

# Нейронные сети для распознавания формы по медиальному представлению

Н.А. Ломов<sup>1</sup>, С.П. Арсеев<sup>1</sup>  
nikita-lomov@mail.ru | 9413serg@gmail.com

<sup>1</sup>Московский государственный университет имени М.В. Ломоносова, Москва, Россия

*Статья посвящена разработке архитектур для нейронных сетей, оперирующих данными особого вида — медиальным представлением формы, которое рассматривается как частный случай неориентированного графа. Предложены способы решения проблем, затрудняющих обработку данных такого типа традиционными нейросетями: различная длина входных данных, неоднородность их структуры, неупорядоченность составных элементов. Сформулированы скелетные аналоги стандартных операций, используемых в свёрточных нейронных сетях. Проведены эксперименты по классификации рукописных цифр.*

**Ключевые слова:** глубинное обучение, свёрточные нейронные сети, медиальное представление, скелет.

## Neural Networks for Shape Recognition by Medial Representation

N.A. Lomov<sup>1</sup>, S.P. Arseev<sup>1</sup>  
nikita-lomov@mail.ru | 9413serg@gmail.com  
<sup>1</sup>Lomonosov Moscow State University, Moscow, Russia

*The article is dedicated to the development of neural networks that process data of a special kind — a medial representation of the shape, which is considered as a special case of an undirected graph. Methods for solving problems that complicate the processing of data of this type by traditional neural networks — different length of input data, heterogeneity of their structure, unordered constituent elements — are proposed. Skeletal counterparts of standard operations used in convolutional neural networks are formulated. Experiments on the classification of handwritten digits are carried out.*

**Keywords:** deep learning, convolutional neural networks, medial representation, skeleton.

### 1. Введение

Стремительный рост популярности методов глубинного обучения несколько лет назад начался с успехов в ряде задач распознавания изображений. Изображения обычно задаются матрицами точек, поэтому их можно считать примером организованных, стандартизированных данных. В дальнейшем методы глубинного обучения также добились серьёзного прогресса в задачах с данными более сложной природы, которые можно представить последовательностями произвольной длины: распознавание речи, распознавание рукописных текстов, машинный перевод и т. д.

Предложенные архитектуры используют структурированные (по времени или положению) данные. Вместе с тем проблема обработки данных менее «удобной» для нейросетей природы остаётся сравнительно мало изученной. Характерным примером данных такого типа являются графы, а в целом подобные данные часто возникают в задачах вычислительной геометрии: это облака точек, полигональные модели, триангуляции, контурно-скелетные представления. Трудности заключаются в том, что объекты, подлежащие обработке, не упорядочены и могут иметь произвольную размерность и структуру связей. обстоятельное исследование этой проблемы и возможных подходов к её решению можно найти в статье [3].

Цель данной работы — применить нейросетевые методы к графам особого класса — скелетам, которые являются описанием формы в виде её некоторой основы, «каркаса». Хотя мы исходим из того, что скелет

является частным случаем неориентированного графа, некоторые его особенности, в частности геометрическая природа и сильная разреженность, позволяют извлечь из них дополнительную пользу.

Непосредственно скелеты используются в нейросетевых методах в очень ограниченном контексте: как правило, это задачи распознавания действий и жестов [5, 7], а сами скелеты получены с помощью сенсорных устройств (например, Kinect). Такие скелеты предназначены исключительно для описания фигуры человека и имеют строго заданную структуру. Примеры использования в качестве входных данных скелетов для описания произвольных форм авторам не известны. В статье предпринята попытка начать исследования в данной области.

### 2. Обработка скелетов при помощи нейросети

#### 2.1. Базовые понятия

*Скелетом* называется множество центров всех вписанных пустых кругов фигуры.

*Радиальной функцией* — функция, которая каждой точке скелета ставит в соответствие радиус вписанного круга.

Скелет и радиальная функция в совокупности образуют *медиальное представление формы*.

*Стрижкой* (регуляризацией) скелета называется процесс отсечения ветвей, вносящих незначительный вклад в образование формы. Способ стрижки, который будет использоваться далее, описан в [1].

