Компьютерный анализ текстов

Контекстно-независимый метод быстрой детекции текста для распознавания номеров телефонов

А.В. Гайер[I,II]

[I] Федеральный исследовательский центр «Информатика и управление» Российской академии наук», г. Москва, Россия;
[II] ООО «Смарт Энджинс Сервис», г. Москва, Россия

Аннотация. Современные методы детекции текста на изображениях основаны на вычислительно затратных моделях глубокого обучения и требуют большое количество данных для обучения, в том числе реальных. В случае поиска текста в произвольных сценариях, процесс сбора и аннотирования настоящих данных для обучения крайне трудозатратен и дорог из-за высокой вариативности возможных сцен. В данной работе представлен новый метод детекции текста на произвольных изображениях, который не требует для обучения фотографий текста в реальных сценах и может быть обучен на простых синтетических данных в виде строк. Предложенная нейросетевая модель в 42 раза меньше, чем детектор текста в одной из лучших в плане качества и скорости работы системе распознавания текста PaddleOCR (84 КБ против 3.6 МБ), что делает ее отличным выбором для мобильных устройств. Модель была протестирована в составе системы распознавания номеров телефонов, где с ее помощью удалось достичь 80,35% правильно распознанных номеров.

Ключевые слова: *глубокое обучение, детекция объектов, сегментация изображений, детекция текста.*

DOI: 10.14357/20790279240305 **EDN: HREWAU**

Введение

Поиск текста на изображении представляет собой классическую задачу компьютерного зрения. Подходы, решающие данную проблему, сегодня активно применяются в задачах распознавания документов [1], перевода текста в режиме дополненной реальности, а также в задачах распознавания кодифицированных строк, т.е. имеющих фиксированный формат записи: номер телефона или банковской карты, платежные реквизиты, серийные номера деталей на производстве и т.д. Общей целью при постановке большинства задач, связанных с поиском текста, является упрощение ввода данных, уменьшение количества ошибок ввода и повышение удобства использования прикладных систем.

Современные методы поиска текста на изображении используют подходы глубокого обучения, для которых необходимы большие массивы обучающих данных. И если для задачи распознавания документов можно успешно использовать синтез данных [2], то для задачи поиска текста в произвольной сцене это сделать гораздо сложнее. Так, для задачи детекции номера телефона нужно учесть возможные поверхности, на которых может быть текст, а также сами сцены съемки, т.е. контекст. Т. о., есть необходимость разработки таких моделей поиска текста в произвольных сценах, которые можно обучить на простых синтезированных данных.

Уже более 10 лет для решения задач компьютерного зрения активно используются смартфоны

и планшеты, а также иные мобильные устройства со встроенной камерой. Мощность современных мобильных процессоров позволяет запускать глубокие нейронные сети прямо на устройстве, что важно для безопасной работы с персональными данными в виде документов и платежных реквизитов. Однако вычислительная сложность алгоритмов поиска текста на изображении, а также размер используемых для этого моделей все еще играют важную роль в виду двух факторов: а) большинство устройств относятся к бюджетному классу, мощность которых ограничена; б) энергоэффективные модели экономят расход заряда аккумулятора, который также ограничен.

В данной работе представлен новый нейросетевой подход поиска текстовых строк на изображении, который не завязан на контексте сцены при обучении и может быть эффективно применен на мобильных устройствах: модель детектора содержит всего 21 тысячу параметров и весит 84 КБ против 3,6 МБ компактного детектора текста в одной из лучших систем распознавания PaddleOCR. Обученный детектор был протестирован в задаче распознавания номера телефона на реальных данных, где он показал свою эффективность – количество правильно найденных и при этом распознанных номеров равняется 80,35%. Примеры целевых изображений задачи приведены на рис. 1. Входное изображение содержит номер телефона, который необходимо локализовать и распознать. Предложенный подход детекции текста работает с условием, что искомая строка имеет формат, позволяющий отличить ее от других строк на изображении. Тем самым, после этапа распознавания найденных строк минимизируется негативный эффект в виде ложных срабатываний детектора текста. В то же время, это позволяет значительно упростить модель по количеству обучаемых параметров.

