

# Нейронные сети в машинном обучении

Лекция №5  
Глубинные нейронные сети

Евгений Богатырев



Не забывайте отмечаться  
и оставлять отзывы!

# Nesterov accelerated gradient

$$\nu_t = \gamma \nu_{t-1} + \eta_t \nabla_{\theta} J(\theta_{t-1} - \gamma \nu_{t-1})$$

$$\theta_t = \theta_{t-1} - \nu_t$$

Сделаем замену переменных так, чтобы не приходилось считать градиент в смещенной точке:

$$\theta_t^{prev} = \theta_t - \gamma \nu_t$$

$$\nu_t = \gamma \nu_{t-1} + \eta_t \nabla J(\theta_{t-1}^{prev})$$

## Nesterov accelerated gradient

$$\theta_t = \theta_{t-1} - \nu_t$$

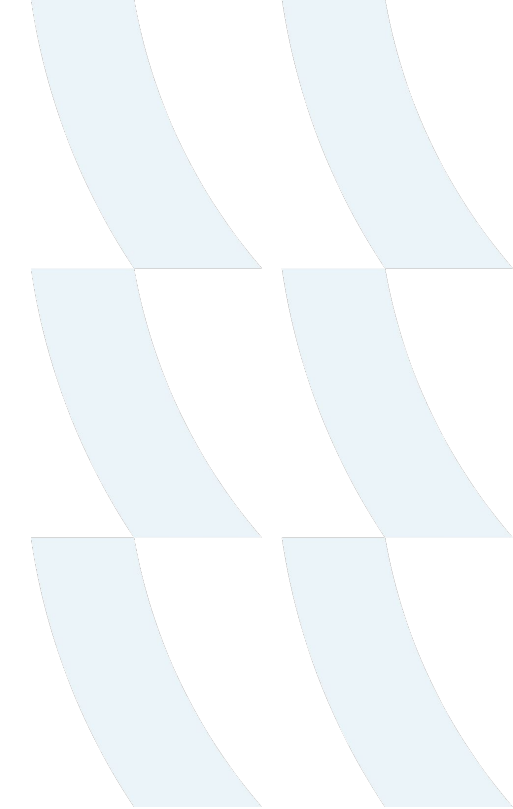
$$\theta_t^{prev} + \gamma \nu_t = \theta_{t-1}^{prev} + \gamma \nu_{t-1} - \nu_t$$

$$\begin{aligned}\theta_t^{prev} - \theta_{t-1}^{prev} &= -\gamma \nu_t + \gamma \nu_{t-1} - \nu_t = \\ &= -\nu_t(1 + \gamma) + \gamma \nu_{t-1}\end{aligned}$$

$$\theta_t^{prev} - = \nu_t(1 + \gamma) - \gamma \nu_{t-1}$$

# Содержание

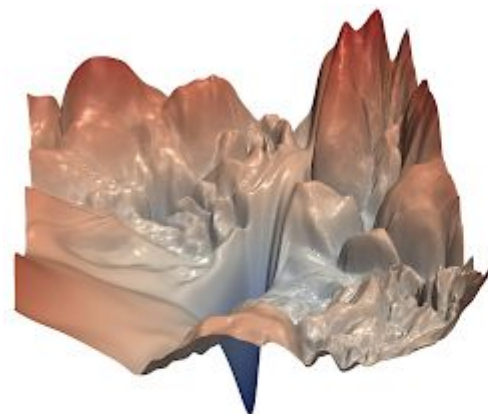
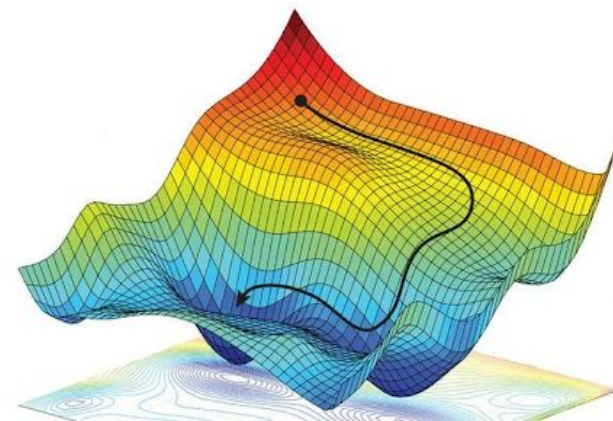
1. Оптимизации обучения
  - a. Инициализация
  - b. Регуляризация
  - c. Нормализация
2. Архитектуры CNN
  - a. LeNet
  - b. AlexNet
  - c. VGG
  - d. ResNet
  - e. Модификации ResNet
3. Сравнение моделей



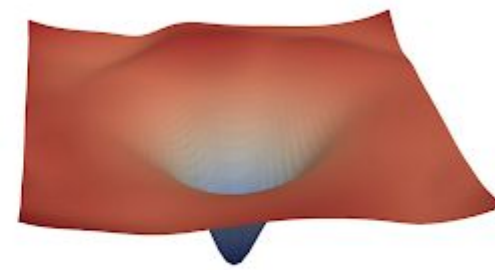
# Резюме с прошлого занятия

На прошлой лекции мы изучили методы оптимизации обучения нейронных сетей:

На этой лекции мы обсудим, какие еще существуют подходы, которые могут ускорить обучение и предотвратить переобучение



(a) without skip connections



(b) with skip connections

Figure 1: The loss surfaces of ResNet-56 with/without skip connections. The proposed filter normalization scheme is used to enable comparisons of sharpness/flatness between the two figures.

# Оптимизации обучения

# Инициализация Xavier / Glorot

Рассмотрим нечетную функцию с единичной производной в нуле в качестве активации (напр. tanh)

- Хотим начать из линейного региона, чтобы избежать затухающих градиентов

$$z^{i+1} = f(\underbrace{z^i W^i}_{s^i})$$

$$\mathbb{D}[z^i] = \mathbb{D}[x] \prod_{k=0}^{i-1} n_k \mathbb{D}[W^k]$$

$$\mathbb{D}\left[\frac{\partial L}{\partial s^i}\right] = \mathbb{D}\left[\frac{\partial L}{\partial s^d}\right] \prod_{k=i}^d n_{k+1} \mathbb{D}[W^k]$$

Где  $n_i$  — размерность  $i$ -го слоя



# Инициализация Xavier / Glorot

Хорошая инициализация:

$$\forall(i, j) \begin{cases} \mathbb{D}[z^i] = \mathbb{D}[z^j] \\ \mathbb{D}[\frac{\partial L}{\partial s^i}] = \mathbb{D}[\frac{\partial L}{\partial s^j}] \end{cases}$$

Это эквивалентно следующему:

$$\forall i \begin{cases} n_i \mathbb{D}[W^i] = 1 \\ n_{i+1} \mathbb{D}[W^i] = 1 \end{cases}$$

Компромисс:

$$\mathbb{D}[W^i] = \frac{2}{n_i + n_{i+1}}$$

$$W^i \sim U\left[-\frac{\sqrt{6}}{\sqrt{n_i + n_{i+1}}}, \frac{\sqrt{6}}{\sqrt{n_i + n_{i+1}}}\right]$$

# Инициализация He

Рассмотрим ReLU в качестве активации:

- Функция не симметрична
- Не дифференцируема в нуле

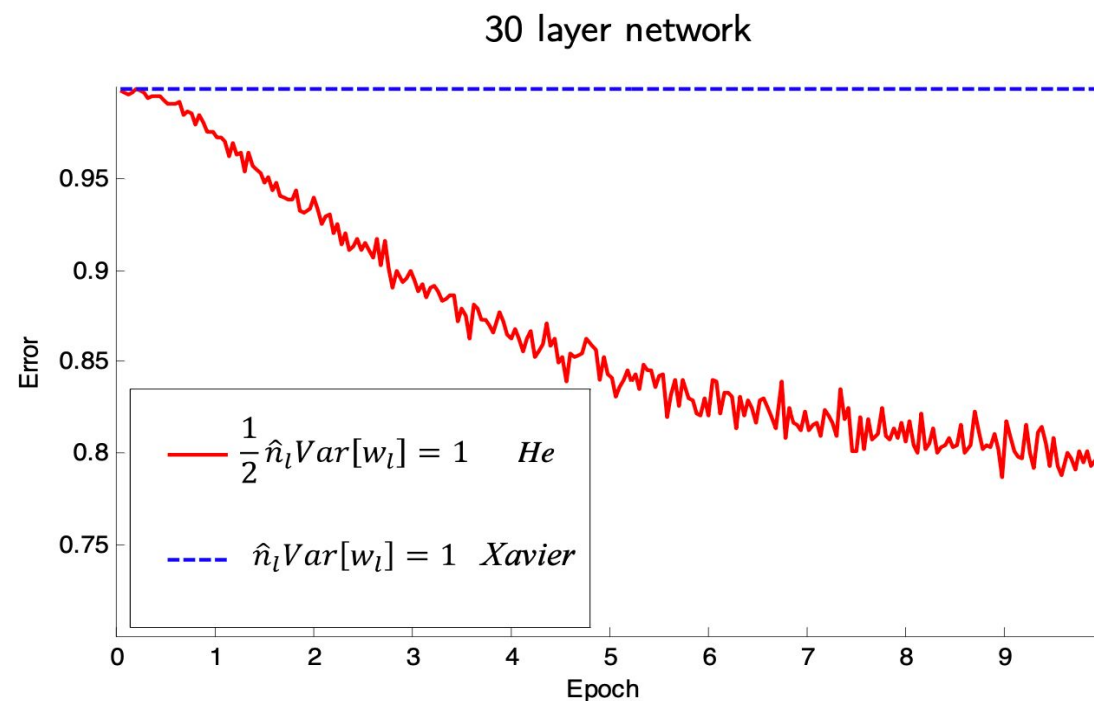
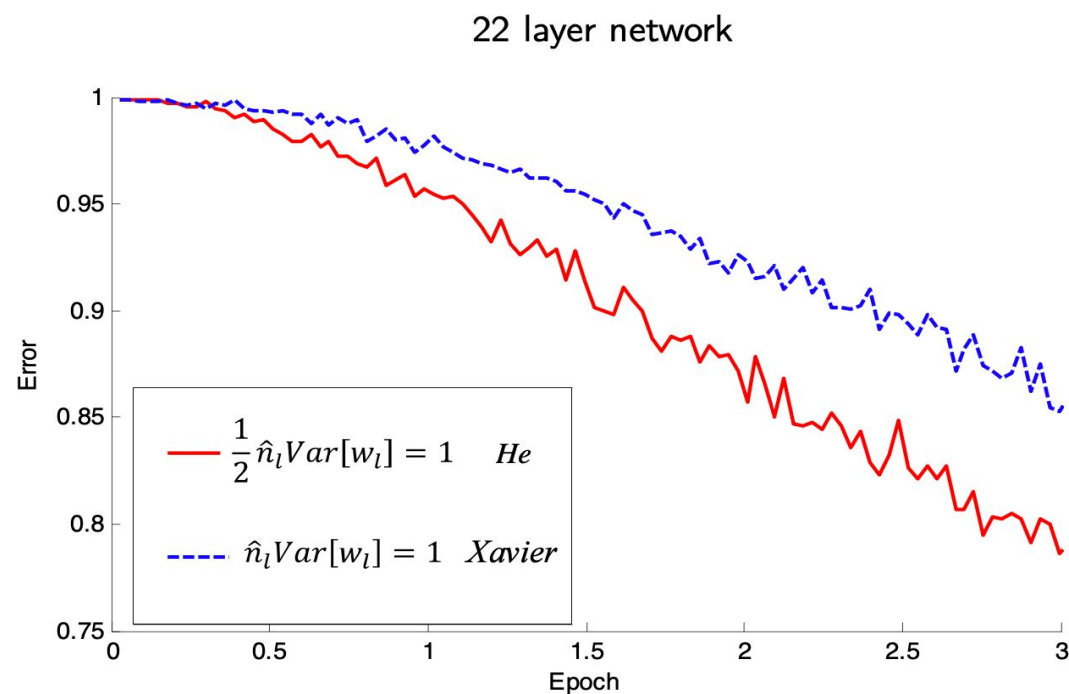
$$\mathbb{D}[z^i] = \mathbb{D}[x] \left( \prod_{k=0}^{i-1} \frac{1}{2} n_k \mathbb{D}[W^k] \right) \Rightarrow \mathbb{D}[W^k] = \frac{2}{n_k}$$

$$\mathbb{D}\left[\frac{\partial L}{\partial s^i}\right] = \mathbb{D}\left[\frac{\partial L}{\partial s^d}\right] \left( \prod_{k=i}^d \frac{1}{2} n_{k+1} \mathbb{D}[W^k] \right) \Rightarrow \mathbb{D}[W^k] = \frac{2}{n_{k+1}}$$

$$\mathbb{D}\left[\frac{\partial L}{\partial s^i}\right] = \mathbb{D}\left[\frac{\partial L}{\partial s^d}\right] \prod_{k=1}^d \frac{1}{2} n_{k+1} \mathbb{D}[W^k] = \frac{n_2}{n_d} \mathbb{D}\left[\frac{\partial L}{\partial s^d}\right]$$

