Московский
государственный
университет
имени М.В. Ломоносова

ГЕНЕРАЦИЯ ИЗОБРАЖЕНИЙ ДЛЯ АНАЛИЗА УСТОЙЧИВОСТИ НЕЙРОННЫХ СЕТЕЙ В ЗАДАЧЕ КЛАССИФИКАЦИИ

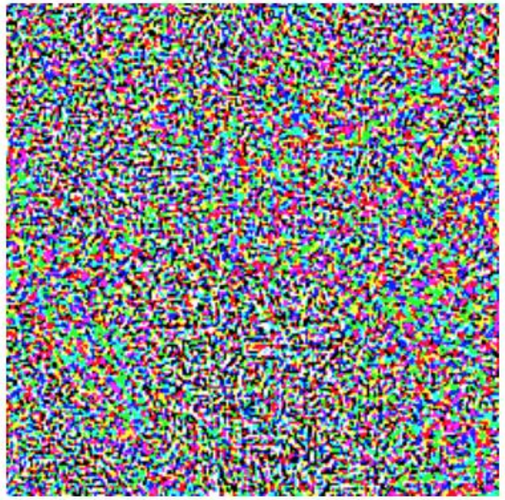
Студент 535 группы | Клиентов Г.А.

Научные руководители:
д. ф.-м. н., профессор | Голубцов П.В.
д. м.-м. н., профессор РАН | Дьяконов А.Г.

27.12.2024

Adversarial attack на примере FGSM

$$+ .007 \times$$



$$=$$

$\boldsymbol{x}$

“panda”
57.7% confidence

$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$

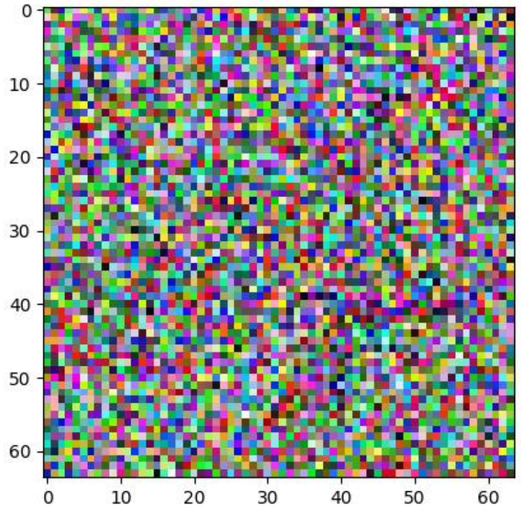
“nematode”
8.2% confidence

$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$

“gibbon”
99.3 % confidence

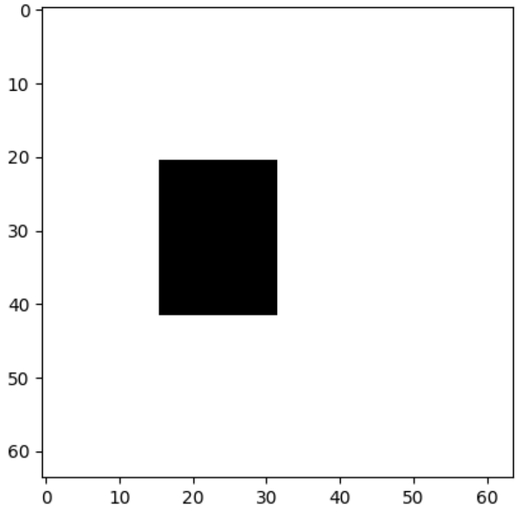
Adversarial Example Generation — PyTorch Tutorials
https://pytorch.org/tutorials/beginner/fgsm_tutorial.html2.3.0+cu121 documentation

Можем ли мы подобрать изображение, которое бы относилось к любому классу?



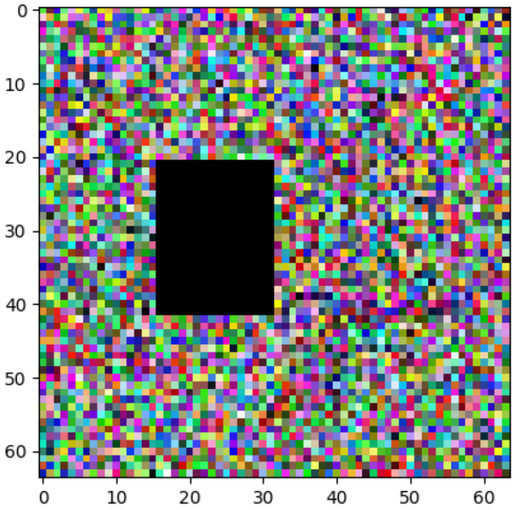
Универсальное изображение

×



Маска для класса **“gibbon”**

=



“gibbon”

Способ генерации изображения

- Используем метод градиентного спуска
- Фиксируем веса модели
- В качестве параметров для оптимизации используются значения пикселей самого изображения
- Батч собираем из всего набора масок (размер батча равен количеству классов)

Используемые модели

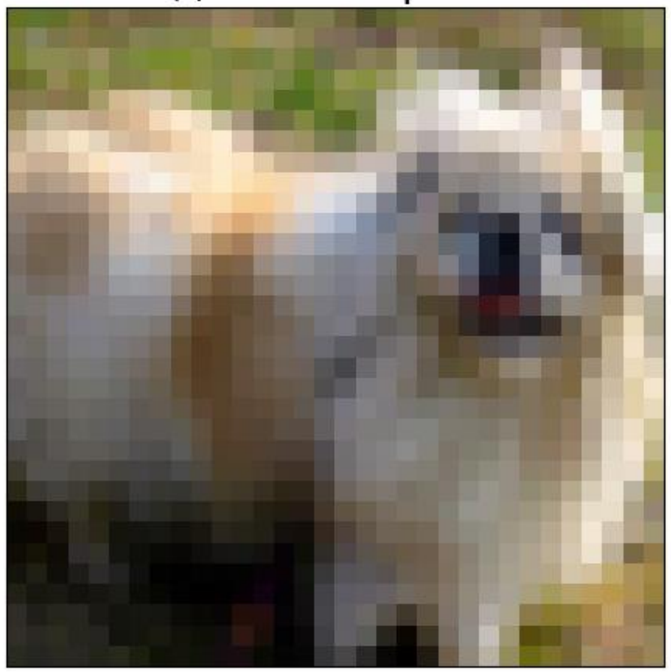
- Несколько SOTA моделей, предобученных на ImageNet-1K
- Размер искомого изображения – 3x64x64 (RGB)
- Количество классов – от 2 до 62 с шагом 8
- Количество эпох – до 5000
- Loss-функция – CrossEntropyLoss
- Оптимизатор – SGD с learning rate=0.1

Используемые модели

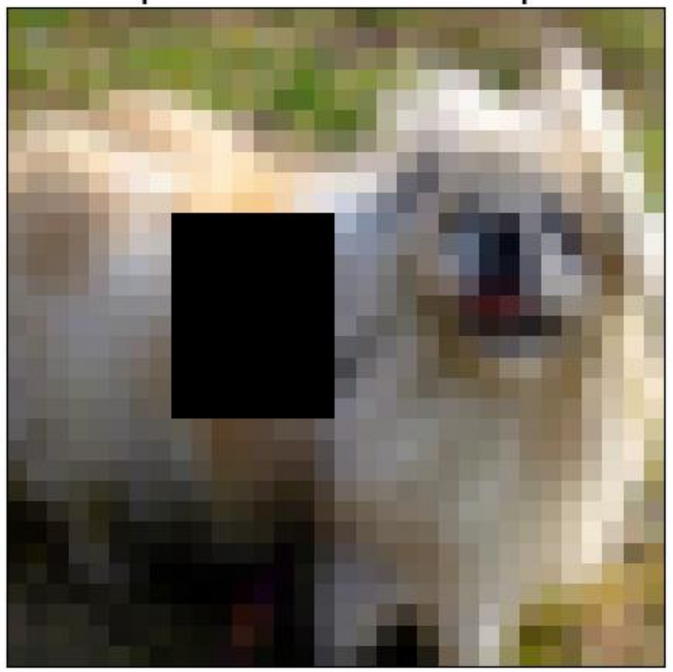
Модель	Acc@1	Acc@5	Params	GFLOPS
ResNet-50	80.9 %	95.4 %	25.6M	4.09
VGG-16	71.6 %	90.4 %	138.4M	15.47
EfficientNet_B4	**83.3 %**	**96.6 %**	19.3M	4.39
AlexNet	56.5 %	79.0 %	61.1M	0.71
GoogleNet	69.8 %	89.5 %	6.6M	1.5
MobileNet_v3_Small	67.7 %	87.4 %	**2.5M**	**0.06**

Эксперимент 1
(маска, разбитая на k равных частей)

