Тверской государственный технический университет

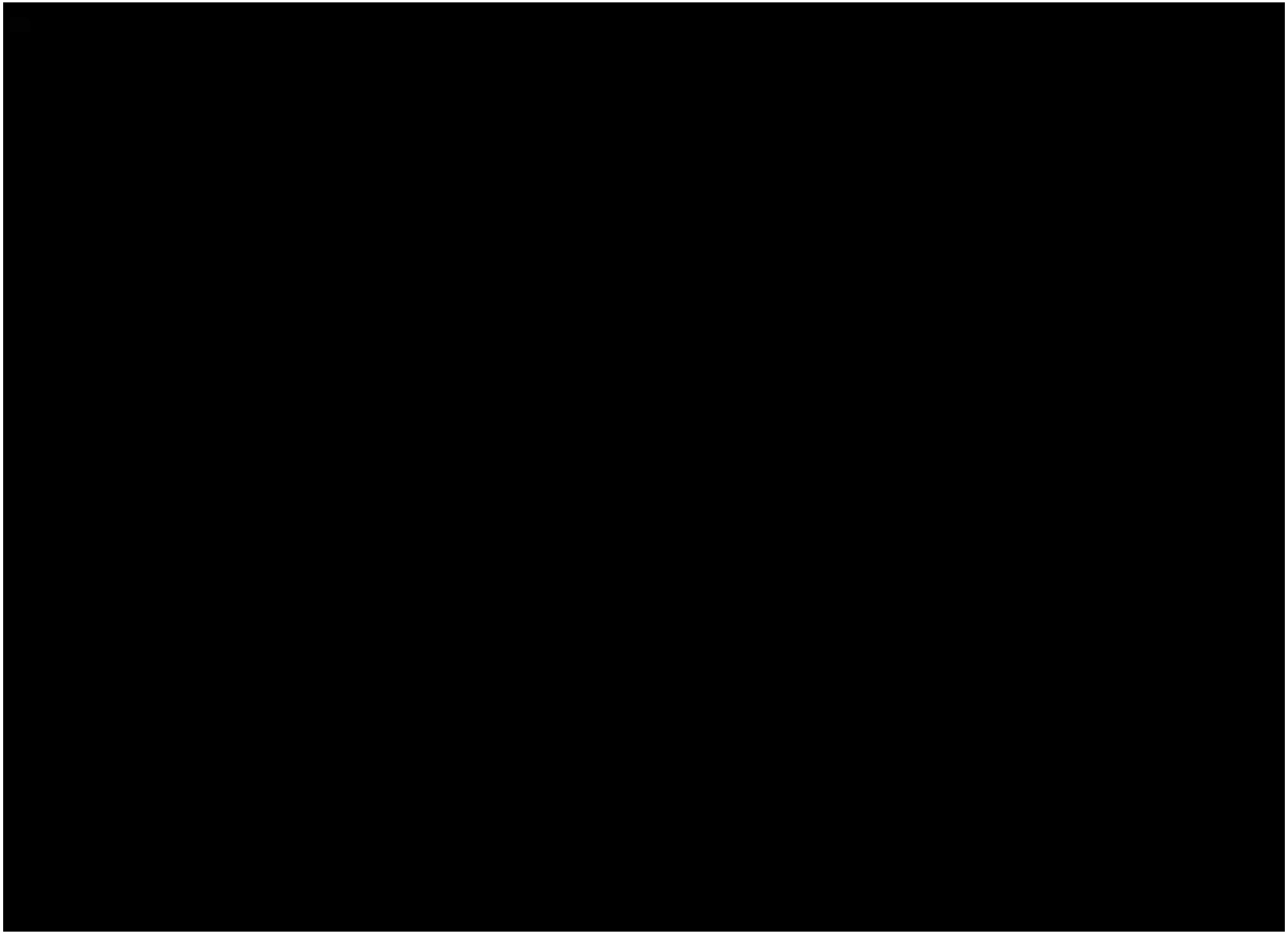
Интеллектуальные информационные системы

Сверточные нейронные сети

Материалы курса доступны по ссылке:
https://github.com/AndreyShpigar/ML-course

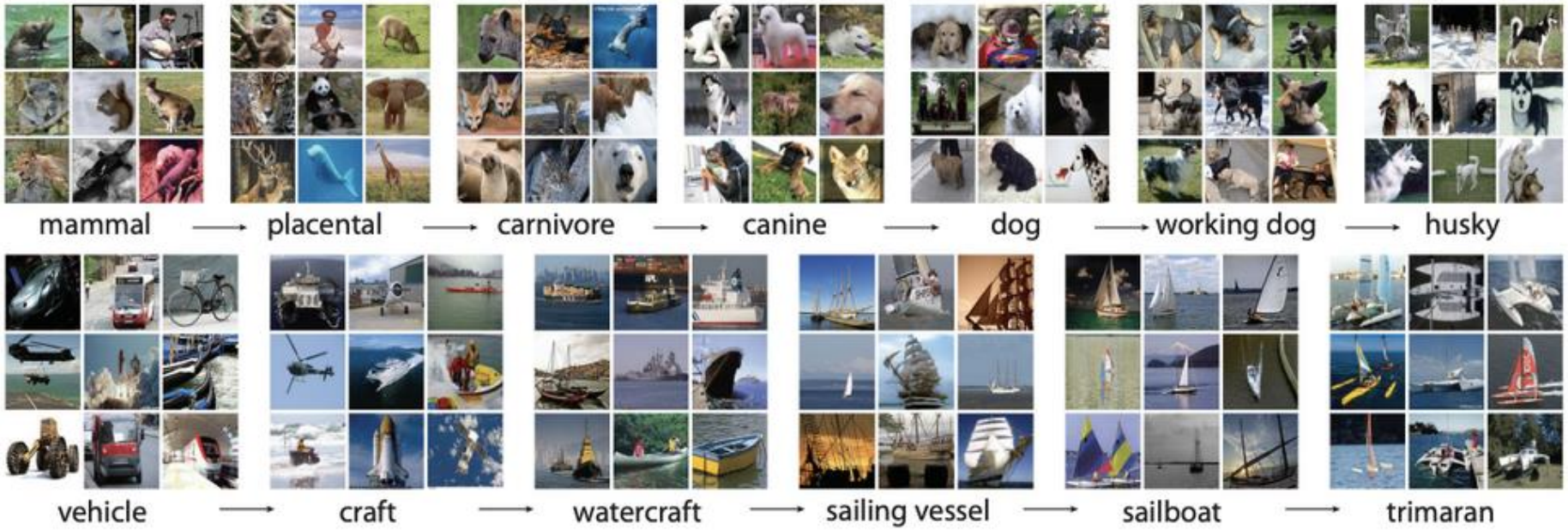
2024 г.

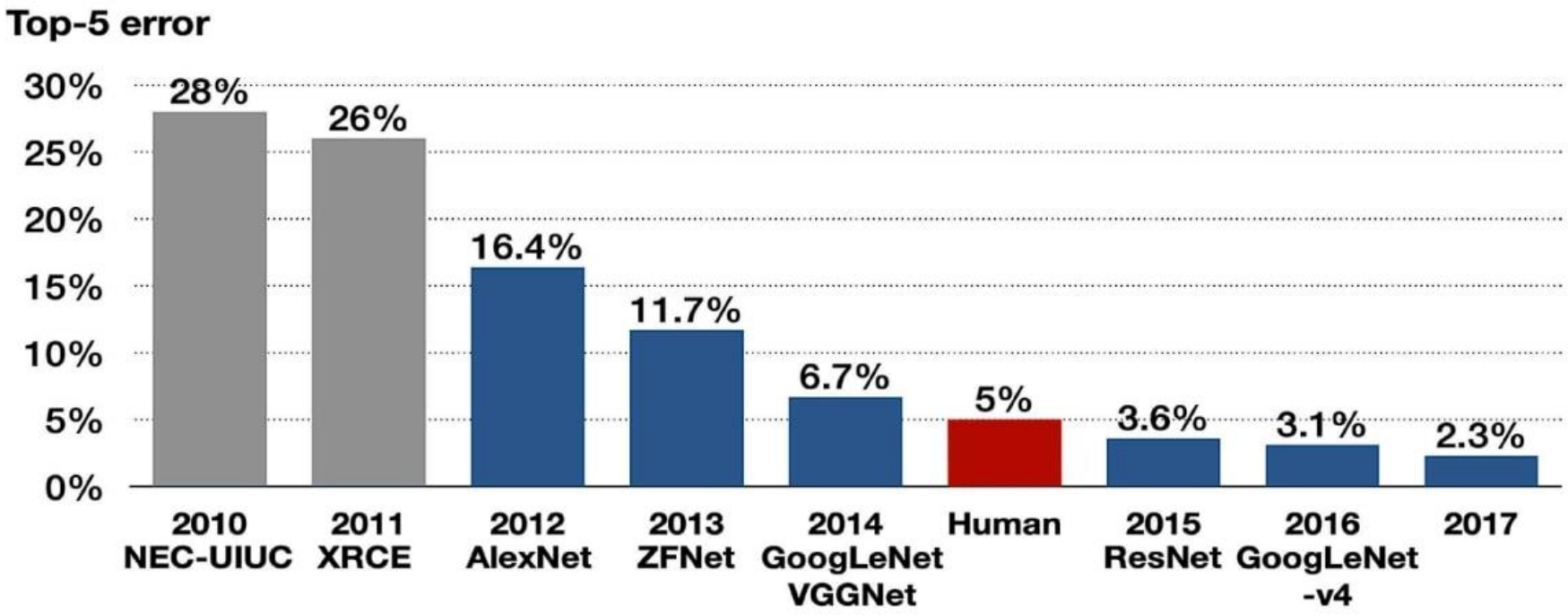
В 1989 году 32-летний Ян ЛеКун представил революционную сверточную нейросеть, способную распознавать цифры в различных стилях написания. Это было логическим продолжением его идеи, предложенной годом ранее, в 1988-м.



The **ImageNet** project is a large visual database designed for use in visual object recognition software research

ImageNet: A Large-Scale Hierarchical Image Database

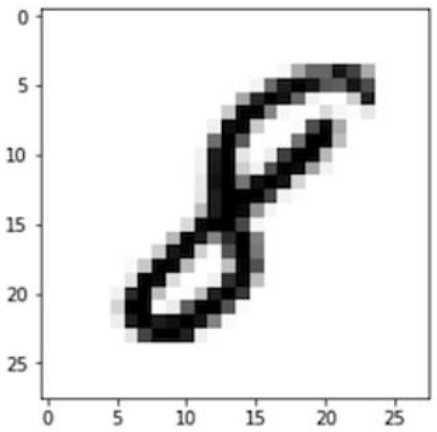



Top-5 error
30%
25%
20%
15%
10%
5%
0%
28%
26%
16.4%
11.7%
6.7%
5%
3.6%
3.1%
2.3%
2010
NEC-UIUC
2011
XRCE
2012
AlexNet
2013
ZFNet
2014
GoogLeNet
VGGNet
Human
2015
ResNet
2016
GoogLeNet
-v4
2017

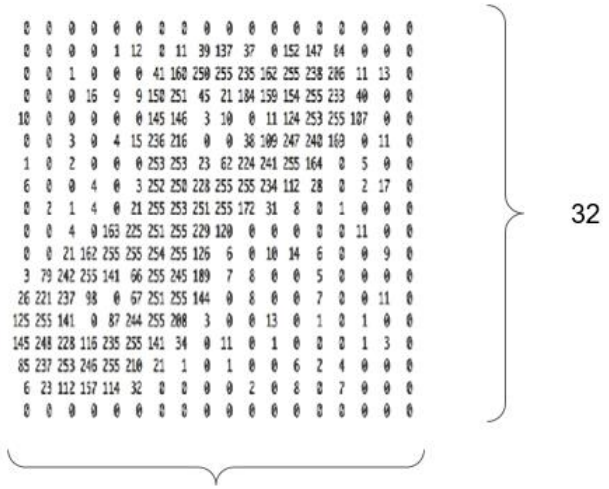
GT: sunscreen
1: hair spray
2: ice lolly
3: sunscreen
4: water bottle
5: lotion

