УДК 004.89 **DOI 10.52928/2070-1624-2024-43-2-10-15**

ЭФФЕКТИВНОЕ УМЕНЬШЕНИЕ РАЗРЕШЕНИЯ ИЗОБРАЖЕНИЙ В НЕЙРОННЫХ СЕТЯХ НА ОСНОВЕ ВЕЙВЛЕТОВ

В. А. ВОРОБЕЙ, канд. физ.-мат. наук, доц. А. Э. МАЛЕВИЧ
(Белорусский государственный университет, Минск)

U. Varabei ORCID https://orcid.org/0009-0006-9604-8894
A. Malevich ORCID https://orcid.org/0000-0001-8716-8655

На основе дискретного вейвлет-преобразования реализованы несколько вариантов блоков для уменьшения разрешения изображений внутри моделей компьютерного зрения. Блоки протестированы с использованием моделей ResNetV2-50 и MobileNetV2 на наборе данных Flowers. При незначительном увеличении количества параметров и близком уровне качества удалось сократить количество эпох для сходимости на 34 %, а требования к видеопамяти на 18 %. Благодаря особенностям реализации предложенных блоков они могут быть использованы для замены слоев уменьшения разрешения изображений и в моделях для других задач с целью экономии вычислительных ресурсов и ускорения процесса тренировки. В разработанных блоках для вычисления вейвлет-преобразования используются стандартные операция сложения и умножения, что позволяет при необходимости легко экспортировать обученные модели в другие форматы.

Ключевые слова: *нейронные сети, глубокое обучение, вейвлеты, дискретное вейвлет-преобразование, классификация изображений.*

Введение. Для улучшения работы моделей компьютерного зрения в различных задачах было предложено большое количество сверточных архитектур, которые позволяют лучше извлекать признаки из входных изображений: Dense CNN [1], SE Networks [2], CSPNet [3] и др. Однако, несмотря на все разнообразие архитектур, пространственный размер изображений в нейронных сетях зачастую понижается при помощи субдискретизации (пулинга) или свертки с шагом больше 1 (иногда фильтры могут быть фиксированными). При использовании этих методов либо теряется часть информации, либо же она извлекается слишком простым способом, который не подстраивается под данные. Для устранения этих недостатков в данной работе предлагается для понижения разрешения изображений использовать вейвлеты: они естественным образом выделяют из входного сигнала признаки с высокими и низкими частотами (всего 4 комбинации), у каждой из которых сохраняется исходное количество каналов, а размер по ширине и высоте уменьшается в 2 раза, что соответствует стандартному уменьшению разрешения тензоров по ширине и высоте в нейронных сетях. Схема двумерного вейвлет-преобразования представлена на рисунке 1. При необходимости можно применить вейвлет-преобразование несколько раз, на каждом шаге понижая разрешение тензора в 2 раза. Существует ряд работ, в которых авторы для сохранения большего количества информации использовали вейвлеты похожим образом [4–6], однако зачастую в них из выходных компонент вейвлет-преобразования не выделяются новые признаки в рамках реализованных блоков, либо же это делается достаточно простым способом. В данной работе предложены варианты вейвлет-блоков, в которые добавлены механизмы внимания с обучаемыми параметрами, а также показано, как значительно увеличить вычислительную эффективность методов построения карт внимания в сверточных нейронных сетях.

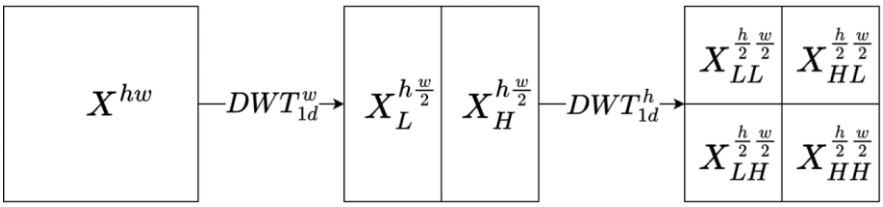


Рисунок 1. – Схема двумерного вейвлет-преобразования

Вейвлет-блоки. Для уменьшения разрешения изображений внутри моделей были реализованы 4 варианта блоков на основе вейвлет-преобразования, каждый из которых понижает разрешение по ширине и высоте в 2 раза.

