

# Обработка и анализ изображений и сигналов

## Реализация классических алгоритмов анализа изображений через полносверточные нейронные сети\*

А.Е. Жуковский<sup>I,II</sup>, Е.Е. Лимонова<sup>II,III</sup>, Д.П. Николаев<sup>IV</sup>

<sup>I</sup> Московский физико-технический институт (государственный университет), г. Москва, Россия

<sup>II</sup> ООО «Смарт Энджинс Сервис», г. Москва, Россия

<sup>III</sup> Институт системного анализа Федерального исследовательского центра «Информатика и управление» Российской академии наук, г. Москва, Россия

<sup>IV</sup> Институт проблем передачи информации Российской академии наук, г. Москва, Россия

**Аннотация.** В работе представлены дифференцируемые реализации нескольких классических алгоритмов анализа изображений: детектора краев Канни, детектора углов Харисса и бинаризации Ниблэка. Каждая реализация представлена в виде полносверточной нейронной сети, внутреннее устройство которой в точности повторяет устройство оригинального алгоритма. Использование подобной реализации позволяет настраивать внутренние параметры при помощи градиентного спуска. Как составную часть реализации также представлено обобщение субдискретизирующих (pooling) слоев, позволяющее использовать их совместно с произвольным структурным элементом. Приводится анализ приведенных архитектур сетей и показывается связи с современными подходами.

**Ключевые слова:** детектор краев Канни, детектор углов Харисса, бинаризация Ниблэка, полносверточная нейронная сеть.

**DOI:** 10.14357/20790279180512

### Введение

Подходы к решению задач с помощью нейронных сетей показывают впечатляющие результаты. В последних исследованиях в области обучения нейронных сетей уделено внимание также задачам, для которых предоставить точную формулировку проблематично. К таким задачам можно, например, отнести задачи обнаружения краев или углов на изображении или задачу бинаризации изображения. Был представлен ряд работ по обнаружению краев [1–4]. Также внимание было обращено к бинаризации [5] и обнаружению особых точек [6], но здесь исследования пока что находятся на начальной стадии.

Классические алгоритмы, тем не менее, часто используются. Они решают поставленные задачи с приемлемым качеством и в то же самое время имеют на порядки меньшее количество параметров, что делает их быстрее и предпочтительнее для применения в ряде случаев. В настоящее время параметры классических алгоритмов подбираются вручную, с использованием не зависящих от производных алгоритмов [7] или при помощи методов, специально разработанных для каждого конкретного случая [8]. Однако внутреннее устройство алгоритмов в большинстве случаев позволяет реализовать их в терминах нейронных сетей и использовать градиентный спуск для оптимизации параметров, тем самым

\* Выполнено при частичной поддержке РФФИ (гранты 17-29-07093 и 17-29-07092)

открывая для них возможности алгоритмов машинного обучения.

Для реализации большинства алгоритмов требуется расширить привычный набор используемых в нейронных сетях слоев дополнительными функциями активации (которые реализуют стандартные функции, например, возведение в квадрат или взятие квадратного корня) и поэлементные матричные операции. Данными дополнениями варианты расширения модулей не ограничиваются: были показаны возможности включения в состав нейронных сетей проективного преобразования изображений [9], преобразования Хафа [10], также можно с легкостью представить субдискретизирующие слои по операциям минимума или медианы. Связующим звеном между всеми модулями нейронной сети является их дифференцируемость, позволяющая оптимизировать их комбинацию с помощью алгоритма обратного распространения ошибки. Теория, лежащая в основе классических алгоритмов, также находит свое применение в обучении нейронных сетей, позволяя достигать им лучших результатов и даже превосходить человека в ряде задач. Например, в [3,4] авторы используют ориентацию краев для предсказания более точного положения границ, а в [9] – представляют модуль нейронной сети, позволяющий пространственные преобразования данных при помощи проективного преобразования.

Мы представляем в работе архитектуры полностью сверточных нейронных сетей [11], в точности соответствующие нескольким алгоритмам анализа изображений. Мы рассмотрим возможности и ограничения реализации алгоритмов и также обсудим возможности дальнейшего развития.