Исходное изображение



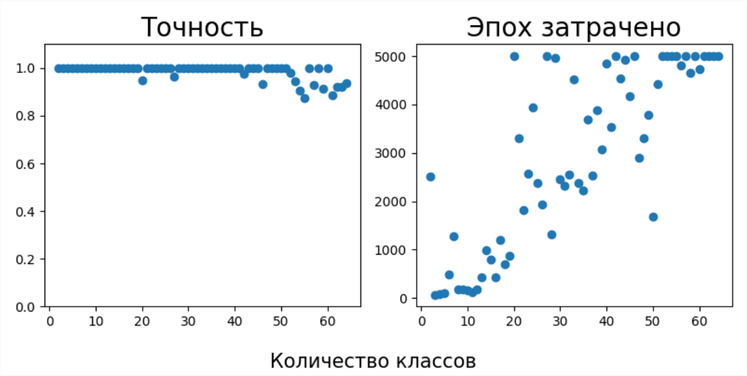
Скрыта часть номер 5

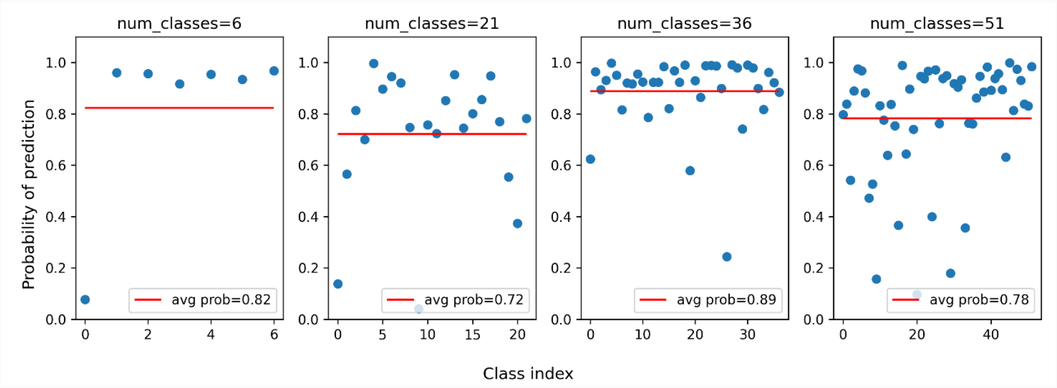


Уменьшаем рецептивное поле модели, из-за чего она классифицирует изображение только по немаскированной части

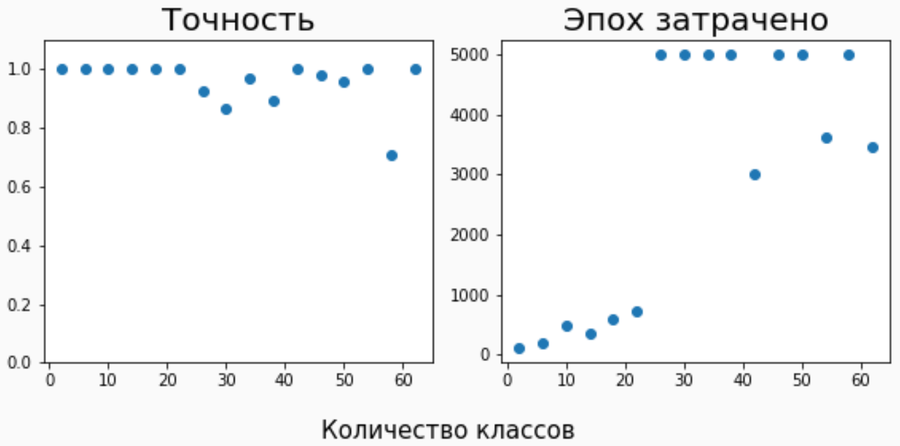
Эксперимент 1 (маска, разбитая на k равных частей)

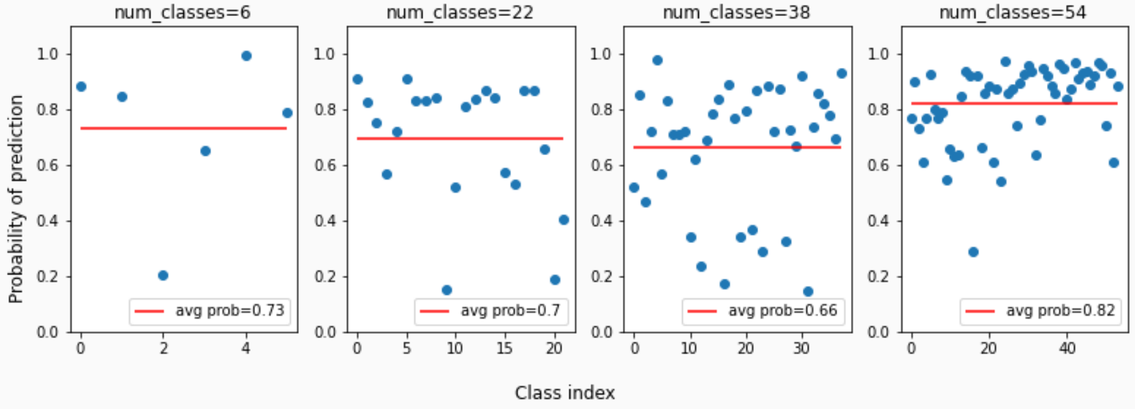
ResNet-50





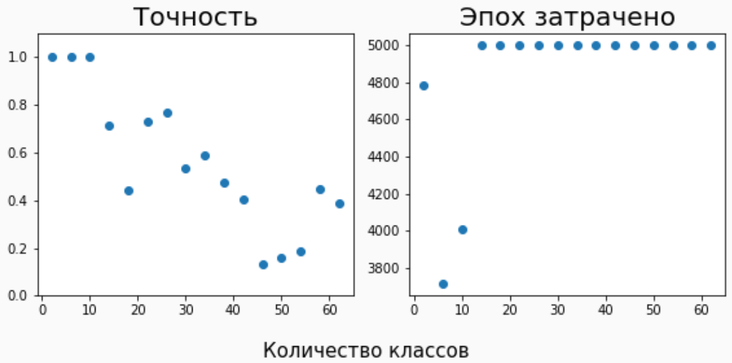
VGG-16

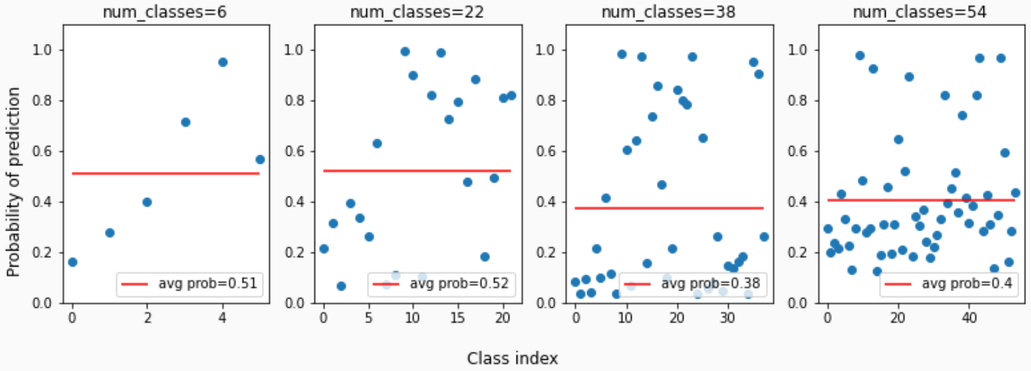




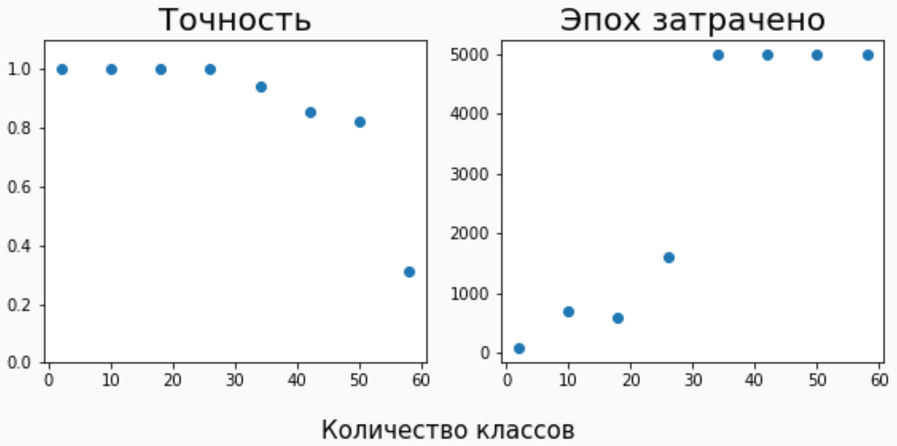
Эксперимент 1 (маска, разбитая на k равных частей)

