

Сравнение некоторых методов глубокого обучения для решения задачи классификации автомобилей на изображениях*

В.Д. Кустикова¹, Д.А. Кручинин¹, П.Н. Дружков^{1,2}

valentina.kustikova@itmm.unn.ru | kruch.dmitriy@gmail.com | pavel.druzhkov@itseez.com

Россия, ¹Институт информационных технологий, математики и механики,

Нижегородский государственный университет им. Н.И. Лобачевского;

² Itseez

Рассматривается задача классификации автомобилей на изображениях. Дается обзор методов компьютерного зрения и машинного обучения для решения задачи. Сравнивается подход, основанный на иерархическом представлении исходных данных и обучении ансамблей деревьев решений, который позволил ранее получить свыше 97.5% точности на небольших изображениях в отсутствии мелких видимых деталей и естественного фона, с “переносом обучения” глубоких сверточных нейронных сетей. На реальных данных первый подход позволяет получить 86% средней точности, второй – свыше 97% для большинства выбранных архитектур нейросетей. Таким образом, метод, основанный на ансамблях деревьев решений, не позволяет добиться сравнимых показателей точности на реальных изображениях. При этом эксперименты показывают, что увеличение глубины модели, использующей деревья решений, приводит к росту точности классификации, а увеличение числа обучаемых параметров нейронной сети не позволяет получить дополнительных информативных признаков и ведет к снижению точности.

Ключевые слова: машинное обучение, глубокое обучение, сверточные сети, библиотека Caffe, деревья решений, компьютерное зрение, классификация изображений

1. Введение

Задача классификации изображений является одной из классических задач компьютерного зрения [30], которая заключается в том, чтобы определить наличие или отсутствие объекта определенного класса на изображении. В данной работе рассматривается задача классификации автомобилей на изображениях, возникающая при анализе транспортных потоков, обеспечении автоматического регулирования и выявлении нарушений ПДД.

Автомобили являются геометрически структурными объектами, поэтому для классификации широко используются методы, основанные на построении трехмерных каркасов [1] и выделении контуров [22]. Неустойчивость указанных методов к степени освещенности, аффинным преобразованиям и разного рода перекрытиям приводит к появлению методов, основанных на извлечении инвариантных признаков (HOG-признаки [9], [26], признаки Габора [28]) и последующем обучении моделей, описывающих представителей всего класса объектов (метод опорных векторов (Support Vector Machine, SVM), Adaboost [14]). В настоящее время значительное внимание уделяется подходам, использующим глубокое обучение (deep learning) [7]. Основу

подходов составляет идея построения многоуровневых иерархий представления данных, в основном с помощью нейронных сетей [11]. Так, в [10] строится *сверточная нейронная сеть* (Convolutional Neural Network), которая по изображению позволяет выделить локальные признаки, характеризующие ребра, отдельные части объекта, высокоуровневые признаки полного изображения и на выходе выдает достоверность присутствия автомобиля. В [15] предлагается построить *глубокую машину Больцмана* (Deep Boltzmann Machine) – вероятностную модель нейросети, – используя в качестве входа смесь инвариантных признаков. Также интерес проявляется к такому направлению в глубоком обучении, как “*перенос обучения*” (transfer learning) [32]. Цель “переноса” состоит в том, чтобы обучить глубокое представление (например, веса нейронной сети) для решения одной задачи и использовать его при решении другой.

В работе сравнивается подход, основанный на построении многоуровневых иерархий представления данных и обучении ансамблей деревьев решений [12], с “переносом обучения” глубоких сверточных нейронных сетей. При решении задачи классификации автомобилей рассматриваются сверточные сети GoogLeNet [29], CaffeNet [18], VGGNet [25], для которых получены лучшие результаты [23] на задаче многоклассовой классификации изображений базы ImageNet [16]. Подход, основанный на обучении ансамблей деревьев решения, в отличие от нейросетевых позволяет работать не только с интенсивностями пикселей, но и с более сложными исходными признаковыми описаниями. Поэтому сравне-

Работа выполнена при поддержке гранта РФФИ по теме «Исследование применимости методов глубокого обучения к решению задач компьютерного зрения на примере классификации изображений с большим числом категорий, детектирования пешеходов и автомобилей» (договор № НК 14-07-31269/15). Работа опубликована при финансовой поддержке РФФИ, грант №16-07-20482

ние качества работы обоих подходов представляет интерес.

