УДК 004.8

# ИССЛЕДОВАНИЕ ВЛИЯНИЯ МЕТОДА СРАВНЕНИЯ КАНАЛОВ НА ЭФФЕКТИВНОСТЬ АЛГОРИТМОВ ПОКАНАЛЬНОГО ПРОРЕЖИВАНИЯ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ

**Чернышов Николай Дмитриевич[1], Буряк Дмитрий Юрьевич[2]**

*[1]Студент;*
*Московский государственный университет им. М. В. Ломоносова;*
*Россия, 119991, г. Москва, ул. Ленинские горы, д. 1;*
*e-mail: chernyshovnd@yandex.ru.*

*[2]Кандидат физико-математических наук, доцент;*
*Московский государственный университет им. М. В. Ломоносова;*
*Россия, 119991, г. Москва, ул. Ленинские горы, д. 1;*
*e-mail: dyb04@yandex.ru.*

*Работа посвящена решению задачи прореживания нейронной сети, целью которой является уменьшение количества параметров сети при сохранении высокой точности ее работы на тестовой выборке. Проводится обзор существующих методов прореживания, которые принадлежат к разным группам подходов в зависимости от их свойств, таких как зависимость от входных данных и необходимость рассмотрения каналов сети в совокупности. Для решения поставленной задачи предлагаются подходы к сравнению каналов сети, на основе результатов которого происходит выбор удаляемых параметров. Подходы основаны на выборе эффективной метрики оценки близости каналов и кластеризации каналов. Описываются методы прореживания с использованием предложенных подходов. Рассматриваются детали программной реализации методов. Приводятся результаты экспериментального исследования эффективности предложенных методов.*

Ключевые слова: прореживание, глубокое обучение, сверточные нейронные сети, сравнение каналов, корреляция.

**Для цитирования:**

Чернышов Н. Д., Буряк Д. Ю. Исследование влияния метода сравнения каналов на эффективность алгоритмов поканального прореживания сверточных нейронных сетей // Системный анализ в науке и образовании: сетевое научное издание. 2025. № 1. С. 16-22. EDN: JFTUOQ. URL: https://sanse.ru/index.php/sanse/article/view/648.

# RESEARCH INTO IMPACT OF CHANNEL COMPARISON METHOD ON EFFICIENCY OF ALGORITHMS FOR CHANNEL-WISE PRUNING OF CONVOLUTIONAL NEURAL NETWORKS

**Chernyshov Nikolai D.[1], Buryak Dmitrii Yu.[2]**

*[1]Student;*
*Lomonosov Moscow State University;*
*1 Leninskiye Gory, Moscow, 119991, Russia;*
*e-mail: chernyshovnd@yandex.ru.*

*[2]PhD in Physical and Mathematical Sciences, associate professor;*
*Lomonosov Moscow State University;*
*1 Leninskiye Gory, Moscow, 119991, Russia;*
*e-mail: dyb04@yandex.ru.*

*The paper is devoted to solving the problem of pruning a neural network, the purpose of which is to reduce the number of network parameters while maintaining its high accuracy on test data. A review is carried out of existing methods for pruning, which belong to different groups of approaches depending on their characteristics, such as dependence on input data and the need to consider network channels collectively. To solve this problem, approaches are proposed to compare network channels, based on the results of which parameters are selected to be removed. The approaches are based on the selection of an effective metric for assessing channel proximity and channel clustering. Pruning methods using the proposed approaches are described. The details of the software implementation of the methods are considered. The results of an experimental study of the efficiency of the proposed methods are presented.*

Keywords: pruning, deep learning, convolutional neural networks, channel comparison, correlation.

**For citation:**
Chernyshov N. D., Buryak D. Yu. Research into impact of channel comparison method on efficiency of algorithms for channel-wise pruning of convolutional neural networks. *System analysis in science and education,* 2025;(1):16-22 (in Russ). EDN: JFTUOQ. Available from: https://sanse.ru/index.php/sanse/article/view/648.

