

Цифровая обработка изображений

Лекция 5

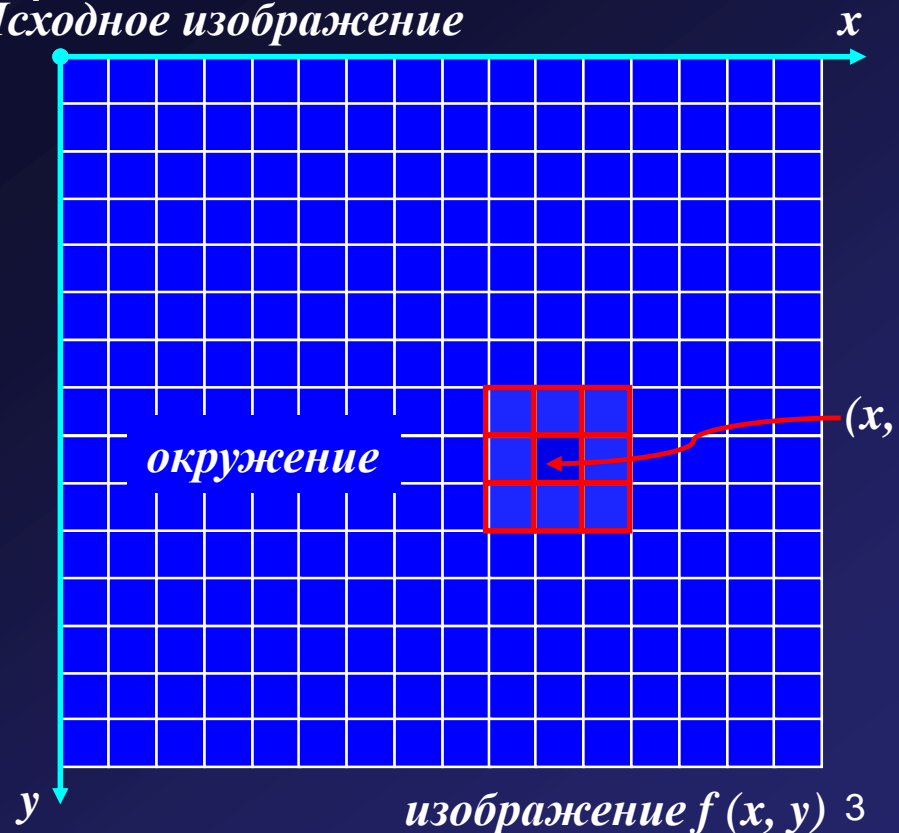
Локальная обработка изображений. Конволюция.

Окрестность пикселя

Операция в окружении

- Операции работают на большей окрестности, чем для точечных операций
- Области определяются, чаще всего, прямоугольником вокруг центрального пиксела
- Зависит от размера прямоугольника и формы фильтра.

Исходное изображение



Простые операции

- Некоторые простые операции в окрестности включают:

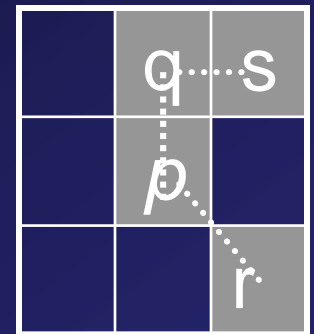
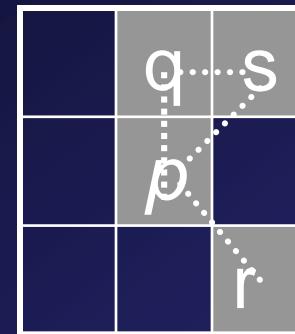
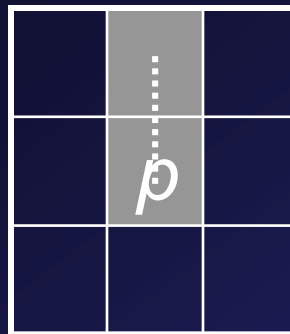
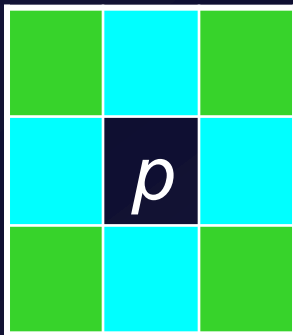
- » Мин: Установите значение пикселя до минимума в окрестности

- » Макс: Установите значение пикселя до максимума в окрестности

- » Медиана: медианное значение набора чисел является серединой значение в этом наборе (например, из множества [1, 7, 15, 18, 24] 15 медиану). Иногда средний работает лучше, чем в среднем

Соседство и связность

- $N_4(p)$: 4-соседа
- $N_D(p)$: 4-соседа по диагонали
- $N_8(p) = N_4(p) \cup N_D(p)$: 8-соседей
- Соседство:
- r : набор уровней яркости, если уровень серого пиксела в r , то он будет использоваться для установления определения соседства.
 - » 4-связности
 - » 8-связности
 - » M-связности



Путь, Связность, Область

- Путь:
 - » Путь из $p = (s, t)$ to $q = (x, y)$ является множеством связанных точек
$$P(p, q) = \{(s, t), \dots, (x_i, y_i), \dots, (x, y)\}$$
 - » If $(s, t) = (x, y)$, P является замкнутым путем.
- p и q являются связанными в S если $P(p, q) \subset S$. Множество $\{q; P(p, q) \subset S\}$ является связанной компонентой $p \in S$.
- S является связанным множеством если имеет хотя бы один смежный пиксель в компонентах. S также называется областью.
- Граница области $R = \{p; p \in R, \text{существуют } q \notin R, q \in N(p)\}$
- If R is entire image, its boundary is the first and last rows and columns of pixels.

Метрики и расстояния

- Нормированные расстояния

Задаются p, q, r , в функции $D(p, q)$, которая является функцией расстояния

a) $D(p, q) \geq 0$ and

$D(p, q) = 0$ iff $p = q$

b) $D(p, q) = D(q, p)$

c) $D(p, r) \leq D(p, q) + D(q, r)$

- Евклидово расстояние

» L_2 norm

$$D_e(p, q) = \sqrt{(x - s)^2 + (y - t)^2}$$

- Городская метрика

» L_1 norm

$$D_4(p, q) = |x - s| + |y - t|$$

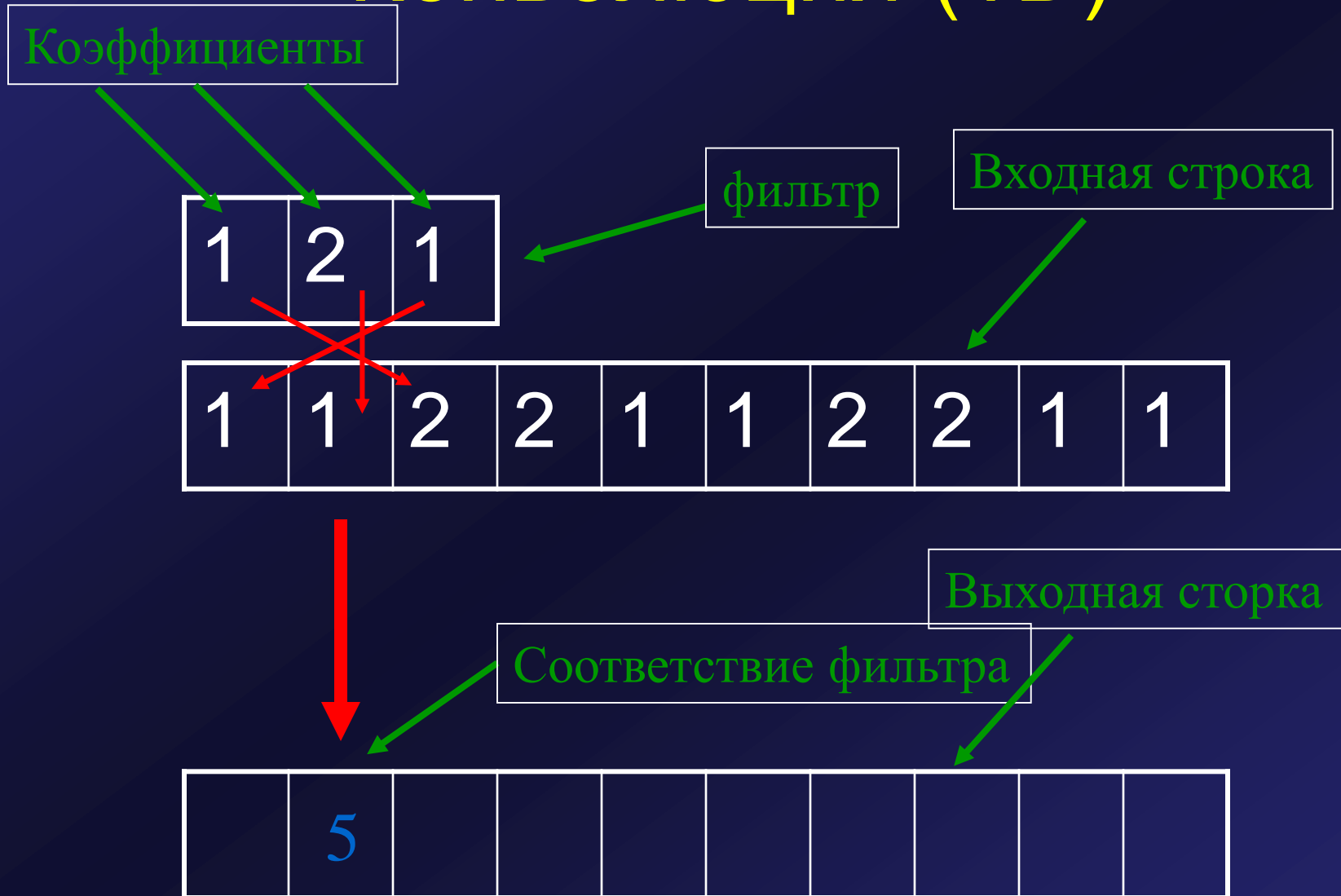
- Шахматная метрика

» L_∞ norm

$$D_8(p, q) = \max(|x - s|, |y - t|)$$

Конволюция

Конволюция (1D)



Нормализация отклика фильтра

Коэффициенты фильтра

A	B	C
---	---	---

Максимальное значение
изображения

...	255	255	255		...
-----	-----	-----	-----	--	-----

$$\begin{aligned}\text{Максимальный} &= A \cdot 255 + B \cdot 255 + C \cdot 255 \Leftrightarrow \\ \text{отклик фильтра} &= 255 \cdot (A + B + C)\end{aligned}$$

Если максимальный отклик фильтра = 255 (один байт) тогда

Отклик с нормализацией = отклик / (A+B+C)

КОНВОЛЮЦИЯ (1D)

1	2	1
---	---	---

1	1	2	2	1	1	2	2	1	1
---	---	---	---	---	---	---	---	---	---



	$\frac{5}{4}$								
--	---------------	--	--	--	--	--	--	--	--

КОНВОЛЮЦИЯ (1D)

1	2	1
---	---	---

1	1	2	2	1	1	2	2	1	1
---	---	---	---	---	---	---	---	---	---

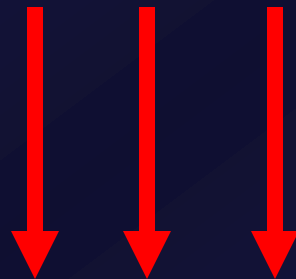


	$\frac{5}{4}$	$\frac{7}{4}$							
--	---------------	---------------	--	--	--	--	--	--	--