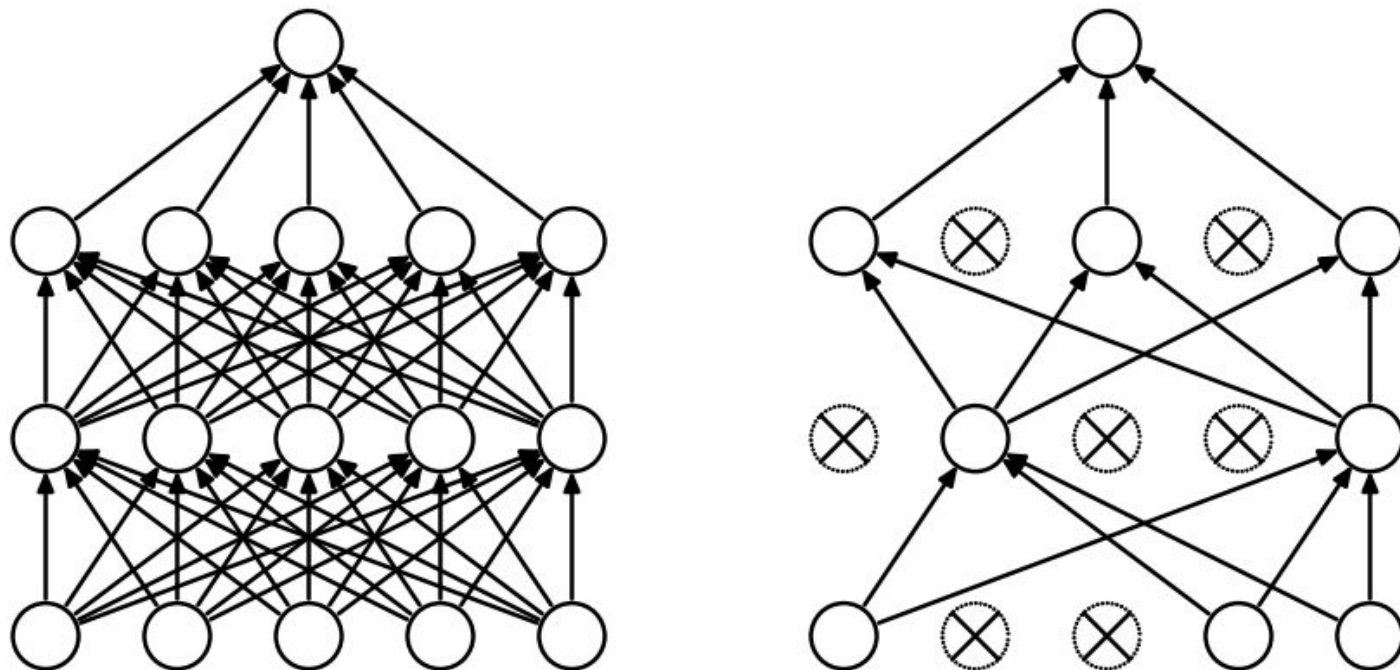
$$\begin{aligned} W^i &\sim N\left(0, \frac{2}{n_i}\right) \\ \text{or} \\ W^i &\sim N\left(0, \frac{2}{n_{i+1}}\right) \end{aligned}$$

# Инициализация He



Kaiming He et al.: Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification, 2015  
<https://arxiv.org/pdf/1502.01852>

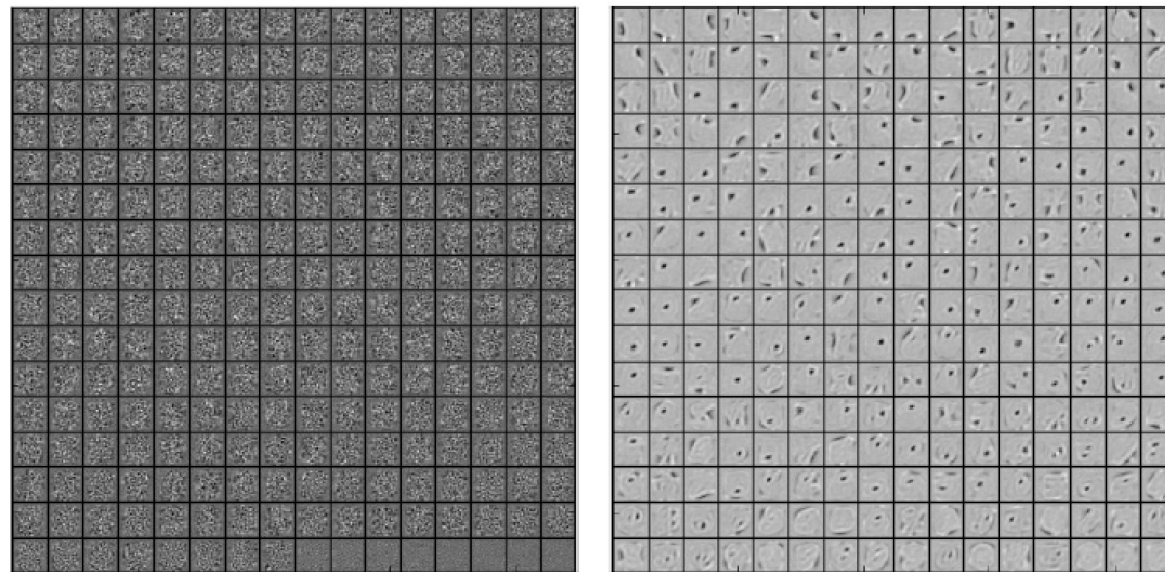
# Регуляризация: Dropout



- С вероятностью  $p$  занулим выход нейрона (например,  $p = 0.5$ )
- В test-time домножаем веса на вероятность сохранения
- Не стоит выкидывать нейроны последнего слоя

# Регуляризация: Dropout

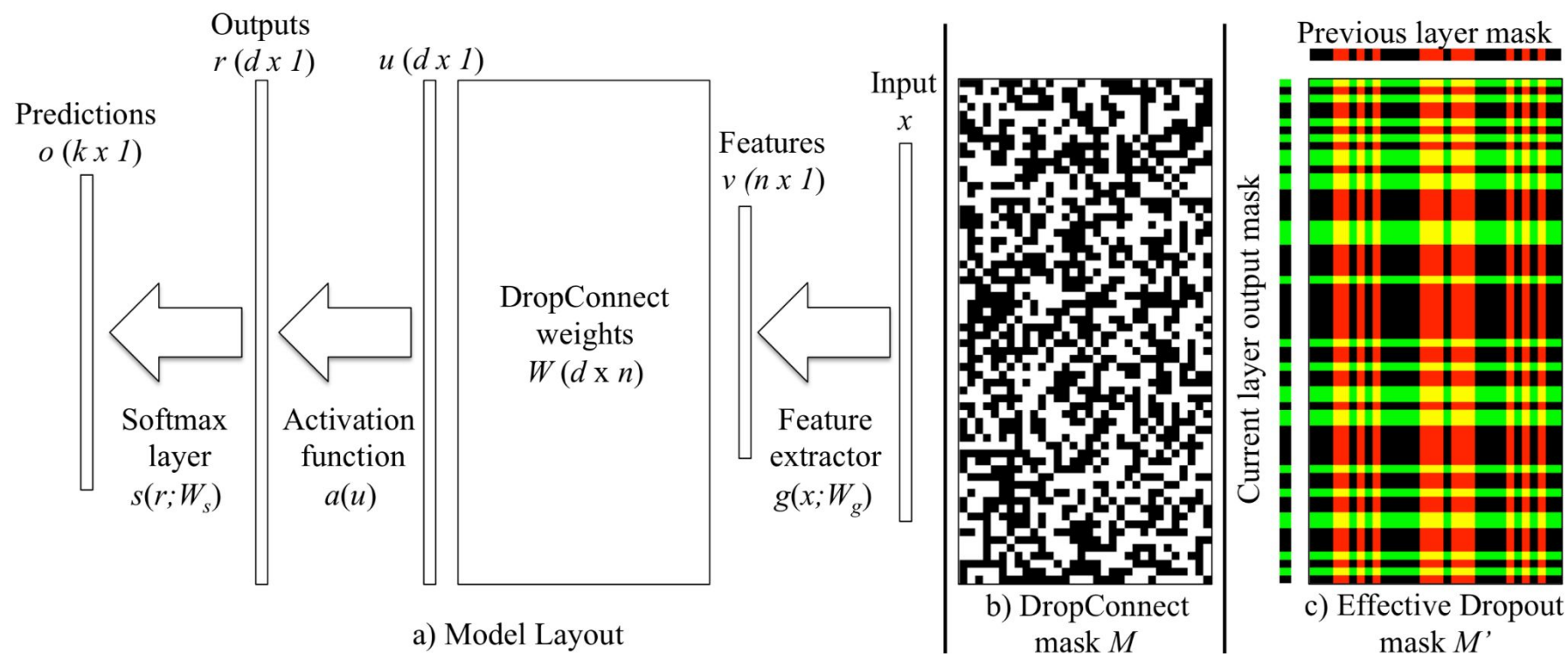
- Борьба с коадаптацией – нейроны больше не могут рассчитывать на наличие соседей
- Биология: не все гены родителей будут присутствовать у потомков
- Усреднение большого ( $2n$ ) числа моделей



Выученные признаки на MNIST.  
Слева: без Dropout, справа – с Dropout

# Регуляризация: Dropconnect

Зануляем не выходы нейронов, а каждый вес по отдельности



# Batch Norm

- Covariate shift: изменение распределения входов во время обучения
- Цель — уменьшить covariate shift скрытых слоев
- Нормализуем значения по батчу
- Для инференса накапливаем статистику экспоненциальным средним

$$E_{i+1} = (1 - \alpha)E_i + \alpha E_{\mathcal{B}}$$

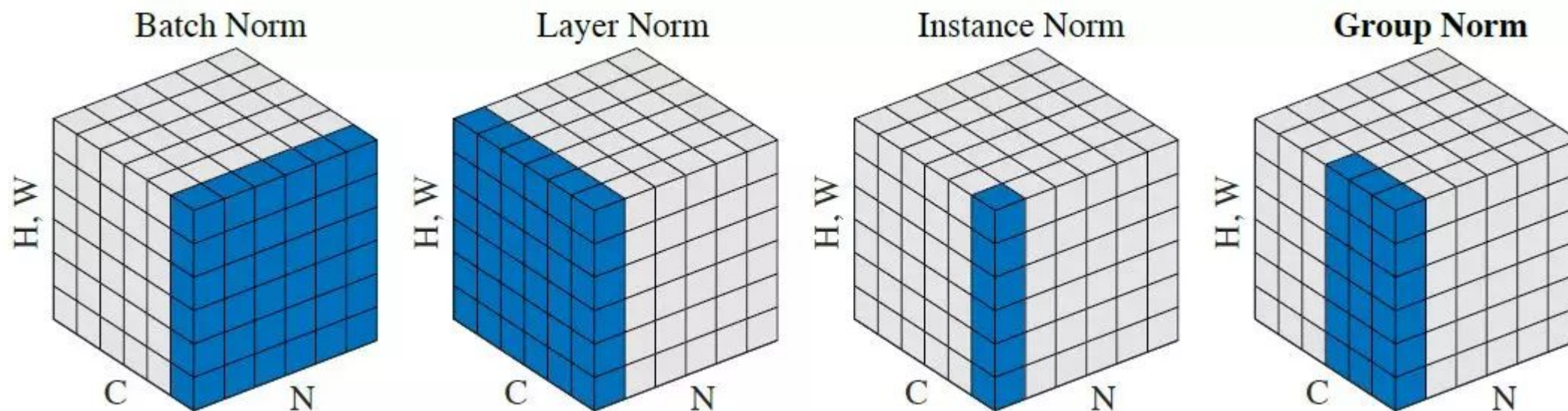
$$\hat{x} = \frac{x - \mathbb{E}[x]}{\sqrt{\mathbb{D}[x] + \epsilon}}$$

$$y = \gamma \cdot \hat{x} + \beta$$

$$y = \frac{\gamma}{\sqrt{\mathbb{D}[x] + \epsilon}} \cdot x + \left( \beta - \frac{\gamma \mathbb{E}[x]}{\sqrt{\mathbb{D}[x] + \epsilon}} \right)$$