## *Введение*

Современные сверточные нейронные сети становятся все более востребованными в различных сферах, однако их архитектуры зачастую имеют огромное количество параметров, что создает высокие требования к памяти и вычислительным ресурсам. Это затрудняет использование этих моделей на устройствах с ограниченными ресурсами, таких как, например, смартфоны и планшеты. Помимо того, широко известно, что нейронные сети имеют много излишних параметров, которые почти не влияют на результат работы нейронной сети. Поэтому методы прореживания сетей становятся особенно актуальными.

Прореживание предобученной нейронной сети подразумевает уменьшение числа параметров путем удаления наименее значимых из них. Оно позволяет более эффективно использовать имеющиеся ресурсы, сохраняя при этом высокую эффективность работы сети. Также оно позволяет повысить скорость работы сети и её энергоэффективность.

В ходе прореживания выявляются и удаляются из сети параметры, которые можно считать наименее важными по определенному критерию, зависящему от выбранного метода прореживания. После этого модель обычно проходит этап дообучения для компенсации потери эффективности, которая могла произойти в процессе удаления параметров.

Существует множество методов для прореживания сверточных нейронных сетей, и их эффективность может значительно варьироваться [1]. Поэтому одной из актуальных задач является поиск наиболее эффективных способов прореживания. Решение этой проблемы можно достичь через выдвижение гипотез для улучшения существующих методов и проведение сравнительных исследований их эффективности в одинаковых условиях, что и является целью данной работы.

## *1. Обзор существующих методов прореживания*

Прореживание делится на структурированное и неструктурированное [2]. Неструктурированное прореживание состоит в удалении отдельных параметров сети, вообще говоря, расположенных в независимом порядке относительно друг друга (рис. 1). На практике неструктурированное прореживание позволяет получить преимущество в работе сети лишь при очень большом проценте удаленных параметров в связи с особенностями работы с разреженными матрицами, используемыми для хранения параметров в этом случае. Если удалять столь значительную часть параметров, то с большой вероятностью способность сети к аппроксимации значительно снизится, что приведет к сильному снижению качества полученного решения. Именно поэтому наиболее актуальны методы структурированного прореживания.

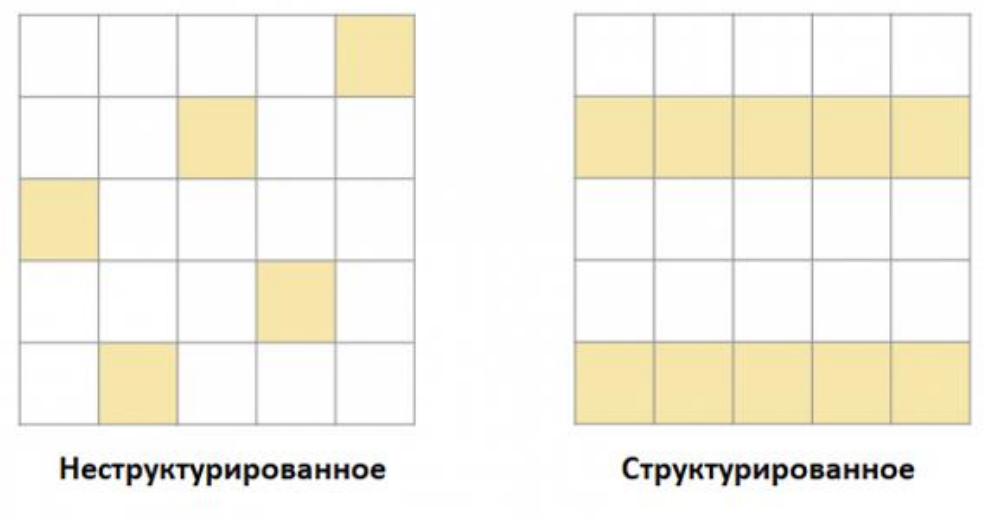


*Рис. 1. Пример выбора удаляемых параметров при структурированном и неструктурированном прореживании. Желтым цветом отмечены удаляемые параметры*

При структурированном прореживании удаление параметров происходит на уровне целых структурных блоков сети (рис. 1), например, удаление целых каналов или фильтров. Такие методы позволяют проводить прореживание быстрее и не требуют затратной работы со структурами хранения весов, такими, как разреженные матрицы. Наиболее актуально поканальное прореживание (Рис. 2), то есть прореживание с удалением параметров, соответствующих целым каналам слоя сверточной нейронной сети. Поэтому было решено исследовать именно методы поканального прореживания.