КОНВОЛЮЦИЯ (1D)

1	2	1
---	---	---

1	1	2	2	1	1	2	2	1	1
---	---	---	---	---	---	---	---	---	---

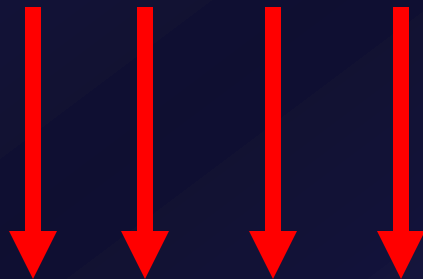


	$\frac{5}{4}$	$\frac{7}{4}$	$\frac{7}{4}$						
--	---------------	---------------	---------------	--	--	--	--	--	--

КОНВОЛЮЦИЯ (1D)

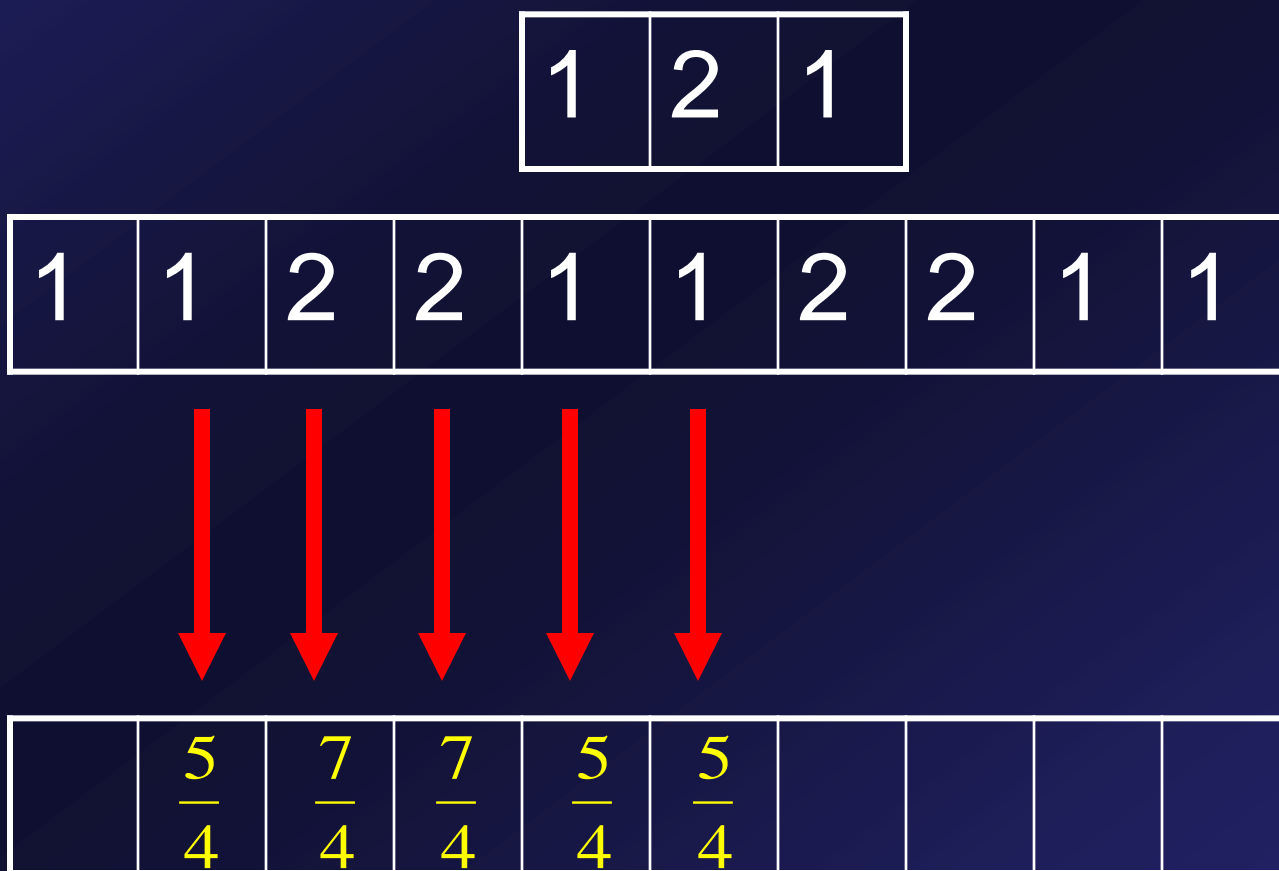
1	2	1
---	---	---

1	1	2	2	1	1	2	2	1	1
---	---	---	---	---	---	---	---	---	---

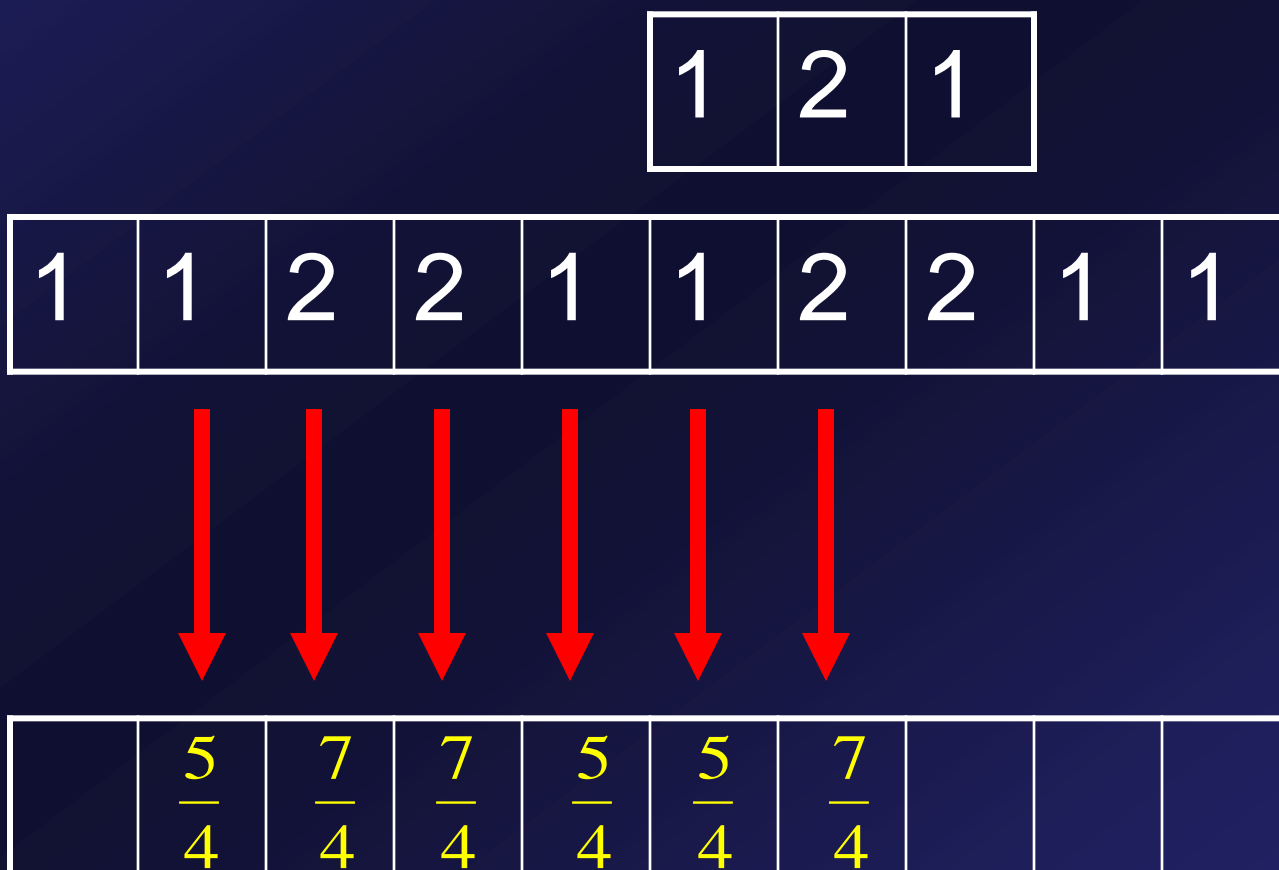


	$\frac{5}{4}$	$\frac{7}{4}$	$\frac{7}{4}$	$\frac{5}{4}$					
--	---------------	---------------	---------------	---------------	--	--	--	--	--

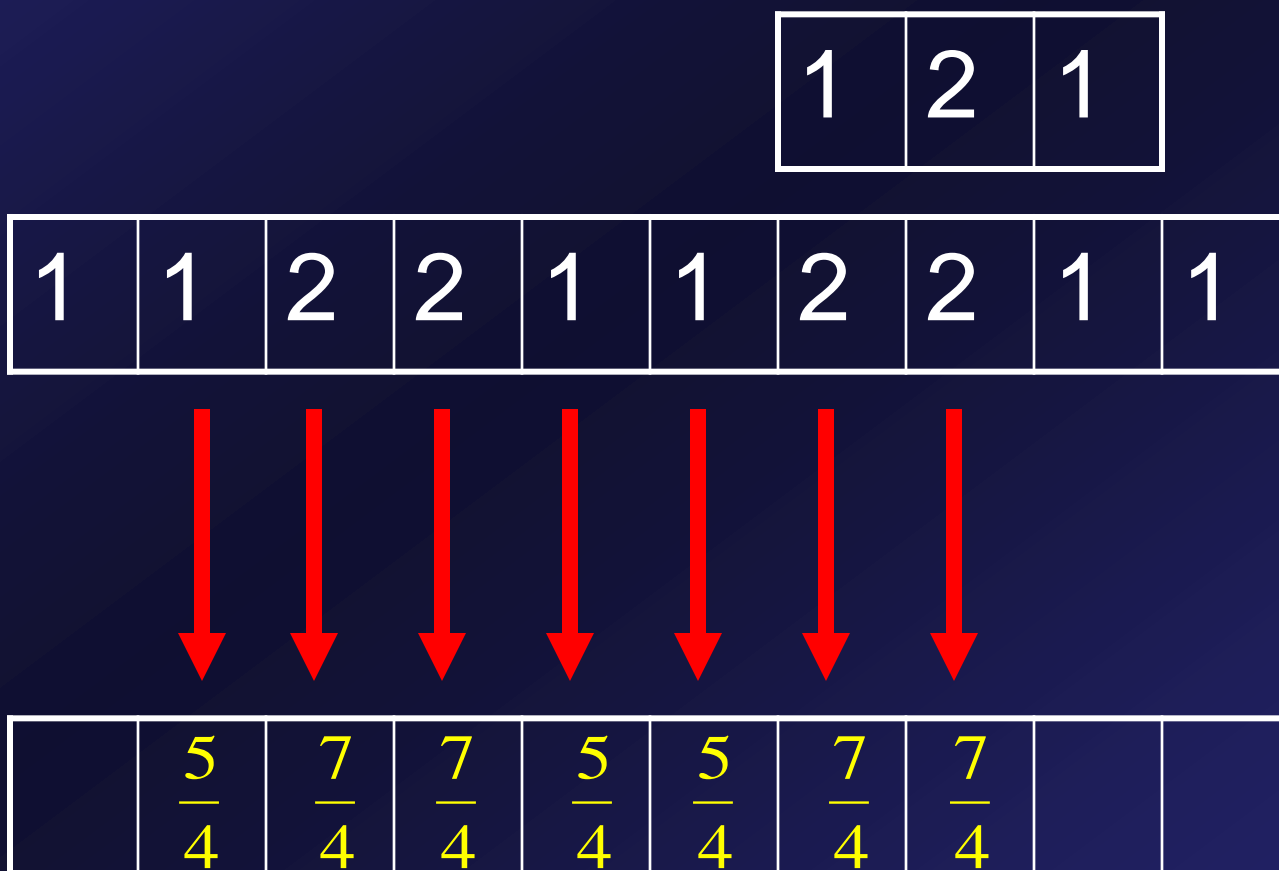
КОНВОЛЮЦИЯ (1D)