# Нормализация



	Weight matrix re-scaling	Weight matrix re-centering	Weight vector re-scaling	Dataset re-scaling	Dataset re-centering	Single training case re-scaling
Batch norm	Invariant	No	Invariant	Invariant	Invariant	No
Weight norm	Invariant	No	Invariant	No	No	No
Layer norm	Invariant	Invariant	No	Invariant	No	Invariant

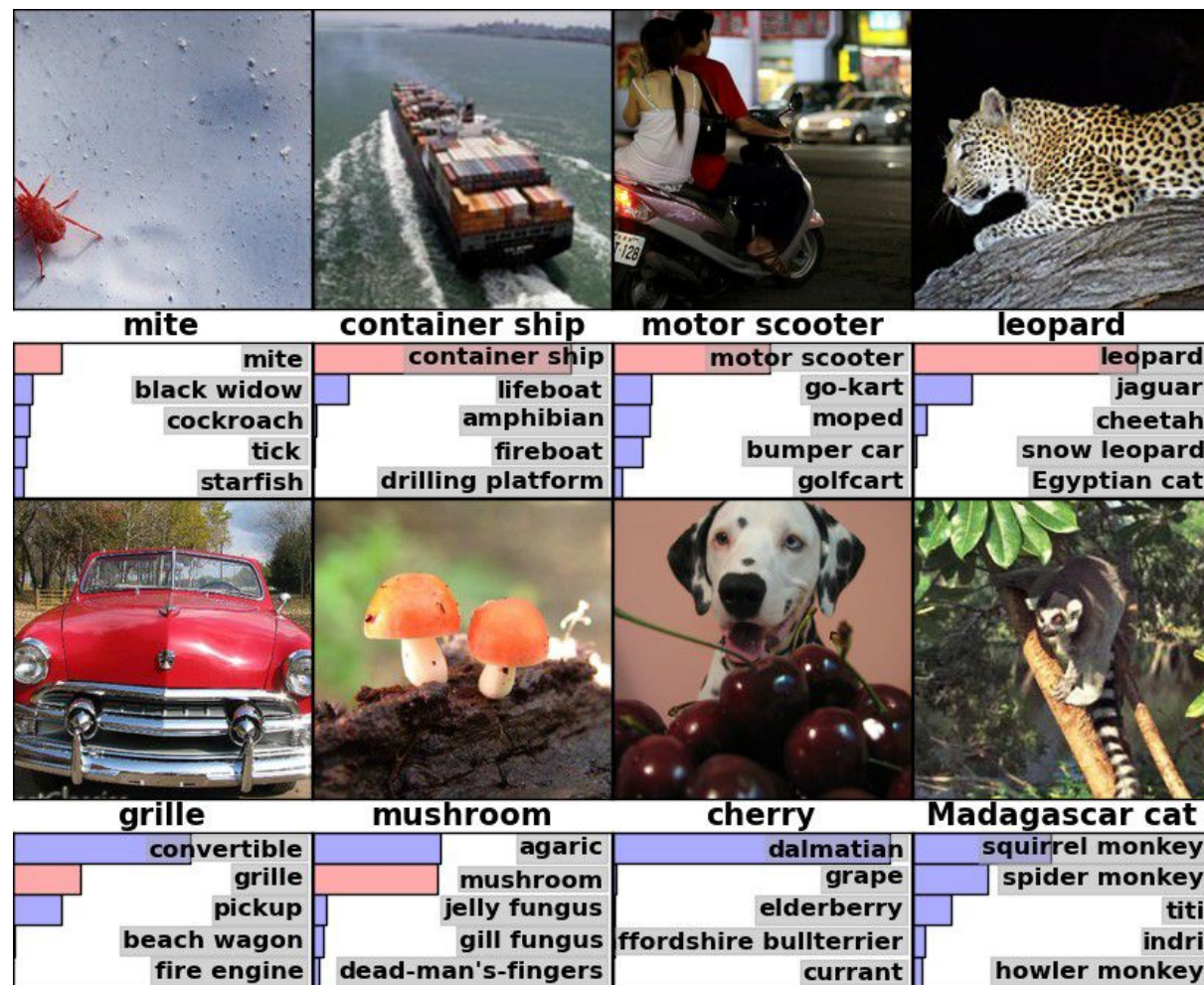


# Архитектуры CNN

# Датасет ImageNet

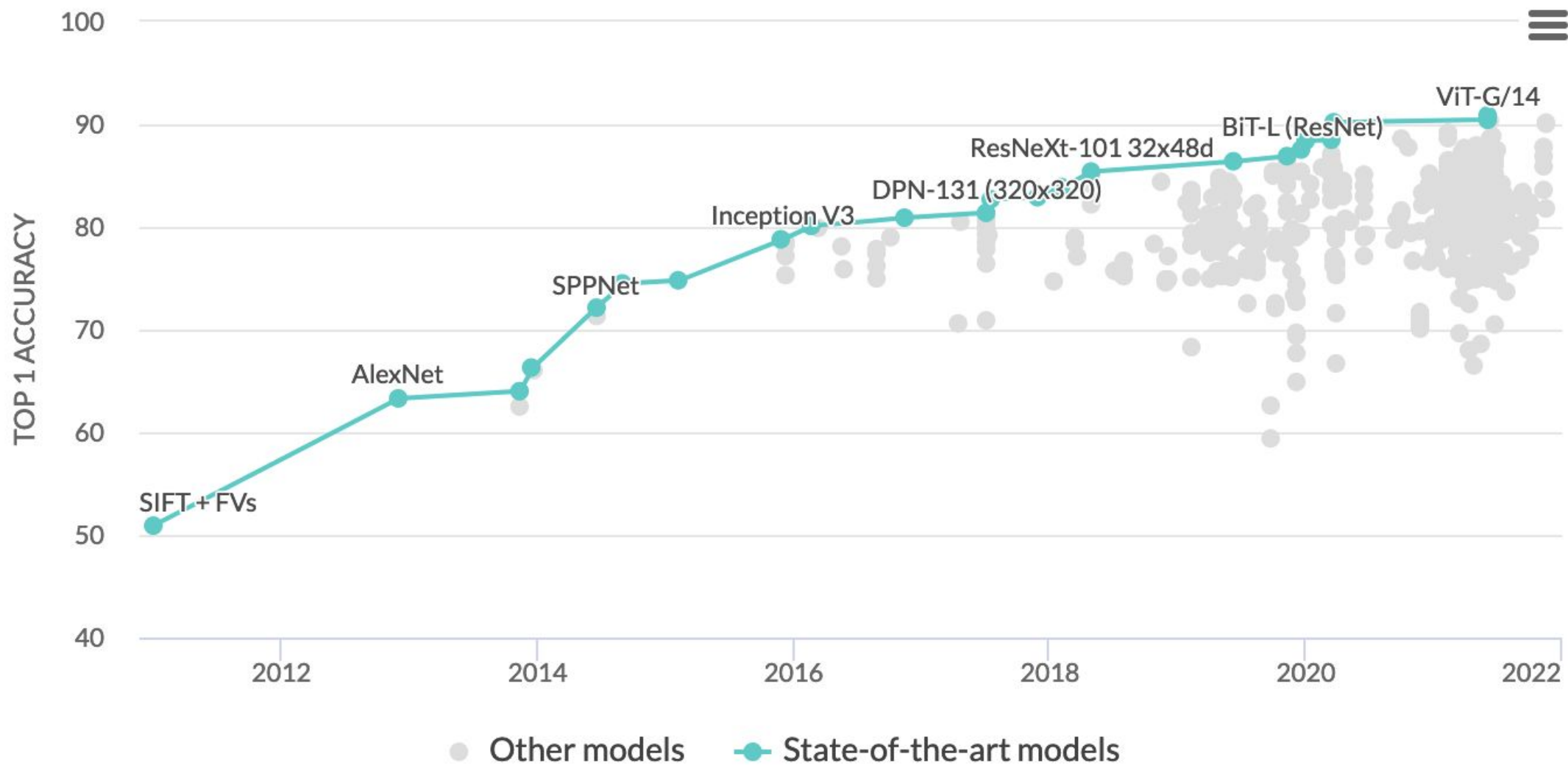


- 1000 классов
- Около 1000 изображений в каждом классе
- Около 1 000 000 изображений всего
- Несколько номинаций: таких как распознавание и детектирование/локализация