Существует множество методов поканального прореживания сверточной нейронной сети. Рассмотрим некоторые из них.

Прореживание может зависеть или не зависеть от входных данных сети. Прореживание, основанное на величине параметров [3] – один из классических методов прореживания, не зависящих от данных. В данном случае удаляются те параметры сети, величина которых меньше всего, будь то отдельные веса, нейроны, каналы или фильтры. При этом в каждом случае под величиной понимаются разные значения, например, абсолютное значение веса при удалении отдельных весов или норма тензора параметров фильтра при прореживании фильтров. Другой метод, не зависящий от данных – случайное прореживание [4]. Параметры сети, подлежащие удалению, выбираются случайно, но эффективность этого способа сравнима с другими методами прореживания. *Optimal Brain Damage* [5] – метод, зависящий от данных. В нём используется матрица Гессе, составленная из вторых производных функции потерь. Удаляется тот параметр, которому соответствует наименьшее увеличение функции потерь. Недостатками метода являются высокие затраты на вычисление матрицы Гессе и дополнительные требования на функцию потерь и состояние модели перед прореживанием.

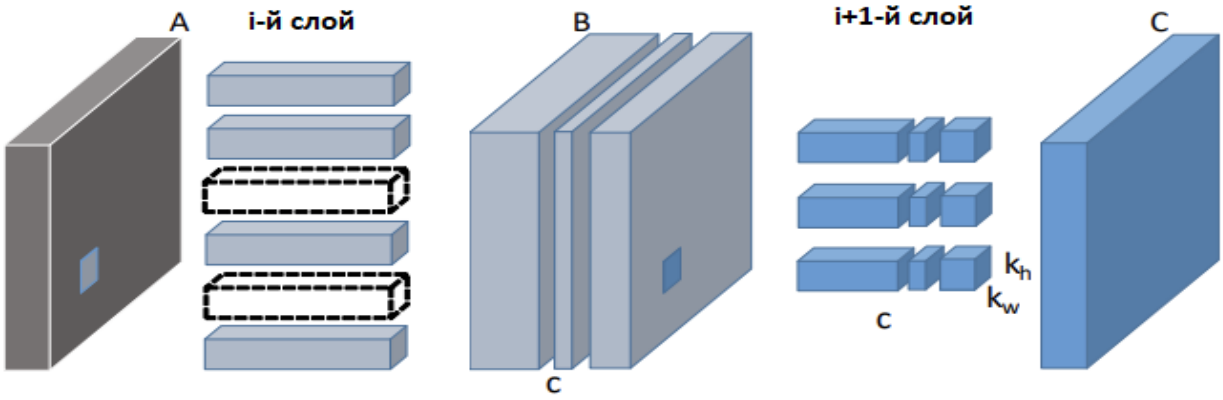


*Рис. 2. Поканальное прореживание сверточной нейронной сети [6]. A, B, C – тензоры на входе i-го слоя сети, на выходе i-го слоя сети и на выходе (i+1)-го слоя сети соответственно; c – число*

*каналов входного тензора слоя; $k_h$, $k_w$ – размеры ядра свертки. Из (i+1)-го слоя удаляется часть каналов и впоследствии можно удалить и часть фильтров i-го слоя, которые им соответствуют (отмечены пунктиром). Для упрощения на рисунке изображены только сверточные слои сети*

Прореживание с использованием *LASSO*-регрессии [6] и *ThiNet* [7] – два метода, в которых для уменьшения ошибки, вызываемой удалением части параметров, используется метод наименьших квадратов для реконструирования выходного тензора слоя. Первый из них использует для определения множества удаляемых каналов *LASSO*-регрессию, то есть регрессию с *L1*-регуляризацией, второй – жадный алгоритм.

В работе было решено рассматривать только методы с использованием обучающих данных.