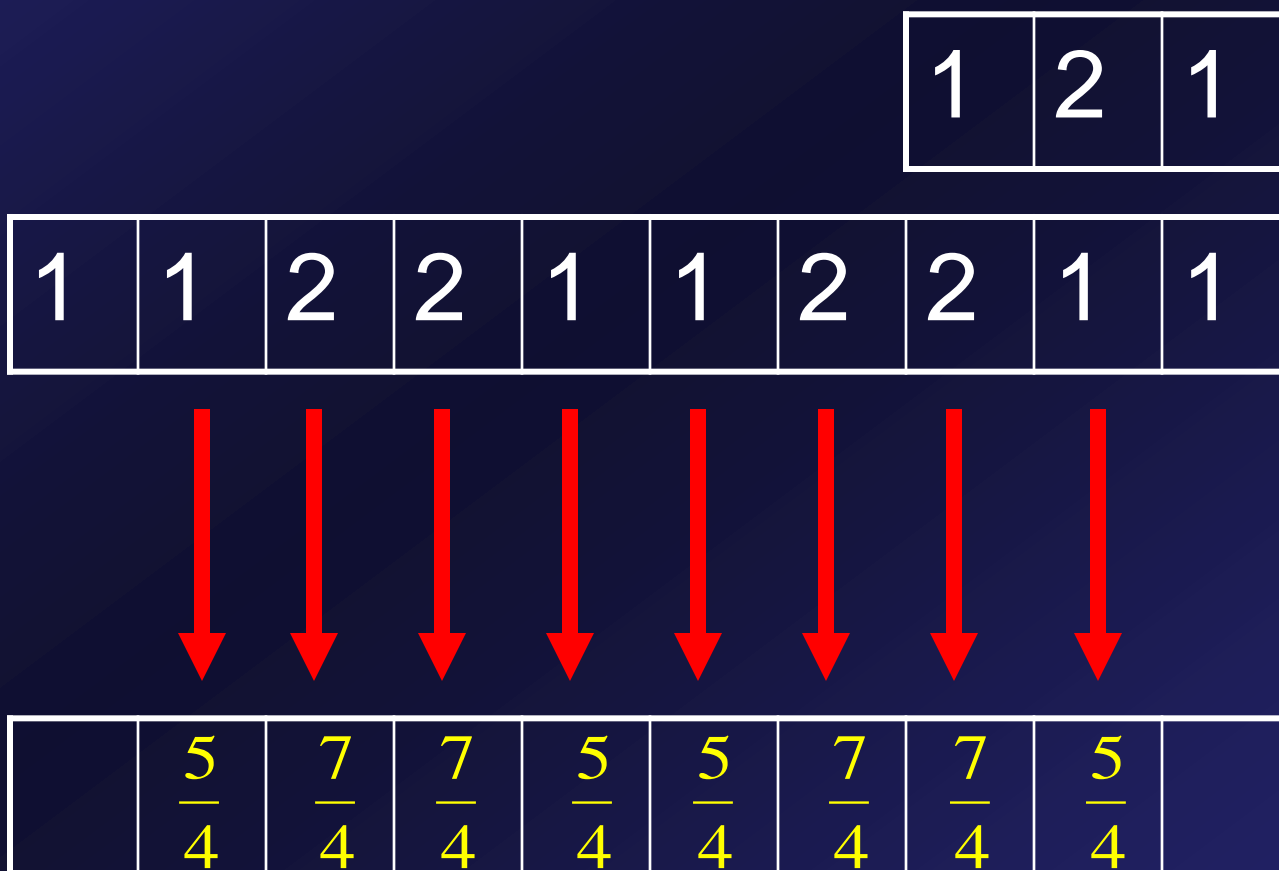
КОНВОЛЮЦИЯ (1D)



КОНВОЛЮЦИЯ (1D)



КОНВОЛЮЦИЯ (1D)



Формулы конволюции

$$g(x) = h * f(x) = \sum_{i=-n}^n h(i) f(x-i)$$

$g(x)$: выход, h : фильтр, $*$ действие конволюции,

$f(x)$: вход, $n = \lfloor \text{ширина фильтра} / 2 \rfloor$

$\lfloor _ \rfloor$: округляет в меньшую сторону, например : $\lfloor 1.7 \rfloor = 1$

На пример: фильтр (h):

$c = 3 \Rightarrow n=1$

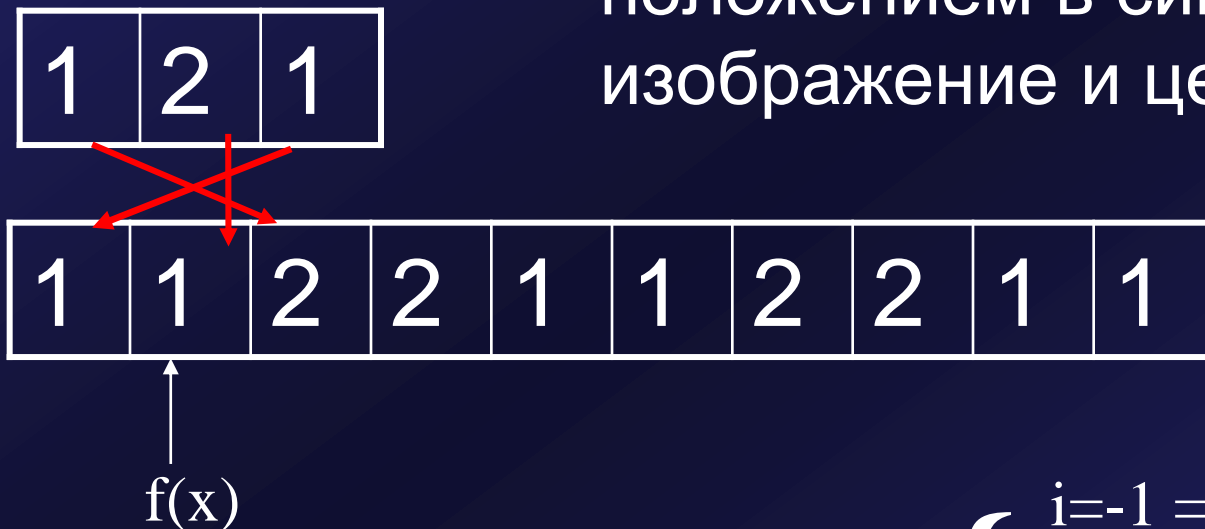


$h(-1)=1$ $h(0)=2$ $h(1)=1$

Формулы конволюции

$$g(x) = h * f(x) = \sum_{i=-n}^n h(i) f(x-i)$$

- x является пикселем интереса, т. е. положением в сигнале / изображении и центром фильтра



$$n = 1 \Rightarrow i \in \{-1, 0, 1\} \in \left\{ \begin{array}{l} i = -1 \Rightarrow f(x - (-1)) = f(x + 1) = 2 \\ i = 0 \Rightarrow f(x - 0) = f(x) = 1 \\ i = 1 \Rightarrow f(x - 1) = f(x - 1) = 1 \end{array} \right.$$

Формулы конволюции



$$g(x) = h * f(x) = \sum_{i=-n}^n h(i) f(x-i)$$



f(x)

$$i = -1: h(-1) \cdot f(x+1) = 1 \cdot 2 = 2$$

$$i = 0: h(0) \cdot f(x) = 2 \cdot 1 = 2$$

$$i = 1: h(1) \cdot f(x-1) = 1 \cdot 1 = 1$$

$$g(x) = 2 + 2 + 1 = 5$$

$$\text{Normalise : } g(x) = 5 / 4$$

Корреляция

Корреляция (1D)



Корреляция vs Конволюция

корреляция

1	2	1
---	---	---

$$g(x) = h \circ f(x) = \sum_{i=-n}^n h(i) f(x+i)$$

1	1	2	2	1	1	2	2	1	1
---	---	---	---	---	---	---	---	---	---

КОНВОЛЮЦИЯ

1	2	1
---	---	---

$$g(x) = h * f(x) = \sum_{i=-n}^n h(i) f(x-i)$$

1	1	2	2	1	1	2	2	1	1
---	---	---	---	---	---	---	---	---	---

В обработке изображений используется корреляция

но (почти) всегда называют его сверкой, т.е. конволюцией!!!

Примечание: Когда фильтр симметричен: корреляция = конволюции!

Convolution/correlation on images

Normalisation

- The filter is now 2D
- Kernel (mask), kernel coefficients
- Size: **3x3**, 5x5, 7x7,

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

Input

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

Output

	$\frac{12}{9}$				

Convolution/correlation on images

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

Input

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

Output

	$\frac{12}{9}$	$\frac{11}{9}$			

конволюция/корреляция изображений

$\frac{1}{9}$

1	1	1
1	1	1
1	1	1

Input

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

Output

	$\frac{12}{9}$	$\frac{11}{9}$	$\frac{14}{9}$		

конволюция/корреляция изображений

 $\frac{1}{9}$

1	1	1
1	1	1
1	1	1

Input

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

Output

	$\frac{12}{9}$	$\frac{11}{9}$	$\frac{14}{9}$		
	$\frac{13}{9}$	$\frac{11}{9}$	$\frac{13}{9}$		
	$\frac{16}{9}$	$\frac{12}{9}$	$\frac{11}{9}$		

Формулы 2D

конволюция/корреляция

КОНВОЛЮЦИЯ

$$g(x, y) = h * f(x, y) = \sum_{j=-n}^n \sum_{i=-m}^m h(i, j) f(x-i, y-j)$$

корреляция

$$g(x, y) = h \circ f(x, y) = \sum_{j=-n}^n \sum_{i=-m}^m h(i, j) f(x+i, y+j)$$

1	1	1
1	1	1
1	1	1

2	3	2
-1	0	-1
2	3	2

Применение корреляции

- Много, определится программистом
и некоторыми стандартными операциями
 - » обнаружение объекта
 - » Размытие изображения
 - » удаление шума
 - » Морфология (позже)
 - » Обнаружение края (позже)

Простая детекция объектов

- Поиск конкретного объекта на изображении
- 1D пример: объект дается (известный) в виде изображения, например,
- Задача: Найти этот объект на изображении:

30	60	30
----	----	----

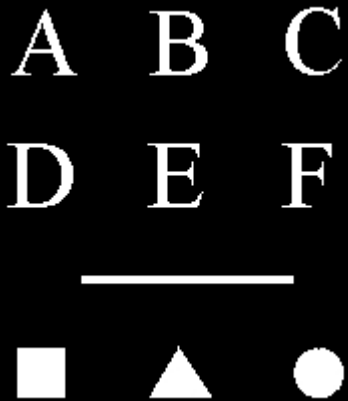
Input	20	25	30	60	30	20	40	60	10	2
Output		300 0	435 0	540 0	420 0	430 0	480 0	510 0	246 0	

Для изображений это называется correlation или шаблон соответствия!