## 1. Реализация классических алгоритмов анализа изображений

В данной части мы представим нейросетевые реализации четырех алгоритмов анализа изображения. Для введения, мы начнем с упрощенного примера реализации градиента изображения в части 1.1. Далее мы опишем реализацию алгоритма бинаризации Ниблэка в части 1.2, детектора углов Харисса в части 1.3, и детектора краев Канни в части 1.4. По ходу повествования мы будем комментировать возможности обобщения и расширения приведенных реализаций и их ограничения.

Значительная часть информации об устройстве каждой из реализаций приведена на рис. 1 и 2, иллюстрирующих архитектуры соответствующих нейронных сетей. Опишем цвета, используемые

нами для иллюстрации модулей нейронных сетей. Черным цветом обозначены входные, исходные или промежуточные данные. Светло серым обозначены сверточные слои; для каждого из них указаны размеры фильтра, количество фильтров (если не указано иного, то 1), в квадратных скобках указаны значения весов фильтра, и, в случаях, где присутствует сдвиг, указано значение сдвига. Темно серым цветом обозначены поканальные операции, такие как объединение каналов нескольких входов в один или поэлементное умножение нескольких входов. Белым цветом обозначены поэлементные операции, например, функции активации. Белым цветом с указанием слева используемого структурного элемента также обозначен слой выделения локальных максимумов, представленный в части 1.3. В объединяющих каналах слоев предполагается, что входные данные упорядочены слева направо в соответствии с входящими стрелками.

### 1.1 Градиент изображения

Начнем описание реализаций с упрощенного примера взятия градиента от изображения. Обычно эта операция реализуется при помощи двух сверток, вычисляющих частные производные по каждой из координат, например, с фильтрами Собеля. Она представляет собой ни что иное, как нейронную сеть с единственным сверточным слоем, состоящим из двух фильтров, инициализированных значениями фильтра Собеля. Для получения абсолютного значения градиента  $M = \sqrt{I_x^2 + I_y^2}$  мы можем поэлементно возвести в квадрат полученные значения частных производных, сложить значения с помощью свертки  $1 \times 1$ , реализующей линейную комбинацию входных каналов и поэлементно взять квадратный корень. Реализация данной сети проиллюстрирована на рис. 2 как часть алгоритма обнаружения границ Канни.

Модификация фильтра для использования на цветных изображениях требует обработки увеличенного количества входных каналов. Этого можно добиться, не увеличивая количество каналов в сверточном слое, а путем применения одних и тех же фильтров к каждому каналу, и добавления дополнительного слоя, реализующего либо линейную комбинацию каналов через свертку  $1 \times 1$ , либо взятие поканального максимума.

### 1.2 Бинаризация Ниблэка

Опишем теперь реализацию алгоритма локальной бинаризации Ниблэка [12]. Реализация изображена на рис. 1 слева. Алгоритм реализует вычисление локальных порогов для каждого пикселя изображения исходя из среднего значения и

стандартного отклонения значений пикселей в окрестности, и по ним бинаризует изображение:

$$I_{binary}(x, y) = \theta(I(x, y) - T(x, y)) = \theta(I(x, y) - \mu(x, y) - k \times \sigma(x, y) - a). \quad (1)$$

Здесь:

- $I$  – входное изображение;
- $I_{binary}$  – итоговое бинарное изображение;
- $T$  – пороги для каждого пикселя;
- $\theta(x) = \begin{cases} 1, & x < 0 \\ 0, & x \geq 0 \end{cases}$  – пороговая функция

Хэвисайда, которую можно в нейронной сети реализовать в явном виде или заменить сигмоидальной функцией для получения сглаженных значений бинаризации;

- $\mu(x, y)$  – среднее значение пикселей в окрестности. Может быть подсчитано с помощью фильтра среднего, или же фильтра со всеми единицами, считающего сумму в окрестности и последующей нормализацией на количество пикселей в окрестности;
- $\sigma(x, y)$  – несмещенная оценка стандартного отклонения значений пикселей в окрестности. Реализуется как линейная комбинация (при помощи свертки  $1 \times 1$ ) квадрата среднего значения и среднего квадрата значений в окрестности, с последующим взятием корня;
- $k$  и  $a$  – параметры алгоритма.

Для сравнения значений пикселей изображения и вычисленных пороговых значений мы применяем свертку  $1 \times 1$  с весами  $[1, -1]$  и далее вычисляем сигмоидальную функцию от полученного ответа.