EfficientNet-B4



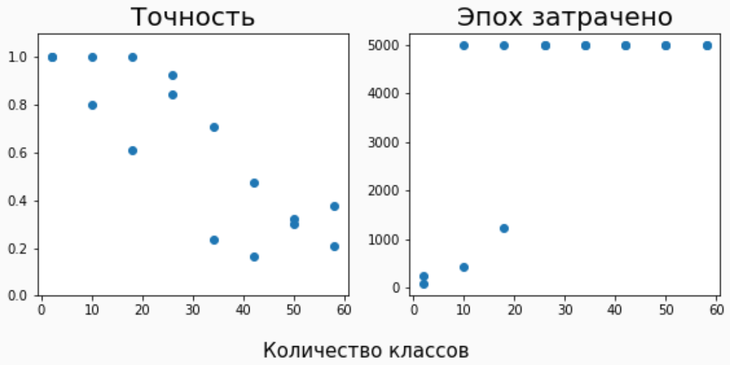


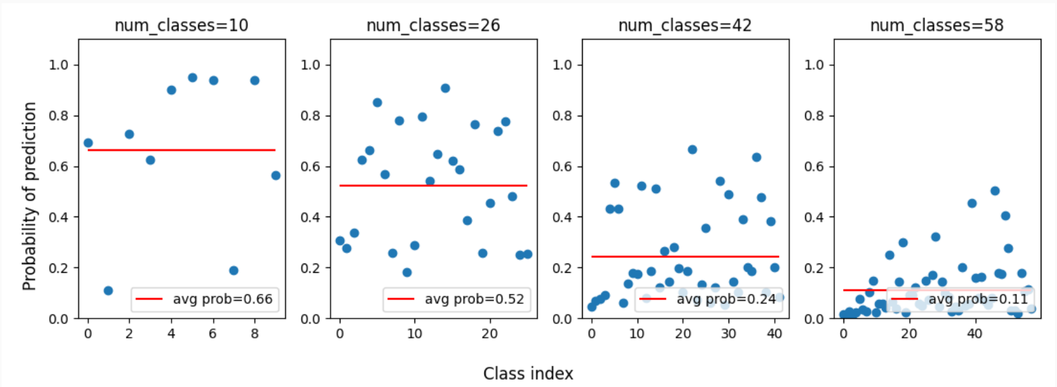
MobileNet-V3-Small



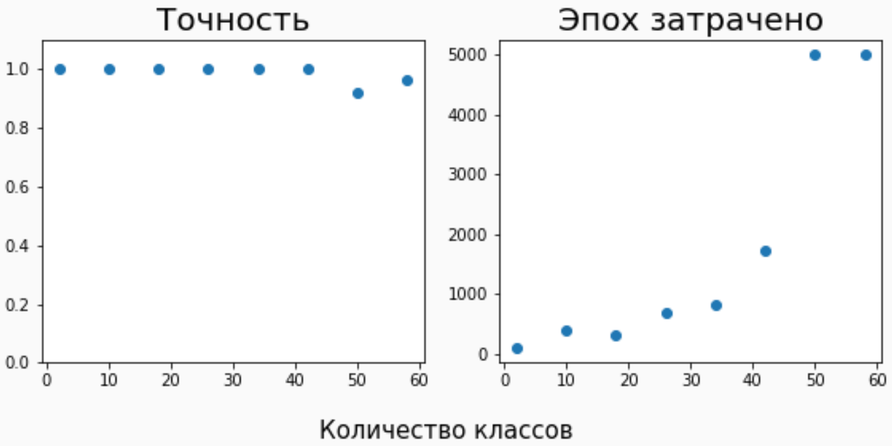
Эксперимент 1 (маска, разбитая на k равных частей)

AlexNet



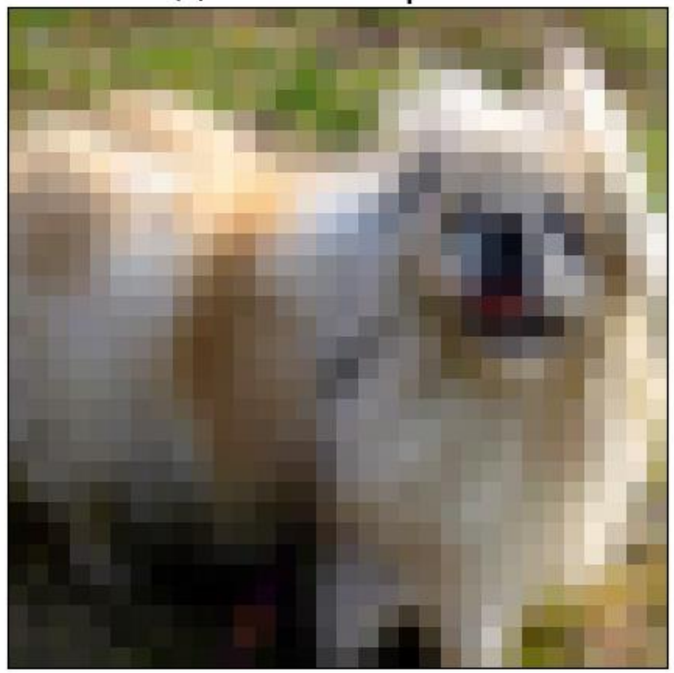


GoogleNet



Эксперимент 2
(маска, разбитая на k равных полос)

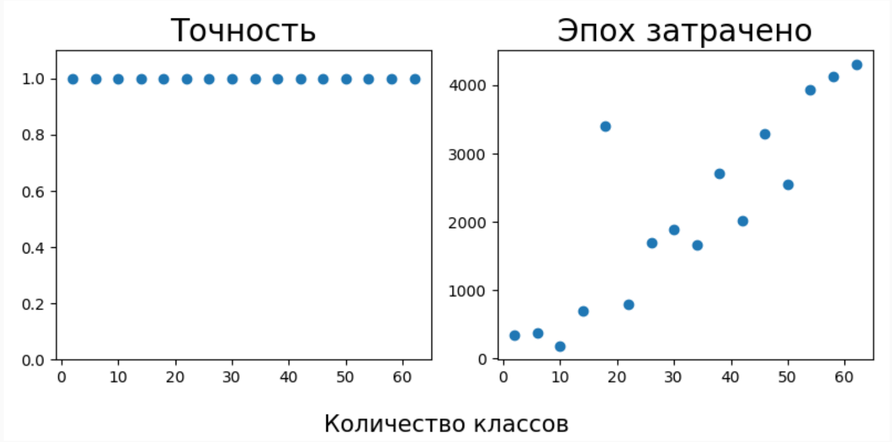
Исходное изображение

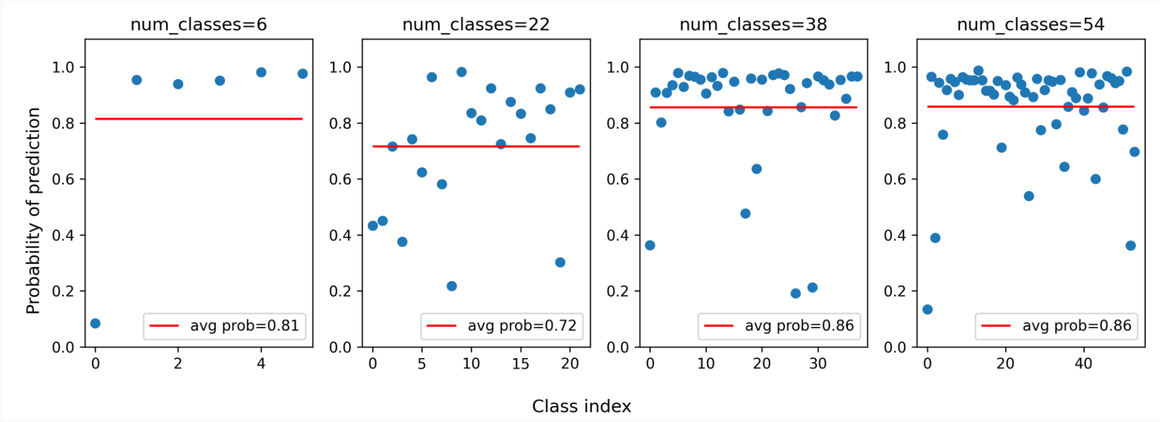


Скрыта часть номер 5

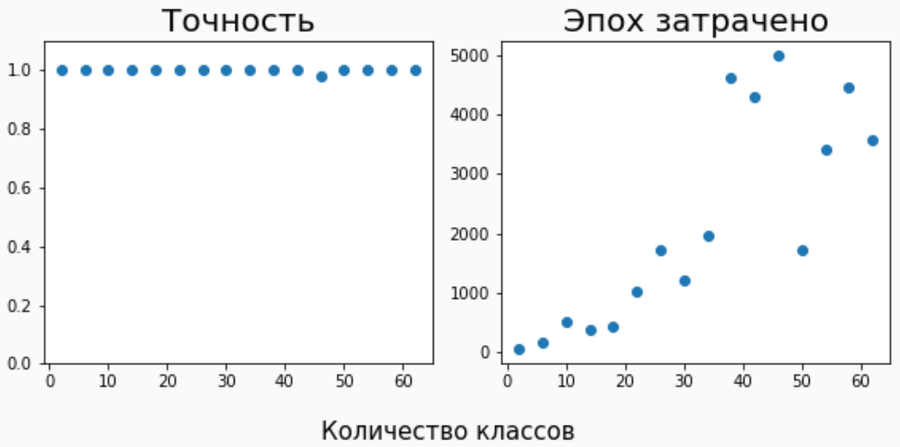
Эксперимент 2 (маска, разбитая на k равных полос)

