Компьютерное Зрение Лекция №2, осень 2024

Обработка сигналов

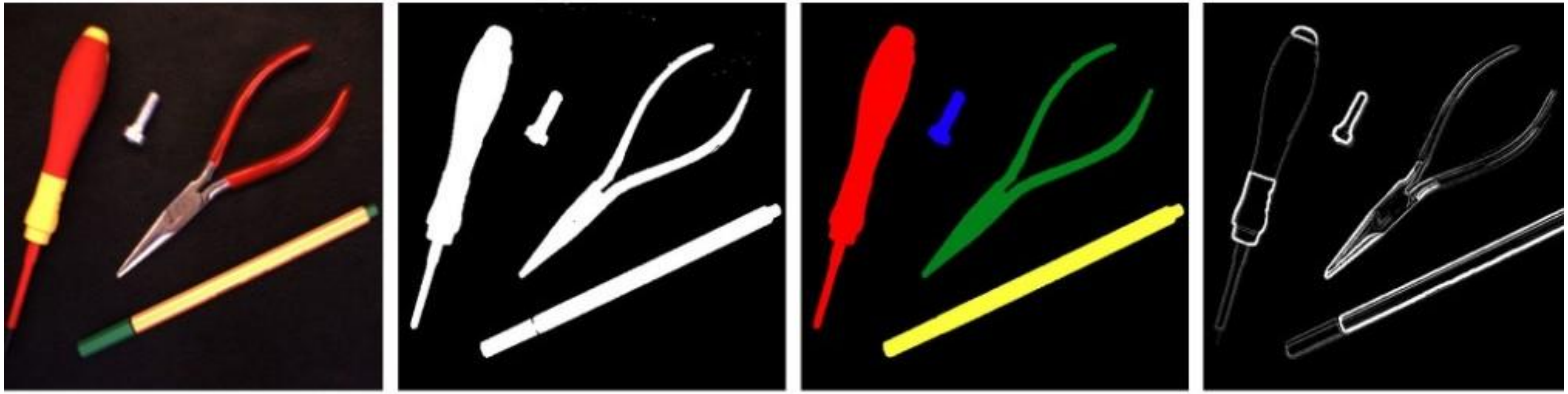
Мотивация к обработке изображений

De-noising

Super-resolution

In-painting

Мотивация к обработке изображений



Бинаризация

Выделение компонент связности

Выделение краев

План лекции

- Представление изображения в частотной области. Преобразование Фурье
- Системы и фильтры
- Свертки

Изображение как дискретная функция

Декартовые
координа

$$f[n,m] = \begin{bmatrix} \ddots & & \vdots & & \\ & f[-1,1] & f[0,1] & f[1,1] & \\ \dots & f[-1,0] & \underline{f[0,0]} & f[1,0] & \dots \\ & f[-1,-1] & f[0,-1] & f[1,-1] & \\ & & \vdots & & \ddots \end{bmatrix}$$

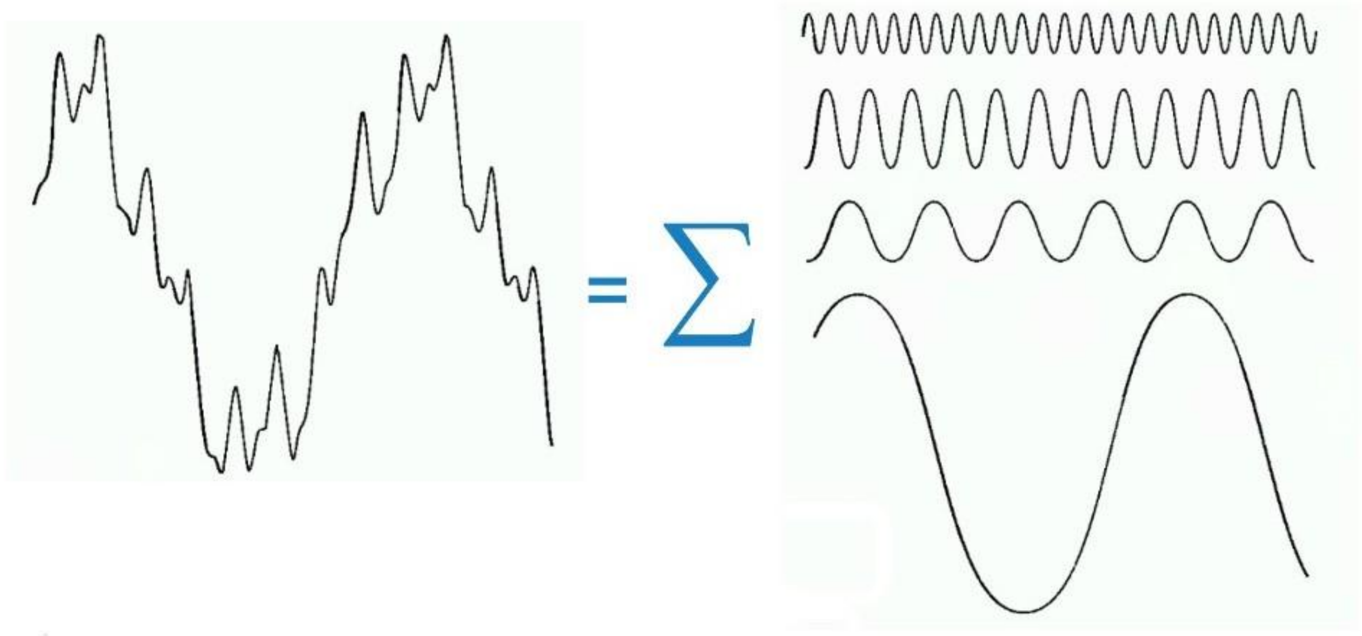
Изображение как функция *f* от R^2 до R^M:

- *f*(*x, y*) дает интенсивность в позиции (*x, y*)
- Определяется через прямоугольник, с конечным диапазоном:

  f: [*a*,*b*] x [*c*,*d*] [0,255]

Ряд Фурье

Периодический сигнал может быть представлен в виде суммы



Ряд и преобразование Фурье

Ряд Фурье́ — представление функции f с периодом τ в виде ряда

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{+\infty} A_k \cos\left(k\frac{2\pi}{\tau}x + \theta_k\right)$$

Этот ряд может быть также записан в виде

$$f(x) = \sum_{k=-\infty}^{+\infty} \hat{f}_k e^{ik\frac{2\pi}{\tau}x},$$

где

A_k — амплитуда k-го гармонического колебания,

$k\frac{2\pi}{\tau} = k\omega$ — круговая частота гармонического колебания,

θ_k — начальная фаза k-го колебания,

$\hat{f}_k$ — k-я комплексная амплитуда

Преобразование Фурье

Прямое

$$\hat{f}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x)e^{-ix\omega}\,dx.$$

Обратное

$$f(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \hat{f}(\omega)e^{ix\omega}\,d\omega$$

Преобразование Фурье для двумерного случая

Прямое преобразовани
е

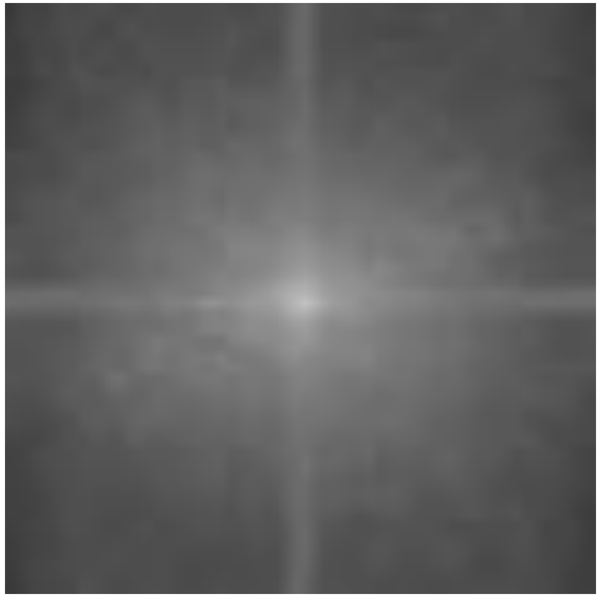
$$F(k,l) = \sum_{p=0}^{N-1} \sum_{q=0}^{N-1} f(p,q) e^{-2i\pi\left(\frac{kp}{N} + \frac{lq}{N}\right)}$$

Обратное преобразовани
е

$$f(p,q) = \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} F(k,l) e^{-2i\pi\left(\frac{kp}{N} + \frac{lq}{N}\right)}$$