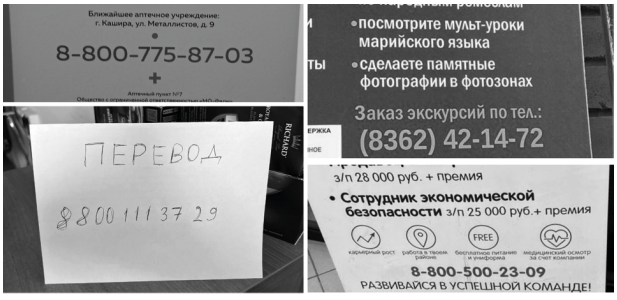


Рис. 1. Примеры изображений целевой задачи

1. Методы детекции текста на изображении

Ранние работы по поиску текста на изображении в основном рассматривали задачу распознавания сканов документов и предлагали подходы, основанные на методах обработки изображений. Предложенные в работах [3–5] алгоритмы поиска текста используют фильтрацию изображения, морфологические операции, бинаризацию и поиск связанных компонент для выделения текстовых блоков. В работе [6] для поиска длинных текстовых строк на изображении документа используется преобразование Хафа и анализ проекций изображения для оценки наклона документа и последующего извлечения строк. Такие подходы показывают хорошие результаты для сканов документов в хорошем качестве с учетом подбора параметров алгоритмов под размер шрифта, однако в сложных сценах с неоднородным фоном они малопригодны.

С развитием методов глубокого обучения, нейросетевые модели значительно повысили эффективность и применимость поиска текста в сценариях, отличных от распознавания сканов документов. Лучшие на сегодня модели используют в качестве извлекателя признаков из изображения сверточные архитектуры нейросетей с пирамидой масштабов [7]. Выходом сети могут быть карта плотности вероятности текста [8–11] или непосредственно коллекция объектов (текстовых строк или отдельных слов). Для последнего сегодня, как правило, используются архитектуры типа трансформер с механизмом внимания [12–14]. Приведенные модели способны находить текст в произвольных сценах, однако требуют значительных вычислительных ресурсов и объема памяти. Так, например, CRAFT [8] весит 79 МБ, а более легкий детектор DBNet [10] – 53 МБ. В то же время, модели с использованием модуля трансформер имеют куда больший размер: 522,8 МБ у DPText-DETR [14]. Время работы моделей поиска текста в публикациях зачастую замеряют на высокопроизводительных видеокартах настольных ПК. Доля работ, которые фокусируются на вычислительно эффективных методах поиска текста на центральных процессорах, относительно невелика. В качестве примера такой работы можно отнести PaddleOCR [15]: предложенный ими детектор текста является оптимизированной версией DBNet [10] и имеет размер в 3,6 МБ.

В работе [16] был предложен подход для поиска текстовых регионов на изображениях в социальных медиа для модерации. В его основе лежит простой LeNet-подобный классификатор квадратных областей изображения на текст и фон. Будучи примененным к картинке целиком, на выходе получается карта плотности вероятности текстовых регионов. Особенностью алгоритма является то, что для его обучения не требуются настоящие картинки в полном размере: классификатор обучается

на нарезанных частях изображения, тем самым в модель не закладывается контекст сцены и повышается робастность к компоновке текста и фона. Данный метод «склеивает» строки в абзацы, поэтому для распознавания строк необходим дополнительный алгоритм, который сначала их извлечет из найденных текстовых блоков.

2. Постановка задачи

Из обзора литературы видно, что задача быстрой и вычислительно эффективной детекции текстовых строк на изображении является актуальной. Применение моделей глубокого обучения на мобильных устройствах накладывает ограничения на их вычислительную сложность, энергоэффективность и размер. Также важным является тот факт, что большинство моделей детекторов дообучаются на настоящих данных. Т.к. их сбор и разметка для обучения весьма трудоемкая и дорогостоящая процедура, то есть потребность в методах, для обучения которых можно использовать только синтетические данные.