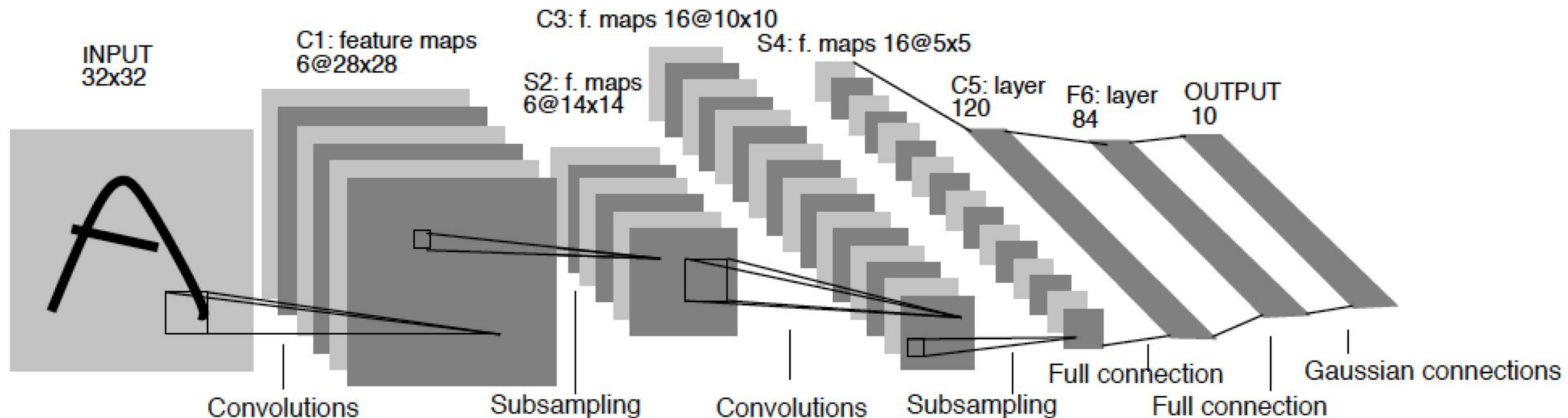


Примеры предсказаний  
на ImageNet

# Прогресс на ImageNet



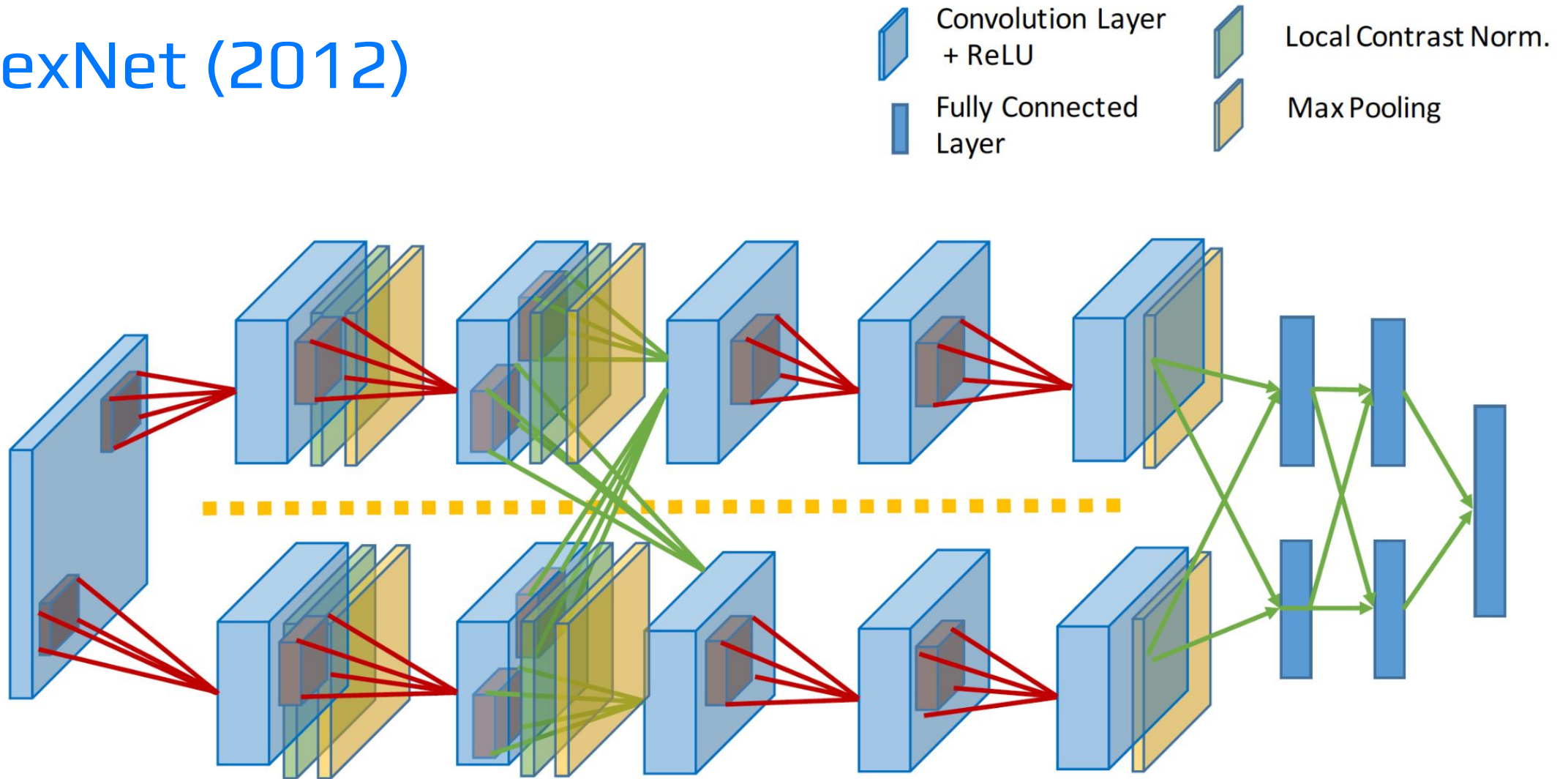
# LeNet (1989)



Использованные подходы:

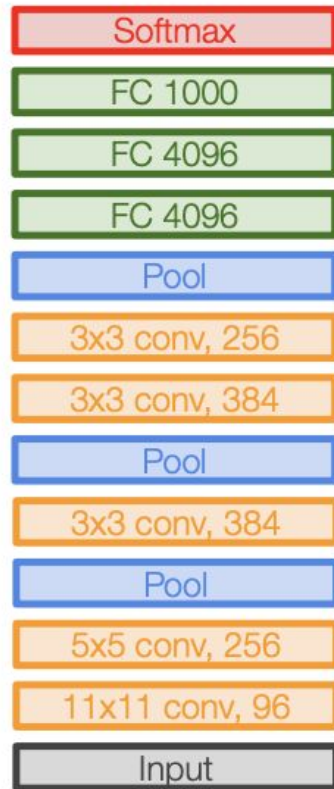
- Сверточные слои
- Активация sigmoid
- Max pooling

# AlexNet (2012)





# AlexNet (2012)

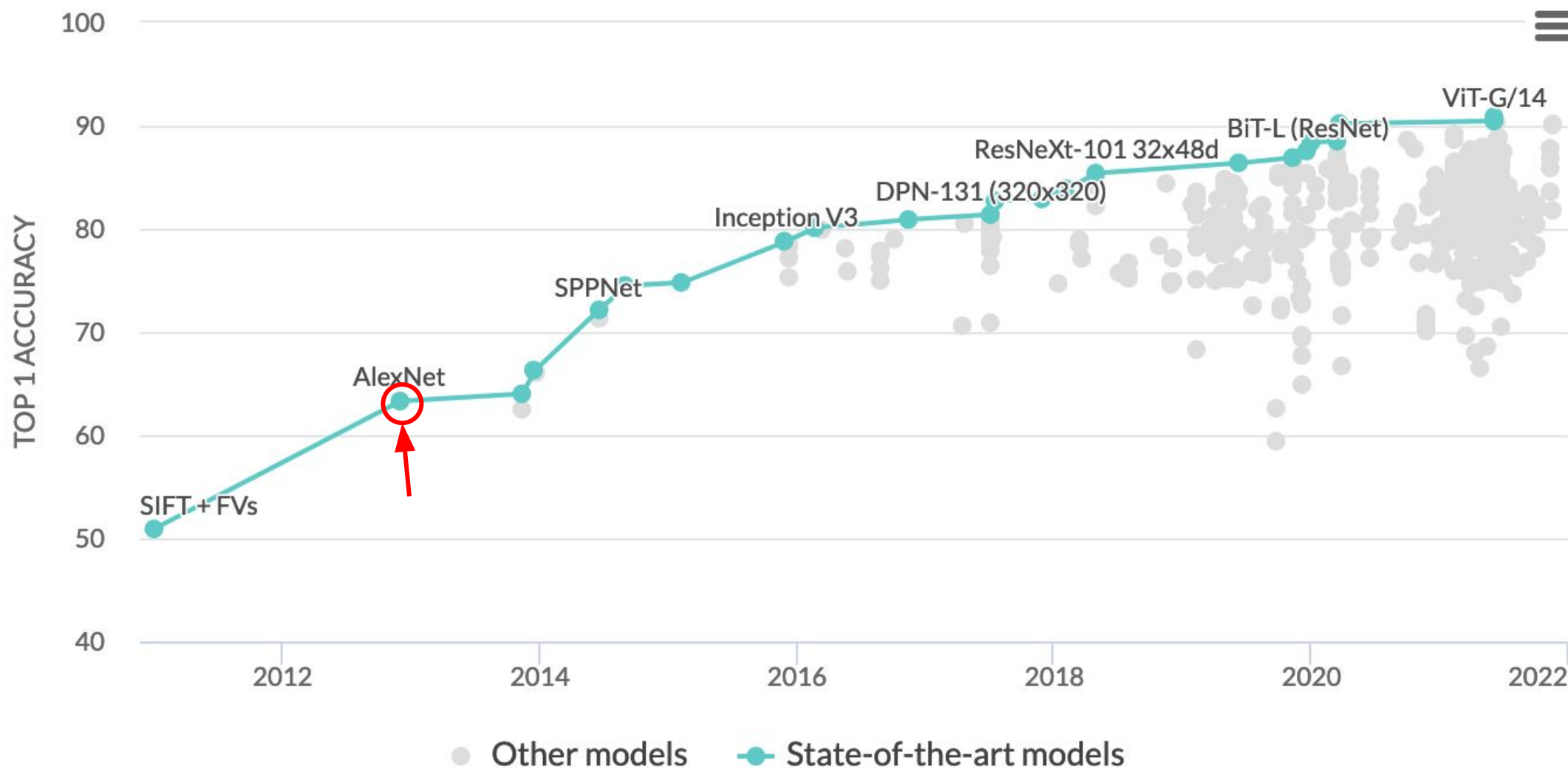


## AlexNet

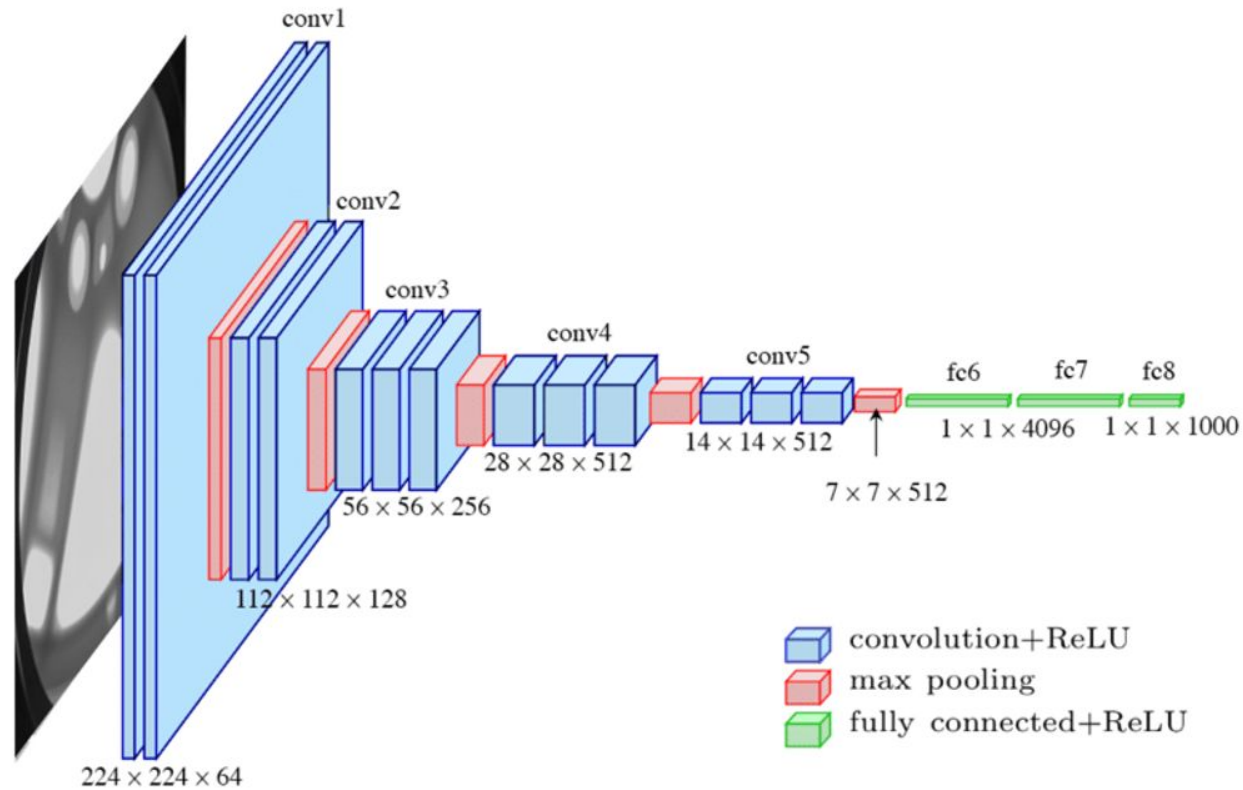
Source: Stanford CS231n lecture 6

- 5 сверточных и 3 полносвязных слоя
- 60M параметров
- Распараллеливание на 2 GPU
- Свёртки 11x11, 5x5, 3x3
- ReLU, т.к. не надо вычислять exp
- Dropout
- Затухание learning rate
- Аугментации: случайный 224x224 кроп, горизонтальное отражение, шум

