

Нейронные сети в машинном обучении

Лекция 3
Сверточные нейронные сети



Содержание

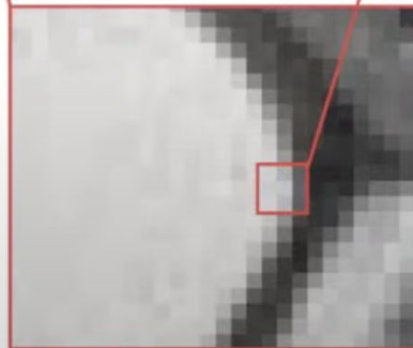
1. Проблемы полносвязных нейронных сетей
2. Сверточные нейронные сети
3. Интерпретация обученных моделей
4. Transfer learning
5. Домашнее задание

Как устроены изображения: черно-белые



[214, 176, 90]
[207, 204, 97]
[218, 186, 93]

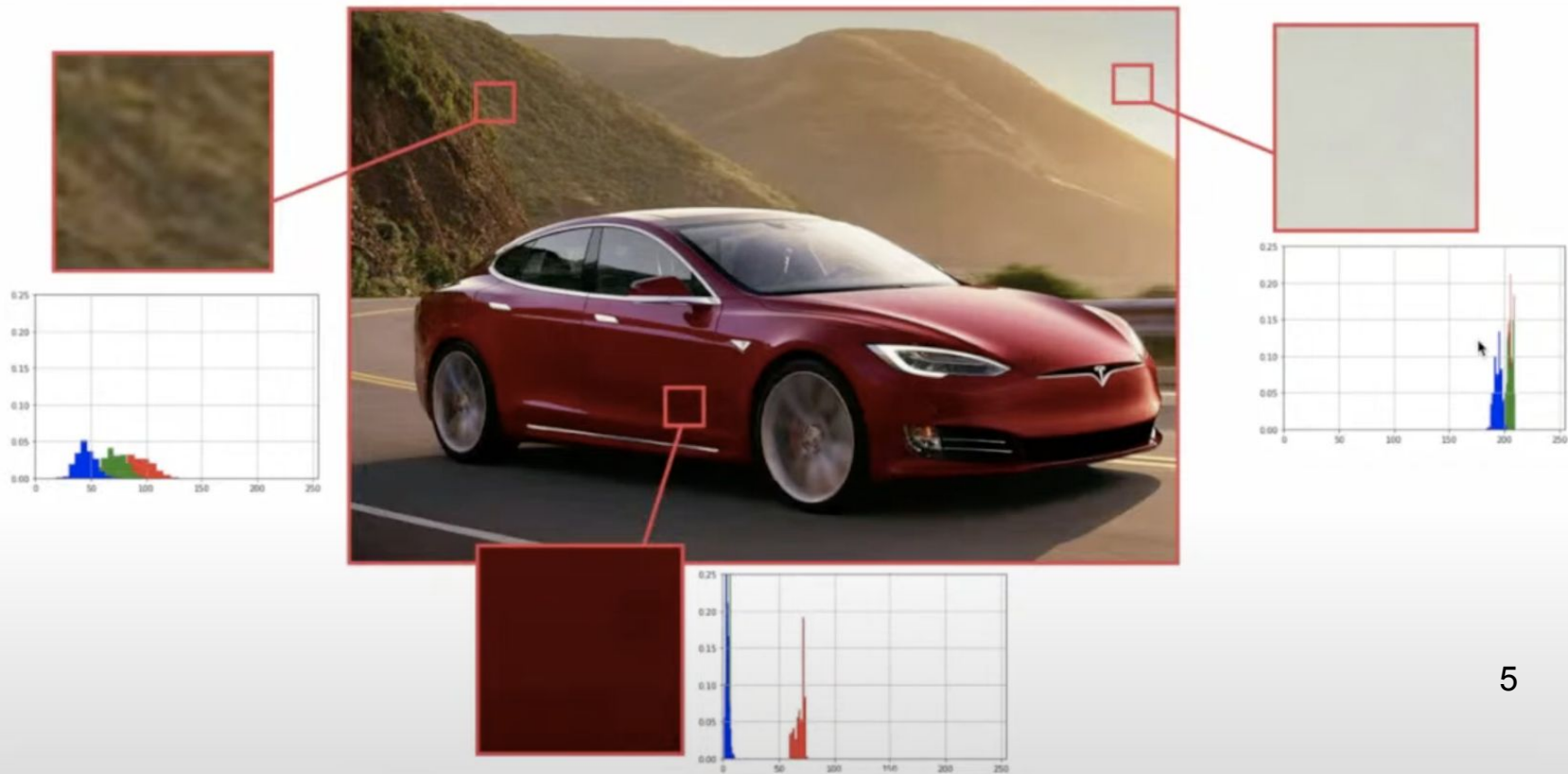
214	176	90
207	204	97
218	186	93



Как устроены изображения: цветные



Как устроены изображения: цветные



Datasets

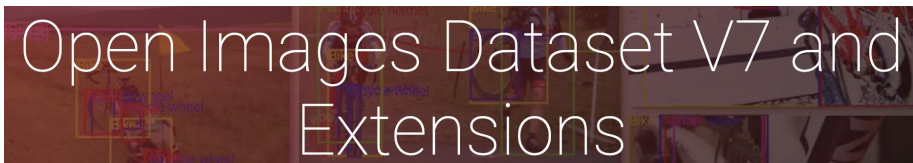


Classes: 1000

Training images: 1,128,167

Validation images: 50,000

Test images: 100,000



Classes: 600

Training images: 14,610,229

Validation images: 303,980

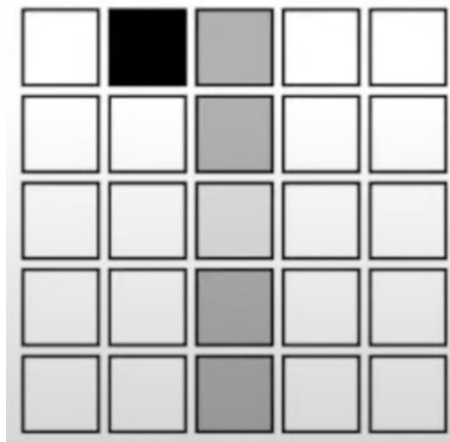
Test images: 937,237

Признаки - это яркости всех пикселей

Хотим сделать классификатор grayscale-изображений цифр от 0 до 9:

- Размер изображения: 5 x 5 пикселей
- Используем полносвязную сеть с одним скрытым слоем

Какова размерность весов этого слоя?



Признаки - это яркости всех пикселей

Хотим сделать классификатор grayscale-изображений цифр от 0 до 9:

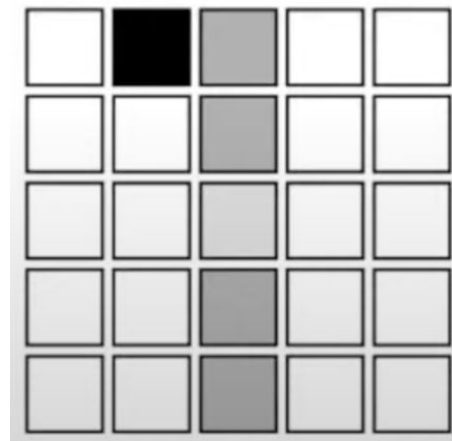
- Размер изображения: 5 x 5 пикселей
- Используем полносвязную сеть с одним скрытым слоем

Какова размерность весов этого слоя?