2. Задача классификации автомобилей на изображениях

Задача классификации автомобилей на изображениях [33] предполагает бинарный ответ о наличии/отсутствии объекта. Задача состоит в том, чтобы исходному изображению I поставить в соответствие метку 1, если автомобиль присутствует на изображении, и 0, в противном случае. Поэтому необходимо построить отображение $\varphi : I \rightarrow \{0, 1\}$. φ может быть представлено как суперпозиция $\varphi = \psi \cdot \theta$, где $\theta : I \rightarrow F$ обеспечивает построение признакового описания F для I , а $\psi : F \rightarrow \{0, 1\}$ по признаковому описанию вычисляет метку. *Признаковое описание* представляет собой множество вещественных векторов или *дескрипторов*, построенных на основании значений функции интенсивности пикселей.

3. Общая схема решения задачи с использованием глубокого обучения

Общая схема решения задачи классификации с помощью методов глубокого обучения состоит из следующих этапов:

1. *Обучение классификатора.* Этап обучения предполагает, что на входе имеется набор изображений с известными метками классов — *тренировочная выборка*. Для каждого изображения выполняется построение иерархического признакового описания. Описание представляется в виде матриц вещественных векторов, построенных на каждом уровне представления. Далее на основании множества признаковых описаний и меток классов осуществляется подбор — *обучение* — параметров классификатора.
2. *Тестирование классификатора (классификация).* На этапе классификации на вход приходит новое изображение, для которого выполняется построение признакового описания согласно тому же алгоритму, что и при обучении классификатора. Описание передается обученному классификатору, который принимает решение о принадлежности изображения тому или иному классу и, возможно, вычисляет достоверность принадлежности каждому из допустимых классов.

4. Сравнимые методы

В процессе исследования сравнивается метод, основанный на построении иерархического представления исходных данных и обучении ансамблей деревьев решений, с методами “переноса обучения” сверточных нейронных сетей.

Описание основных идей метода, обеспечивающего построение иерархического представления данных, дается в [12]. Ниже предлагается его модификация, состоящая в том, что на входе рассматривается не полное изображение, а набор его частей, и решение о принадлежности изображения классу принимается на основании решений, полученных для частей. Необходимость модификации обусловлена существенным усложнением данных (вариативность фона и ракурсов автомобилей). Модифицированный метод предполагает выполнение следующей последовательности действий:

1. Исходное изображение разбивается на небольшие части фиксированного размера (*патчи*), для каждой из которых вычисляется признаковое описание. В результате для изображения формируется *карта признаков*.
2. Строится ансамбль деревьев для классификации патчей. Каждое дерево ансамбля определяет новый признак. Значением признака является индекс листа, в который попадает рассматриваемый объект.
3. С помощью полученных деревьев строится новое описание патчей, определяющее новую карту признаков, которая вновь разбивается на патчи. Шаги 2–3 повторяются до момента достижения требуемой глубины модели.
4. На итоговых признаковых описаниях выполняется построение конечного классификатора.
5. Классификация изображения осуществляется на основе классификации патчей, например, посредством голосования.

Необходимо отметить, что в качестве исходного представления данных (ранее изображение) можно использовать как значения функции интенсивности пикселей, так и более сложные признаковые описания. Это отличает предложенный метод от нейросетевых, основанных на обучении глубоких сверточных сетей.

Нейросетевые методы включают в себя построение архитектуры нейронной сети и обучение весов сети с использованием *метода обратного распространения ошибки* (back propagation). “*Перенос обучения*” предполагает, что архитектура сети разработана, и сеть обучена для решения некоторой исходной задачи. Далее необходимо модифицировать выходные слои нейронной сети, чтобы специализировать ее для решения интересующей задачи. Типичная модификация состоит в изменении количества выходов сети, например, при переходе от задачи классификации изображений с N классами к задаче с K классами. На этапе обучения сети веса неизменных слоев инициализируются значениями, полученными при обучении сети для решения исходной задачи. Затем сеть так же обучается с помощью метода обратного распространения ошибки. При этом

параметры обучения можно настроить таким образом, что веса обучаются исключительно для модифицированных слоев (для остальных остаются практически неизменными), либо для всех слоев сети.