GT: flute
1: flute
2: oboe
3: panpipe
4: trombone
5: bassoon

GT: wooden spoon
1: wok
2: frying pan
3: spatula
4: wooden spoon
5: hot pot



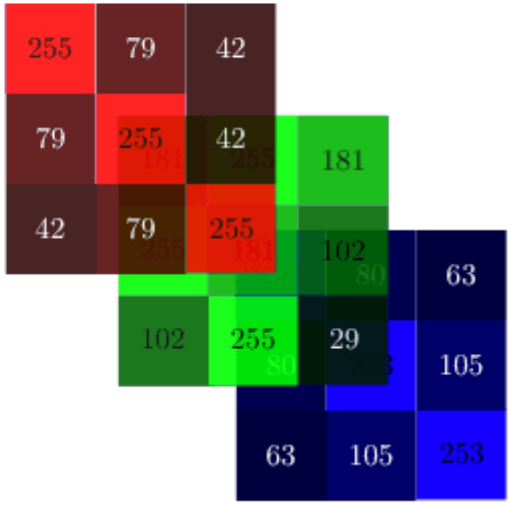
=



Grayscale

223	150	91
39	71	150
221	150	221

RGB

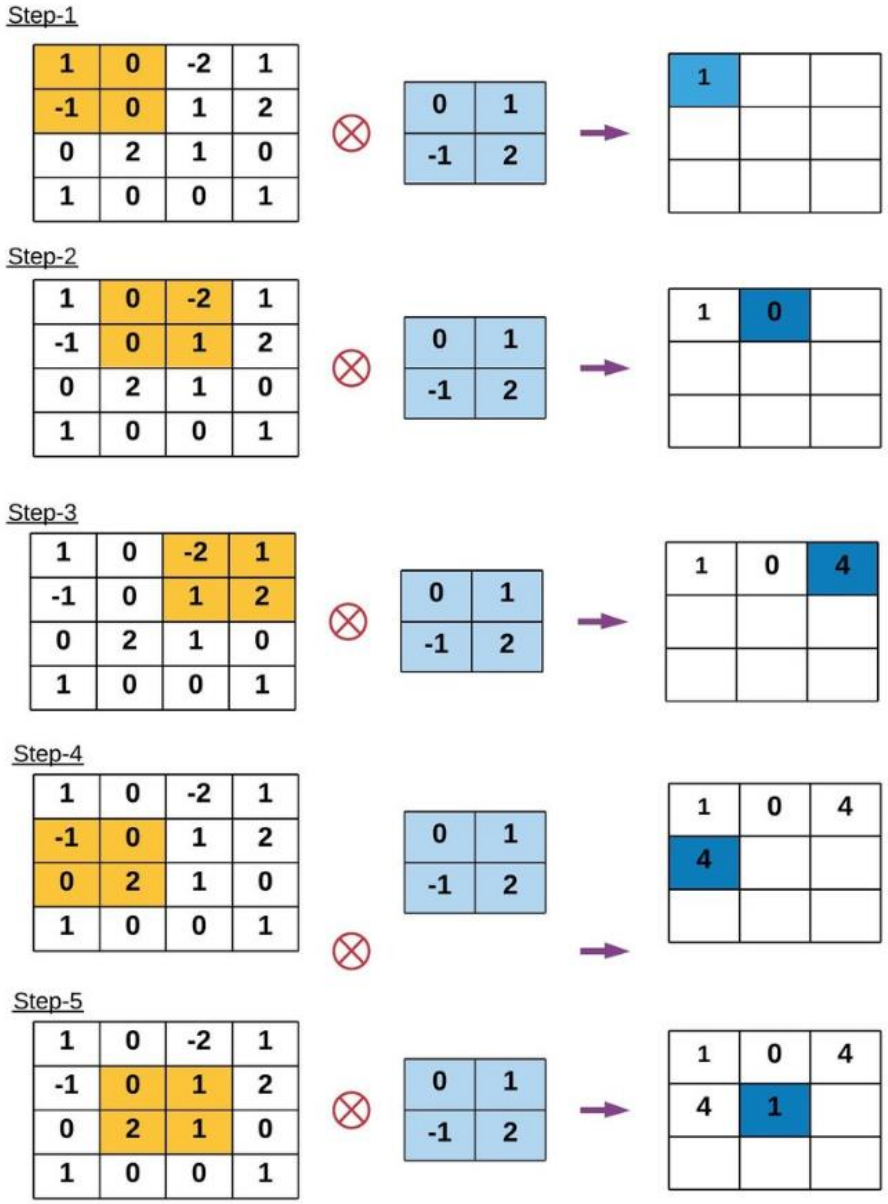


Классификация с помощью полносвязной НС с кросс-энтропийной функцией потерь:

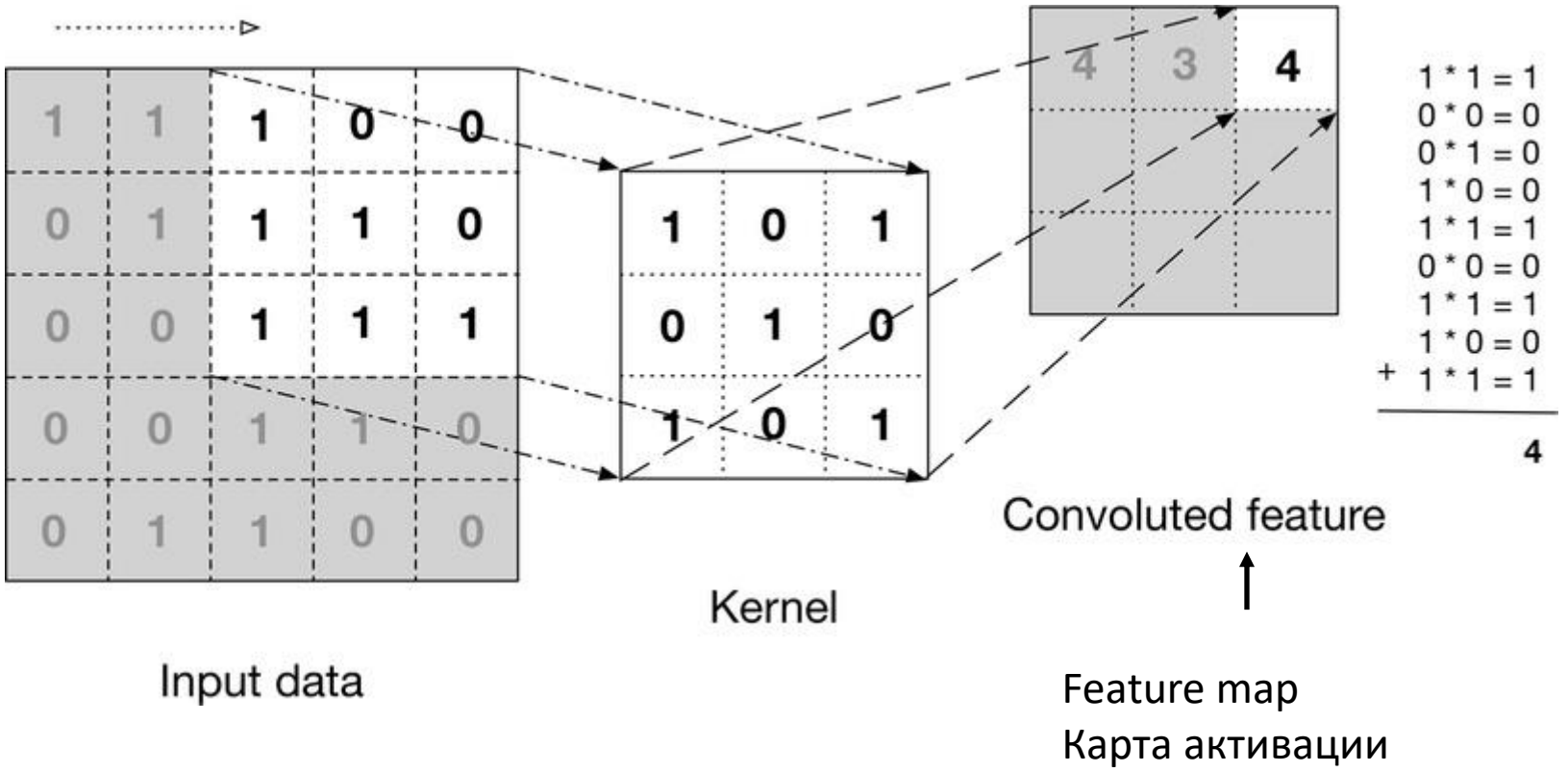
- Никак не учитывается структура данных – хочется получить инвариантность к различным преобразованиям исходного изображения

- Большое количество параметров (нейронов в первом слое сети). Например, RGB изображение размером 128x128 это вектор из 49152 элементов

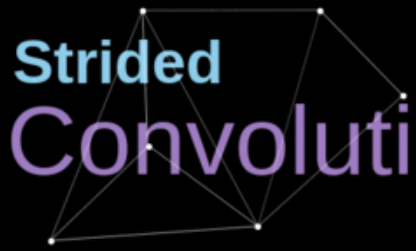
Операция свертки – поэлементное умножение и суммирование



С чем сворачиваем? С ядром свертки (фильтром)



Feature map
Карта активации



Strided Convolution

Input Image

1	9	8	4	4	5	7
4	8	6	7	9	1	7
4	0	5	9	3	8	4
7	3	5	9	0	5	4
7	4	1	1	8	1	2
7	6	6	9	8	7	6
3	6	3	5	4	2	7