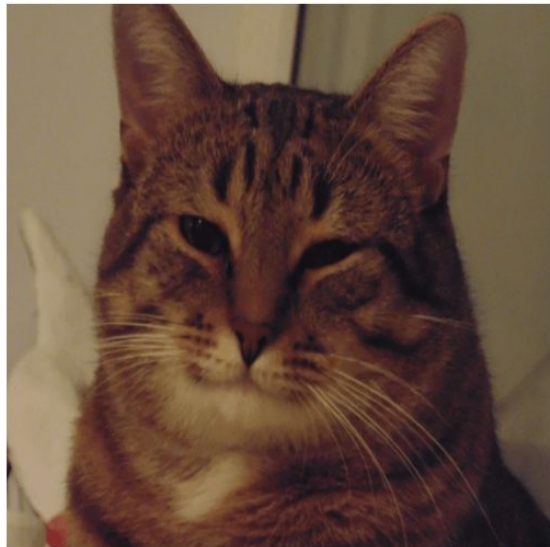
Число входов = 25

Число выходов = 10

Число весов = 260



Проблемы полносвязных сетей



$h = 224$

$w = 224$

1. Требуется огромное количество нейронов

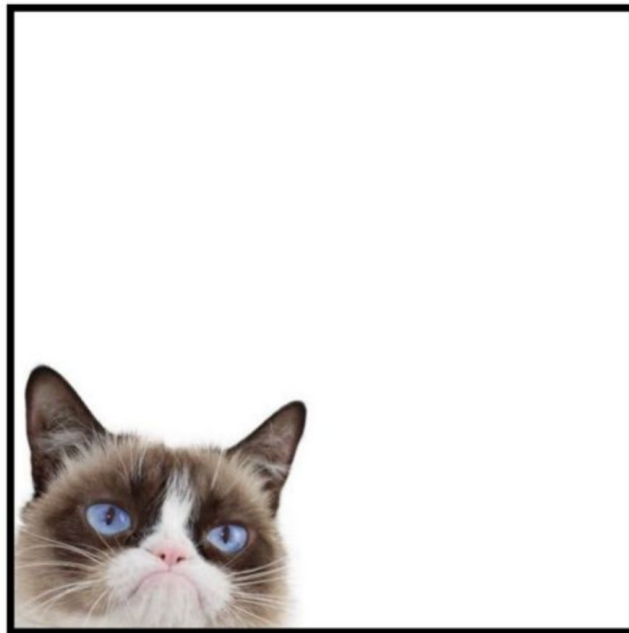
ImageNet: 1000 классов

Число параметров однослойной сети:

$224 \times 224 \times 3 \times 1000 \sim 150\,000\,000$

Проблемы полносвязных сетей

2. Нет инвариантности к смещениям: карты активаций различаются



Решение

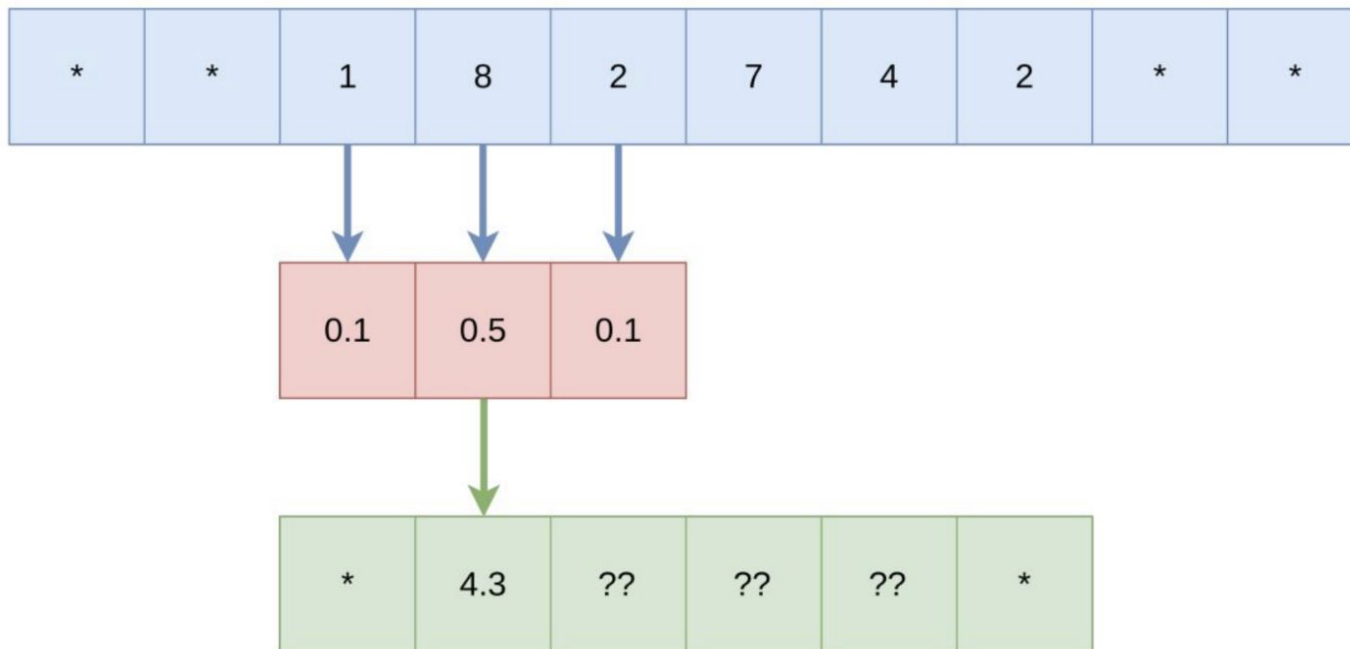
Возможное решение – введение новых типов слоев:

- Сверточные слои
- Пулинг
- Dropout (лекция 5)
- Нормализация (лекция 5)
- ...

