ by the method of decomposition into elementary fractions. See input data in the table 5.3.

Table 5.3 - Task options for task 2 and 3.

N	$X(z)$	N	$X(z)$	N	$X(z)$
1	$\frac{1-3z^{-1}-2z^{-2}}{1-3z^{-1}+2z^{-2}}$	11	$\frac{1-3z^{-1}-2z^{-2}}{1+5z^{-1}+6z^{-2}}$	21	$\frac{1-3z^{-1}-2z^{-2}}{1-3z^{-1}+2z^{-2}}$
2	$\frac{1-3z^{-1}-2z^{-2}}{1+z^{-1}-2z^{-2}}$	12	$\frac{1-3z^{-1}-2z^{-2}}{1-z^{-1}-6z^{-2}}$	22	$\frac{1-3z^{-1}-2z^{-2}}{1+z^{-1}-2z^{-2}}$
3	$\frac{1-3z^{-1}-2z^{-2}}{1-z^{-1}-2z^{-2}}$	13	$\frac{1+3z^{-1}-2z^{-2}}{1-5z^{-1}+6z^{-2}}$	23	$\frac{1-3z^{-1}-2z^{-2}}{1-z^{-1}-2z^{-2}}$
4	$\frac{1-3z^{-1}-2z^{-2}}{1+3^{-1}+2z^{-2}}$	14	$\frac{1+3z^{-1}-2z^{-2}}{1+z^{-1}-6z^{-2}}$	24	$\frac{1-3z^{-1}-2z^{-2}}{1+3^{-1}+2z^{-2}}$
5	$\frac{1+3z^{-1}-2z^{-2}}{1-3z^{-1}+2z^{-2}}$	15	$\frac{1+3z^{-1}-2z^{-2}}{1+5z^{-1}+6z^{-2}}$	25	$\frac{1+3z^{-1}-2z^{-2}}{1-3z^{-1}+2z^{-2}}$
6	$\frac{1+3z^{-1}-2z^{-2}}{1+z^{-1}-2z^{-2}}$	16	$\frac{1+3z^{-1}-2z^{-2}}{1-z^{-1}-6z^{-2}}$	26	$\frac{1+3z^{-1}-2z^{-2}}{1+z^{-1}-2z^{-2}}$
7	$\frac{1+3z^{-1}-2z^{-2}}{1-z^{-1}-2z^{-2}}$	17	$\frac{1-3z^{-1}-2z^{-2}}{1+5z^{-1}+6z^{-2}}$	27	$\frac{1+3z^{-1}-2z^{-2}}{1-z^{-1}-2z^{-2}}$
8	$\frac{1+3z^{-1}-2z^{-2}}{1+3z^{-1}+2z^{-2}}$	18	$\frac{1-3z^{-1}-2z^{-2}}{1-z^{-1}-6z^{-2}}$	28	$\frac{1+3z^{-1}-2z^{-2}}{1+3z^{-1}+2z^{-2}}$
9	$\frac{1-3z^{-1}-2z^{-2}}{1-5z^{-1}+6z^{-2}}$	19	$\frac{1+3z^{-1}-2z^{-2}}{1-5z^{-1}+6z^{-2}}$	29	$\frac{1-3z^{-1}-2z^{-2}}{1-5z^{-1}+6z^{-2}}$
10	$\frac{1-3z^{-1}-2z^{-2}}{1+z^{-1}-6z^{-2}}$	20	$\frac{1+3z^{-1}-2z^{-2}}{1+z^{-1}-6z^{-2}}$	30	$\frac{1-3z^{-1}-2z^{-2}}{1+z^{-1}-6z^{-2}}$

Practical task № 6.
Primary image processing

Sampling

Sampling is used to obtain a discrete representation from continuous analog images of the real world. In practice, sampling is carried out by input devices (digital camera, scanner, or others).

For visual perception of processed images on output devices (display, plotter, etc.), an analog image is reconstructed according to its discretized representation

Sampling of a two-dimensional signal also leads to periodization of its spectrum and vice versa. For rectangular sampling, the direct and inverse Fourier transforms are defined by the expressions:

$$S[k,l]=\sum_{n=0}^{N-1}\sum_{m=0}^{M-1}s[n,m]e^{-j\frac{2\pi}{N}nk-j\frac{2\pi}{M}ml}; \qquad (6.1)$$

$$S[k,l]=\sum_{n=0}^{N-1}e^{-j\frac{2\pi}{N}nk}\sum_{m=0}^{M-1}s[n,m]e^{-j\frac{2\pi}{M}ml}; \qquad (6.2)$$

$$s[n,m]=\frac{1}{NM}\sum_{k=0}^{N-1}\sum_{l=0}^{M-1}S[k,l]e^{j\frac{2\pi}{N}nk+j\frac{2\pi}{M}ml}; \qquad (6.3)$$

$$s[n,m]=\frac{1}{NM}\sum_{k=0}^{N-1}e^{j\frac{2\pi}{N}nk}\sum_{l=0}^{M-1}S[k,l]e^{j\frac{2\pi}{M}ml}. \qquad (6.4)$$

Expressions (6.1) and (6.2) are used to calculate the direct two-dimensional DFT, and expressions (6.3) and (6.4) are used to calculate the inverse two-dimensional DFT. These expressions show that the two-dimensional DFT on a rectangular data sampling raster can be calculated using one-dimensional sequential DFTs. The other expressions (6.2) and (6.4) are one-dimensional DFTs of the cross sections of the functions $s[n,m]$ and $S[k,l]$ along the lines n and k, respectively, and the first ones are one-dimensional DFTs of the calculated functions in the cross sections m and l. In other words, the initial matrices of values $s[n,m]$ and $S[k,l]$ are recalculated first in the intermediate matrices with DFT by rows (or columns), and intermediate - in the final with DFT by columns (or by rows, respectively).

Examples of calculation of two-dimensional DFT

Example № 1.

Suppose a two-dimensional signal is specified

$$\mathbf{s} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 \\ 4 & 4 & 4 & 4 \end{bmatrix}.$$

Calculate by expression (6.1) the first few samples of the spectrum of a given signal:

$$S[0,0] = 1+1+1+1+2+2+2+2+3+3+3+3+4+4+4+4 = 40;$$

$$S[0,1] = 1 + e^{-\frac{\pi}{2}\cdot 1\cdot 1} + e^{-\frac{\pi}{2}\cdot 2\cdot 1} + e^{-\frac{\pi}{2}\cdot 3\cdot 1} + 2 + 2e^{-\frac{\pi}{2}\cdot 1\cdot 1} + 2e^{-\frac{\pi}{2}\cdot 2\cdot 1} + 2e^{-\frac{\pi}{2}\cdot 3\cdot 1} +$$
$$+ 3 + 3e^{-\frac{\pi}{2}\cdot 1\cdot 1} + 3e^{-\frac{\pi}{2}\cdot 2\cdot 1} + 3e^{-\frac{\pi}{2}\cdot 3\cdot 1} + 4 + 4e^{-\frac{\pi}{2}\cdot 1\cdot 1} + 4e^{-\frac{\pi}{2}\cdot 2\cdot 1} + 4e^{-\frac{\pi}{2}\cdot 3\cdot 1} =$$
$$= 1 - j - 1 + j + 2(1 - j - 1 + j) + 3(1 - j - 1 + j) + 4(1 - j - 1 + j) = 0;$$

$$S[0,2]=1+e^{-\frac{\pi}{2}\cdot1\cdot2}+e^{-\frac{\pi}{2}\cdot2\cdot2}+e^{-\frac{\pi}{2}\cdot3\cdot2}+2+2e^{-\frac{\pi}{2}\cdot1\cdot2}+2e^{-\frac{\pi}{2}\cdot2\cdot2}+2e^{-\frac{\pi}{2}\cdot3\cdot2}+$$

$$+3+3e^{-\frac{\pi}{2}\cdot1\cdot2}+3e^{-\frac{\pi}{2}\cdot2\cdot2}+3e^{-\frac{\pi}{2}\cdot3\cdot2}+4+4e^{-\frac{\pi}{2}\cdot1\cdot2}+4e^{-\frac{\pi}{2}\cdot2\cdot2}+4e^{-\frac{\pi}{2}\cdot3\cdot2}=$$

$$=1-1+1-1+2(1-1+1-1)+3(1-1+1-1)+4(1-1+1-1)=0;$$

$$S[0,3]=1+e^{-\frac{\pi}{2}\cdot1\cdot3}+e^{-\frac{\pi}{2}\cdot2\cdot3}+e^{-\frac{\pi}{2}\cdot3\cdot3}+2+2e^{-\frac{\pi}{2}\cdot1\cdot3}+2e^{-\frac{\pi}{2}\cdot2\cdot3}+2e^{-\frac{\pi}{2}\cdot3\cdot3}+$$

$$+3+3e^{-\frac{\pi}{2}\cdot1\cdot3}+3e^{-\frac{\pi}{2}\cdot2\cdot3}+3e^{-\frac{\pi}{2}\cdot3\cdot3}+4+4e^{-\frac{\pi}{2}\cdot1\cdot3}+4e^{-\frac{\pi}{2}\cdot2\cdot3}+4e^{-\frac{\pi}{2}\cdot3\cdot3}=$$

$$=1+j-1-j+2(1+j-1-j)+3(1+j-1-j)+4(1+j-1-j)=0;$$

$$S[1,0]=1+1+1+1+2e^{-j\frac{\pi}{2}\cdot1\cdot1}+2e^{-j\frac{\pi}{2}\cdot1\cdot1}+2e^{-j\frac{\pi}{2}\cdot1\cdot1}+2e^{-j\frac{\pi}{2}\cdot1\cdot1}+3e^{-j\frac{\pi}{2}\cdot2\cdot1}+$$

$$+3e^{-j\frac{\pi}{2}\cdot2\cdot1}+3e^{-j\frac{\pi}{2}\cdot2\cdot1}+3e^{-j\frac{\pi}{2}\cdot2\cdot1}+4e^{-j\frac{\pi}{2}\cdot3\cdot1}+4e^{-j\frac{\pi}{2}\cdot3\cdot1}+4e^{-j\frac{\pi}{2}\cdot3\cdot1}+$$

$$+4e^{-j\frac{\pi}{2}\cdot3\cdot1}=4(1+2e^{-j\frac{\pi}{2}\cdot1\cdot1}+3e^{-j\frac{\pi}{2}\cdot2\cdot1}+4e^{-j\frac{\pi}{2}\cdot3\cdot1})=$$

$$=4(1-2j-3+4j)=-8+8j;$$

$$S[1,1]=1+e^{-j\frac{\pi}{2}\cdot1\cdot1}+e^{-j\frac{\pi}{2}\cdot2\cdot1}+e^{-j\frac{\pi}{2}\cdot3\cdot1}+$$

$$+2e^{-j\frac{\pi}{2}\cdot1\cdot1}+2e^{-j\frac{\pi}{2}\cdot1\cdot1-j\frac{\pi}{2}\cdot1\cdot1}+2e^{-j\frac{\pi}{2}\cdot1\cdot1-j\frac{\pi}{2}\cdot2\cdot1}+2e^{-j\frac{\pi}{2}\cdot1\cdot1-j\frac{\pi}{2}\cdot3\cdot1}+$$

$$+3e^{-j\frac{\pi}{2}\cdot2\cdot1}+3e^{-j\frac{\pi}{2}\cdot2\cdot1-j\frac{\pi}{2}\cdot1\cdot1}+3e^{-j\frac{\pi}{2}\cdot2\cdot1-j\frac{\pi}{2}\cdot2\cdot1}+3e^{-j\frac{\pi}{2}\cdot2\cdot1-j\frac{\pi}{2}\cdot3\cdot1}+$$

$$+4e^{-j\frac{\pi}{2}\cdot3\cdot1}+4e^{-j\frac{\pi}{2}\cdot3\cdot1-j\frac{\pi}{2}\cdot1\cdot1}+4e^{-j\frac{\pi}{2}\cdot3\cdot1-j\frac{\pi}{2}\cdot2\cdot1}+4e^{-j\frac{\pi}{2}\cdot3\cdot1-j\frac{\pi}{2}\cdot3\cdot1}=$$

$$=1-j-1+j+2(-j-1+j+1)+3(-1+j+1-j)+4(j+1-j-1)=0.$$

Obviously, the calculation of two-dimensional DFT by expression (6.1) is a rather labor-intensive process. Therefore, we use one-dimensional DFT (6.2) to calculate the two-dimensional spectrum.