Распознавание по шаблону

- Фильтр называется шаблоном или маской

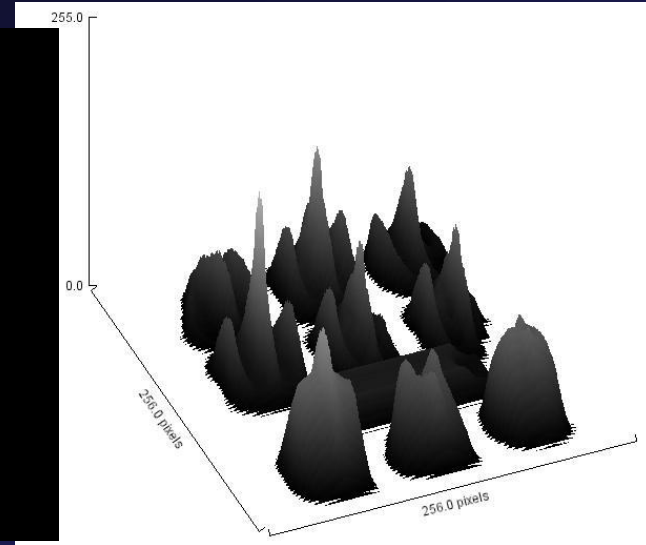
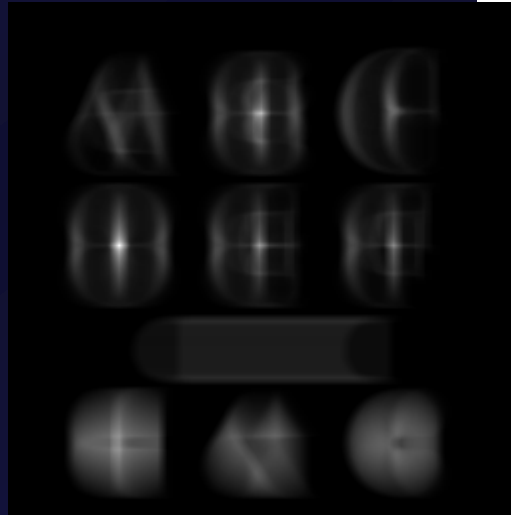
Input image



шаблон



Output



- Чем ярче значение на выходе, тем лучше распознавание

Распознавание по шаблону

- Два важных приложения:
 - » Поиск конкретного объекта
 - Какой тип объекта?

Input image



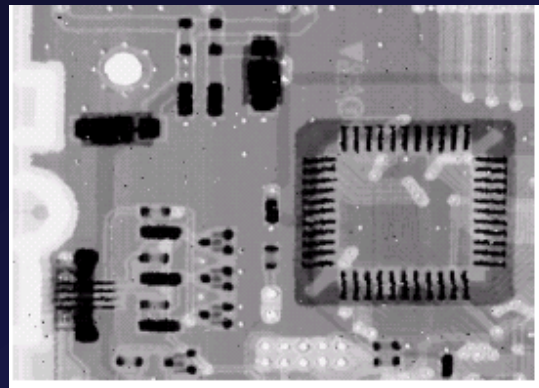
шаблоны:



– Объект на изображении?



Исходное
изображение

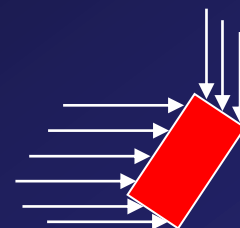


Распознавание по шаблону

- Поиск с поворотом

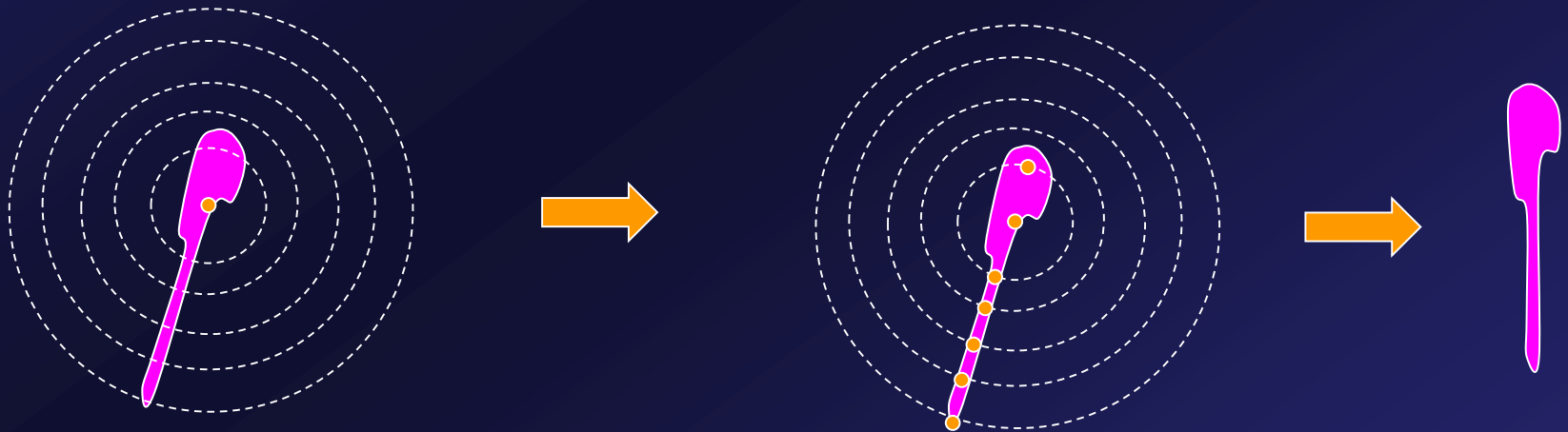


шаблон



Распознавание по шаблону

- вращение
 - » Концентрические круги



Сглаживание изображений

- И известны как: Сглаживание, усредняющий фильтр, фильтр нижних частот
- Самый простой фильтр:
- Пространственный фильтр нижних частот

$$\frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$$

- Другая маска:
- Фильтр Гаусса:

$$\frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$$

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

1	1	1
1	1	1
1	1	1

$$\frac{1}{16}$$

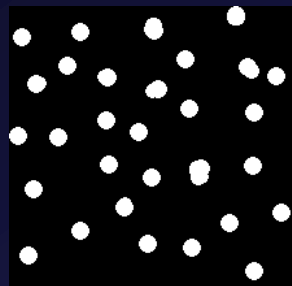
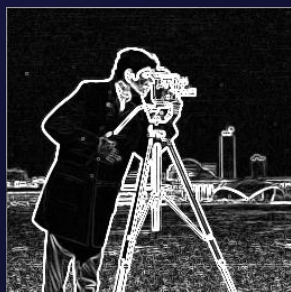
1	2	1
2	4	2
1	2	1

Обработка изображений

Конволюция и фильтры

Для чего используется?

- Многие операции определяются при программировании:
 - » Сглаживание
 - » Удаление шума
 - » Поиск объектов
 - » Морфология
 - » Определение края



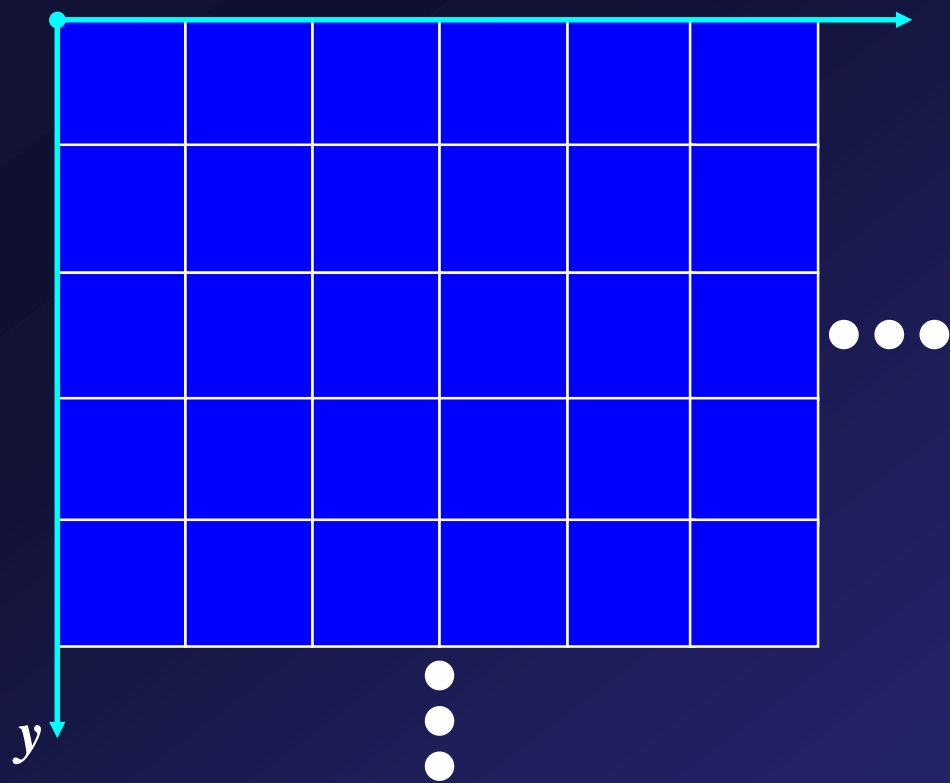
Простые операции. Пример

Исходное изображение

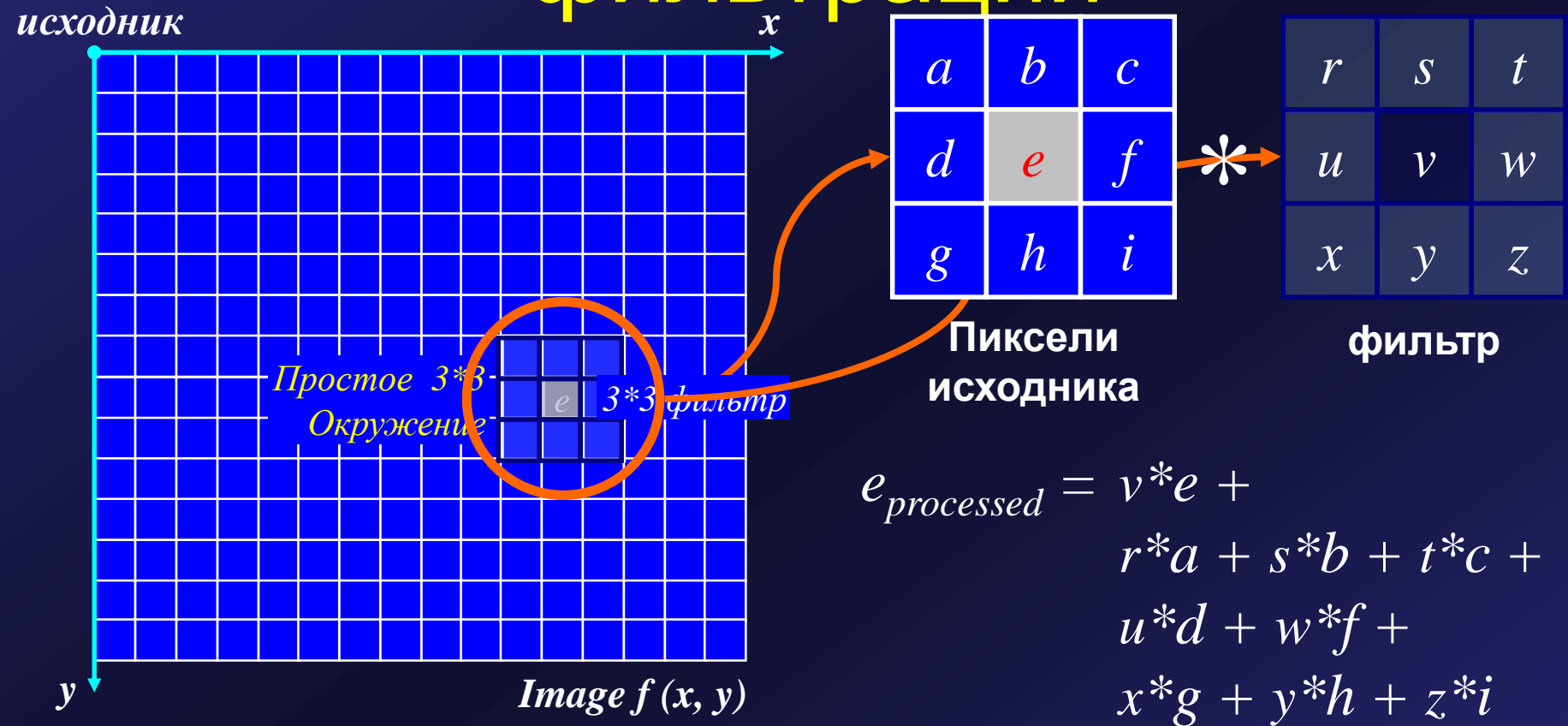


123	127	128	119	115	130
140	145	148	153	167	172
133	154	183	192	194	191
194	199	207	210	198	195
164	170	175	162	173	151