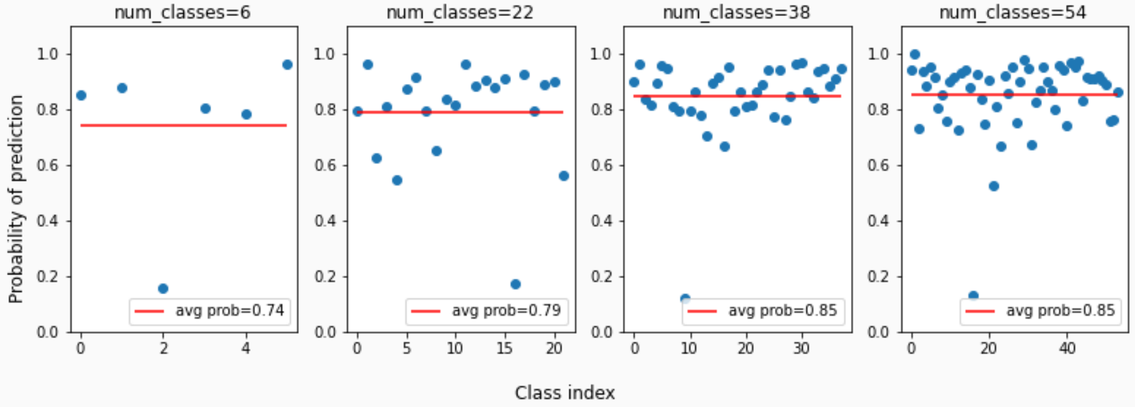
ResNet-50





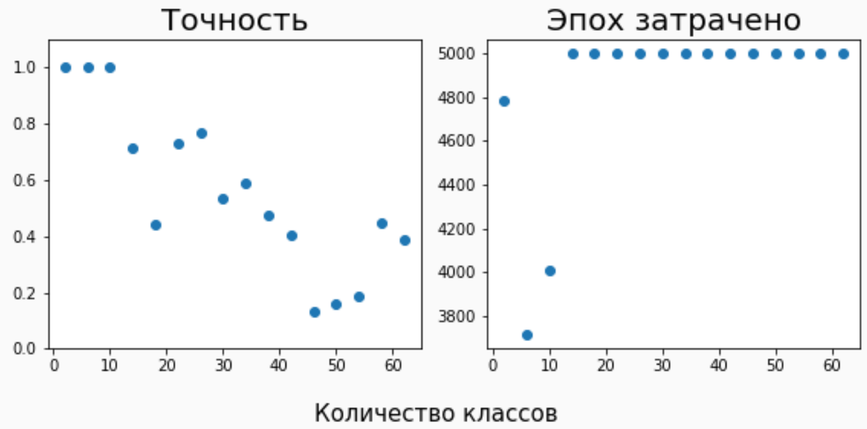
VGG-16

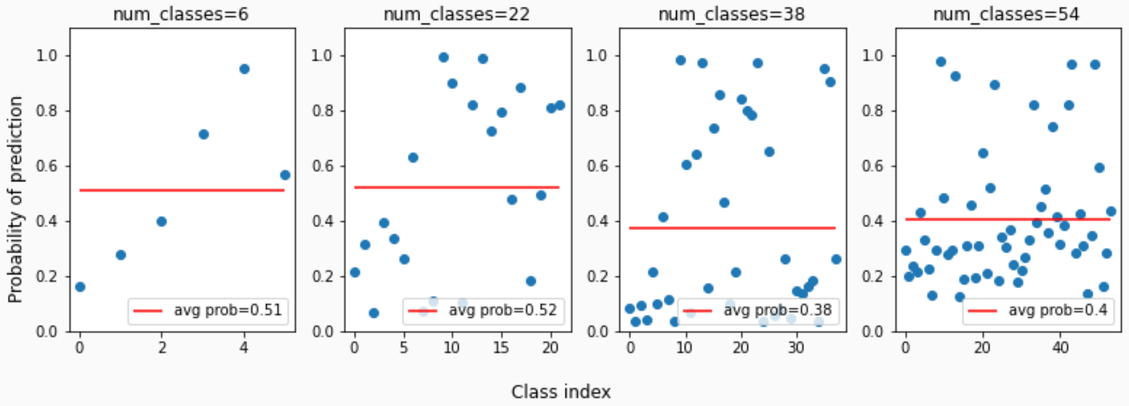




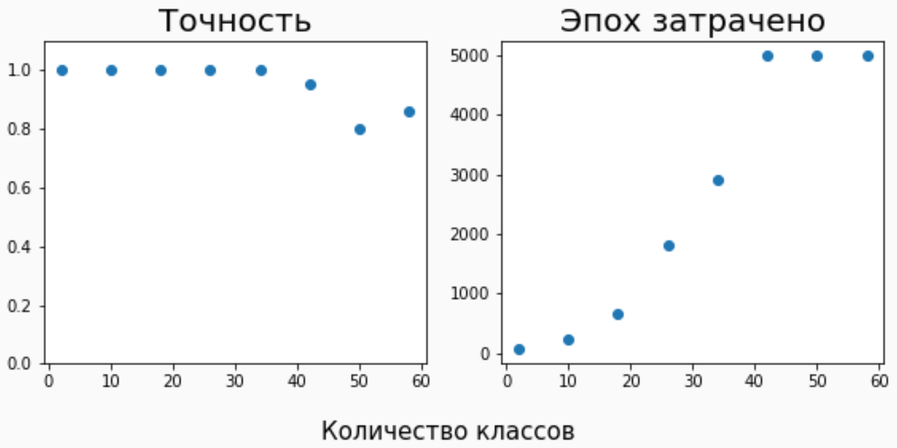
Эксперимент 2 (маска, разбитая на k равных полос)

EfficientNet-B4



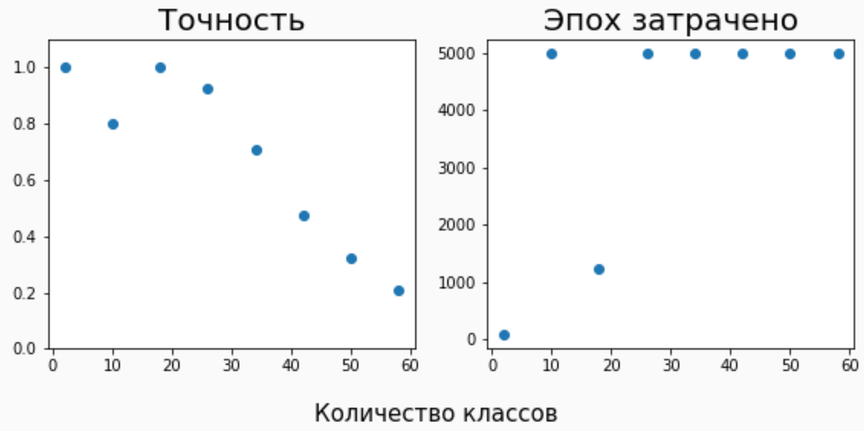


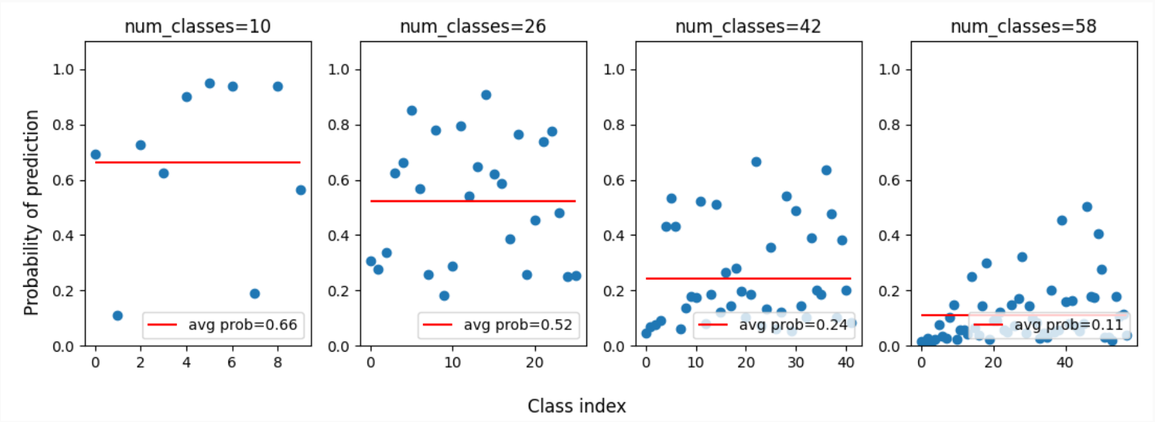
MobileNet-V3-Small



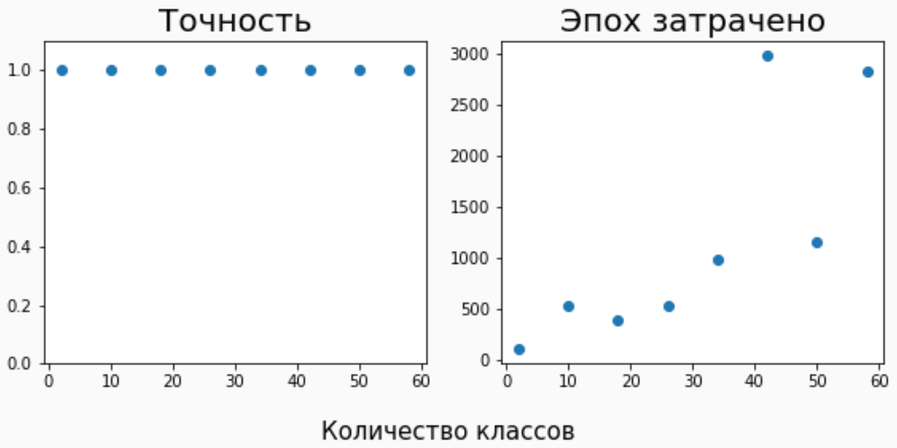
Эксперимент 2 (маска, разбитая на k равных полос)