Стоит отметить, что все операции после получения стандартного отклонения являются либо объединениями каналов, либо сверточными слоями, а значит, могут быть выражены единственными объединением и следующим за ним сверточным слоем. Также стоит отметить, что операции возведения в квадрат и взятия корня можно представить в виде дифференцируемого слоя поэлементного возведения в степень с обучаемым значением степени.

### 1.3 Детектор углов Харриса

Далее рассмотрим реализацию детектора углов Харриса [13]. Реализация проиллюстрирована на рис. 1 справа. В алгоритме по частным производным вычисляется структурный тензор  $M$ , и исследуются его собственные значения в локальной окрестности каждой точки:

$$R = \lambda_1 \lambda_2 - k \times (\lambda_1 + \lambda_2)^2 = \det(M) - k \times \text{trace}(M)^2 \quad (2)$$

Точки, соответствующие углам на изображении, будут иметь большие значения обоих собственных значений и, соответственно, большое значение отклика детектора.

Компоненты структурного тензора считаются как произведения вычисленных при помощи свертки с фильтром Собеля частных производных. Для обобщения значений из локальной окрестности используется оконная функция  $N \times N$ , обыкновенно равномерная или функция Гаусса, обе из которых реализуются при помощи свертки и в таком же виде укладываются в архитектуру нейронной сети. Обратим внимание, что, пользуясь линейностью операции свертки, реализацию получения усредненных по окрестности значений структурного тензора можно упростить до единственного сверточного слоя с фильтрами  $(N+2) \times (N+2)$ , применяемого к квадрату изображения.

Для анализа собственных значений используются след и определитель структурного тензора. Вычисление следа представляется как сумма двух входов (как и прежде, реализуемая при помощи свертки  $1 \times 1$ ), а определителя – как операция разности между поэлементными произведениями входных данных.

По полученному отклику детектора Харриса  $R$  далее требуется выбрать соответствующие углам точки. Для начала применим ограничение снизу на величину отклика, оставляя для рассмотрения только точки, в которых  $R$  больше порога  $T$ , являющегося параметром алгоритма. Реализация данного отсека будет представлять собой добавление сдвига со значением  $-T$  и применение выпрямляющей активационной функции ReLU [14]. Данная операция сделает нулевыми все значения, меньшие порога  $T$ , и уменьшит на  $T$  все положительные значения. Мы можем избежать этого, используя дополнительно операцию бинаризации данного изображения, умножение бинаризованной матрицы на тот же порог  $T$  и сложение с начальным изображением. Но, так как данная особенность реализации не повлияет на дальнейшее использование алгоритма, мы пропустим этот этап.

Опять же отметим, что и вычисление следа, и вычисление определителя в данной реализации можно выразить через взятие квадрата от входных данных  $I_x^2$ ,  $I_y^2$ ,  $I_x I_y$  и применения к ним единственного сверточного слоя. Так как нелинейности присутствуют только в виде умножения или взятия квадрата, их можно вынести в самое начало и использовать далее четвертую степень входного изображения. Тогда этапы вычисления усредненных по окрестности значений структурного тензора, следа и определителя, и порогового ограничения

