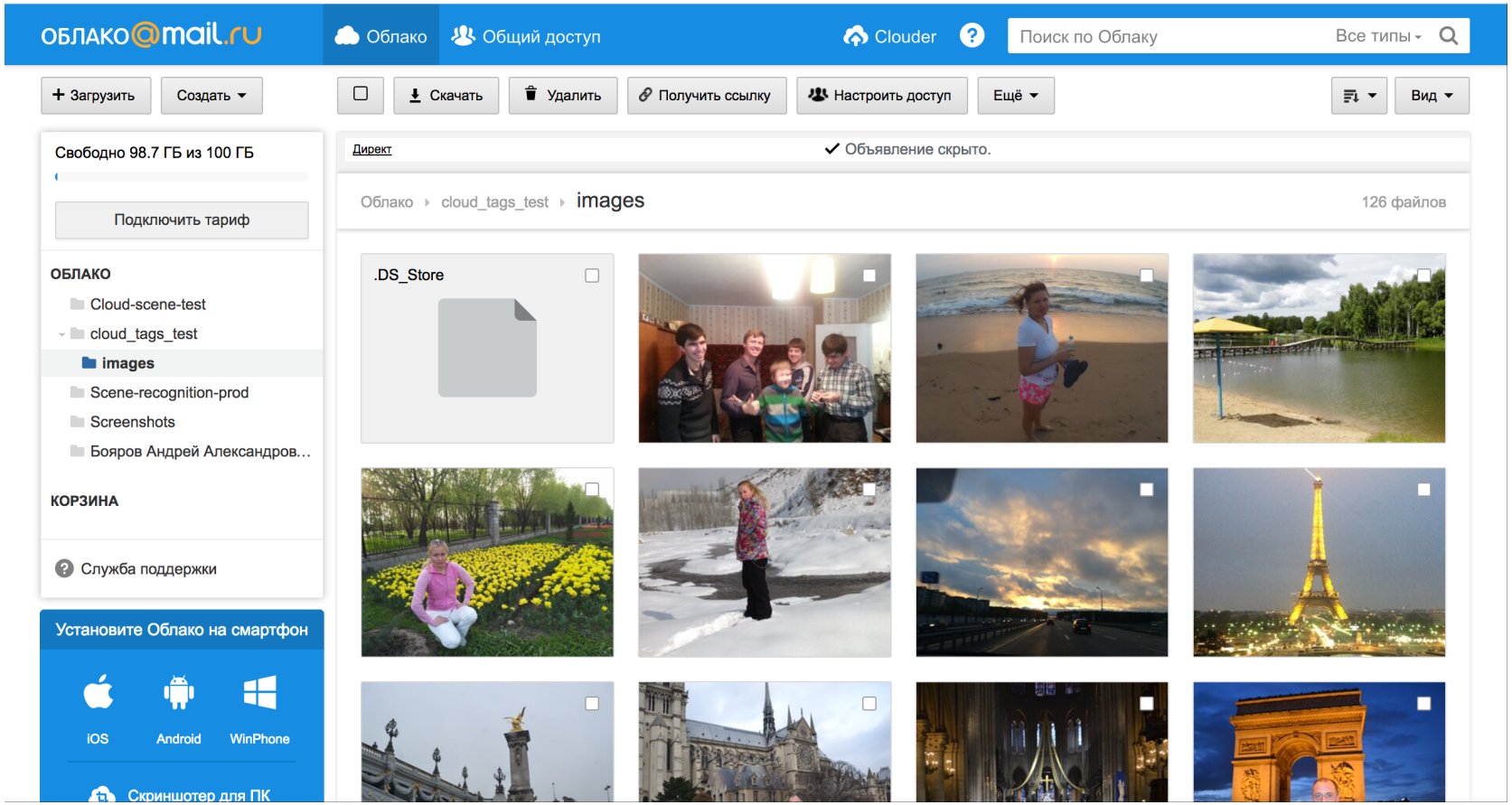
DeepLearning: Распознавание сцен и достопримечательностей на изображениях

Андрей Бояров

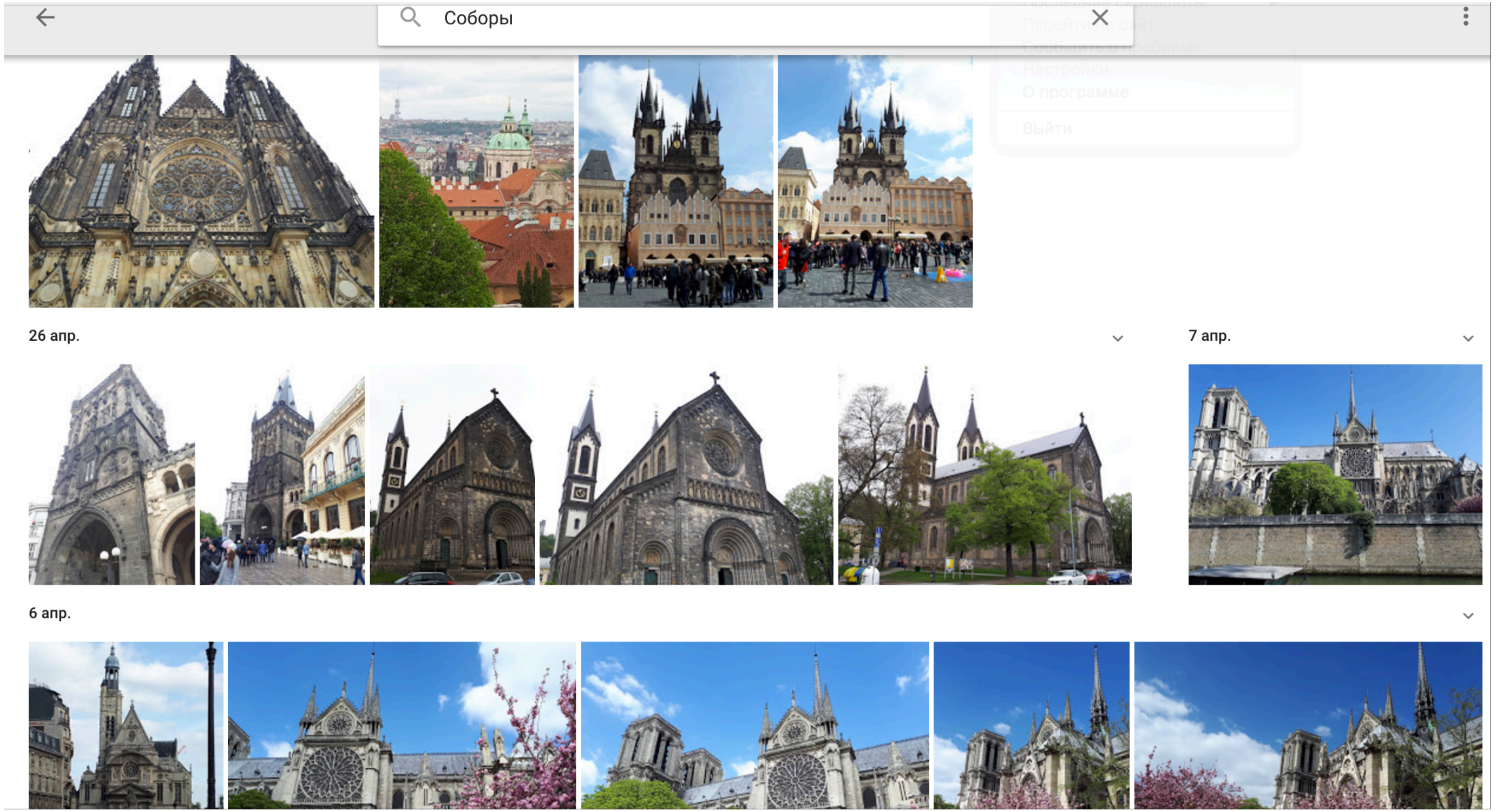
@mail.ru group

Цель

- Создание инструмента для теггирования изображений по сценам
- Часть сервиса Computer Vision API для использования в продуктах компании
  - Пример: поиск фото в Облако@mail.ru



Пример использования



Object recognition vs Scene recognition

Object recognition

Scene recognition

Синонимы: Places recognition, Scene recognition, Scene classification

Доступные сервисы

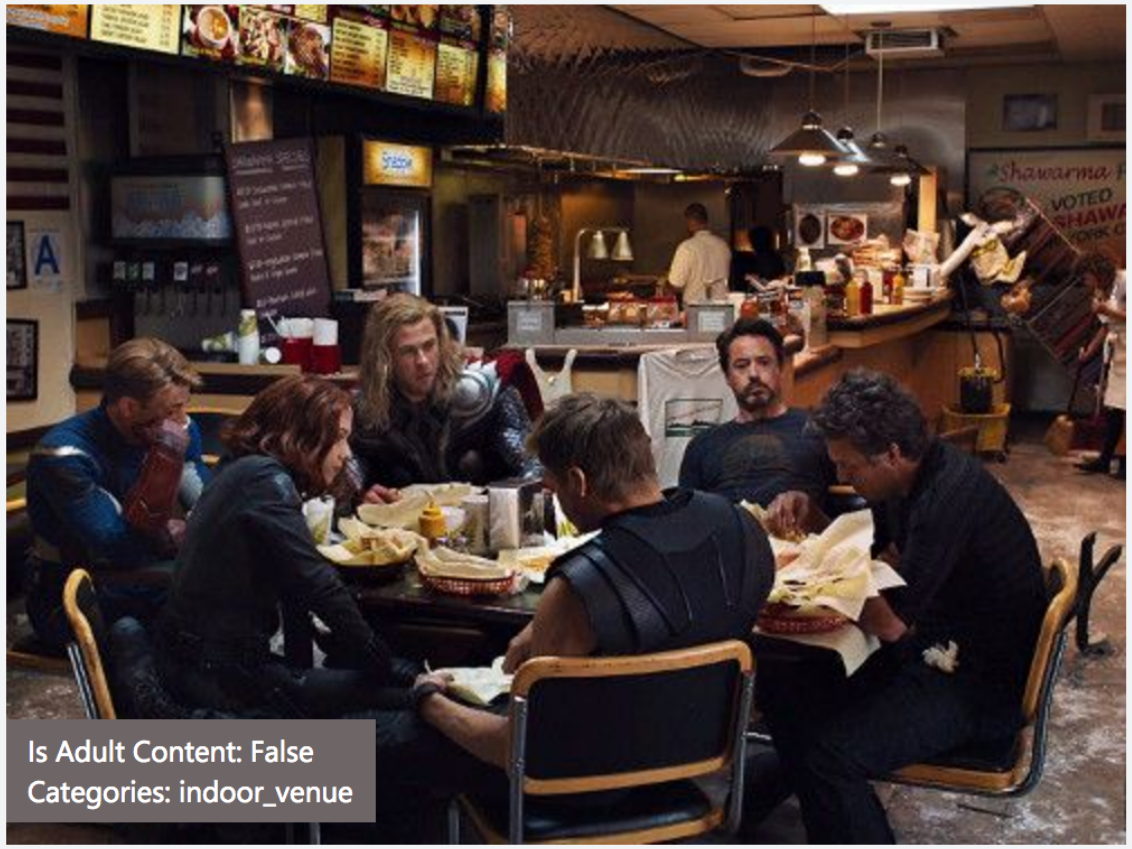
Google Vision API

Labels | Text | Colors | Safe Search | JSON Response

pizzeria_1.jpg

Meal	81%
Lunch	70%
Restaurant	68%

Microsoft Computer Vision API



https://portalstoragewuprod2.azureedge.net/v

Feature Name	Value
Description	{ "type": 0, "captions": [{ "text": "a group of people sitting at a table in a restaurant", "confidence": 0.9718332823686703 }] }
Tags	[{ "name": "person", "confidence": 0.9991776347160339 }, { "name": "table", "confidence": 0.9836546778678894 }, { "name": "sitting", "confidence": 0.9808034896850586 }, { "name": "indoor", "confidence": 0.9717702865600586 }, { "name": "people", "confidence": 0.9362378716468811 }, { "name": "group", "confidence": 0.8897872567176819 }, { "name": "restaurant", "confidence": 0.8656893968582153 }, { "name": "ceiling", "confidence": 0.8505259156227112 }, { "name": "scene", "confidence": 0.7870252132415771 }, { "name": "meal", "confidence": 0.3727511465549469 }, { "name": "dinner", "confidence": 0.36563196778297424 }, { "name": "seated", "confidence": 0.30345749855041504 }, { "name": "several", "confidence": 0.11127004027366638 }]
Image Format	Jpeg
Image Dimensions	500 x 375
Clip Art Type	0 Non-clipart