First, calculate the one-dimensional DFT by lines. Since there are 4 values in each line, the FFT algorithm can be used to calculate the spectrum.

We use the FFT algorithm with time decimation (Fig. 3.2).

Calculate the FFT of the 1st line, i.e. $x[n]=(1,1,1,1)$.

Step 1. Since all values of the discrete sequence are the same, then $x^1[n]=(1,1,1,1)$.

Step 2.

$$X^2[0]=x^1[0]+W_2^0x^1[1]=1+1=2\;;$$
$$X^2[1]=x^1[0]-W_2^0x^1[1]=1-1=0\;;$$
$$X^2[2]=x^1[2]+W_2^0x^1[3]=1+1=2\;;$$
$$X^2[3]=x^1[2]-W_2^0x^1[3]=1-1=0\;.$$

Step 3.

$$X[0]=X^2[0]+W_4^0X^2[2]=2+2=4\;;$$
$$X[1]=X^2[1]+W_4^1X^2[3]=0+0e^{-j\frac{2\pi}{4}1}=0\;;$$
$$X[2]=X^2[0]-W_4^0X^2[2]=2-2=0\;;$$
$$X[3]=X^2[1]-W_4^1X^2[3]=0-0e^{-j\frac{2\pi}{4}1}=0\;.$$

Calculate the FFT of the 2-nd line, i.e. $x[n]=(2,2,2,2)$.

Step 1. Since all values of the discrete sequence are the same, then $x^1[n]=(2,2,2,2)$.

Step 2.

$$X^2[0]=x^1[0]+W_2^0x^1[1]=2+2=4\;;$$
$$X^2[1]=x^1[0]-W_2^0x^1[1]=2-2=0\;;$$
$$X^2[2]=x^1[2]+W_2^0x^1[3]=2+2=4\;;$$
$$X^2[3]=x^1[2]-W_2^0x^1[3]=2-2=0\;.$$

Step 3.

$$X[0]=X^2[0]+W_4^0X^2[2]=4+4=8\ ;$$

$$X[1]=X^2[1]+W_4^1X^2[3]=0+0e^{-j\frac{2\pi}{4}1}=0\ ;$$

$$X[2]=X^2[0]-W_4^0X^2[2]=4-4=0\ ;$$

$$X[3]=X^2[1]-W_4^1X^1[3]=0-0e^{-j\frac{2\pi}{4}1}=0\ .$$

Calculate the FFT of the 3-rd line, i.e. $x[n]=(3,3,3,3)$.

Step 1. Since all values of the discrete sequence are the same, then $x^1[n]=(3,3,3,3)$.

Step 2.

$$X^2[0]=x^1[0]+W_2^0x^1[1]=3+3=6\ ;$$

$$X^2[1]=x^1[0]-W_2^0x^1[1]=3-3=0\ ;$$

$$X^2[2]=x^1[2]+W_2^0x^1[3]=3+3=6\ ;$$

$$X^2[3]=x^1[2]-W_2^0x^1[3]=3-3=0\ .$$

Step 3.

$$X[0]=X^2[0]+W_4^0X^2[2]=6+6=12\ ;$$

$$X[1]=X^2[1]+W_4^1X^2[3]=0+0e^{-j\frac{2\pi}{4}1}=0\ ;$$

$$X[2]=X^2[0]-W_4^0X^2[2]=6-6=0\ ;$$

$$X[3]=X^2[1]-W_4^1X^1[3]=0-0e^{-j\frac{2\pi}{4}1}=0\ .$$

Finally, we calculate the FFT of the 4-th line, i.e. $x[n]=(4,4,4,4)$.

Step 1. Since all values of the discrete sequence are the same, then $x^1[n]=(4,4,4,4)$.

Step 2.

$$X^2[0] = x^1[0] + W_2^0 x^1[1] = 4 + 4 = 8\,;$$
$$X^2[1] = x^1[0] - W_2^0 x^1[1] = 4 - 4 = 0\,;$$
$$X^2[2] = x^1[2] + W_2^0 x^1[3] = 4 + 4 = 8\,;$$
$$X^2[2] = x^1[1] - W_2^0 x^1[3] = 4 - 4 = 0\,.$$

Step 3.

$$X[0] = X^2[0] + W_4^0 X^2[2] = 8 + 8 = 16\,;$$
$$X[1] = X^2[1] + W_4^1 X^2[3] = 0 + 0e^{-j\frac{2\pi}{4}1} = 0\,;$$
$$X[2] = X^2[0] - W_4^0 X^2[2] = 8 - 8 = 0\,;$$
$$X[3] = X^2[1] - W_4^1 X^1[3] = 0 - 0e^{-j\frac{2\pi}{4}1} = 0\,.$$

As a result, we obtain the following intermediate matrix, each row of which is composed of the obtained one-dimensional DFT rows of the initial matrix **s**:

$$\mathbf{S}' = \begin{bmatrix} 4 & 0 & 0 & 0 \\ 8 & 0 & 0 & 0 \\ 12 & 0 & 0 & 0 \\ 16 & 0 & 0 & 0 \end{bmatrix}.$$

Then for the matrix $\mathbf{S}'$ we calculate one-dimensional DFT on columns. Since the 2-nd, 3-rd and 4-th columns of the intermediate matrix contain zero elements, the corresponding spectra will also contain only zero values. Therefore, it is necessary to calculate only the spectrum of the 1-st column, i.e. the sequence $x[n] = (4,8,12,16)$. We will also use the FFT algorithm with time decimation (Fig. 3.2).

Step 1. Reversal of samples

$x^1[0]=x[0]=4$, $x^1[1]=x[2]=12$, $x^1[2]=x[1]=8$, $x^1[3]=x[3]=16$, i.e. $x^1[n]=(4{,}12{,}8{,}16)$.

Step 2.

$$X^2[0]=x^1[0]+W_2^0x^1[1]=4+12=16;$$
$$X^2[1]=x^1[0]-W_2^0x^1[1]=4-12=-8;$$
$$X^2[2]=x^1[2]+W_2^0x^1[3]=8+16=24;$$
$$X^2[3]=x^1[2]-W_2^0x^1[3]=8-16=-8.$$

Step 3.

$$X[0]=X^2[0]+W_4^0X^2[2]=16+24=40;$$
$$X[1]=X^2[1]+W_4^1X^2[3]=-8-8e^{-j\frac{2\pi}{4}1}=-8+8j;$$
$$X[2]=X^2[0]-W_4^0X^2[2]=16-24=-8;$$
$$X[3]=X^2[1]-W_4^1X^2[3]=-8+8e^{-j\frac{2\pi}{4}1}=-8-8j.$$

As a result, we obtain a two-dimensional spectrum of the input two-dimensional signal:

$$\mathbf{S}=\begin{bmatrix} 40 & 0 & 0 & 0 \\ -8+8j & 0 & 0 & 0 \\ -8 & 0 & 0 & 0 \\ -8-8j & 0 & 0 & 0 \end{bmatrix}.$$

Comparing the results obtained using expressions (6.1) and (6.2), it is easy to see that the values of the corresponding samples of the two-dimensional spectrum completely coincide.

Perform the inverse two-dimensional DFT of the obtained spectrum by expression (6.4). To calculate, we use the FFT algorithm with frequency decimation (Fig. 4.4).

Calculate IFFT 1-st line, i.e. $X[k]=(40{,}0{,}0{,}0)$.

Step 1. Reversal of samples $X^1[0]=X[0]=40$; $X^1[1]=X[2]=0$; $X^1[2]=X[1]=0$; $X^1[3]=X[3]=0$.

Step 2.

$$x^2[0]=X^1[0]+W_2^0X^1[1]=40+0=40\ ;$$
$$x^2[1]=X^1[0]-W_2^0X^1[1]=40-0=40\ ;$$
$$x^2[2]=X^1[2]+W_2^0X^1[3]=0+0=0\ ;$$
$$x^2[3]=X^1[2]-W_2^0X^1[3]=0-0=0\ .$$

Step 3.

$$x^3[0]=x^2[0]+W_4^0x^2[2]=40+0=40\ ;$$
$$x^3[1]=x^2[1]+W_4^{-1}x^2[3]=40+0je^{j\frac{\pi}{2}}=40;$$
$$x^3[2]=x^2[0]-W_4^0x^2[2]=40-0=40;$$
$$x^3[3]=x^2[1]-W_4^{-1}x^2[3]=40-0je^{j\frac{\pi}{2}}=40\ .$$

Step 4. Rationing on *N*, i.e. $x[n]=\dfrac{1}{4}x^3[n]$,

$$x[n]=(10{,}10{,}10{,}10)\ .$$

Calculate IFFT 2-nd line, i.e. $X[k]=(-8+8j{,}0{,}0{,}0)$.

Step 1. Reversal of samples $X^1[0]=X[0]=-8+8j$; $X^1[1]=X[2]=0$; $X^1[2]=X[1]=0$; $X^1[3]=X[3]=0$.

Step 2.

$$x^2[0]=X^1[0]+W_2^0X^1[1]=-8+8j+0=-8+8j\ ;$$

$x^2[1] = X^1[0] - W_2^0 X^1[1] = -8 + 8j - 0 = -8 + 8j$;
$x^2[2] = X^1[2] + W_2^0 X^1[3] = 0 + 0 = 0$;
$x^2[3] = X^1[2] - W_2^0 X^1[3] = 0 - 0 = 0$.

Step 3.

$x^3[0] = x^2[0] + W_4^0 x^2[2] = -8 + 8j + 0 = -8 + 8j$;
$x^3[1] = x^2[1] + W_4^{-1} x^2[3] = -8 + 8j + 0je^{j\frac{\pi}{2}} = -8 + 8j$;
$x^3[2] = x^2[0] - W_4^0 x^2[2] = -8 + 8j - 0 = -8 + 8j$;
$x^3[3] = x^2[1] - W_4^{-1} x^2[3] = -8 + 8j - 0je^{j\frac{\pi}{2}} = -8 + 8j$.

Step 4. Rationing on *N*, i.e. $x[n] = \frac{1}{4} x^3[n]$,

$$x[n] = (-2 + 2j, -2 + 2j, -2 + 2j, -2 + 2j).$$

Calculate IFFT 3-rd line, i.e. $X[k] = (-8,0,0,0)$.

Step 1. Reversal of samples $X^1[0] = X[0] = -8$; $X^1[1] = X[2] = 0$; $X^1[2] = X[1] = 0$; $X^1[3] = X[3] = 0$.

Step 2.

$x^2[0] = X^1[0] + W_2^0 X^1[1] = -8 + 0 = -8$;
$x^2[1] = X^1[0] - W_2^0 X^1[1] = -8 - 0 = -8$;
$x^2[2] = X^1[2] + W_2^0 X^1[3] = 0 + 0 = 0$;
$x^2[3] = X^1[2] - W_2^0 X^1[3] = 0 - 0 = 0$.