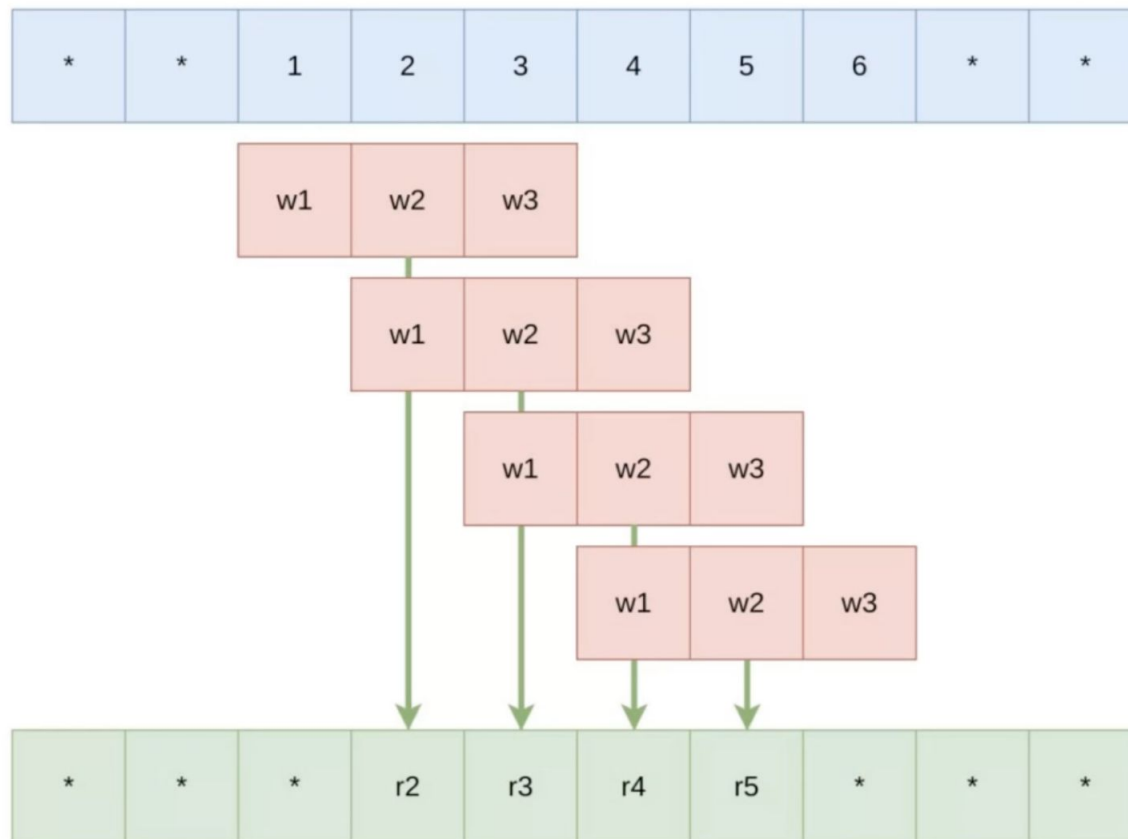
Операция свертки (одномерный случай)

Определение

Результатом операции свертки массива m с ядром a называется сигнал n : $n[k] = \sum_{i=-w}^w m[k+i]a[-i]$. Обозначение: $n = m * a$



Операция свертки (одномерный случай)



Операция свертки на ч/б изображениях (двумерный случай)

Шаг 1

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

Receptive field

1	0
0	1

Фильтр (=ядро, =kernel)

1	2
3	4

(траектория)

5		

...

Шаг 4

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

Receptive field

1	1
0	1

Фильтр

1	2
3	4

5	9	4
5	7	

...

Шаг 8

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

Receptive field

1	1
0	1

Фильтр

1	2
3	4

5	9	4
5	7	

Инвариантность свертки к сдвигам

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

Фильтр (=ядро, =kernel)

*

1	0
0	1

=

0	0	0
0	1	0
0	0	2

Интуитивно: область изображения, которая сильно откликается на фильтр

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

Фильтр

*

1	0
0	1

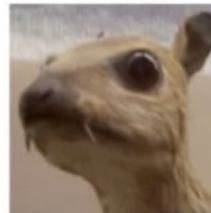
=

2	0	0
0	1	0
0	0	0

Примеры фильтров (ядер)

► Тожественное

0	0	0
0	1	0
0	0	0



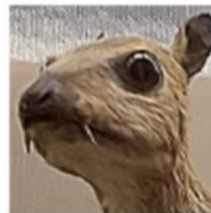
► Детектор границ

0	1	0
1	-4	1
0	1	0



► Увеличение резкости

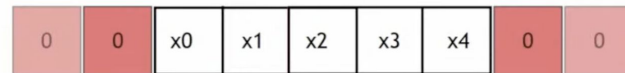
0	1	0
1	5	1
0	1	0



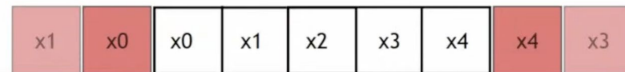
Padding и stride

Для 1D-свертки:

- Zero padding

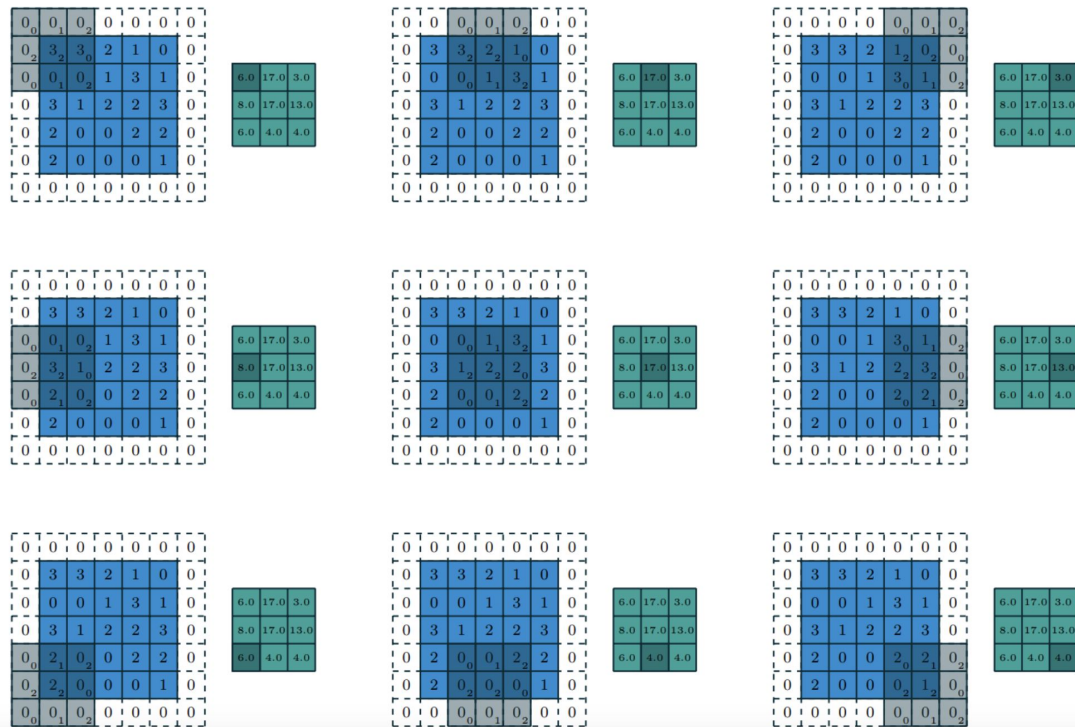


- Reflect padding

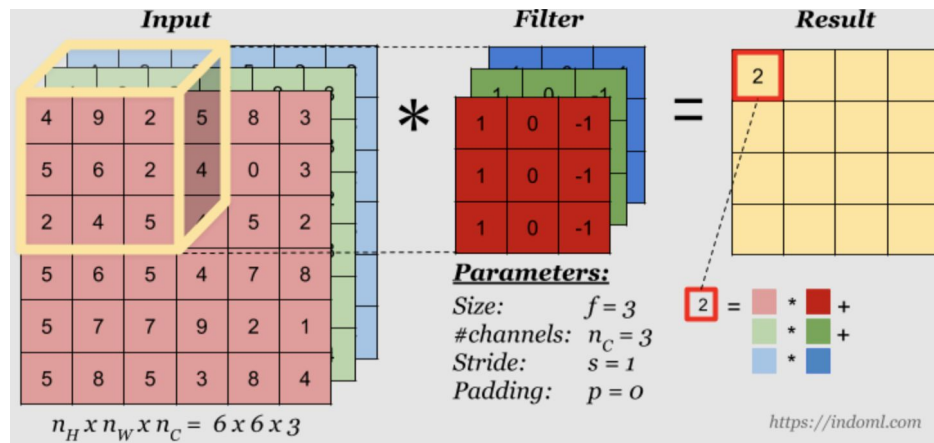
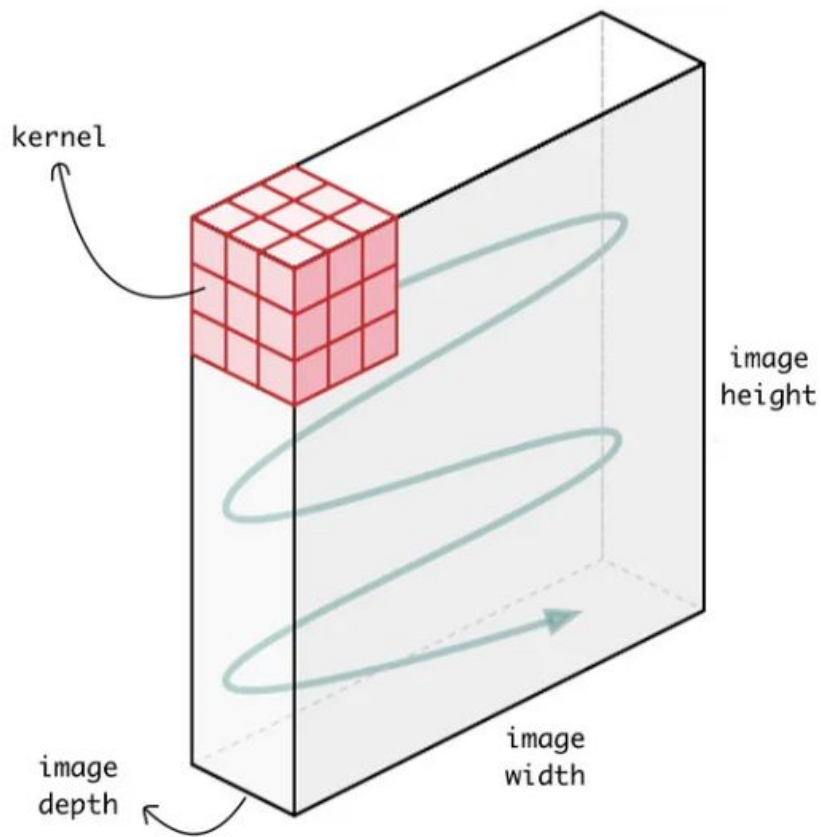


padding = дополнение (отступ)
изображения
stride = шаг свертки

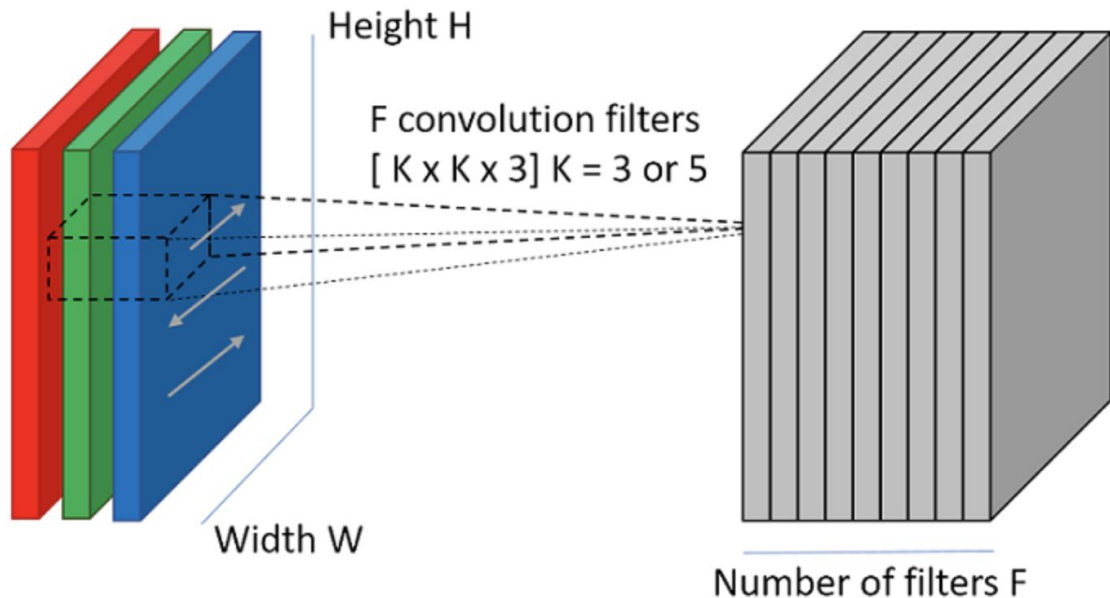
Для 2D-свертки:



Операция свертки на цветных изображениях (трехмерный случай)



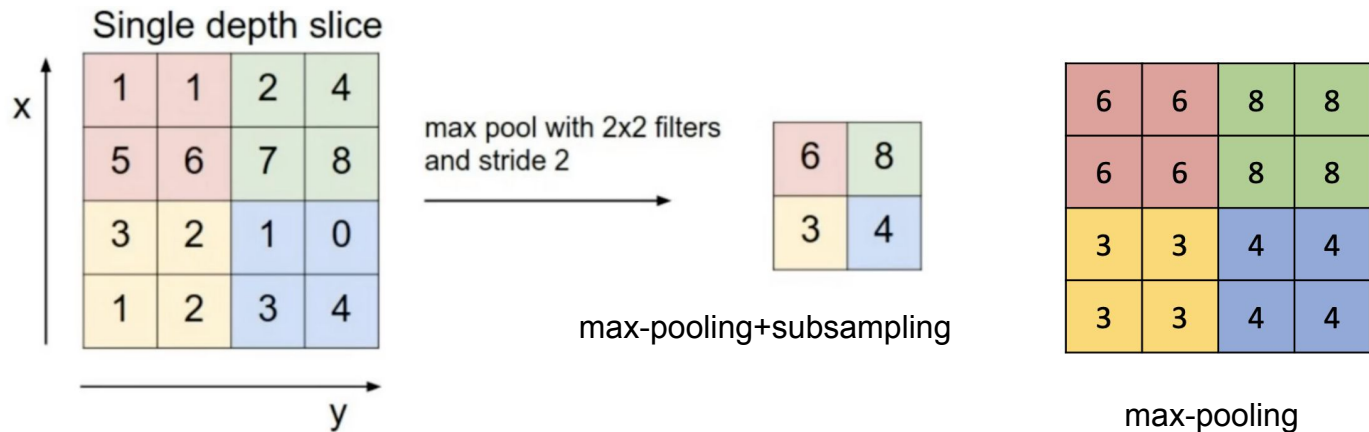
Операция свертки на цветных изображениях (трехмерный случай)