Данные

Существующие базы

	SUN (2012)	Places2-Standard (2016)	Places2-Challenge (2016)
Volume	108 754	1 803 486	8 026 628
# classes	397	365	365
Per class	> 100	3068 – 5000	3068 – 40000

- ImageNet 2016 Scene classification (Places2-Challenge): 9.01% top-5 error

Примеры Places2

Canyon

Примеры Places2

Runway

Примеры Places2

Kitchen

Примеры Places2

Soccer field

Адаптация данных для обучения: маленькая база

- CIFAR-10, CIFAR-100 вместо ImageNet
- SUN Reduce
  - 89 классов
  - Train: 50 000
  - Val: 10 000
- Обучение занимает 6-10 часов

Адаптация данных для обучения: Places2

- Некоторые классы не нужны, т.к.:
  - В них нет необходимости в production
  - Мало данных
  - Примеры: aqueduct, tree house, barndoor
- Получаем Places “Sift”
  - 314 классов
  - Standard train: около 1 500 000 изображений
  - Challenge train: около 7 500 000 изображений

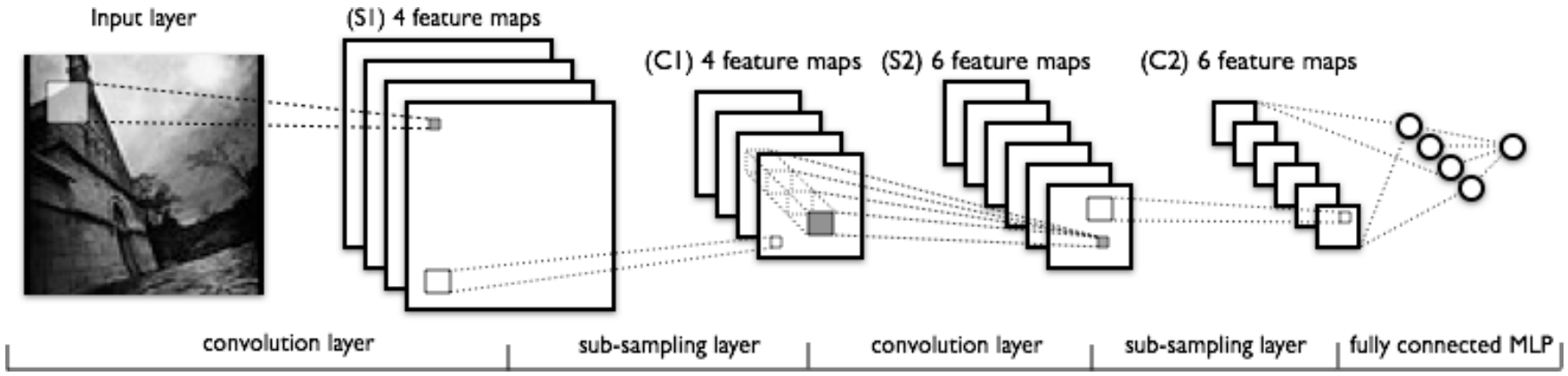
Адаптация данных для обучения: Scene mapping

- Слишком много классов для production
- Scene mapping
  - bamboo forest, forest broadleaf, forest path, forest road, rainforest -> forest
  - hospital, hospital room, nursery, operating room -> hospital
  - hotel outdoor, hotel room, inn outdoor, youth hostel -> hotel
- Используется только для тестирования и для конечного пользователя

Подходы, решения

Классические подходы

- Deep convolutional neural networks ☺
  - Top ImageNet и Places2
  - Семейство Inception
  - Семейство Residual Networks



Residual Networks

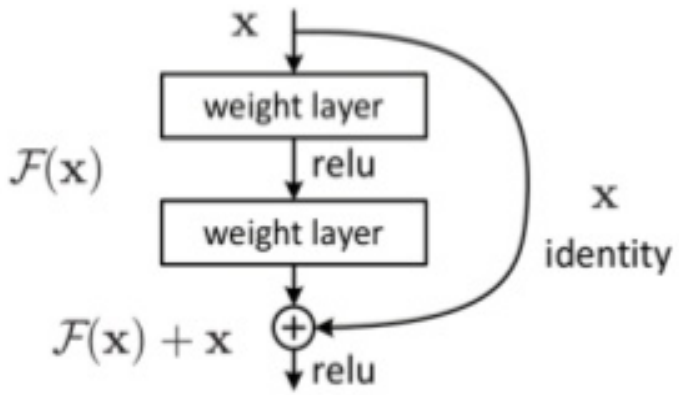
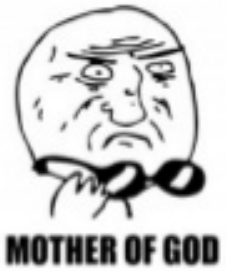
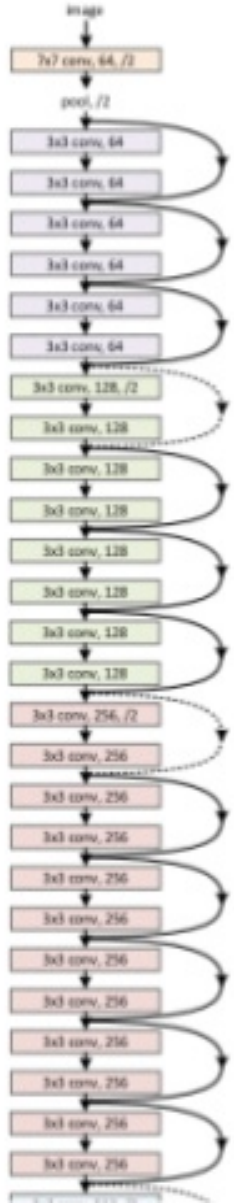


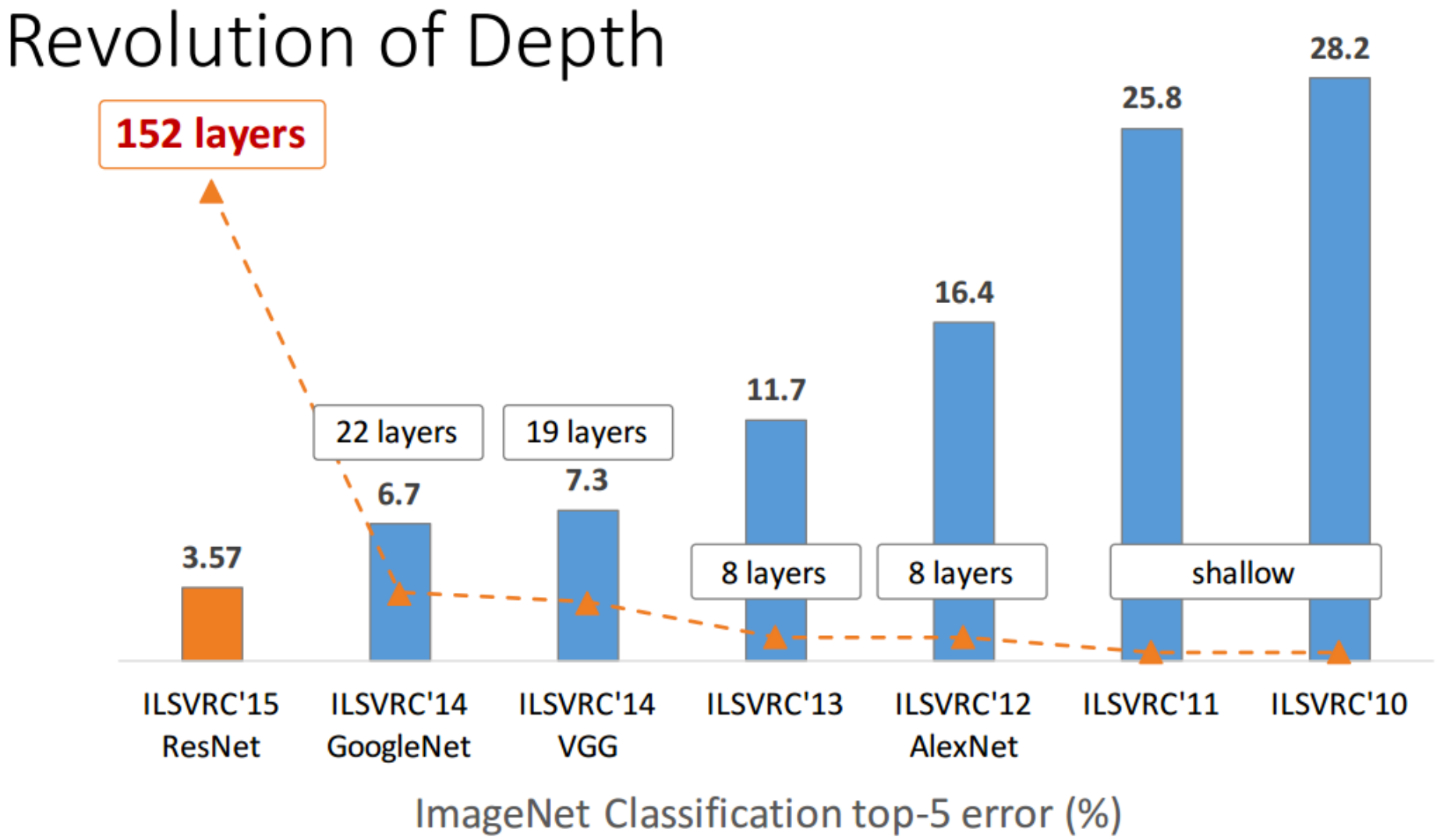
Figure 2. Residual learning: a building block.



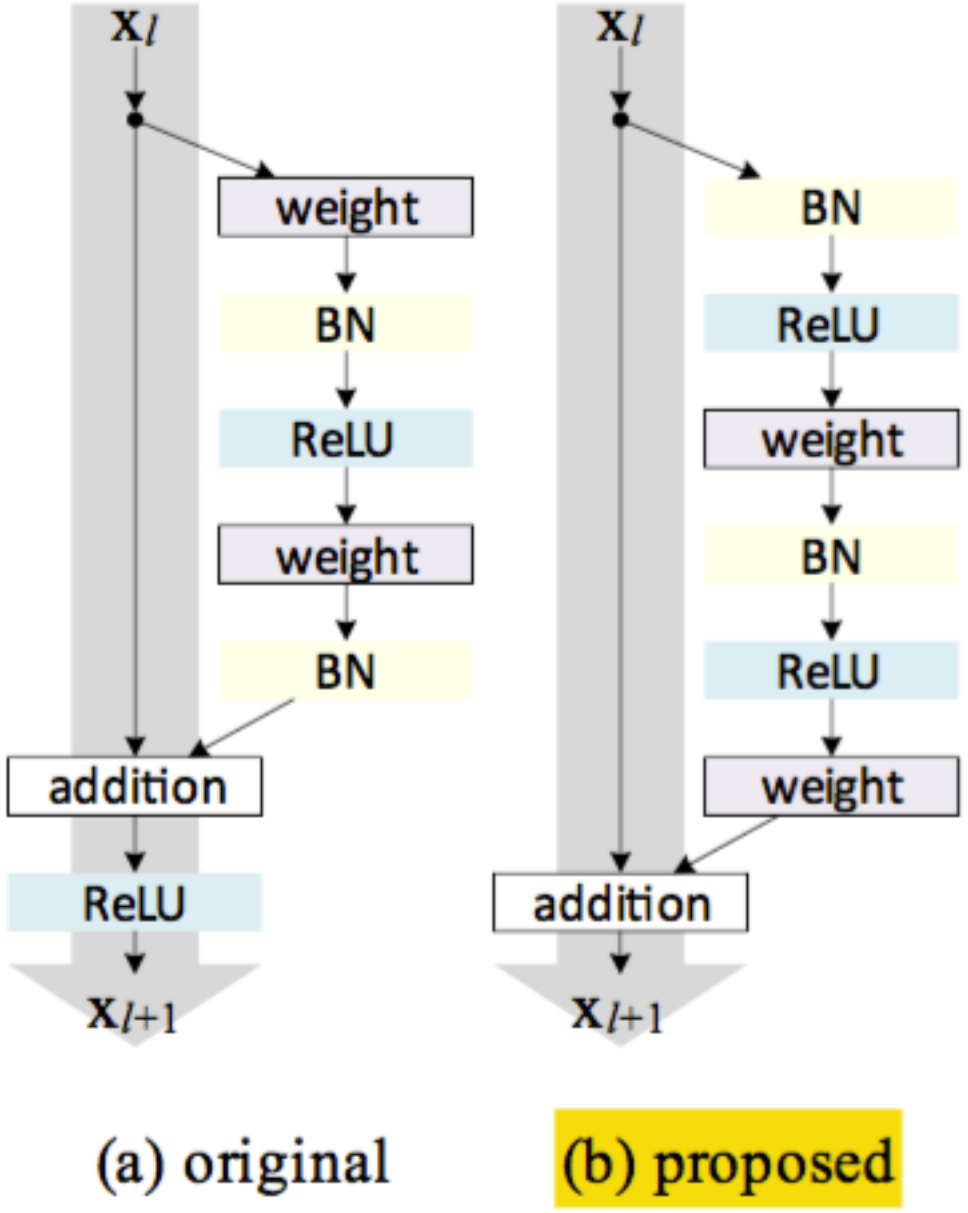
Network can decide how deep it needs to be…