Скелет можно рассматривать как геометрический граф, рёбрами которого являются отрезки прямых линий и парабол. К такому представлению ведёт получение скелета из диаграммы Вороного [2]. В рамках данной задачи мы будем рассматривать скелет в виде  $S = (V, E)$ , где каждая вершина  $v \in V$  представляется в виде тройки  $(x, y, r)$  — координат и значения радиальной функции, а каждое ребро  $e \in E$  является неупорядоченной парой вершин.

**2.2. Упорядочение вершин**

Наша первоначальная цель — добиться принципиальной возможности подавать скелеты на вход нейронной сети. На этом пути мы видим два препятствия: 1) скелеты имеют различную размерность в силу различного числа рёбер и вершин, 2) элементы скелета не имеют «естественного» порядка. В работе [10] предложено решение этой задачи: вершины скелета должны быть упорядочены согласно их центральности по посредничеству (betweenness centrality). Для вершины  $v$  эта величина определяется следующим образом:

$$g(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

где  $\sigma_{st}$  — число кратчайших путей из вершины  $s$  в вершину  $t$ , а  $\sigma_{st}(v)$  — число таких путей, проходящих через  $v$ . Таким образом, в начало списка попадут вершины, которые имеют больше соседей по сравнению с остальными и достаточно удалены от терминальных. В конец же списка попадут изолированные и терминальные вершины. В качестве дополнительной характеристики мы используем центральность по близости (closeness centrality,  $h(v) = \frac{1}{\sum_u d(u,v)}$ ), а в случае совпадения будем сравнивать расстояния ( $d(u, v)$ ) вершин-кандидатов до вершин, уже добавленных в список.

**Algorithm 1** Упорядочение вершин скелета

```

Дано: множество вершин  $\{v_i\}_{i=1}^n$ 
for  $i \in 1, \dots, n$  do
     $F_i = [g(v_i), h(v_i)]$ 
 $I = [1, \dots, n]$ 
 $L = []$ 
while  $I \neq []$  do
    Упорядочить список  $I$  лексикографически по убыванию  $F_i$ 
     $k = \text{pop}(I)$  // Извлечь первый элемент из списка
    Приписать  $k$  в конец  $L$ 
    for  $i \in I$  do
        Приписать  $-d(v_i, v_k)$  в конец  $F_i$ 
    
```

Отметим, что упорядочение может и не понадобиться, если на последнем слое мы будем применять ко всей карте признаков симметричную pooling-операцию: например, максимум или среднее значение.

**2.3. Унификация скелета**

Вновь обратимся к работе [10]: чтобы оставить в графе заданное число вершин  $n$ , предлагается взять из полученного на предыдущем списка первые  $n$  вершин с сохранением порядка, а если вершин меньше  $n$  — дополнить список фиктивными вершинами. В случае ске-

лета это исказит сам геометрический облик графа: если рёбер было слишком много, концы ветвей будут отсечены. Возникает идея использовать скелет, сохраняющий очертания исходного, но с фиксированным числом вершин и рёбрами примерно одинаковой длины.

**Algorithm 2** Унификация скелета

**Дано:** скелет  $S$ , число вершин  $n$ , минимальная относительная длина  $\varepsilon$ .

Если в скелете больше одной компоненты связности, сделать скелет связным с помощью дополнительных рёбер минимальной длины.

Разбить скелет на ветви — линии, концами которых являются точки степени 1 и 3, и которые не содержат промежуточных точек такого типа. Число ветвей обозначим за  $n_e$ , число их концов — за  $n_v$ .

**if**  $n_v > n$  **then**

Для  $n_v - n$  кратчайших ветвей стянуть концы в одну точку посередине ветви

**else**

Определить длину каждой ветви  $l_i$ , общую длину скелета  $L$  и среднюю длину  $l = \frac{L}{n_e + n - n_v}$