5. Программная реализация

Для проверки применимости предложенного метода к решению задач классификации автомобилей выполнена программная реализация [8] на базе алгоритмов *случайного леса* (random forest) [2] и *градиентного бустинга деревьев решений* (gradient boosting trees) [13]. Реализация разработана на языке программирования Python с использованием библиотек OpenCV [6], XGBoost [20] и scikit-learn [24].

Обеспечивается возможность обучения модели произвольной глубины, параметры каждого слоя могут указываться независимо. Реализация является кроссплатформенной.

Сравнение с нейросетевыми подходами, основанными на идеях “переноса обучения”, проводилось с использованием библиотеки Caffe [4]. Библиотека содержит функционал, необходимый для конфигурирования, обучения и тестирования полностью связанных и сверточных нейронных сетей [19]. Caffe в зависимости от типа сборки может работать, как на центральном процессоре в последовательном режиме, так и на графическом процессоре с целью ускорения вычислений.

6. Наборы данных

Обзор литературы последних лет показывает, что наборы данных Stanford Dataset [27] и KITTI [17] являются широко применяемыми при анализе качества классификации и детектирования автомобилей на изображениях. Указанные наборы также могут быть использованы и для бинарной классификации посредством извлечения изображений объектов и фоновых областей. Основные причины выбора наборов – большой объем выборок и вариативность внешнего вида автомобилей на изображениях (таблица 1).

Для проведения экспериментов тренировочная выборка сформирована из данных KITTI и содержит 21798 положительных и 81043 отрицательных примеров. Следует отметить, что набор данных KITTI состоит из сцен дорожного движения, в связи с этим для решения задачи классификации предварительно из изображений вырезаются транспортные средства согласно разметке. Выборка дополняется изображениями, не содержащими автомобилей, — *отрицательными примерами* — посредством извлечения областей, соответствующих фону. Для этого сцена разделяется на изображения фиксированного размера. Фрагмент добавляется

Таблица 1: Описание наборов данных.

Набор	Описание данных
Stanford Dataset [16]	1. Вид в 3/4 спереди и сзади. 2. 196 классов автомобилей. 3. Минимальный размер автомобиля на изображении — 73×43, максимальный размер — 3896×2428. 4. Количество сцен в тренировочном наборе: 8144, количество сцен в тестовом наборе — 8041.
KITTI Dataset [17]	1. Разный ракурс, в разметке указан угол поворота объекта (ориентация). 2. На сайте опубликованы результаты качества классификации с использованием ранее разработанных методов. 3. Метки классов тестовых изображений не g5 в разметке, для тестирования используется сервер. 4. Минимальный размер автомобиля на изображении — 16×12, максимальный размер — 500×375. 5. Количество сцен в тренировочном наборе: 7481, количество сцен в тестовом наборе — 7518.

к множеству отрицательных примеров, если перекрытие с размеченным автомобилем не превышает порогового значения в 30%.

Тестовая выборка составлена из 8041 положительного примера Stanford Dataset и 70471 отрицательных примеров, полученных случайным образом из набора PASCAL VOC 2007 [31]. Отметим, что разметка тестовой выборки KITTI является закрытой, что осложняет использование этих данных и многократное оценивание качества.

7. Результаты экспериментов

В ходе выполнения экспериментов метод, основанный на построении иерархического представления данных и обучении ансамблей деревьев решений, запускается для различных исходных признаков описаний. Лучшие результаты получены для функции интенсивности пикселей и гистограммы ориентированных градиентов (Histogram of Oriented Gradients, HOG). Детальное описание параметров экспериментов приведено ниже (таблица 2). В качестве классификатора используется градиентный бустинг деревьев решений (GBT) [13].