# Прогресс на ImageNet: AlexNet



# VGG (2014)



VGG-16 architecture

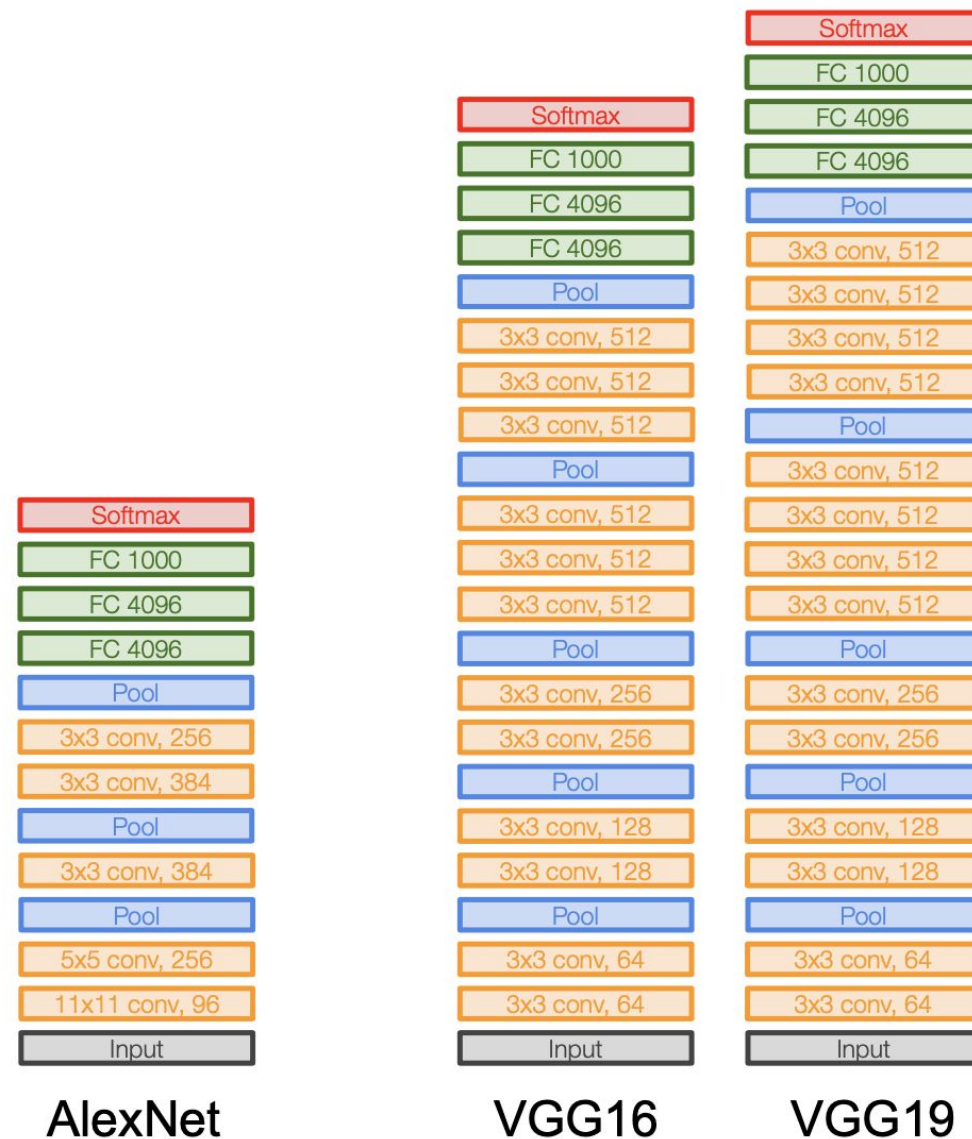
Karen Simonyan, Andrew Zisserman: Very Deep Convolutional Networks for Large-Scale Image Recognition, 2014

<https://arxiv.org/pdf/1409.1556.pdf>



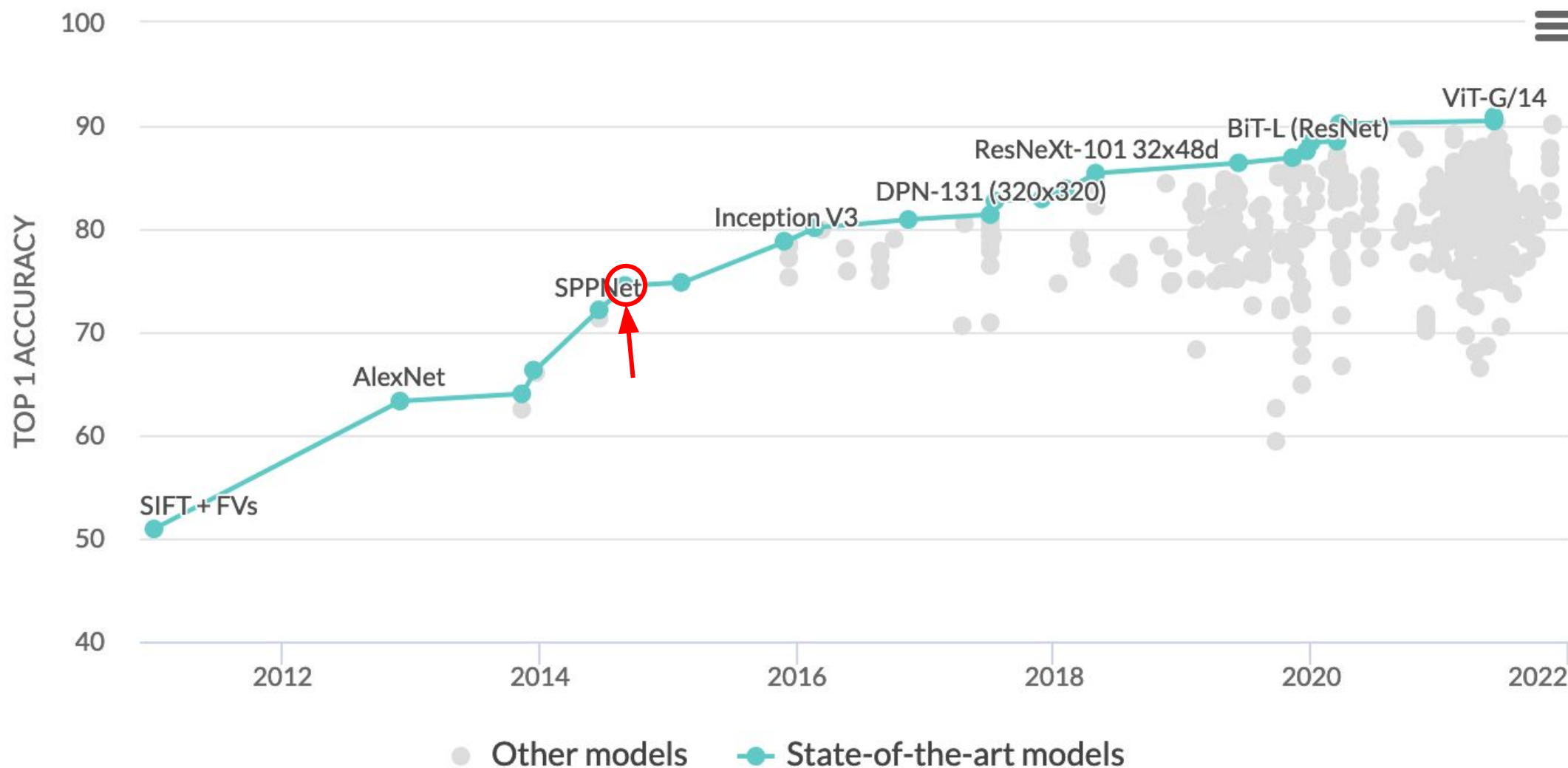
# VGG (2014)

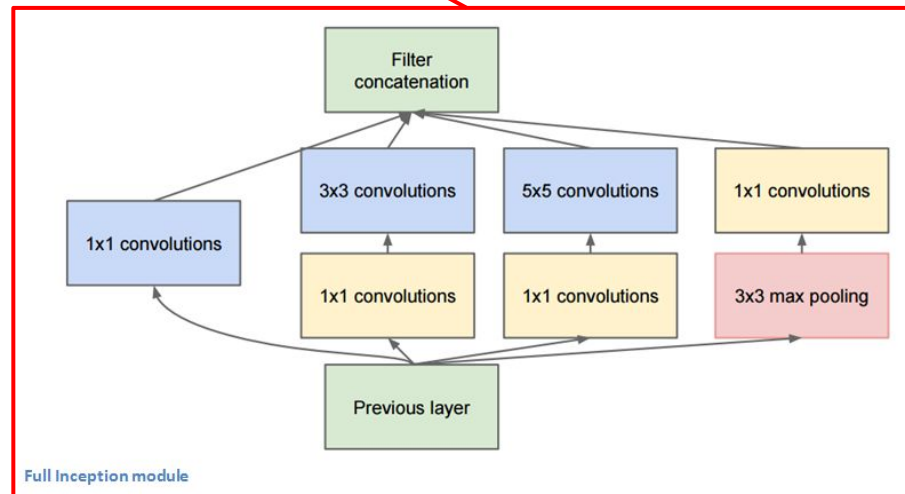
- VGG-19 (E): 144M параметров
- Только свертки 3x3
- Больше слоев (Deep Neural Network)
- Выходы промежуточных слоев можно использовать как “семантику” изображения:
  - в качестве функции потерь  
<https://arxiv.org/abs/1603.08155>
  - для переноса стиля  
[https://www.cv-foundation.org/openaccess/content\\_cvpr\\_2016/papers/Gatys\\_Image\\_Style\\_Transfer\\_CVPR\\_2016\\_paper.pdf](https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/Gatys_Image_Style_Transfer_CVPR_2016_paper.pdf)
  - для оценки качества  
<https://arxiv.org/abs/1801.03924>



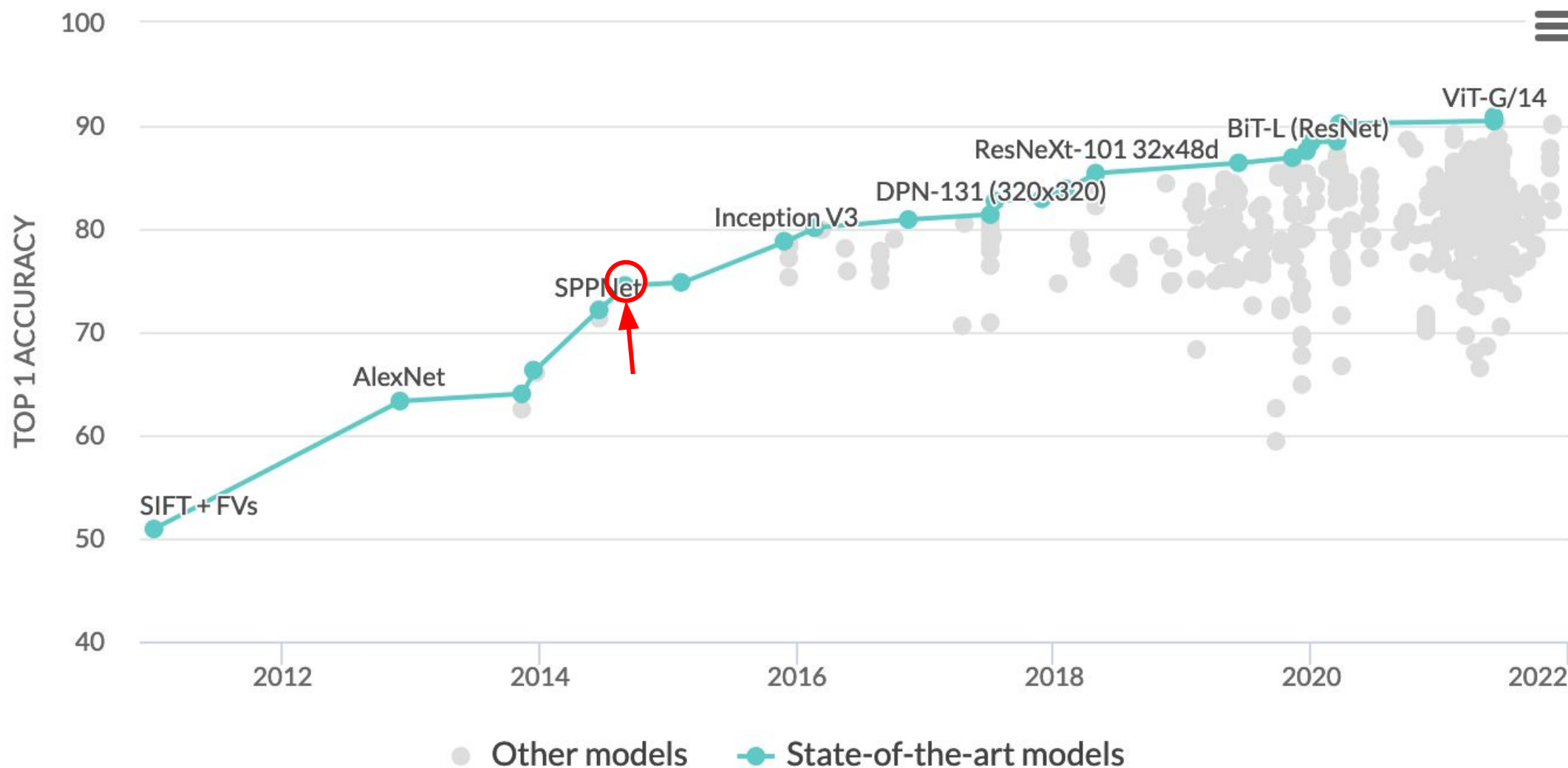
Source: Stanford CS231n lecture 6

# Прогресс на ImageNet: VGG





# Прогресс на ImageNet: GoogLeNet



# Скачок глубины нейронных сетей

AlexNet, 8 layers  
(ILSVRC 2012)