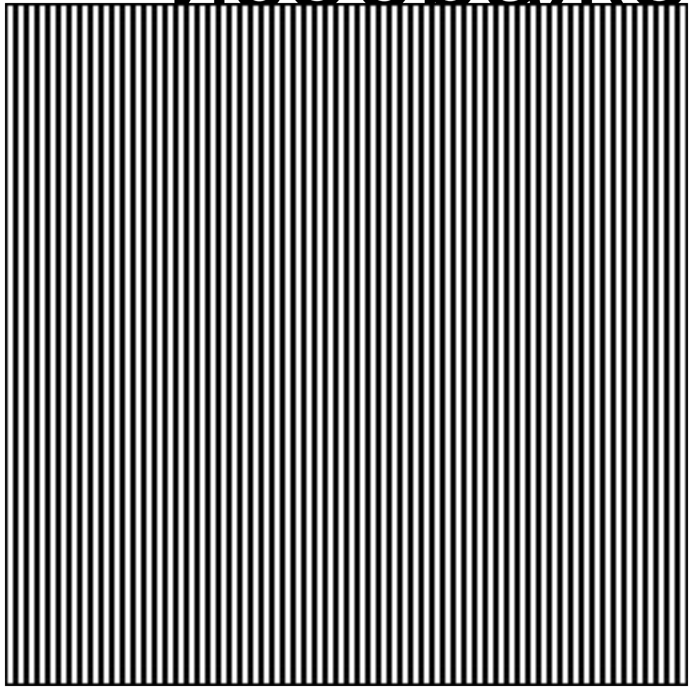
Преобразования Фурье для изображений

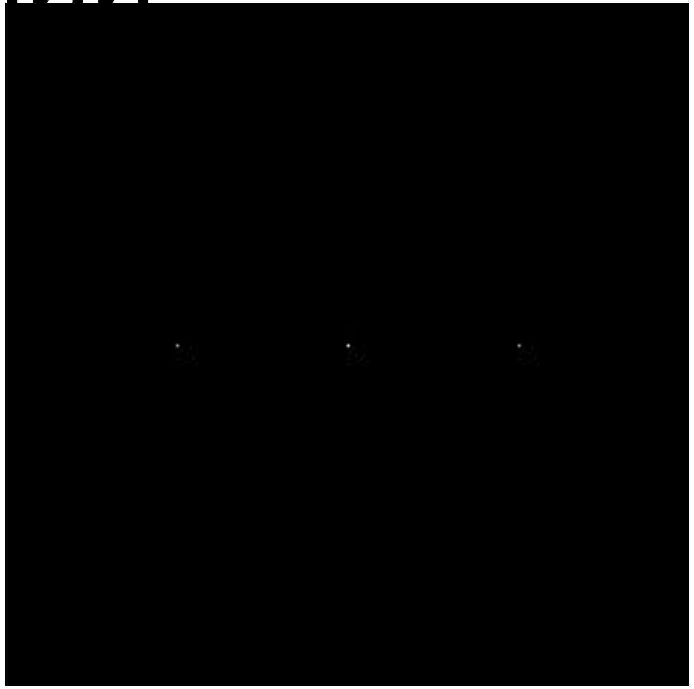
Исходное изображение



В частотном спектре

Преобразования Фурье для изображений





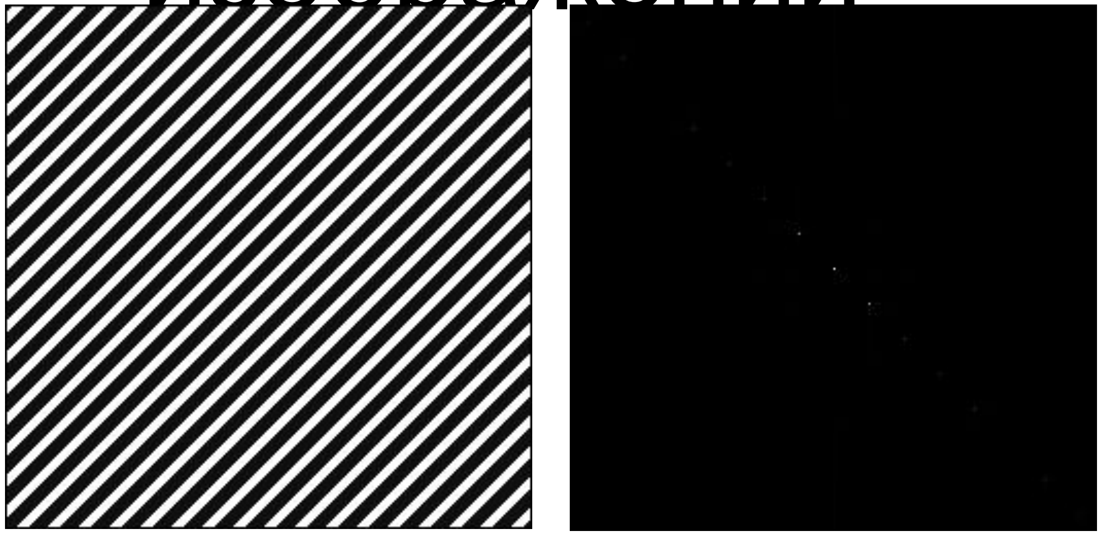
Расстояние точек до центра можно объяснить следующим образом: максимальная частота, которая может быть представлена в пространственной области, равна двум парам полос шириной в пиксель (одна белая, одна черная).

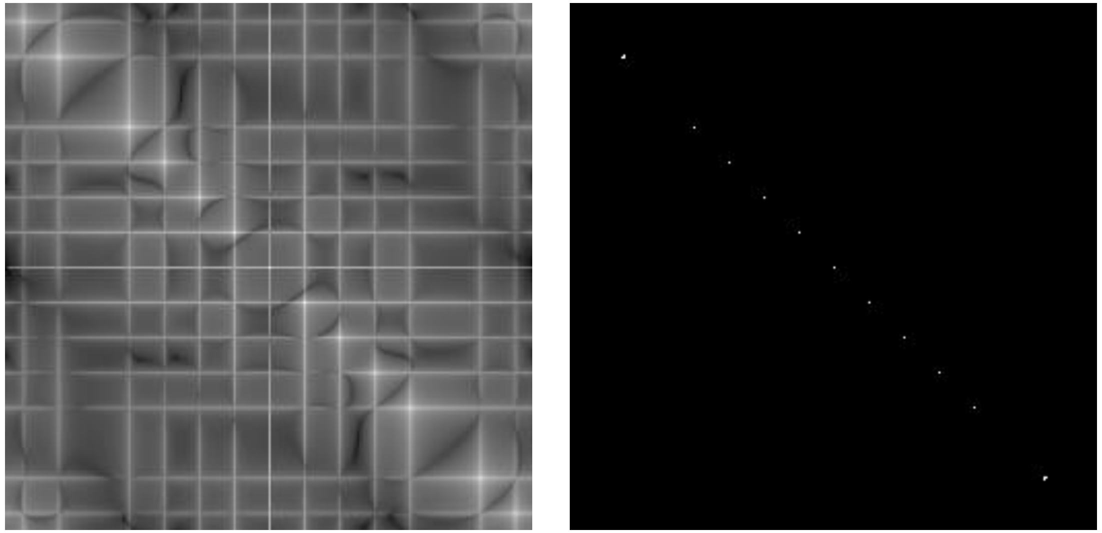
$$f_{max} = \frac{1}{2 \text{ pixels}}$$

Полосы шириной в два пикселя на приведенном выше изображении представляют

$$f = \frac{1}{4 \text{ pixels}} = \frac{f_{max}}{2} \qquad D = \frac{W}{N}$$

Преобразования Фурье для изображений

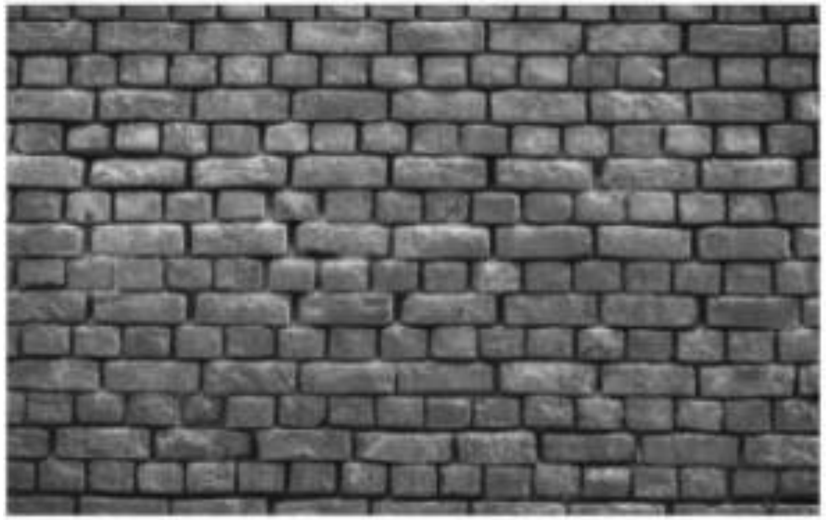


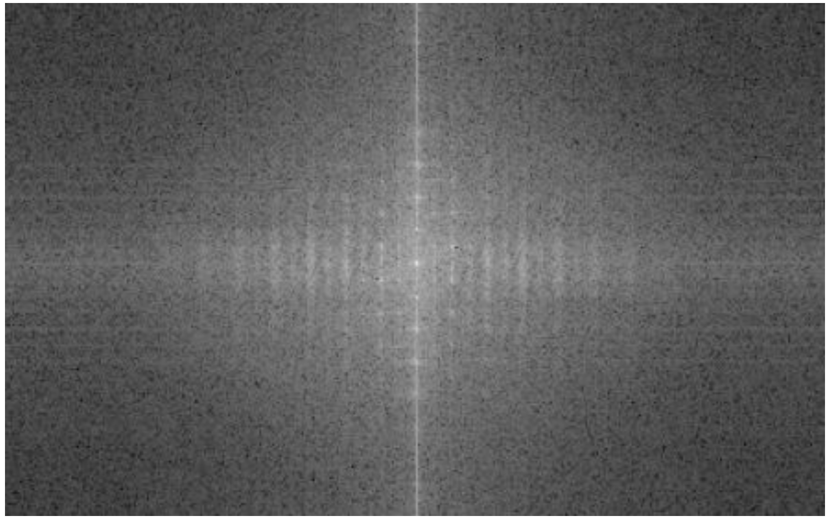


отображаются все частоты, величина которых составляет не менее 5% от основного пика

Все представленные частоты кратны базовой частоте полос на изображении пространственной области.