Для ветвей, длина которых меньше  $\varepsilon l$ , стянуть концы

Пересчитать  $n_e$  и  $n_v$

**for**  $i \in 1, \dots, n_e$  **do**

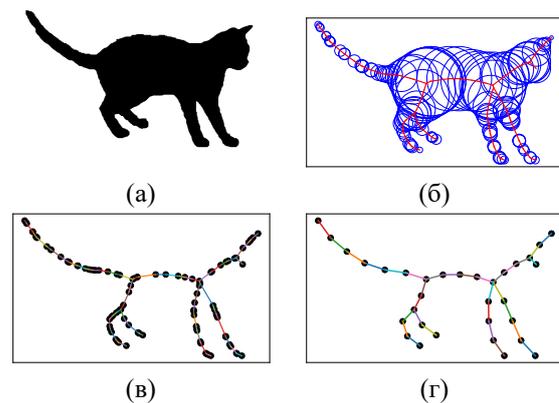
$$n_i = \lceil \frac{l_i}{l} \rceil - 1$$

**while**  $n_v + \sum_{i=1}^{n_e} n_i \geq n$  **do**

Выбрать вершину с наименьшим  $\frac{l_i}{n_i - 1}$  и уменьшить  $n_i$  на 1

На каждую ветвь равномерно добавить  $n_i$  вершин

Процесс подготовки входных данных проиллюстрирован рис. 1.



**Рис. 1.** Построение скелета с рёбрами равной длины. а) Изображение б) медиальное представление в) скелет г) унифицированный скелет.

**2.4. Свёрточный слой**

В случае изображений операция свёртки — это линейная комбинация признаков пикселей, находящихся

ся в окрестности выбранного. Действуя по аналогии, сформируем для выбранной вершины окрестность из  $r$  ближайших вершин в графе, а если достижимых вершин меньше  $r$ , дополним окрестность фиктивными вершинами с нулевыми признаками. Чтобы избежать элемента случайности, используем дополнительные критерии:

1. Список длин кратчайших путей до уже выбранных для окрестности точек.
2. Центральность по посредничеству.
3. Центральность по близости.

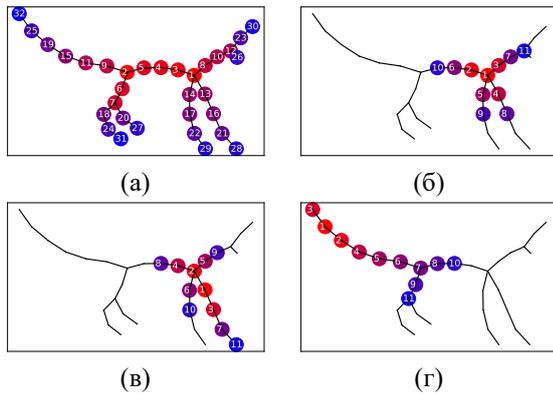
Общий вид процедуры аналогичен случаю упорядочения вершин в скелете.

Можно формально определить процедуру графовой свёртки окрестностей следующим образом. Пусть  $F \in \mathbb{R}^{n \times m}$  — матрица вершин-признаков,  $P \in \mathbb{N}_0^{n \times r}$  — матрица индексов элементов, образующих окрестности. Тогда  $E(F, P) \in \mathbb{R}^{n \times r \times m}$ , и

$$e_{ijk} = \begin{cases} f_{p_{ij}k}, & \text{если } p_{ij} \geq 0, \\ 0, & \text{если } p_{ij} = 0. \end{cases}$$

Теперь, если  $W \in \mathbb{R}^{r \times m \times q}$ , результат свёртки  $C(E, W) \in \mathbb{R}^{n \times q}$  определяется как:

$$c_{ij} = \sum_{1 \leq t \leq r, 1 \leq s \leq m} e_{its} w_{tsj}.$$



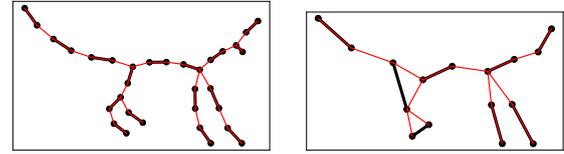
**Рис. 2.** Упорядочение всех вершин (а) и построение окрестностей длины 11 для отдельных вершин (б,в,г). При проходе по списку цвета меняются от красного к синему.

Примеры построения окрестностей, а также результат упорядочения всех вершин отражены на рис. 2.