улучшенное изображение

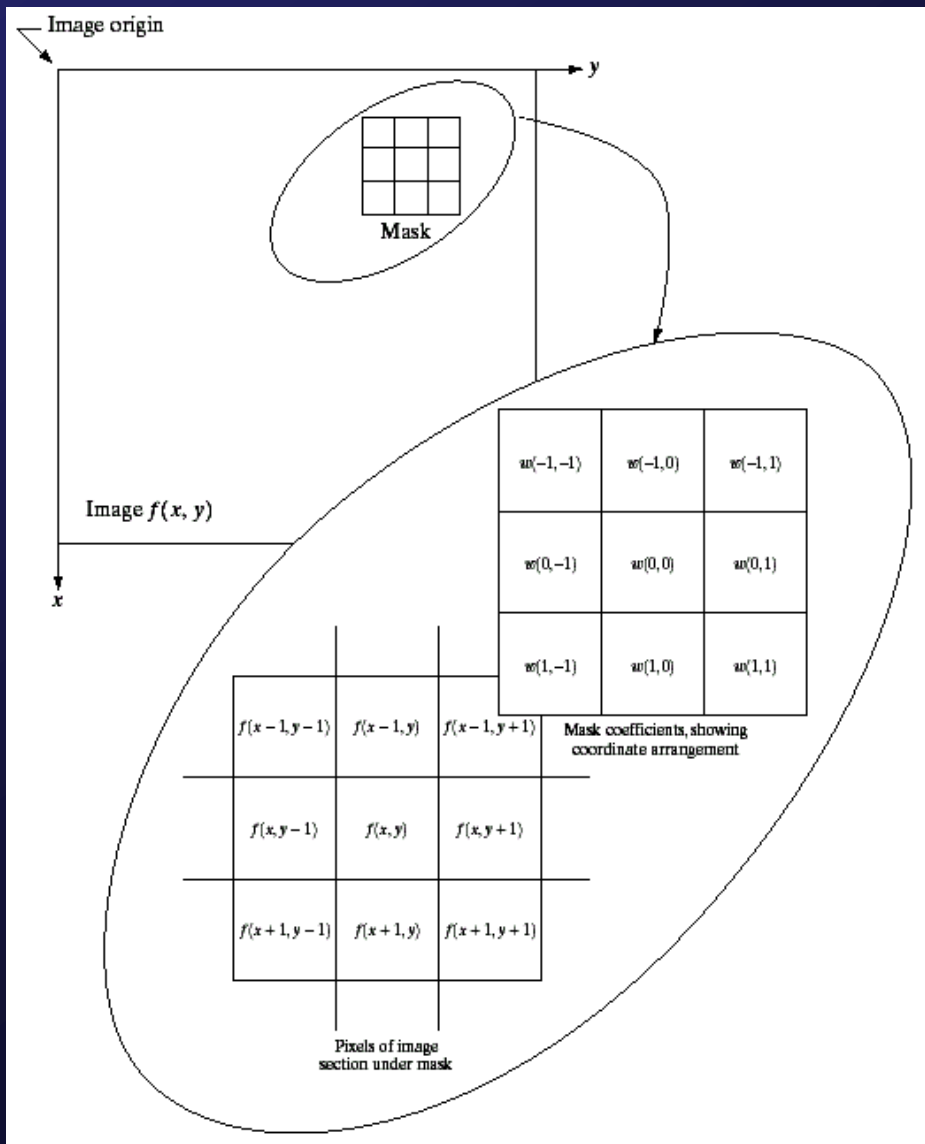


Процесс пространственной фильтрации



повторяется для каждого пикселя в исходном изображении для генерации фильтрованное изображение

Пространственная фильтрация: форма уравнения



$$g(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x + s, y + t)$$

Фильтрация может быть дано в форме уравнения, как показано выше

Обозначения основаны на изображении, показанном слева

Одна из карт
предыдущего слоя

0	1	1	1	0	0	0
0	0	1	1	1	0	0
0	0	0	1	1	1	0
0	0	0	1	1	0	0
0	0	1	1	0	0	0
0	1	1	0	0	0	0
1	1	0	0	0	0	0

I

Ядро 3x3

1	0	1
0	1	0
1	0	1

K

Одна из карт
сверточного слоя

1	4	3	4	1
1	2	4	3	3
1	2	3	4	1
1	3	3	1	1
3	3	1	1	0

I * K

Обработка соседства

- В отличие от точки (пикселя) обработку

Input

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

Output

	$\frac{12}{9}$				

Сглаживающая фильтрация

- Один из самых простых операций пространственной фильтрации мы можем выполнить это операция сглаживания

- » Просто в среднем все пиксели в окрестности вокруг центрального значения

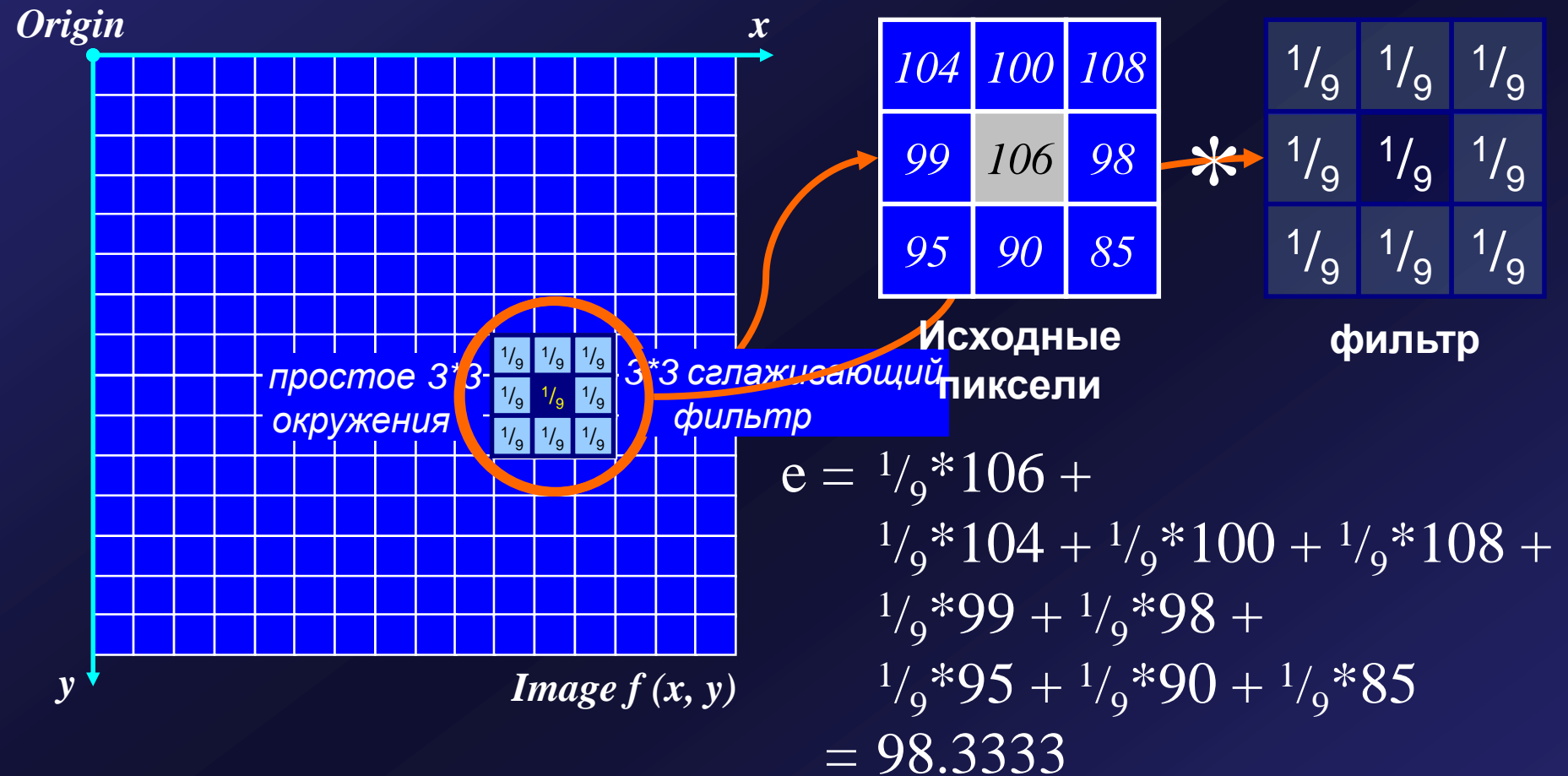
- » Особенно полезен при удалении шума

- » Также полезно для уточнения деталей

$1/9$	$1/9$	$1/9$
$1/9$	$1/9$	$1/9$
$1/9$	$1/9$	$1/9$

Простые
сглаживающие
фильтры

Сглаживающая пространственная фильтрация



повторяется для каждого пикселя в исходном изображении для создания сглаженного изображения