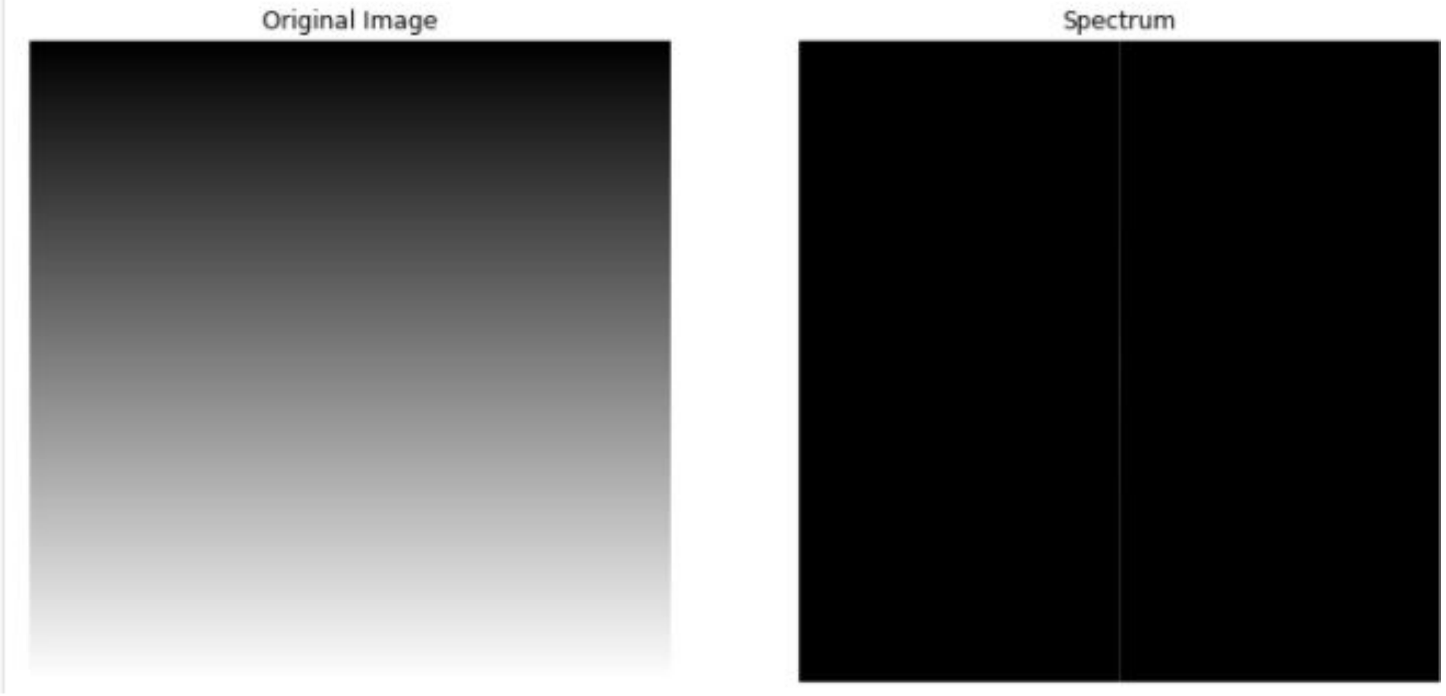
Преобразования Фурье для изображений





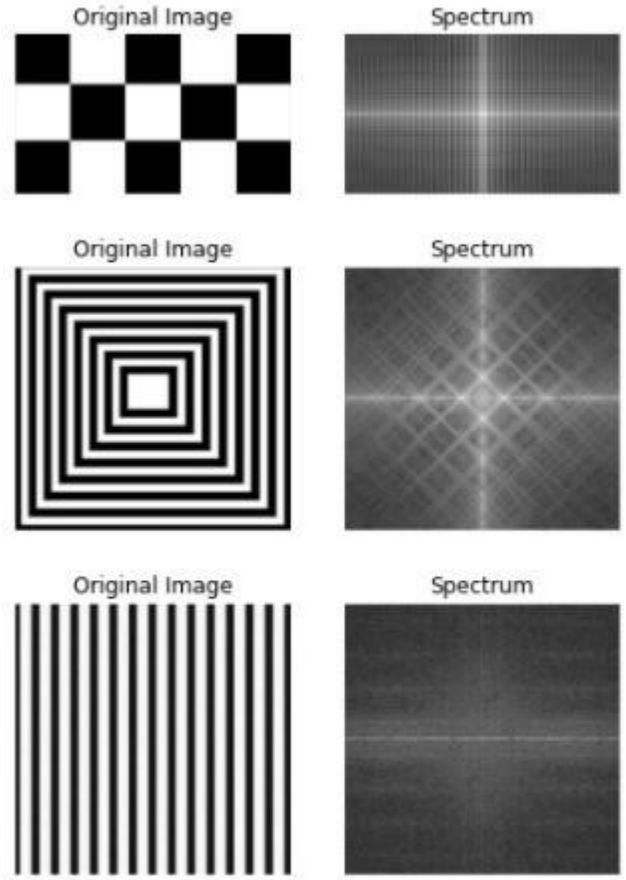
«Высокие» частоты: область с сильными и частыми перепадами значений пикселей

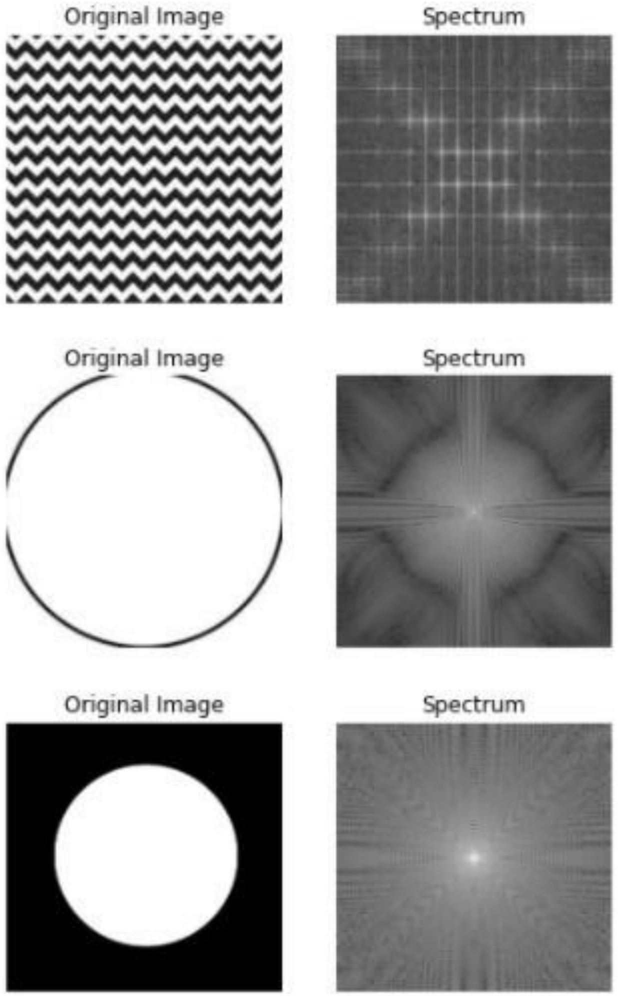
Преобразования Фурье для изображений



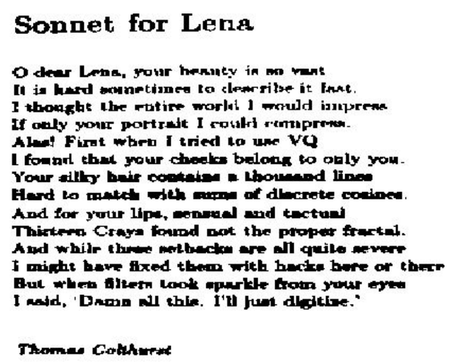
«Низкие» частоты: области с слабыми и редкими перепадами значений пикселей

Интерпретация спектра изображения



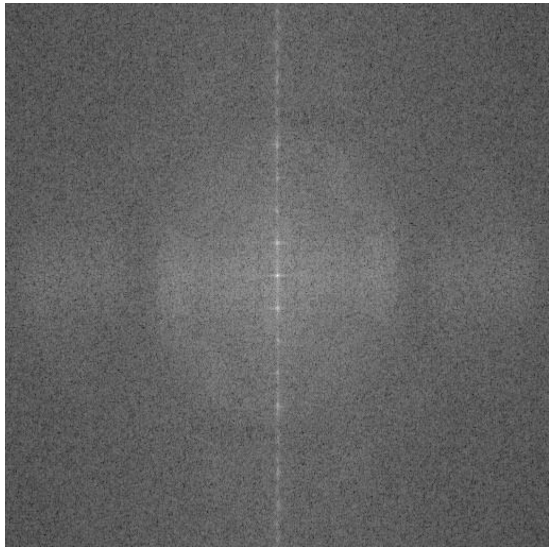


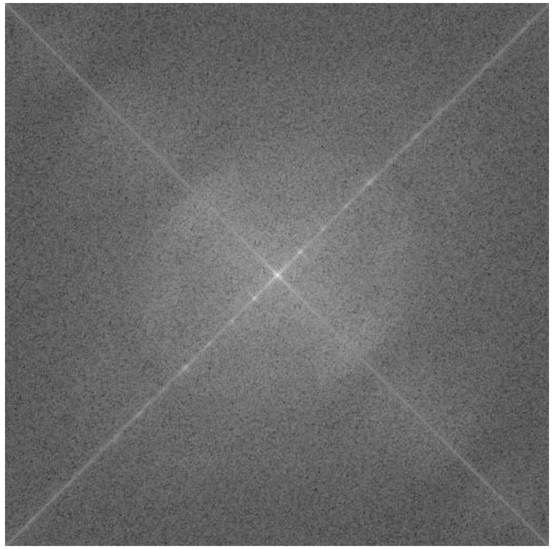
Интерпретация спектра изображения


Sonnet for Lena

O dear Lena, your beauty is so vast
It is hard sometimes to describe it fast.
I thought the entire world I would impress
If only your portrait I could compress.
Alas! First when I tried to use VQ
I found that your cheeks belong to only you.
Your silky hair contains a thousand lines
Hard to match with sums of discrete cosines.
And for your lips, sensual and tactual
Thirteen Crays found not the proper fractal.
And while these setbacks are all quite severe
I might have fixed them with hacks here or there
But when filters took sparkle from your eyes
I said, 'Damn all this. I'll just digitize.'