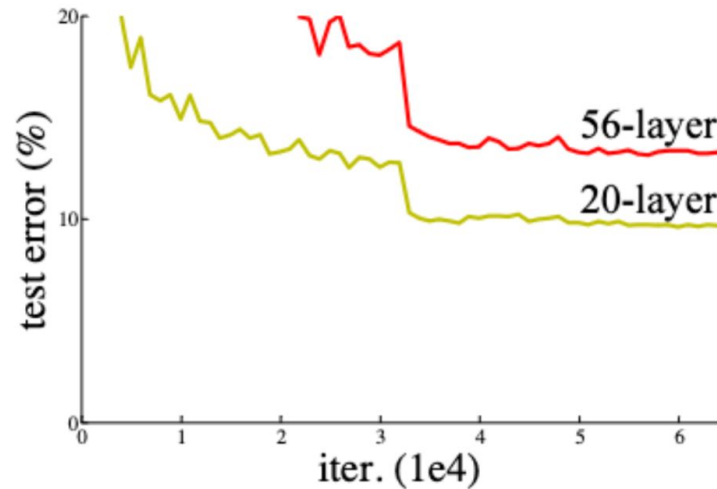
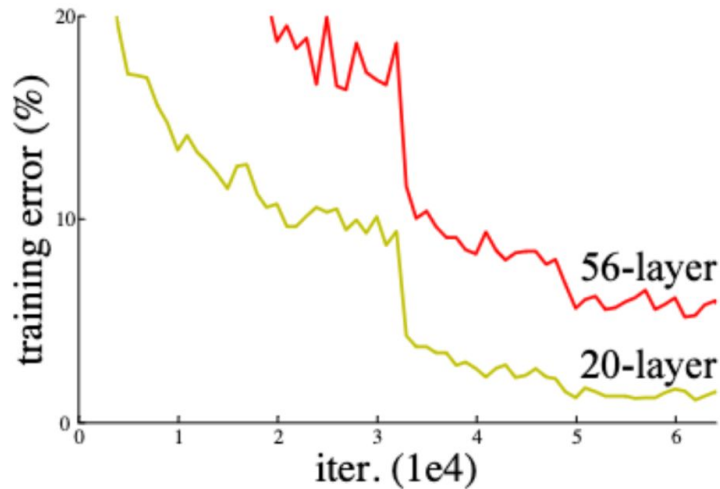
VGG, 19 layers  
(ILSVRC 2014)



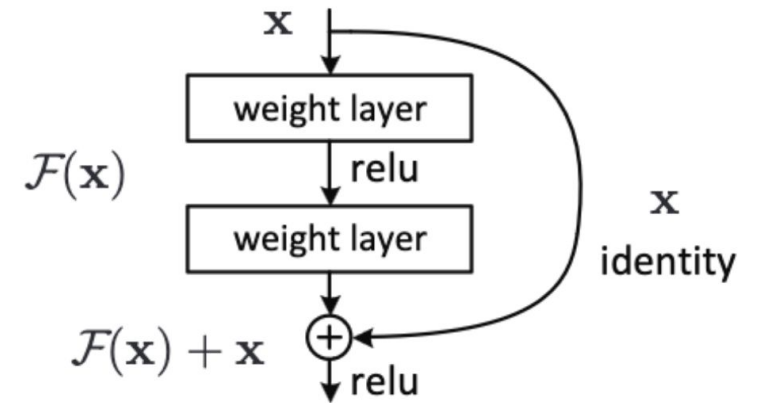
ResNet, 152 layers  
(ILSVRC 2015)



# ResNet (2015)



Обучение “обычной сети”

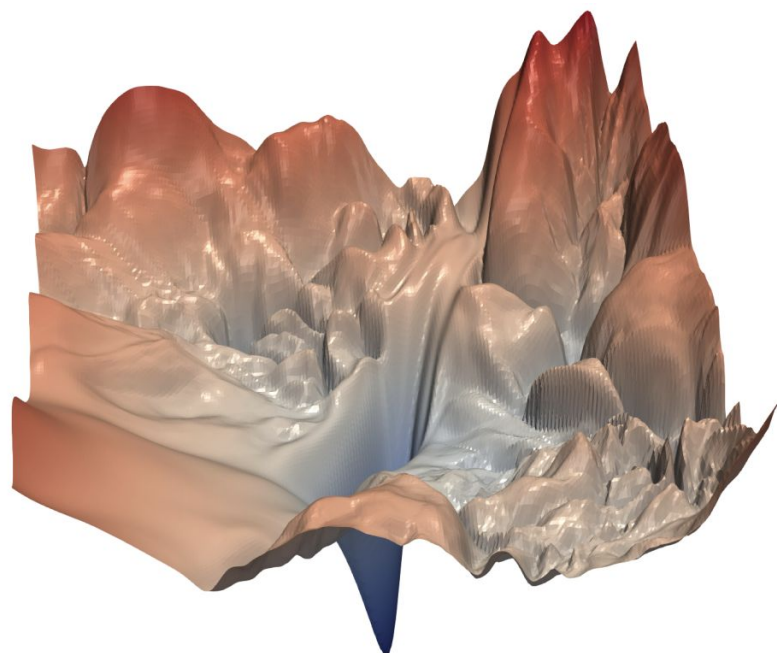


Residual block

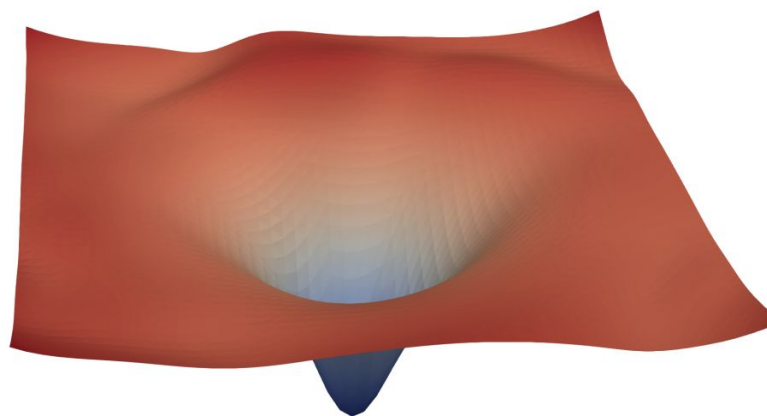
- Результат на 56 слоях хуже. Проблема не в переобучении
- Решение заведомо существует: 20 слоев, затем  $F(x) = x$
- Выучить  $F(x) = x$  тяжело, а  $F(x) = 0$  просто
- Residual block решает эту проблему



# ResNet (2015)



(a) without skip connections



(b) with skip connections

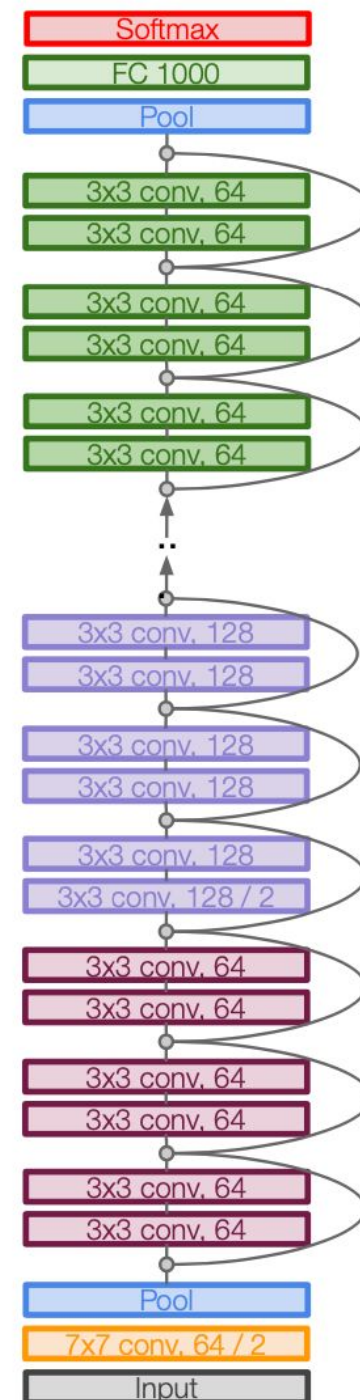
Figure 1: The loss surfaces of ResNet-56 with/without skip connections. The proposed filter normalization scheme is used to enable comparisons of sharpness/flatness between the two figures.

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

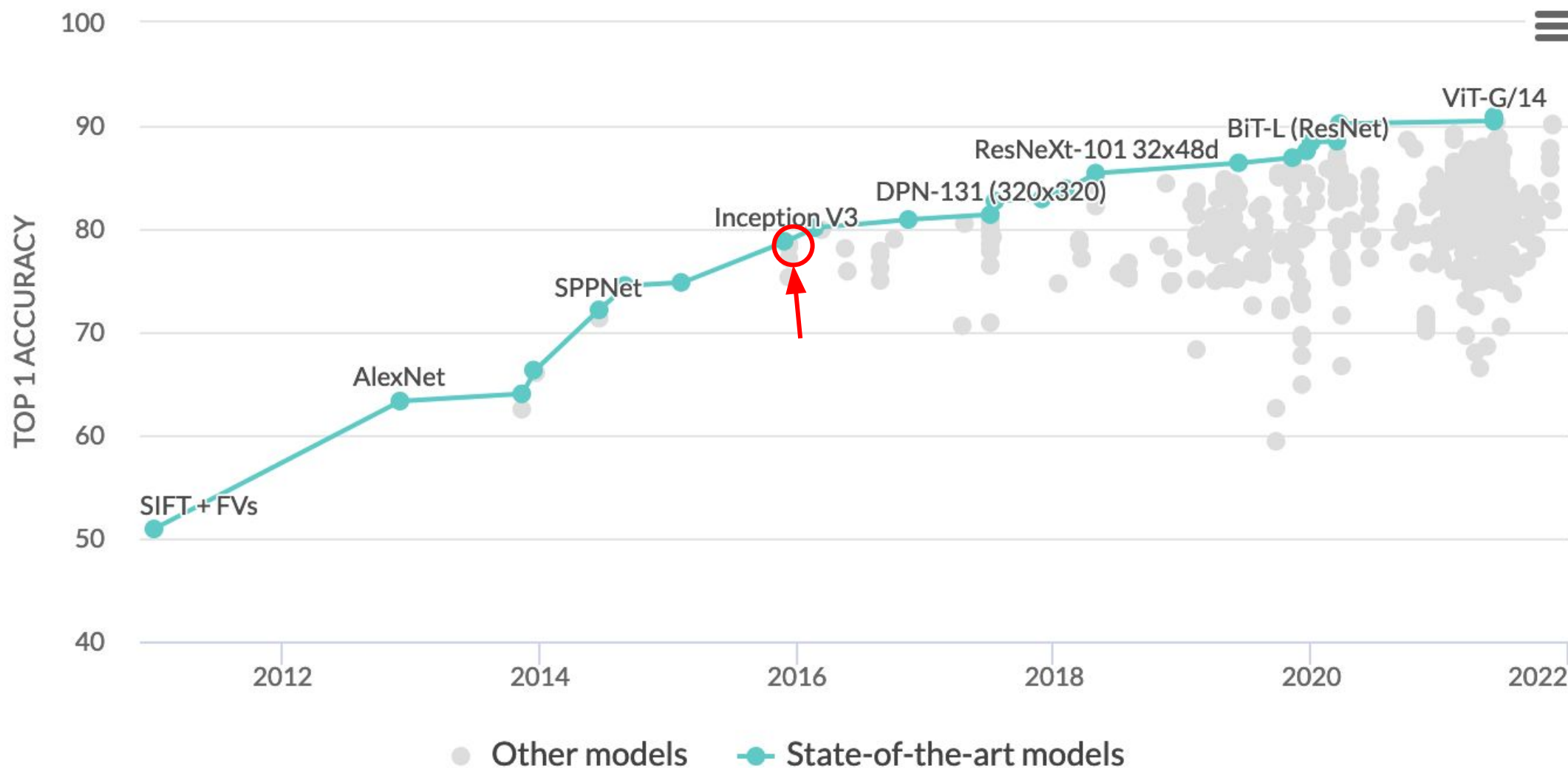
Kaiming He et al.: Deep Residual Learning for Image Recognition, 2015