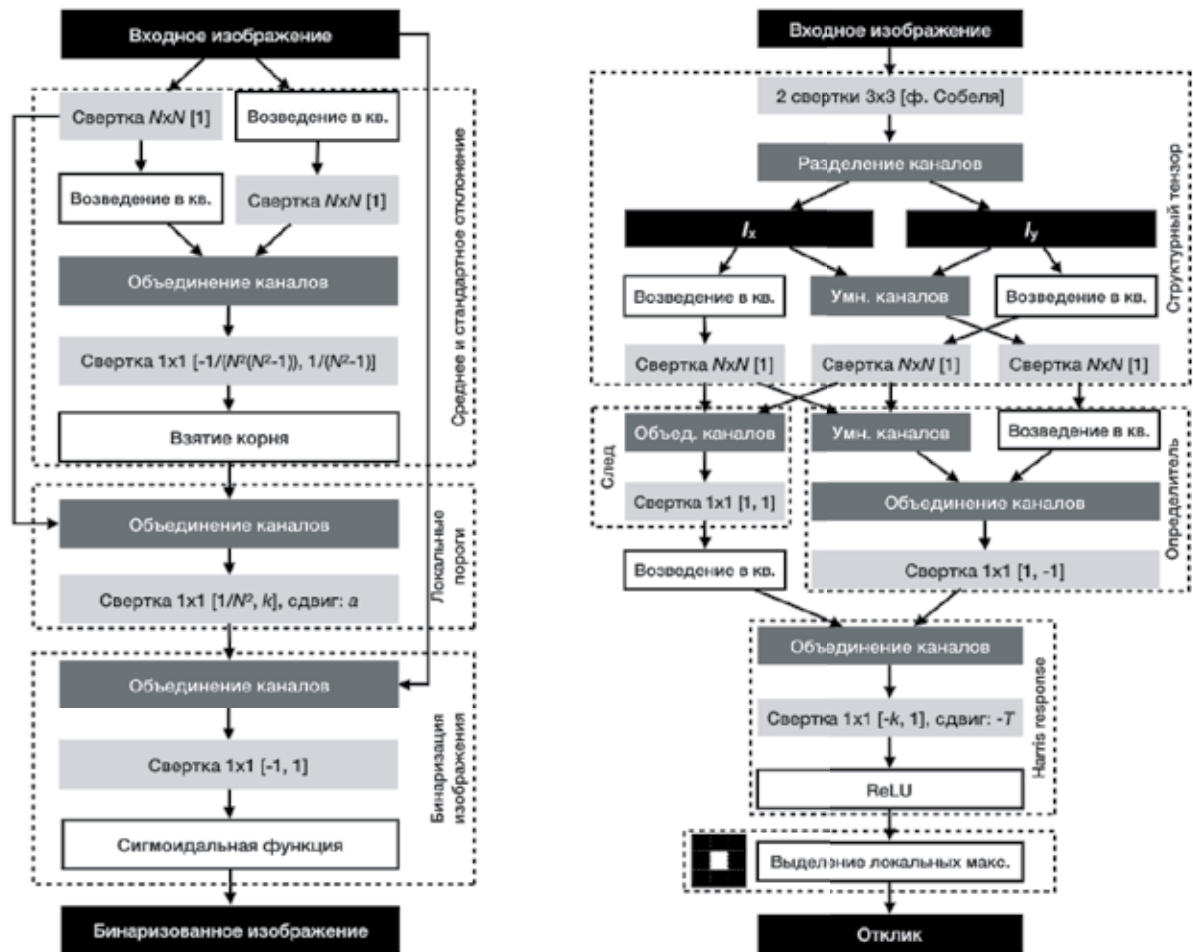


Рис. 1. Слева: реализация бинаризации Ниблэка; справа: реализация детектора углов Харисса

можно объединить в единственный сверточный слой с выпрямляющей активационной функцией.

Далее, для точного выделения углов требуется реализовать вычисление локальных максимумов функции отклика  $R$ . Для того, чтобы сделать это, нам потребуется обобщение субдискретизирующих слоев нейронной сети.

Операция морфологической дилатации [15] на полутоновых изображениях является фильтром локального максимума с соответствующим структурным элементом. В этом плане слой субдискретизации по операции максимума (max-pooling) представляет собой морфологическую дилатацию с прямоугольным структурным элементом с последующей подвыборкой значений. Структурный элемент может иметь произвольную форму, которую можно представить в виде бинарной маски. Используя операцию морфологической дилатации и принимая во внимание, что структурный элемент  $kernel$  представляет собой выколотую

окрестность точки, становится возможным получить максимальное значение в окрестности точки. Сравнивая с ним значение в самой точке, можно определить, является ли она локальным максимумом и обнулить значение, если это не так:

$$nms(x) = \begin{cases} x, & \text{если } x > x \oplus kernel, \\ 0, & \text{иначе} \end{cases} \quad (3)$$

Обобщая все вышесказанное, и принимая во внимание, что фильтры большого размера можно декомпозировать в последовательность фильтров меньшего размера, имеющую идентичное рецептивное поле, можно заключить, что детектор углов Харисса можно представить в виде классической полностью сверточной сети с последующим за ней слоем выделения локальных максимумов.

#### 1.4 Детектор краев Канни

Перейдем далее к реализации детектора границ Канни [16] (рис. 2).