Задается: кол-во входных каналов,
кол-во выходных каналов F ,
stride (шаг свертки),
padding (дополнение изображения),
размер ядра

Pooling

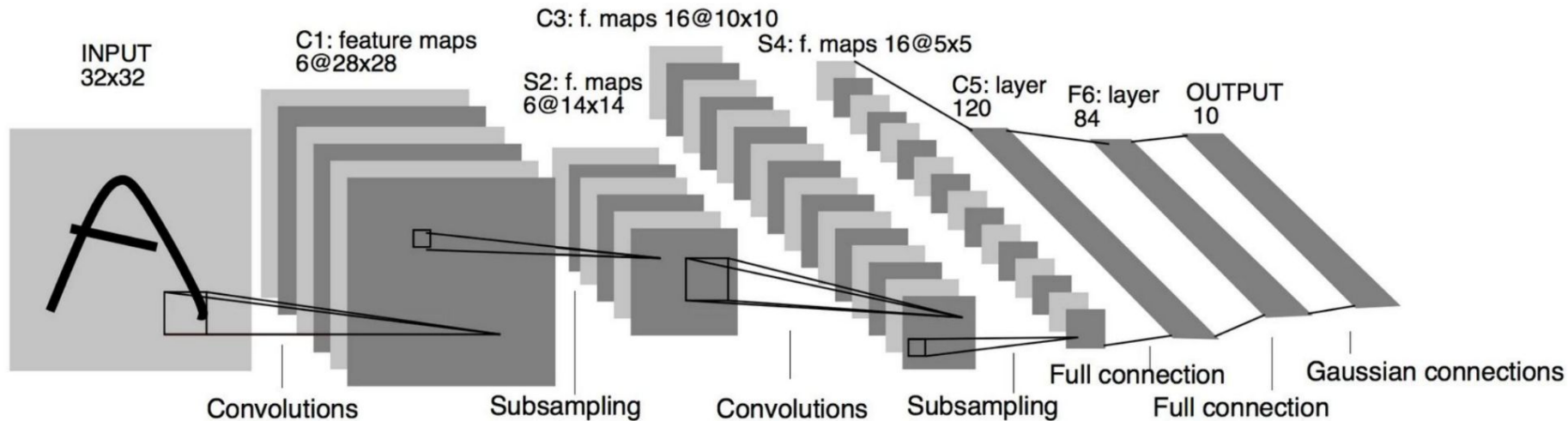
1. Разбиваем картинку фильтром pooling (например, 2x2)
2. Внутри каждой области считаем:
 - максимум (max-pooling);
 - минимум (min-pooling);
 - среднее (mean-pooling)



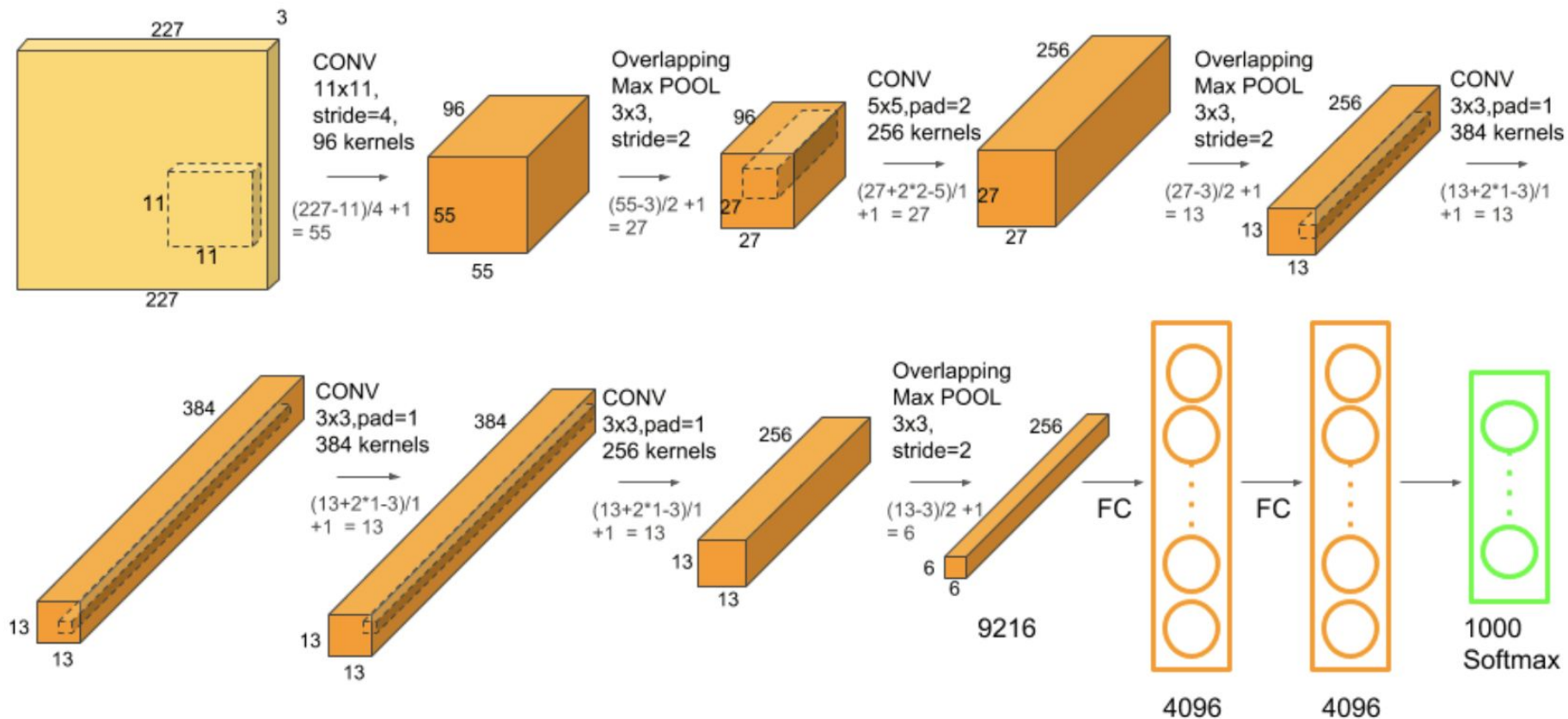
Для чего нужен pooling?