Step 3.

$x^3[0] = x^2[0] + W_4^0 x^2[2] = -8 + 0 = -8$;

$$x^3[1] = x^2[1] + W_4^{-1} x^2[3] = -8 + 0 je^{j\frac{\pi}{2}} = -8\ ;$$
$$x^3[2] = x^2[0] - W_4^0 x^2[2] = -8 - 0 = -8\ ;$$
$$x^3[3] = x^2[1] - W_4^{-1} x^2[3] = -8 - 0 je^{j\frac{\pi}{2}} = -8\ .$$

Step 4. Rationing on *N*, i.e. $x[n] = \frac{1}{4} x^3[n]$,

$$x[n] = (-2,-2,-2,-2)\ .$$

Calculate IFFT 4-th line, i.e. $X[k] = (-8-8j,0,0,0)$

Step 1. Reversal of samples $X^1[0] = X[0] = -8-8j$; $X^1[1] = X[2] = 0$; $X^1[2] = X[1] = 0$; $X^1[3] = X[3] = 0$.

Step 2.

$$x^2[0] = X^1[0] + W_2^0 X^1[1] = -8-8j+0 = -8-8j\ ;$$
$$x^2[1] = X^1[0] - W_2^0 X^1[1] = -8-8j-0 = -8-8j\ ;$$
$$x^2[2] = X^1[2] + W_2^0 X^1[3] = 0+0 = 0\ ;$$
$$x^2[3] = X^1[2] - W_2^0 X^1[3] = 0-0 = 0\ .$$

Step 3.

$$x^3[0] = x^2[0] + W_4^0 x^2[2] = -8-8j+0 = -8-8j\ ;$$
$$x^3[1] = x^2[1] + W_4^{-1} x^2[3] = -8-8j+0 je^{j\frac{\pi}{2}} = -8-8j\ ;$$
$$x^3[2] = x^2[0] - W_4^0 x^2[2] = -8-8j-0 = -8-8j\ ;$$
$$x^3[3] = x^2[1] - W_4^{-1} x^2[3] = -8-8j-0 je^{j\frac{\pi}{2}} = -8-8j\ .$$

Step 4. Rationing on *N*, i.e. $x[n] = \frac{1}{4} x^3[n]$,

$$x[n] = (-2-2j, -2-2j, -2-2j, -2-2j).$$

As a result, we obtain an intermediate matrix

$$\mathbf{s}' = \begin{bmatrix} 10 & 10 & 10 & 10 \\ -2+2j & -2+2j & -2+2j & -2+2j \\ -2 & -2 & -2 & -2 \\ -2-2j & -2-2j & -2-2j & -2-2j \end{bmatrix}.$$

Now calculate the one-dimensional DFT on the columns of the intermediate matrix $\mathbf{s}'$. Perform the calculation for the 1-st column, i.e.

$X[k] = (10, -2+2j, -2, -2-2j)$.

Step 1. Reversal of samples $X^1[0] = X[0] = 10$; $X^1[1] = X[2] = -2$; $X^1[2] = X[1] = -2+2j$; $X^1[3] = X[3] = -2-2j$.

Step 2.

$$x^2[0] = X^1[0] + W_2^0 X^1[1] = 10 - 2 = 8\ ;$$
$$x^2[1] = X^1[0] - W_2^0 X^1[1] = 10 + 2 = 12\ ;$$
$$x^2[2] = X^1[2] + W_2^0 X^1[3] = -2 + 2j + (-2-2j) = -4\ ;$$
$$x^2[3] = X^1[2] - W_2^0 X^1[3] = -2 + 2j - (-2-2j) = 4j\ .$$

Step 3.

$$x^3[0] = x^2[0] + W_4^0 x^2[2] = 8 - 4 = 4\ ;$$
$$x^3[1] = x^2[1] + W_4^{-1} x^2[3] = 12 + 4je^{j\frac{\pi}{2}} = 12 + 4j \cdot j = 12 - 4 = 8\ ;$$
$$x^3[2] = x^2[0] - W_4^0 x^2[2] = 8 - (-4) = 12\ ;$$

$$x^3[3] = x^2[1] - W_4^{-1} x^2[3] = 12 - 4je^{j\frac{\pi}{2}} = 12 - 4j \cdot j = 12 + 4 = 16.$$

Step 4. Rationing on N, i.e. $x[n] = \frac{1}{4} x^3[n]$, $x[n] = (1,2,3,4)$.

Since all the columns of the intermediate matrix are equal to each other, the resulting two-dimensional signal has the following form

$$\mathbf{s} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 \\ 4 & 4 & 4 & 4 \end{bmatrix},$$

which completely coincides with the specified input signal.

Tasks № 6 for independent work

Calculate the direct DFT of a given two-dimensional array. See the input data in the table 6.1 (*N* - student number in the journal).

Table 6.1 - Input two-dimensional arrays

N	*S*	*N*	*S*	*N*	*S*
1	$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 0 & 0 & 0 & 0 \\ -1 & -2 & -3 & -4 \\ 0 & 0 & 0 & 0 \end{bmatrix}$	8	$\begin{bmatrix} 1 & 0 & 2 & 0 \\ 1 & 1 & 2 & 0 \\ 1 & 0 & 2 & 0 \\ 1 & 1 & 2 & 0 \end{bmatrix}$	15	$\begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 2 & 2 & 0 & 0 \\ 2 & 2 & 0 & 0 \end{bmatrix}$
2	$\begin{bmatrix} -1 & -2 & -3 & -4 \\ 0 & 0 & 0 & 0 \\ 1 & 2 & 3 & 4 \\ 0 & 0 & 0 & 0 \end{bmatrix}$	9	$\begin{bmatrix} 1 & 0 & 2 & 0 \\ 3 & 0 & 4 & 0 \\ 0 & 1 & 0 & 2 \\ 0 & 3 & 0 & 4 \end{bmatrix}$	16	$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 2 & 2 \\ 0 & 0 & 2 & 2 \end{bmatrix}$
3	$\begin{bmatrix} 1 & 0 & -1 & 0 \\ 2 & 0 & -2 & 0 \\ 3 & 0 & -3 & 0 \\ 4 & 0 & -4 & 0 \end{bmatrix}$	10	$\begin{bmatrix} -1 & 0 & -2 & 0 \\ -3 & 0 & -4 & 0 \\ 0 & -1 & 0 & -2 \\ 0 & -3 & 0 & -4 \end{bmatrix}$	17	$\begin{bmatrix} -1 & -1 & 0 & 0 \\ -1 & -1 & 0 & 0 \\ 0 & 0 & -2 & -2 \\ 0 & 0 & -2 & -2 \end{bmatrix}$
4	$\begin{bmatrix} -1 & 0 & 1 & 0 \\ -2 & 0 & 2 & 0 \\ -3 & 0 & 3 & 0 \\ -4 & 0 & 4 & 0 \end{bmatrix}$	11	$\begin{bmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 2 & 3 \\ 1 & 2 & 0 & 3 \\ 1 & 2 & 3 & 0 \end{bmatrix}$	18	$\begin{bmatrix} 0 & 0 & -1 & -1 \\ 0 & 0 & -1 & -1 \\ -2 & -2 & 0 & 0 \\ -2 & -2 & 0 & 0 \end{bmatrix}$
5	$\begin{bmatrix} 1 & 2 & 1 & 2 \\ 0 & 4 & 0 & 4 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$	12	$\begin{bmatrix} 0 & -1 & -2 & -3 \\ -1 & 0 & -2 & -3 \\ -1 & -2 & 0 & -3 \\ -1 & -2 & -3 & 0 \end{bmatrix}$	19	$\begin{bmatrix} 1 & 0 & 1 & 0 \\ 2 & 0 & 2 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 2 & 0 & 2 \end{bmatrix}$

6	$\begin{bmatrix}0&0&0&0\\1&1&1&1\\1&2&1&2\\3&4&3&4\end{bmatrix}$	13	$\begin{bmatrix}-1&0&-2&0\\-3&0&-4&0\\0&-1&0&-2\\0&-3&0&-4\end{bmatrix}$	20	$\begin{bmatrix}-1&0&-1&0\\-2&0&-2&0\\0&-1&0&-1\\0&-2&0&-2\end{bmatrix}$
7	$\begin{bmatrix}-1&1&-1&1\\0&0&0&0\\1&2&3&4\\4&3&2&1\end{bmatrix}$	14	$\begin{bmatrix}0&0&0&0\\-1&-1&-1&-1\\-1&-2&-1&-2\\-3&-4&-3&-4\end{bmatrix}$	21	$\begin{bmatrix}1&2&-1&-2\\1&2&-1&-2\\-1&-2&1&2\\-1&-2&1&2\end{bmatrix}$

Practical task № 7.

Element-by-element processing of images

In a large number of information systems, the presentation of the results of data processing in the form of an image displayed on the screen for use by the observer is used. The procedure that provides such a representation is called *visualization.* It is desirable to give the displayed image such qualities due to which its perception by the person would be as comfortable as possible. It is often useful to emphasize, strengthen some features, peculiarities, nuances of the picture, in order to improve its subjective perception.

Element-by-element processing is a group of procedures in which the result of processing at any point in the frame depends only on the value of the input image at the same point.

Advantages:

- simplicity;
- subjective improvement of visual quality.

Very often, element-by-element processing is used as the final stage in solving a more complex image processing problem.

Let us x_{ij} , y_{ij} are the values of the brightness of the original image and obtained after processing image, respectively, at a point of the frame with Cartesian coordinates *i* (row number) and *j* (column number).

Element-by-element processing means that there is a functional one-to-one relationship between these brightness,

$$y_{ij} = f_{ij}(x_{ij}) , \tag{7.1}$$

allowing to determine the value of the output signal by the value of the original signal.

In the general case, as taken into account in (7.1), the form or parameters of the function f_{ij} , which is describing processing, depend on the current coordinates. Moreover, the processing is ununiform.

Most of the procedures, which are used in practice, use uniform element processing. In this case, the indices i and j in expression (7.1) may be absent. Then the dependence between the brightness of the original and output images is described by a function that is the same for all points of the frame:

$$y = f(x), \tag{7.2}$$

Consider in more detail the most common procedures for element-by-element processing.

Linear contrasting of images

The task of contrasting is to improve the harmonization of the dynamic range of the image and the screen on which the visualization is performed.

If for the digital representation of each sample of the image 1 byte (8 bits) of the memory is allocated, then the input or output signals can take one of 256 values (working range 0 ... 255; in this case, during visualization, the value 0 corresponds to the black level, and the value 255 - white level).

Suppose $x_{\min}$ and $x_{\max}$ are the minimum and maximum brightness of the original image, respectively. If these parameters or one of them differ significantly from the boundary values of the brightness range, then the visualizated picture looks like an unsaturated, uncomfortable. For example, such picture will be observed if $x_{\min} = 180$, $x_{\max} = 240$.

With linear contrasting, a linear element-by-element transformation of the form (7.3) is used:

$$y = ax + b , \tag{7.3}$$

where *a, b* are parameters that are determined by the desired values of the minimum $y_{\min}$ and maximum $y_{\max}$ output brightness. Solving the system of equations

$$\begin{cases} y_{\min} = ax_{\min} + b \\ y_{\max} = ax_{\max} + b \end{cases}$$

about the transformation parameters *a* and *b*, it is easy to bring (7.3) to the form:

$$y = \frac{x - x_{\min}}{x_{\max} - x_{\min}} (y_{\max} - y_{\min}) + y_{\min} \, . \tag{7.4}$$

Image solarization

With this type of processing, transformation (7.2) has the form:

$$y = kx(x_{\max} - x) \, , \tag{7.5}$$

where $x_{\max}$ is the maximum value of the original signal, *k* is a constant that allows you to control the dynamic range of the converted image.

