

Описание программной реализации

Разработан скрипт реализующий:

1. Загрузку данных с помощью модуля Pillow. Входные изображения хранятся в формате .jpeg и имеют 3 цветовых канала R,G,B. Преобразование входных данных: нормализация $[0..255] \rightarrow [0..1]$, приведение размера к 128x128, разбиение на тренировочную и тестовую выборки в соотношении 70% к 30%.
2. Загрузку основных модулей Keras и TensorFlow для дальнейшей работы, установка начальных параметров. Описание тестируемых моделей глубоких сверточных сетей.

Тестовые конфигурации сетей

Была использована известная сеть – VGG16

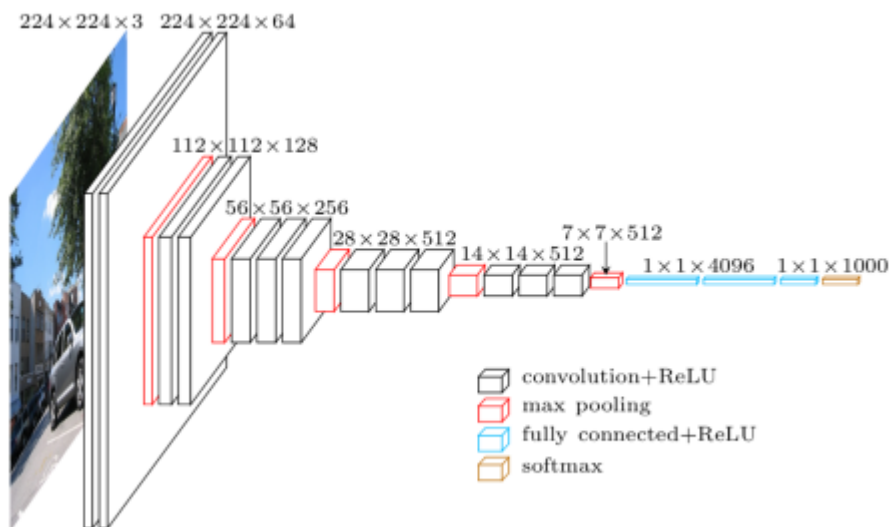
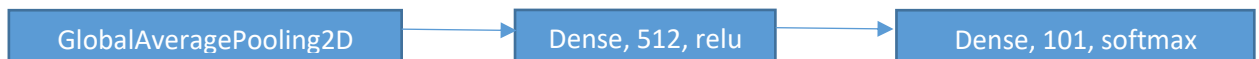


Рис 1 (<http://www.cs.toronto.edu/~frossard/post/vgg16/>)

с измененным размером входных данных (128x128x3) и классификатором



Для сверточных слоев были взяты обученные веса (https://github.com/fchollet/deep-learning-models/releases/vgg16_weights_tf_dim_ordering_tf_kernels_notop.h5)

model – сеть для исходной задачи (VGG16).

my_model – сеть для целевой задачи, отличается от сети для исходной по структуре только классификатором

Целевая задача – распознавание 101 класса на данных **101_ObjectCategories**

Исходная задача – распознавание 1000 классов на данных

Описание экспериментов:

1. Замена классификатора в model для распознавания 101-ого класса, обучение нового классификатора на 101_ObjectCategories (веса в сверточных слоях заморожены), тестирование на 101_ObjectCategories
2. Замена классификатора в model для распознавания 101-ого класса, обучение всей сети на 101_ObjectCategories (веса в сверточных слоях проинициализированы обученными весами из исходной AlexNet), тестирование на 101_ObjectCategories

3. Замена классификатора в model для распознавания 101-ого класса, обучение всей сети на 101_ObjectCategories, при этом веса сверточных слоев переинициализируются, то есть используется только структура исходной AlexNet. Тестирование на 101_ObjectCategories

Эксперименты

Тип	Общее время выполнения (с)	Качество решения целевой задачи (точность на тестовом наборе %)
Использование модели, построенной для решения исходной задачи, в качестве фиксированного метода извлечения признаков при построении модели, решающей целевую задачу	570	0.771
Тонкая настройка параметров модели, построенной для решения исходной задачи, с целью решения целевой	1380	0.8
Использование структуры глубокой модели, построенной для решения исходной задачи, с целью обучения аналогичной модели для решения целевой задачи	1374	0.559