Residual Networks



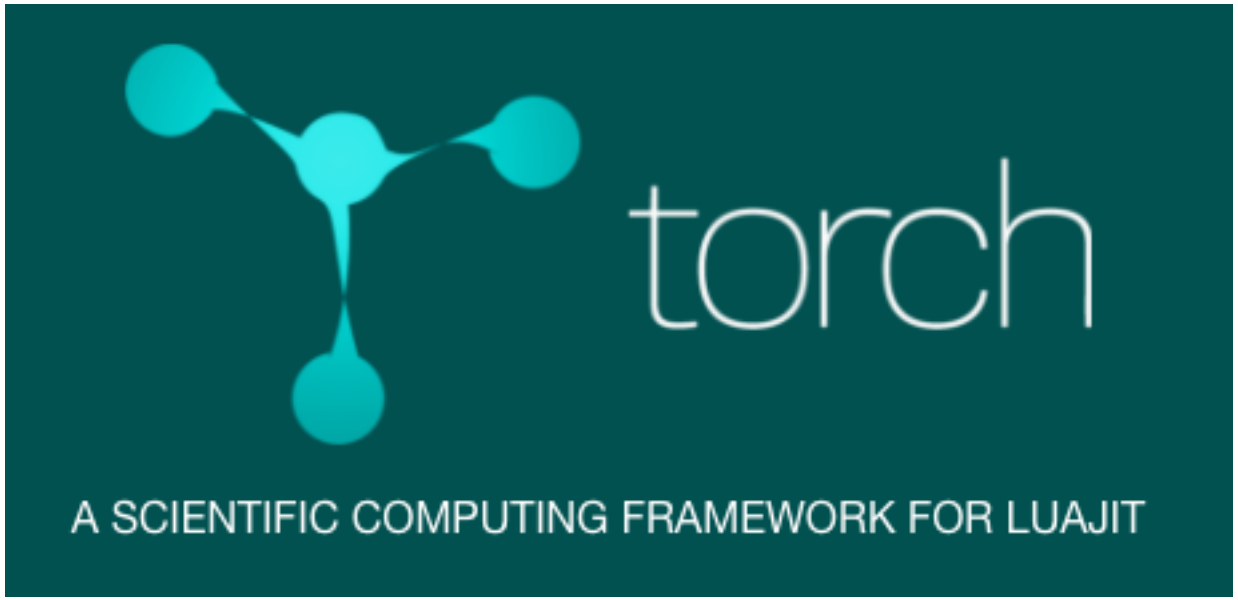
Residual block



(a) original (b) proposed

Обучение ResNet

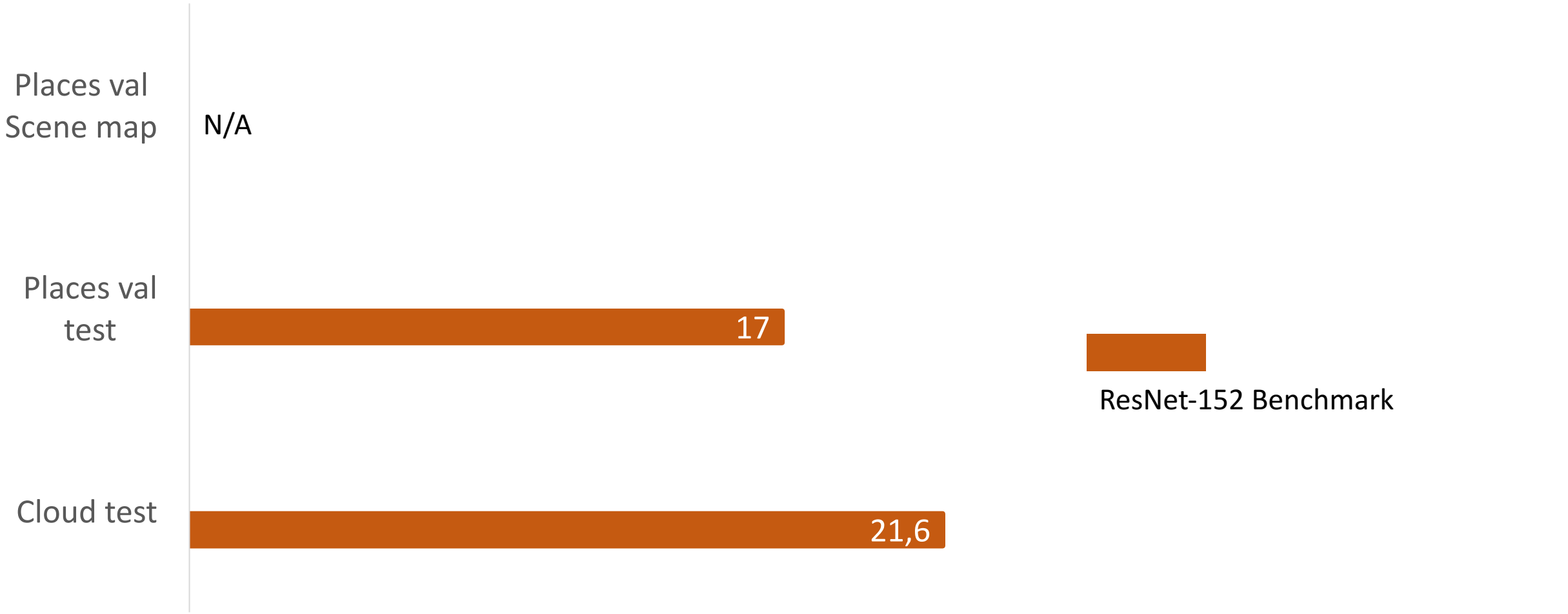
- Для обучения CNN используется framework Torch
- Форк с реализации ResNet от Facebook (de facto SOTA):
  - https://github.com/facebook/fb.resnet.torch



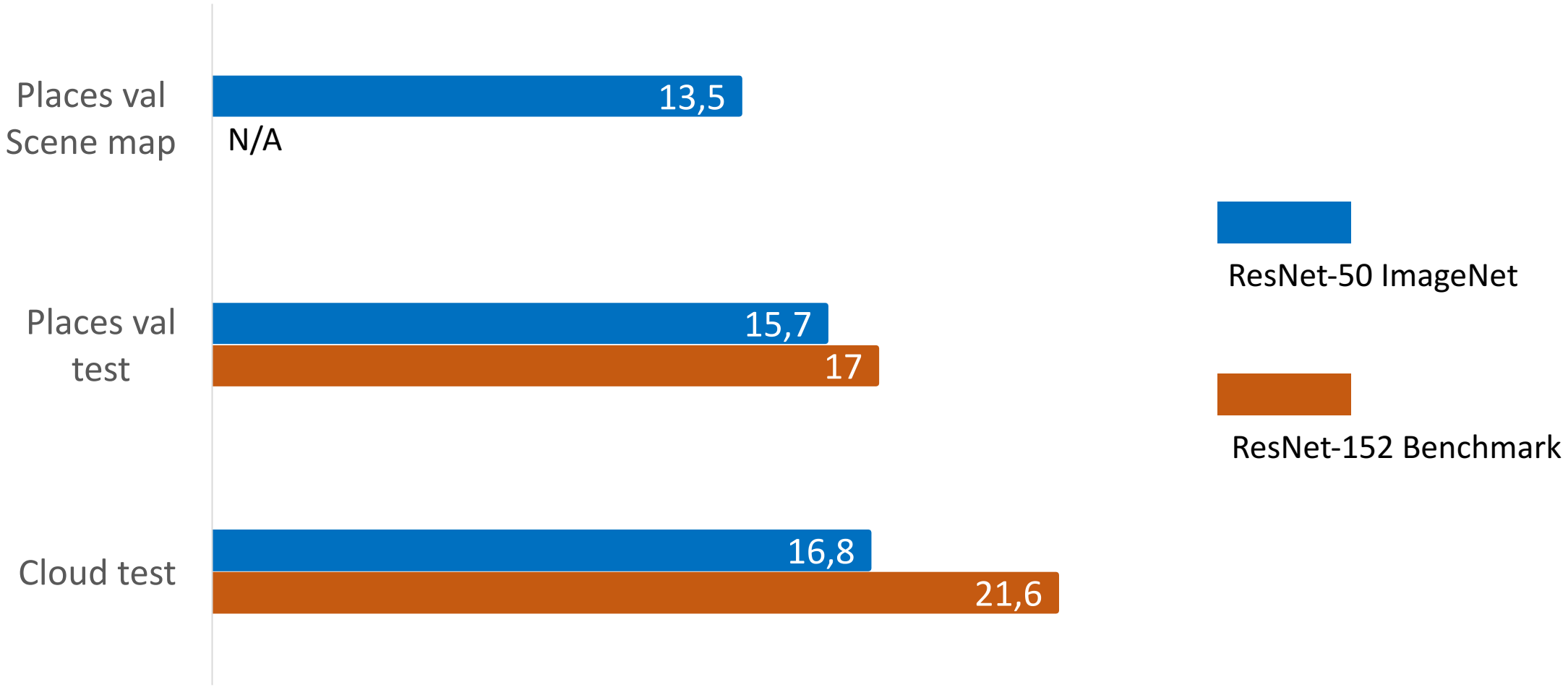
Тесты качества работы

- Places val – validation set Places “Sift”
- Places val Scene map – validation set Places “Sift” + Scene mapping
- Cloud test: изображения сотрудников из облака, размеченные вручную