<https://arxiv.org/pdf/1512.03385v1.pdf>

<https://arxiv.org/pdf/1712.09913.pdf>



# Прогресс на ImageNet: ResNet



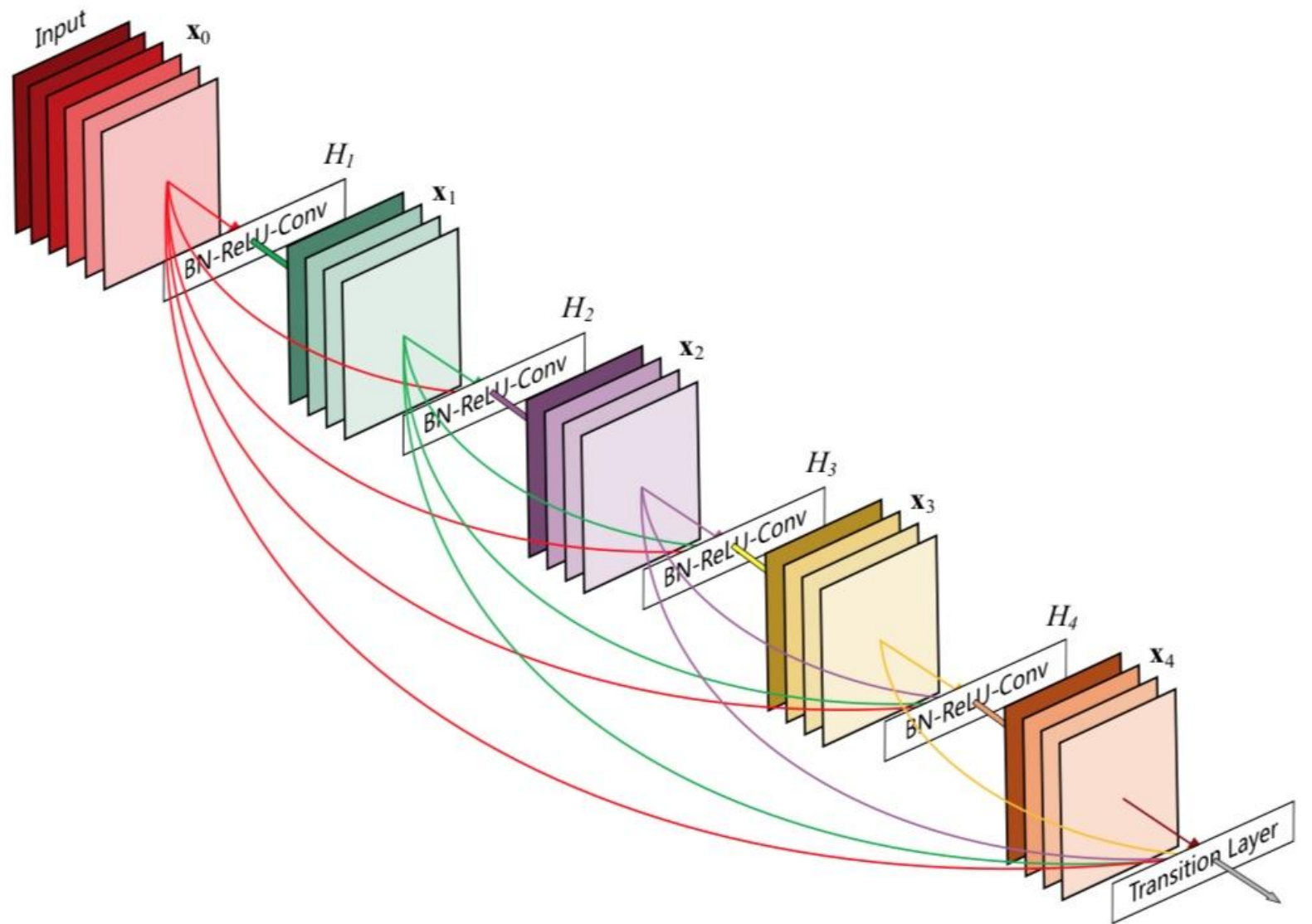


# Модификации моделей

Существует множество модификаций ResNet:

- ResNet in ResNet
- DenseNet
- ResNeXt
- WideResNets

# DenseNet



**Figure 1:** A 5-layer dense block with a growth rate of  $k = 4$ . Each layer takes all preceding feature-maps as input.

# ResNeXt

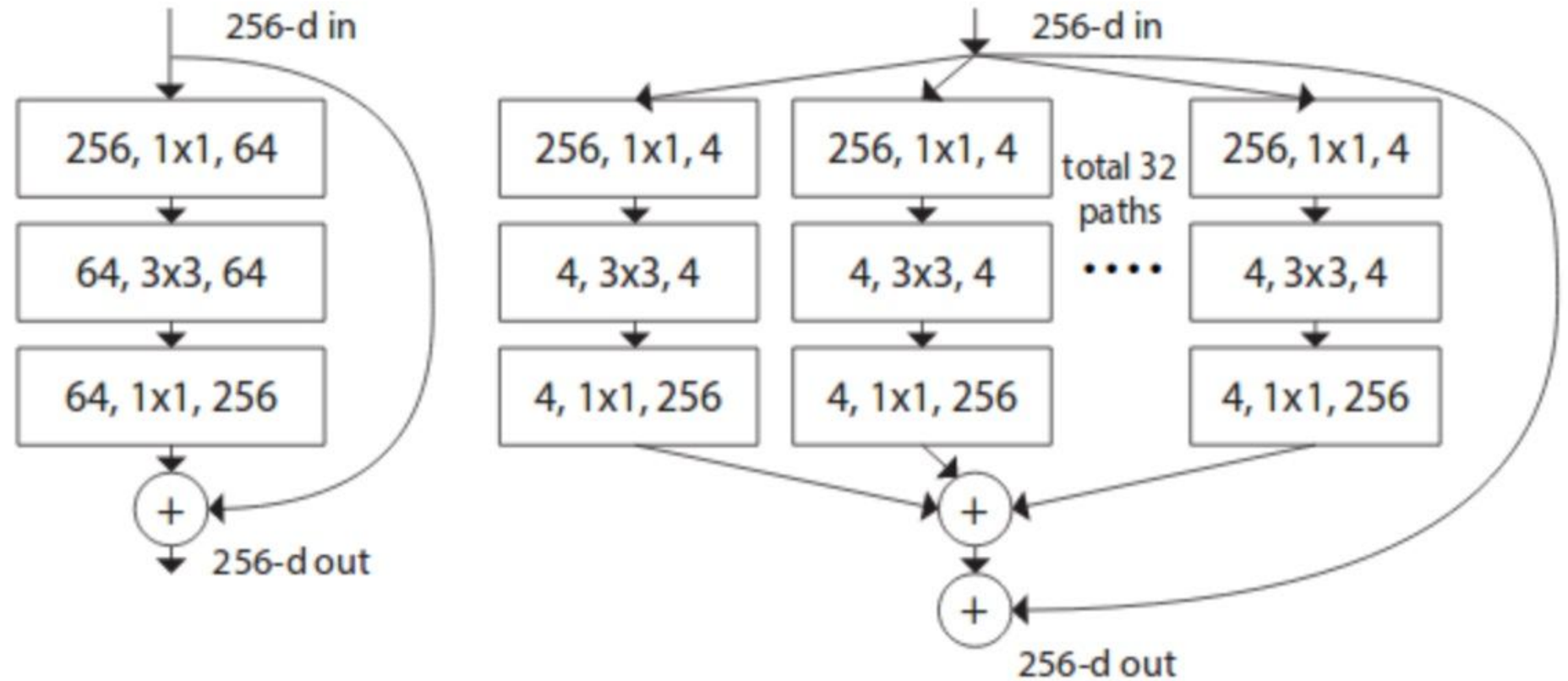


Figure 1. **Left:** A block of ResNet [14]. **Right:** A block of ResNeXt with cardinality = 32, with roughly the same complexity. A layer is shown as (# in channels, filter size, # out channels).

# Сравнение моделей

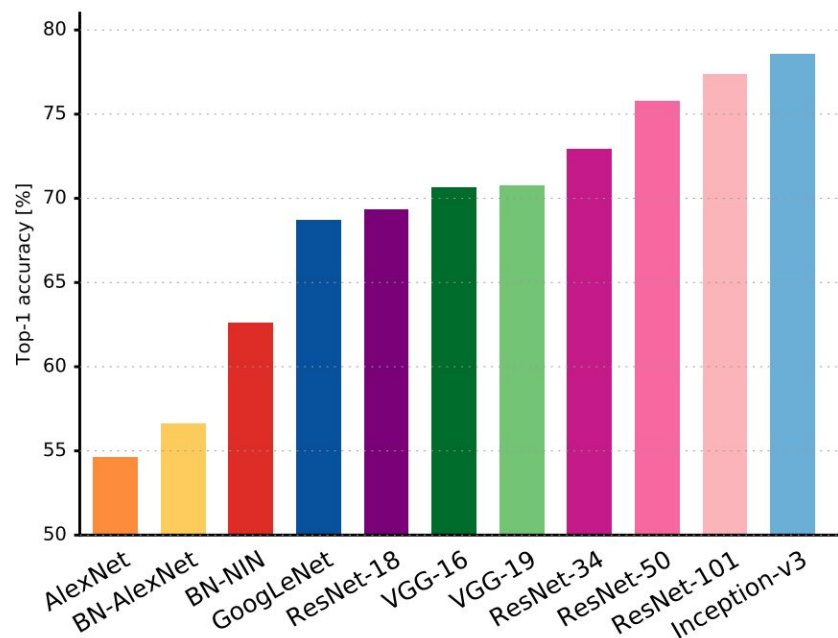


Figure 1: **Top1 vs. network.** Single-crop top-1 validation accuracies for top scoring single-model architectures. We introduce with this chart our choice of colour scheme, which will be used throughout this publication to distinguish effectively different architectures and their correspondent authors. Notice that network of the same group share colour, for example ResNet are all variations of pink.

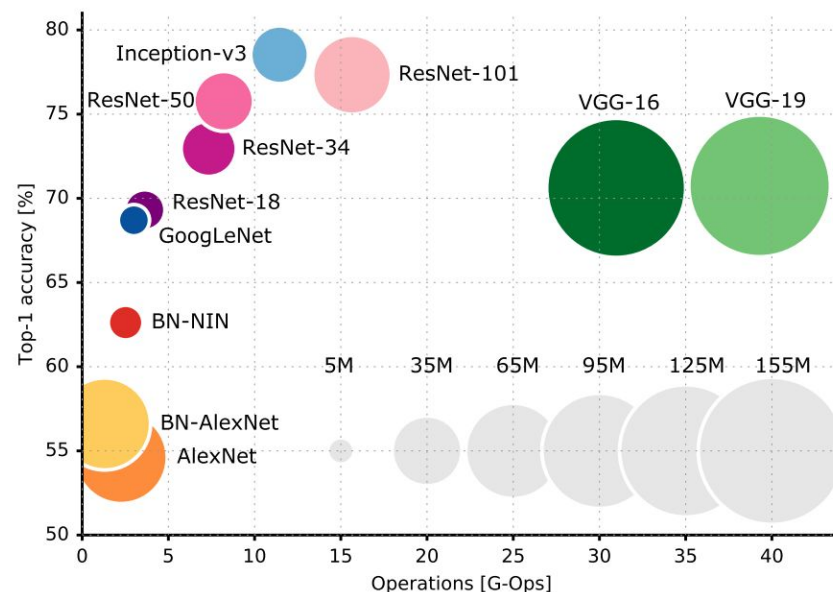


Figure 2: **Top1 vs. operations, size  $\propto$  parameters.** Top-1 one-crop accuracy versus amount of operations required for a single forward pass. The size of the blobs is proportional to the number of network parameters; a legend is reported in the bottom right corner, spanning from  $5 \times 10^6$  to  $155 \times 10^6$  params.