Постановка задачи заключается в следующем. Имеется входное изображение, содержащее номер телефона. Имеется также алгоритм распознавания текстовых строк и алгоритм сверки распознанной строки на соответствие формату номера телефона. Найденные некоторой моделью поиска текстовые строки распознаются и проверяются на соответствие формату, среди которых затем выбираются наиболее похожие на номер телефона строки. Блок-схема алгоритма распознавания номеров телефонов приведена на рис. 2. В данной работе предлагается подход для быстрой детекции текстовых строк (выделенный блок) при условии, что ложные срабатывания детектора текста будут отфильтрованы за счет распознавания и проверки соответствия строки определённому формату.

Необходимо разработать метод поиска текстовых строк на изображении, который удовлетворяет следующим критериям:

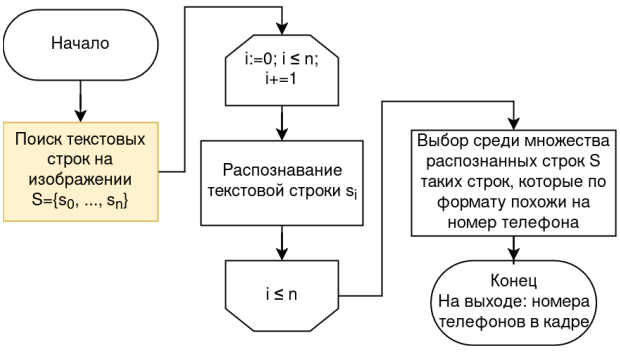


Рис. 2. Блок-схема алгоритма работы системы распознавания номеров телефонов на изображении

1) Быстрая скорость работы и компактный размер модели в сравнении с существующими детекторами текста общего вида;
2) Для обучения не требуются настоящие контекстно-зависимые данные, т.е. изображения сцен с разметкой в виде геометрических зон для вклейки текста или текстовое наполнение строк.

Важно отметить, что количество ложных срабатываний детектора не является оптимизируемой величиной т.к. по результату сверки распознанной строки с форматом номера телефона можно будет отфильтровать ложные срабатывания.

3. Предлагаемый метод

Для решения задачи предлагается использовать нейросетевую модель классификации зон изображения на «текст» и «не текст». Архитектура сети приведена в табл. 1. Она состоит только из сверточных слоев, что дает возможность применять ее к изображениям произвольного размера. В качестве функции активации используется $\mathrm{symrelu}(x) = \max(-1, \min(1, x))$. Модель содержит всего 21 тысячу обучаемых параметров и крайне компактна: в вещественном

Табл. 1

Архитектура предлагаемой модели детектора текста.

#	Тип слоя	Параметры	Функция активации
1	Conv	8 фильтров 4x4, шаг 1x1, без отступов	Symrelu
2	Conv	8 фильтров 1x3, шаг 1x2, без отступов	Symrelu
3	Conv	8 фильтров 3x1, шаг 2x1, без отступов	Symrelu
4	Conv	8 фильтров 3x3, шаг 2x2, отступы 1x1	Symrelu
5	Conv	12 фильтров 3x3, шаг 1x1, без отступов	Symrelu
6	Conv	16 фильтров 3x3, шаг 1x1, без отступов	Symrelu
7	Conv	32 фильтров 3x2, шаг 1x1, без отступов	Symrelu
8	Conv	48 фильтров 3x1, шаг 1x1, без отступов	Symrelu
9	Conv	64 фильтров 3x1, шаг 1x1, без отступов	Symrelu
10	Conv	3 фильтра 5x1, шаг 1x1, без отступов	Softmax

типе (4 байта на значение) веса будут иметь размер всего в 84 КБ.

Обучающая выборка состоит из синтезированных текстовых строк. Синтез происходил путем печати случайных последовательностей символов различными шрифтами на изображения фонов. Преимуществом таких данных является простота их генерации и отсутствие контекста сцены. Важность независимости от контекста при обучении можно продемонстрировать следующим примером. Допустим, для обучения сети для поиска номера телефона были собраны настоящие фото с рынков, мастерских, домов быта и т.п. мест. В превалирующем большинстве случаев, номер телефона для денежного перевода пишут на листе бумаги или картона. При этом количество примеров, когда его пишут на необычной поверхности, такой как стекло или меловая доска, будет значительно меньше. Обучение на такой выборке может привести к переобучению сети под наиболее частый случай, т.е. номеров, записанных на листе бумаги. Таким образом, при обучении на настоящих данных возникает проблема несбалансированной выборки. В предложенном же методе, сеть учится на строках выявлять признаки текста без зависимости от сцены и прочего контекста.