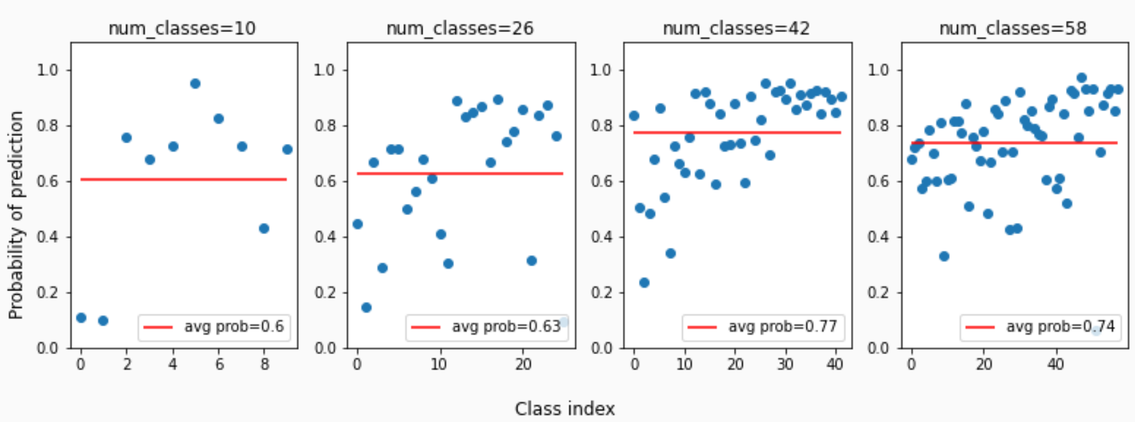
AlexNet





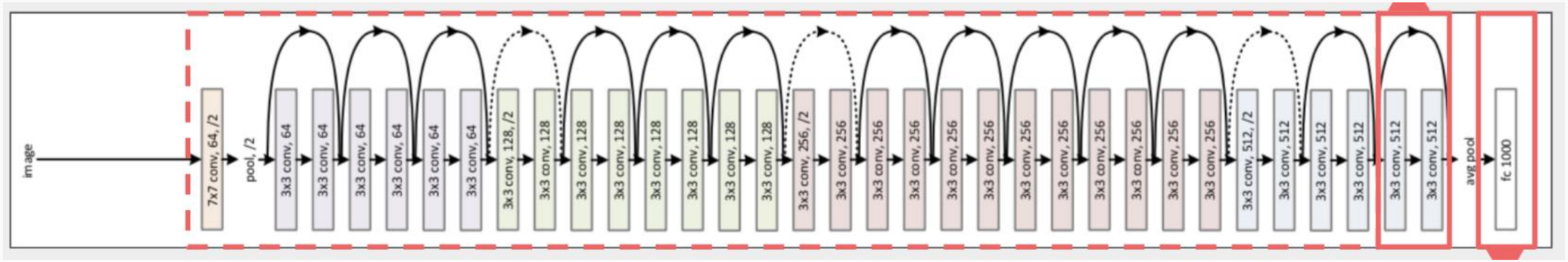
GoogleNet



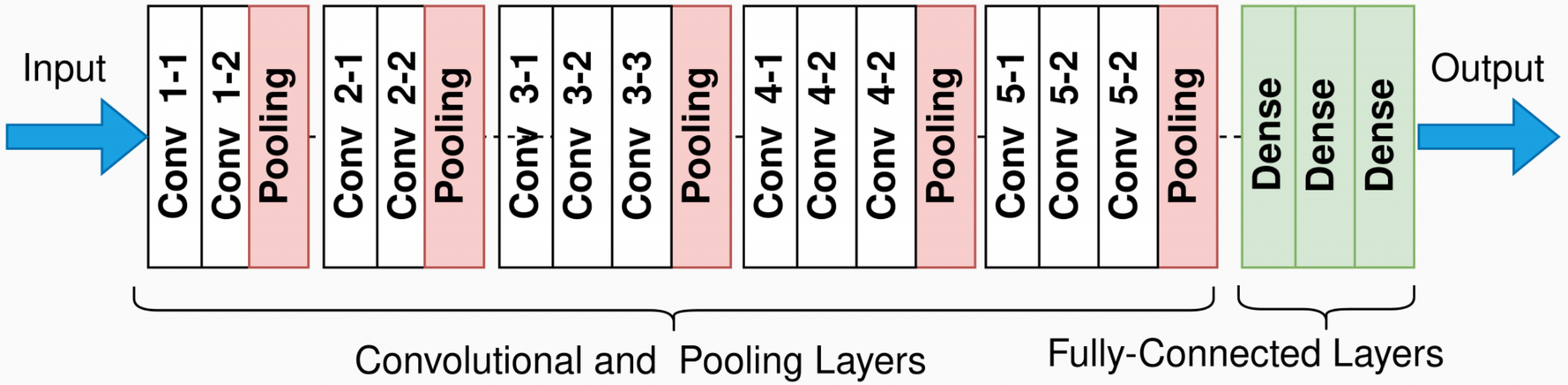


Рассмотрим архитектуры моделей

ResNet-50

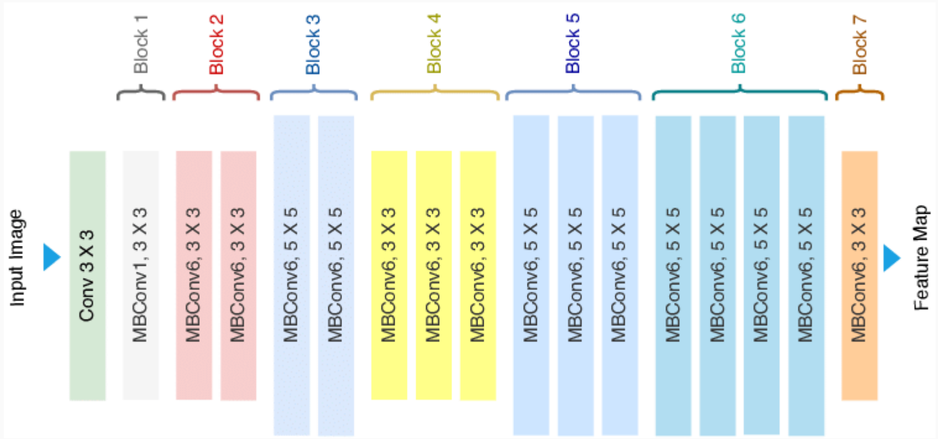


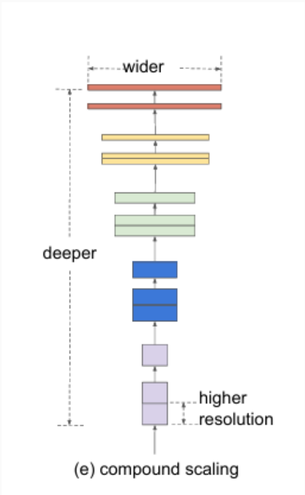
VGG16 Model Architecture

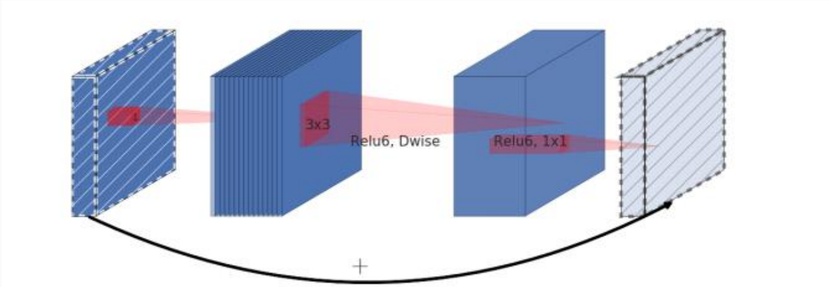


Рассмотрим архитектуры моделей

EfficientNet-B4

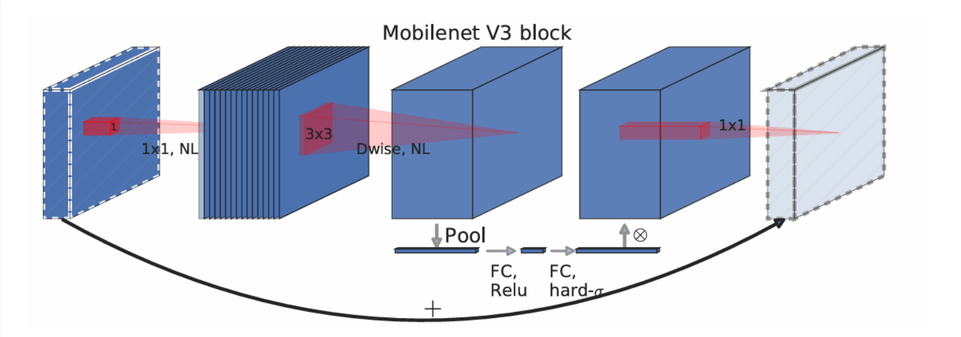






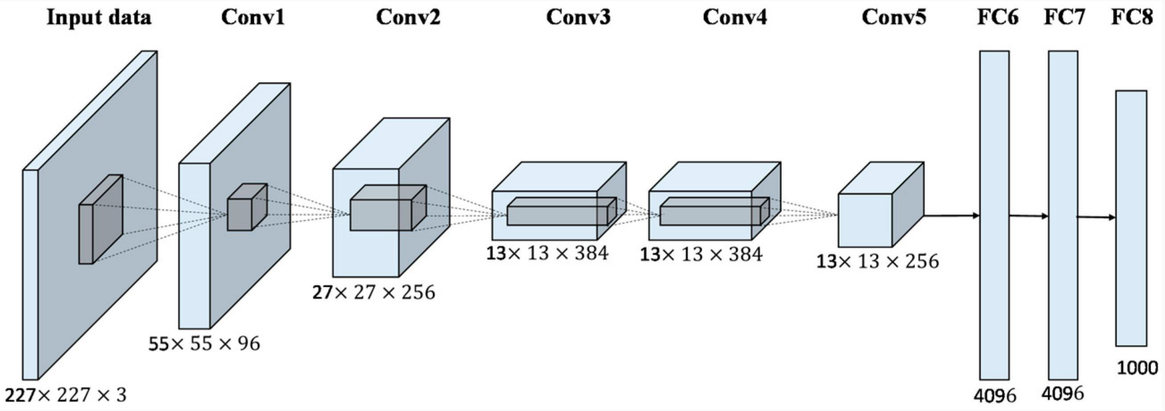
Structure of an inverted residual block (MBconv) [3]