Thomas Colthurst





План лекции

- Представление изображения в частотной области. Преобразование Фурье
- **Системы и фильтры**
- Свертки

Фильтры

Фильтрация – формирование нового изображения, значения пикселей которого трансформируются из исходных значений пикселей.

Мотивация:

- Выделить полезную информацию
- Изменить или улучшить свойства полезных признаков на изображении

Пример фильтра №1: Размытие

Original image

Smoothed image

Пример фильтра №1: Размытие

2D DS moving average over a 3 × 3 window of neighborhood

$$g[n,m] = \frac{1}{9}\sum_{k=n-1}^{n+1}\sum_{l=m-1}^{m+1} f[k,l]$$

$$= \frac{1}{9}\sum_{k=-1}^{1}\sum_{l=-1}^{1} f[n-k, m-l]$$

h

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Пример фильтра №1: Размытие

$\mathrm{f}[n,m]$ $\mathrm{g}[n,m]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

	0								

Пример фильтра №1: Размытие

$\mathrm{f}[n, m]$ $\mathrm{g}[n, m]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

	0	10							

Пример фильтра №1: Размытие

$f[n, m]$ $g[n, m]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

	0	10 20							

Пример фильтра №1: Размытие

$f[n,m]$ $g[n,m]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

	0	10	20 30						

Пример фильтра №1: Размытие

$\mathrm{f}[n, m]$ $\mathrm{g}[n, m]$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

	0	10	20	30	30	30	20	10	
	0	20	40	60	60	60	40	20	
	0	30	60	90	90	90	60	30	
	0	30	50	80	80	90	60	30	
	0	30	50	80	80	90	60	30	
	0	20	30	50	50	60	40	20	
	10	20	30	30	30	30	20	10	
	10	10	10	0	0	0	0	0	

Пример фильтра №1: Размытие

Подводя итог:

- Данный фильтр "Заменяет" каждый пиксель средним значением по окрестностям.

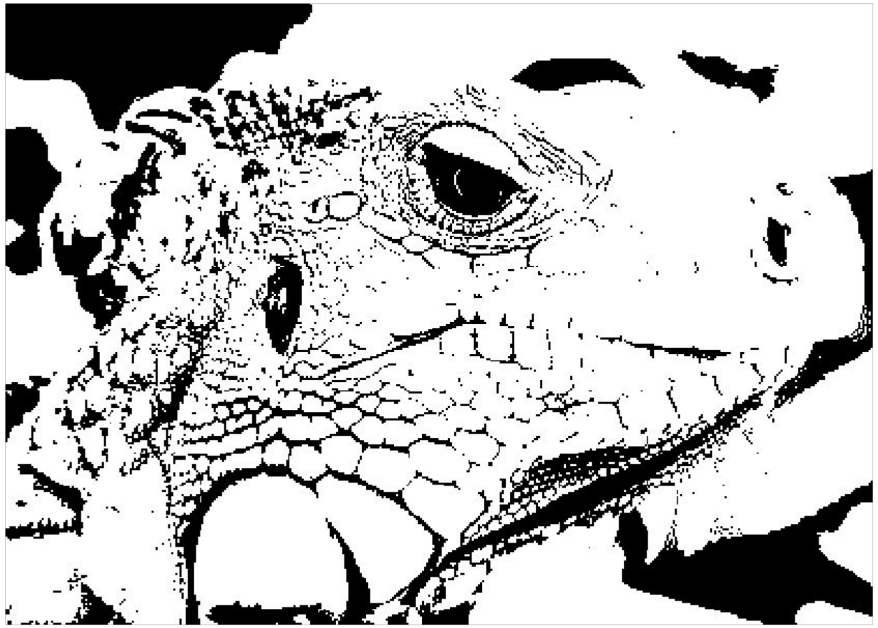
- Достигается эффект сглаживания (осреднение резких переходов значений пикселей).

h

$$\frac{1}{9}\begin{bmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{bmatrix}$$

Пример фильтра №2: Пороговое правило

$$g[n,m] = \begin{cases} 1, & f[n,m] > 100 \\ 0, & \text{otherwise.} \end{cases}$$



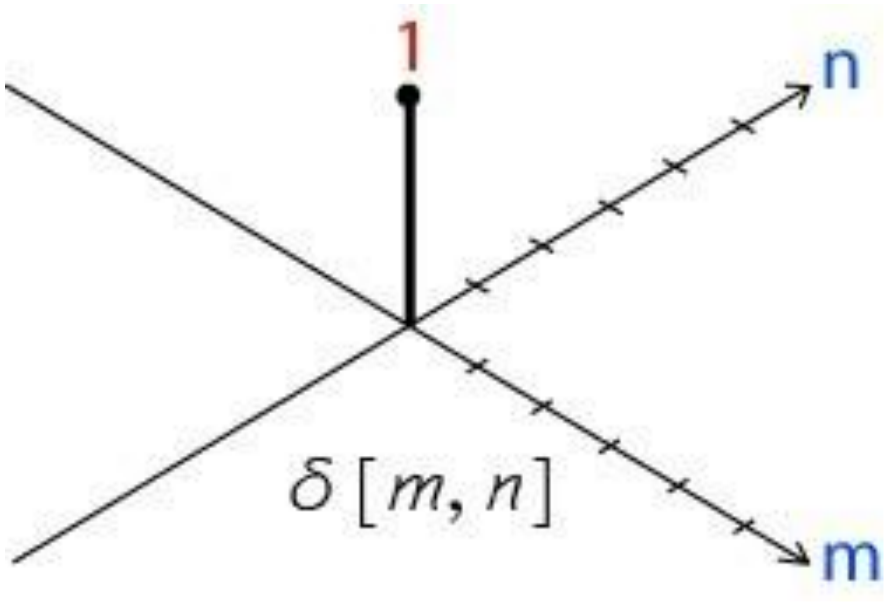
План лекции

- Представление изображения в частотной области. Преобразование Фурье
- Системы и фильтры
- **Свертки**

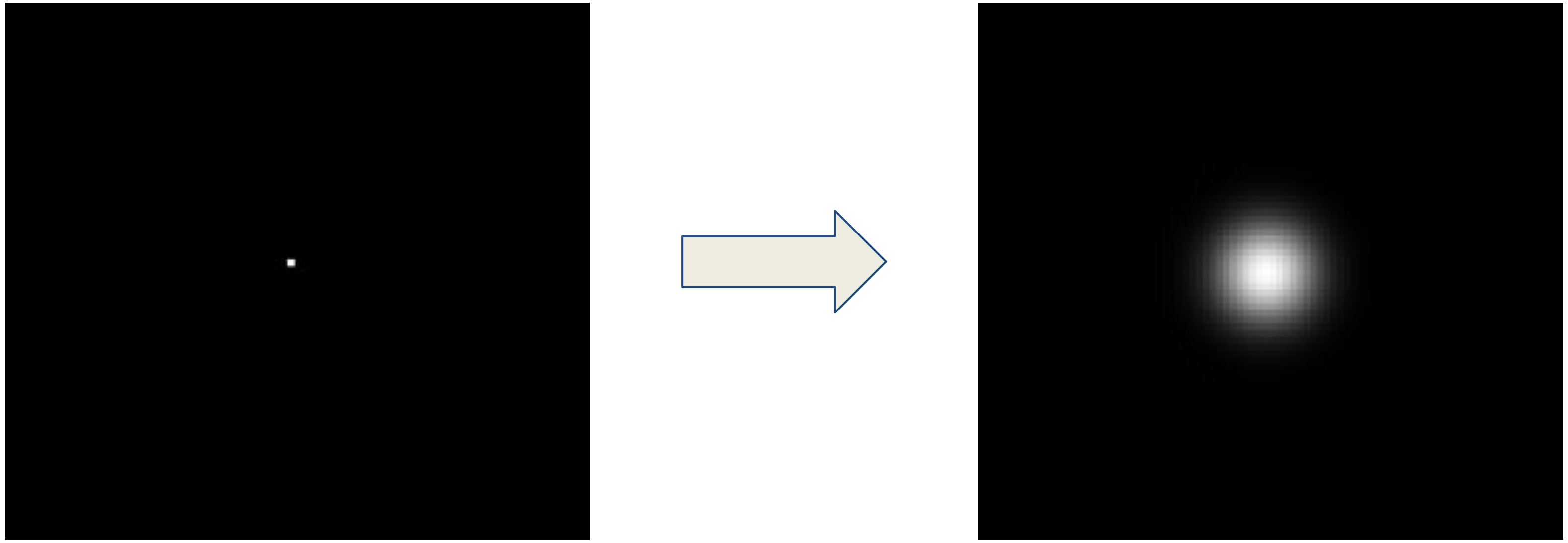
Импульсная функция

Рассмотрим специальную функцию:

- равна 1, в точке [0,0].
- равна 0, во всех остальных точках



Импульсный отклик от фильтра размытия



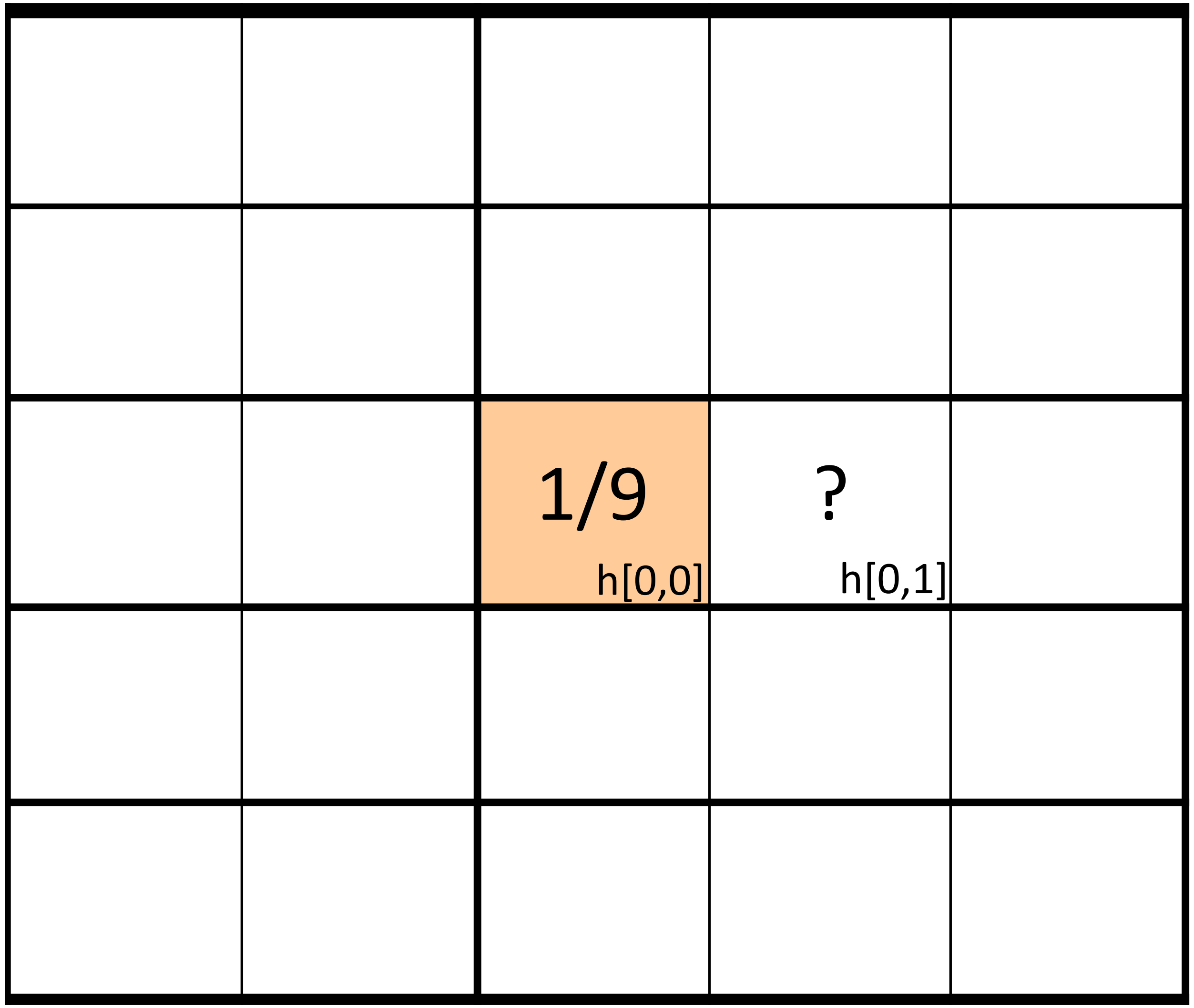
$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix}$$

Импульсный отклик от фильтра размытия

		? h[0,0]		

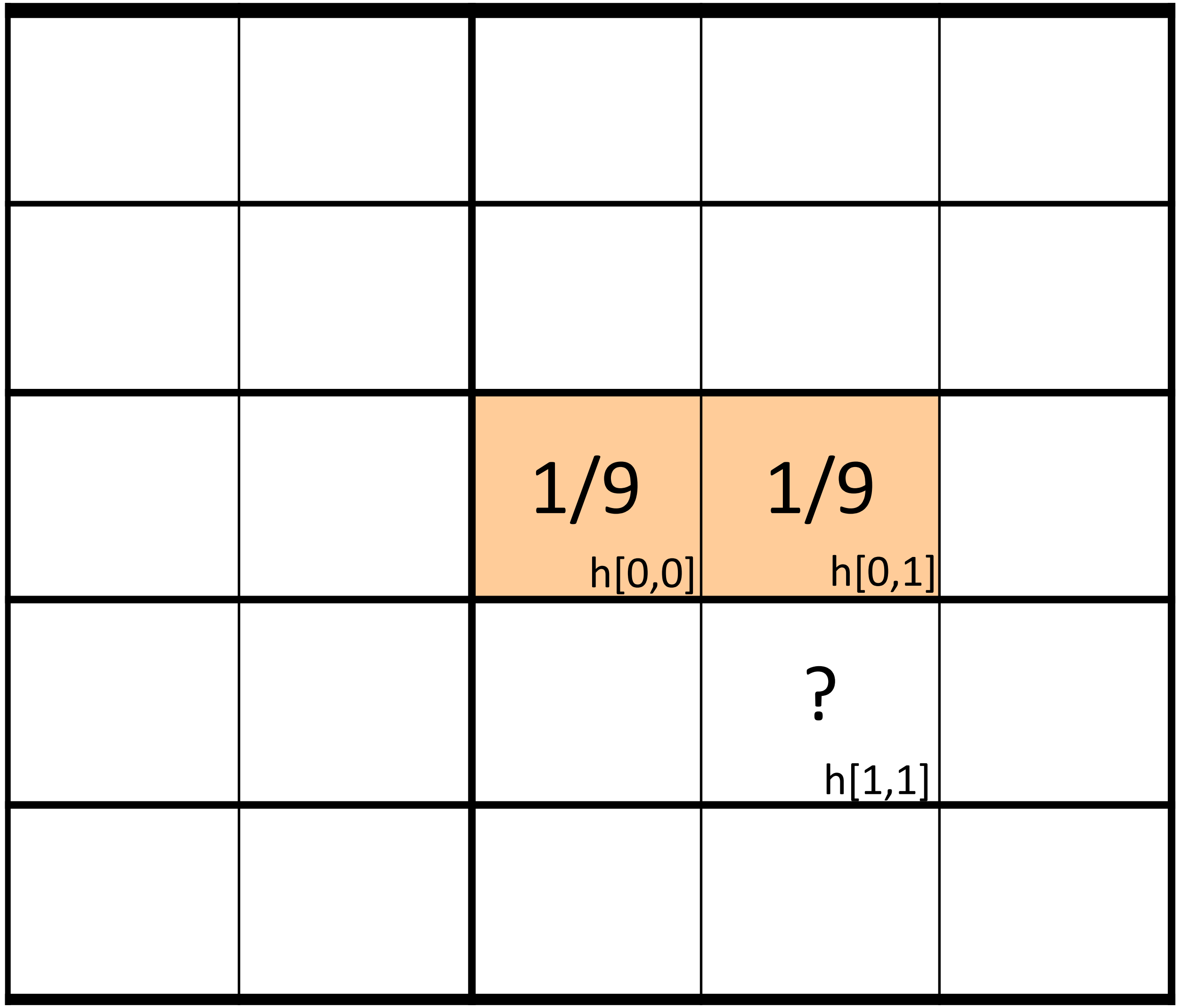
$$\delta_2 \overset{S}{\rightarrow} h[n,m] = \frac{1}{9}\sum_{k=-1}^{1}\sum_{l=-1}^{1}\delta_2[n-k, m-l]$$

Импульсный отклик от фильтра размытия



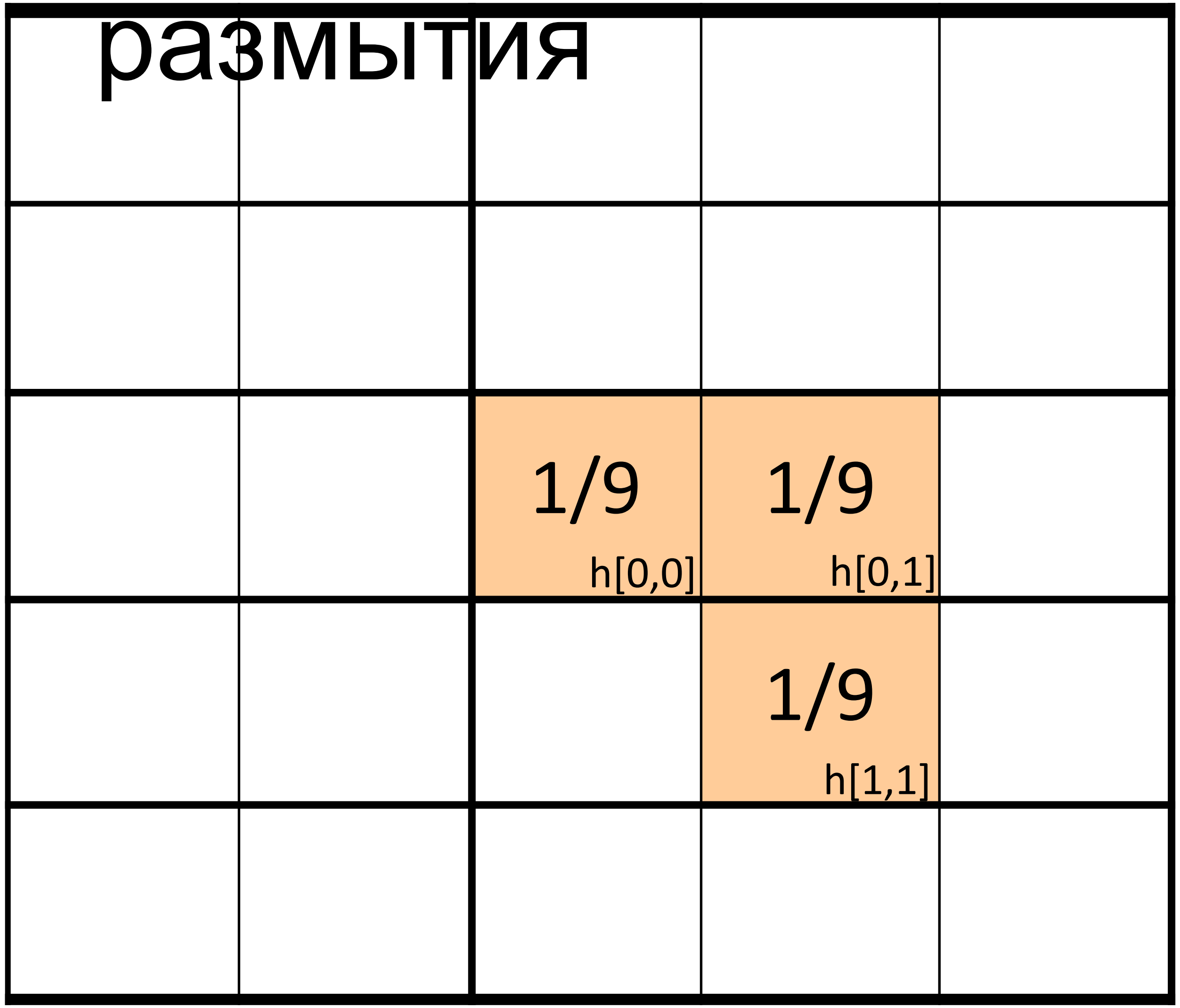
$$\delta_2 \xrightarrow{S} h[n,m]$$
$$= \frac{1}{9}\sum_{k=-1}^{1}\sum_{l=-1}^{1}\delta_2[n-k,m-l]$$