### 2.5. Слой субдискретизации

Целью слоя субдискретизации является снижение размерности данных. Будем разбивать вершины графа на группы с тем расчётом, чтобы подграфы, образованные вершинами в группе, были связными. Можно предложить метод, который разбивает вершины по парам, убирая из графа пару из какой-то терминальной вершины и инцидентной ей, если её степень не больше 2. В том случае, если такой шаг невозможен, используется ряд эвристических правил для разрешения коллизий. Результатом применения такой процедуры будет новый граф, в котором пары считаются смежными

если есть хотя бы одно ребро, соединяющее вершины из разных пар. Повторяя такую процедуру  $k$  раз, можно добиться разбиения множества вершин на группы мощностью  $2^k$  (рис. 3).



**Рис. 3.** Объединение вершин в пары и симплификация графа путём замены пары вершин на одну новую

Теперь можно определять архитектуру «скелетной» сети с помощью тех же обозначений, которые традиционно используются для свёрточных сетей.

### 3. Архитектура без унификации

Альтернативным способом применения свёрточных нейронных сетей для обработки скелетов является использование графовой свёрточной архитектуры, не требующей унификации графа. Для сравнения подходов был реализован алгоритм на основе архитектуры, описанной в [9]. Сеть в этой архитектуре является полностью свёрточной и состоит из последовательности слоёв, описываемых формулой:

$$H^{(l)} = \sigma \left( \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l-1)} W^{(l)} \right).$$

Здесь  $l$  — номер слоя,  $H^{(l)} \in \mathbb{R}^{N \times D}$  — значения активаций на слое  $l$ ,  $\tilde{A} = A + I_N$  — матрица смежности графа с добавленными петлями на каждой вершине:  $A$  — матрица смежности, а  $I_N$  — единичная матрица.  $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$ ,  $W^{(l)}$  — обучаемая матрица весов слоя  $l$ , а  $\sigma$  — функция активации. На каждом слое выход предыдущего слоя (на первом слое используется матрица признаков исходного графа) умножается слева на симметрическую нормализованную матрицу Кирхгофа для данного графа, а справа — на матрицу весов слоя.

Поскольку при такой операции информация распространяется только по непосредственным соседям узла графа, в граф добавляется фиктивная вершина, соединённая со всеми остальными. Также, метка графа при классификации определяется выходом последнего свёрточного слоя для этой вершины.

### 4. Эксперименты

Эксперименты проводились на популярной базе изображений рукописных цифр MNIST. Сначала изображения были бинаризованы методом Оцу, далее для них были вычислены скелеты, которые подверглись процедуре стрижки с параметром в 1 пиксель. Унифицированные скелеты содержали 32 вершины. Архитектура сети состояла из следующих слоёв:

1. Свёрточный слой размера  $7 \times 3 \times 16$  (окрестность мощности 7, 3 признака —  $x, y, r$  и 16 фильтров) + функция активации ReLU ( $f(x) = x^+ = \max(0, x)$ ).
2. Свёрточный слой размера  $9 \times 7 \times 32 + \text{ReLU}$ .

3. Полносвязный слой размера 256.
4. Свёрточный слой с размерностью признаков на выходе 10.
5. Слой softmax-функции ( $\sigma(z)_i = \frac{e^{z_i}}{\sum_{k=1}^K e^{z_k}}$ ) для определения итоговой метки класса.

В свёрточных слоях применялась регуляризация по методу dropout, нейроны исключались из сети с вероятностью 50%. Точность классификации составила 98,36%. Это несколько уступает результату, который показывают наиболее эффективные из графовых сетей ChebNet[4] и MoNet[8], когда вершины в графе соответствуют пикселям изображения (таблица 1). Следует, однако, отметить, что все графы при этом имеют одинаковую структуру, поэтому архитектура сети оказывается близкой к традиционным свёрточным сетям, таким как [6]. Неоднородность возникает при выделении на изображениях суперпикселей, особым способом описания которых можно считать рёбра скелета. Таким образом, в случае произвольных графов предложенный метод показывает результат, превосходящий существующие аналоги. Отметим также компактность скелетного описания: каждый граф имеет лишь 32 вершины с 3 признаками; если потребовать от других графовых сетей подобной длины описания, превосходство нашего метода станет ещё более ощутимым.