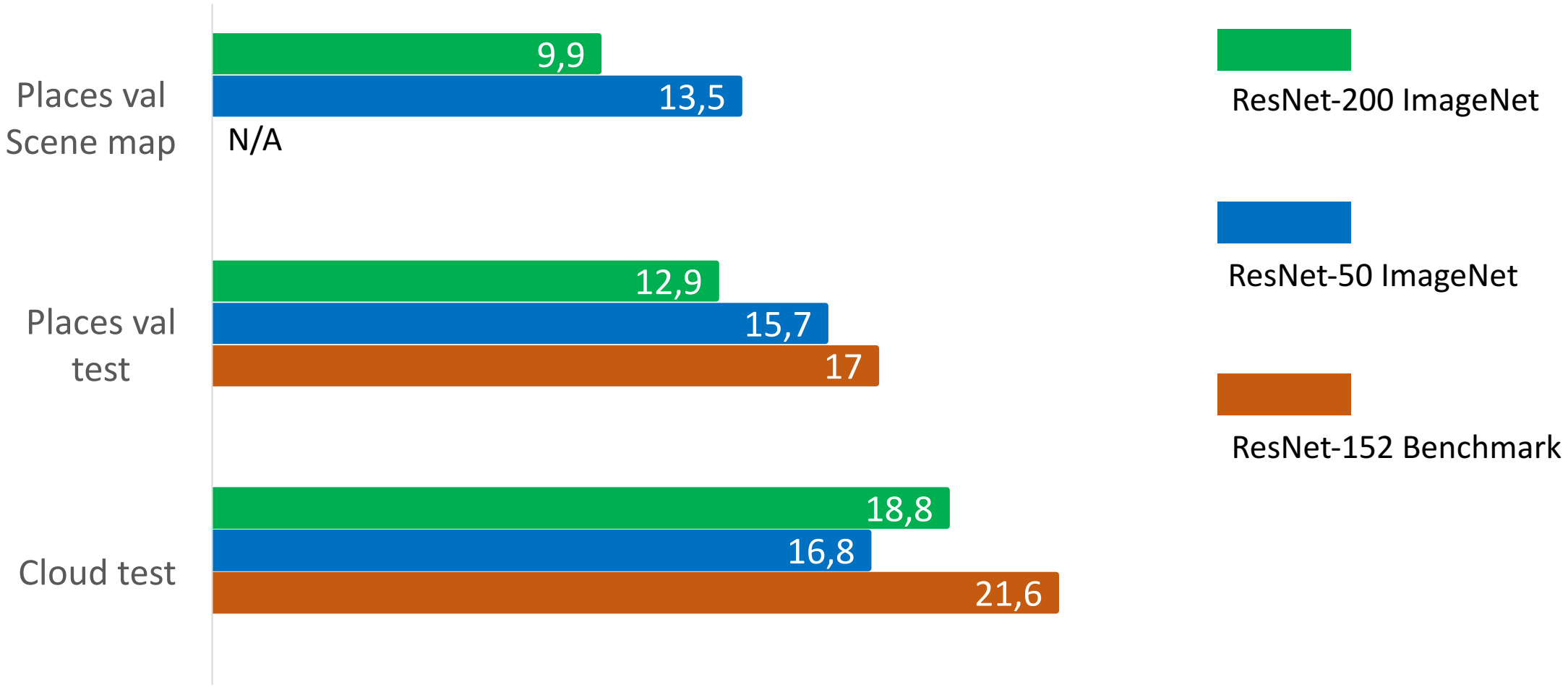
Результаты ResNet (top 5 error, %)



Результаты ResNet (top 5 error, %)



Результаты ResNet (top 5 error, %)



ResNet-152 Benchmark

Predicted:

- museum/indoor 0.026
- coffee_shop 0.024
- clothing_store 0.019
- pet_shop 0.017
- bakery/shop 0.015

Mapped labels:

- shop_none_food 0.036
- museum 0.026
- restaurant 0.024
- shop_food 0.015

ResNet-200

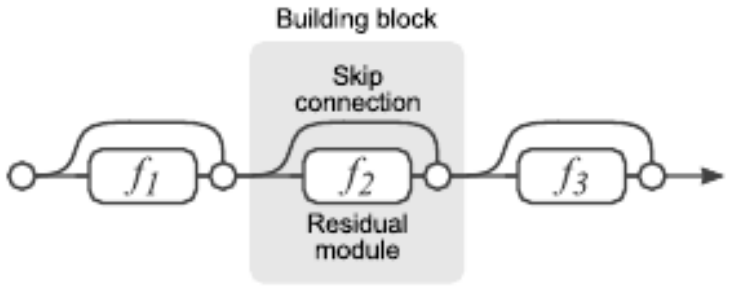
Predicted:

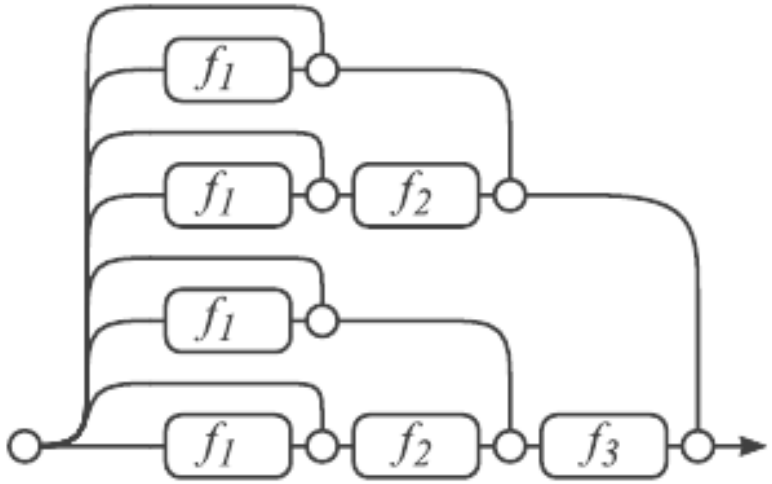
- cafeteria 0.18
- sushi_bar 0.18
- restaurant 0.17
- food_court 0.09
- pub/indoor 0.07

Mapped labels:

- restaurant 0.7

Улучшение ResNet





- Обучать очень большие ResNet всё сложнее с ростом их глубины
- Уменьшим глубину, но увеличим ширину (т.е. количество фильтров) Residual блока

Wide Residual Network

- WRN-50-2
  - ResNet-50 с увеличенным в 2 раза количеством фильтров во внутренней bottleneck 3x3 свёртке
  - ImageNet top-1 err:
    - WRN-50-2: 21.9 %
    - ResNet-200: 21.7 %
  - WRN-50-2 почти в 2 раза быстрее, чем ResNet-200

Wide Residual block

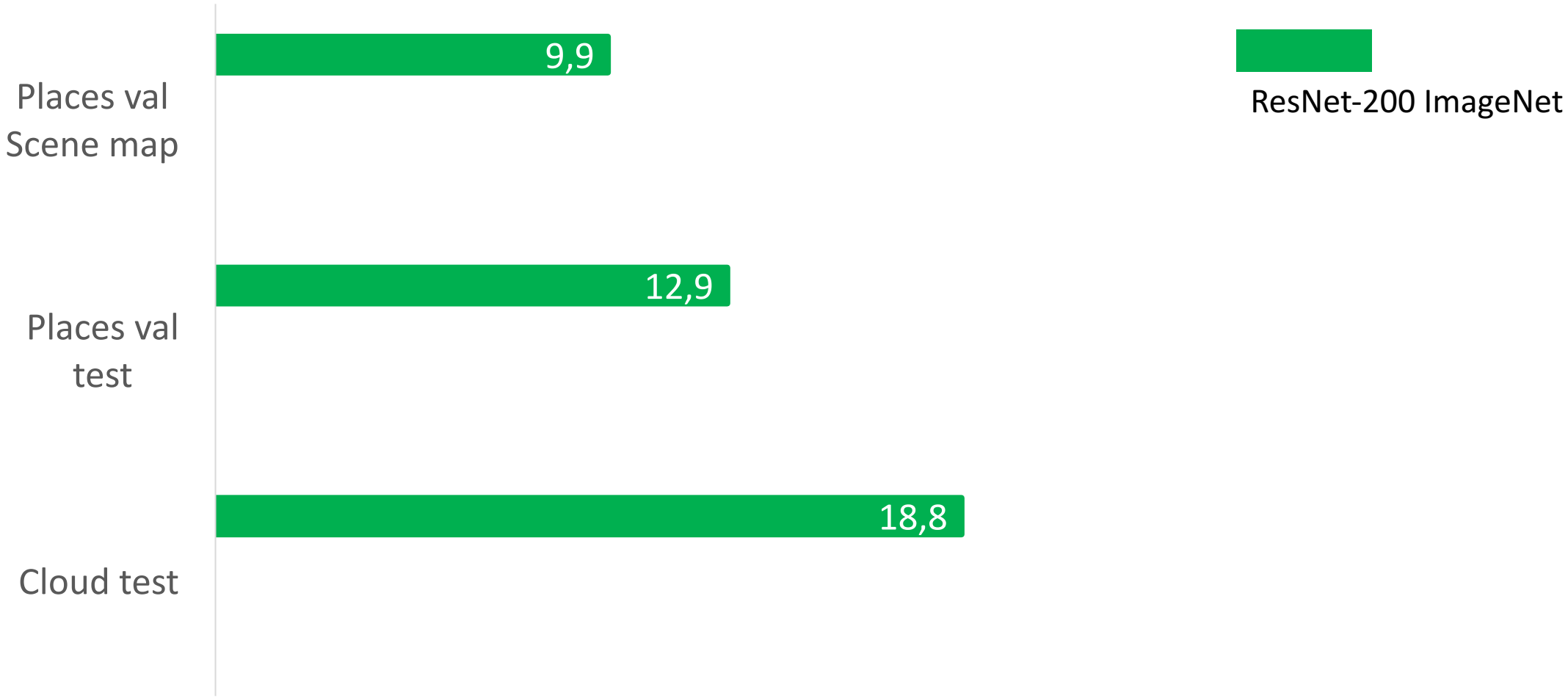
ResNet Residual block

```
(1): nn.SpatialConvolution(256 -> 64, 1x1)
(2): nn.SpatialBatchNormalization (4D) (64)
(3): nn.ReLU
(4): nn.SpatialConvolution(64 -> 64, 3x3, 1,1, 1,1)
(5): nn.SpatialBatchNormalization (4D) (64)
(6): nn.ReLU
(7): nn.SpatialConvolution(64 -> 256, 1x1)
```

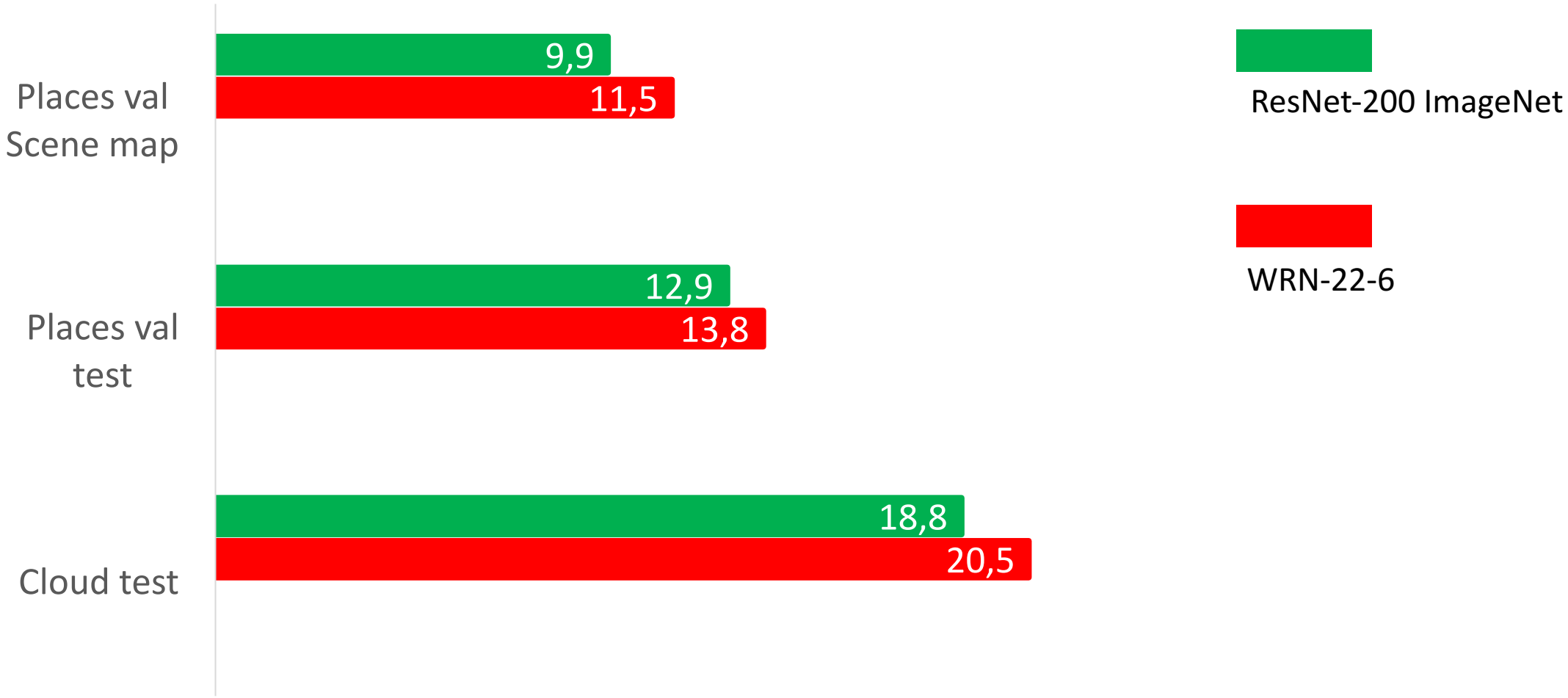
WRN Residual block

```
(1): cudnn.SpatialConvolution(256 -> 128, 1x1)
(2): nn.SpatialBatchNormalization (4D) (128)
(3): cudnn.ReLU
(4): cudnn.SpatialConvolution(128 -> 128, 3x3, 1,1, 1,1)
(5): nn.SpatialBatchNormalization (4D) (128)
(6): cudnn.ReLU
(7): cudnn.SpatialConvolution(128 -> 256, 1x1)
(8): nn.SpatialBatchNormalization (4D) (256)
```

