

Государственное образовательное учреждение высшего профессионального
образования

«Московский государственный технический
университет имени Н.Э. Баумана»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКИ И СИСТЕМ УПРАВЛЕНИЯ
КАФЕДРА ТЕОРЕТИЧЕСКОЙ ИНФОРМАТИКИ И КОМПЬЮТЕРНЫХ
ТЕХНОЛОГИЙ

Отчет о проведенной НИРС

Студент _____ Батусов П. В.

Научный руководитель _____ Вишняков И. Э.

Москва 2016

Содержание

1	Существующие подходы определения и классификации автомобилей	2
1.1	Использование трехмерной структуры и нейросети	2
1.2	Deep Belief Network	2
1.3	Использование цепочки классификаторов	3
1.4	Байесовские сети	3
1.5	Car-Rec	4
2	Классификация без явного определения признаков	4
2.1	Сверточные нейронные сети	5
2.2	Глубокие сверточные нейронные сети	5
2.3	Плиточные сверточные нейронные сети	6
3	Обучение нейросетей	6
3.1	Эффективное обратное распространение ошибки	6
3.2	Использование графического процессора	7
3.3	Дообучение сети	7

1 Существующие подходы определения и классификации автомобилей

1.1 Использование трехмерной структуры и нейросети [1]

Данный метод основан на представлении автомобилей в виде полигональной модели, по которой происходит выделение признаков, передаваемых на вход нейросети. Трехмерная структура описания автомобиля разделена на восемь частей, в каждой из которых выделяется опорная вершина. В качестве признаков используется расстояние между каждой парой опорных вершин. Дополнительно рассматриваются параметры колес автомобиля, их радиус и положение.

Авторы работ использовали трехслойную, полносвязную нейронную сеть с 30 входами (количество признаков) и 120 выходами (6×20 , 6 — количество классов, 20 — число выходных нейронов для каждого класса). Результат определялся поиском максимального значения в выходном слое.

При обучении сети использовался метод обратного распространения ошибки. В качестве ошибочной, для улучшения сходимости, использовалась экспоненциальная функция вместо квадратичной.

Обучающая выборка состояла из 500 изображений с дорожных камер. Классификация производилась на большие/небольшие грузовики и автомобили. Тестовая выборка включала в себя еще 300 изображений, из которых правильно классифицировано 91%, ошибочно 4% и в 5% случаев система не смогла определить автомобиль на изображении.

1.2 Deep Belief Network [2]

Развитие идеи использования рекуррентных нейронных сетей, которые тяжело обучать из-за наличия обратных связей. Рассматривается архитектура при которой внутри скрытых слоев используется ограниченная машина Больцмана.

Авторы рассматривают задачу определения автомобиля по фотографии сзади (как с камеры авторегистратора). Входной слой имеет размерность соответствующую размерности изображения (признаки — пиксели). Выходов

два — классы автомобиль и не автомобиль.

Тестирование происходило на 735 изображениях, рассматривались модели с одним, двумя и тремя скрытыми слоями. Наилучшая точность была получена при использовании двух скрытых слоев — более 96%. Также была произведена оценка точности для других алгоритмов: искусственных и сверточных нейросетей, k ближайших соседей и метода опорных векторов. Среди них наибольшая точность результата получилась при использовании сверточных нейронных сетей, почти 95%.

1.3 Использование цепочки классификаторов

В работе [3] ставилась задача определения боковых стоек автомобиля. Решение задачи представляло собой многослойную схему классификации, от общего к частному. Такой подход называют усилением простых классификаторов (boosting classifier). Первые слои предназначены для обработки входного изображения, определения направления движения, колес, выделение боковой части автомобиля и только после этого поиск боковых стоек на основе гистограммы ориентированных градиентов и геометрических моделей.

Обучение системы производилось на изображениях автомобильного потока, в тестовую выборку также включены негативные примеры не содержащие транспортных средств. Тренировочное множество состоит из 100000 изображений без автомобилей и 4000 с автомобилями. Тестовая выборка содержит 1000 положительных и 200000 негативных примеров. Итоговая точность 90% для положительных примеров и 99% для негативных.