The function describing this transformation is a quadratic parabola, its graph for $k = 1$ is shown in Fig. 7.1.

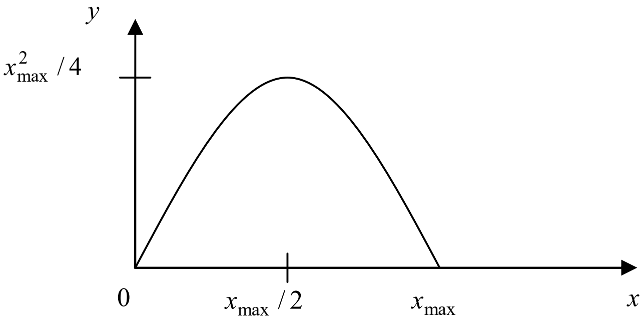


Figure 7.1 - Function describing solarization for $k = 1$

When $y_{\max} = x_{\max}$ the dynamic ranges of the images coincide, what can be achieved with $k = 4x_{\max}$. In order to match the dynamic ranges of the images $y_{\max} = x_{\max}$, normalization of dynamic range is performed. To do this, it is necessary to apply one of the contrasting methods, for example, linear contrasting.

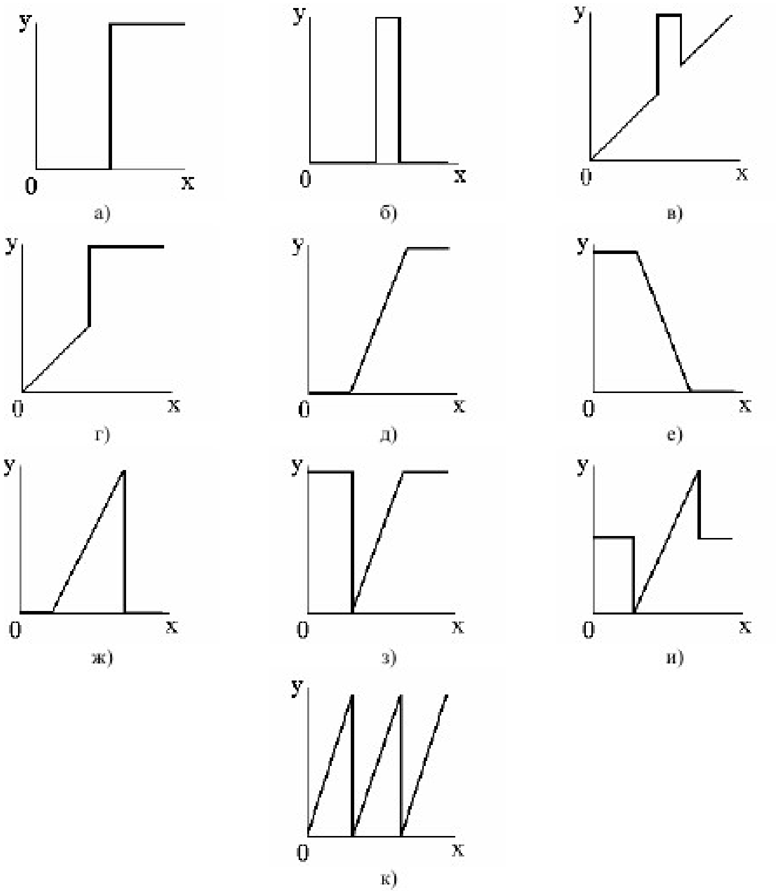
The meaning of solarization is that areas of the original image that have a white level or a brightness level close to it, after processing have a black level. At the same time, the black level in the original image are preserved. On the other hand, areas that have an average brightness level (gray level) in the input image will have white level in the output image.

Dissection of image

Dissection is a whole class of element-by-element transformations of images, examples of which are shown in Fig. 7.2.

The transformation with a threshold characteristic (Fig. 7.2a) is called *binarization* or *binary quantization*, in which a halftone image containing all brightness levels is converted into a binary image, the points of which have brightness $y = 0$ or $y = y_{\max}$.The main problem with such processing is the determination of the threshold x_0. The most reasonable for the mathematical description of the image is the application of the theory of probability, random processes and random fields. In this case, the determination of the optimal threshold of binary quantization is a statistical problem.

The essence of other transformations presented in fig. 7.2, it is easy to understand, considering their characteristics. For example, the transformation of fig. 7.2b performs a luminous slice of the image, highlighting those areas where the brightness corresponds to the selected interval. In this case, other areas are completely "extinguished" (have a brightness that corresponds to the level of black). By moving the selected interval on the luminance scale and changing its width, you can examine in detail the content of the picture.



а) б) в) г) д) е) ж) з) и) к)

Figure 7.2 - Examples of transformations used during dissection

Examples

Example № 1.

The image is given:

$$X = \begin{bmatrix} 34 & 9 & 40 & 20 \\ 17 & 21 & 23 & 45 \\ 8 & 42 & 22 & 0 \\ 7 & 24 & 22 & 14 \end{bmatrix}.$$

It is necessary to perform a linear contrasting with the parameters $y_{\max} = 30$, $y_{\min} = 1$.

The solution.

For calculation we will use expression (7.4). Find the maximum and minimum values of the two-dimensional signal X: $x_{\max} = 45$, $x_{\min} = 0$. Substituting the contrast parameters in (7.4), we obtain

$$y = \frac{x}{45}(30 - 1) + 1 = \frac{x}{45}29 + 1.$$

Then for the first line we have:

$$y_{00} = \frac{x_{00}}{45}29 + 1 = \frac{34}{45}29 + 1 = 22.911;\ y_{01} = \frac{x_{01}}{45}29 + 1 = \frac{9}{45}29 + 1 = 6.8;$$

$$y_{02} = \frac{x_{02}}{45}29 + 1 = \frac{40}{45}29 + 1 = 26.778;\ y_{03} = \frac{x_{03}}{45}29 + 1 = \frac{20}{45}29 + 1 = 13.889.$$

So, after calculating all the samples we get the next two-dimensional signal:

$$Y = \begin{bmatrix} 22.911 & 6.800 & 26.778 & 13.889 \\ 11.956 & 14.533 & 15.822 & 30.000 \\ 6.156 & 28.067 & 15.178 & 1.000 \\ 5.511 & 16.467 & 15.178 & 10.022 \end{bmatrix}.$$

Now you need to round the obtained values. As a result, we obtain the following image:

$$I = \begin{bmatrix} 23 & 7 & 27 & 14 \\ 12 & 15 & 16 & 30 \\ 6 & 28 & 15 & 1 \\ 61 & 16 & 15 & 10 \end{bmatrix}.$$

Example № 2.

The image is given:

$$X = \begin{bmatrix} 34 & 9 & 40 & 20 \\ 17 & 21 & 23 & 45 \\ 8 & 42 & 22 & 0 \\ 7 & 24 & 22 & 14 \end{bmatrix}.$$

It is necessary to perform solarization for $k = 1$.

The solution.

For calculation we will use expression (7.5). Find the maximum value of the two-dimensional signal X: $x_{\max} = 45$. Substituting the parameters in (7.5), we obtain $y = x(45 - x)$.

Then for the first line we have:

$$y_{00} = x_{00}(45 - x_{00}) = 34(45 - 34) = 374;$$

$$y_{01} = x_{01}(45 - x_{01}) = 9(45 - 9) = 324;$$

$$y_{02} = x_{02}(45 - x_{02}) = 40(45 - 40) = 200;$$

$$y_{03} = x_{03}(45 - x_{03}) = 20(45 - 20) = 500.$$

$$Y = \begin{bmatrix} 374 & 324 & 200 & 500 \\ 476 & 504 & 506 & 0 \\ 296 & 126 & 506 & 0 \\ 266 & 504 & 506 & 434 \end{bmatrix}.$$

Obviously, after solarization, a linear contrasting step is required to bring the resulting image to a given dynamic brightness range. Perform a linear contrasting with the same parameters as in the previous example. As a result, we get

$$I = \begin{bmatrix} 22.4348 & 19.5692 & 12.4625 & 29.6561 \\ 28.2806 & 29.8854 & 30.0000 & 1.0000 \\ 17.9644 & 8.2213 & 30.0000 & 1.0000 \\ 16.2451 & 29.8854 & 30.0000 & 25.8735 \end{bmatrix}.$$

After rounding we get the following resulting image:

$$J = \begin{bmatrix} 22 & 20 & 12 & 30 \\ 28 & 30 & 30 & 1 \\ 18 & 8 & 30 & 1 \\ 16 & 30 & 30 & 26 \end{bmatrix}.$$

It is easy to see that the pixels with the average brightness value of the input image (values from 20 to 24) received the maximum values after solarization. In this case, the pixels with both the maximum value of brightness ($x_{13} = 45$) and the minimum value of brightness ($x_{23} = 0$) after solarization received the minimum value ($j_{13} = 1$, $j_{23} = 1$).

Example № 3.

The image is given:

$$X = \begin{bmatrix} 34 & 9 & 40 & 20 \\ 17 & 21 & 23 & 45 \\ 8 & 42 & 22 & 0 \\ 7 & 24 & 22 & 14 \end{bmatrix}.$$

It is necessary to perform the dissection in accordance with the law shown in Fig. 7.9.

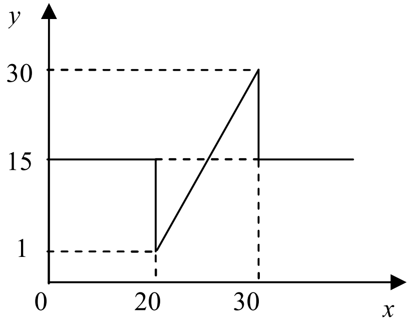


Figure 7.3 - Transformation for image dissection

The solution.

According to the law for dissection (Fig. 7.3) we write down the law of transformation

$$y = \begin{cases} 15, & \text{if } x < 20 \text{ or } x > 30; \\ ax + b, & \text{if } 20 \le x \le 30. \end{cases}$$

The law shows that for $20 < x < 30$ it is necessary to perform a linear contrasting (7.4), in this case

$x_{\max} = 30$, $y_{\min} = 1$, $y_{\max} = 30$.