11. WL + LL. Входной сигнал проходит через прямое дискретное вейвлет-преобразование и для дальнейших операций используется лишь компонента LL, которая содержит в себе признаки низкой частоты и может рассматриваться как уменьшенная версия исходной картинки. Блок не содержит дополнительных обучаемых параметров. Его схема представлена на рисунке 2 (V1).

12. WL + LHAttn. Ко входному сигналу применяется прямое дискретное вейвлет-преобразование, и компоненты LH и HL поэлементно перемножаются, а затем проходят через операцию softmax вдоль каждого пикселя. Полученная карта внимания применяется к выходу LL при помощи операций поэлементного сложения и умножения. Компонента HH никак не используется, поскольку зачастую в ней может содержаться достаточно много шума. Этот блок не содержит в себе дополнительных обучаемых параметров. Он был предложен в [6], его схема представлена на рисунке 2 (V2).

13. WL + HCBAM. Блок отличается от предыдущего варианта тем, что для построения карты внимания здесь используется модуль CBAM [7]: он применяется ко всем признакам с низкой частотой, которые предварительно проходят через свертку с ядром 1×1 (без последующей нормализации и активации). Операция свертки необходима для того, чтобы количество каналов совпадало с тем, что было у исходного тензора. Далее построенная карта внимания применяется к компоненте LL при помощи операций поэлементного сложения и умножения. Для уменьшения вычислительной сложности в модуле CBAM использовалось лишь внимание по каналам, а остальные параметры не менялись относительно тех, что установлены по умолчанию его авторами. Схема данного блока представлена на рисунке 2 (V3).

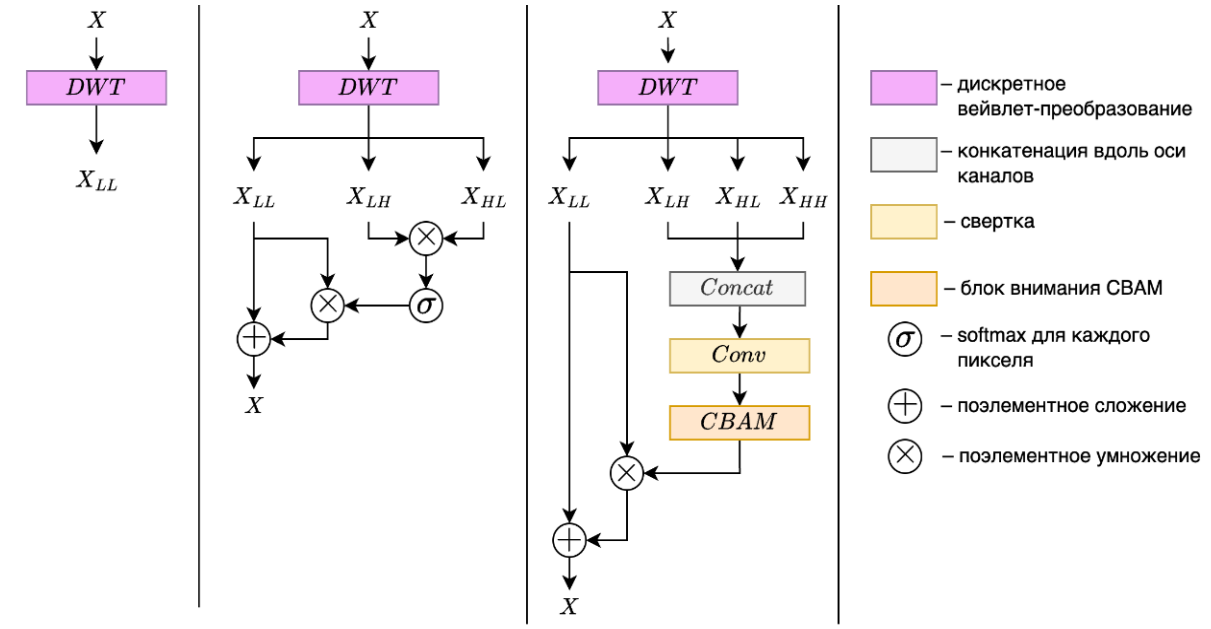
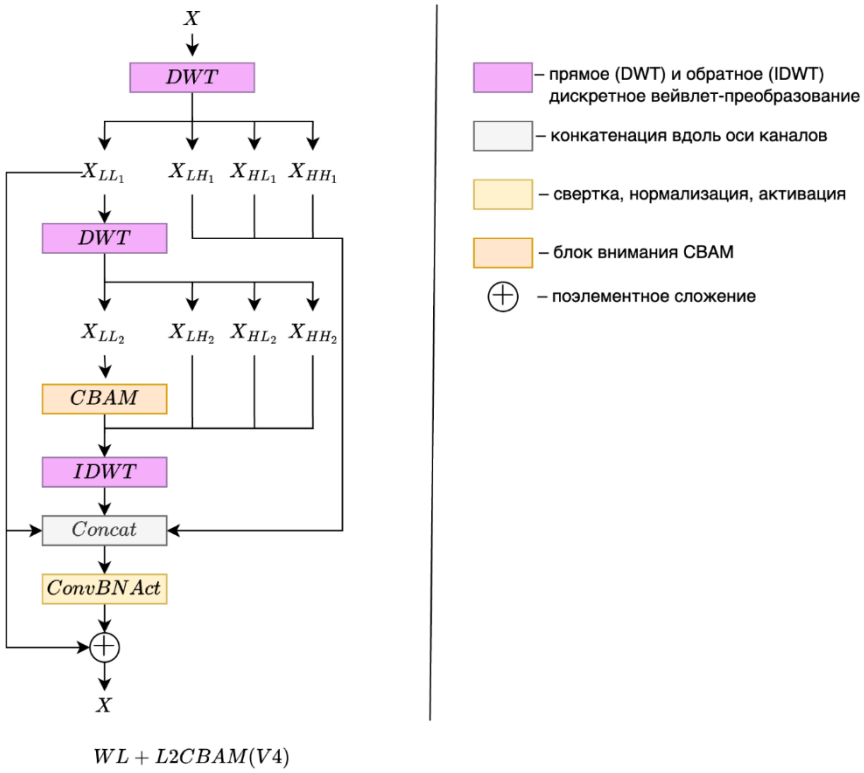