- Сокращение вычислительных затрат;
- Вырабатывается инвариантность к небольшим сдвигам;
- Побеждают наиболее активные нейроны → получаем местонахождение самого сильного отклика на изображении;
- Увеличивает receptive field (важно на последних слоях сверточной сети находить крупные объекты, которые могут занимать всё исходное изображение)

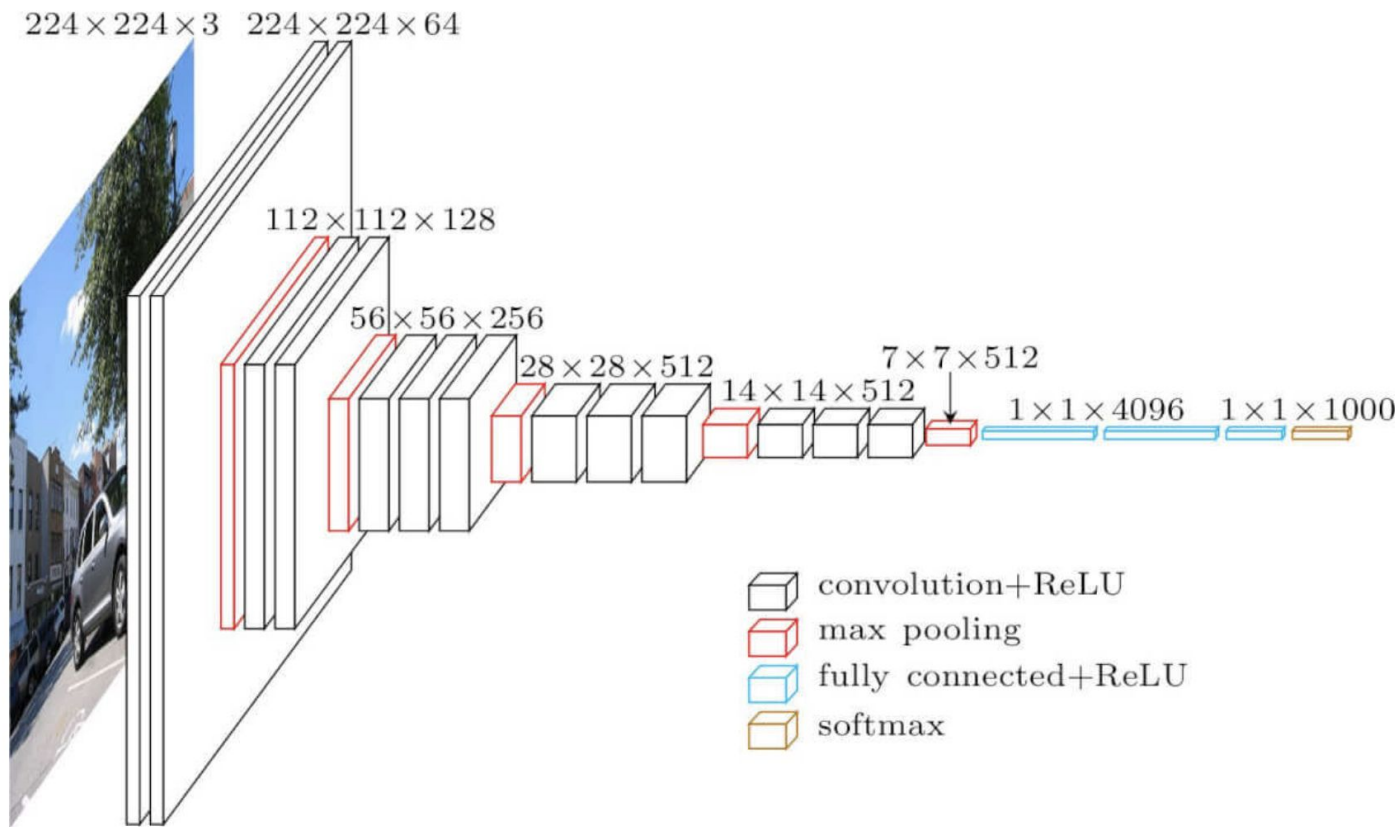
Сверточные нейронные сети: LeNet (1998)



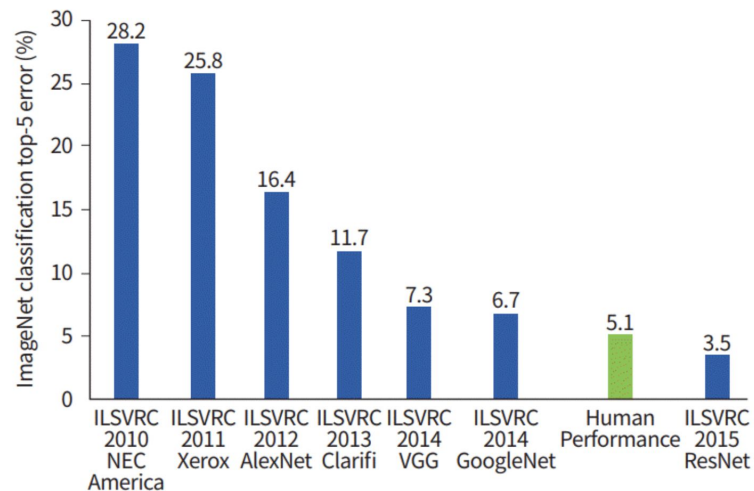
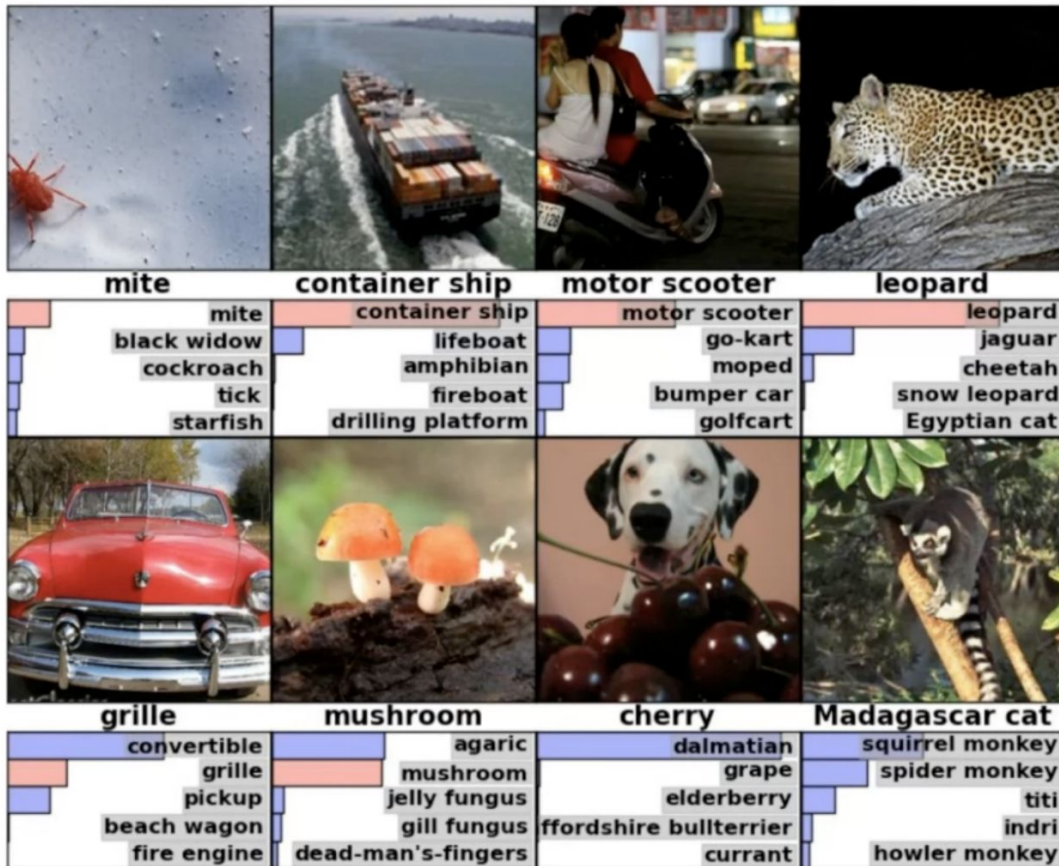
Сверточные нейронные сети: AlexNet (2012)