Для оценки качества работы “переноса обучения” исследуются широко известные конфигурации сверточных нейронных сетей: GoogLeNet [29], CaffeNet [18], VGGNet [25]. Последние слои сетей изменяются в соответствии с числом выходов в задаче классификации автомобилей. При дообучении веса инициализируются значениями, полученными в результате обучения сетей на данных ImageNet [16] в задаче многоклассовой классификации. При этом выполняется три типа экспериментов: обучение весов последнего слоя (part tuning), дообучение весов всех слоев (fine tuning) и полное обучение весов сети (full training) (таблица 3).

Для оценки качества классификации используются три показателя точности. Первый показатель отра-

Таблица 2: Параметры экспериментов для метода, основанного на обучении ансамблей деревьев решений.

№	Параметры
1	1. Исходное признаковое описание: интенсивность. 2. Исходный размер изображения: 32×32. 3. Размеры патчей: 8×8 для исходного изображения, 2×2 для карт признаков. 4. Шаг патчей: 4×4 для исходного изображения, 1×1 для карт признаков. 5. Модель для извлечения признаков: случайный лес [2], обученный на случайном разбиении выборки на классы. 6. Количество уровней: 1.
2	1. Исходное признаковое описание: интенсивность. 2. Исходный размер изображения: 32×32. 3. Размеры патчей: 8×8 для исходного изображения, 2×2 для карт признаков. 4. Шаг патчей: 4×4 для исходного изображения, 1×1 для карт признаков. 5. Модель для извлечения признаков: случайный лес [2], обученный на случайном разбиении выборки на классы. 6. Количество уровней: 2.
3	1. Исходное признаковое описание: интенсивность. 2. Исходный размер изображения: 32×32. 3. Размеры патчей: 8×8 для исходного изображения, 2×2 для карт признаков. 4. Шаг патчей: 4×4 для исходного изображения, 1×1 для карт признаков. 5. Модель для извлечения признаков: градиентный бустинг деревьев решений (GBDT). 6. Количество уровней: 3.
4	1. Исходное признаковое описание: гистограмма ориентированных градиентов (HOG). 2. Исходный размер изображения: 64×64. 3. Размеры патчей: 16×16. 4. Шаг патчей: 8×8. 5. Модель для извлечения признаков: случайный лес [2]. 6. Количество уровней: 3.

жает процент правильно классифицированных положительных примеров, второй – процент правильно классифицированных отрицательных (фоновых) примеров, третий – средняя точность классификации по всем примерам. В таблицах 4 и 5 приведены полученные значения показателей точности для всех экспериментов, параметры которых представлены в таблицах 2 и 3. Номера строк таблиц параметров соответствуют номерам строк таблиц значений точности.

Результаты экспериментов показывают, что модификация метода, основанного на обучении деревьев решений, в среднем показывает более 78% точности, лучший результат составляет 86% (таблица 4). При этом увеличение числа уровней в иерархии признаков в целом (для примера приведены первые два эксперимента) приводит к росту точности классификации.

Применение “переноса обучения” при решении поставленной задачи дает существенный прирост качества классификации. Сразу несколько сетей показывают свыше 97% точности (таблица 5). При этом ошибка классификации отрицательных примеров не превышает 5.13%. Следует отметить, что с усложнением архитектуры сети (CaffeNet и GoogLeNet имеют более простую архитектуру по сравнению с VGGNet) показатели точности снижаются. По результатам экспериментов для данной

Таблица 3: Параметры экспериментов с “переносом обучения” сверточных нейронных сетей.