Нейронная сеть при обучении классифицирует каждый столбец входной картинки строки с учетом его окрестности (размером с рецептивное поле сети) на 3 класса: «фон», «символ» и «частично символ». Класс «частично символ» включает в себя стыки отдельных символов или горизонтальные пространства между близкими строками – т.е. области, обладающие признаками текста, но не принадлежащие центру отдельного символа. Похожий подход применялся в детекторе текста CRAFT [8], где аналогичный класс использовался как соединяющее звено между отдельными символами в словах и помогал уменьшить количество ложных срабатываний в высокочастотных областях изображения, имеющих некоторые признаки текстовой строки в целом, но не отдельных символов.

Рассмотрим алгоритм формирования векторов-ответов для изображений текстовых строк в обучающей выборке. Вектор-ответ для изображения $W \times H \times C$ имеет размерность $W \times 1 \times 1$. Порядок его заполнения следующий. Сперва он инициализируется значениями -1. Затем, в геометрический центр по оси X каждого символа ставится индекс класса «символ», т.е. 1. Затем, между соседними символами с вероятностью p_{polt}, в случайной выбранной точке отрезка, указывается индекс 2, соответствующий классу «частично символ». Наконец, проставляются индексы точек фона (значение 0). Они соответствуют как пробелам в строке, так и пустым пространствам слева или справа от строки. Точки фона выбираются случайно, но с условием, что их суммарное количество во всей обучающей выборке должно быть равно k_{bg} × (количество отдельных символов в данных). Иллюстрация заполненного вектора-ответа для изображения приведена на рис. 3. Важным моментом алгоритма является то, что в итоге часть точек вектора-ответа будет иметь индекс -1. Во время обучения сети, функция ошибки не вычисляется в таких точках (ее значение будет равно нулю). Таким образом, необязательность присваивания меток классов всем точкам вектора-ответа делает возможным балансировку количества точек разных классов за счет параметров p_{polt} и k_{bg}. За счет балансирования кол-ва точек классов «символ» и «фон» можно добиться необходимой устойчивости сети к сложному фону. В данной работе были использованы значения $p_{polt} = 0{,}3$ и $k_{bg} = 2$.

Как было сказано ранее, предложенная нейронная сеть имеет полносверточную архитектуру, а потому ее можно применять к изображениям нефиксированного размера. Будучи примененной к произвольному изображению, выход сети будет представлять собой изображение, каждый пиксель которого соответствует некоторой области на входном изображении и содержит вероятность его принадлежности текстовой строке.

Результат детекции в виде прямоугольников найденных текстовых строк получается на этапе