Сверточные нейронные сети: VGG-16 (2014)



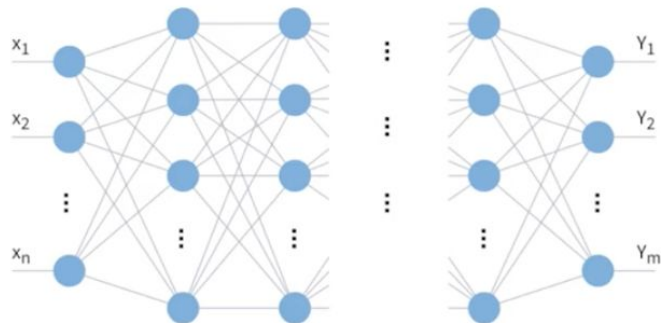
Метрики качества в соревнованиях



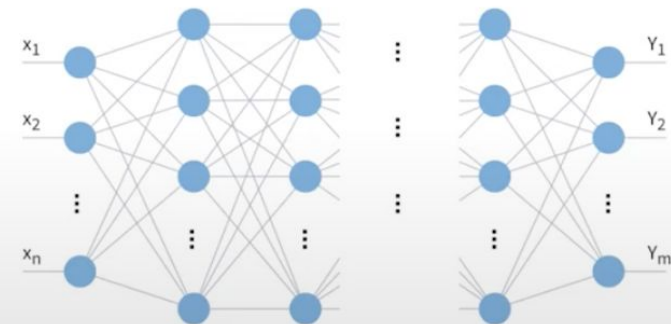
Transfer learning

Обучаем на ImageNet

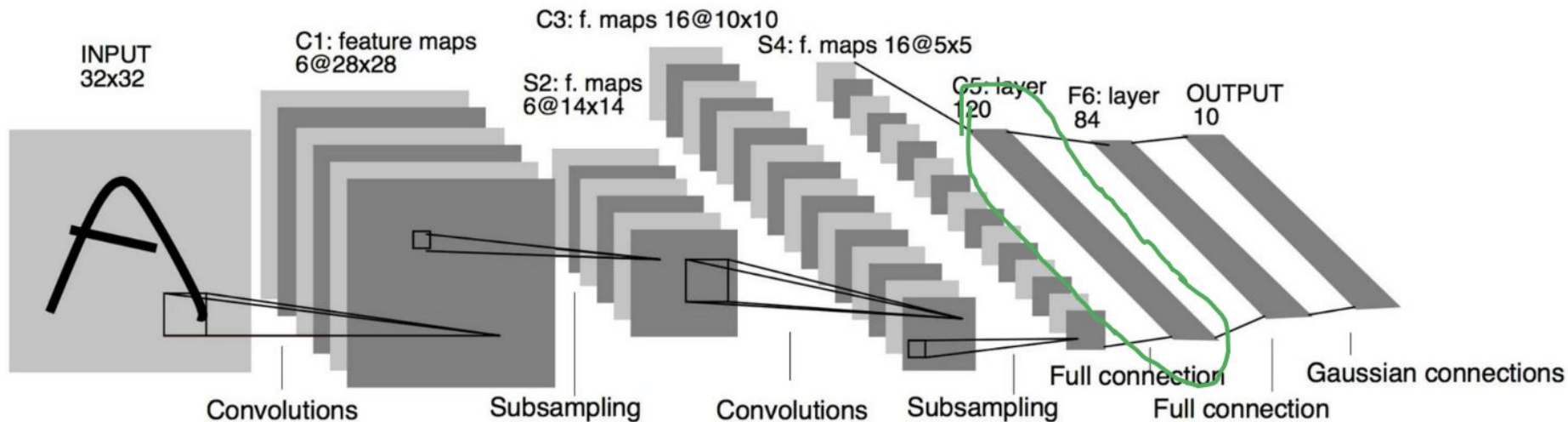
1.



2.