Импульсный отклик от фильтра размытия



$$\delta_2 \xrightarrow{S} h[n,m]$$
$$= \frac{1}{9}\sum_{k=-1}^{1}\sum_{l=-1}^{1}\delta_2[n-k,m-l]$$

Импульсный отклик от фильтра размытия



$$\delta_2 \xrightarrow{S} h[n,m]$$
$$= \frac{1}{9}\sum_{k=-1}^{1}\sum_{l=-1}^{1}\delta_2[n-k,m-l]$$

Импульсный отклик от фильтра размытия

		1/9 h[0,0]	1/9 h[0,1]	? h[0,2]
			1/9 h[1,1]	

$$\delta_2 \xrightarrow{S} h[n,m]$$
$$= \frac{1}{9}\sum_{k=-1}^{1}\sum_{l=-1}^{1}\delta_2[n-k,m-l]$$

Импульсный отклик от фильтра размытия

		1/9 h[0,0]	1/9 h[0,1]	0 h[0,2]
			1/9 h[1,1]	

$$\delta_2 \xrightarrow{S} h[n,m]$$
$$= \frac{1}{9}\sum_{k=-1}^{1}\sum_{l=-1}^{1}\delta_2[n-k,m-l]$$

Импульсный отклик от фильтра

0	0	0	0	0
0	1/9 h[-1,-1]	1/9	1/9	0
0	1/9	1/9 h[0,0]	1/9 h[0,1]	0 h[0,2]
0	1/9	1/9	1/9 h[1,1]	0
0	0	0	0	0

$$\delta_2 \xrightarrow{S} g[n,m] = \frac{1}{9}\sum_{k=-1}^{1}\sum_{l=-1}^{1}\delta_2[n-k,m-l]$$

Фильтр размытия через импульсные функции

$$h[n,m] = \frac{1}{9}\sum_{k=-1}^{1}\sum_{l=-1}^{1}\delta_2[n-k, m-l]$$

$$= \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

h

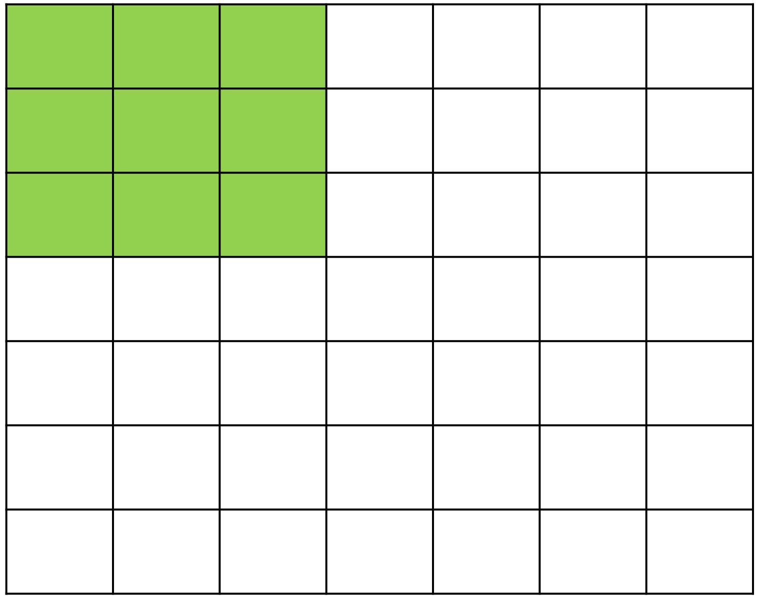
$$\frac{1}{9}$$

1	1	1
1	1	1
1	1	1

Двумерная свертка

2D свёртка очень похожа на 1D.
Основное отличие состоит в том, что теперь нам приходится проводить итерации по 2 осям вместо 1.

$$f[n, m] \text{⇤} h[n, m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k, l]\, h[n-k, m-l]$$

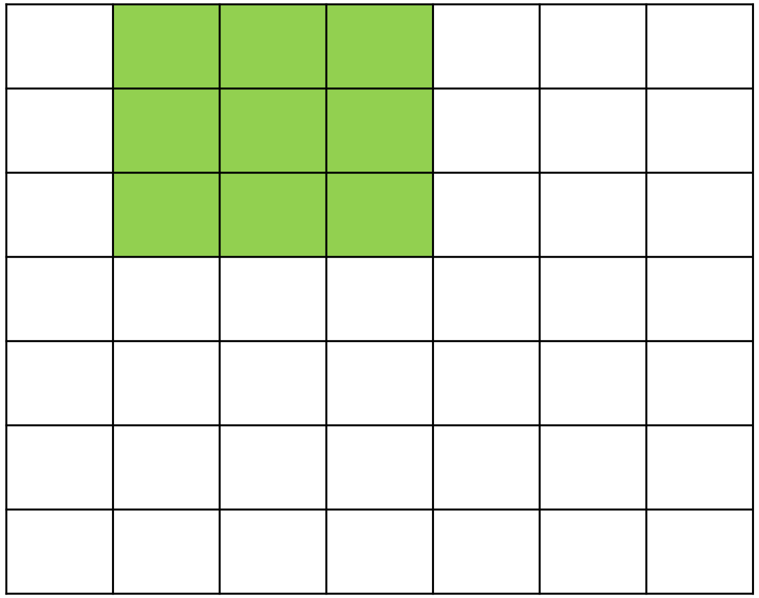


Предположим, что у нас есть фильтр (h[,]) размером 3x3 и изображение (f[,]) размером 7x7.

Двумерная свертка

2D свёртка очень похожа на 1D.
Основное отличие состоит в том, что теперь нам приходится проводить итерации по 2 осям вместо 1.

$$f[n, m] \ast h[n, m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k, l]\, h[n-k, m-l]$$

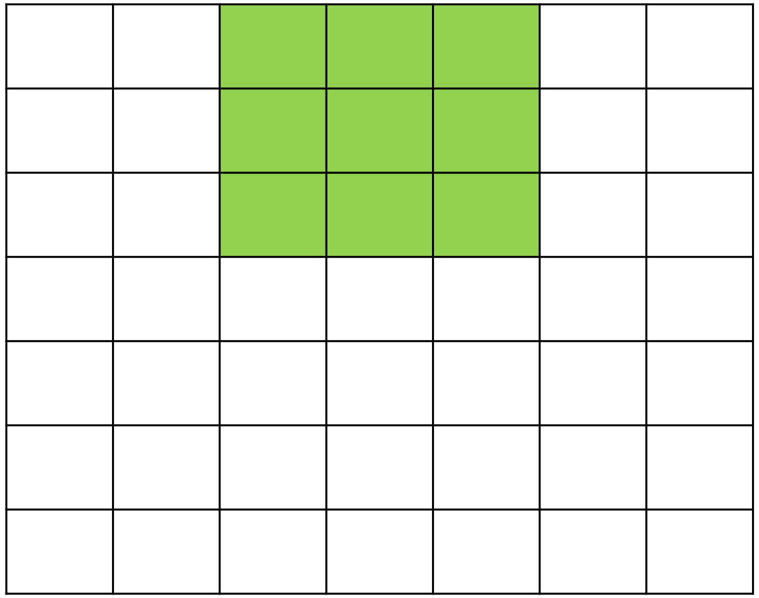


Предположим, что у нас есть фильтр (h[,]) размером 3x3 и изображение (f[,]) размером 7x7.