Метод	LeNet5[6]	ChebNet[4]	MoNet[8]
Полная сетка	99,33%	99,14%	99,19%
$\frac{1}{4}$ сетки	98,59%	97,70%	98,16%
300 суперпикс.	—	88,05%	97,30%
150 суперпикс.	—	80,94%	96,75%
75 суперпикс.	—	75,62%	91,11%

**Таблица 1.** Точность классификации для различных форматов входных данных

Для альтернативной архитектуры без унификации графа сеть состояла из следующих слоёв:

1. Два свёрточных слоя с размерностью признаков на выходе 32 + ReLU.
2. Четыре свёрточных слоя с размерностью 64 + ReLU.
3. Свёрточный слой с размерностью 10.
4. Слой softmax-функции для определения итоговой метки класса.

Несмотря на более сложную архитектуру, точность классификации составила всего 90%. Это свидетельствует о том, что выбранный подход, успешно применявшийся в [9] для задачи классификации научных работ, плохо поддаётся переносу на задачу классификации изображений. Вероятно, это связано с оператором графовой свёртки, в результате применения которого может теряться информация о взаимном расположении узлов. Ещё одним источником трудностей является отсутствие аналога полносвязного слоя для таких данных, что влечёт за собой усложнение архитектуры.

## 5. Заключение

В работе были рассмотрены два метода глубинного

обучения с использованием подхода графовой свёртки на примере задачи классификации изображений. Были предложены два подхода, позволяющих использовать медиальное представление, заданное графом произвольной структуры, для решения задачи классификации. Первый подход, основанный на предобработке и унификации медиального представления, показал на наборе данных MNIST качество работы, сопоставимое с традиционными нейросетевыми алгоритмами. Попытка исключить этап унификации, предпринятая во втором подходе, показала снижение точности работы алгоритма.

## 6. Благодарности

Работа выполнена при поддержке РФФИ грант 17-01-00917.

## 7. Литература

- [1] *Местецкий Л.М.* Непрерывная морфология бинарных изображений: фигуры, скелеты циркуляры. М.: ФИЗМАТЛИТ, 2009.
- [2] *Местецкий Л. М.* Скелет многоугольной фигуры — представление плоским прямолинейным графом. — Труды 20-й международной конференции «Графикон-2010». СПб, ИТМО, с. 222–229, 2010.
- [3] *M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, P. Vandergheynst.* Geometric deep learning: going beyond Euclidean data // IEEE Signal Processing Magazine, Vol. 34(4), pp. 18–42, 2017.
- [4] *M. Defferrard, X. Bresson, P. Vandergheynst.* Convolutional neural networks on graphs with fast localized spectral filtering // Proceedings of the 30th International Conference on Neural Information Processing Systems, pp. 3844–3852, 2016.
- [5] *Y. Du, W. Wang, L. Wang.* Hierarchical recurrent neural network for skeleton based action recognition // 2015 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1110–1118, 2015.
- [6] *Y. LeCun, L. Bottou, Y. Bengio, P. Haffner.* Gradient-based learning applied to document recognition // Proc. IEEE, 86(11):2278–2324, 1998.
- [7] *C. Li, P. Wang, S. Wang, Y. Hou, W. Li.* Skeleton-based action recognition using LSTM and CNN // 2017 IEEE International Conference on Multimedia & Expo Workshops, pp. 585–590, 2017.
- [8] *F. Monti, D. Boscaini, J. Masci, E. Rodolà, J. Svoboda, M. Bronstein.* Geometric Deep Learning on Graphs and Manifolds Using Mixture Model CNNs // 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5425–5434, 2017.
- [9] *T. N. Kipf, M. Welling.* Semi-Supervised Classification with Graph Convolutional Networks // arXiv preprint, arXiv:1609.02907, 2016.
- [10] *M. Niepert, M. Ahmed, K. Kutzkov.* Learning Convolutional Neural Networks for Graphs // Proceedings of the 33rd International Conference on International Conference on Machine Learning, Vol. 48, pp. 2014–2023, 2016.