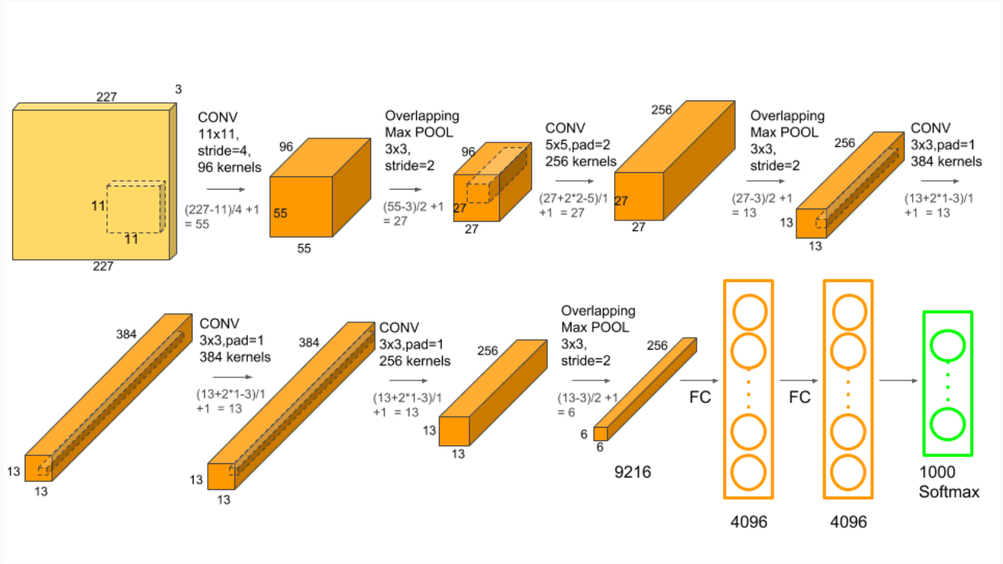
MobileNet-V3-Small



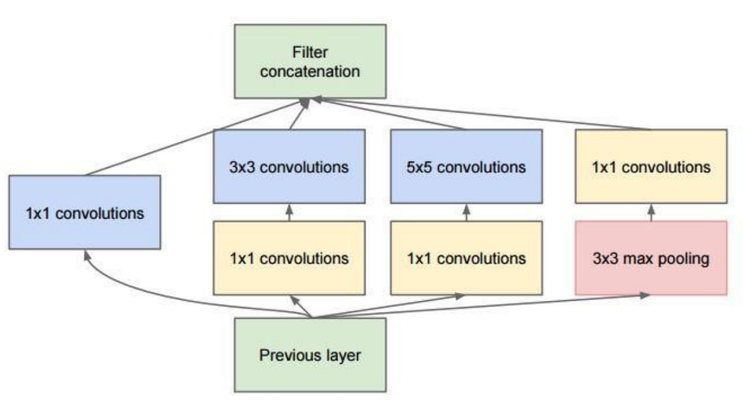
Рассмотрим архитектуры моделей

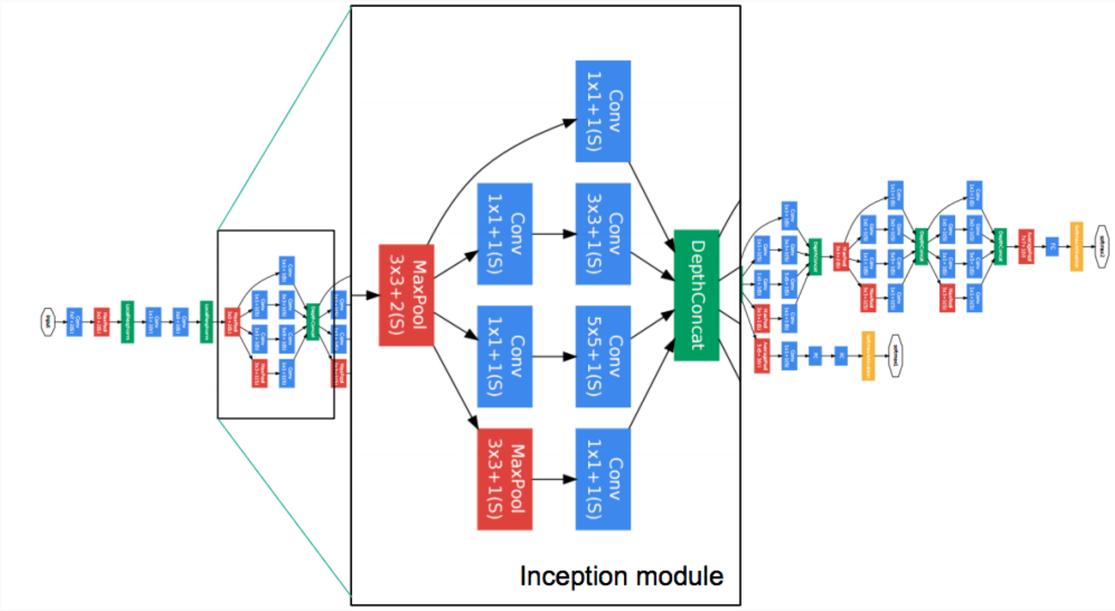
AlexNet





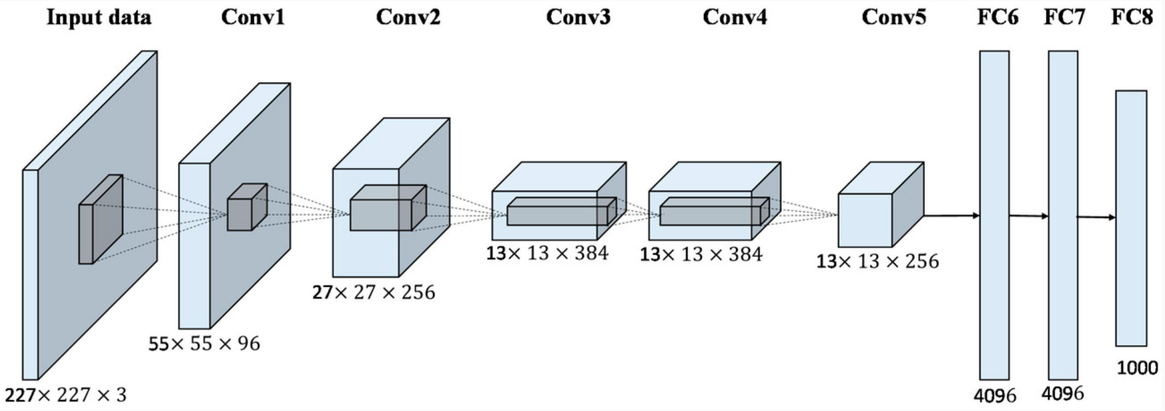
GoogleNet