Двумерная свертка

2D свёртка очень похожа на 1D.
Основное отличие состоит в том, что теперь нам приходится проводить итерации по 2 осям вместо 1.

$$f[n, m] \ast h[n, m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k, l]\, h[n-k, m-l]$$

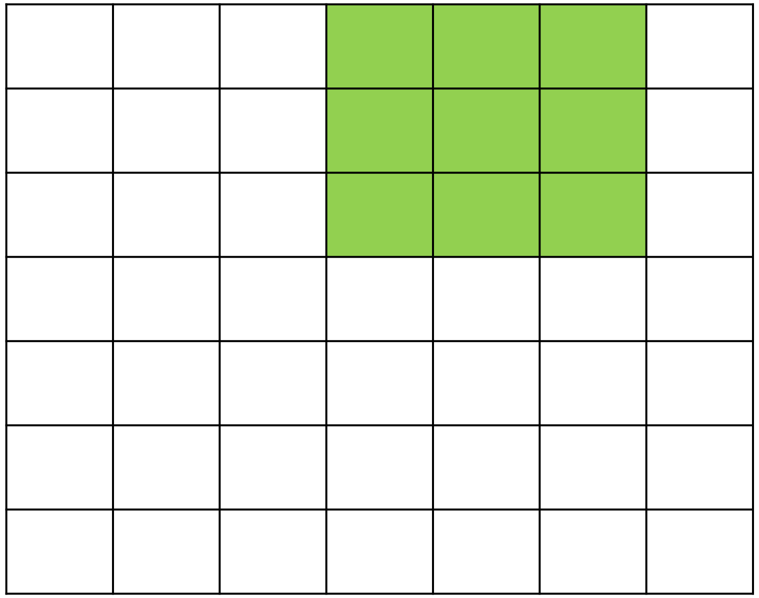


Предположим, что у нас есть фильтр (h[,]) размером 3x3 и изображение (f[,]) размером 7x7.

Двумерная свертка

2D свёртка очень похожа на 1D.
Основное отличие состоит в том, что теперь нам приходится проводить итерации по 2 осям вместо 1.

$$f[n, m] \ast h[n, m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k, l]\, h[n-k, m-l]$$

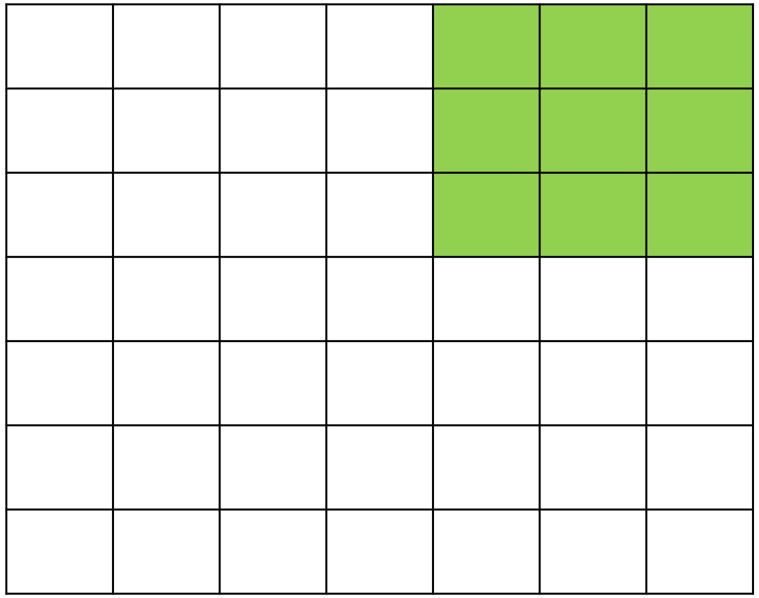


Предположим, что у нас есть фильтр (h[,]) размером 3x3 и изображение (f[,]) размером 7x7.

Двумерная свертка

2D свёртка очень похожа на 1D.
Основное отличие состоит в том, что теперь нам приходится проводить итерации по 2 осям вместо 1.

$$f[n, m] \ast h[n, m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k, l]\, h[n-k, m-l]$$

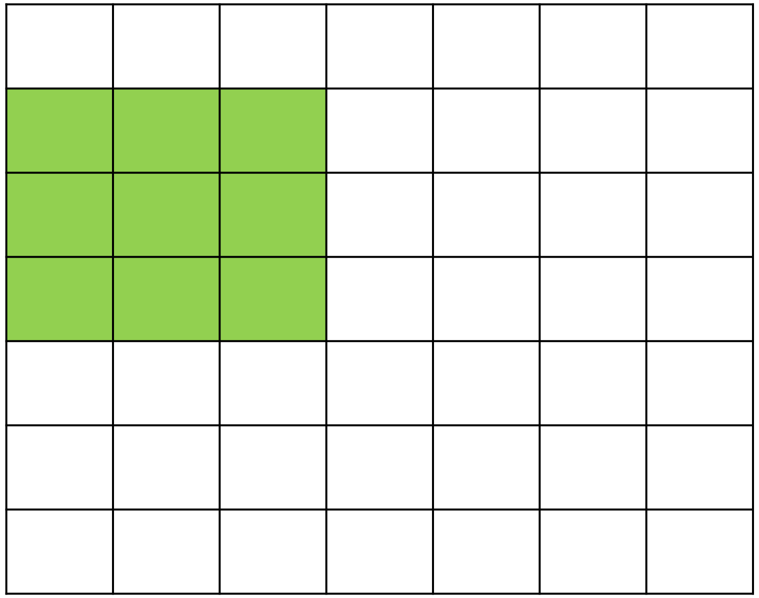


Предположим, что у нас есть фильтр (h[,]) размером 3x3 и изображение (f[,]) размером 7x7.

Двумерная свертка

2D свёртка очень похожа на 1D.
Основное отличие состоит в том, что теперь нам приходится проводить итерации по 2 осям вместо 1.

$$f[n, m] \mid\!\!\leftarrow h[n, m] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k, l]\, h[n-k, m-l]$$



Предположим, что у нас есть фильтр (h[,]) размером 3x3 и изображение (f[,]) размером 7x7.

Пример двумерной свертки

1	2	3
4	5	6
7	8	9

Input

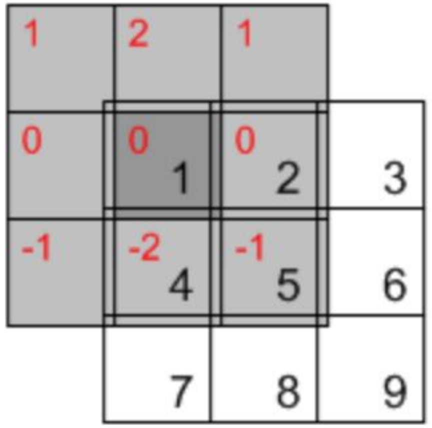
n \ m	-1	0	1
-1	-1	-2	-1
0	0	0	0
1	1	2	1

Kernel

-13	-20	-17
-18	-24	-18
13	20	17

Output

Пример двумерной свертки

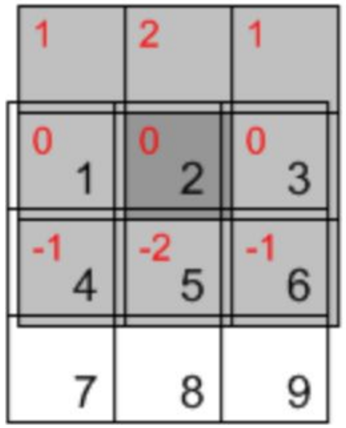


$$
\begin{aligned}
y[0,0] = {} & x[-1,-1]\cdot h[1,1] + x[0,-1]\cdot h[0,1] + x[1,-1]\cdot h[-1,1] \\
& + x[-1,0]\cdot h[1,0] + x[0,0]\cdot h[0,0] + x[1,0]\cdot h[-1,0] \\
& + x[-1,1]\cdot h[1,-1] + x[0,1]\cdot h[0,-1] + x[1,1]\cdot h[-1,-1] \\
= {} & 0\cdot 1 + 0\cdot 2 + 0\cdot 1 + 0\cdot 0 + 1\cdot 0 + 2\cdot 0 + 0\cdot(-1) + 4\cdot(-2) + 5\cdot(-1) = -13
\end{aligned}
$$

-13	-20	-17
-18	-24	-18
13	20	17

Output

Пример двумерной свертки

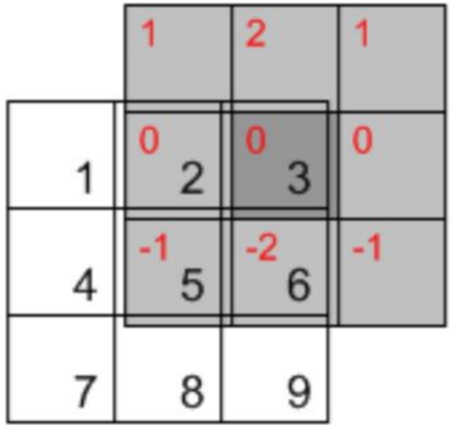


$$
\begin{aligned}
y[1,0] = {} & x[0,-1]\cdot h[1,1] + x[1,-1]\cdot h[0,1] + x[2,-1]\cdot h[-1,1] \\
& + x[0,0]\cdot h[1,0] + x[1,0]\cdot h[0,0] + x[2,0]\cdot h[-1,0] \\
& + x[0,1]\cdot h[1,-1] + x[1,1]\cdot h[0,-1] + x[2,1]\cdot h[-1,-1] \\
= {} & 0\cdot 1 + 0\cdot 2 + 0\cdot 1 + 1\cdot 0 + 2\cdot 0 + 3\cdot 0 + 4\cdot(-1) + 5\cdot(-2) + 6\cdot(-1) = -20
\end{aligned}
$$

-13	-20	-17
-18	-24	-18
13	20	17

Output

Пример двумерной свертки

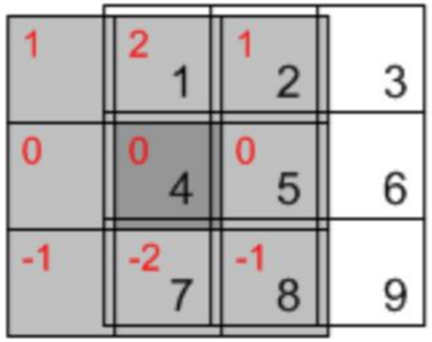


$$
\begin{aligned}
y[2,0] &= x[1,-1]\cdot h[1,1] + x[2,-1]\cdot h[0,1] + x[3,-1]\cdot h[-1,1] \\
&\quad + x[1,0]\cdot h[1,0] + x[2,0]\cdot h[0,0] + x[3,0]\cdot h[-1,0] \\
&\quad + x[1,1]\cdot h[1,-1] + x[2,1]\cdot h[0,-1] + x[3,1]\cdot h[-1,-1] \\
&= 0\cdot 1 + 0\cdot 2 + 0\cdot 1 + 2\cdot 0 + 3\cdot 0 + 0\cdot 0 + 5\cdot(-1) + 6\cdot(-2) + 0\cdot(-1) = -17
\end{aligned}
$$