Рисунок 2. – Схема блоков WL + LL(V1), WL + LHAttn(V2), WL + HCBAM(V3)

14. WL + L2CBAM. У входного сигнала считается 2 уровня вейвлет-разложения, после чего к компоненте LL2 (2 – номер уровня разложения) применяется модуль CBAM, который используется в полной версии: внимание считается как по каналам, так и по пикселям. Параметр reduction устанавливался равным 2 для извлечения большего количества информации (стандартное значение – 16). Далее по модифицированной компоненте LL2,CBAM и деталям второго уровня при помощи обратного вейвлет-преобразования восстанавливалась компонента LL1,CBAM, которая объединялась со всеми признаками первого уровня разложения и пропускалась через слой свертки с ядром 1×1, пакетную нормализацию и активацию GELU. Полученный тензор по аналогии с предыдущими вариантами блоков использовался вместе с компонентой LL1. Применение модуля CBAM к разложению второго уровня призвано уменьшить вычислительную сложность всего блока, поскольку в таком случае внимание считается на тензоре, у которого 2 размерности уменьшены в 2 раза, а для восстановления исходной размерности используется легковесное обратное вейвлет-преобразование. Схема данного блока представлена на рисунке 3.



$WL + L2CBAM(V4)$

Рисунок 3. – Схема блока WL + L2CBAM(V4)

Модификация моделей. Для проверки работы реализованных блоков использовались две популярные модели компьютерного зрения: ResNetV2-50 [8] и MobileNetV2 [9]. Эти архитектуры хорошо подходили также и потому, что позволяли оценить эффект как на вычислительно тяжелых, так и на легковесных моделях.