1.4 Байесовские сети

Имея возможность выделения признаков для дальнейшего решения задачи классификации можно воспользоваться байесовской сетью. Такой подход использовался в работе выделения автомобилей на фотоаэроснимках низкого разрешения [4]. Основная проблема при решении этой задачи — высокое количество шумов: тени, солнечные блики, кроны деревьев и прочее.

Авторы предложили метод, основанный на использовании дополнительной известной информации о времени и месте съемки. Эти данные и вы-

деленные признаки объектов вместе попадают в байесовскую сеть, которая производит отсев таких неправильно определенных кандидатов как тени от настоящих автомобилей и объектов находящихся не на дорожном полотне.

1.5 Car-Rec [5]

Еще одно предложенное решение для задачи классификации автомобилей, заключающееся в выделении признаков и их обработке с использованием деревьев поиска по заготовленной базе. Для определения регионов интереса на изображении используется алгоритм SURF [6], основанный на интегральном представлении входной картинки. Разработанная система имеет точность более 90%.

2 Классификация без явного определения признаков

Практически все перечисленные выше подходы для решения задачи определения или классификации автомобилей занимались выделением опорных признаков по которым строился ответ. В пункте 1.1 использовалась заготовленная трехмерная модель, описывающая транспортное средство, а в 1.4 рассматривались физические известные факторы предметной области.

Однако, не всегда имеется возможность применить данный подход, так как в общем случае не понятно по какому принципу выбирать признаки для классификации. Кроме этого, не известно как с течением времени будет развиваться предметная область и какие еще признаки необходимо зарезервировать для дополнительных классов.

При использовании нейросетей, признаки для классификации автоматически вырабатываются при обучении модели. В случае необходимости добавить новые классы в классификатор, производят его переобучение, в результате чего выводятся новые признаки. Также возможно произвести более быструю настройку, скорректировав весовые коэффициенты без перестроения признаков (дообучение сети).

2.1 Сверточные нейронные сети

В работе [7] рассматривалась задача распознавания номерных табличек на домах. Использовалась традиционная архитектура сверточной нейросети: первый сверточный слой выделял 16 карт признаков со сверткой 5×5 , второй — 512, со сверткой 7×7 , два слоя классификации по 20 нейронов каждый. Применялся многоступенчатый подход выделения признаков (слоям классификации были доступны все предыдущие карты признаков). Описанная модель реализована на фреймворке EBLearn [8].

Был произведен подбор оптимального параметра степени свертки для алгоритма выделения признаков Lp-Pooling. Наилучшая точность получена при использовании коэффициента $p = 4$. Кроме этого, перед передачей изображения в сеть, вокруг него достраивалась рамка из двух пикселей с нулевым значением, чтобы отцентрировать первый этап свертки рамкой 5×5 на границе рассматриваемой картинки.

Набор данных состоял из трех частей: обучающей, тестовой и дополнительной выборках. Дополнительная выборка состояла из большого количества легких для классификации изображений, а обучающая содержала небольшое число более сложных примеров. Обучение происходило на 6000 примеров, выбранных случайным образом из обучающего множества ($2/3$) и из дополнительного ($1/3$). В результате, при тестировании была получена точность более 95%.

2.2 Глубокие сверточные нейронные сети

Один из вопросов систем распознавания и классификации изображений — на сколько большую предметную область можно охватить используя автоматические методы. В рамках состязания ILSVRC проводится попытка классифицировать огромный тестовый набор изображений (более миллиона) по тысяче категорий.

Авторы работы [9] занимались созданием одной из самых больших сверточных нейросетей для участия в ILSVRC-2012. Она состояла из 5 сверточных и 3 полносвязных слоев, включающих 60 миллионов параметров и 650 тысяч нейронов. В работе говорилось о необходимости и важности большой размерности сети, так как при попытке избавиться хотя бы от одного слоя

происходило сильное падение качества классификатора.