*

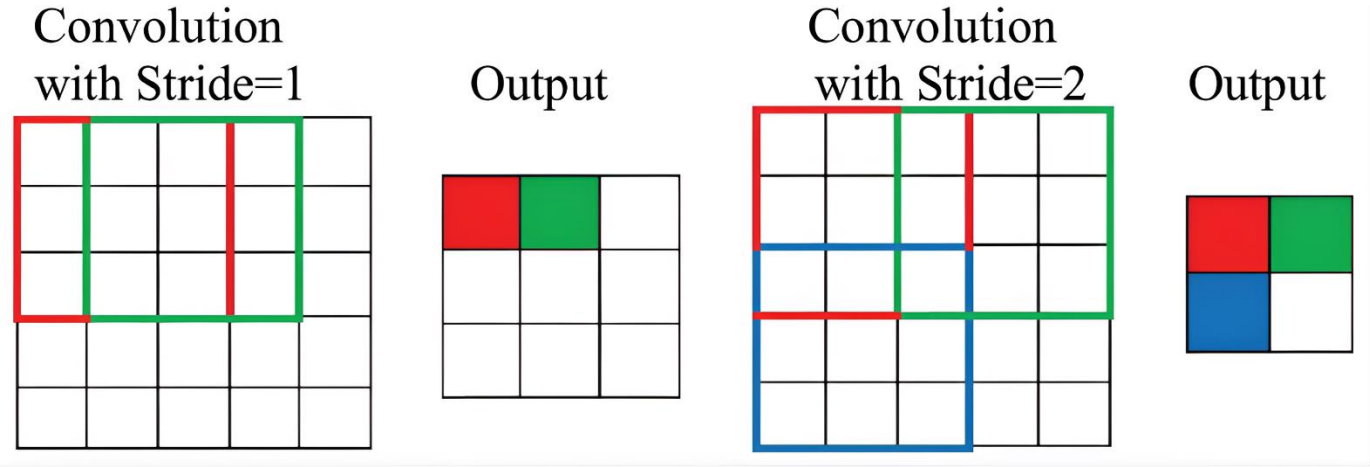
Kernel

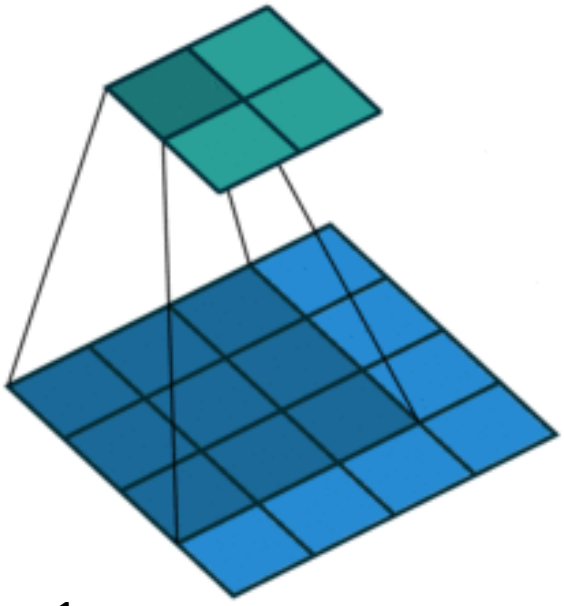
0	-1	0
-1	5	-1
0	-1	0

=

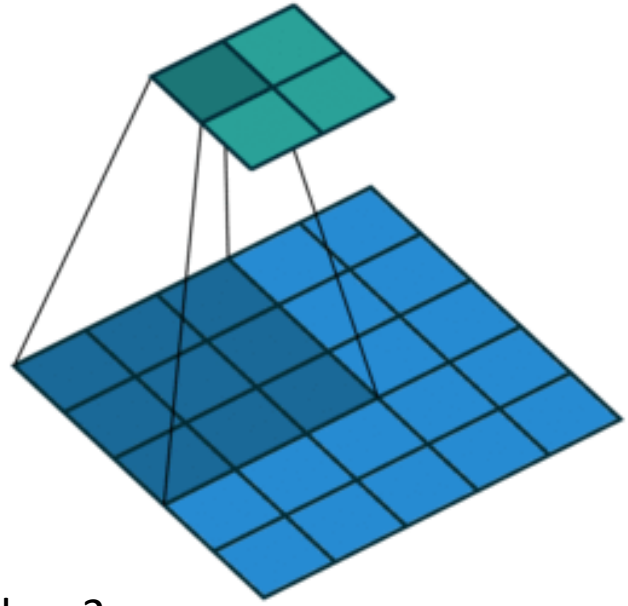
Feature Map

- Stride – величина смещения фильтра





Stride = 1



Stride = 2

- Padding – дополнение исходного изображения пикселями (рамка вокруг изображения)

0	0	0	0	0	0	0	0
0	1	2	3	4	5	6	0
0	7	8	9	10	11	12	0
0	1	2	3	4	5	6	0
0	7	8	9	10	11	12	0
0	1	2	3	4	5	6	0
0	7	8	9	10	11	12	0
0	0	0	0	0	0	0	0

Zero padding with a one-pixel thick boundary

11	10	9	10	11	12	11	10
7	6	5	6	7	8	7	6
3	2	1	2	3	4	3	2
7	6	5	6	7	8	7	6
11	10	9	10	11	12	11	10
15	14	13	14	15	16	15	14
11	10	9	10	11	12	11	10
7	6	5	6	7	8	7	6

Mirror padding with a two-pixel thick boundary

1	1	1	2	3	4	4	4
1	1	1	2	3	4	4	4
1	1	1	2	3	4	4	4
5	5	5	6	7	8	8	8
1	1	1	2	3	4	4	4
5	5	5	6	7	8	8	8
5	5	5	8	9	8	8	8
5	5	5	8	9	8	8	8