№	Общие параметры	Частные параметры
GoogLeNet		
1	Параметры обучения с использованием стохастического градиентного спуска:	1. Дообучение весов всех слоев. 2. Параметры обучения: - Коэффициент спуска (learning rate multiplier): на последнем слое -- 1, на остальных -- 0.1. 3. Параметры тестирования: - Размер “пачки” (batch): 50.
2	- Количество итераций: 60000. - Размер “пачки” (batch): 32. - Скорость обучения (learning rate): 0.0001.	1. Обучение весов последнего слоя при фиксированных начальных весах остальных слоев. 2. Параметры обучения: - Коэффициент спуска (learning rate multiplier): на трех последних слоях -- 1, на остальных -- 0.1. 3. Параметры тестирования: - Размер “пачки” (batch): 50.
3	- Коэффициент изменения скорости обучения (learning rate decay): 0.96.	1. Полное обучение весов сети. 2. Параметры обучения: - Коэффициент спуска (learning rate multiplier), везде-1. 3. Параметры тестирования: - Размер “пачки” (batch): 50.
CaffeNet		
4	Параметры обучения с использованием стохастического градиентного спуска:	1. Дообучение весов всех слоев. 2. Параметры обучения: - Коэффициент спуска (learning rate multiplier): на последнем слое -- 1, на остальных -- 0.1. 3. Параметры тестирования: - Размер “пачки” (batch): 50.
5	- Количество итераций: 100000. - Размер “пачки” (batch): 32. - Скорость обучения (learning rate): 0.001.	1. Обучение весов последнего слоя при фиксированных начальных весах остальных слоев. 2. Параметры обучения: - Коэффициент спуска (learning rate multiplier): на трех последних слоях -- 1, на остальных -- 0.1. 3. Параметры тестирования: - Размер “пачки” (batch): 50.
6	- Коэффициент изменения скорости обучения (learning rate decay): 0.1.	1. Полное обучение весов сети. 2. Параметры обучения: - Коэффициент спуска (learning rate multiplier), везде-1. 3. Параметры тестирования: - Размер “пачки” (batch): 50.
VGGNet		
4	Параметры обучения с использованием стохастического градиентного спуска:	1. Дообучение весов всех слоев. 2. Параметры обучения: - Коэффициент спуска (learning rate multiplier): на последнем слое -- 1, на остальных -- 0.1. 3. Параметры тестирования: - Размер “пачки” (batch): 50.
5	- Количество итераций: 100000. - Размер “пачки” (batch): 32. - Скорость обучения (learning rate): 0.0005.	1. Обучение весов последнего слоя при фиксированных начальных весах остальных слоев. 2. Параметры обучения: - Коэффициент спуска (learning rate multiplier): на трех последних слоях -- 1, на остальных -- 0.1. 3. Параметры тестирования: - Размер “пачки” (batch): 50.
6	- Коэффициент изменения скорости обучения (learning rate decay): 0.1.	1. Полное обучение весов сети. 2. Параметры обучения: - Коэффициент спуска (learning rate multiplier), везде-1. 3. Параметры тестирования: - Размер “пачки” (batch): 50.

Таблица 4: Качество классификации с помощью метода, основанного на обучении ансамблей деревьев решений.

№	Точность классификации положительных примеров, %	Точность классификации отрицательных примеров, %	Средняя точность, %
1	77.5	87.3	82.4
2	83.8	88.4	86.1
3	79.1	89	84.05
4	86.5	70.9	78.7

задачи полное обучение весов сети для всех тестируемых архитектур показывает лучшие результаты (таблица 5, каждый третий эксперимент).

Таблица 5: Качество классификации с помощью метода, основанного на обучении ансамблей деревьев решений.

№	Точность классификации положительных примеров, %	Точность классификации отрицательных примеров, %	Средняя точность, %
1	98.75	96.22	97.49
2	99.32	92.37	95.85
3	99.55	94.87	97.21
4	99.33	95.53	97.43
5	98.93	96.99	97.96
6	98.19	97.73	97.96
7	99.65	84.38	92.02
8	99.50	88.93	94.22
9	99.57	92.96	96.27

Ниже (рис. 1) показаны примеры ошибочных срабатываний обученных классификаторов: положительные примеры, для которых классификатор принял решение об отсутствии автомобиля, и отрицательные примеры, где классификатор принял решение о наличии такового.

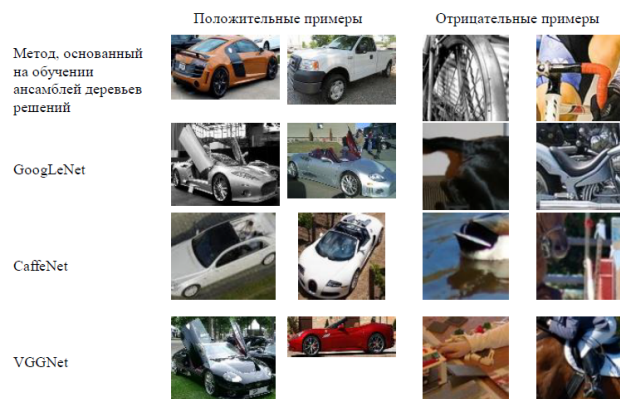


Рис. 1: Примеры неправильно классифицированных изображений.