Then

$$y = \begin{cases} 15, & \text{if } x < 20 \text{ or } x > 30; \\ \dfrac{x-20}{30-20}(30-1)+1, & \text{if } 20 \le x \le 30, \end{cases}$$

$$y = \begin{cases} 15, & \text{if } x < 20 \text{ or } x > 30; \\ 2.9(x-20)+1, & \text{if } 20 \le x \le 30, \end{cases}$$

Therefore, after the calculation we obtain the following two-dimensional signal:

$$Y = \begin{bmatrix} 15 & 15 & 15 & 1 \\ 15 & 3.9 & 9.7 & 15 \\ 15 & 15 & 6.8 & 15 \\ 15 & 12.6 & 6.8 & 15 \end{bmatrix}.$$

Now we need to round the obtained values. As a result, we obtain the following image

$$Y = \begin{bmatrix} 15 & 15 & 15 & 1 \\ 15 & 4 & 10 & 15 \\ 15 & 15 & 7 & 15 \\ 15 & 13 & 7 & 15 \end{bmatrix}.$$

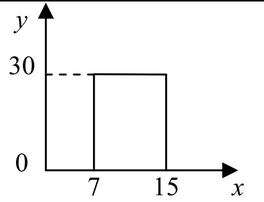
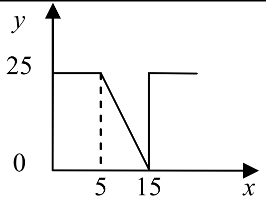
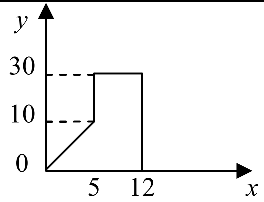
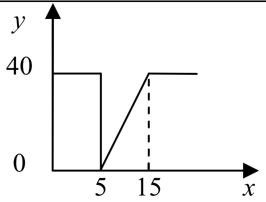
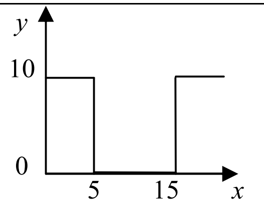
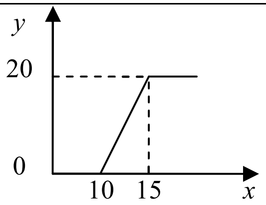
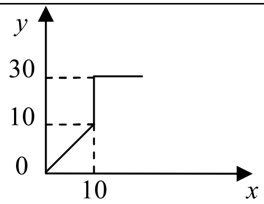
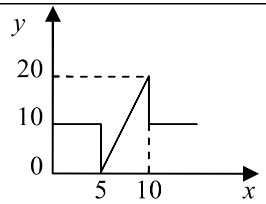
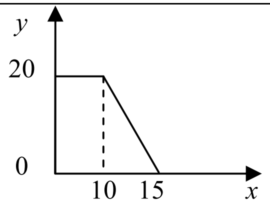
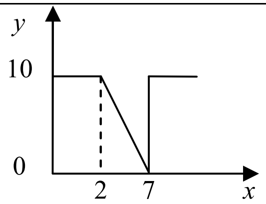
Tasks № 7 for independent work

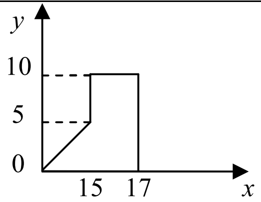
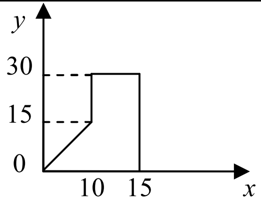
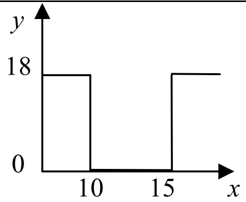
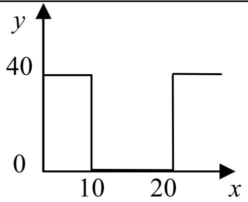
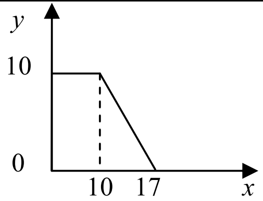
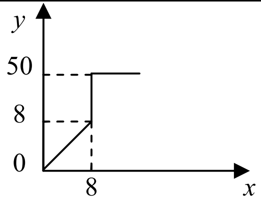
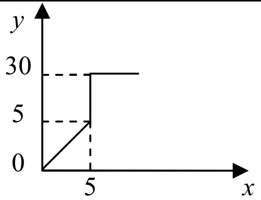
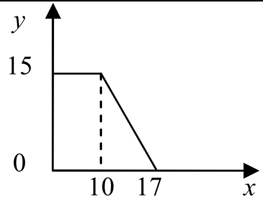
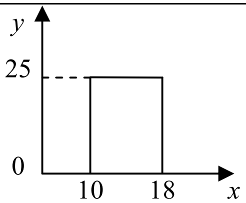
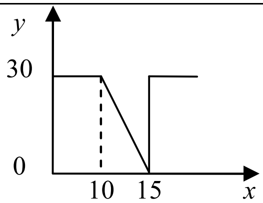
A two-dimensional signal is given

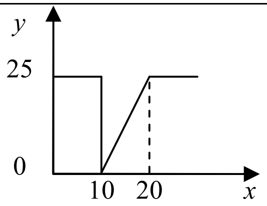
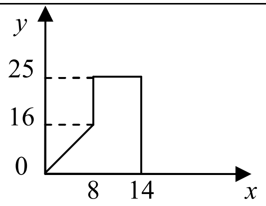
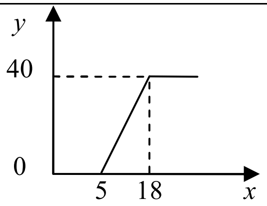
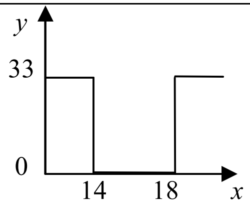
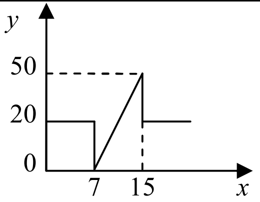
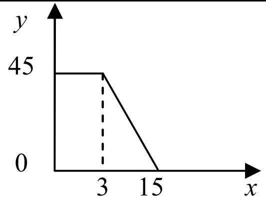
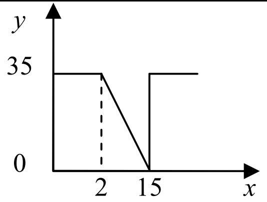
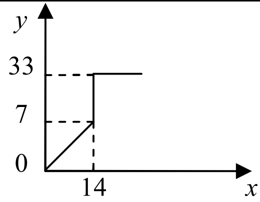
$$x = \begin{bmatrix} 14 & 16 & 20 & 3 & 13 \\ 11 & 19 & 5 & 0 & 6 \\ 9 & 10 & 5 & 18 & 9 \\ 14 & 18 & 18 & 4 & 1 \\ 12 & 3 & 15 & 6 & 20 \end{bmatrix}.$$

Perform two-dimensional signal dissection according to the specified law. Task options is given in table 7.1 (N - student's number in the journal).

Table 7.1 - Variants of laws for dissection

N	Dissection function	*N*	Dissection function
1		6	
2		7	
3		8	
4		9	
5		10	

N	Dissection function	*N*	Dissection function
11		16	
12		17	
13		18	
14		19	
15		20	

N	Dissection function	*N*	Dissection function
21		25	
22		26	
23		27	
24		28	

Practical task № 8.
Image filtering

Often, images generated by various information systems are distorted by noise.

During solving some problems of image processing, certain components of the image itself can act as noise. The decrease of noise is achieved by filtering.

During filtering, the brightness of each point of the original image, distorted by the noise, is replaced by some other brightness value, which is recognized as the least distorted by the noise.

8.1 Linear filters

Linear filters have a very simple mathematical description. Assume that the initial halftone image A is given, and denote the intensity of its pixels $A[x,y]$. The linear filter is determined by the valid function h (filter core) specified on the raster. The filtering itself is performed using a discrete convolution operation (weighted summation):

$$B[x,y] = h[i,j] ** A[x,y] = \sum_i \sum_j h[i,j] A[x-i, y-j]. \qquad (8.1)$$

The result is image B. Usually, the filter kernel $h[i,j] \neq 0$ is only in some neighborhood N of the point (0,0). Outside this neighborhood or very close to it $h[i,j]=0$ and it can be neglected. The value of each pixel $B[x,y]$ is determined by the pixels of the image A, which lie in the window N, centered at the point (x,y). The filter kernel defined on a rectangular neighborhood N can be viewed as a matrix $n \times m$, where n, m are odd numbers. During specifying a kernel as a matrix, it should be centered.

If the pixel is near the borders of the image, then the coordinates $A[x-i, y-j]$ for the defined (i, j) may correspond to non-existent pixels of A outside the image. There are several ways to solve this problem:

- do not filter for such pixels by trimming image B at the borders or applying the initial values of image A to their values;
- do not include the missing pixel in the summation, distributing its weight $h[i, j]$ evenly among other pixels around;
- determine the value of pixels outside the image by extrapolation;
- additional value of the pixel values behind the image boundaries, with the help of the mirror image continuation.

Vibration is the way to see a specific filter and image features.

Smoothing filters

The simplest *rectangular smoothing filter* of radius r is specified using a matrix of size $(2r+1)\times(2r+1)$, all values of which are equal $1/(2r+1)^2$, and the sum of the values is equal to one.

It is a two-dimensional analogue of a low-frequency one-dimensional U-shaped filter of moving average. During filtering with such a kernel, the pixel value is replaced by the average value of pixels in a square with a side $2r+1$ around it. Example of filter mask 3×3 is:

$$M_1^{low} = \frac{1}{9}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}.$$

One of the applications of filters is noise reduction

Noise reduction with a rectangular filter has a significant disadvantage: all the pixels in the filter mask, at any distance from the processed, affect the result with the same effect. Slightly better result is obtained during modifying the filter with increasing weight of the central point:

$$M_2^{low} = \frac{1}{10}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{pmatrix} \text{ або } M_3^{low} = \frac{1}{16}\begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}.$$

Contrast-enhancing filters

While smoothing filters reduce the local contrast of an image by blurring it, *contrast-enhancing filters* produce the reverse effect and are high spatial frequency filters.

The contrast-enhancing filter kernel at point (0,0) has a value greater than 1, with a total sum of values equal to 1.

For example, contrast-enhancing filters are filters with a kernel defined by matrices:

$$M_1^{contr} = \begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix}, \; M_2^{contr} = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{pmatrix},$$

$$M_3^{contr} = \begin{pmatrix} 1 & -2 & 1 \\ -2 & 5 & -2 \\ 1 & -2 & 1 \end{pmatrix}.$$

8.2 Non-linear filters

In digital image processing, nonlinear algorithms based on rank statistics are widely used to restore images distorted by various noise models.

Let us introduce the concept of an *M*-neighborhood of an image element $A[x, y]$, which is central for this neighborhood. In the simplest case, the *M*-neighborhood contains *N* pixels - the points falling into the filter mask, including (or not including) the central one.

The values of these *N* elements can be placed in a variation series $V(r)$, which is ranked in ascending (or descending) order. Certain moments of this series, for example, the average value of brightness m_N and dispersion d_N can be calculated.

The calculation of the filter output, which replaces the center sample, is performed by the formula:

$$B[x,y] = \alpha A[x,y] + (1-\alpha)m_N .$$

The value of the coefficient $\alpha = [0,1]$ is associated with a certain relationship with the statistics of samples in the filter window, for example:

$$\alpha = \frac{d_N}{d_N + kd_S},$$

where d_S is the dispersion of noise in the whole image or in the *S*-neighborhood for $S > M$ and $M \in S$; k is the constant of trust to the dispersion of *S*-neighborhoods

For $k = 1$ and $d_N \approx d_S$ we have $\alpha \approx 0.5$, therefore, $B[x,y] = (A[x,y] + m_N)/2$ i.e. the value of the central sample and the average value of the pixels of its *M*-neighborhood are added equally.

The most simple and common types of non-linear filters for image processing are *threshold* and *median filters*.

Threshold filtering

Threshold filtering is set, for example, as follows:

$$B[x,y] = \begin{cases} A[x,y], & A[x,y] - m_N \le p \\ m_N, & A[x,y] - m_N > p \end{cases}$$

The p value is the filtering threshold. If the value of the center point of the filter exceeds the average value of the samples m_N in its *M*-neighborhood by the value of the threshold, then it is replaced by the average value.

Median filtering

Median filtering is defined as follows:

$$B[x,y] = med\{M[x,y]\},$$

Those, the filtering result is the median value of the pixels of the neighborhood, the shape of which is determined by the filter mask.

Median filtering can effectively remove noise, which independently affects individual pixels. For example, such noises are “dead” pixels in digital shooting, “snow” noise when some of the pixels are replaced by pixels with maximum intensity, and so on.