Replicate padding with a two-pixel thick boundary

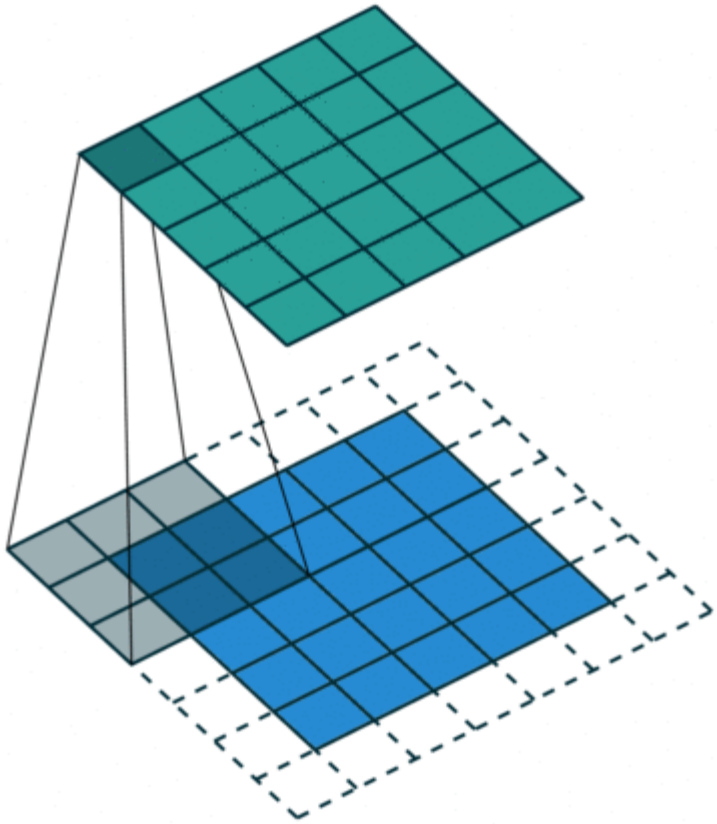
zero padding

mirror padding

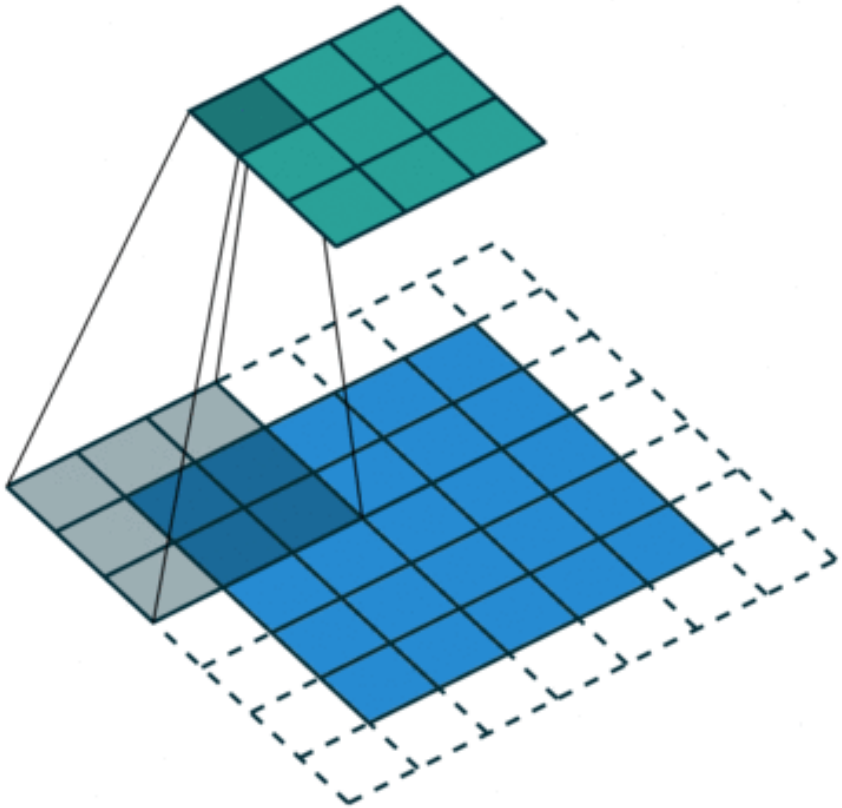
circular padding

Источник: https://habr.com/ru/articles/795223/

Padding = 1
Stride = 1

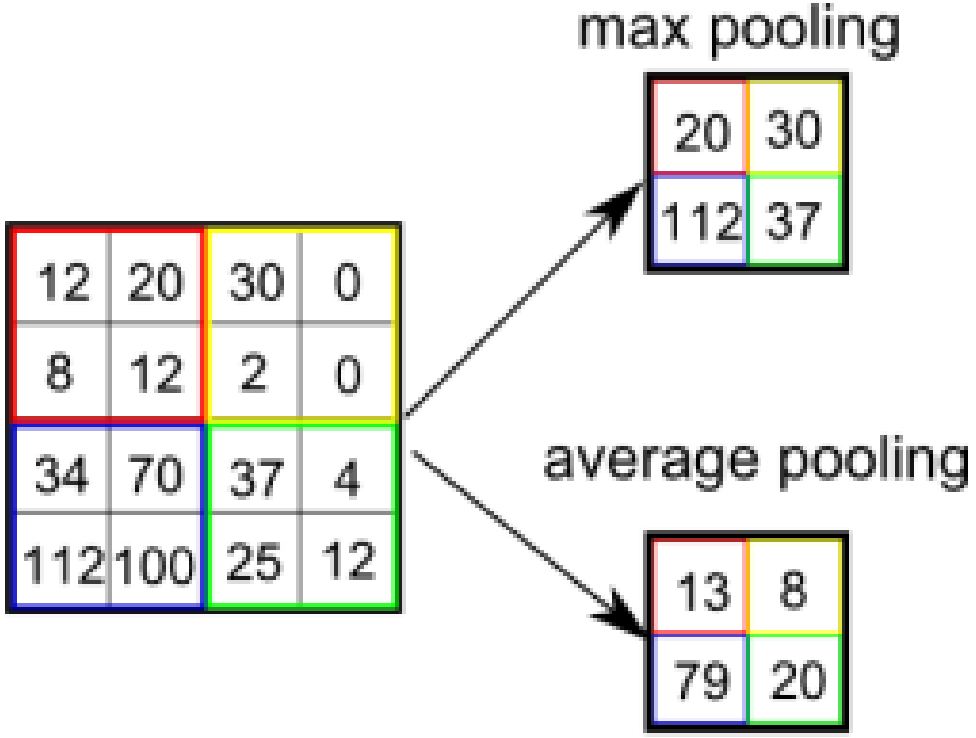


Padding = 1
Stride = 2

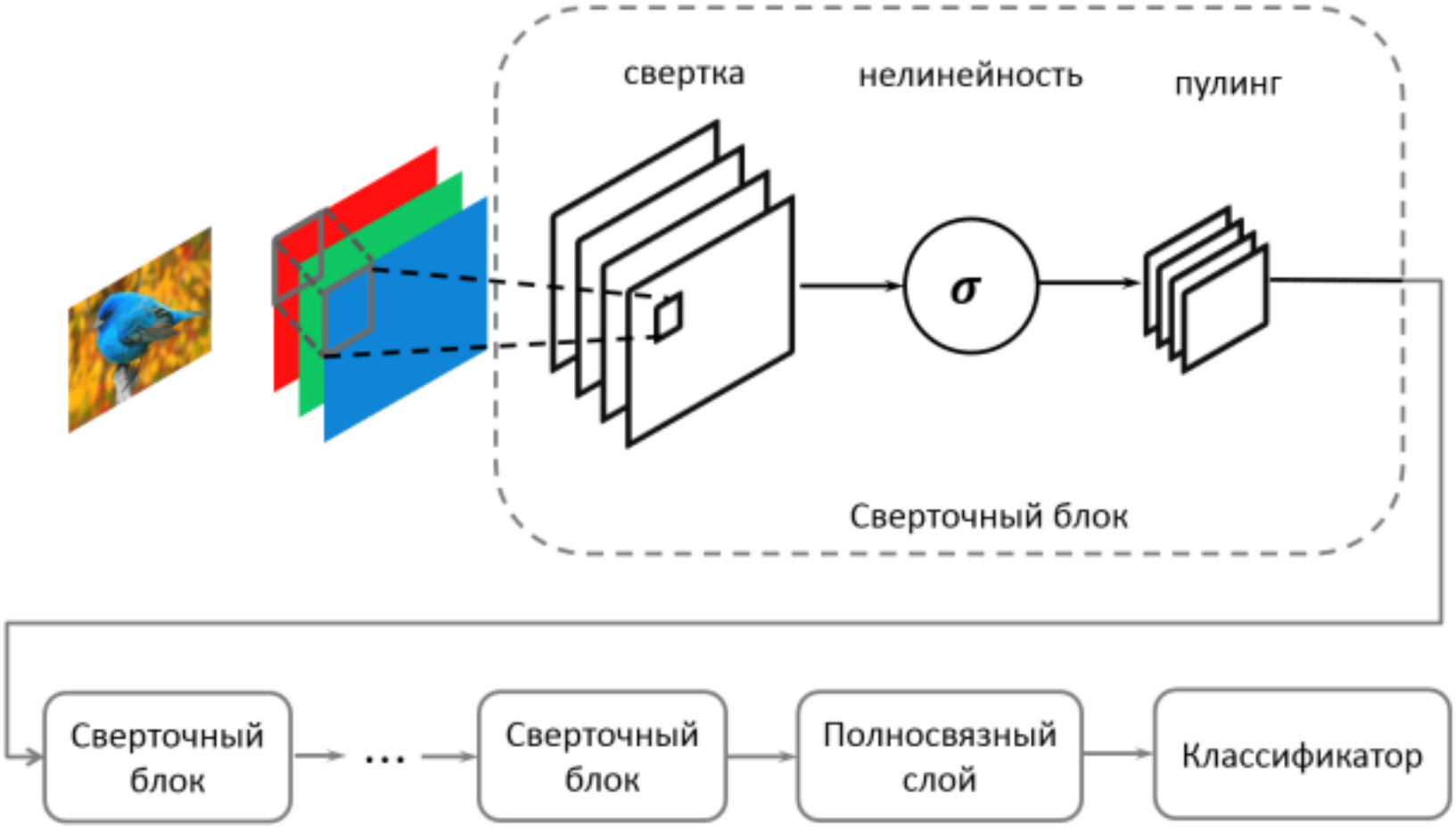


Источник: https://hannibunny.github.io/mlbook/neuralnetworks/convolutionDemos.html

- Pooling – техника уменьшения размерности (downsampling) карт активации. Если некая область содержит ярко выраженные свойства, то мы можем отказаться от поиска других свойств в этой области



Базовая схема сверточной сети



Источник: www.machinelearning.ru/wiki/images/7/71/Voron-ML-DeepLearning-slides.pdf

Фильтры “реагируют” на паттерны на изображении. Если паттерн присутствует на изображении, то карта активации после соотв. фильтра будет содержать большие числа

x

Фильтр, который
реагирует
на вертикальные линии

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

Активация сильнее, на
карте активации большие
числа

34	55	64	73	23
13	15	23	-86	-96
12	-3	0.4	71	19
11	14	17	-35	19

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

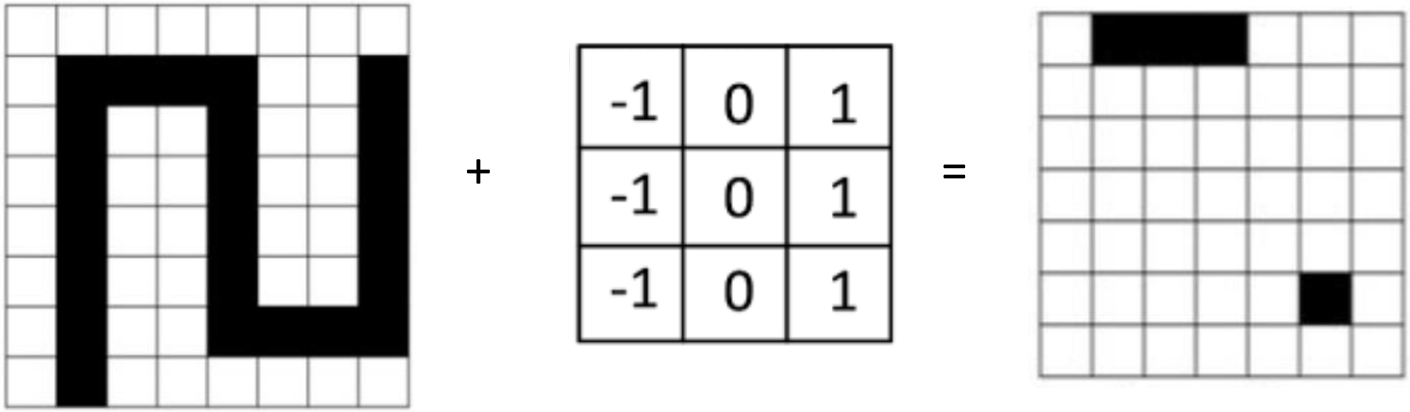
0.01	-0.2	1.8	2	-1.5
3	-0.5	6	-7	0.4
4	5	-0.8	-5	-1
1.2	0.5	3	-3	0.4

Фильтр, который
реагирует
на изогнутые линии

Активация слабее, на карте
активации маленькие числа

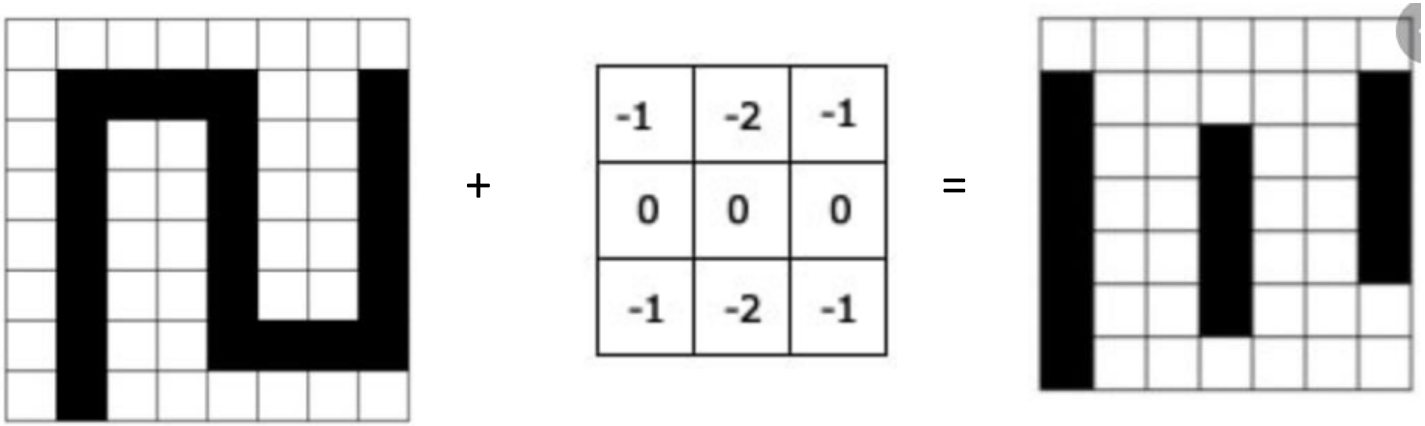
Источник: https://www.youtube.com/watch?v=HpKGv-kYurk

Как мог бы выглядеть фильтр, реагирующий на горизонтальные линии



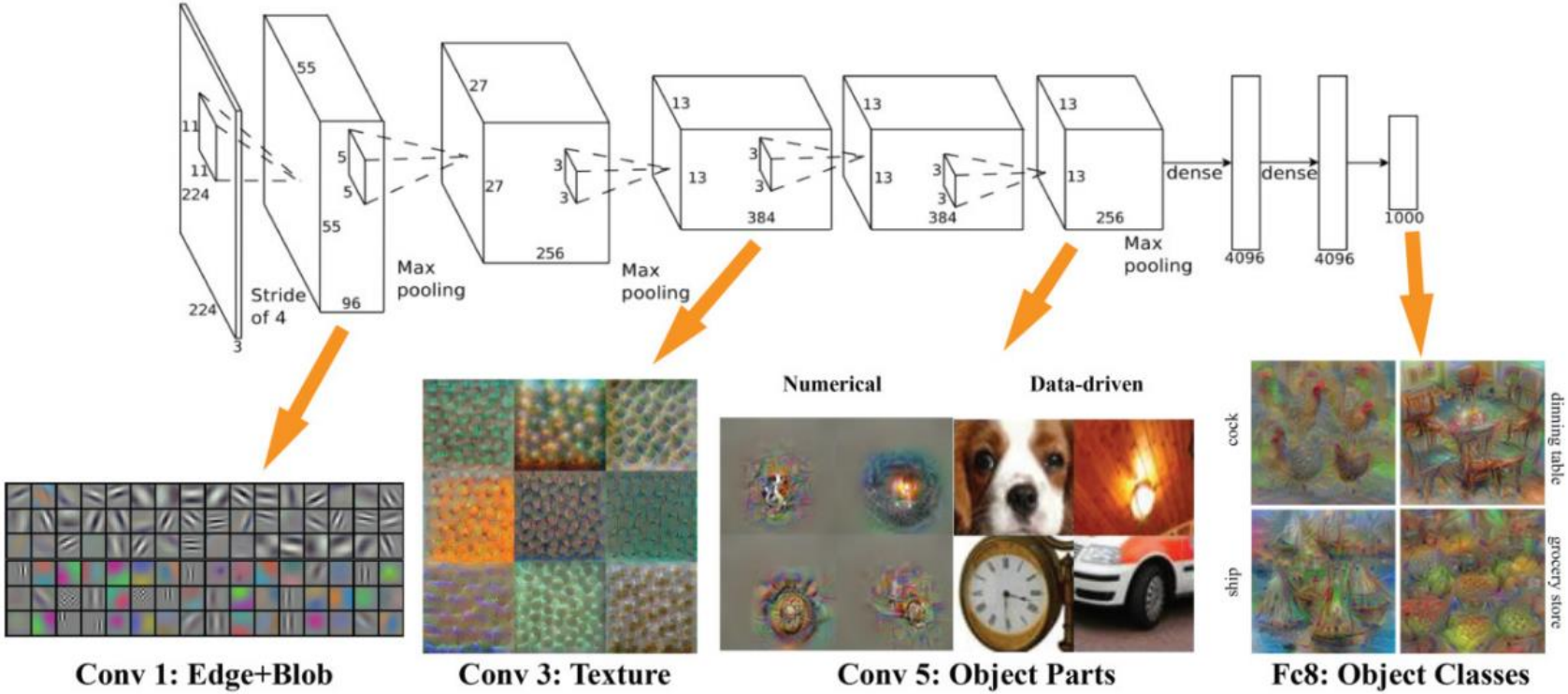
Источник: https://www.youtube.com/watch?v=HpKGv-kYurk