# MobileNet

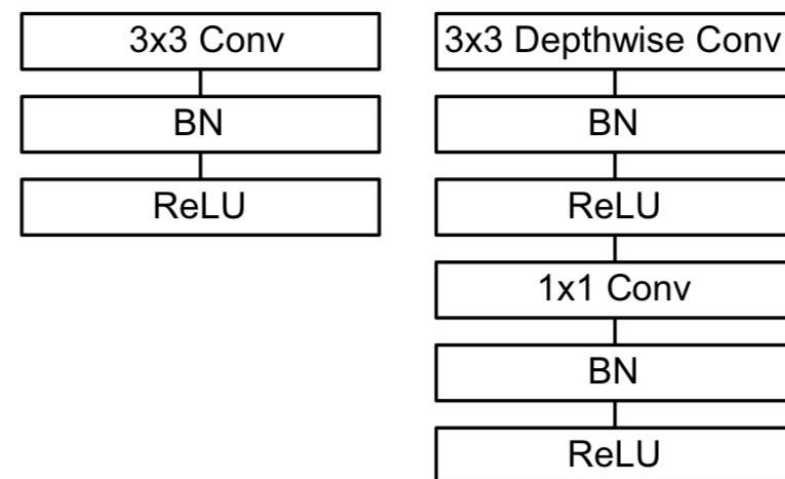
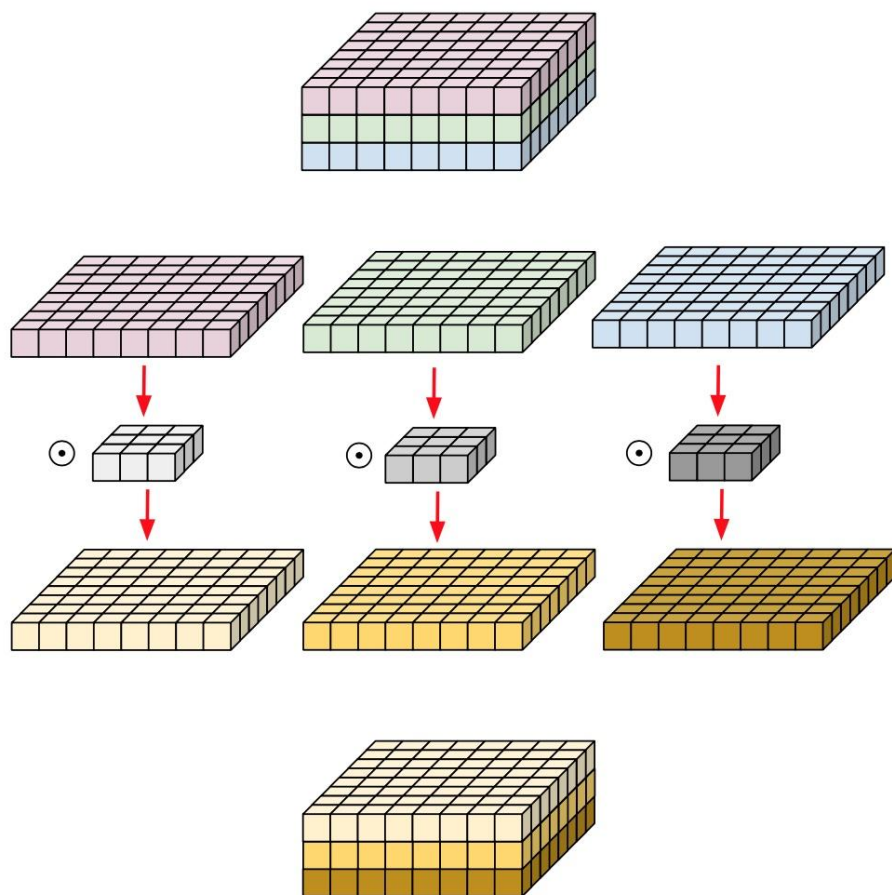


Figure 3. Left: Standard convolutional layer with batchnorm and ReLU. Right: Depthwise Separable convolutions with Depthwise and Pointwise layers followed by batchnorm and ReLU.



# EfficientNet

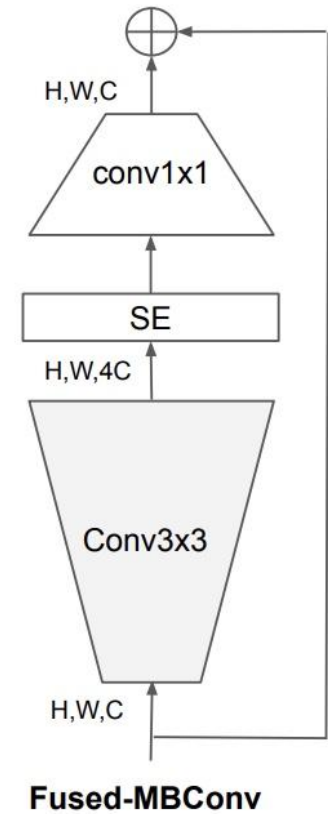
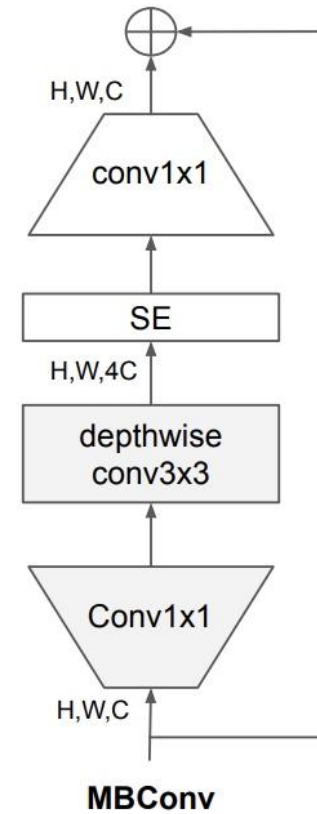
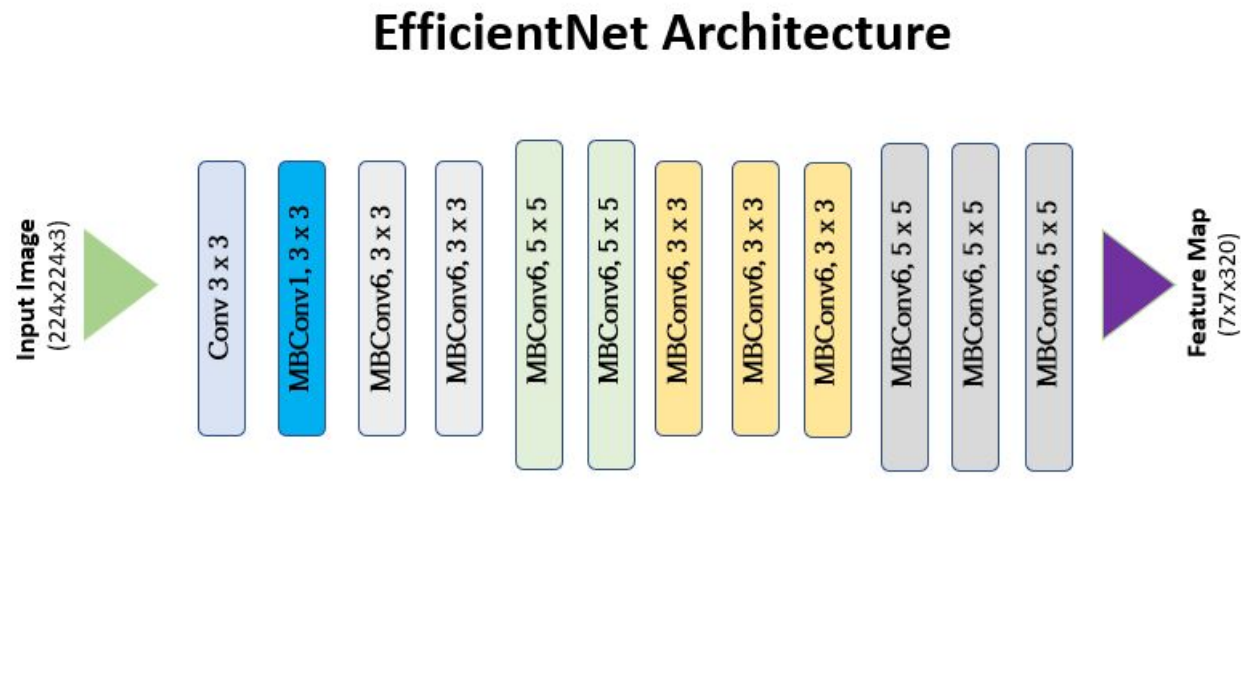
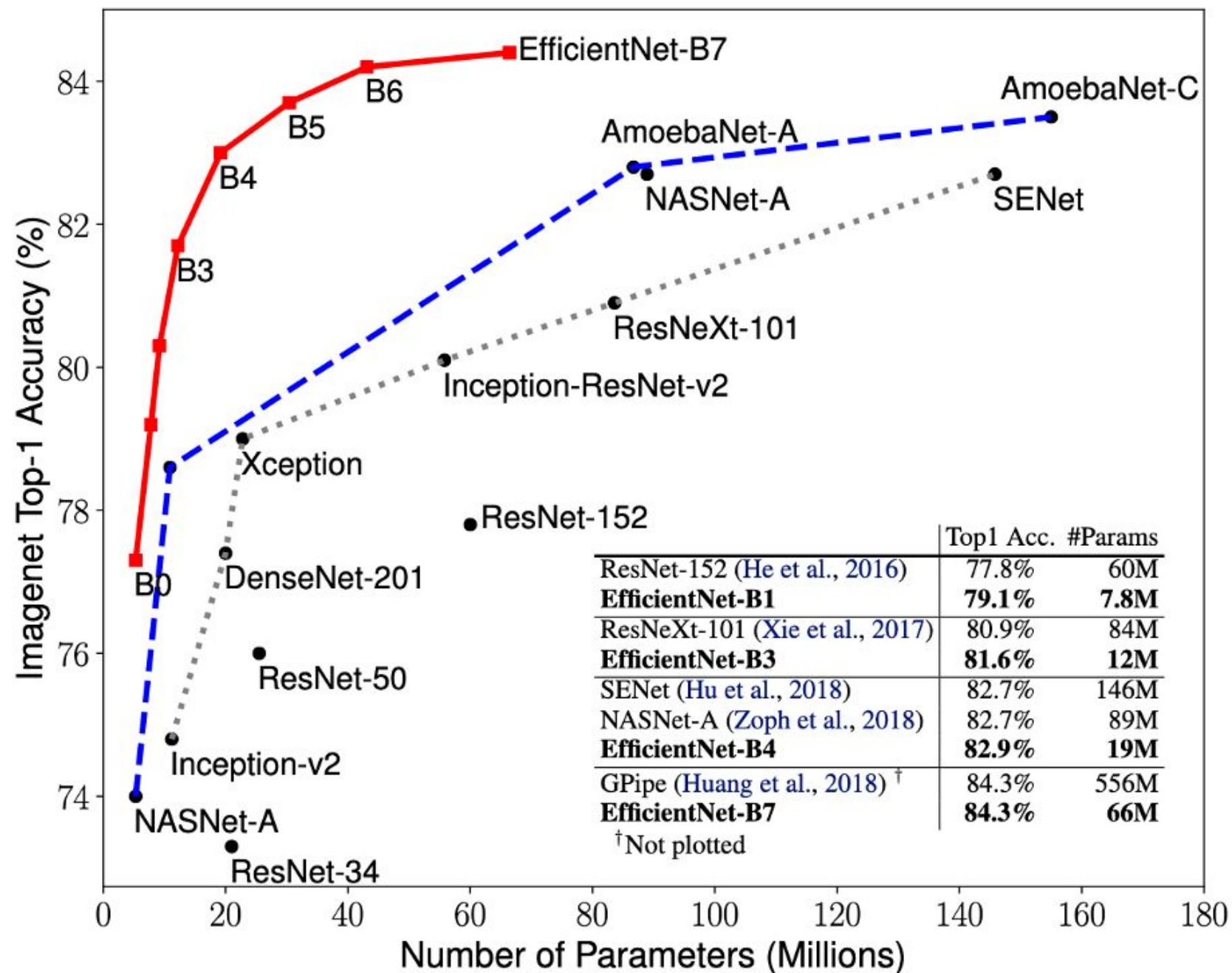


Figure 2. Structure of MBConv and Fused-MBConv.

# EfficientNet



# Резюме

В этой лекции мы изучили:

1. Подходы к оптимизации обучения:
  - a. Инициализация
  - b. Регуляризация
  - c. Нормализация
2. Архитектуры CNN и качество их работы на бенчмарке ImageNet



# Рекомендованные источники

1. Kaiming He et al, “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification,” 2015  
[Available online](#)
2. ImageNet benchmark –  
<https://paperswithcode.com/sota/image-classification-on-imagenet>
3. Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks”, 2012  
[Available online](#)
4. Karen Simonyan, Andrew Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” 2014  
[Available online](#)
5. Christian Szegedy et al, “Going Deeper with Convolution”, 2014  
[Available online](#)
6. Kaiming He et al, “Deep Residual Learning for Image Recognition”, 2015  
[Available online](#)

# Спасибо!