Extremum filters

Extremum filters are defined according to the rules:

$$B_{\min}[x,y] = \min\{M[x,y]\},$$

$$B_{\max}[x,y] = \max\{M[x,y]\},$$

those, the filtering result is the minimum and maximum pixel values in the filter mask. Such filters are used, as a rule, for binary images.

Examples

Example № 1.

The image is given:

$$X = \begin{bmatrix} 34 & 9 & 40 & 20 \\ 17 & 21 & 23 & 45 \\ 8 & 42 & 22 & 0 \\ 7 & 24 & 22 & 14 \end{bmatrix}.$$

It is necessary to perform linear smoothing with the following masks:

а) $h = \frac{1}{9}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$;

б) $h = \frac{1}{10}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{pmatrix}$.

The solution.

Enter the notation:

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \end{bmatrix}, \ Y = \begin{bmatrix} y_{11} & y_{12} & y_{13} & y_{14} \\ y_{21} & y_{22} & y_{23} & y_{24} \\ y_{31} & y_{32} & y_{33} & y_{34} \\ y_{41} & y_{42} & y_{43} & y_{44} \end{bmatrix},$$

where X is the original image, Y is the filtered image.

Before starting the calculation it is necessary to choose the method of processing edge points. Let non-existent pixels outside the image be zero.

а)The calculation of the first two lines of the image taking into account the given filter mask $h = \frac{1}{9}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$ is performed as follows

$$y_{11} = \frac{x_{11} + x_{12} + x_{21} + x_{22}}{9} = (34 + 9 + 17 + 21)/9 = 9;$$

$$y_{12} = \frac{x_{11} + x_{12} + x_{13} + x_{21} + x_{22} + x_{23}}{9} = (34 + 9 + 40 + 17 + 21 + 23)/9 = 16;$$

$$y_{13} = \frac{x_{12} + x_{13} + x_{14} + x_{22} + x_{23} + x_{24}}{9} = (9 + 40 + 20 + 21 + 23 + 45)/9 = 17.56;$$

$$y_{14}=\frac{x_{13}+x_{14}+x_{23}+x_{24}}{9}=(40+20+23+45)/9=14.22\,;$$

$$y_{21}=\frac{x_{11}+x_{12}+x_{21}+x_{22}+x_{31}+x_{32}}{9}=(34+9+17+21+8+42)/9=14.56\,;$$

$$y_{22}=\frac{x_{11}+x_{12}+x_{13}+x_{21}+x_{22}+x_{23}+x_{31}+x_{32}+x_{33}}{9}=$$

$$=(34+9+40+17+21+23+8+42+22)/9=24;$$

$$y_{23}=\frac{x_{12}+x_{13}+x_{14}+x_{22}+x_{23}+x_{24}+x_{32}+x_{33}+x_{34}}{9}=$$

$$=(9+40+20+21+23+45+42+22+0)/9=24.67;$$

$$y_{24}=\frac{x_{13}+x_{14}+x_{23}+x_{24}+x_{33}+x_{34}}{9}=(40+20+23+45+22+0)/9=16.67.$$

Similarly, the calculation is performed for the next 2 lines. The result is the following filtered image:

$$Y=\begin{bmatrix} 9 & 16 & 17.56 & 14.22 \\ 14.56 & 24 & 24.67 & 16.67 \\ 13.22 & 20.67 & 23.67 & 14 \\ 9 & 13.89 & 13.78 & 6.44 \end{bmatrix}.$$

б) The calculation of the first two lines of the image taking into account the given filter mask $h=\frac{1}{10}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{pmatrix}$ is performed as follows:

$$y_{11}=(2x_{11}+x_{12}+x_{21}+x_{22})/10=(2\cdot 34+9+17+21)/10=11.5\,;$$

$$y_{12}=(x_{11}+2x_{12}+x_{13}+x_{21}+x_{22}+x_{23})/10=$$

$$=(34+2\cdot 9+40+17+21+23)/10=15.3;$$

$$y_{13} = (x_{12} + 2x_{13} + x_{14} + x_{22} + x_{23} + x_{24})/10 =$$
$$= (9 + 2 \cdot 40 + 20 + 21 + 23 + 45)/10 = 19.8;$$

$$y_{14} = (x_{13} + 2x_{14} + x_{23} + x_{24})/10 = (40 + 2 \cdot 20 + 23 + 45)/10 = 14.8 ;$$

$$y_{21} = (x_{11} + x_{12} + 2x_{21} + x_{22} + x_{31} + x_{32})/10 =$$
$$= (34 + 9 + 2 \cdot 17 + 21 + 8 + 42)/10 = 14.8;$$

$$y_{22} = (x_{11} + x_{12} + x_{13} + x_{21} + 2x_{22} + x_{23} + x_{31} + x_{32} + x_{33})/10 =$$
$$= (34 + 9 + 40 + 17 + 2 \cdot 21 + 23 + 8 + 42 + 22)/10 = 23.7;$$

$$y_{23} = (x_{12} + x_{13} + x_{14} + x_{22} + 2x_{23} + x_{24} + x_{32} + x_{33} + x_{34})/10 =$$
$$= (9 + 40 + 20 + 21 + 2 \cdot 23 + 45 + 42 + 22 + 0)/10 = 24.5;$$

$$y_{24} = (x_{13} + x_{14} + x_{23} + 2x_{24} + x_{33} + x_{34})/10 =$$
$$= (40 + 20 + 2 \cdot 23 + 45 + 22 + 0)/10 = 19.5.$$

Similarly, the calculation is performed for the next 2 lines. The result is the following filtered image:

$$Y = \begin{bmatrix} 11.5 & 15.3 & 19.8 & 14.8 \\ 14.8 & 23.7 & 24.5 & 19.5 \\ 12.7 & 22.8 & 23.5 & 12.6 \\ 8.8 & 14.9 & 14.6 & 7.2 \end{bmatrix}.$$

Example № 2.

The image is given:

$$X = \begin{bmatrix} 34 & 9 & 40 & 20 \\ 17 & 21 & 23 & 45 \\ 8 & 42 & 22 & 0 \\ 7 & 24 & 22 & 14 \end{bmatrix}.$$

It is necessary to perform high-frequency filtering with a filter mask

$$h = \begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix}.$$

The solution.

Enter the notation:

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \end{bmatrix}, \ Y = \begin{bmatrix} y_{11} & y_{12} & y_{13} & y_{14} \\ y_{21} & y_{22} & y_{23} & y_{24} \\ y_{31} & y_{32} & y_{33} & y_{34} \\ y_{41} & y_{42} & y_{43} & y_{44} \end{bmatrix},$$

where X is the original image, Y is the filtered image.

Before starting the calculation it is necessary to choose the method of processing edge points. Let non-existent pixels outside the image be zero.

The calculation of the first two lines of the image, taking into account the specified filter mask, is performed as follows:

$y_{11} = 5x_{11} - x_{12} - x_{21} = 5 \cdot 34 - 9 - 17 = 144$;

$y_{12} = -x_{11} + 5x_{12} - x_{13} - x_{22} = -34 + 5 \cdot 9 - 40 - 21 = -50$;

$y_{13} = -x_{12} + 5x_{13} - x_{14} - x_{23} = -9 + 5 \cdot 40 - 20 - 23 = 148$;

$y_{14} = -x_{13} + 5x_{14} - x_{24} = -40 + 5 \cdot 20 - 45 = 15$;

$y_{21} = -x_{11} + 5x_{21} - x_{22} - x_{31} = -34 + 5 \cdot 17 - 21 - 8 = 22$;

$y_{22} = -x_{12} - x_{21} + 5x_{22} - x_{23} - x_{32} = -9 - 17 + 5 \cdot 21 - 23 - 42 = 14$;

$y_{23} = -x_{13} - x_{22} + 5x_{23} - x_{24} - x_{33} = -40 - 21 + 5 \cdot 23 - 45 - 22 = -13$;

$y_{24} = -x_{14} - x_{23} + 5x_{24} - x_{34} = -20 - 23 + 5 \cdot 45 + 0 = 182$.

Similarly, the calculation is performed for the next 2 lines. The result is the following filtered image:

$$Y=\begin{bmatrix} 144 & -50 & 148 & 15 \\ 22 & 14 & -13 & 182 \\ -26 & 135 & 23 & -81 \\ 3 & 49 & 50 & 48 \end{bmatrix}.$$

Note: since the brightness level values (when the value is stored in 1 byte) cannot be out of range $[0{,}255]$, negative values are reset and values greater than 255 are set to 255.

Example № 3.

The image is given:

$$X=\begin{bmatrix} 16 & 19 & 1 & 18 & 3 \\ 0 & 19 & 1 & 6 & 11 \\ 11 & 9 & 6 & 7 & 17 \\ 7 & 9 & 1 & 15 & 8 \\ 4 & 13 & 0 & 14 & 15 \end{bmatrix}.$$

It is necessary to perform median filtering with linear aperture 3×3.

The solution.

Before starting the calculation it is necessary to choose the method of processing of border points. Let non-existent pixels outside the image be zero.

Then to calculate the pixel value of the output image y_{ij}, you must perform the following steps:

1) write in a separate array $\vec{z}$ the values of pixels that fall into the aperture of the filter;

2) sort the resulting array $\vec{z}$ in ascending order;

3) $y_{ij} = z_5$.

For sample y_{11} we will receive:

1) $\vec{z} = (0,0,0,0,16,19,0,0,19)$;

2) filtered array $\vec{z} = (0,0,0,0,\mathbf{0},0,16,19,19)$;

3) $y_{11} = 0$.

For sample y_{12} we will receive:

1) $\vec{z} = (0,0,0,16,19,1,0,19,1)$;

2) filtered array $\vec{z} = (0,0,0,0,\mathbf{1},1,16,19,19)$;

3) $y_{12} = 1$.

The result is the following filtered image:

$$Y = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 9 & 9 & 7 & 6 & 6 \\ 7 & 7 & 7 & 7 & 7 \\ 7 & 7 & 9 & 8 & 8 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix}.$$

Analysis of the result shows that the selected method of processing border points was unsuccessful. For example, clusters of large values in the upper left corner and lower right corner are replaced by small ones.

Let's use another way of processing border points. Duplicate the border points for the missing.

Then for sample y_{11} we will receive:

1) $\vec{z} = (16,16,19,16,16,19,0,0,19)$;

2) filtered array $\vec{z} = (0,0,16,16,\mathbf{16},16,19,19,19)$;

3) $y_{11} = 16$.

For sample y_{12} we will receive:

1) $\vec{z} = (16,19,1,16,19,1,0,19,1)$;

2) filtered array $\vec{z} = (0,1,1,1,\mathbf{16},16,19,19,19)$;

3) $y_{12} = 16$.

The result is the following filtered image:

$$Y = \begin{bmatrix} 16 & 16 & 18 & 3 & 6 \\ 11 & 9 & 7 & 6 & 11 \\ 9 & 7 & 7 & 7 & 11 \\ 9 & 7 & 9 & 8 & 15 \\ 7 & 4 & 13 & 14 & 15 \end{bmatrix}.$$

Tasks № 8 for independent work

1) Perform filtering the two-dimensional signal. Filter mask $h = \frac{1}{10}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$.

2) Perform median filtering of a two-dimensional signal with linear aperture 3×3.

Choose the method of processing border points. See input data in table 8.1 (N - student's number in the journal).