Как мог бы выглядеть фильтр, реагирующий на вертикальные линии



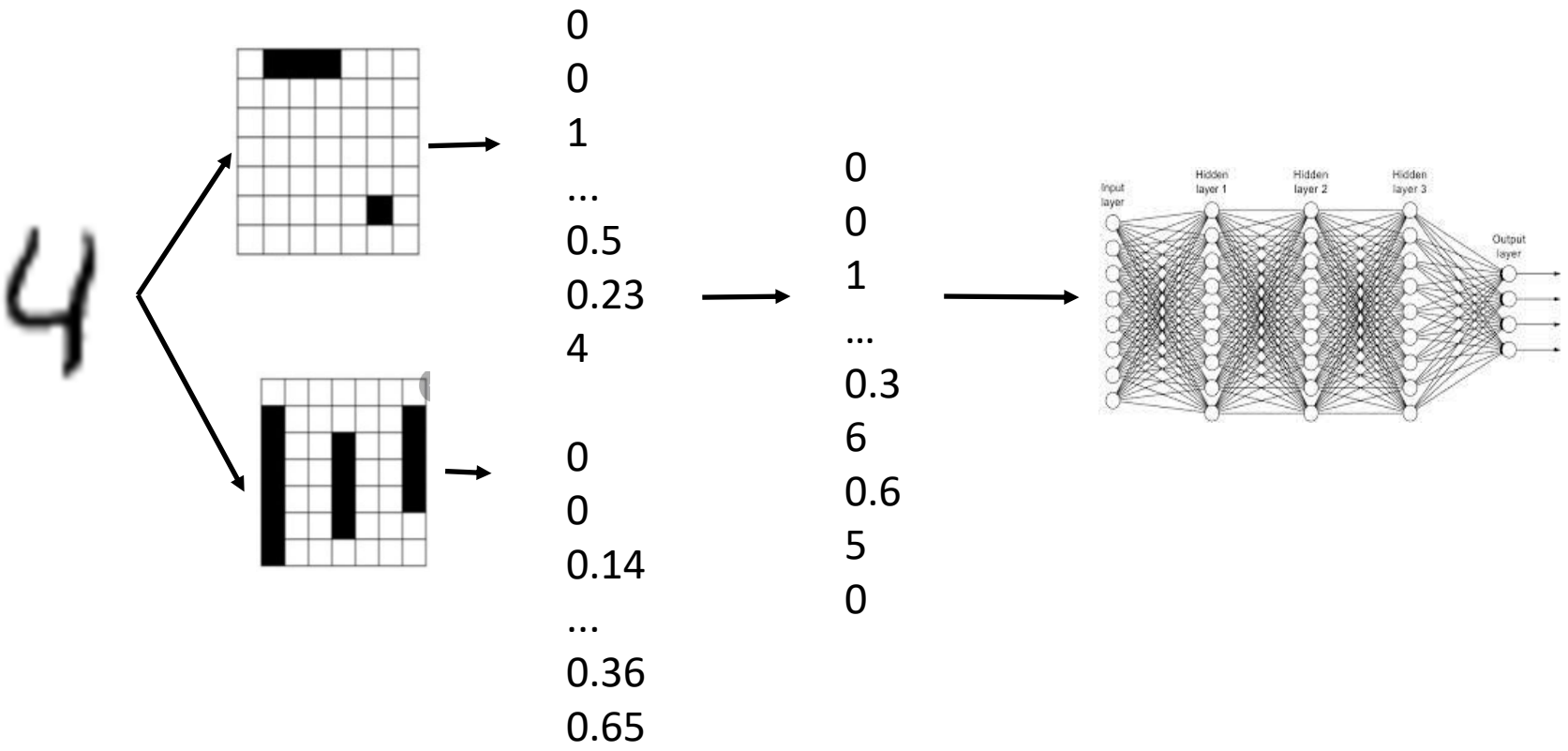
Источник: https://www.youtube.com/watch?v=HpKGv-kYurk

- Сверточная сеть обучается извлечению признаков. Чем выше слой, тем более крупные и сложные элементы изображений он способен распознавать



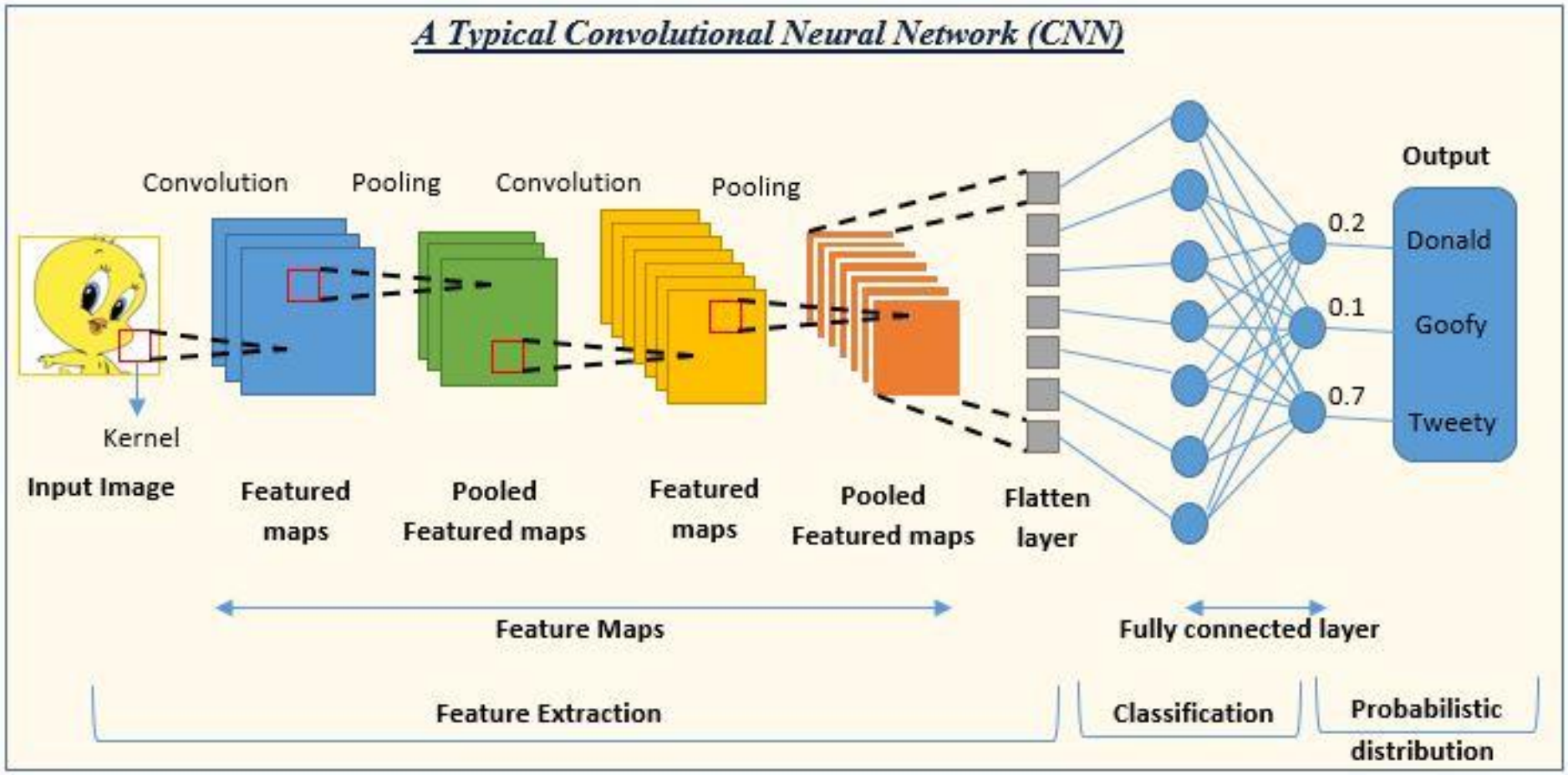
Krizhevsky A., Sutskever I., Hinton G. ImageNet classification with deep convolutional neural networks. 2012.

После получения карт активаций, мы **развернем все карты в векторы, сконкатенируем** и подадим на вход полносвязной сети

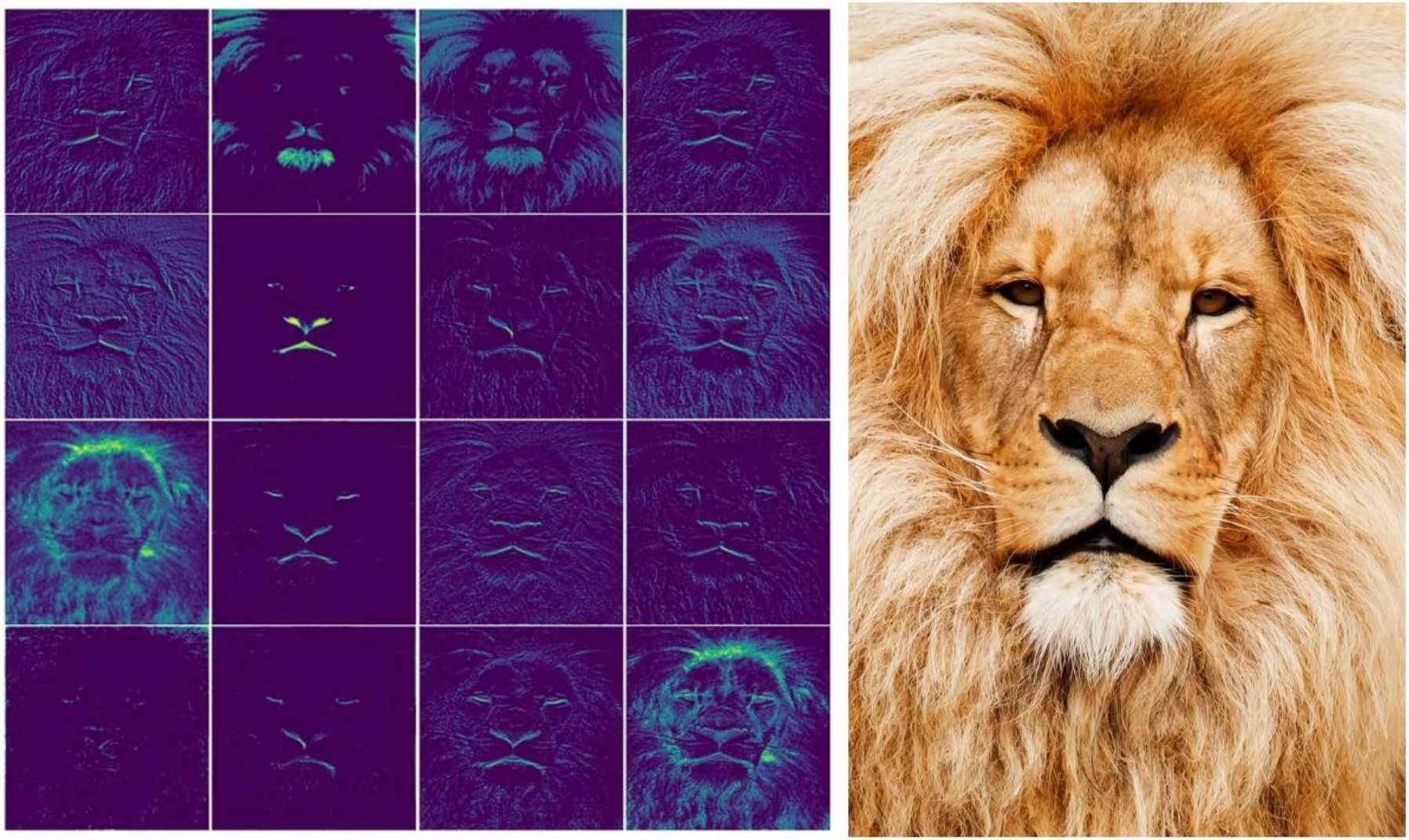


Источник: https://www.youtube.com/watch?v=HpKGv-kYurk

Вместо ручного конструирования признаков изображения с помощью операции свертки НС сама обучается извлечению признаков. Но для комплексных изображений необходимо много больше промежуточных представлений



Источник: https://ubiai.tools/the-secret-to-mastering-feature-extraction-in-convolutional-neural-network/



(**Left**) Feature extraction performed over the image of a lion using vgg19 CNN architecture (image by author). (**Right**) Original picture of the lion (public domain, availabe at Pexels).