Детектор границ состоит из сглаживания изображения с фильтром Гаусса, вычисления модуля градиента, получения тонких границ за счет вычисления локальных максимумов в направлении, перпендикулярном направлению границы, и применения двухпорогового отсечения. Реализация вычисления частных производных и модуля градиента показаны в части 1.1. Для поиска локальных максимумов также предполагается вычисление направления градиента в каждой точке  $\theta = \text{atan2}(I_x, I_y)$ . В рамках алгоритма, принимая во внимание, что направления градиента далее используются только для определения одного из четырех направлений границы, можно ограничиться только вычислением отношения  $\theta = \frac{I_y}{I_x}$ . Оно реализуется как поэлементное деление двух матриц с дополнительным ограничением на значения для гарантированного запрета деления на ноль.

Для определения локальных максимумов в направлении градиента будем рассматривать каждое из четырех направлений по отдельности. При помощи операции дилатации, представленной в части 1.3, можно сравнить модуль градиента в точке с двумя соседними значениями в заданном направлении и подавить значения во всех точках изображения, не представляющих собой локальный максимум в данном направлении. Также требуется выбрать точки, соответствующие данному направлению градиента. Для этого построим бинарную маску, дискретизируя значения направления градиента. Например, для горизонтального направления маска будет иметь следующий вид:

$$mask = \begin{cases} 1, & \text{если } -\tan\left(\frac{\pi}{8}\right) \leq \theta < \tan\left(\frac{\pi}{8}\right) \\ 0, & \text{иначе} \end{cases} \quad (4)$$

Объединяя полученные локальные максимумы в текущем направлении с маской соответствия

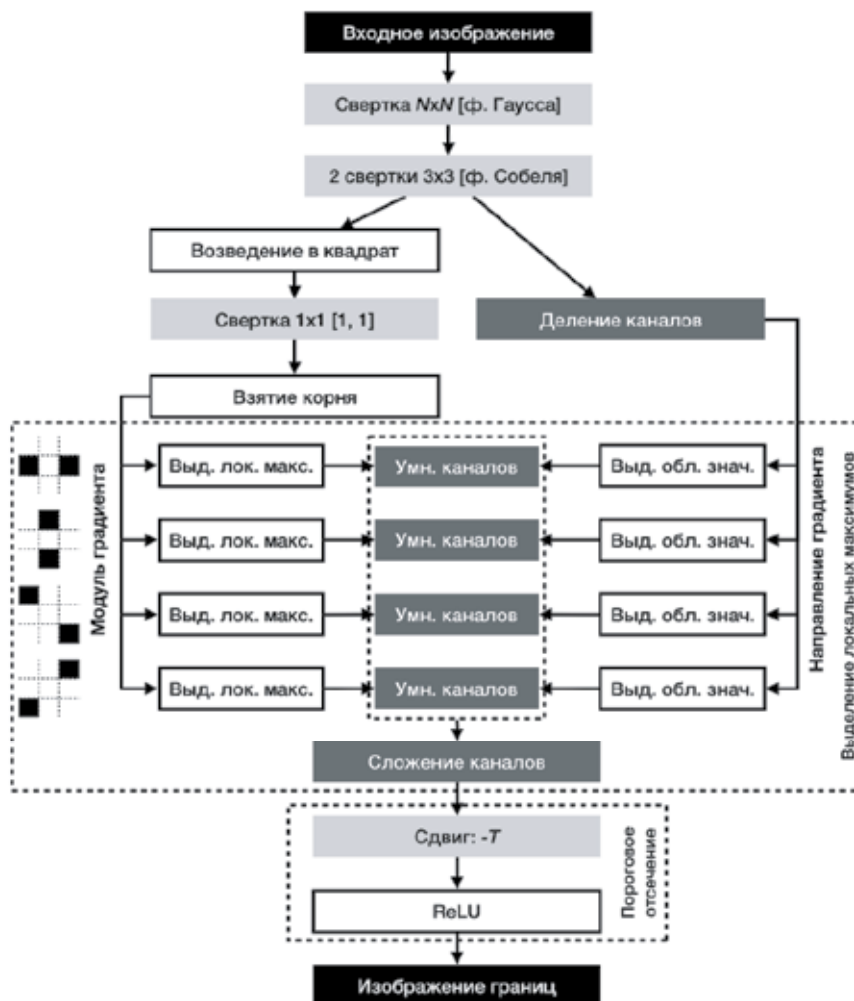


Рис. 2. Реализация детектора границ Канни

точки данному направлению умножением, получим изображение с тонкими краями для текущего направления. Далее, объединяя полученные для четырех направлений изображения с тонкими краями при помощи сложения, получим изображение с присутствующим на нем только локальными максимумами и тонкими краями.