Table 8.1 - Input two-dimensional arrays

N	X	N	X	N	X
1	$\begin{bmatrix} 4 & 0 & 8 & 16 & 10 \\ 3 & 14 & 16 & 0 & 14 \\ 12 & 8 & 10 & 13 & 8 \\ 5 & 18 & 4 & 7 & 6 \\ 3 & 9 & 13 & 16 & 3 \end{bmatrix}$	5	$\begin{bmatrix} 19 & 4 & 8 & 13 & 4 \\ 19 & 12 & 14 & 4 & 12 \\ 15 & 6 & 5 & 16 & 12 \\ 8 & 19 & 8 & 12 & 7 \\ 9 & 14 & 18 & 2 & 11 \end{bmatrix}$	9	$\begin{bmatrix} 13 & 8 & 9 & 5 & 11 \\ 6 & 17 & 9 & 0 & 8 \\ 3 & 9 & 8 & 13 & 3 \\ 3 & 16 & 18 & 13 & 12 \\ 3 & 9 & 0 & 19 & 14 \end{bmatrix}$
2	$\begin{bmatrix} 3 & 13 & 9 & 13 & 14 \\ 13 & 7 & 17 & 6 & 6 \\ 6 & 17 & 16 & 5 & 16 \\ 10 & 17 & 12 & 6 & 11 \\ 3 & 11 & 16 & 10 & 7 \end{bmatrix}$	6	$\begin{bmatrix} 9 & 7 & 12 & 1 & 1 \\ 0 & 13 & 0 & 7 & 9 \\ 0 & 1 & 0 & 12 & 8 \\ 6 & 0 & 3 & 14 & 7 \\ 0 & 12 & 11 & 13 & 3 \end{bmatrix}$	10	$\begin{bmatrix} 7 & 18 & 3 & 10 & 12 \\ 0 & 16 & 18 & 18 & 13 \\ 8 & 7 & 11 & 6 & 7 \\ 15 & 12 & 12 & 13 & 8 \\ 15 & 14 & 4 & 7 & 13 \end{bmatrix}$
3	$\begin{bmatrix} 14 & 15 & 19 & 2 & 13 \\ 10 & 19 & 5 & 0 & 5 \\ 8 & 10 & 5 & 17 & 9 \\ 13 & 17 & 17 & 3 & 1 \\ 12 & 3 & 14 & 5 & 19 \end{bmatrix}$	7	$\begin{bmatrix} 13 & 2 & 5 & 4 & 3 \\ 13 & 9 & 17 & 4 & 16 \\ 14 & 14 & 4 & 0 & 3 \\ 9 & 17 & 16 & 1 & 3 \\ 11 & 5 & 18 & 12 & 19 \end{bmatrix}$	11	$\begin{bmatrix} 16 & 14 & 14 & 2 & 16 \\ 7 & 10 & 19 & 13 & 13 \\ 8 & 15 & 16 & 7 & 19 \\ 11 & 9 & 14 & 2 & 19 \\ 11 & 3 & 9 & 11 & 1 \end{bmatrix}$

N	X	N	X	N	X
4	$\begin{bmatrix} 11 & 4 & 4 & 11 & 8 \\ 8 & 11 & 7 & 15 & 6 \\ 10 & 15 & 15 & 1 & 17 \\ 6 & 10 & 13 & 12 & 0 \\ 8 & 12 & 9 & 1 & 15 \end{bmatrix}$	8	$\begin{bmatrix} 8 & 11 & 18 & 12 & 2 \\ 6 & 2 & 5 & 19 & 16 \\ 6 & 0 & 3 & 13 & 8 \\ 7 & 9 & 17 & 17 & 17 \\ 7 & 17 & 4 & 0 & 14 \end{bmatrix}$	12	$\begin{bmatrix} 7 & 11 & 8 & 18 & 14 \\ 10 & 14 & 12 & 12 & 8 \\ 5 & 19 & 16 & 5 & 19 \\ 11 & 15 & 1 & 17 & 1 \\ 0 & 14 & 18 & 10 & 11 \end{bmatrix}$
13	$\begin{bmatrix} 5 & 7 & 2 & 13 & 1 \\ 17 & 9 & 17 & 4 & 1 \\ 6 & 8 & 15 & 5 & 5 \\ 13 & 11 & 15 & 12 & 8 \\ 1 & 12 & 16 & 10 & 9 \end{bmatrix}$	16	$\begin{bmatrix} 12 & 19 & 6 & 19 & 8 \\ 9 & 13 & 12 & 9 & 6 \\ 16 & 19 & 10 & 8 & 11 \\ 13 & 18 & 8 & 11 & 12 \\ 7 & 9 & 14 & 15 & 14 \end{bmatrix}$	19	$\begin{bmatrix} 17 & 7 & 7 & 10 & 9 \\ 10 & 12 & 11 & 3 & 6 \\ 10 & 8 & 11 & 6 & 5 \\ 5 & 4 & 10 & 12 & 6 \\ 3 & 16 & 13 & 5 & 9 \end{bmatrix}$
14	$\begin{bmatrix} 18 & 3 & 16 & 7 & 19 \\ 11 & 16 & 12 & 3 & 2 \\ 6 & 19 & 14 & 16 & 17 \\ 9 & 11 & 1 & 16 & 15 \\ 11 & 0 & 8 & 9 & 8 \end{bmatrix}$	17	$\begin{bmatrix} 9 & 9 & 9 & 13 & 9 \\ 11 & 5 & 15 & 20 & 7 \\ 10 & 14 & 15 & 17 & 6 \\ 12 & 7 & 11 & 4 & 13 \\ 20 & 12 & 2 & 12 & 10 \end{bmatrix}$	20	$\begin{bmatrix} 12 & 11 & 8 & 10 & 14 \\ 10 & 11 & 12 & 4 & 17 \\ 16 & 13 & 11 & 20 & 7 \\ 10 & 13 & 8 & 13 & 12 \\ 3 & 17 & 2 & 5 & 10 \end{bmatrix}$
15	$\begin{bmatrix} 9 & 7 & 12 & 11 & 10 \\ 14 & 10 & 6 & 17 & 11 \\ 13 & 6 & 12 & 13 & 6 \\ 12 & 7 & 6 & 15 & 5 \\ 7 & 19 & 5 & 6 & 17 \end{bmatrix}$	18	$\begin{bmatrix} 7 & 10 & 17 & 9 & 12 \\ 12 & 18 & 11 & 8 & 18 \\ 2 & 9 & 14 & 7 & 9 \\ 10 & 0 & 17 & 11 & 12 \\ 14 & 6 & 1 & 20 & 8 \end{bmatrix}$	21	$\begin{bmatrix} 5 & 1 & 6 & 6 & 5 \\ 15 & 16 & 9 & 5 & 10 \\ 11 & 6 & 8 & 20 & 10 \\ 7 & 8 & 19 & 2 & 13 \\ 11 & 18 & 11 & 19 & 8 \end{bmatrix}$

Practical task № 9.
Image compression

Basics of wavelet transformations

The idea of wavelet compression of images is simple. First, a wavelet transformation is applied to the image, and then some coefficients are deleted from the transformed image. Coding can be applied to the remaining coefficients. The compressed image is reconstructed by decoding the coefficients, if necessary, and applying the inverse transform to the result. It is assumed that not too much information is lost in the process of deleting some of the transform coefficients.

Consider an image consisting of two pixels (x_1, x_2). These values can be replaced by the average value *a* and half-difference *d*:

$$a = (x_1 + x_2)/2\,,\ d = (x_1 - x_2)/2\,,$$

(coefficient $1/2$ is introduced into the definition of *d* for the convenience of notation). Note that we can write (x_1, x_2) in terms of (a, d):

$$x_1 = a + d\,,\ x_2 = a - d\,.$$

The "wavelet transformation" of the original sequential (x_1, x_2) is (a, d). In this form, information is not added or lost.

If the two values (x_1, x_2) are close to each other, then the difference *d* will be small, and the image (x_1, x_2) can be replaced by its approximation (a). The recovered image will be the image (a, a) with error from $(|x_1 - a|, |x_2 - a|) = (|d|, |d|)$. Because *d* is small, then the error will be small.

Consider a larger image (x_1, x_2, x_3, x_4).

Let's calculate the average values

$$a_{1,0} = (x_1 + x_2)/2, \quad a_{1,1} = (x_3 + x_4)/2$$

and the differences

$$d_{1,0} = (x_1 - x_2)/2, \quad d_{1,1} = (x_3 - x_4)/2.$$

Double subscripts indicate a multi-step process (this is its first step). Then a new representation of the original image

$$(a_{1,0}, a_{1,1}, d_{1,0}, d_{1,1}).$$

It contains exactly the same number of samples as the original one.

If you need to compress this image, then you need to pay attention to the magnitude of the values $d_{1,0}, d_{1,1}$ and decide whether they can be deleted without damage. In this case, we get a compressed image $(a_{1,0}, a_{1,1})$.

If it is necessary to compress the image more, then you can apply the same procedure to the remaining samples $(a_{1,0}, a_{1,1})$ and calculate the average value and difference again:

$$a_{0,0} = (a_{1,0} + a_{1,1})/2, \; d_{0,0} = (a_{1,0} - a_{1,1})/2.$$

If the difference $d_{0,0}$ is small enough, then you can replace the entire original image (x_1, x_2, x_3, x_4) with an image $(a_{0,0})$ consisting only of one pixel. In this case, the value $(a_{0,0})$ is the average value of all samples of the original image:

$$a_{0,0} = (a_{1,0} + a_{1,1})/2 = ((x_1 + x_2)/2 + (x_3 + x_4)/2)/2 = (x_1 + x_2 + x_3 + x_4)/4.$$

If the original image is uniformly gray (that is, all x are equal to the same value), then it can be compressed, replacing it with a single value which is equal to that grayscale value.

The value $a_{0,0}$ represents the most approximate level of information about this image, i.e. information at the lowest resolution or at the coarsest scale.

The values $a_{1,0}$, $a_{1,1}$ taken together represent information at the next higher resolution level, or at the next better scale. Image $(a_{1,0}, a_{1,1})$ can be restored from $(a_{0,0}, d_{0,0})$ using the above procedure.

The original sample values (x_1, x_2, x_3, x_4) represent the highest resolution or best scale possible for this image. These values can be restored from $(a_{1,0}, a_{1,1}, d_{1,0}, d_{1,1})$.

Because $a_{1,0}$, $a_{1,1}$ can be obtained from $a_{0,0}$, $d_{0,0}$, then it is possible to restore the pixel values of the original image from the total average value $a_{0,0}$ and differences $d_{0,0}$, $d_{1,0}$, $d_{1,1}$. Thus, the sequence

$$(a_{0,0}, d_{0,0}, d_{1,0}, d_{1,1}) \tag{9.1}$$

is an alternative representation of the original image and consists of a total average value and difference values representing two different levels of detail.

Sequence (9.1) is a **wavelet transformation** of the original sequence (x_1, x_2, x_3, x_4) . However, there are more variants for compression. If $d_{1,0}$ and $d_{1,1}$ are too large to ignore, then you can try to exclude the next level of detail, namely $d_{0,0}$.

If we consider a larger image, we can continue the process of averaging and highlighting details at a coarser level of resolution.

Two-dimensional wavelet transformation

Previously, wavelet transformations were considered only for one-dimensional sequences. It is possible to extend the idea of wavelet transformation to larger sequences. The first way is to first convert the rows of the image and then convert the columns of the image with the converted rows.

Consider a two-dimensional analogue of a simple 4-element sequence. Suppose we have a 4×4 image

$$\begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} \\ x_{2,1} & x_{2,2} & x_{2,3} & x_{2,4} \\ x_{3,1} & x_{3,2} & x_{3,3} & x_{3,4} \\ x_{4,1} & x_{4,2} & x_{4,3} & x_{4,4} \end{bmatrix}.$$

One way to get a two-dimensional wavelet transformation of an image of size $2^n \times 2^n$ is to first apply a one-dimensional wavelet transformation to each of the 2^n rows, and then apply a one-dimensional wavelet transformation to each of the 2^n columns.