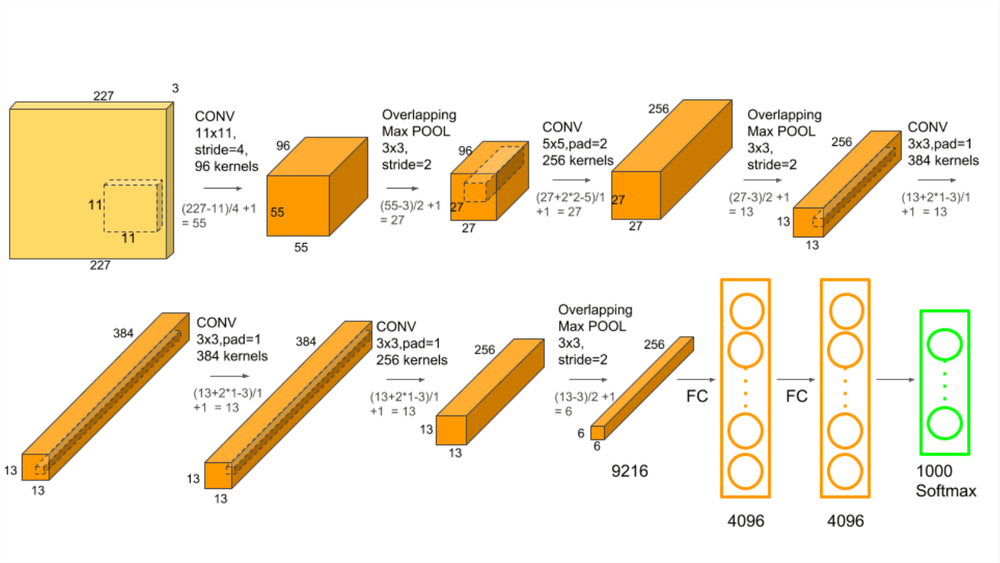




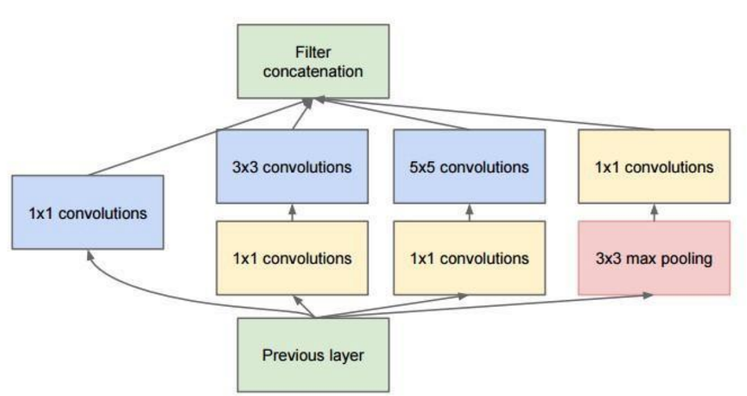
Рассмотрим архитектуры моделей

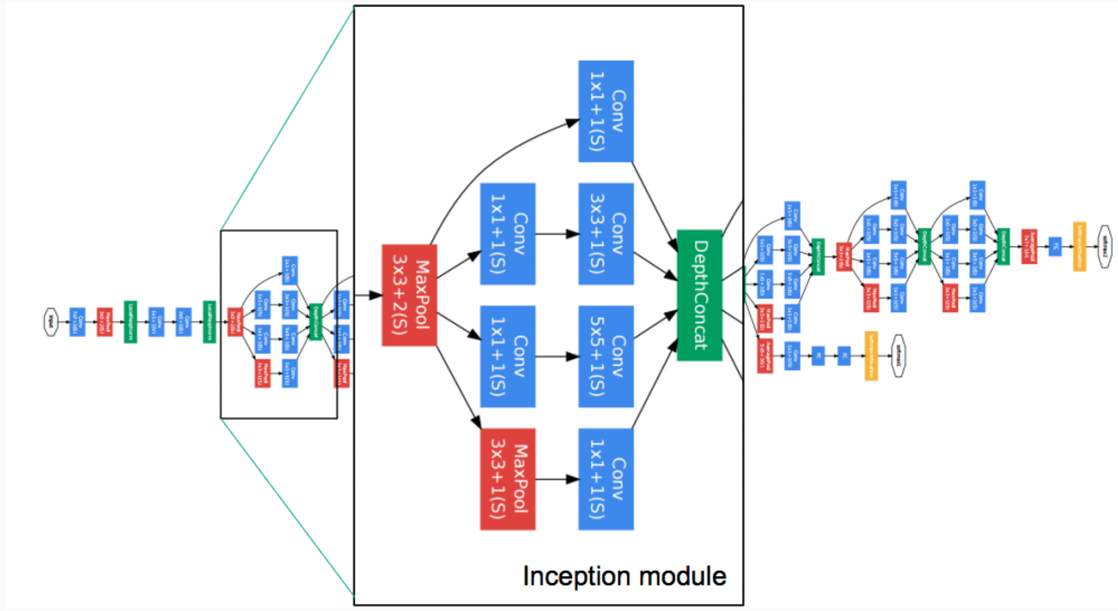
AlexNet





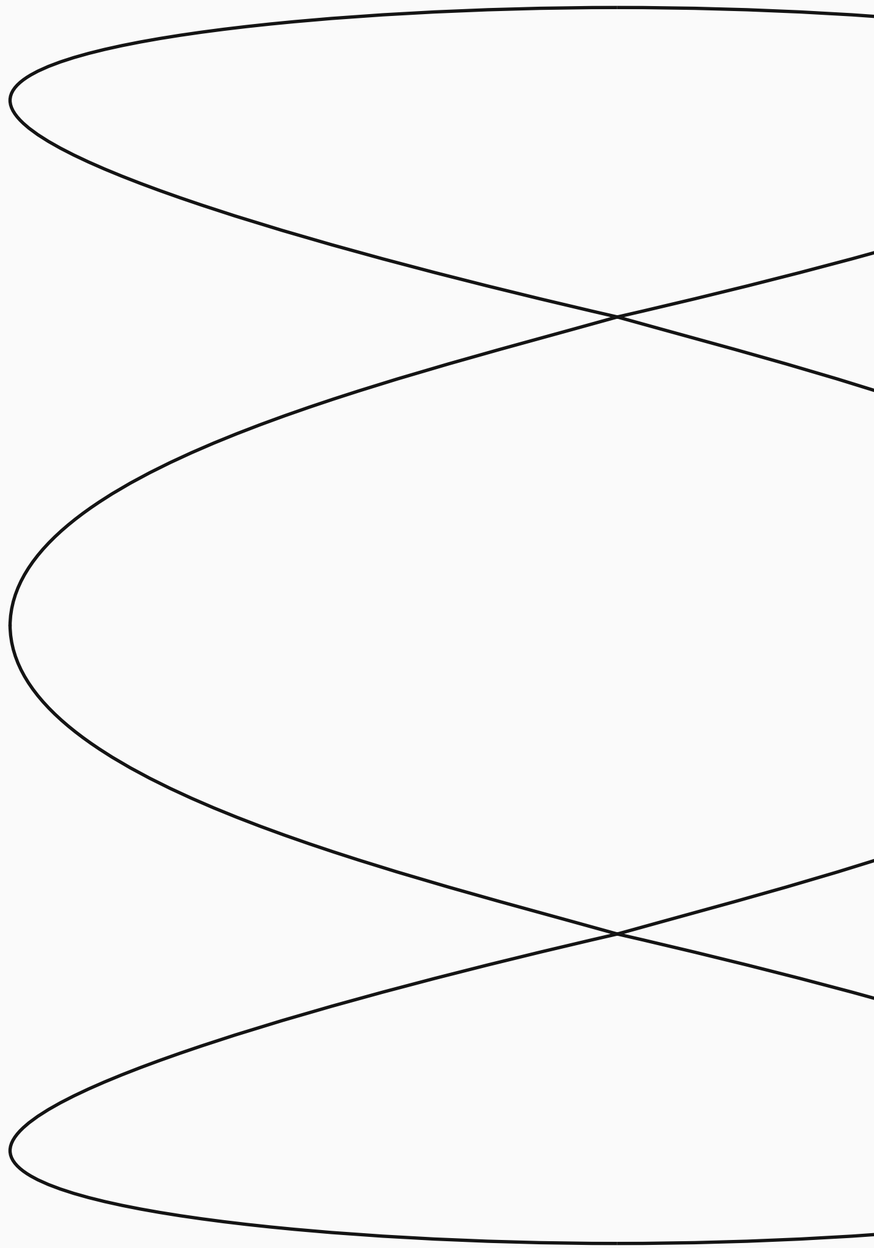
GoogleNet





Какие выводы можно сделать?