output

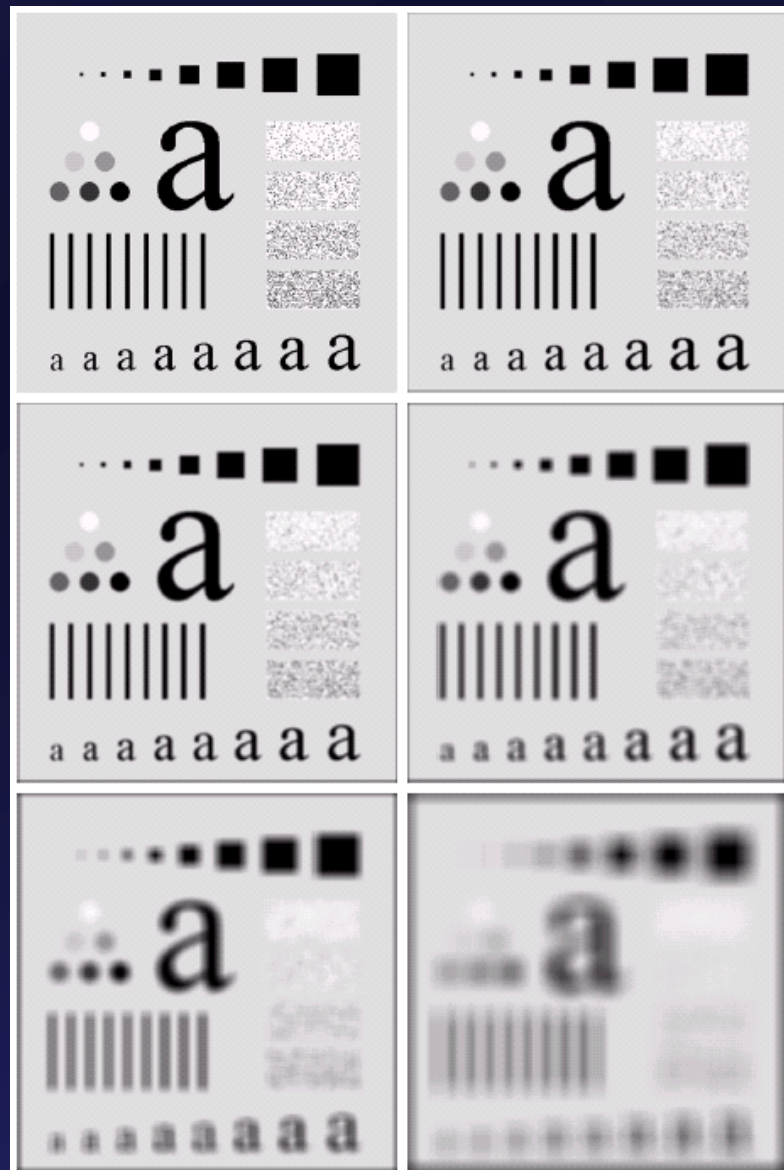
0	-1	0
-1	3	-1
0	-1	0

7	0	0	0	0	7
0	0	4	3	3	0
0	4	3	2	2	4
0	3	3	2	2	3
0	4	3	2	2	4
7	0	0	0	0	7

input

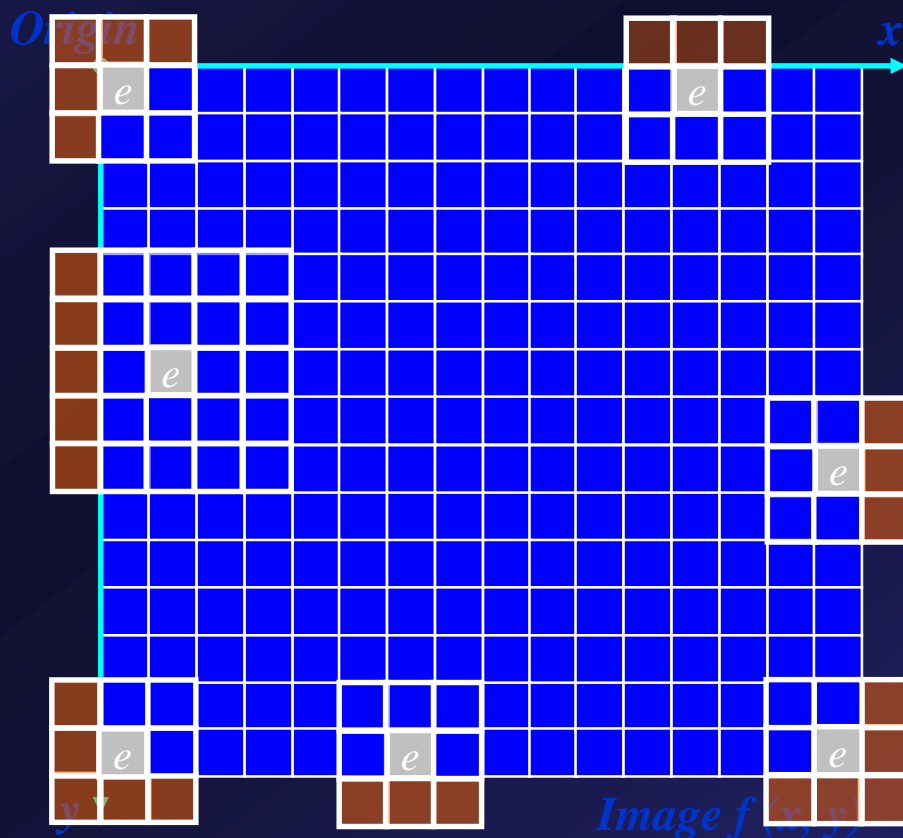
Пример сглаживания изображений

- Изображение в левом верхнем углу является оригинальным с размером 500 * 500 пикселей
- Последующие изображения показать изображение после Фильтрации с усреднением фильтра с возрастающим размером
- 3, 5, 9, 15 и 35
- Обратите внимание, как меняются детали



Краевые эффекты

По краям изображения нам не хватает пикселей для формирования окрестности



$$x^l = f(x^{l-1} * k^l + b^l),$$

где x^l – выход слоя l ;

$f()$ – функция активации;

b^l – коэффициент сдвига слоя l ;

$*$ – операция свертки входа x с ядром k .

$$x_j^l = f\left(\sum_i x_i^{l-1} * k_j^l + b_j^l\right),$$

где x_j^l – карта признаков j (выход слоя l);

$f()$ – функция активации;

b^l – коэффициент сдвига слоя l для карты признаков j ;

k_j^l – ядро свертки j карты, слоя l ;

$*$ – операция свертки входа x с ядром k .

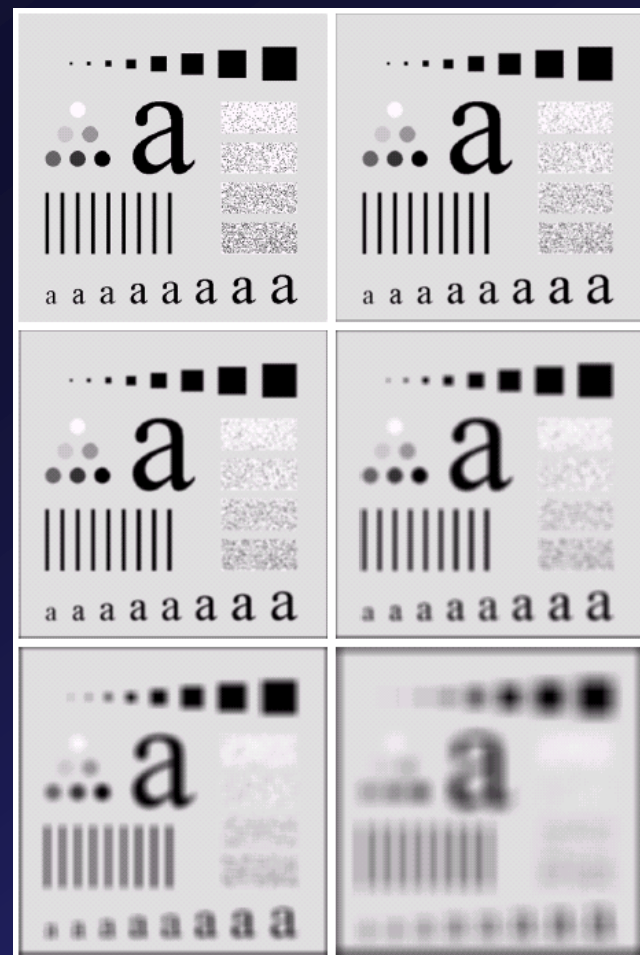
С учетом
краевых
эффектов
размер
исходных
матриц
уменьшается

Применение сглаживания

- Размывание удаляет особенности или другие детали
- Степень размытия = размер ядра

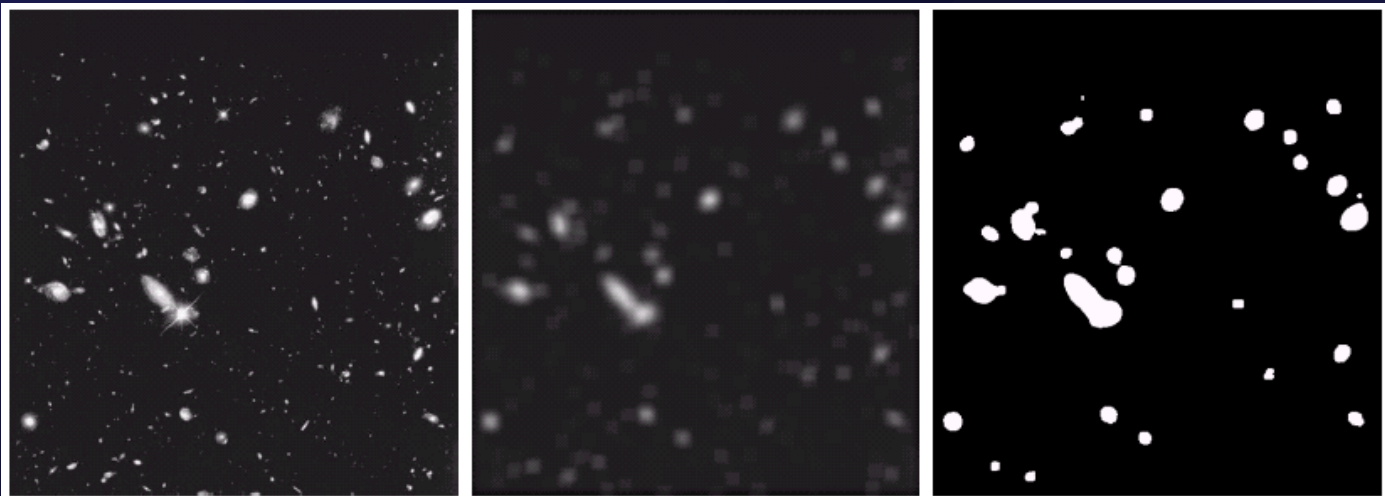


Show: camera, mean, convolution



Применение

- Предварительная обработка: повысить объекты
- размытость + бинаризация



a b c

FIGURE 3.36 (a) Image from the Hubble Space Telescope. (b) Image processed by a 15×15 averaging mask. (c) Result of thresholding (b). (Original image courtesy of NASA.)

Применение

- Удаление шума



Проблемы на границе

- Почему вывод изображения меньше исходного?
- Нам не хватает информации
- Чем больше ядро, тем больше проблема
- Какое это имеет значение?