We can think of averaging and detailing operations using Haar wavelets as applying low-pass and high-pass filters. A low-pass filter allows information transmitted at low frequencies (i.e., a small amount of details) to pass through, while blocking information transmitted at high frequencies (i.e., containing a large amount of details), and a high-pass filter does the opposite.

Scheme of fig. 9.1 shows what a two-dimensional wavelet transformation looks like.

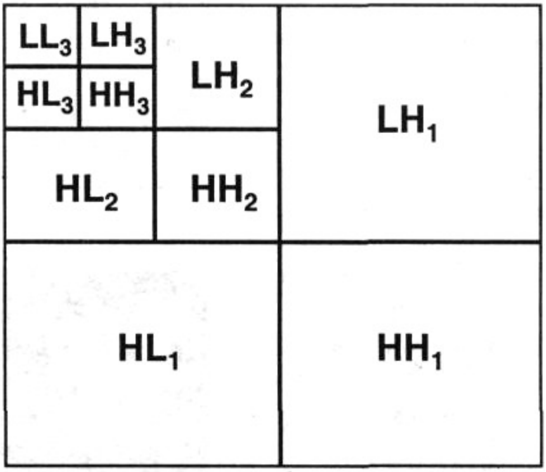


Figure 9.1 - Scheme of two-dimensional wavelet transformation

Blok HH_1 – the result of the HPF (H) action on the lines, and then its action on the resulting image columns.

Blok LH_1 – the result of the action of the LPF (L) on the image columns that have already passed through the HPF.

Blok HL_1 – the result of low-frequency processing of rows, followed by high-frequency processing of columns.

The upper left corner of the diagram is subdivided into smaller blocks, indicating that the wavelet transformation operation sequentially affects less amount of coefficients, up to the last coefficient in the upper left corner, to which only the LPF was applied.

Examples

Example № 1.

A sequence $x = [1,3,5,7]$ is given. Calculate forward wavelet transformation

The solution

1-st step:

Let's calculate the average values and the differences

$(1+3)/2 = 2$, $(5+7)/2 = 6$, $(1-3)/2 = -1$, $(5-7)/2 = -1$ $\Rightarrow$ $[2,6,-1,-1]$.

2-nd step:

$(2+6)/2 = 4$, $(2-6)/2 = -2$ $\Rightarrow$ $[4,-2,-1,-1]$

Result: $y = [4,-2,-1,-1]$

Example № 2.

A wavelet transformation $y = [4,-2,-1,-1]$ is given. Calculate original sequence.

The solution

1-st step: $4-2=2$, $4+2=6$ $\Rightarrow [2,6,-1,-1]$.

2-nd step: $2-1=1$, $2+1=3$, $6-1=5$, $6+1=7$ $\Rightarrow$ $[1,3,5,7]$

Result : $x = [1,3,5,7]$

Example № 3.

The original matrix is given. Calculate two-dimensional forward wavelet transformation

$$I = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{bmatrix}$$

1-st step

Transformation by strings

$$I=\begin{bmatrix}1.5 & 3.5 & -0.5 & -0.5\\ 1.5 & 3.5 & -0.5 & -0.5\\ 1.5 & 3.5 & -0.5 & -0.5\\ 1.5 & 3.5 & -0.5 & -0.5\end{bmatrix}$$

Transformation by columns

$$I=\begin{bmatrix}1.5 & 3.5 & -0.5 & -0.5\\ 1.5 & 3.5 & -0.5 & -0.5\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}$$

2-nd step (performed only for the upper left corner

Transformation by strings

$$I=\left[\begin{array}{cc:cc}2.5 & -1 & -0.5 & -0.5\\ 2.5 & -1 & -0.5 & -0.5\\ \hdashline 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{array}\right]$$

Transformation by columns

$$I=\left[\begin{array}{cc:cc}2.5 & -1 & -0.5 & -0.5\\ 0 & 0 & -0.5 & -0.5\\ \hdashline 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{array}\right]$$

Tasks № 9 for independent work

Calculate the wavelet transformation:

1) for a one-dimensional sequence

The initial data see in table 9.1 (N - student number in the journal).

Table 9.1 - Task options (N - student number in the journal)

N	Discrete signal (for task 1)
1	$x[n] = (8;9;4;1;2;0;1;4)$
2	$x[n] = (0;4;6;7;2;4;4;6)$
3	$x[n] = (1;2;4;8;6;3;5;9)$
4	$x[n] = (7;4;3;9;1;4;4;5)$
5	$x[n] = (7;0;1;2;8;9;2;6)$
6	$x[n] = (8;1;5;4;2;8;8;4)$
7	$x[n] = (8;5;0;6;3;3;9;6)$
8	$x[n] = (2;9;5;9;5;6;1;8)$
9	$x[n] = (0;1;3;3;6;1;0;4)$
10	$x[n] = (4;0;8;1;8;5;4;2)$
11	$x[n] = (1;7;2;1;3;3;2;9)$
12	$x[n] = (1;2;4;6;2;9;8;3)$
13	$x[n] = (0;6;3;6;3;8;8;1)$
14	$x[n] = (6;9;3;5;2;3;5;8)$
15	$x[n] = (4;7;1;2;9;1;5;5)$
16	$x[n] = (6;5;6;1;0;0;1;3)$

2) for a two-dimensional signal

The input data see in table 9.2 (N - student number in the journal).

Table 9.2 - Input two-dimensional arrays

N	X	N	X	N	X
1	$\begin{bmatrix}4 & 0 & 8 & 16\\3 & 14 & 16 & 0\\12 & 8 & 10 & 13\\5 & 18 & 4 & 7\end{bmatrix}$	5	$\begin{bmatrix}19 & 4 & 8 & 13\\19 & 12 & 14 & 4\\15 & 6 & 5 & 16\\8 & 19 & 8 & 12\end{bmatrix}$	9	$\begin{bmatrix}13 & 8 & 9 & 5\\6 & 17 & 9 & 0\\3 & 9 & 8 & 13\\3 & 16 & 18 & 13\end{bmatrix}$
2	$\begin{bmatrix}3 & 13 & 9 & 13\\13 & 7 & 17 & 6\\6 & 17 & 16 & 5\\10 & 17 & 12 & 6\end{bmatrix}$	6	$\begin{bmatrix}9 & 7 & 12 & 1\\0 & 13 & 0 & 7\\0 & 1 & 0 & 12\\6 & 0 & 3 & 14\end{bmatrix}$	10	$\begin{bmatrix}7 & 18 & 3 & 10\\0 & 16 & 18 & 18\\8 & 7 & 11 & 6\\15 & 12 & 12 & 13\end{bmatrix}$
3	$\begin{bmatrix}14 & 15 & 19 & 2\\10 & 19 & 5 & 0\\8 & 10 & 5 & 17\\13 & 17 & 17 & 3\end{bmatrix}$	7	$\begin{bmatrix}13 & 2 & 5 & 4\\13 & 9 & 17 & 4\\14 & 14 & 4 & 0\\9 & 17 & 16 & 1\end{bmatrix}$	11	$\begin{bmatrix}16 & 14 & 14 & 2\\7 & 10 & 19 & 13\\8 & 15 & 16 & 7\\11 & 9 & 14 & 2\end{bmatrix}$
4	$\begin{bmatrix}11 & 4 & 4 & 11\\8 & 11 & 7 & 15\\10 & 15 & 15 & 1\\6 & 10 & 13 & 12\end{bmatrix}$	8	$\begin{bmatrix}8 & 11 & 18 & 12\\6 & 2 & 5 & 19\\6 & 0 & 3 & 13\\7 & 9 & 17 & 17\end{bmatrix}$	12	$\begin{bmatrix}7 & 11 & 8 & 18\\10 & 14 & 12 & 12\\5 & 19 & 16 & 5\\11 & 15 & 1 & 17\end{bmatrix}$
13	$\begin{bmatrix}5 & 7 & 2 & 13\\17 & 9 & 17 & 4\\6 & 8 & 15 & 5\\13 & 11 & 15 & 12\end{bmatrix}$	16	$\begin{bmatrix}12 & 19 & 6 & 19\\9 & 13 & 12 & 9\\16 & 19 & 10 & 8\\13 & 18 & 8 & 11\end{bmatrix}$	19	$\begin{bmatrix}17 & 7 & 7 & 10\\10 & 12 & 11 & 3\\10 & 8 & 11 & 6\\5 & 4 & 10 & 12\end{bmatrix}$
14	$\begin{bmatrix}18 & 3 & 16 & 7\\11 & 16 & 12 & 3\\6 & 19 & 14 & 16\\9 & 11 & 1 & 16\end{bmatrix}$	17	$\begin{bmatrix}9 & 9 & 9 & 13\\11 & 5 & 15 & 20\\10 & 14 & 15 & 17\\12 & 7 & 11 & 4\end{bmatrix}$	20	$\begin{bmatrix}12 & 11 & 8 & 10\\10 & 11 & 12 & 4\\16 & 13 & 11 & 20\\10 & 13 & 8 & 13\end{bmatrix}$
15	$\begin{bmatrix}9 & 7 & 12 & 11\\14 & 10 & 6 & 17\\13 & 6 & 12 & 13\\12 & 7 & 6 & 15\end{bmatrix}$	18	$\begin{bmatrix}7 & 10 & 17 & 9\\12 & 18 & 11 & 8\\2 & 9 & 14 & 7\\10 & 0 & 17 & 11\end{bmatrix}$	21	$\begin{bmatrix}5 & 1 & 6 & 6\\15 & 16 & 9 & 5\\11 & 6 & 8 & 20\\7 & 8 & 19 & 2\end{bmatrix}$

RECOMMENDED BOOKS

1. Gérard Blanchet, Maurice Charbit. Digital Signal and Image Processing Using MATLAB, Volume 1. – Wiley-ISTE, 2014 – 512 P.

2. Gérard Blanchet, Maurice Charbit. Digital Signal and Image Processing Using MATLAB, Volume 2. – Wiley-ISTE, 2015 – 276 P.

3. D. Sundararajan. Digital Image Processing. A Signal Processing and Algorithmic Approach. – Springer, 2017 – 468 P.

4. S. Allen Broughton, Kurt Bryan. Discrete Fourier Analysis and Wavelets. Applications to Signal and Image Processing. – Wiley, 2018 – 464 P.

5. Dietrich Schlichthärle. Digital Filters. Basics and Design. – Springer, 2011 – 527 P.

6. Digital Filters Design for Signal and Image Processing / edited by Mohamed Najim – Wiley, 2013 – 369 P.

7. Leland B. Jackson. Digital Filters and Signal Processing. With MATLAB® Exercises. – Springer, 2010 – 502 P.

8. Mohand Mokhtari, Michel Marie. Engineering Applications of MATLAB® 5.3 and SIMULINK® 3. – Springer London, 2014 – 538 P.

9. Lakshman Prasad, S. Sitharama Iyengar. Wavelet Analysis with Applications to Image Processing. – CRC Press, 2020 – 304 P.

10. Stephen J. Chapman. Essentials of MATLAB Programming. – Cengage Learning, 2018 – 492 P.

Educational edition

POVOROZNYUK Oksana
FILATOVA Anna

SIGNAL AND IMAGE PROCESSING

Methodological instructions for practical work

The work was recommended to the publication by prof. M.Y. Zapolovsky

In the author's edition

Plan 2024, item 139