Из представленных изображений можно видеть, что сети некорректно классифицируются автомобили нестандартной конструкции (с дверцами, открывающимися вверх) или ракурса (вид сверху). Данный факт объясняется небольшим количеством аналогичных положительных примеров в тренировочном множестве. Поскольку классификация изображения с помощью глубоких ансамблей деревьев решений основана на классификации отдельных патчей небольшого размера, наличие на изображении, например, колес велосипеда, может привести к принятию классификатором неправильного решения, что можно видеть на примерах, показанных ниже (рис. 1, строка 1, изображение 3). Устранение указанных недостатков метода является одним из направлений дальнейших исследований.

8. Заключение

В работе рассмотрена задача классификации автомобилей на изображениях. Выполнено сравнение подхода, основанного на построении иерархического представления исходных данных и обучении ансамблей деревьев решений, с “переносом обучения” глубоких сверточных нейросетей. Метод, основанный на ансамблях деревьев решений, продемонстрировав неплохие результаты (свыше 97.5%) [12] на небольших изображениях в отсутствии мелких видимых деталей и естественного фона, не позволил добиться сравнимых с “переносом обучения” показателей точности на реальных изображениях. В связи с этим выполнена модификация данного метода. “Перенос обучения” для большинства выбранных архитектур сверточных нейросетей показал более 97% средней точности, в то время как

модификация предложенного метода – 86%. При этом увеличение числа обучаемых параметров нейронной сети привело к падению показателей точности, а увеличение глубины модели, использующей ансамбли деревьев решений, — к увеличению точности классификации.

В дальнейшем предполагается исследовать направления для улучшения качества работы метода, основанного на построении иерархического представления исходных данных и обучении ансамблей деревьев решений. Также планируется рассмотреть решение задачи детектирования автомобилей на изображениях с использованием метода “бегущего окна” (sliding window). Данный метод позволяет свести задачу детектирования к классификации областей изображения, в которых потенциально могут находиться автомобили. Для классификации планируется использовать методы “переноса обучения”, которые показали наиболее перспективные результаты в ходе настоящего сравнения.

9. Благодарности

Работа выполнена при поддержке гранта РФФИ по теме «Исследование применимости методов глубокого обучения к решению задач компьютерного зрения на примере классификации изображений с большим числом категорий, детектирования пешеходов и автомобилей» (договор № НК 14-07-31269\15) в лаборатории «Информационные технологии» Института Информационных технологий, математики и механики ННГУ им. Н.И. Лобачевского.

Литература

- [1] Buch N., Orwell J., Velastin S.A. 3D Extended Histogram of Oriented Gradients (3DHOG) for Classification of Road Users in Urban Scenes // BMVC 2009.
- [2] Breiman L. Random Forests // Machine Learning. – 2001. – Vol. 45, No. 1. – P. 5-32.
- [3] Breiman L., Friedman J.H., Olshen R.A., Stone C.J. Classification and Regression Trees. – 1984.
- [4] Caffe URL: <http://caffe.berkeleyvision.org>.
- [5] Community Model Zoo URL: <https://github.com/BVLC/caffe/wiki/Model-Zoo>.
- [6] Computer Vision library OpenCV URL: <http://opencv.org>.
- [7] Deep Learning URL: <http://deeplearning.net>.
- [8] Deep forest algorithm implementation URL: https://github.com/druzhkov-paul/deep_forest.
- [9] Rob G.J., et al. Unsupervised Sub-categorization for Object Detection: Finding Cars from a Driving Vehicle // IEEE ICCV Workshops. – 2011. – P. 2077-2083.
- [10] Dong Zh., Wu Yu., Pei M., Jia Yu. Vehicle Type Classification Using Semi-Supervised Convolutional Neural Network // IEEE Transactions on Intelligent