-13	-20	$\boxed{-17}$
-18	-24	-18
13	20	17

Output

Пример двумерной свертки



$$
\begin{aligned}
y[0,1] = {} & x[-1,0]\cdot h[1,1] + x[0,0]\cdot h[0,1] + x[1,0]\cdot h[-1,1] \\
& + x[-1,1]\cdot h[1,0] + x[0,1]\cdot h[0,0] + x[1,1]\cdot h[-1,0] \\
& + x[-1,2]\cdot h[1,-1] + x[0,2]\cdot h[0,-1] + x[1,2]\cdot h[-1,-1] \\
= {} & 0\cdot 1 + 1\cdot 2 + 2\cdot 1 + 0\cdot 0 + 4\cdot 0 + 5\cdot 0 + 0\cdot(-1) + 7\cdot(-2) + 8\cdot(-1) = -18
\end{aligned}
$$

-13	-20	-17
-18	-24	-18
13	20	17

Output

Пример двумерной свертки

$$* \begin{bmatrix} \bullet 0 & \bullet 0 & \bullet 0 \\ \bullet 0 & \bullet 1 & \bullet 0 \\ \bullet 0 & \bullet 0 & \bullet 0 \end{bmatrix} = \quad ?$$

Пример двумерной свертки

*

•0	•0	•0
•0	•1	•0
•0	•0	•0

=

Пример двумерной свертки

*

•0	•0	•0
•0	•0	•1
•0	•0	•0

=

?

Пример двумерной свертки

*

•0	•0	•0
•0	•0	•1
•0	•0	•0

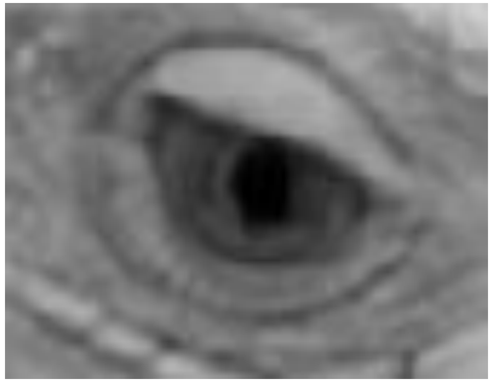
=

Пример двумерной свертки

$$* \frac{1}{9} \begin{bmatrix} \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \end{bmatrix} = ?$$

Пример двумерной свертки

$$* \ \frac{1}{9} \begin{bmatrix} \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \\ \bullet 1 & \bullet 1 & \bullet 1 \end{bmatrix} =$$



Пример двумерной свертки

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = ?$$

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \overbrace{\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}}$$

Что отнимает размытость?

Оригинально
е
изображение

-

Размыто
е

=

Детал
и

Оригинально
е

+

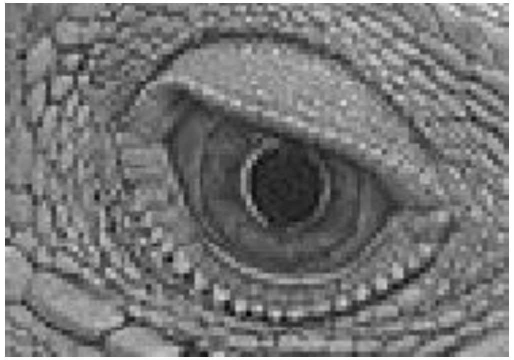
Детал
и

=

Повышенная
резкость

Пример двумерной свертки – фильтр резкости

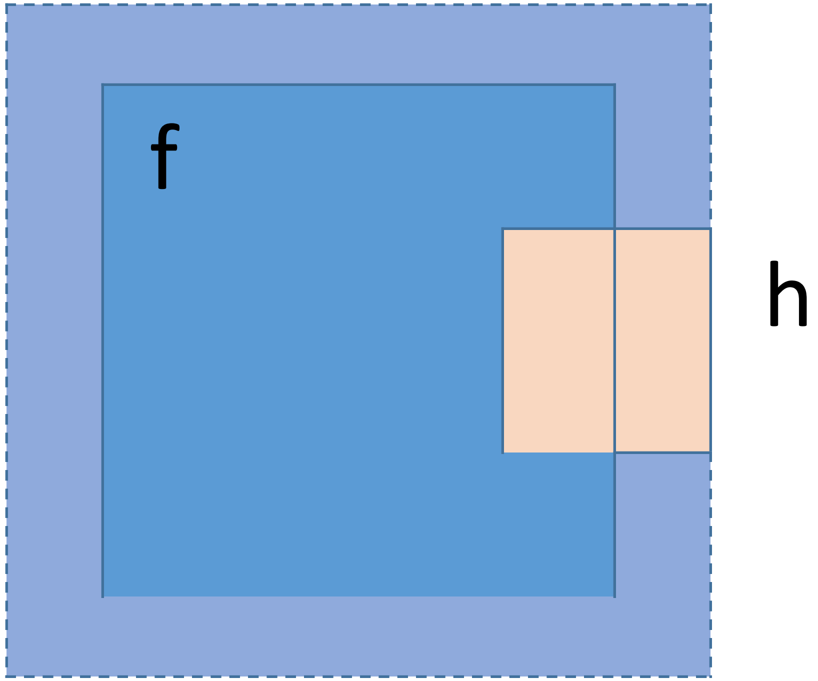
$$\begin{bmatrix}\bullet 0 & \bullet 0 & \bullet 0\\ \bullet 0 & \bullet 2 & \bullet 0\\ \bullet 0 & \bullet 0 & \bullet 0\end{bmatrix} - \frac{1}{9}\begin{bmatrix}\bullet 1 & \bullet 1 & \bullet 1\\ \bullet 1 & \bullet 1 & \bullet 1\\ \bullet 1 & \bullet 1 & \bullet 1\end{bmatrix} =$$



Фильтр резкости: подчеркивает разность со средним местным значениями пикселей

Краевой эффект

- Компьютер будет вызывать только **конечные сигналы**.
- Что происходит на краю?



- нулевой паддинг
- повторение на краях
- отзеркаливание

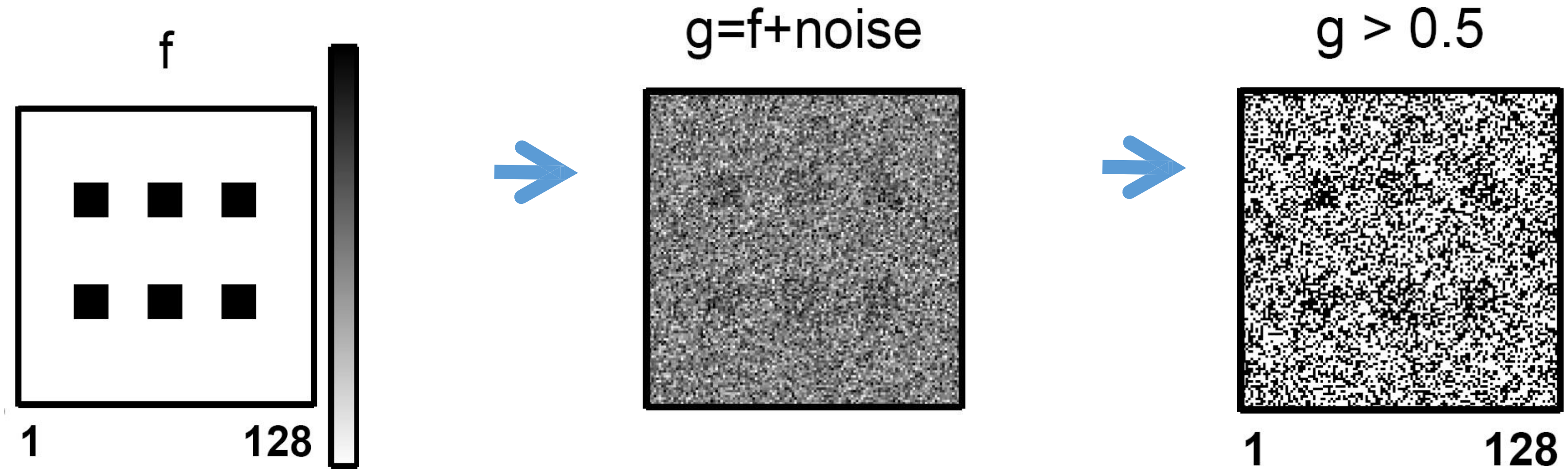
Кросс-корреляция

Кросс-корреляция двух 2D сигналов f[n,m] и h[n,m].

$$f[n,m] ** h[n,m] = \sum_k \sum_l f[k,l] h[n-k, m-l]$$

- Эквивалент свертывания без переворачивания
- Измерения "сходства" между f и h.

Пример кросс-корреляции



Свертка vs кросс-корреляция

- **Свертка** – это интеграл, выражающий величину перекрытия одной функции при ее смещении по другой.
  - свертка – это операция фильтрации

- **Корреляция** сравнивает **сходство** двух **наборов данных**. Корреляция рассчитывает меру сходства двух входных сигналов при их смещении друг от друга. Результат корреляции достигает максимума в тот момент, когда два сигнала совпадают наилучшим образом .
  - корреляция является мерой сходства двух сигналов.

Итоги

- Рассмотрено частотное представление изображения
- Показаны методы фильтрации в пространственной и частотной областях
- Изучено понятие свертки и кросс-корреляции