Применение меньшего порога, ограничивающего значение модуля градиента, реализуется через добавление сдвига и выпрямляющей активационной функции, как и в части 1.3. Применение верхнего порога требует вычисления максимального значения для каждой из компонент связности на финальном изображении. Данная операция требует обработки изображения целиком и, очевидно, не может быть реализована при помощи локально действующей полностью сверточной сети, так что она пропущена в реализации и оставлена для последующей обработки вне нейронной сети.

## 2. Обсуждение

При наличии набора данных (например, [5] для бинаризации или [17] для обнаружения границ) корректной метрики для оценки качества работы и настроенного процесса обучения для данных реализаций становится возможным использовать алгоритм обратного распространения ошибки для оптимизации параметров. Единственным параметром, который потребуется подобрать вручную, является размер рецептивного поля оператора.

С процессом обучения становится возможным не только оптимизировать параметры алгоритма, но и модифицировать используемые в нем фильтры. Для того, чтобы позволить это потребуется добавить небольшой шум к теоретическим значениям и использовать их, как начальные значения весов фильтров.

Как уже упоминалось ранее, степени, используемые в реализации, также можно рассматривать в качестве обучаемых параметров при введении дополнительного специального слоя.

Имея возможность использовать алгоритм обратного распространения ошибки для обучения данных архитектур, также становится возможным исследование их модификаций и использование современных подходов, разработанных в машинном обучении. Мы можем добавить регуляризационные члены к функции потерь, использовать дополнительные механизмы регуляризации, такие как прореживание весов при обучении (dropout, [18]) или нормализация промежуточных данных (batch normalization, [19]). Также возможно наложить дополнительную регуляризацию на выход сети, что

должно быть крайне полезным для задачи обнаружения границ, где отношение краев и фона крайне несбалансированно. Вместо подбора размера сверток для управления размером рецептивного поля можно управлять количеством последовательных сверточных слоев, что заодно уменьшит количество параметров нейронной сети [20].

## Заключение

В работе были представлены архитектуры полностью сверточных сетей, которые явным образом реализуют алгоритмы детектора краев Канни, детектора углов Харисса и алгоритма локальной бинаризации Ниблэка. Было показано, что данные реализации позволяют настраивать параметры алгоритмов при помощи градиентного спуска. Была показана возможность включения в состав сети операции морфологической дилатации с произвольным структурным элементом и ее связь с слоем субдискретизации. На основе операции дилатации был разработан модуль поиска локальных максимумов. Для детектора границ Харисса было показано, что алгоритм можно интерпретировать как классическую полностью сверточную сеть с последующим выделением локальных максимумов.

## Литература

1. Xie S. and Tu Z. Holistically-nested edge detection, ICCV, 2015.
2. Kokkinos I. Pushing the boundaries of boundary detection using deep learning, ICLR, 2016.
3. Maninis K.K., Pont-Tuset J., Arbeláez P. and Van Gool L. Convolutional Oriented Boundaries, ECCV, 2016.
4. Maninis K.K., Pont-Tuset J., Arbeláez P. and Van Gool L. Convolutional Oriented Boundaries: From Image Segmentation to High-Level Tasks, TPAMI, 2018.
5. Pratikakis I., Zagoris K., Barlas G., Gatos B. ICDAR2017 Competition on Document Image Binarization (DIBCO 2017), ICDAR, 2017.
6. Lenc K., Vedaldi A. Learning Covariant Feature Detectors, GMLD workshop at ECCV, 2016.
7. Audet C., Dennis J.E. Mesh Adaptive Direct Search Algorithms for Constrained Optimization, SIAM J. Optim. 17, 2006. p.188–217.
8. Алиев М.А., Николаев Д.П., Сапарев А.А. Построение быстрых вычислительных схем настройки алгоритма бинаризации Ниблэка // Труды ИСА РАН. – 2014. – Т. 64. – № 3. – С. 25-34.
9. Jaderberg M., Simonyan K., Zisserman A., Kavukcuoglu K. Spatial Transformer Networks, NIPS, 2015.