Ещё один способ классифицировать методы прореживания каналов — разделять их на *intra-channel* и *inter-channel*. *Intra-channel* – методы, в которых каждый канал рассматривается по отдельности. Например, для каждого канала рассчитывается некоторая метрика, а затем значения метрики для разных каналов сравниваются, после чего производится выбор удаляемых каналов. Примерами таких методов могут быть метод с жадным алгоритмом [8] и *Oracle* [9]. *Inter-channel* – методы, в которых каналы рассматриваются в совокупности с другими каналами. Например, в случае метода *CHIP* [10] для каждого канала рассчитывается величина *Channel Independence*, показывающая, насколько сильно один канал зависит от других, а в случае метода прореживания с корреляцией [11] выбираются и рассматриваются пары каналов, у которых наиболее высокий коэффициент корреляции.

В процессе поиска наиболее эффективного метода прореживания можно не только использовать уже существующие методы, но и предлагать новые подходы, способные повысить эффективность прореживания. Среди таких подходов в данной работе предлагается рассмотреть использование кластеризации каналов вместо их попарного сравнения и оптимизацию расчета метрики оценки схожести каналов.

Эффективность методов прореживания оценивается по следующим критериям:

- количество параметров у прореженной сети;
- точность прореженной сети на тестовой выборке;
- вычислительная сложность прореженной сети.

От количества параметров сети зависит объем памяти, занимаемый сетью. Очевидно, что при прочих равных значениях метрик стоит выбрать ту сеть, которая занимает меньший объем памяти. Под точностью сети подразумевается процент изображений из тестовой выборки, которым сеть правильно предсказала класс. Эта метрика позволяет оценить, насколько хорошо сеть решает задачу (в случае задачи классификации).

Вычислительную сложность сети можно измерять в *MACs* (сокращение от ”*multiplication accumulation*” – «умножение и накапливание»). Если вычислительная сложность сети уменьшится, то станет возможным её запуск на устройствах с меньшими вычислительными ресурсами.

## *2. Предложенные подходы*

В данной работе предлагаются подходы к повышению эффективности методов прореживания за счет оптимизации сравнения каналов, выполняемого в процессе прореживания для отбора удаляемых параметров сети.

Первый подход – использование кластеризации вместо попарного рассмотрения каналов. Обычно [1] метрика схожести каналов рассчитывается для каждой существующей пары каналов и затем выбираются те пары, для которых метрика схожести показывает наибольшие значения, то есть те пары, в которых каналы больше всего похожи друг на друга. Если не рассматривать каналы попарно, а проводить их кластеризацию, то будет возможно рассмотреть все похожие друг на друга каналы в совокупности. Тогда можно будет удалять лишние каналы быстрее, а также будут исключены ситуации, когда для одного из нескольких похожих каналов не нашлось достаточно

похожей на него пары, так как все остальные каналы из числа похожих уже выбраны для других пар и, соответственно, не могут быть выбраны в пару с ещё одним каналом.

Второй подход – оптимизация расчета метрики схожести за счет уменьшения влияния шумов промежуточных активаций. При прореживании с использованием обучающих данных для выбора удаляемых из слоя сверточной сети каналов необходимо подать на вход сети батч входных данных и получить промежуточные активации, поступающие на вход прореживаемому слою. Мера близости любых двух каналов слоя сети рассчитывается как функция от соответствующих им каналов тензора промежуточных активаций. Этот подход позволяет сохранить информацию о каждом отдельном изображении в батче. Это может быть полезно, если имеет место значительное разнообразие в данных, поскольку он позволяет выявить зависимости, которые могут быть уникальными для отдельных изображений. Однако, этот метод также может быть подвержен шуму и выбросам из-за того что батч содержит изображения с различными характеристиками или если данные имеют высокую вариативность. Это может привести к тому, что оценка схожести каналов будет неточной. Перед расчетом меры схожести можно было бы усреднить тензор промежуточных активаций по батчу входных данных. Это позволило бы сгладить данные и уменьшить влияние выбросов или шумов и могло бы помочь лучше выявлять общие зависимости между каналами, которые присутствуют во всех изображениях батча, но при этом усредненный вектор может скрыть уникальные зависимости, которые могут быть важны для конкретных изображений. В данной работе вместо двух описанных способов предлагается следующий: для каждого элемента входного батча рассчитывается отдельный тензор промежуточной активации, и мера схожести для любой пары каналов слоя сети рассчитывается как функция от соответствующих каналов этого тензора. Таким образом рассчитывается мера схожести для каждой комбинации элемента входного батча и пары каналов слоя. Затем для каждой пары каналов итоговая мера схожести рассчитывается как среднее значений меры схожести этой пары по всем элементам батча. Такой подход позволит уменьшить влияние шумов, при этом учитывая влияние уникальных зависимостей.