Источник: https://towardsdatascience.com/exploring-feature-extraction-with-cnns-345125cefc9a

Свертка:

- Выявляет наличие на изображении паттерна, который задается ядром свертки
- Результат операции свертки – промежуточное представление (новое изображение)
- Чем сильнее на исходном изображении представлен паттерн, тем выше будут значения свертки
- Хотим много различных паттернов – используем много сверток

А что вообще обучать?

- Ядро свертки – подбирается в процессе обучения
- Веса полносвязного слоя – подбираются в процессе обучения

- Первые свертки выделяют какие-то низкоуровневые паттерны (изгибы, край, линии)

- Последние сверточные слои выделяют части изображений. Выход последнего слоя – признаковое описание изображения

- Полученные представления (признаковые описания) подаются на вход полносвязной сети. Например, для задачи многоклассовой классификации: размер выходного слоя – количество классов, функия активации - sotfmax

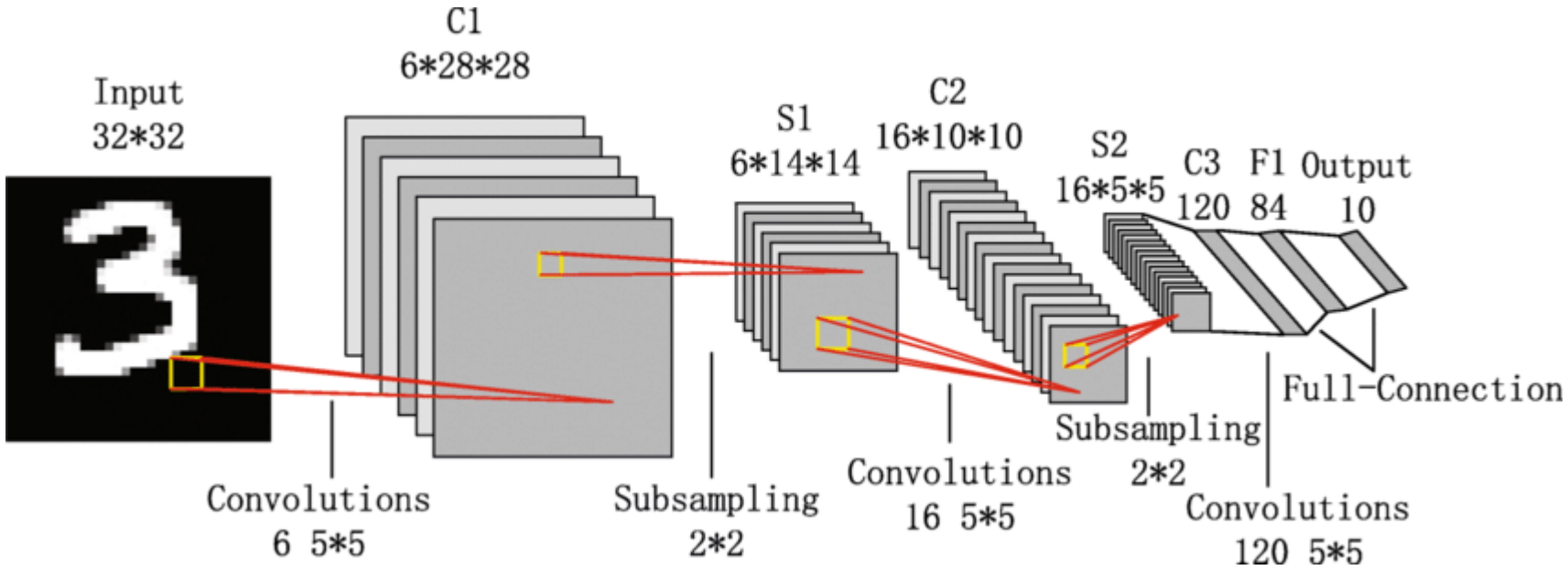
- Зная параметры сети (размер ядра свертки, паддинг, пуллинг и страйд) можно посчитать размер выходной матрицы признаков:

$$n_{out} = \left\lfloor \frac{n_{in} + 2p - k}{s} \right\rfloor + 1$$

n_{in}: number of input features
n_{out}: number of output features
k: convolution kernel size
p: convolution padding size
s: convolution stride size

Источник: https://syncedreview.com/2017/05/11/a-guide-to-receptive-field-arithmetic-for-convolutional-neural-networks/

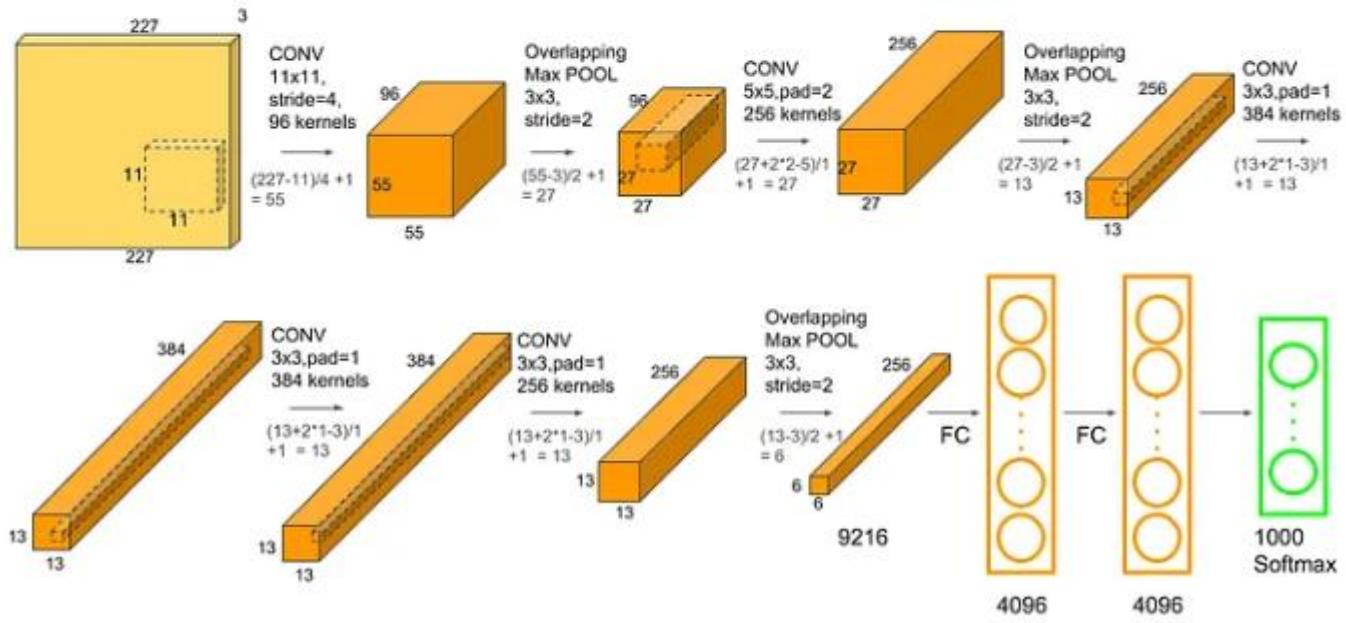
Насколько глубокие должны быть сети?



Архитектура нейронной сети LeNet-5, 1998 год. Первая версия LeNet-1 была представлена в 1989 году

Насколько глубокие должны быть сети?

- AlexNet — 2012. Uses max pooling instead of average pooling. Uses ReLU activation instead of tanh. Takes in 3 channels (RGB) as input. Obtained significantly better results in ImageNet Large Scale Visual Recognition Challenge compared to other models at the time.



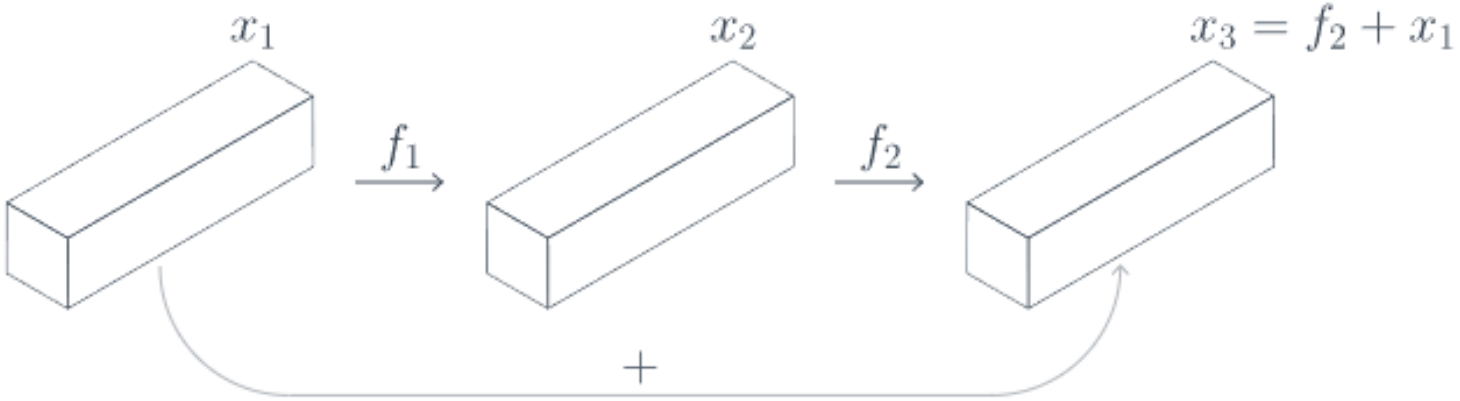
Layer	Type	Maps	Size	Kernel size	Stride	Padding	Activation
out	Fully connected	__	1000	__	__	__	Softmax
F9	Fully connected	__	4096	__	__	__	ReLU
F8	Fully connected	__	4096	__	__	__	ReLU
C7	Convolution	256	13*13	3*3	1	Same	ReLU
C6	Convolution	384	13*13	3*3	1	Same	ReLU
C5	Convolution	384	13*13	3*3	1	Same	ReLU
S4	Max Pooling	256	13*13	3*3	2	Valid	__
C3	Convolution	256	27*27	5*5	1	Same	ReLU
S2	Max Pooling	96	27*27	3*3	2	Valid	__
C1	Convolution	96	55*55	11*11	4	Valid	ReLU
In	Input	3(RGB)	227*227	__	__	__	__

AlexNet Summary

- Чем больше сверток, тем лучше?