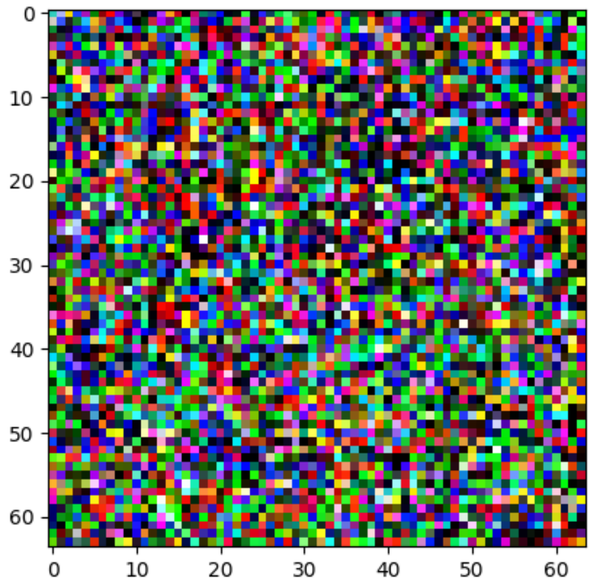
- Модели были использованы в задаче, на которой они не тренировались. Однако EfficientNet показала хороший результат
- Результаты AlexNet обусловлены её плохим качеством на оригинальной задаче классификации изображений
- Видно, что в моделях, в которых используется архитектура только на свертках (без учета глубины) – разделение на классы довольно четкое



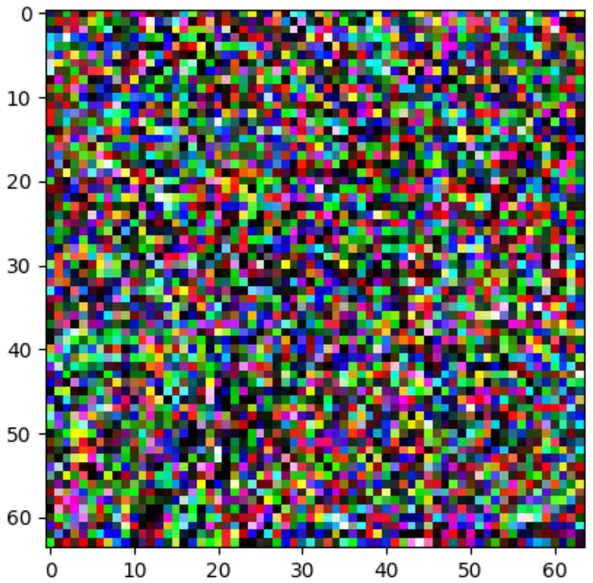
Примеры генерируемого изображения

Изображение не несет какого-либо смысла для человеческого восприятия и имеет вид белого шума.
Изображения получены в экспериментах с количеством классов равным 62.

ResNet-50 (маска из равных квадратов)



ResNet-50 (маска из горизонтальных полос)

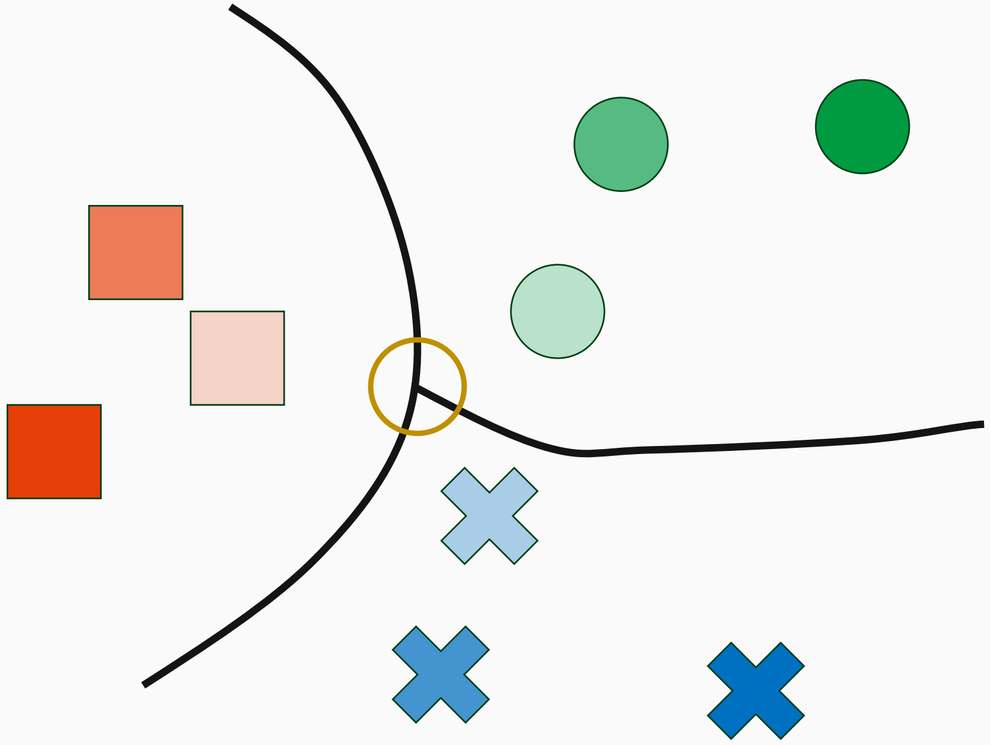


Теоретическая интерпретация

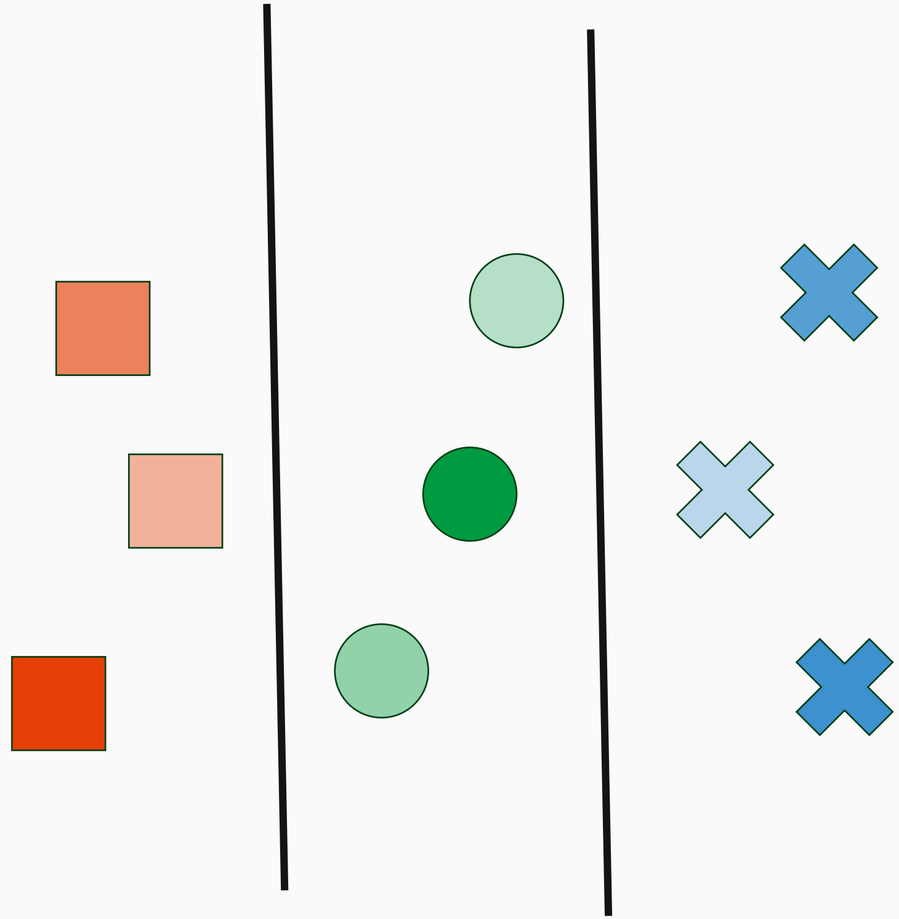
- Представим изображение как объект в линейном пространстве (все множество изображений – гиперкуб со стороной 1)
- При применении масок мы получаем объекты из некоторой окрестности универсального изображения
- Внутри этой окрестности содержатся объекты, которые модель переводит в любой класс из заданного набора

Пример – при применении горизонтальной маски с количеством классов = 64, относительная разность норм для маскированных изображений не более 1.6%. И в этом шаре содержатся объекты всех 64 классов.

Иллюстрация



В данном случае разделяющие гиперплоскости пересекаются в некоторой точке – в её окрестности есть любой класс



Здесь разделяющие гиперплоскости НЕ пересекаются – нет окрестности, в которой есть объекты любого класса

Основные итоги работы

Разработан алгоритм для генерации универсальных изображений, которые при применении различных масок имеют разные классы на выходе модели

Алгоритм был применен на 6 моделей SOTA для задачи Image Classification

Такие исследования могут помочь в теоретических исследованиях работы нейронных сетей (о том, как именно они работают изнутри)

Вопросы