Да, если мы собираемся после объединить изображения

Input

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

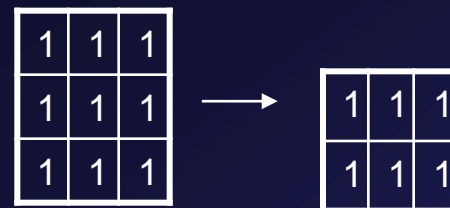
Output

	$\frac{12}{9}$	$\frac{11}{9}$	$\frac{14}{9}$		
	$\frac{13}{9}$	$\frac{11}{9}$	$\frac{13}{9}$		
	$\frac{16}{9}$	$\frac{12}{9}$	$\frac{11}{9}$		

Проблемы на границе

Решения

- » Добавить значение: 0, 255, соседний (вход / выход)
 - Изменение гистограммы, совсем другое значение, новую модель, и т.д.
- » Обрезать ядро: $3 \times 3 \Rightarrow$ например 2×3 :
 - Последовательность действий, а не четко определены



Input

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

Output

	$\frac{12}{9}$	$\frac{11}{9}$	$\frac{14}{9}$		
	$\frac{13}{9}$	$\frac{11}{9}$	$\frac{13}{9}$		
	$\frac{16}{9}$	$\frac{12}{9}$	$\frac{11}{9}$		

Проблемы на границе

- Решение в расширении границ

Циркулярная индексация

1
2
Отражение
Reflected (mirrored)

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

Input

3	7	1	2	0	1	3	7
		2	1	4	2	2	4
		1	0	1	0	1	3
		1	2	1	0	2	3
		2	5	3	1	2	2
		4	5	2	1	1	1

Input

Корреляция & Конволюция

- Фильтрацию называют корреляцией с фильтром, сама называется корреляцией ядра
- Свертывание является подобная операция, с помощью только одного тонкое различие

a	b	c
d	e	e
f	g	h

Original Image
Pixels

$*$

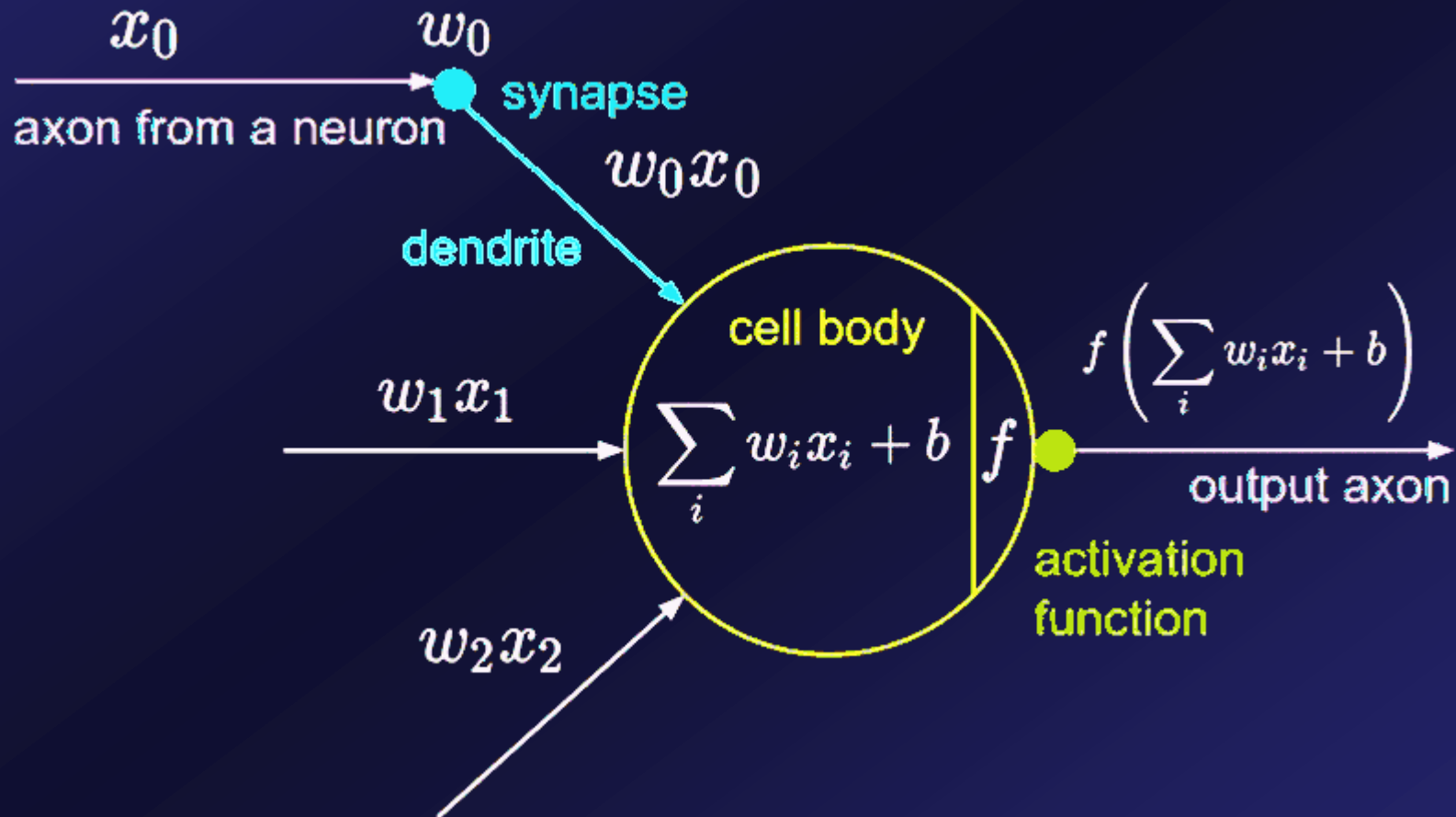
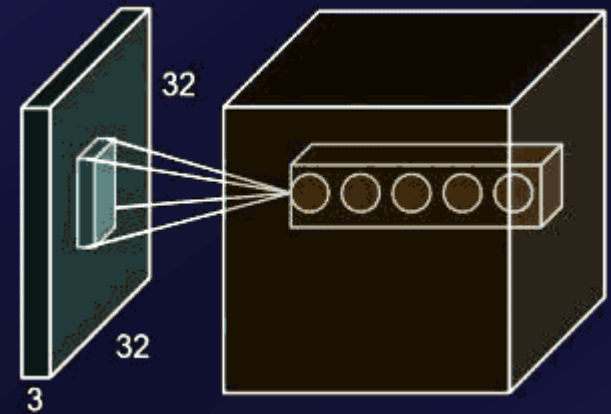
r	s	t
u	v	w
x	y	z

Filter

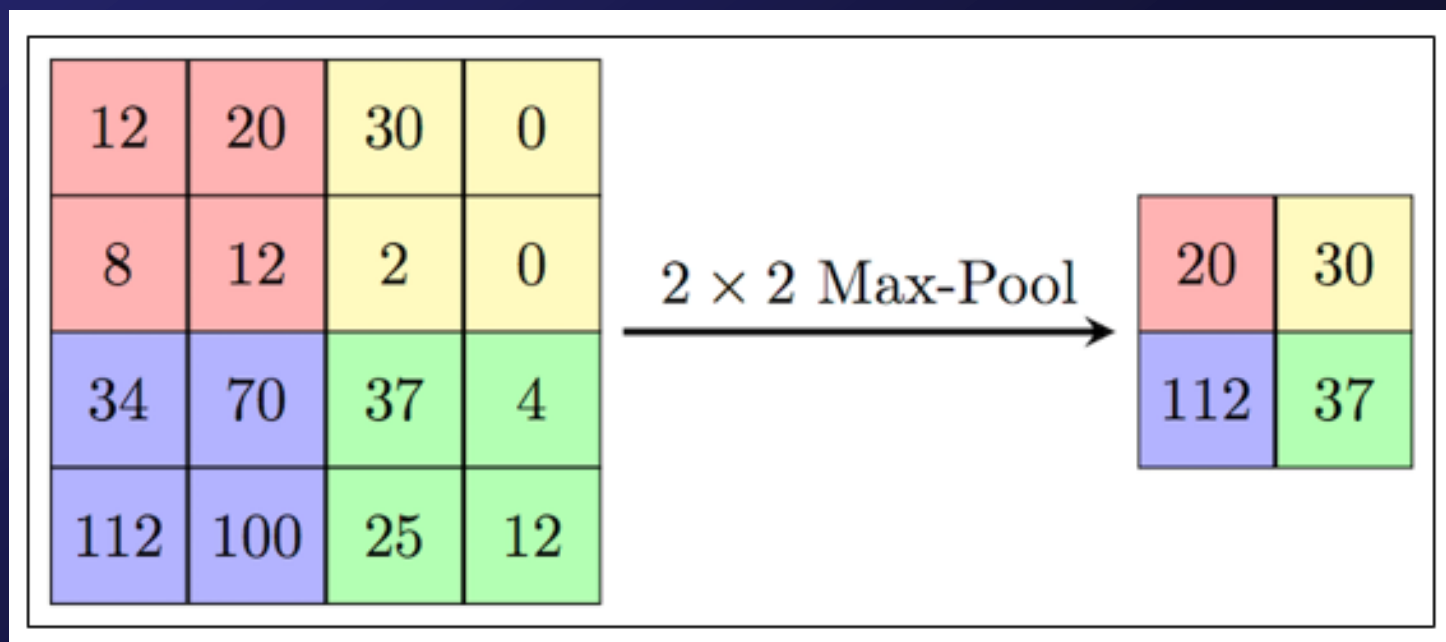
$$e_{processed} = v * e + \\ z * a + y * b + x * c + \\ w * d + u * e + \\ t * f + s * g + r * h$$

- Для симметричных фильтров это не имеет никакого значения

Нейронные Сети

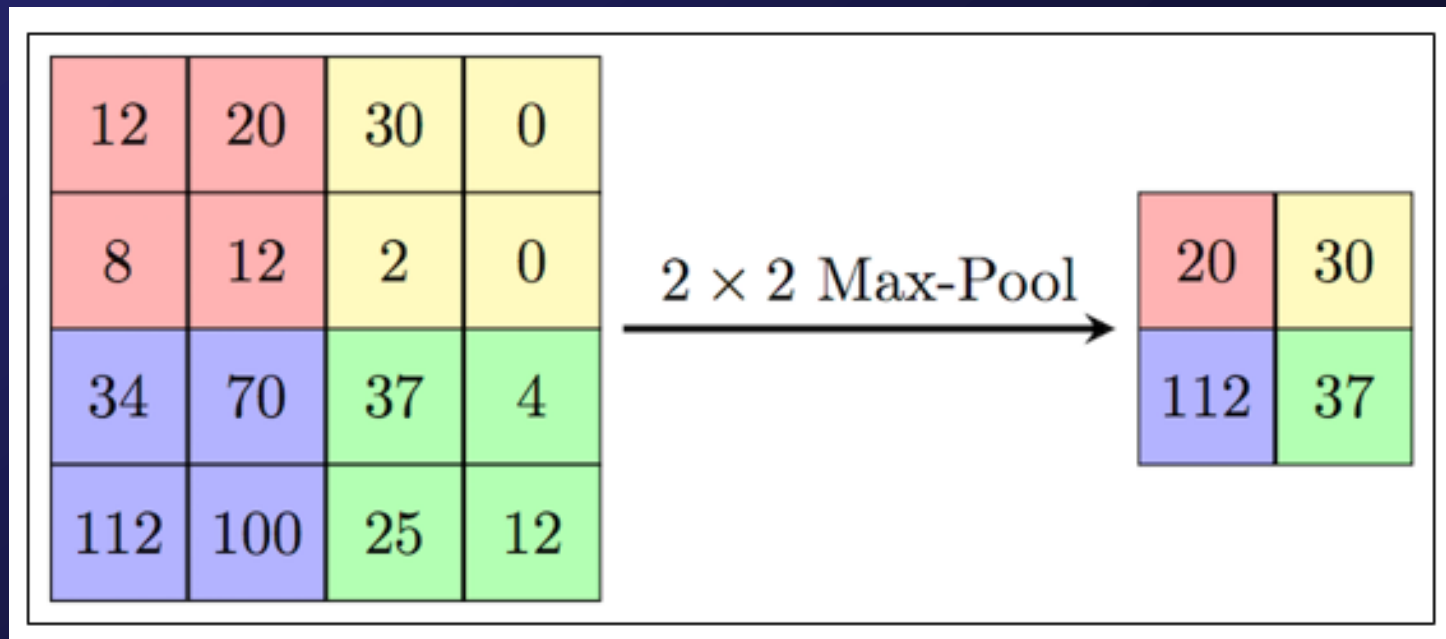


Подвыборочный слой CNN



В процессе сканирования ядром подвыборочного слоя (фильтром) карты предыдущего слоя, сканирующее ядро не пересекается в отличие от сверточного слоя. Обычно, каждая карта имеет ядро размером 2×2 , что позволяет уменьшить предыдущие карты сверточного слоя в 2 раза. Вся карта признаков разделяется на ячейки 2×2 элемента, из которых выбираются максимальные по значению.