Если мы будем бесконтрольно добавлять сверточные слои, то, несмотря на использование ReLU и batchNorm, градиенты все равно будут затухать и на первых слоях будут близки к нулю

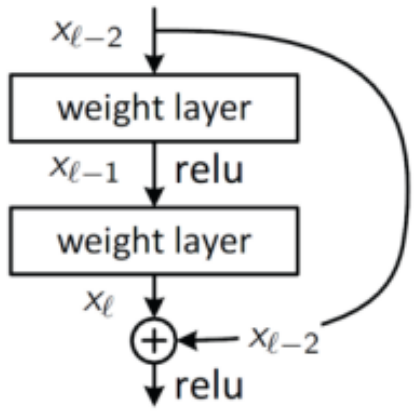
- Residual connection (архитектура ResNet, Residual NN) – прокидываем признаки на предыдущем слое мимо сверток на следующем



- DenseNet – вместо суммы конкатенация, результаты лучше, но вычислительно сложно

Сквозная связь (skip connection) слоя ℓ с предшествующим слоем $\ell - d$:

$$x_\ell = \sigma(Wx_{\ell-1}) + x_{\ell-d}$$



Слой ℓ выучивает не новое векторное представление x_ℓ, а его приращение $x_\ell - x_{\ell-d}$

- Приращения более устойчивы $\Rightarrow$ улучшается сходимость
- Появляется возможность увеличивать число слоёв
- Обобщение — Highway Networks:

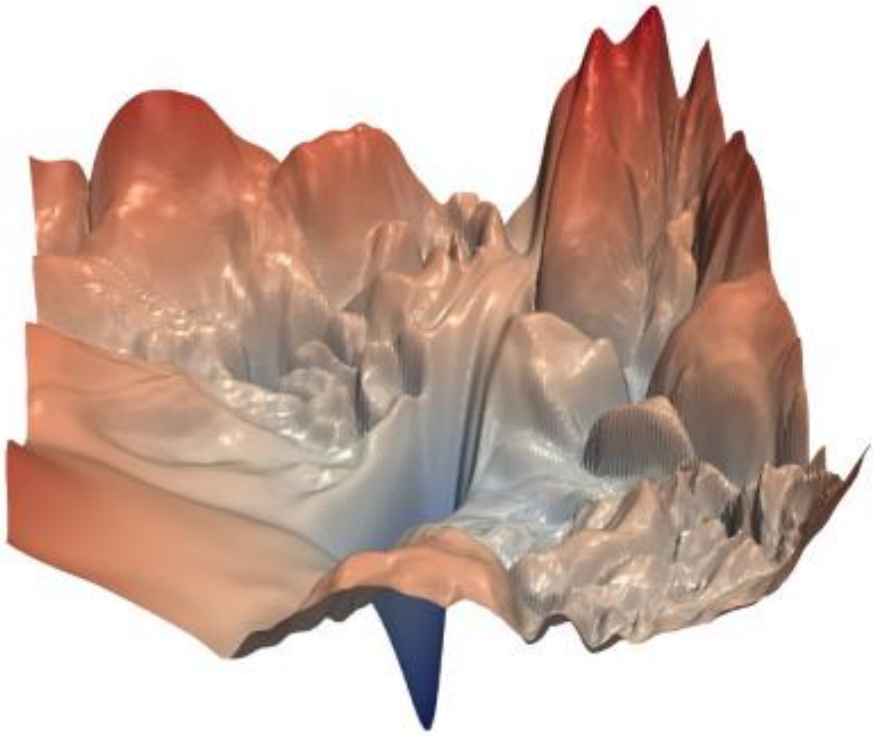
$$x_\ell = \sigma(Wx_{\ell-1})\underbrace{\tau(W'x_{\ell-1})}_{\text{transform gate}} + x_{\ell-d}\underbrace{\big(1-\tau(W'x_{\ell-1})\big)}_{\text{carry gate}}$$

Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. Deep Residual Learning for Image Recognition. 2015

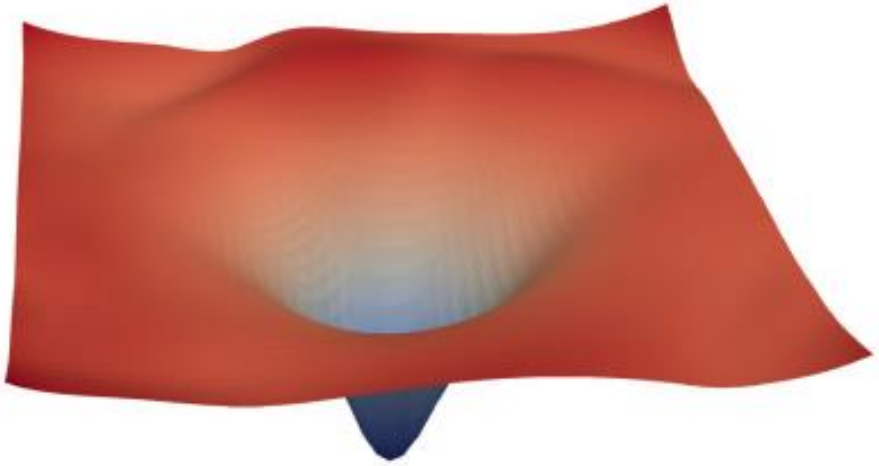
R.K.Srivastava, K.Greff, J.Schmidhuber. Highway Networks. 2015

Источник: http://www.machinelearning.ru/wiki/images/7/71/Voron-ML-DeepLearning-slides.pdf

Сквозные связи (skip connection) упрощают оптимизируемый критерий, устраняя локальные экстремумы и седловые точки:



without skip connections



with skip connections

Hao Li et al. Visualizing the Loss Landscape of Neural Nets. 2018

Источник: http://www.machinelearning.ru/wiki/images/7/71/Voron-ML-DeepLearning-slides.pdf

Знаковые архитектуры в мире сверточных сетей для задач классификации изображений:

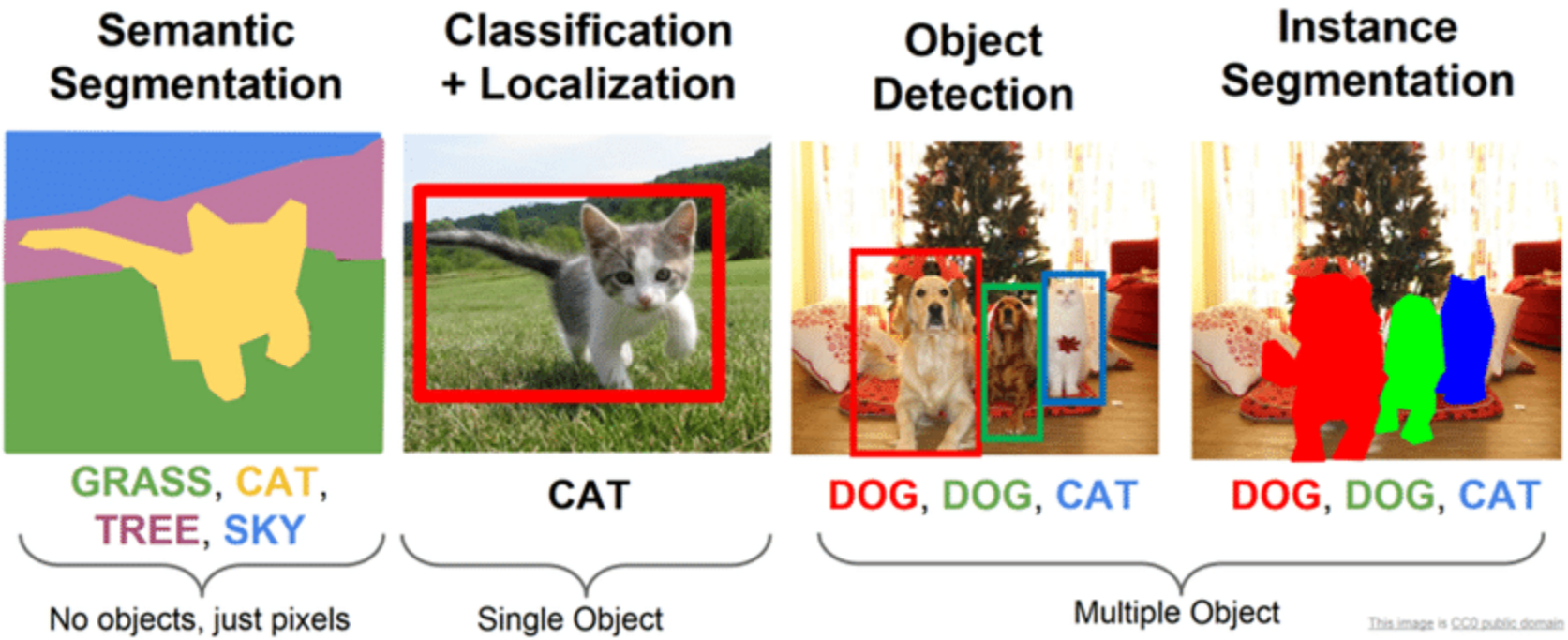
1. LeNet – 1998
2. AlexNet – 2012
3. Network in network - 2013
4. VGG - 2014
5. GoogLeNet (inception) – 2014
6. ResNet – 2015
7. MobileNet – 2017
8. EfficientNet - 2019

Источник: https://education.yandex.ru/handbook/ml/article/svyortochnye-nejroseti

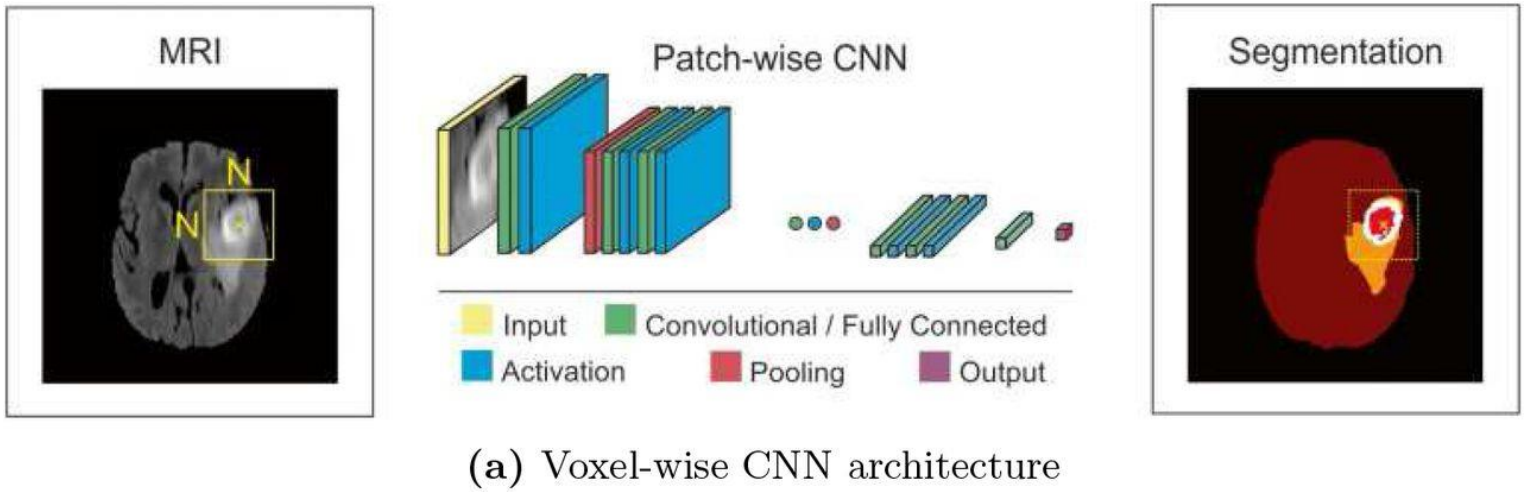
Что учитываем при обучении CNN:

- Функции активации без горизонтальных асимптот (ReLU)
- Адаптивные градиентные методы
- Dropout
- Batch normalization
- Residual NN
- Подбираем число слоев и размеры слоев
- Используем аугментацию