Для дальнейшего проведения исследования с целью оценки эффективности предложенных подходов было решено использовать метод прореживания с корреляцией [11] с использованием обучающих данных. Производилось экспериментальное сравнение этого метода с его же модифицированными версиями, использующими предложенные подходы.

Далее приводится описание метода прореживания с корреляцией без использования предложенных подходов. Данный метод относится к группе *inter-channel*. Пусть необходимо удалить часть каналов из k-го слоя сети. Для этого нужно:

1) Из всех возможных пар каналов входных данных k-го слоя выбрать те пары (множество *pairs*), у которых наиболее высокий коэффициент корреляции.

2) Дообучить сеть таким образом, чтобы в каждой паре два канала стали еще более похожи, то есть чтобы их корреляция стала еще выше.

3) Удалить из слоя по 1 каналу из каждой пары.

Дообучение сети, упомянутое в пункте 2, достигается за счет добавления к используемой функции потерь слагаемого $L_{corr}$, рассчитываемого по формуле

$$L_{corr} = exp\left(-\sum c_{XY}\right), \quad (1)$$

где $c_{XY}$ – коэффициент корреляции между каналами $X$ и $Y$; $pair_i$ – i-я выбранная пара каналов, состоящая из $X$ и $Y$; $pairs$ – множество всех выбранных пар каналов $pair_i$.

Чтобы применить к данному методу прореживания предложенный подход с кластеризацией, вместо поиска пар с наиболее высокими коэффициентами корреляции нужно кластеризовать каналы с помощью иерархического восходящего метода кластеризации [12]. Для очередного объединения двух кластеров в один выбираются те два кластера, у которых коэффициент корреляции наибольший. Дообучение сети, описанное ранее во втором шаге алгоритма прореживания с корреляцией, проводится по тому же принципу, но вместо пар рассматриваются целые кластеры каналов. После дообучения сети для каждого выбранного кластера из сети удаляются все каналы, принадлежащие этому кластеру, кроме одного.

Применение к данному методу прореживания оптимизации расчета метрики близости за счет уменьшения влияния шумов промежуточных активаций производится согласно алгоритму, приведенному при вышеуказанном описании данного подхода. В случае метода прореживания с корреляцией под мерой схожести каналов необходимо понимать их коэффициент корреляции.

В экспериментальном исследовании было решено рассмотреть три варианта метода:

1) метод без модификаций;

2) метод с оптимизацией расчета метрики схожести каналов;

3) метод с обоими предложенными подходами.

## *4. Условия вычислительных экспериментов*

Для проведения экспериментального исследования было решено рассмотреть задачу классификации изображений *CIFAR-10*. Набор данных для этой задачи состоит из 50000 обучающих и 10000 тестовых цветных изображений размером 32 на 32 пикселя. Обучающая и тестовая выборка поровну делятся на 10 классов. Также было решено в качестве прореживаемой сверточной сети использовать модель *VGG-16*.

Прореживание проводилось послойно. Из каждого слоя удалялся равный процент каналов. Благодаря этому можно свести сравнение эффективности методов прореживания к сравнению показателей точности прореженной сети на тестовой выборке, так как при удалении равного процента каналов из каждого слоя все прореженные сети будут иметь одинаковое количество параметров и одинаковую вычислительную сложность, вне зависимости от выбранного метода прореживания.

Точность непрореженной сети на тестовой выборке составила 92.4%. После удаления из каждого внутреннего слоя сверточной сети 40 процентов каналов с помощью метода прореживания с корреляцией без предложенных модификаций точность прореженной сети после дообучения составила 85.1%. При том же проценте удаляемых каналов сеть, прореженная тем же методом с использованием подхода с оптимизацией расчета метрики близости каналов после дообучения показала точность 87.6%, а сеть, прореженная методом с использованием обоих предложенных подходов — 89.0%. Все сети после прореживания дообучались с одинаковыми гиперпараметрами.