Так как реализованные блоки уменьшают размеры тензора по ширине и высоте в 2 раза, то и в выбранных моделях заменялись соответствующие слои, за тем исключением, что первый шаг понижения разрешения во всех моделях оставался неизменным. Стоит отметить, что зачастую внутри моделей входное изображение по ширине и высоте уменьшается в 5 раз (как и в случае выбранных архитектур), соответственно, с учетом пропуска первого слоя получается, что замене подвергались только 4 набора слоев в каждой модели. За операцию уменьшения разрешения отвечают или слои субдискретизации (пулинга), или же свертки с шагом больше 1. В первом случае слой заменялся на один из реализованных блоков, а во втором же шаг свертки становился равен 1, а перед соответствующим слоем добавлялся блок на основе вейвлетов. Стоит отметить, что в модели MobileNetV2 не используются слои субдискретизации: изображения уменьшаются лишь при помощи сверток с шагом 2. Схема встраивания реализованных блоков показана на рисунке 4.

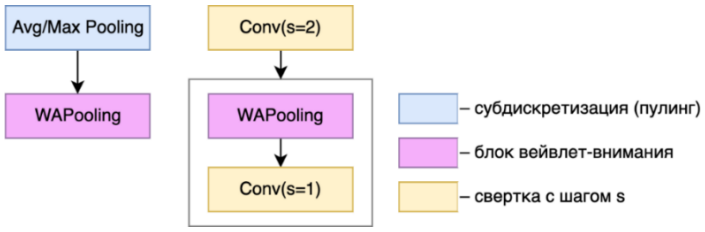


Рисунок 4. – Схема встраивания блоков в модель

Набор данных. Для проверки работы моделей было решено использовать набор данных Flowers[1], который состоит из 4317 изображений цветов 5 категорий: маргаритка, одуванчик, роза, подсолнух, тюльпан. Для тестового датасета выбиралось 20 % данных из каждой категории, а из оставшейся части также 20 % шло на валидацию. Таким образом, в тренировочном, валидационном и тестовом датасетах было по 2763, 691, 863 изображения соответственно.

[1] The TensorFlow Team. TensorFlow Flowers dataset [Электронный ресурс] // TensorFlow datasets. – URL: https://www.tensorflow.org/datasets/catalog/tf_flowers (дата обращения 16.06.2024).

Тренировочные изображения перед подачей в модель вдоль каждой стороны дополнялись случайным количеством нулей в диапазоне [0; 0,2] от ее размера, а затем у полученного изображения брался случайный патч размером 256×352. Также с вероятностью 0,5 изображения отражались относительно вертикальной оси. Изображения для теста и валидации сразу приводились к размеру 256×352 с сохранением соотношения сторон: меньшая из них с обоих краев дополнялась одинаковым количеством нулей. Примеры изображений из тестового датасета показаны на рисунке 5.

Рисунок 5. – Пример тестовых изображений

Обучение модели. В качестве вейвлетов использовалось семейство CDF-9/7, операции прямого и обратного преобразования были реализованы при помощи лифтинг-схемы, которую можно найти в [10]. Параметры были зафиксированы на константных значениях и не обучались.

Все модели использовали входное разрешение картинки 256×352 пикселя. Изображения в тренировочном наборе на старте каждой эпохи случайным образом перемешивались. К валидационным и тестовым изображением не применялись аугментации.

В качестве оптимизатора использовался SGD со скоростью обучения равной 0,01 × bs / 64 (bs – размер батча). Первые 5 эпох модели обучались со скоростью 0,005. Это было сделано для стабилизации обновления весов на ранних стадиях обучения. Скорость затухания весов устанавливалась равной 5e–4, а значения импульса – 0,9. Батчи состояли из 128 примеров, что было сделано для ускорения вычислений, а во избежание ухудшения сходимости данное число было учтено в скорости обучения.

Обучение могло длиться максимум 200 эпох с возможностью ранней остановки после 30, если значение метрики на валидационных данных не увеличивалось в течение этого времени.

Модели реализовывались при помощи библиотеки PyTorch. Обучение проходило на стенде с GPU Nvidia RTX 3090 24 Гб.

Результаты. В ходе исследования было проверено несколько вариантов блоков уменьшения разрешения изображений на основе вейвлетов для моделей ResNetV2-50 и MobileNetV2. Результаты представлены в таблицах 1 и 2 (Base – базовая модель). Для уменьшения влияния случайности каждый вариант модели обучался по 10 раз, а затем усреднялось количество эпох, которое прошло до момента сходимости моделей, а также значения метрики на валидационном и тестовых датасетах, полученные финальными моделями.