10. *Sheshkus A., Ingacheva A., Nikolaev D.P.* Vanishing Points Detection Using Combination of Fast Hough Transform and Deep Learning, Proc. SPIE 10696, Tenth International Conference on Machine Vision (ICMV 2017), 106960H, pp. 1-8, 2018, DOI: 10.1117/12.2310170.
11. *Long J., Shelhamer E., Darrell T.* Fully Convolutional Networks for Semantic Segmentation, CVPR, 2015.
12. *Niblack W.* An Introduction to Digital Image Processing, Englewood Cliffs, PrenticeHall, 1986
13. *Harris C., Stephens M.* A Combined Corner and Edge Detector, Alvey Vision Conference, 15, 1988.
14. *Hahnloser R., Sarpeshkar R., Mahowald M.A., Douglas R.J., Seung H.S.* Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit, Nature, 405, pp. 947–951, 2000.
15. *Serra J.* Image Analysis and Mathematical Morphology, ISBN 0-12-637240-3, 1982.
16. *Canny J.* A Computational Approach To Edge Detection, IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6):679–698, 1986.
17. *Martin D., Fowlkes C., Tal D., Malik J.* A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics, Proc. 8th ICCV, 2001.
18. *Hinton G.E., Srivastava N., Krizhevsky A., Sutskever I., Salakhutdinov R.R.* Improving neural networks by preventing co-adaptation of feature detectors, arXiv:1207.0580, 2012.
19. *Ioffe S., Szegedy C.* Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, ICML, 2015.
20. *Simonyan K., Zisserman A.* Very Deep Convolutional Networks for Large-Scale Image Recognition, ICLR, 2015.

**Жуковский Александр Евгеньевич.** Московский физико-технический институт (государственный университет), г. Москва, Россия. Лектор кафедры алгоритмов и технологий программирования МФТИ. ООО «Смарт Энджинс Сервис», г. Москва, Россия. Научный сотрудник. Количество печатных работ: 15. Область научных интересов: компьютерное зрение, распознавание образов, машинное обучение. E-mail: zhukovsky@phystech.edu

**Лимонова Елена Евгеньевна.** Институт системного анализа Федерального исследовательского центра «Информатика и управление» Российской академии наук, г. Москва, Россия. Математик. ООО «Смарт Энджинс Сервис», г. Москва, Россия. Программист. Количество печатных работ: 16. Область научных интересов: обработка изображений, распознавание образов на мобильных устройствах. E-mail: limonova@smartengines.biz

**Николаев Дмитрий Петрович.** Институт проблем передачи информации Российской академии наук, г. Москва, Россия. Заместитель директора по научной работе. Кандидат физико-математических наук. Количество печатных работ: 158. Область научных интересов: машинное зрение, распознавание образов, быстрые алгоритмы обработки изображений. E-mail: dimonstr@iitp.ru

**Exact implementation of common image processing algorithms using fully convolutional networks**A.E. Zhukovsky<sup>I,II</sup>, E.E. Limonova<sup>II,III</sup>, D.P. Nikolaev<sup>IV</sup><sup>I</sup> Moscow Institute of Physics and Technology (State University), Russia, Moscow<sup>II</sup> LLC “Smart Engines Service”, Moscow, Russia<sup>III</sup> Institute for Systems Analysis, Federal Research Center “Computer Science and Control” of Russian Academy of Sciences<sup>IV</sup> Institute for Information Transmission Problems of Russian Academy of Sciences

**Abstract.** We present differentiable implementations of several common image processing algorithms: Canny edge detector, Niblack thresholding and Harris corner detector. The implementations are presented in the form of fully convolutional networks and explicitly arranged exactly to the original algorithms. Usage of such form of algorithms allows to tune their parameters with gradient descend. As a part of these implementations we introduce a generalization of pooling algorithm, which allows to use arbitrary structure element. We also analyze the given architectures and show the connections with contemporary approaches.