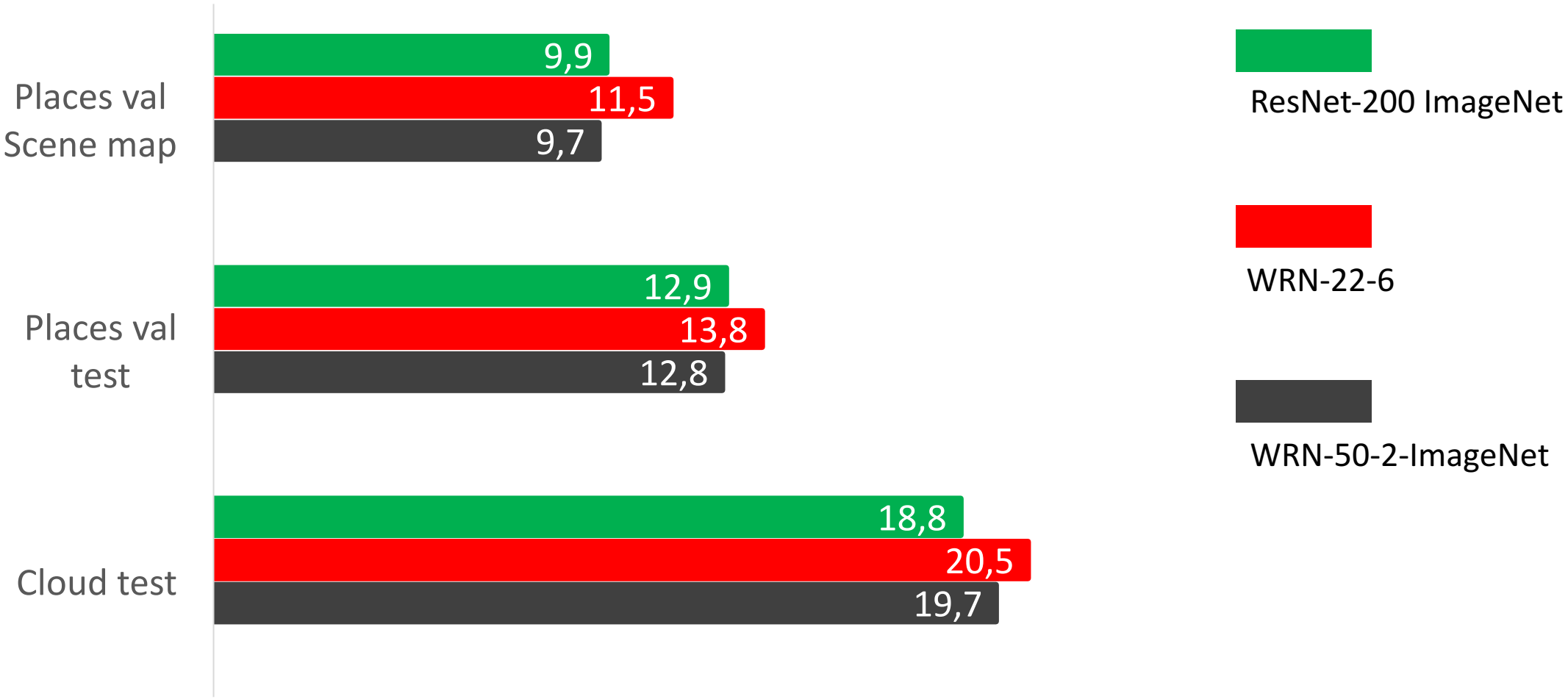
Результаты Wide ResNet (top 5 error, %)



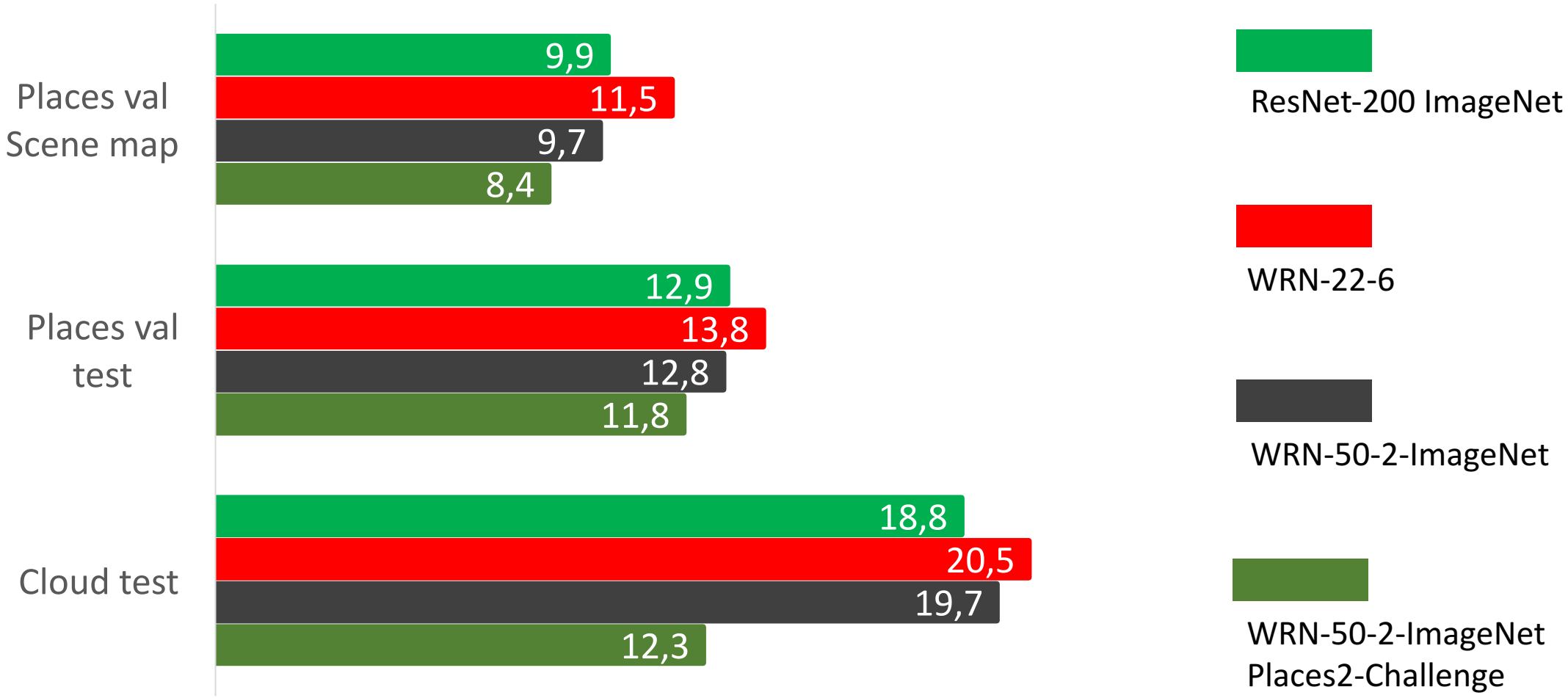
Результаты Wide ResNet (top 5 error, %)



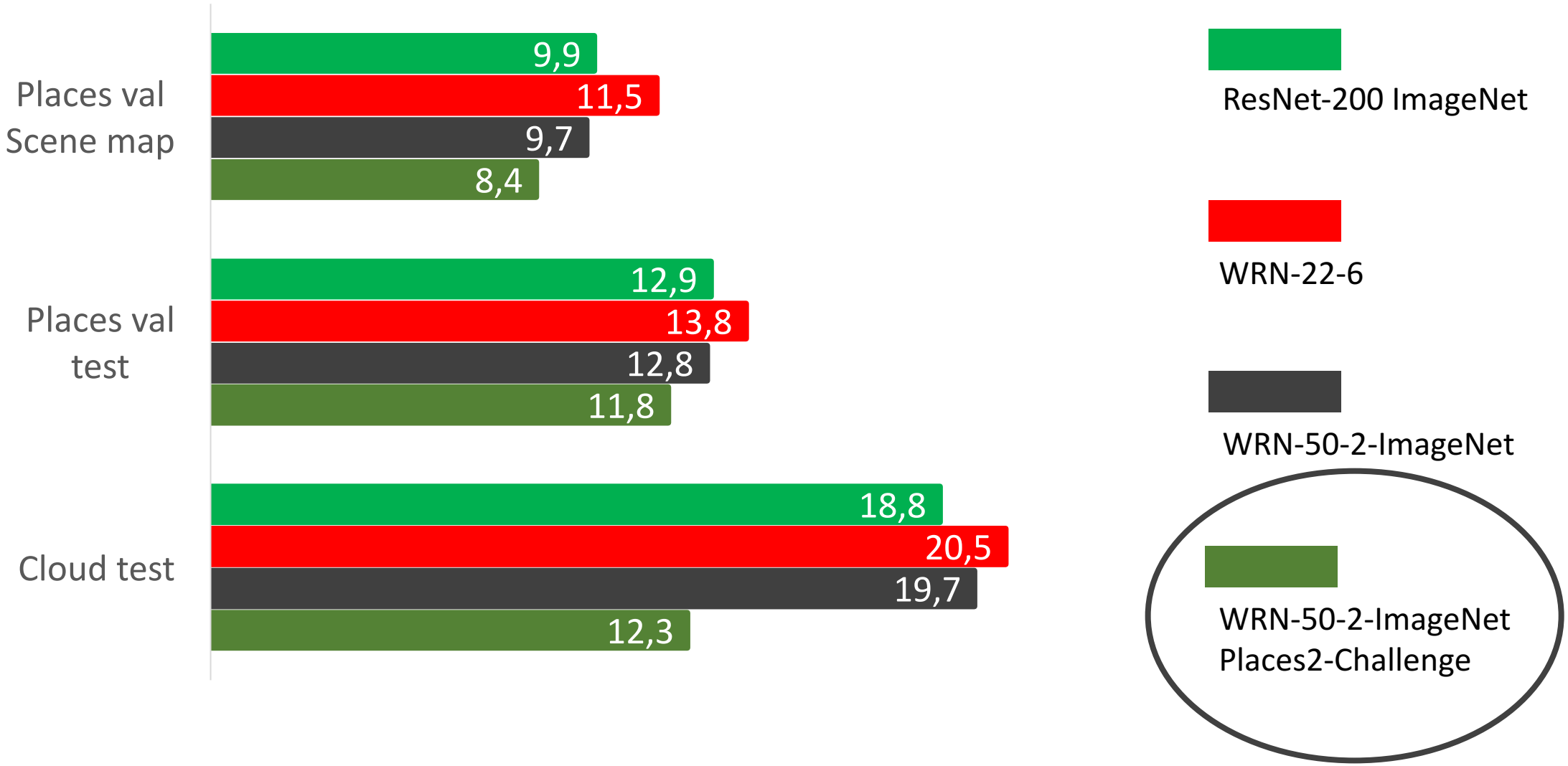
Результаты Wide ResNet (top 5 error, %)



Результаты Wide ResNet (top 5 error, %)



Лучшая сеть: WRN-50-2-ImageNet Places2-Challenge



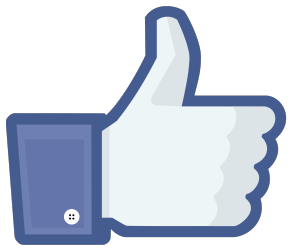
WRN-50-2

Predicted:

- restaurant 0.33
- pub/indoor 0.11
- beer_hall 0.1
- food_court 0.08
- dining_hall 0.07

Mapped labels:

- restaurant 0.62
- dining room 0.07



Примеры работы

Predicted:

- mountain_path 0.2
- rainforest 0.17
- forest_path 0.08
- creek 0.07
- valley 0.06

Mapped labels:

- forest 0.22
- mountain path 0.2
- creek 0.07
- valley 0.06

Predicted:

- harbor 0.42
- coast 0.13
- cliff 0.12
- promenade 0.07
- ocean 0.04

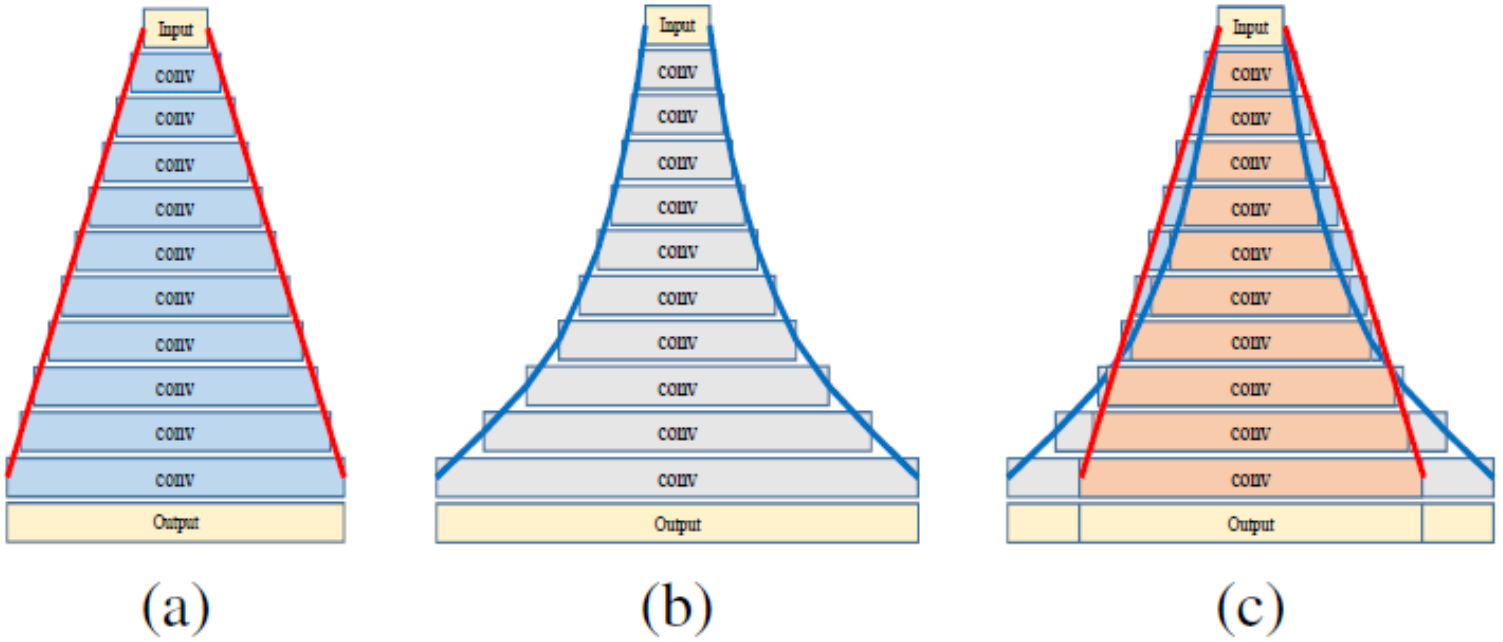
Predicted:

- palace 0.21
- museum 0.16
- plaza 0.12
- yard 0.1
- church 0.13

Улучшение модели

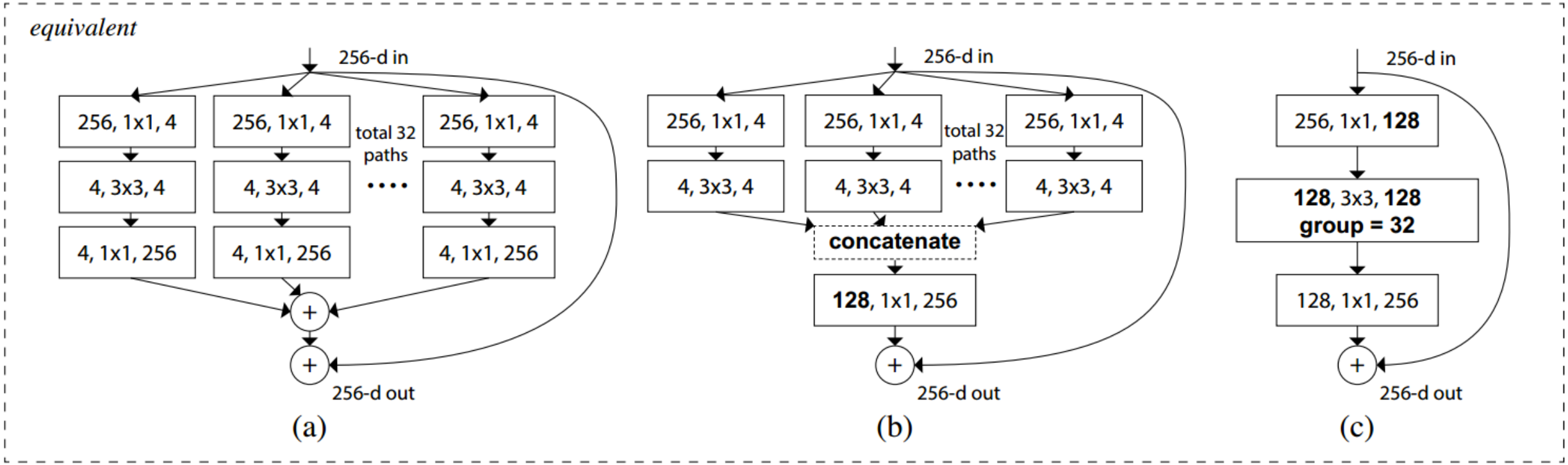
PyramidNet

- Семейство ResNet продолжает улучшаться
- PyramidNet: многообещающие результаты на CIFAR-10/100 и ImageNet
- Places val
  - top1 err: 43.57%
  - top-5 err: 13.12%



ResNext

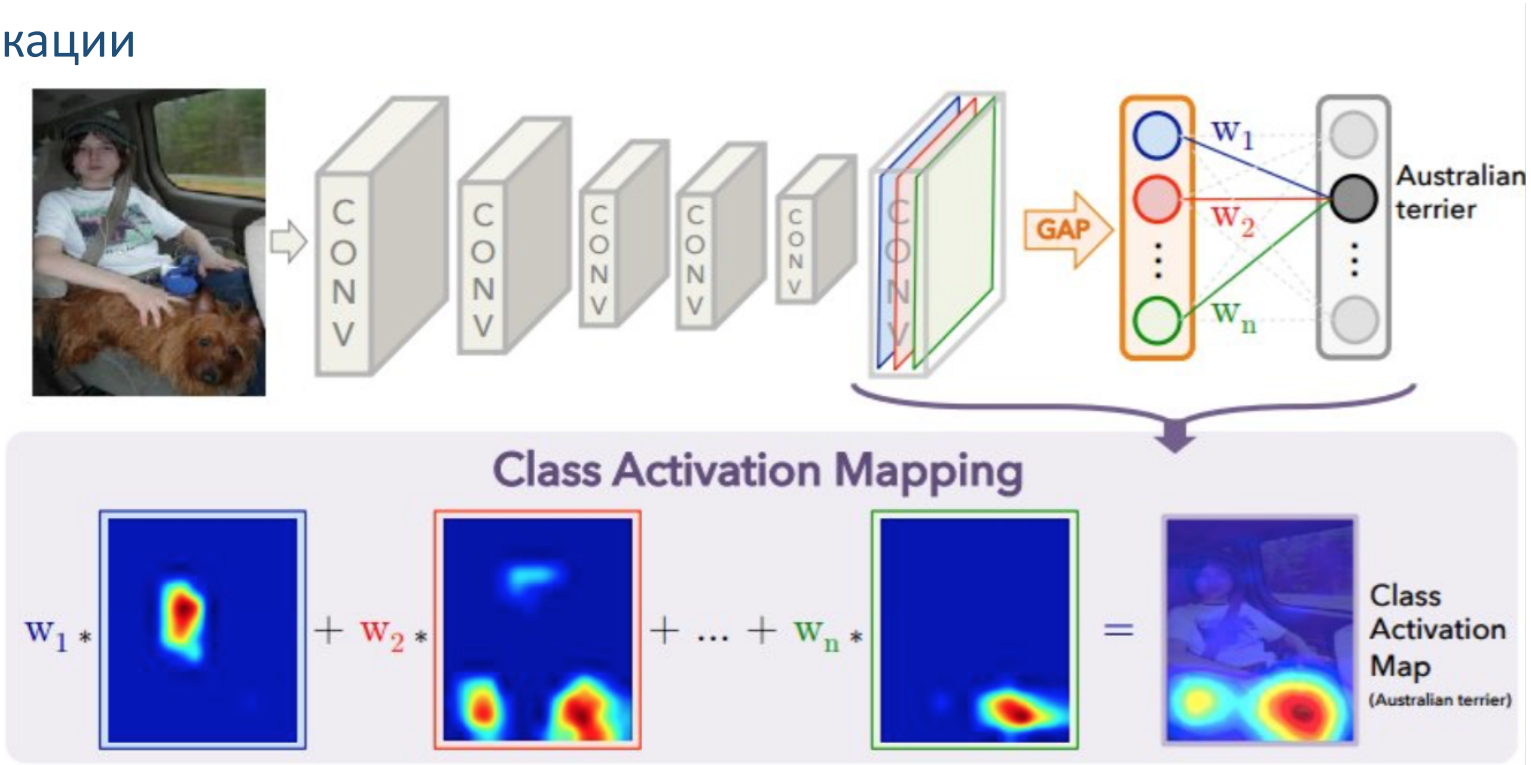
- Идея: разбить Residual блок на несколько параллельных блоков меньшего размера
- Показала результаты хуже нашей модели



“Креативные” подходы к улучшению CNN

Class activation mapping

– Class activation map (CAM) – это то, на что “смотрит” сеть на изображении при его классификации

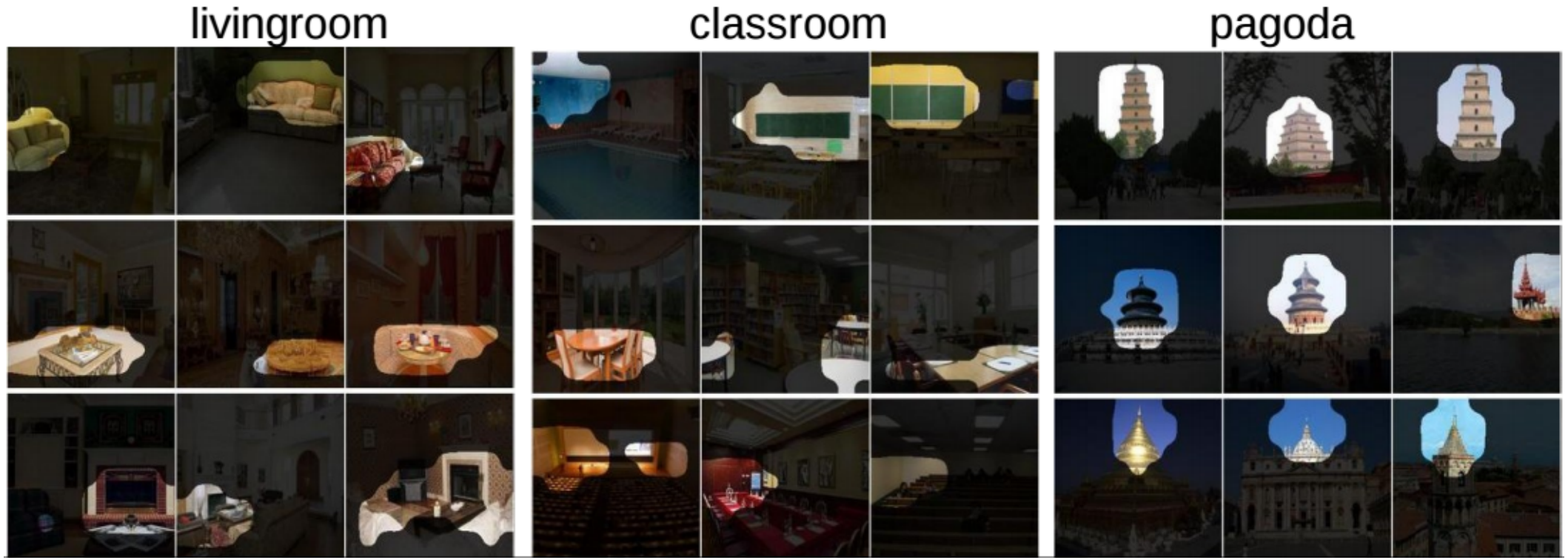


Class activation map для класса c:

$$M_c(x, y) = \sum_k w_k^c f_k(x, y)$$

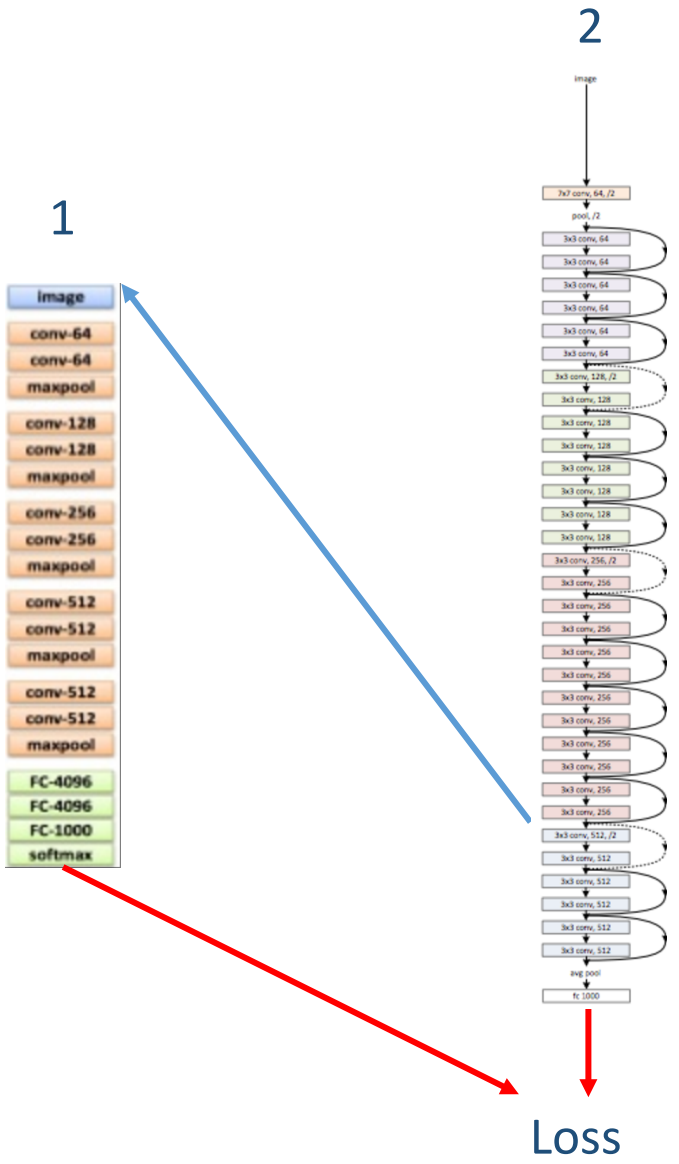
Class activation mapping

- Идея: экземпляры одной сцены должны иметь в качестве Class activation map (CAM) похожие объекты



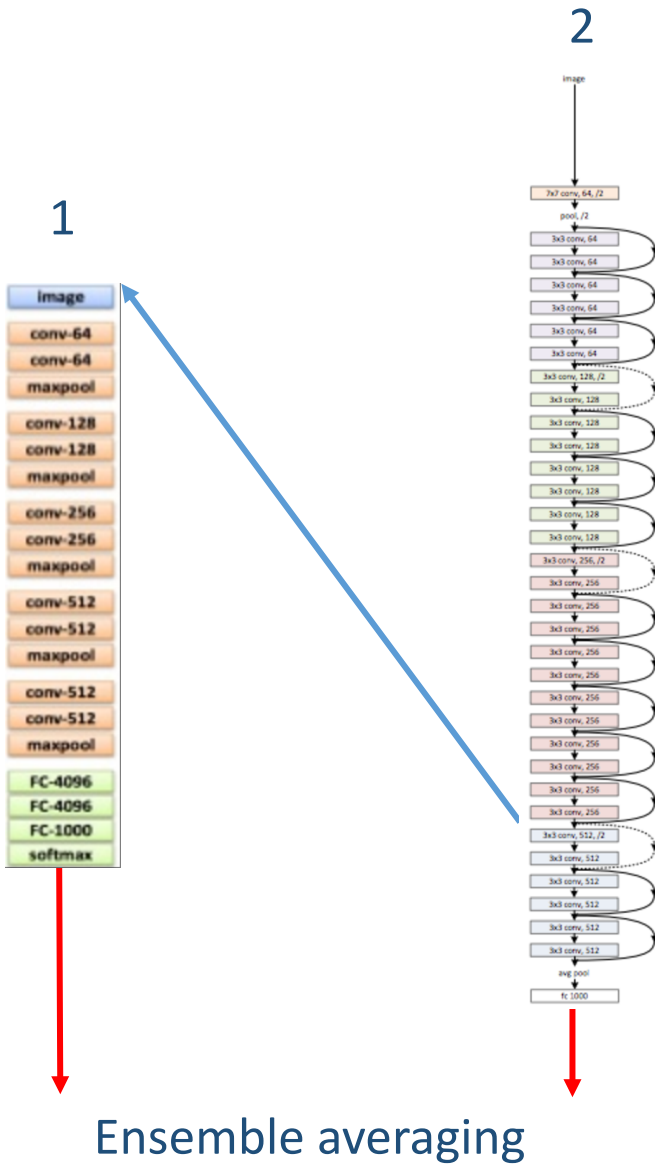
Добавление Class activation mapping в Loss

- Class activation map поступает на вход сети 1
- Её результат добавляется в Loss сети 2
- Сеть 2 обучается с новым Loss



Обучение новой сети на Class activation mapping

- Class activation map поступает на вход сети 1
- Обучается только сеть 1
- На inference используется ансамбль сетей 1 и 2



Class activation mapping: применение

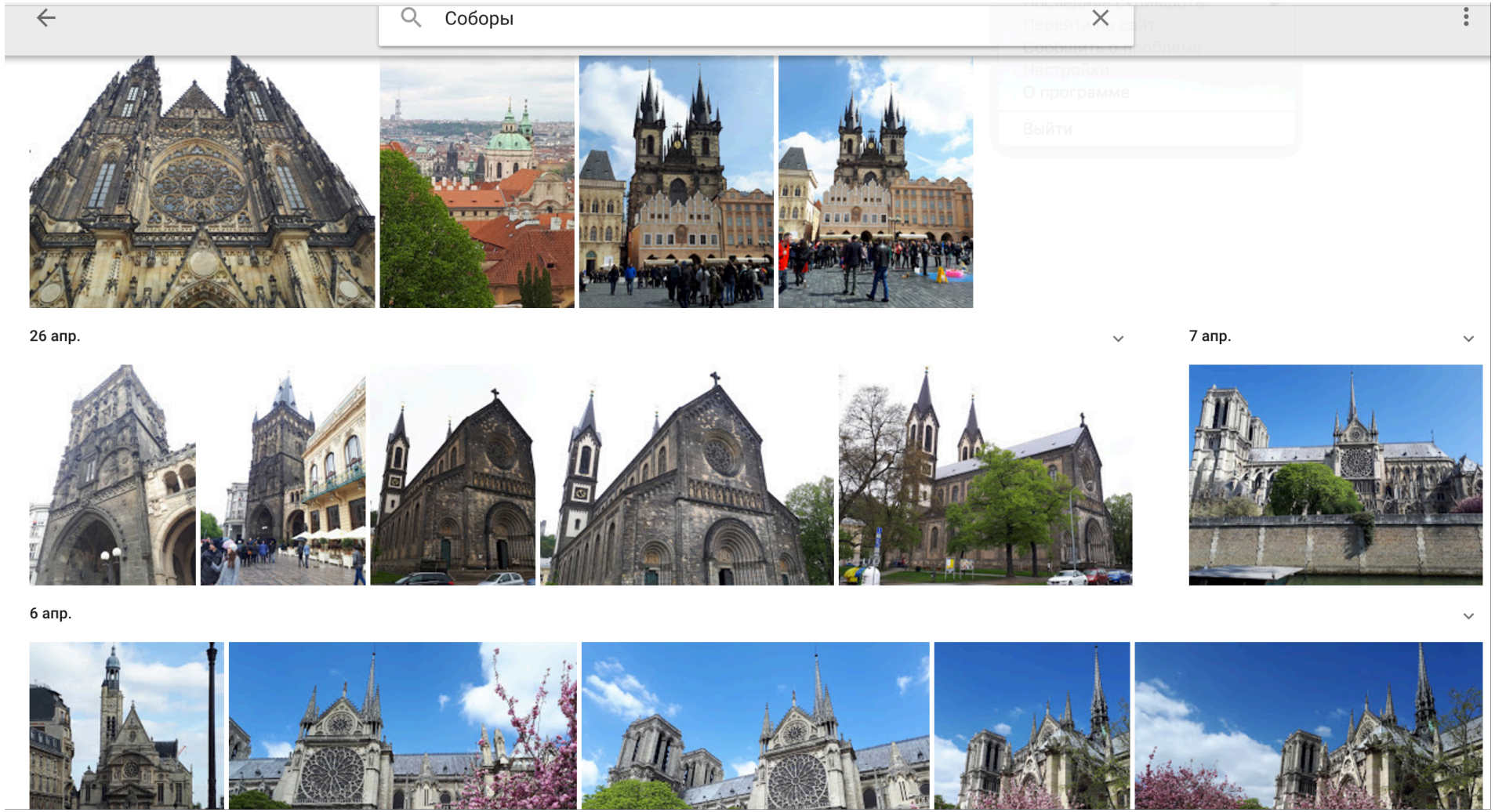
- Дообучали WRN-50-2
- В качестве сети 1 использовалась ResNet-50 ImageNet
- Повысить качество не удалось

Дальнейшие планы

- Обучать новые архитектуры CNN (из семейства ResNet)
- Попробовать другие варианты CAM
- Различные подходы с более “умной” обработкой патчей изображения

Распознавание достопримечательностей

- Необходимо распознавать “знаковые” места
- Пользователи часто фотографируют у достопримечательности



Сложности

- Практически нет исследований
- Нет готовых данных
- Малое количество “чистых” изображений в открытом доступе для каждой достопримечательности
- Неясно, что именно считать “достопримечательностью”
  - Дом с башнями на пл. Льва Толстого(СПб). Tripadvisor : нет, Google: да