Таблица 1. – Результаты моделей на основе ResNetV2-50

Метод	Количество параметров (М)	Видеопамять (Мб)	Среднее количество эпох	Среднее значение метрики (валидация)	Среднее значение метрики (тест)
Base	23,530	9690	72,5	0,864062	0,831793
WL + LL (V1)	23,530	8646	74,1	0,858854	0,791466
WL + LHAttn (V2)	23,530	11275	75,6	0,854323	0,799264
WL + HCBAM (V3)	28,115	11315	79,3	0,850417	0,798738
WL + L2CBAM (V4)	32,337	12895	81,5	0,847500	0,792916

Замена слоев в архитектуре ResNetV2-50 отрицательно сказалась на результатах моделей: уменьшилось значение метрики, увеличилось количество параметров, а также скорость сходимости. Это может быть связано с большим количеством остаточных соединений в блоках моделей, а также с тем, что субдискретизация на основе максимального значения хорошо справляется с выделением признаков, оказывающих наибольшее влияние на итоговый выход сетей.

Таблица 2. – Результаты моделей на основе MobileNetV2

Метод	Количество параметров (М)	Видеопамять (Мб)	Среднее количество эпох	Среднее значение метрики (валидация)	Среднее значение метрики (тест)
Base	1,580	7363	73,4	0,863125	0,840739
WL + LL (V1)	1,580	6007	58,9	0,865260	0,829217
WL + LHAttn (V2)	1,580	6007	77,3	0,864427	0,835155
WL + HCBAM (V3)	1,615	6040	54,8	0,863437	0,829257
WL + L2CBAM (V4)	1,647	6069	53,2	0,859792	0,824236

Модели на основе MobileNetV2 хорошо отреагировали на замену операций уменьшения размера изображений блоками на основе вейвлетов: требования к видеопамяти снизились, для ряда блоков увеличилась скорость сходимости и лишь немного уменьшилось значение метрики на тестовой выборке, а на валидации фактически не изменилось. Такой результат можно связать с тем, что в отличие от ResNetV2-50, модель MobileNetV2 для уменьшения разрешения изображений использует лишь свертки с шагом 1, а субдискретизация по максимальному значению отсутствует.

Заключение. Было реализовано и протестировано несколько вариантов блоков уменьшения разрешения изображений в нейронных сетях на основе вейвлетов на примере моделей ResNetV2-50 и MobileNetV2 на наборе данных Flowers. Замена блоков в модели ResNetV2-50 привела к ухудшению качества и лишь увеличила ее сложность. Использование предложенных блоков в архитектуре MobileNetV2 наоборот хорошо себя показало: увеличилась скорость сходимости, уменьшился потребляемый размер видеопамяти при обучении, значение метрики качества на валидации не изменилось, а на тестовых данных оно лишь слегка снизилось (в пределах 1,651 % в худшем случае). Лучше всего себя показал вариант блока на основе механизма внимания CBAM: при его применении к высокочастотным частям вейвлет-разложения 1-го уровня требования к видеопамяти снизились на 18 %, количество эпох для сходимости уменьшилось на 34 %, количество параметров увеличилось лишь на 2,2 %, а значение метрик на тесте упало на 1,15 %. Благодаря простоте реализации предложенный метод уменьшения размера изображений в моделях может быть использован в более сложных задачах компьютерного зрения, таких как обнаружение и сегментация объектов, где для обучения требуется большее число эпох, а также увеличены требования к расходу видеопамяти.