Для уменьшения вероятности переобучения использовался подход выбывающих нейронов [10]. В качестве активационной функции была выбрана модель параметрического выпрямляемого элемента ReLU [11]. При тестировании по мере оценки ошибки top-5 error была получена точность 84.7%.

На соревнованиях ILSVRC-2014 была представлена еще большая сеть, включающая 27 слоев [12]. Команде GoogLeNet получилось добиться результатов в 6.67% ошибок.

2.3 Плиточные сверточные нейронные сети

Сверточные нейронные сети хорошо применяются для задач классификации и распознавания объектов, однако подход использования разделяемых весов может помешать в процессе обучения выделить некоторые инвариантные признаки. В работе [13] предлагался метод при котором разделяемые веса чередовались с некоторым шагом k . Так при $k = 1$ получалась обычная сверточная нейронная сеть, а при увеличении этого параметра строилась очередная модель, которая является компромиссом между способностью выявления инвариантностей изображения и сложностью обучения системы.

Тестирование производилось на наборе данных CIFAR-10, состоящем из 50000 обучающих изображений и 10000 тестовых, которые необходимо классифицировать по 10 категориям. Полученная точность метода при выбранном параметре $k = 2$ более 73%.

3 Обучение нейросетей

3.1 Эффективное обратное распространение ошибки

Обратное распространение ошибки — это один из классических методов обучения нейронных сетей, так как данный подход является простым при реализации, не очень вычислительно сложным и почти всегда работающим. Однако, его применение для получения действительно высоких результатов требует подбора большого числа параметров. В работе [14] рассматривались

применяемые исследователями уловки для повышения результатов классификации при обучении сети и их обоснование.

Основные проблемы алгоритма обратного распространения ошибки — это низкая скорость обучения глубоких сетей и скатывание в точки локальных минимумов задачи оптимизации. Для ускорения без существенной потери точности можно воспользоваться стохастическим градиентным спуском. Кроме этого, рассматривались подходы нормализации входных переменных при обучении, перемешивания обучающей выборки, выборе активационной функции и начальной инициализации весов.

3.2 Использование графического процессора

Использование обучения без учителя позволяет обрабатывать огромные неразмеченные корпуса информации и автоматически устанавливать общие взаимосвязи предметной области. Такой подход удобен тем что не нужно собирать обучающую и тестовую выборки, которые приходится размечать вручную. С другой стороны, при таком обучении требуется выполнить на много больше вычислительных операций, из-за чего приходится отказываться от стандартного пути использования ЦПУ. В статье [15] рассказывалось об обучении модели из пункта 1.2 на графическом процессоре.

В тестировании использовалась карточка Nvidia GeForce GTX 280 с объемом оперативной памяти 1Гбайт, программирование под CUDA SDK. Производилось обучение четырехслойной DBN, 100 миллионов настраиваемых параметров. Основная идея заключалась в минимизации числа операций между ОЗУ и локальной памятью графического процессора. В итоге был получен прирост производительности в 5-15 раз.

3.3 Дообучение сети

Обучение нейросетей требует много времени: подготовка данных для обучающей выборки и тестирования, решение многомерной задачи оптимизации. При необходимости добавить в классификатор новый подкласс рассматриваемой предметной области не всегда есть возможность выполнить полное переобучение сети. Однако, если это действительно подкласс, то можно попробо-

вать произвести корректировку весов в полносвязных слоях без перестроения признаков.

Такой эксперимент был проведен в работе [16]. После обучения сети на широком классе задачи распознавания растений ее последний классификационный слой был инициализирован случайными весами, а затем произведено дообучение с использованием алгоритма стохастического градиентного спуска на более узком подклассе. Скорость обучения задавалась для каждого слоя в отдельности, больше всего внимания уделялось на полносвязные слои классификации.