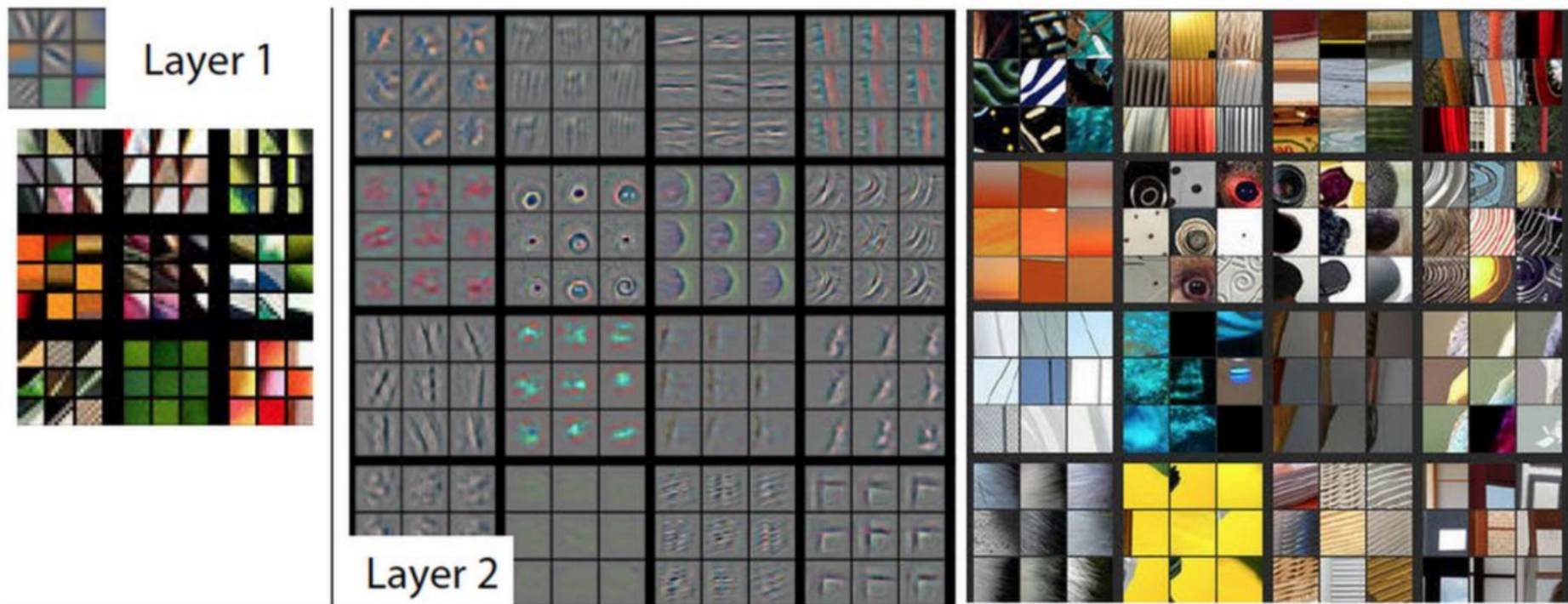


Признаковое представление

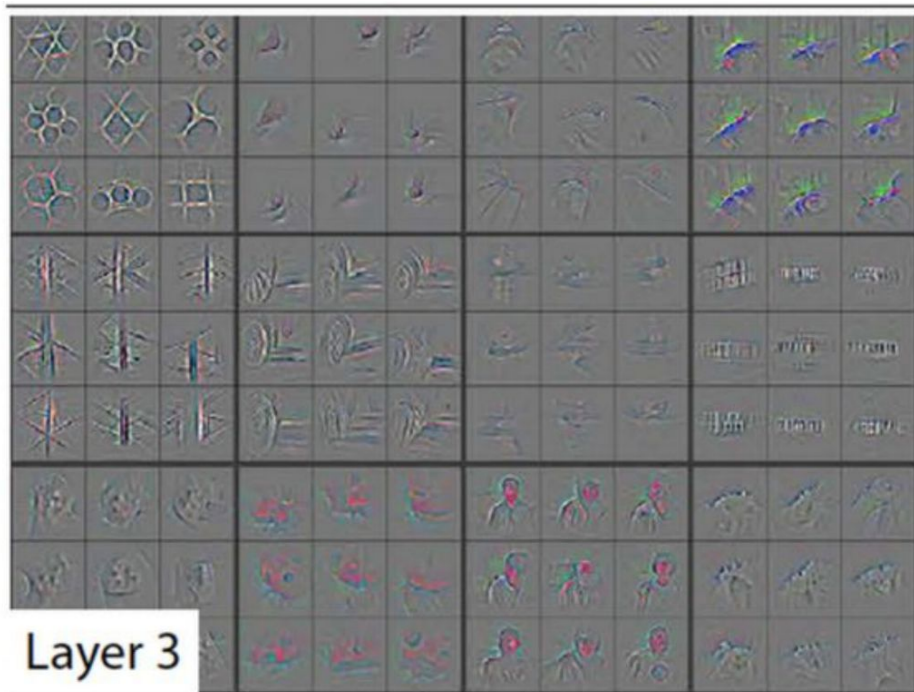


Выходы полносвязных слоев - хорошие признаковые описания изображений (embeddings)

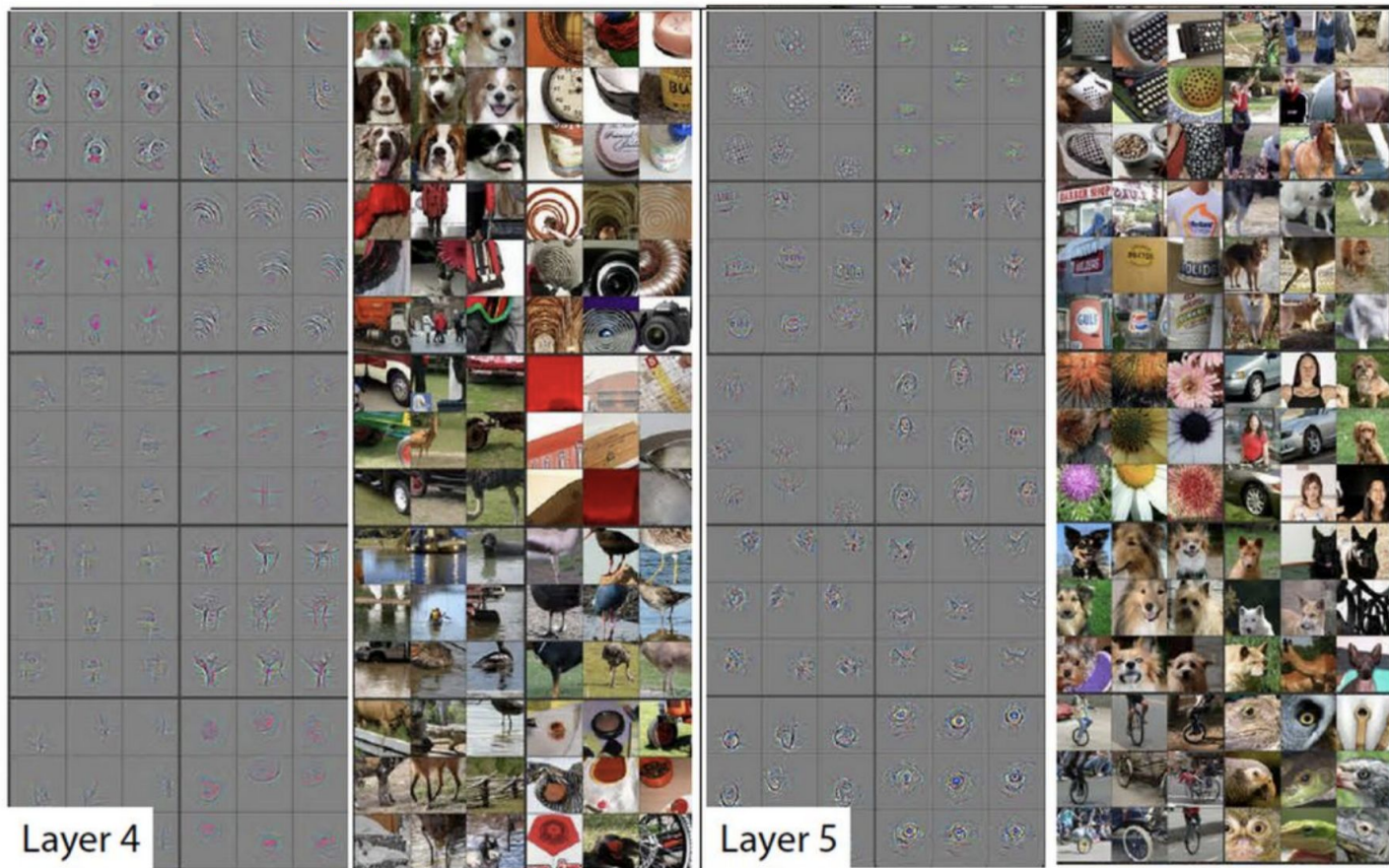
Представления со слоев



Представления со слоев



Представления со слоев



Домашнее задание

Обучить сверточную нейронную сеть с семинара

Спасибо!



Источники

<https://towardsdatascience.com/understanding-1d-and-3d-convolution-neural-network-keras-9d8f76e29610>

<https://medium.com/analytics-vidhya/2d-convolution-using-python-numpy-43442ff5f381>

<https://medium.com/analytics-vidhya/simple-cnn-using-numpy-part-iii-relu-max-pooling-softmax-c03a3377eaf2>

<https://betterdatascience.com/implement-pooling-from-scratch-in-python/>

<https://note.nkmk.me/en/python-opencv-bgr-rgb-cvtColor/>

<https://yann.lecun.com/exdb/publis/pdf/lecun-98.pdf>

<https://arxiv.org/abs/1403.6382>

Материалы курса “Современные методы машинного обучения” НИУ ВШЭ и Deep Learning School