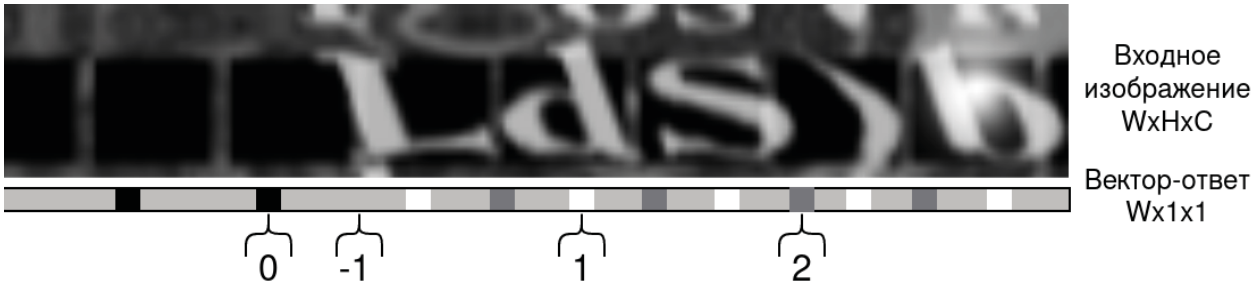
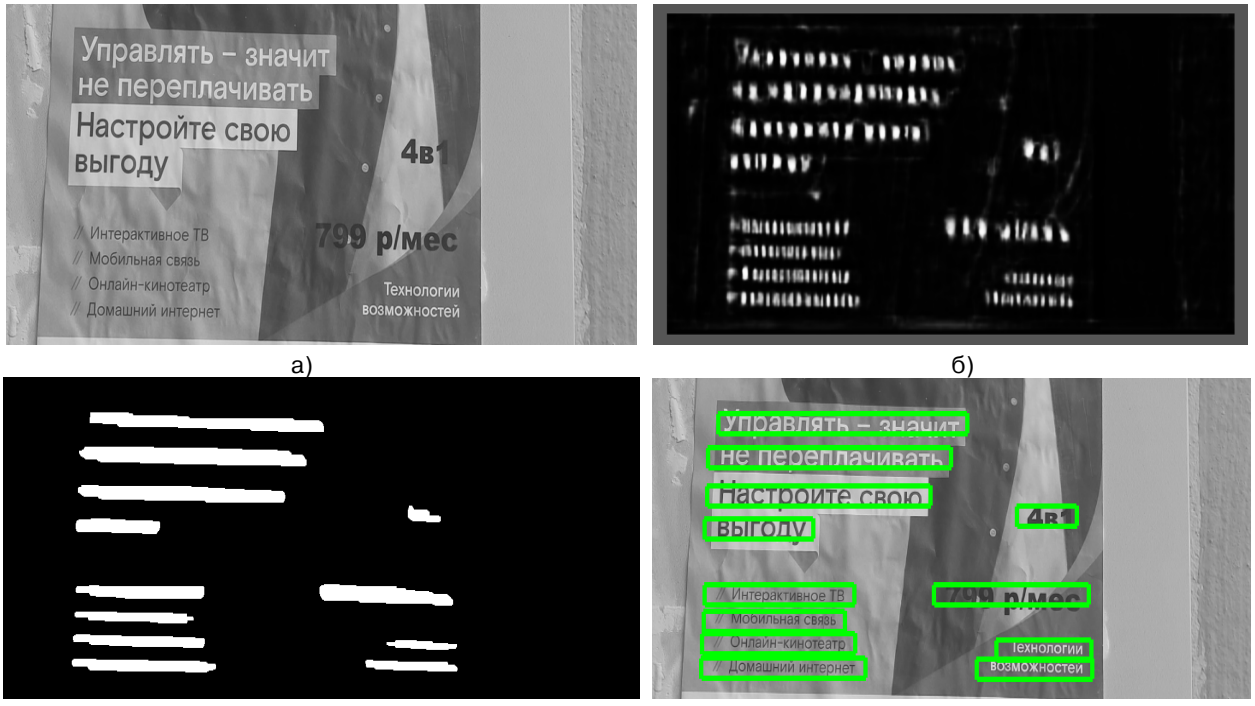


Рис. 3. Пример входного изображения текстовой строки при обучении и вектора-ответа для нее



а) б)

в) г)

Рис. 4. Пример работы предложенного метода детекции текста: а – исходное изображение; б – выход сети в виде тепловой карты текста; в – постобработка выхода сети с помощью морфологических операций и бинаризации; г – результат детекции текста

постобработки выхода сети. Выход сети преобразуется в одноканальное изображение путем взятия значения уверенности класса «символ» в каждой точке. Затем оно приводится к размеру входа и к нему применяются морфологические операции и бинаризация по фиксированному порогу 0,5. После применяется алгоритм поиска связанных компонент, и для каждой компоненты берется ее обрамляющая рамка, которая и является результатом детекции строки. Подробно процесс детекции изображен на рис. 4.

4. Экспериментальные результаты

Обучение модели и замеры вычислительной эффективности производились на ПК с процессором AMD Ryzen 3970X и видеокартой Nvidia RTX 3090. Обучение производилось на 500 тысячах сгенерированных строках размера 256x29x1 в течение 100 эпох. Используемая функция потерь – классическая функция перекрестной энтропии, используемая в задачах классификации. В качестве алгоритма оптимизации использовался стохастический градиентный спуск (SGD) с моментами, скорость обучения была 0.01, значение моментов – 0,9, размер минибатча – 256. Для увеличения репрезентативности данных использовалась аугментация на лету в виде накладывания шумов, бликов, размытия и небольших сдвигов изображения по вертикальной оси [17].