ЛИТЕРАТУРА

1. Densely Connected Convolutional Networks / G. Huang, Z. Liu, L. Van Der Maaten et al. // arXiv. – 2016. – DOI: 10.48550/arXiv.1608.06993.
2. Squeeze-and-Excitation Networks / J. Hu, L. Shen, S. Albanie et al. // arXiv. – 2017. – DOI: 10.48550/arXiv.1709.01507.
3. CSPNet: A New Backbone than can Enhance Learning Capability of CNN / C.-Y. Wang, H.-Y. M. Liao, I-H. Yeh et al. // arXiv. – 2019. – DOI: 10.48550/arXiv.1911.11929.
4. Wavelet Integrated CNNs for Noise-Robust Image Classification / Q. Li, L. Shen, S. Guo et al. // arXiv. – 2020. – DOI: 10.48550/arXiv.2005.03337.
5. Liu P., Zhang H., Lian W. et al. Multi-level Wavelet Convolutional Neural Networks // arXiv.org. – 2019. – DOI: 10.48550/arXiv.1907.03128.
6. Xiangyu Z. Wavelet-Attention CNN for Image Classification // arXiv. – 2022. – DOI: 10.48550/arXiv.2201.09271.
7. CBAM: Convolutional Block Attention Module / S. Woo, J. Park, J.-Y. Lee et al. // arXiv. – 2018. – DOI: 10.48550/arXiv.1807.06521.
8. Identity Mappings in Deep Residual Networks / K. He, X. Zhang, S. Ren et al. // arXiv. – 2016. – DOI: 10.48550/arXiv.1603.05027.
9. Sandler M., Howard A., Zhu M. et al. MobileNetV2: Interested Residuals and Linear Bottlenecks // arXiv. – 2018. – DOI: 10.48550/arXiv.1801.04381.
10. Getreuer P. Wavelet CDF 9/7 Implementation [Электронный ресурс] // Getreuer: On Wavelet Implementation. – 1997. – URL: https://getreuer.info/posts/waveletcdf97/index.html (дата обращения 16.06.2024).

REFERENCES

1. Huang, G., Liu, Z., van der Maaten, L., & Weinberger, K. Q. (2016). Densely Connected Convolutional Networks. *arXiv*. DOI: 10.48550/arXiv.1608.06993.
2. Hu, J., Shen, L., Albanie, S., Sun, G., & Wu, E. (2017). Squeeze-and-Excitation Networks. *arXiv*. DOI: 10.48550/arXiv.1709.01507.

3. Wang, C.-Y., Liao, H.-Y. M., Yeh, J-H., Wu, Y.-H., Chen, P.-Y., Hsieh, J.-W. (2019). CSPNet: A New Backbone than can Enhance Learning Capability of CNN. *arXiv*. DOI: 10.48550/arXiv.1911.11929.
4. Li, Q., Shen, L., Guo, S., Lai, Z. (2020). Wavelet Integrated CNNs for Noise-Robust Image Classification. *arXiv*. DOI: 10.48550/arXiv.2005.03337.
5. Liu, P., Zhang, H., Lian, W., & Zuo, W. (2019). Multi-level Wavelet Convolutional Neural Networks. *arXiv*. DOI: 10.48550/arXiv.1907.03128.
6. Xiangyu, Z. (2022). Wavelet-Attention CNN for Image Classification. *arXiv*. DOI: 10.48550/arXiv.2201.09271.
7. Woo, S., Park, J., Lee, J.-Y., & Kweon, I. S. (2018). CBAM: Convolutional Block Attention Module. *arXiv*. DOI: 10.48550/arXiv.1807.06521.
8. He, K., Zhang, X., Ren, S., Sun, J. (2016). Identity Mappings in Deep Residual Networks. *arXiv*. DOI: 10.48550/arXiv.1603.05027.
9. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.-C. (2018). MobileNetV2: Interested Residuals and Linear Bottlenecks. *arXiv*. DOI: 10.48550/arXiv.1801.04381.
10. Getreuer, P. (1997). Wavelet CDF 9/7 Implementation. URL: https://getreuer.info/posts/waveletcdf97/index.html.

Поступила 25.06.2024

WAVELET-BASED EFFECTIVE IMAGES DOWNSCALING IN NEURAL NETWORKS

U. VARABEI, A. MALEVICH
(Belarusian State University, Minsk)

Based on discrete wavelet transform, several blocks for images downscaling in computer vision models were implemented. The blocks were tested with ResNetV2-50 and MobileNetV2 models on Flowers dataset. With small increase in number of models' parameters and close results in terms of metric the changes allowed to reduce number of training epochs by 34 % and VRAM requirements by 18 %. Due to the implementation details the blocks suggested can be used as a replacement of layers responsible for images downscaling in models for other tasks to save computation resources and speed up training process. In the blocks developed standard operations of addition and multiplication are used for evaluation of wavelet transform, which allows a simple export of trained models into other formats.

Keywords: *neural networks, deep learning, wavelets, discrete wavelet transform, image classification.*