Подвыборочный слой CNN



$$x^l = f(a^l * \text{subsample}(x^{l-1}) + b^l),$$

где x^l – выход слоя l ;

$f()$ – функция активации;

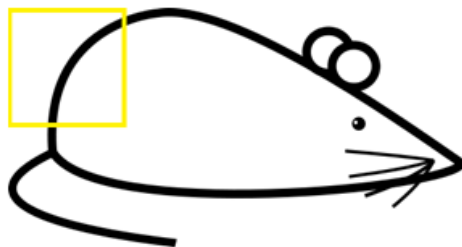
a^l, b^l – коэффициенты сдвига слоя l ;

$\text{subsample}()$ – операция выборки локальных максимальных значений.

Нейронные Сети CNN



Original image



Visualization of the filter on the image



Visualization of the receptive field

0	0	0	0	0	0	30
0	0	0	0	50	50	50
0	0	0	20	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0

Pixel representation of the receptive field

*

0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0

Pixel representation of filter

Multiplication and Summation = $(50*30)+(50*30)+(50*30)+(20*30)+(50*30) = 6600$ (A large number!)



Visualization of the filter on the image

0	0	0	0	0	0	0
0	40	0	0	0	0	0
40	0	40	0	0	0	0
40	20	0	0	0	0	0
0	50	0	0	0	0	0
0	0	50	0	0	0	0
25	25	0	50	0	0	0

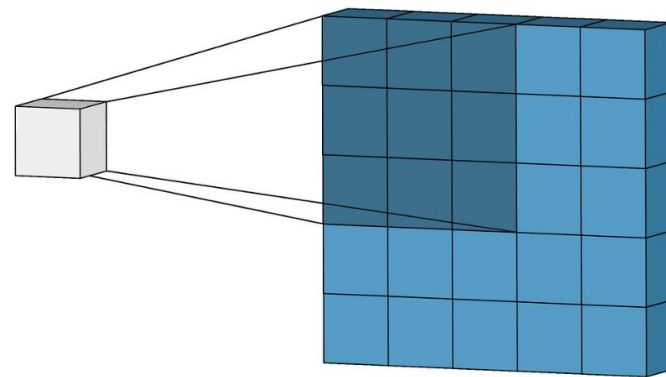
Pixel representation of receptive field

*

0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0

Pixel representation of filter

Multiplication and Summation = 0

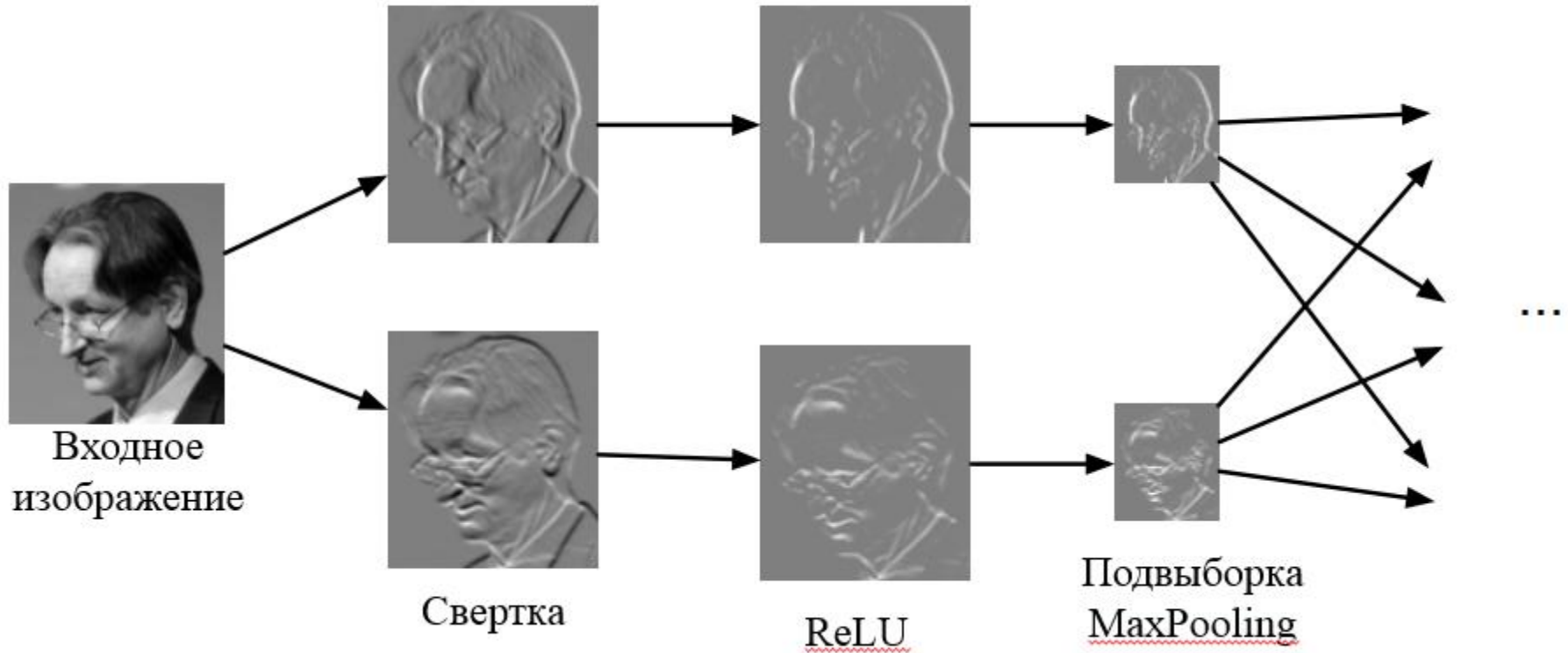


3 ₀	3 ₁	2 ₂	1	0
0 ₂	0 ₂	1 ₀	3	1
3 ₀	1 ₁	2 ₂	2	3
2	0	0	2	2
2	0	0	0	1

12.0	12.0	17.0
10.0	17.0	19.0
9.0	6.0	14.0

Нейронные Сети

Операция свертки + ReLU + подвыборка





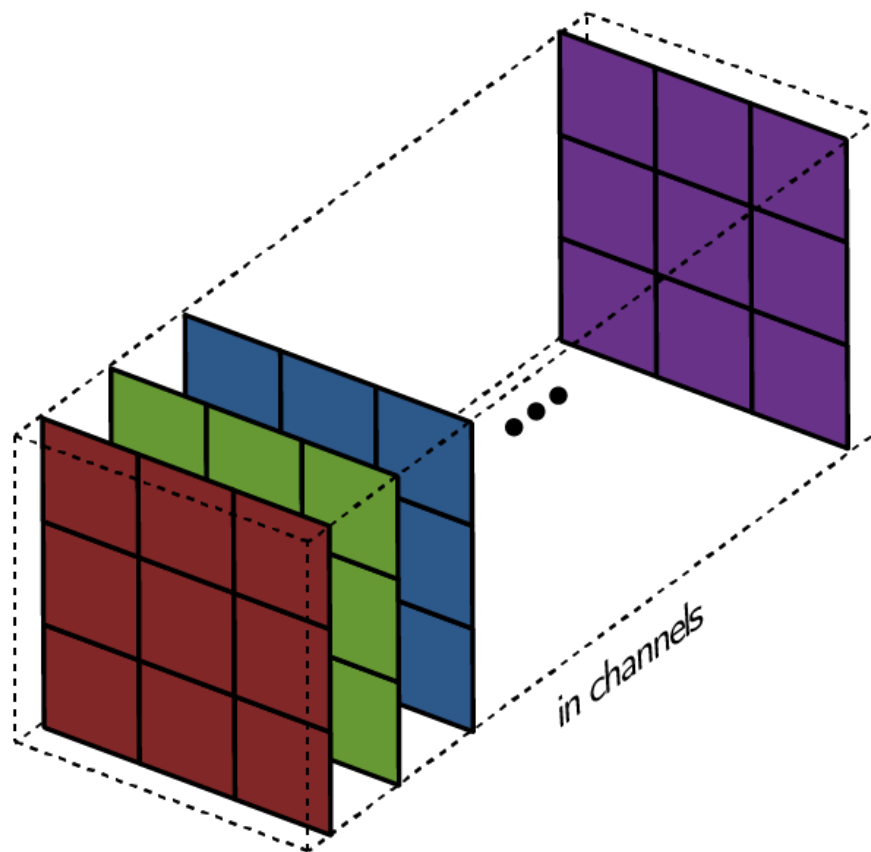
Red



Green



Blue





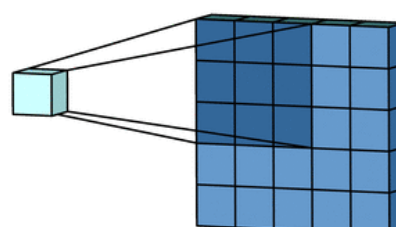
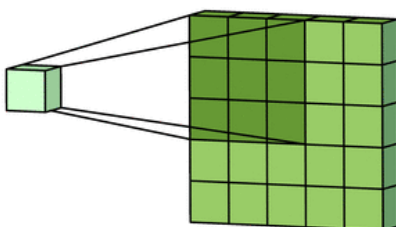
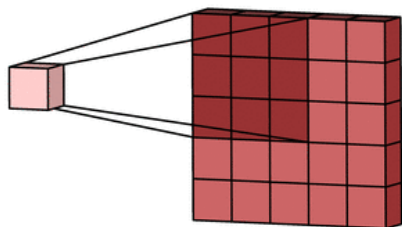
Red



Green



Blue





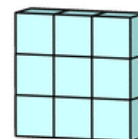
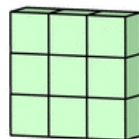
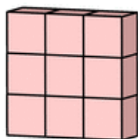
Red



Green



Blue





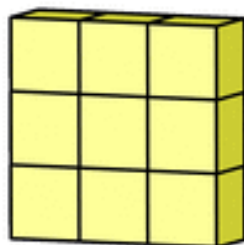
Red



Green



Blue



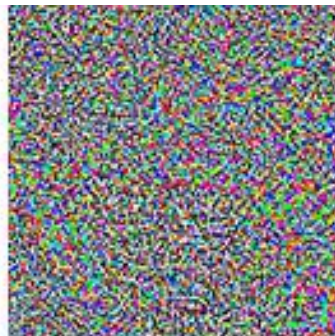
Обобщающие процессы свертки имеет свою обратную сторону – возможность подделки изображений под удовлетворение особенностей распознающих фильтров. На изображениях ниже человек в обоих случаях узнает фотографии панды. А сверточные нейросети можно запутать, добавив шум, подстроенный под фильтры распознавания других образов.



"panda"

57.7% confidence

+ ϵ



=



"gibbon"

99.3% confidence

Заключение

- В этой лекции мы рассмотрели идеи пространственной фильтрации и, в частности:
 - » Район операции
 - » Процесс фильтрации
 - » сглаживающих фильтров
 - » Решение проблем по краям изображения при использовании фильтрации
 - » Корреляцию и Конволюцию