Предложенный метод был протестирован в задаче распознавания номера телефона в кадре. Закрытый набор данных содержал 5264 изображения номеров телефонов в разных сценах, такие как рынки, мастерские, объявления, визитки и т.д. В данных были как печатные, так и рукописные номера. Процесс распознавания был следующий: предложенная модель поиска текста выделяла все возможные текстовые строки на изображении, которые затем распознавались сторонней моделью. Среди распознанных строк в ответ выбиралась та, которая больше всего похожа на номер телефона по формату. Качество распознавания оценивалось как отношение числа правильно распознанных номеров к общему числу. Результат замера качества приведен в табл. 2.

Табл. 2

Результат распознавания номеров телефонов на изображениях с предлагаемой моделью детектора текста

Тип данных	Качество распознавания
Печатные номера	88,33%
Рукописные номера	77,02%
Любые номера	80,35%

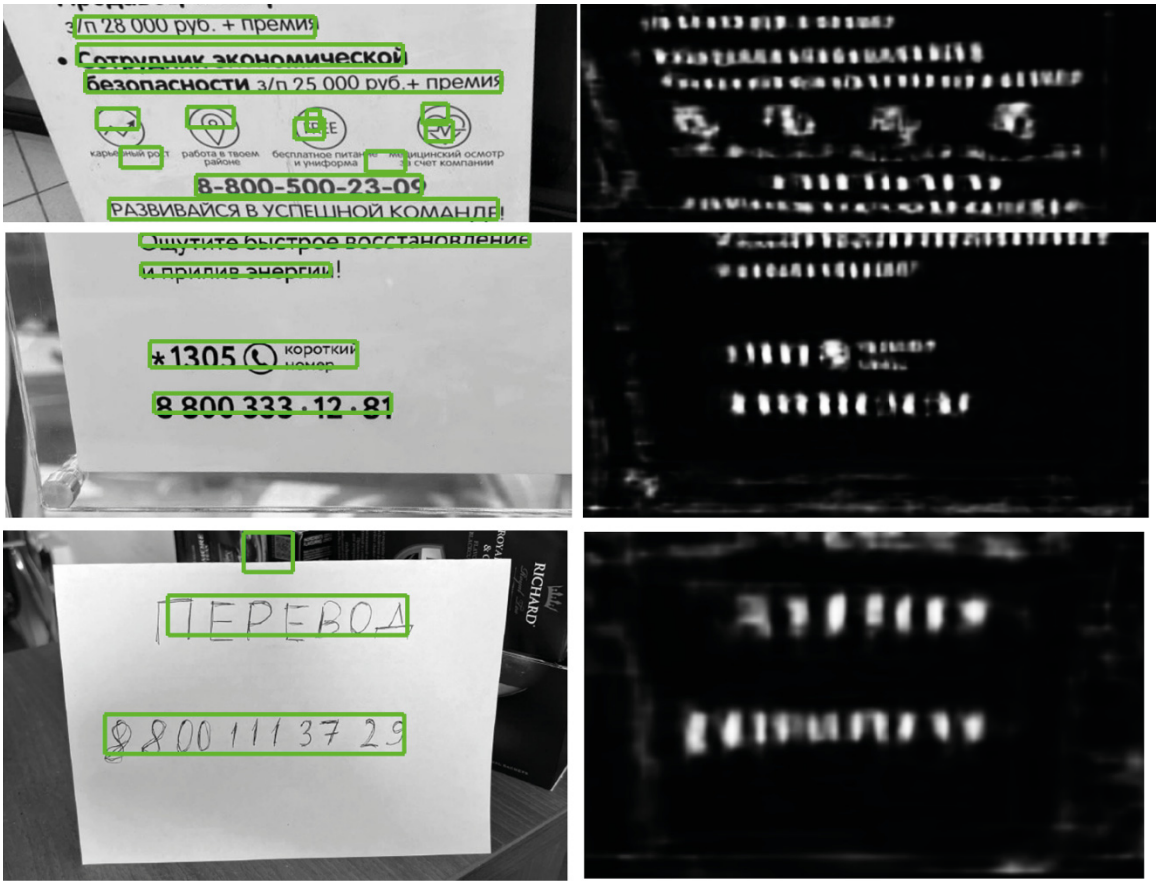


Рис. 5. Примеры работы предложенной модели поиска текста на реальных данных. Слева приведены исходные изображения с найденными рамками строк, справа – тепловая карта текста на основе выхода нейронной сети

Примеры работы предложенного детектора приведены на рис. 5.