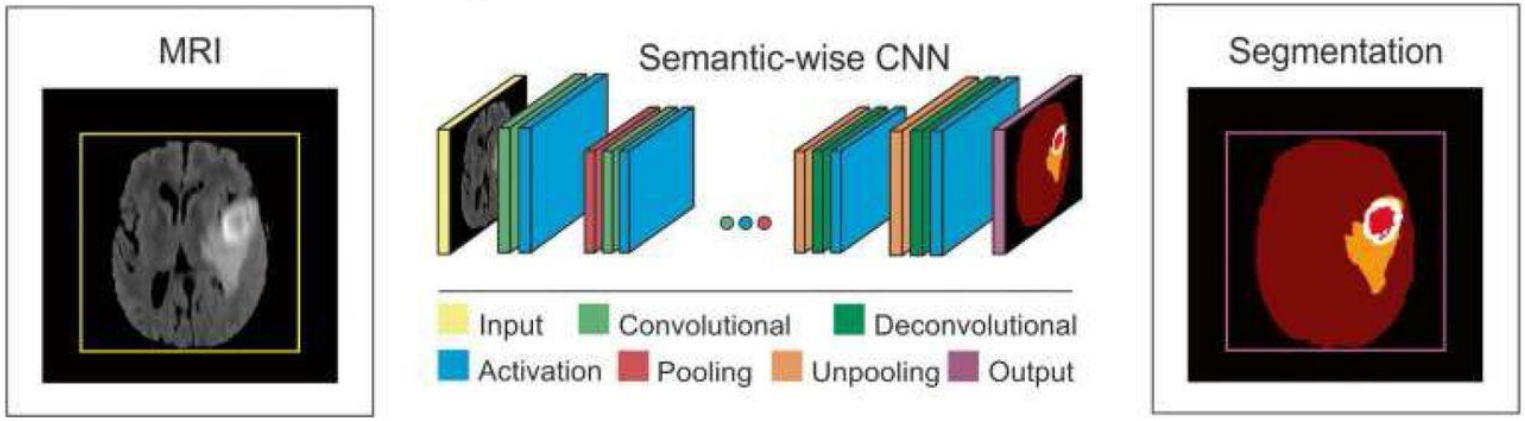
Практическое применение CNN



Применение в медицине



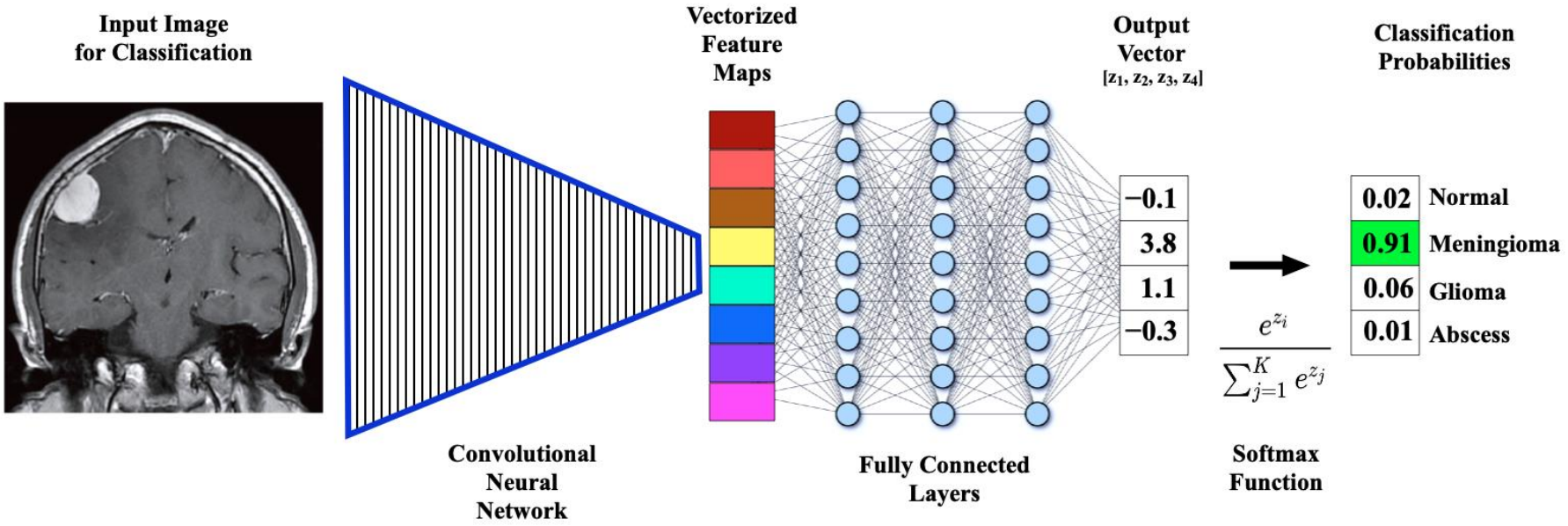
(a) Voxel-wise CNN architecture



(b) Semantic-wise CNN architecture

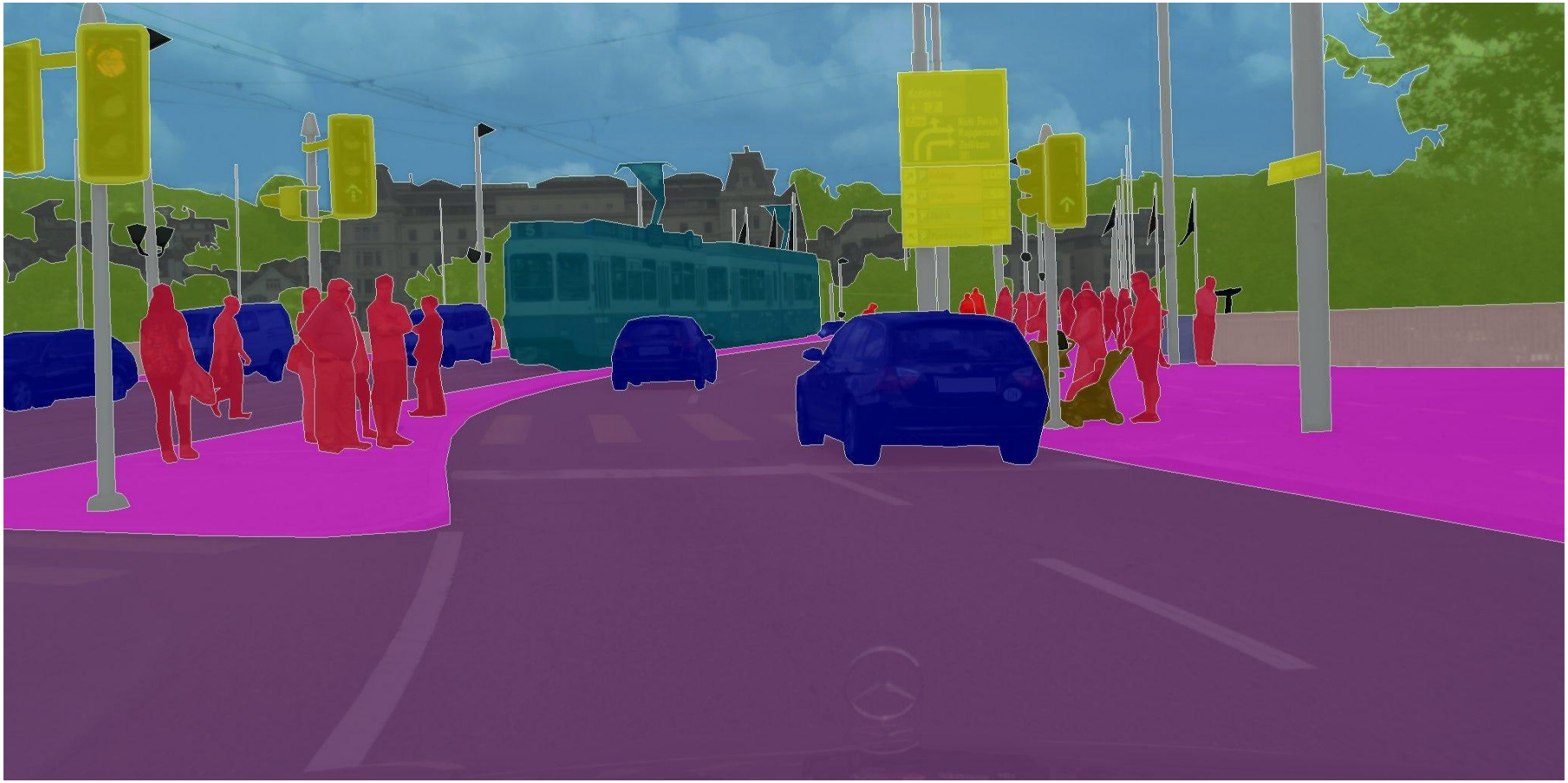
Источник: https://www.biorxiv.org/content/10.1101/2020.01.29.924985v1.full

Применение в медицине



Источник: https://mriquestions.com/convolutional-network.html#/

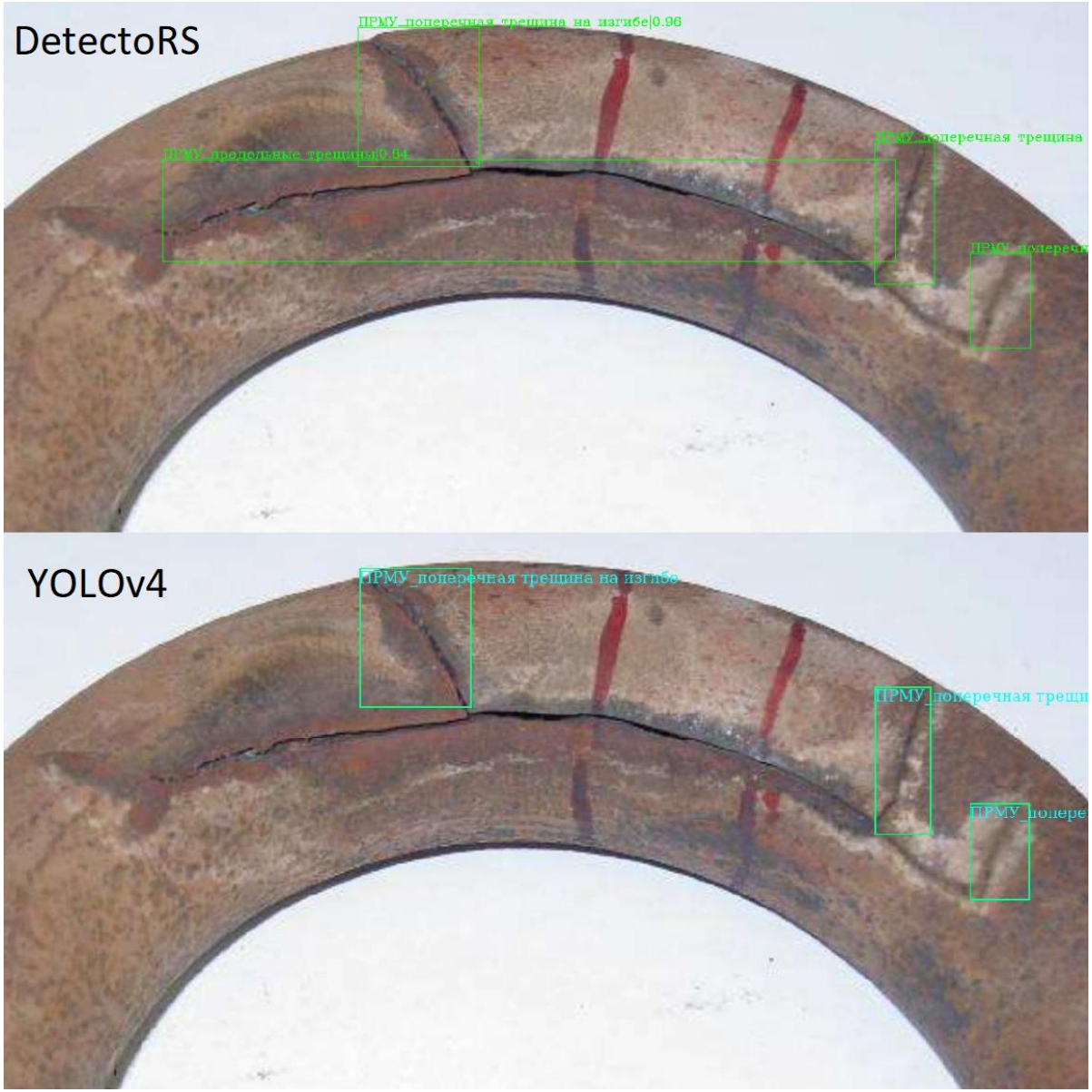
Применение в беспилотном транспорте



Источник: https://medium.com/analytics-vidhya/15-best-open-source-autonomous-driving-datasets-34324676c8d7

Применение в беспилотном транспорте

Применение в промышленности (поиск дефектов)



Источник: https://habr.com/ru/companies/itmai/articles/541858/

CNN explainer – наглядное средство визуализации CNN