Вначале наблюдался резкий подъем точности, однако кривая обучения очень быстро вышла на плато в 60-65%. Сеть выделяла тысячу различных классов. Все вычисления производились с использованием фреймворка Caffe на графическом процессоре NVIDIA Titan Z.

Список литературы

- [1] *Wu, W.* A method of vehicle classification using models and neural networks / W. Wu, Z. QiSen, W. Mingjun // Vehicular Technology Conference, 2001. VTC 2001 Spring. IEEE VTS 53rd / IEEE. — Vol. 4. — 2001. — Pp. 3022–3026.
- [2] *Wang, H.* A vehicle detection algorithm based on deep belief network / H. Wang, Y. Cai, L. Chen // *The scientific world journal*. — 2014. — Vol. 2014.
- [3] *Brehar, R.* Pillars detection for side viewed vehicles / R. Brehar, S. Nedevschi, L. Dăian // Intelligent Computer Communication and Processing (ICCP), 2010 IEEE International Conference on / IEEE. — 2010. — Pp. 247–250.
- [4] *Zhao, T.* Car detection in low resolution aerial images / T. Zhao, R. Nevatia // *Image and Vision Computing*. — 2003. — Vol. 21, no. 8. — Pp. 693–703.
- [5] *Jang, D. M.* Car-rec: A real time car recognition system / D. M. Jang, M. Turk // applications of computer vision (WACV), 2011 IEEE Workshop on / IEEE. — 2011. — Pp. 599–605.
- [6] Speeded-up robust features (surf) / H. Bay, A. Ess, T. Tuytelaars, L. Van Gool // *Computer vision and image understanding*. — 2008. — Vol. 110, no. 3. — Pp. 346–359.
- [7] *Sermanet, P.* Convolutional neural networks applied to house numbers digit classification / P. Sermanet, S. Chintala, Y. LeCun // Pattern Recognition (ICPR), 2012 21st International Conference on / IEEE. — 2012. — Pp. 3288–3291.
- [8] *Sermanet, P.* Eblearn: Open-source energy-based learning in c++ / P. Sermanet, K. Kavukcuoglu, Y. LeCun // Tools with Artificial Intelligence, 2009. ICTAI'09. 21st International Conference on / IEEE. — 2009. — Pp. 693–697.

- [9] *Krizhevsky, A.* Imagenet classification with deep convolutional neural networks / A. Krizhevsky, I. Sutskever, G. E. Hinton // Advances in neural information processing systems. — 2012. — Pp. 1097–1105.
- [10] Dropout: A simple way to prevent neural networks from overfitting / N. Srivastava, G. Hinton, A. Krizhevsky et al. // *The Journal of Machine Learning Research*. — 2014. — Vol. 15, no. 1. — Pp. 1929–1958.
- [11] *Dahl, G. E.* Improving deep neural networks for lvcsr using rectified linear units and dropout / G. E. Dahl, T. N. Sainath, G. E. Hinton // Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on / IEEE. — 2013. — Pp. 8609–8613.
- [12] Going deeper with convolutions / C. Szegedy, W. Liu, Y. Jia et al. // *arXiv preprint arXiv:1409.4842*. — 2014.
- [13] Tiled convolutional neural networks / J. Ngiam, Z. Chen, D. Chia et al. // Advances in Neural Information Processing Systems. — 2010. — Pp. 1279–1287.
- [14] Efficient backprop / Y. A. LeCun, L. Bottou, G. B. Orr, K.-R. Müller // Neural networks: Tricks of the trade. — Springer, 2012. — Pp. 9–48.
- [15] *Raina, R.* Large-scale deep unsupervised learning using graphics processors / R. Raina, A. Madhavan, A. Y. Ng // Proceedings of the 26th annual international conference on machine learning / ACM. — 2009. — Pp. 873–880.
- [16] *Reyes, A. K.* Fine-tuning deep convolutional networks for plant recognition / A. K. Reyes, J. C. Caicedo, J. E. Camargo // Working notes of CLEF 2015 conference. — 2015.