Сбор базы

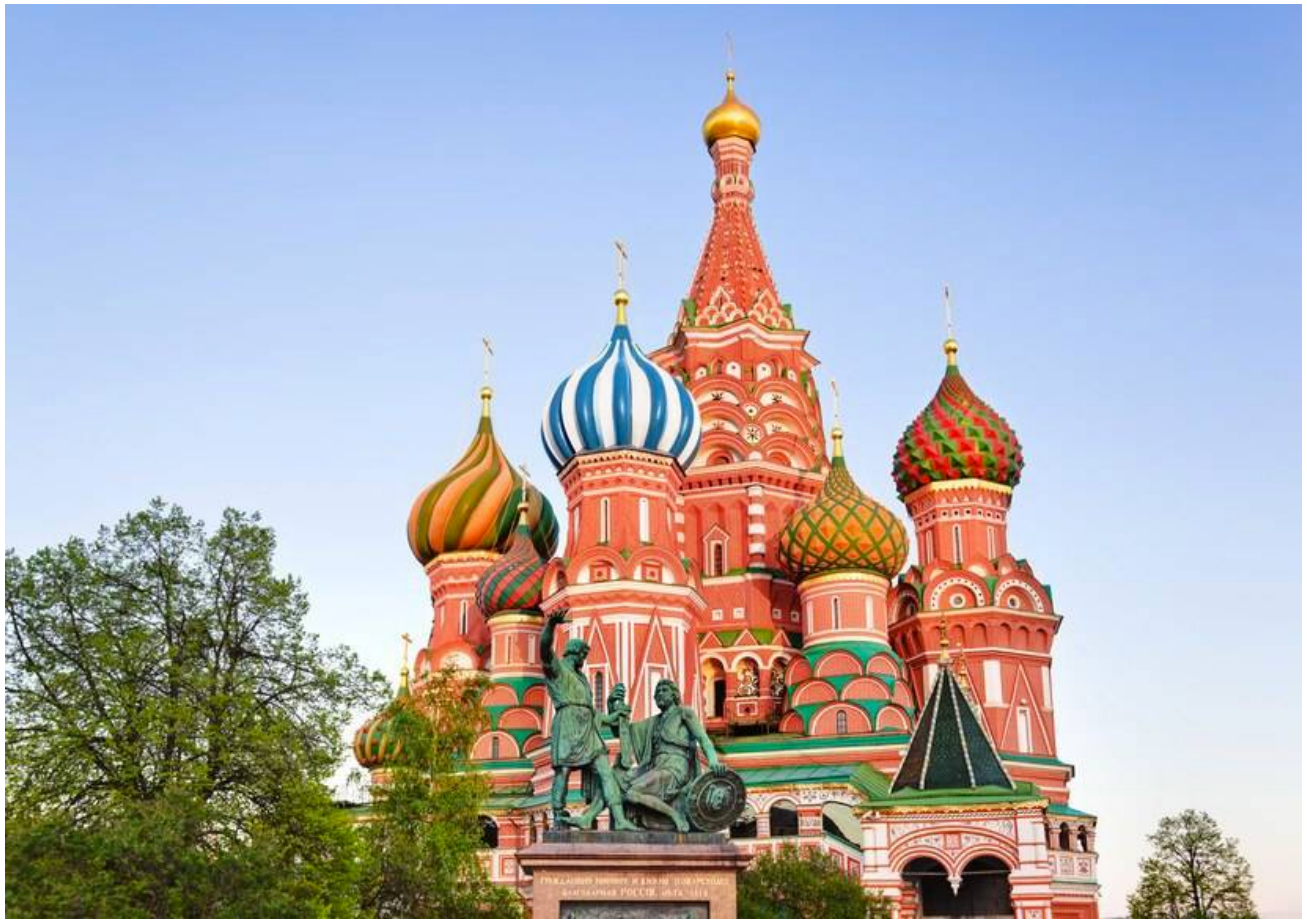
- Составляется список из 100 городов
- Для каждого города скачиваются json данные достопримечательностей с помощью Google Places API
- Эти данные парсятся и фильтруются
- По полученному списку достопримечательностей скачивается из поиска Google соответствующие картинки (по 20 штук)
- Итого: 2827 достопримечательностей, 56 540 изображений

Тесты

- Cloud test
  - Изображения от сотрудников
  - Ручная разметка
  - 200 картинок
  - 15 городов
  - 10 000 изображений не достопримечательностей
- Search test
  - Поиск Mail.Ru
  - 3-10 изображений на достопримечательность

Люди

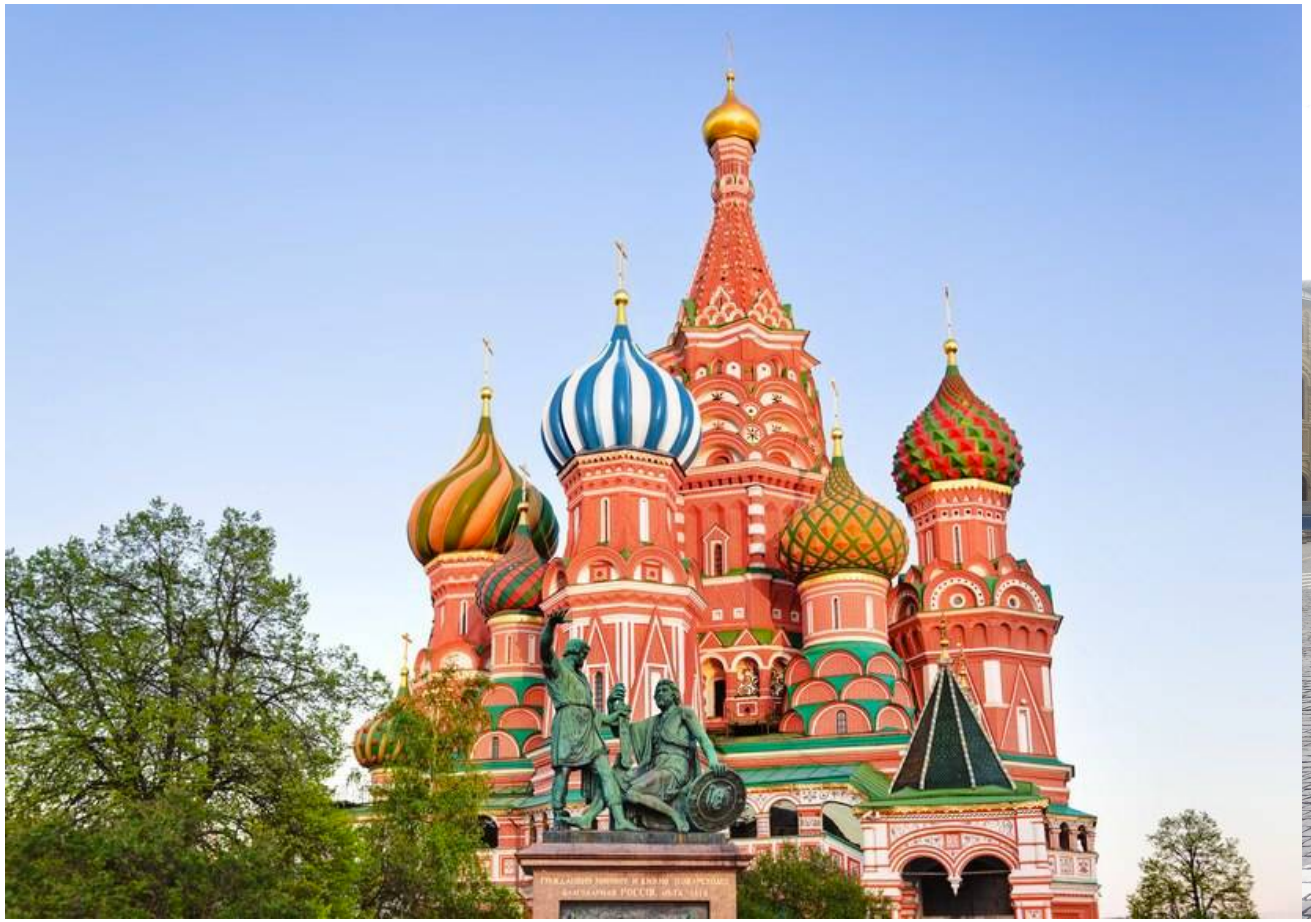
- Первые модели показали плохие результаты на Cloud test и “боевых” фотографиях



Train

Люди

- Первые модели показали плохие результаты на Cloud test и “боевых” фотографиях



Train

Test

“Людская” аугментация

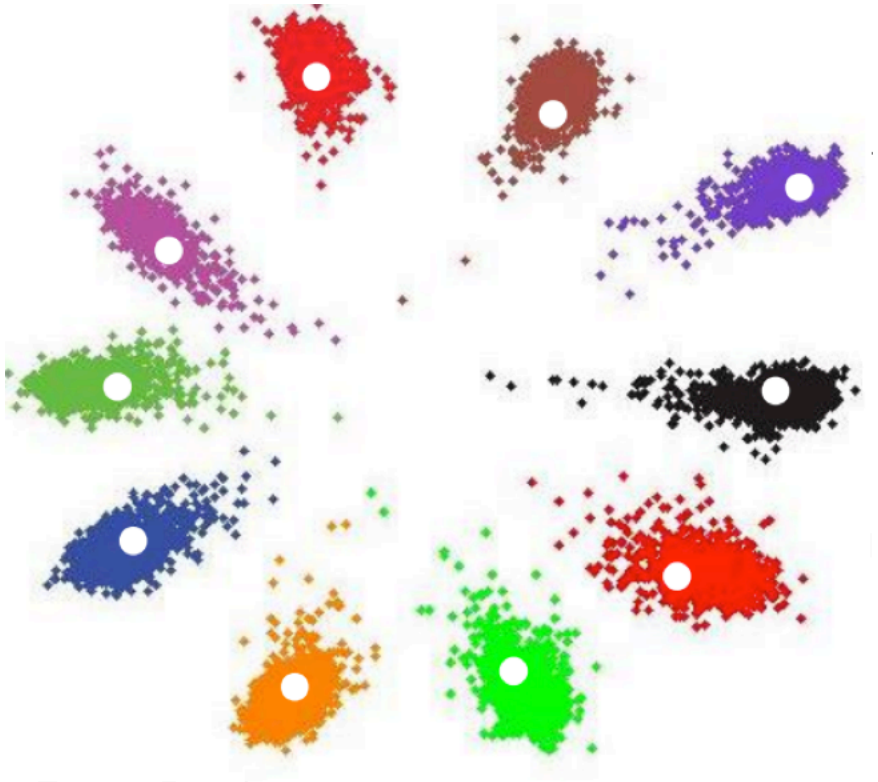
- Стандартные методы аугментации: поворот, случайное вырезание части изображения и т.д.
- В процессе обучения в некоторые изображения добавлялись люди

Fine tuning scene-модели

- База обучения маленькая, а достопримечательности – частный случай scene
- WRN-50-2, обученная на Places Sift
- Сверху сети добавляется несколько полносвязаных и BN слоёв
- Обучаются эти новые слои и 3 верхних Residual блока
- Вместо стандартного softmax loss используется center loss

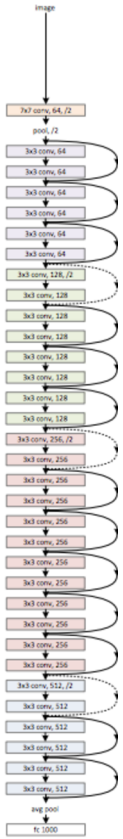
Center loss

- Center loss старается при обучении “растащить” представителей разных классов по разным кластерам
- При обучении в данные был добавлен ещё один класс ”не досторимечательность”
- К нему center loss не применялся



Embedding

- После обучения от сети отрезается последний классифицирующий слой
- Входное изображение после прогона через сеть превращается в числовой вектор (embedding)



embedding

Опорный вектор класса

embedding 1

embedding 2

embedding 3

опорный вектор

Определение достопримечательности

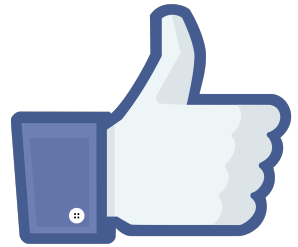
- Входное изображение прогоняется через сеть, и его эмбеддинг сравнивается с опорным вектором каждого класса
- Если результат сравнения меньше порога, то считаем, что на изображении не достопримечательность
- Иначе, берём класс с наибольшим значением сравнения

Результаты на тестах

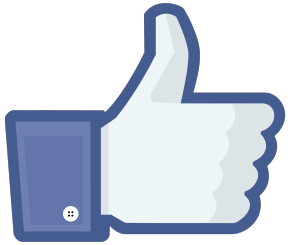
- Cloud test
  - Точность достопримечательности: 0.616
  - Точность не достопримечательности: 0.981
- Search test
  - Средняя точность: 0.669
  - Средняя полнота: 0.576

Примеры работы

Собор Василия Блаженного Москва

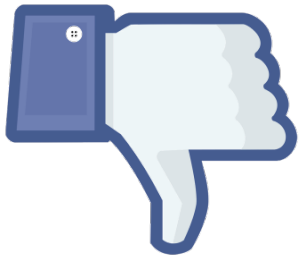


Cathédrale Notre-Dame de Paris



Old Town Square Прага

Not attraction



Дальнейшие планы

- Собирать базу по ещё большему количеству городов
- Новые методы обучения сети для данной задачи
- Исследовать возможности увеличения числа классов без переобучения сети

Заключение

- Существующие наборы данных для Scene recognition
- Wide Residual Network – лучшая модель
- Дальнейшие планы по улучшению модели
- Обзор задачи распознавания достопримечательностей
- Алгоритм сбора базы
- Обучение модели для распознавания достопримечательностей

Спасибо за внимание!
Вопросы?

a.boiarov@corp.mail.ru

@mail.ru group