Время работы предложенного метода на ПК с процессором AMD Ryzen 3970X на изображении 905x1280 составляет всего 199 мс. Предложенная модель была реализована в коммерческом ПО Smart Code Engine, с использованием проприетарной библиотеки обработки изображений min*. Для исполнения сверточных слоев в модели использовался алгоритм p-im2col [18], позволяющий настроить оптимальное потребление оперативной памяти. Сравнение вычислительной эффективности с другими детекторами текста представлено в табл. 3. Предложенная модель значительно превосходит CRAFT и компактный детектор текста PaddleOCR по размеру модели и пиковому потреблению оперативной памяти (RAM). Рассматриваемые модели находятся в разных нишах как по отличающимся на порядки размерам, так и по способу обучения, из-за чего сравнение предложенного метода с PaddleOCR и CRAFT в плане качества работы не является релевантным.

Недостатком метода являются ложные срабатывания на фоне, которые фильтруются на стадии распознавания. Пример некорректной работы приведен на рис. 6. Также недостатком метода является низкая устойчивость к разномасштабному тексту – для стабильной работы с маленьким и большим

Табл. 3

Сравнение вычислительной эффективности предложенного метода детекции текста с другими детекторами

Модель	**Время работы, мс**	**Размер модели, МБ**	**Пиковое потребление RAM, МБ**
CRAFT [7]	1529	79,3	1100
PaddleOCR [14]	200	3,6	415,6
Предложенная модель	199	0,084	101

текстом необходимо обрабатывать вход в разных масштабах и агрегировать результаты детекции, например алгоритмом подавления не-максимумов.

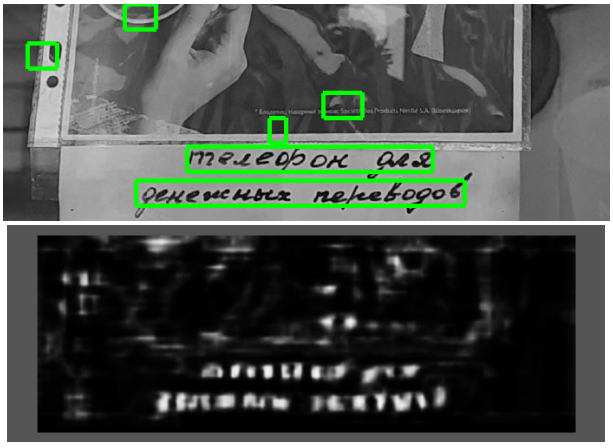


Рис. 6. Пример ложных срабатываний предложенного подхода поиска текста. Лишние найденные рамки на фоне удаляются после этапа распознавания строк

Заключение

В работе представлен быстрый и компактный детектор текста для систем распознавания номеров телефонов. За счет фиксированного формата таких строк можно фильтровать ложные срабатывания детектора на сложном фоне. В сравнении с детектором текста CRAFT, предложенная модель имеет на 3 порядка меньший размер и в 11 раз меньшее потребление оперативной памяти. В сравнении с компактным детектором текста PaddleOCR, предложенная модель меньше в 42 раза (84 КБ против 3,6 МБ) и потребляет 101 МБ оперативной памяти против 415 МБ. Отличительной чертой подхода является независимость от контекста сцены и формата строк при обучении – вместо больших изображений сцен используются сгенерированные изображения строк текста. Таким образом, модель обучается только образам «текст» и «не текст», без учета дополнительной информации сцены. Эффективность метода была продемонстрирована в задаче распознавания номеров телефонов в произвольных сценах – количество правильно найденных и распознанных номеров на закрытом наборе реальных данных составило 80,35%.