- Transportation Systems. – 2012. – Vol. 16, Issue 4. – P. 2247-2256.
- [11] Druzhkov P.N., Kustikova V.D. A Survey of Deep Learning Methods and Software for Image Classification and Object Detection // PRIA. – 2016. – Vol. 26, No. 1. – P. 9-15.
- [12] Druzhkov P., Kustikova V. The Investigation of Deep Data Representations Based on Decision Tree Ensembles for Classification Problems // In the Suppl. Proc. of the 4th Int. Conference on AIST'2015. – 2015. – P. 52-55. – URL:<http://ceur-ws.org/Vol-1452>.
- [13] Friedman J. Greedy function approximation: the gradient boosting machine // Annals of Statistics. – 2001. – Vol. 29, No. 5. – P. 1189-1232.
- [14] Hastie T., et al. The elements of statistical learning. Data mining, inference and prediction. – Springer, 2001. – 745 p.
- [15] Hu A., et al. Deep Boltzmann Machines based Vehicle Recognition // In Proceedings of the 26th CCDC. – 2014. – P. 3033-3038.
- [16] ImageNet Database URL: <http://image-net.org>.
- [17] KITTI URL: <http://www.cvlibs.net/publications/Geiger2012CVPR.pdf>.
- [18] Krizhevsky A., Sutskever I., Hinton G.E. ImageNet Classification with Deep Convolutional Neural Networks // Adv. in Neural Inf. Processing Systems 25. – 2012. – P. 1097-1105. – URL: <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>.
- [19] Kruchinin D., et al. Comparison of Deep Learning Libraries on the Problem of Handwritten Digit Classification // CCIS. – 2015. – Vol. 542. – P.383-393.
- [20] Large-scale and Distributed Gradient Boosting (GBDT, GBRT or GBM) Library URL: <https://github.com/dmlc/xgboost>.
- [21] LeCun Y., et al. Convolutional networks and applications in vision // In the Proc. of 2010 IEEE Int. Symposium on Circuits and Systems (ISCAS). – 2010. – P. 253-256. URL: <http://yann.lecun.com/exdb/publis/pdf/lecun-iscas-10.pdf>.
- [22] Ma X., Eric W., Grimson L. Edge-based rich representation for vehicle classification // In Proceedings of the 10th IEEE ICCV. – 2005. – Vol. 2. – P. 1185-1192.
- [23] Results of Large Scale Visual Recognition Challenge 2014 (ILSVRC2014) URL: <http://image-net.org/challenges/LSVRC/2014/results>.
- [24] Scikit-learn URL:<http://scikit-learn.org>.
- [25] Simonyan K., Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition // CoRR. – 2014. – URL:<http://arxiv.org/abs/1409.1556>.
- [26] Sivaraman S., Trivedi M. A general active-learning framework for on-road vehicle recognition and tracking // IEEE Transactions on Intelligent Transportation systems, 2010. – Vol. 2, No. 2. – P. 267-276.
- [27] Stanford Dataset URL: http://web.eecs.umich.edu/~jiadeng/paper/KrauseStarkDengFei-Fei_3DRR13.pdf.
- [28] Sun Z., Bebis G., Miller R. On-road vehicle detection using Gabor filters and support vector machines // Digital Signal Processing. – 2002. – Vol. 2. – P. 1019-1022.
- [29] Szegedy C., et al. Going Deeper with Convolutions // In the Proc. of the IEEE Conference on CVPR. – 2015. – URL: <http://arxiv.org/abs/1409.4842>.
- [30] Szeliski R. Computer Vision: Algorithms and Applications. – Springer, 2010. – 979 p.
- [31] The PASCAL Visual Object Classes Challenge 2007 URL: <http://host.robots.ox.ac.uk/pascal/VOC/voc2007/index.html>.
- [32] Yang L., et al. A Large-Scale Car Dataset for Fine-Grained Categorization and Verification // In Proc. of the IEEE Conf. on CVPR. – 2015. – P.3973-3981.
- [33] Шапиро Л., Стокман Дж. Компьютерное зрение. – М.: Бином. Лаборатория знаний, 2006. – 752 с.