## *5. Программная реализация*

Для реализации выбранных методов решения поставленной задачи были использованы язык *Python 3* и библиотека *PyTorch*, которая предоставляет обширные возможности для работы с нейронными сетями и поддерживает вычисления на графических процессорах. Проект выполнялся в среде *Google Colab*, которая обеспечивает доступ к графическому процессору *Nvidia Tesla T4*.

В ходе программной реализации были разработаны процедуры для дообучения нейронной сети и обработки наборов данных. Проведена модификация архитектуры предобученной сети *VGG-16* с последующим дообучением. Под модификацией подразумевается замена входного слоя сети и последнего полносвязного слоя сети на аналогичный слой с 10 нейронами для обеспечения совместимости сети с задачей классификации *CIFAR-10*. Также были реализованы вариации метода прореживания, являющиеся предметом исследования.

## *Заключение*

В работе проведен обзор существующих методов прореживания. На основе их анализа предложены два подхода к повышению эффективности прореживания сетей. Первый из них предполагает использование кластеризации каналов вместо их попарного рассмотрения, а второй описывает альтернативный способ расчета метрики близости каналов. Выбран и описан используемый метод прореживания. Зафиксированы условия экспериментального исследования эффективности предложенных подходов. Выполнена программная реализация, необходимая для выполнения вычислительных экспериментов. Их результаты показали эффективность предложенных

подходов, так как точность классификации сети на тестовой выборке при использовании данных подходов повысилась по сравнению с базовым методом прореживания при одинаковых экспериментальных условиях.

## *Список источников*


1. Pruning and quantization for deep neural network acceleration: A survey / T. Liang, J. Glossner, L. Wang [et al.] // Neurocomputing. – 2021. – Vol. 461. – Pp. 370–403. – DOI: https://doi.org/10.1016/J.NEUCOM.2021.07.045.
2. Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks / T. Hoefler, D. Alistarh, T. Ben-Nun [et al.] // Journal of Machine Learning Research. – 2021. – Vol. 22. – №. 241. – Pp. 1-124.
3. Pruning filters for efficient convnets / H. Li, A. Kadav, I. Durdanovic [et al.] // arXive.org e-Print archive. – 2016. – arXiv:1608.08710.
4. Recovering from random pruning: On the plasticity of deep convolutional neural networks / D. Mittal, S. Bhardwaj, M. M. Khapra, B. Ravindran // 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). – IEEE, 2018. – Pp. 848-857.
5. LeCun Y., Denker J., Solla S. Optimal brain damage // Advances in neural information processing systems. – 1989. – Vol. 2. – Pp. 598-605.
6. He Y., Zhang X., Sun J. Channel pruning for accelerating very deep neural networks // Proceedings of the IEEE international conference on computer vision. – 2017. – Pp. 1389-1397.
7. Luo J. H., Wu J., Lin W. Thinet: A filter level pruning method for deep neural network compression // Proceedings of the IEEE international conference on computer vision. – 2017. – Pp. 5058-5066.
8. Good subnetworks provably exist: Pruning via greedy forward selection / M. Ye, C. Gong, L. Nie [et al.] // International Conference on Machine Learning. – PMLR, 2020. – Pp. 10820-10830.
9. Approximated oracle filter pruning for destructive cnn width optimization / X. Ding, G. Ding, Y. Guo [et al.] //International Conference on Machine Learning. – PMLR, 2019. – Pp. 1607-1616.
10. Chip: Channel independence-based pruning for compact neural networks / Y. Sui, M. Yin, Y. Xie [et al.] // Advances in Neural Information Processing Systems. – 2021. – Vol. 34. – Pp. 24604-24616.
11. Leveraging filter correlations for deep model compression / P. Singh, V. K. Verma, P. Rai, V. P. Namboodiri // Proceedings of the IEEE/CVF Winter Conference on applications of computer vision. – 2020. – Pp. 835-844.
12. Comprehensive survey on hierarchical clustering algorithms and the recent developments / X. Ran, Y. Xi, Y. Lu [et al.] // Artificial Intelligence Review. – 2023. – Vol. 56. – №. 8. – Pp. 8219-8264.