**Keywords:** *canny edge detector, Harris corner detector, Niblack thresholding, fully convolutional network.*

**DOI:** 10.14357/20790279180512

**References**

1. *S. Xie and Z. Tu.* Holistically-nested edge detection, ICCV, 2015.
2. *I. Kokkinos.* Pushing the boundaries of boundary detection using deep learning, ICLR, 2016.
3. *K.K. Maninis, J. Pont-Tuset, P. Arbeláez, and L. Van Gool.* Convolutional Oriented Boundaries, ECCV, 2016.
4. *K.K. Maninis, J. Pont-Tuset, P. Arbeláez, and L. Van Gool.* Convolutional Oriented Boundaries: From Image Segmentation to High-Level Tasks, TPAMI, 2018.
5. *I. Pratikakis, K. Zagoris, G. Barlas, B. Gatos.* ICDAR2017 Competition on Document Image Binarization (DIBCO 2017), ICDAR, 2017.
6. *K. Lenc, A. Vedaldi.* Learning Covariant Feature Detectors, GMLD workshop at ECCV, 2016.
7. *M. Jaderberg, K. Simonyan, A. Zisserman, K. Kavukcuoglu.* Spatial Transformer Networks, NIPS, 2015.
8. *A. Sheshkus, A. Ingacheva, D.P. Nikolaev.* Vanishing Points Detection Using Combination of Fast Hough Transform and Deep Learning, Proc. SPIE 10696, Tenth International Conference on Machine Vision (ICMV 2017), 106960H, pp. 1-8, 2018, DOI: 10.1117/12.2310170.
9. *Audet C., Dennis J.E.* Mesh Adaptive Direct Search Algorithms for Constrained Optimization, SIAM J. Optim. 17, 2006. p.188–217.
10. *M. Aliiev, D. Nikolaev, A. Saraev.* Construction of fast computational schemes for tuning the Niblack binarization algorithm, [Postroenie bystrykh vychislitelnykh skhem nastroyki algoritma binarizatsii Nibleka], in Russian, Proceedings of the ISA RAS, 2014.
11. *J. Long, E. Shelhamer, T. Darrell.* Fully Convolutional Networks for Semantic Segmentation, CVPR, 2015.
12. *W. Niblack.* An Introduction to Digital Image Processing, Englewood Cliffs, PrenticeHall, 1986
13. *C. Harris, M. Stephens.* A Combined Corner and Edge Detector, Alvey Vision Conference, 15, 1988.
14. *R. Hahnloser, R. Sarpeshkar, M. A. Mahowald, R. J. Douglas, H. S. Seung.* Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit, Nature, 405, pp. 947–951, 2000.
15. *J. Serra.* Image Analysis and Mathematical Morphology, ISBN 0-12-637240-3, 1982.
16. *Canny, J.* A Computational Approach To Edge Detection, IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6):679–698, 1986.
17. *D. Martin, C. Fowlkes, D. Tal, J. Malik.* A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics, Proc. 8th ICCV, 2001.
18. *G.E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, R.R. Salakhutdinov.* Improving neural networks by preventing co-adaptation of feature detectors, arXiv:1207.0580, 2012.
19. *S. Ioffe, C. Szegedy.* Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, ICML, 2015.
20. *K. Simonyan, A. Zisserman.* Very Deep Convolutional Networks for Large-Scale Image Recognition, ICLR, 2015.



**A.E. Zhukovsky.** Moscow Institute of Physics and Technology (State University), Institutsky per., 9, Moscow, region, Dolgoprudny, 141701, Russia. LLC “Smart Engines Service”, prospect 60-letiya Oktyabrya, 9, Moscow, 117321, Russia. E-mail: zhukovsky@phystech.edu

**E.E. Limonova.** Institute for Systems Analysis, Federal Research Center “Computer Science and Control” of Russian Academy of Sciences, 44/2 Vavilova str., Moscow, 119333, Russia. LLC “Smart Engines Service”, prospect 60-letiya Oktyabrya, 9, Moscow, 117321, Russia, E-mail: limonova@smartengines.biz

**D.P. Nikolaev.** Institute for Information Transmission Problems of Russian Academy of Sciences, Bolshoy Karetny per. 19, build.1, Moscow, 127051, Russia. PhD in Computer Science. E-mail: dimonstr@iitp.ru