В качестве будущей работы по развитию предложенного метода можно выделить повышение устойчивости детектора к разномасштабному тексту, а также уменьшение количества ложных срабатываний, что приведет к меньшему количеству запусков распознающей модели и ускорит систему распознавания номеров телефонов.

Литература

1. *Arlazarov V.L., Slavin O.A.* Issues of recognition and verification of text documents. ITiVS 3. P. 55–61. 2023. DOI: 10.14357/20718632230306.
2. *Bulatov K.B., Emelyanova E.V., Tropin D.V., Skoryukina N.S., Chernyshova Y.S., Sheshkus A.V., Usilin S.A., Ming Z., Burie J.C., Luqman M.M., Arlazarov V.V.* Midv-2020: A comprehensive benchmark dataset for identity document analysis. Computer Optics 46(2). P. 252–270 (2022). DOI: 10.18287/2412-6179-CO-1006.
3. *Okun O., Yan Y., Pietikainen M.* Robust text detection from binarized document images. In: 2002 International Conference on Pattern Recognition. Vol. 3. P. 61–64 vol.3 (2002). https://doi.org/10.1109/ICPR.2002.1047795.
4. *Diem M., Kleber F., Sablatnig R.* Text line detection for heterogeneous documents. In: 2013 12th International Conference on Document Analysis and Recognition. P. 743–747 (2013). https://doi.org/10.1109/ICDAR.2013.152.
5. *dos Santos R.P., Clemente G.S., Ren T.I., Cavalcanti G.D.* Text line segmentation based on morphology and histogram projection. In: 2009 10th International Conference on Document Analysis and Recognition. P. 651–655 (2009). https://doi.org/10.1109/ICDAR.2009.183.
6. *Gatos B., Papamarkos N., Chamzas C.* Skew detection and text line position de- termination in digitized documents. Pattern Recognition 30(9), 1505–1519 (1997). https://doi.org/ https://doi.org/10.1016/S0031-3203(96)00157-4.
7. *Lin T.Y., Dollár P., Girshick R., He K., Hariharan B. and Belongie S.* “Feature Pyramid Networks for Object Detection,” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017. P. 936-944. DOI: 10.1109/CVPR.2017.106.
8. *Baek Y., Lee B., Han D., Yun S., Lee H.* Character region awareness for text detection. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). P. 9357–9366 (06 2019), DOI: 10.1109/CVPR.2019.00959.
9. *Chen Z., Wang J., Wang W., Chen G., Xie E., Luo P., Lu T.* Fast: Faster arbitrarily-shaped text detector with minimalist kernel representation. In: arXiv (2021), 2111.02394.
10. *Liao M., Wan Z., Yao C., Chen K., Bai X.* Real-time scene text detection with differentiable binarization. Proceedings of the AAAI Conference on Artificial Intelligence 34(07), 11474–11481 (Apr 2020). https://doi.org/10.1609/aaai.v34i07.6812.
11. *Liao M., Zou, Z., Wan Z., Yao C., Bai X.* Real-time scene text detection with differentiable

binarization and adaptive scale fusion. arXiv (2022), 2202.10304.
12. *Zhang S.X., Zhu X., Yang C., Yin X.C.* Arbitrary shape text detection via boundary transformer. IEEE Transactions on Multimedia 26. P. 1747–1760 (2022), https://api.semanticscholar.org/CorpusID:248693243.
13. *Bu Q., Park S., Khang M. & Cheng Y.* (2024). SRFormer: Text Detection Transformer with Incorporated Segmentation and Regression. Proceedings of the AAAI Conference on Artificial Intelligence, 38(2). P. 855-863. https://doi.org/10.1609/aaai.v38i2.27844.
14. *Ye M., Zhang J., Zhao S., Liu J., Du B., Tao D.* Dptext-detr: towards better scene text detection with dynamic points in transformer. In: Proceedings of the AAAI Conference on Artificial Intelligence. AAAI'23/IAAI'23/EAAI'23, AAAI Press (2023). https://doi.org/10.1609/aaai.v37i3.25430, https://doi.org/10.1609/aaai.v37i3.25430.
15. *Li C., Liu W., Guo R., Yin X., Jiang K., Du Y., Du Y., Zhu L., Lai B., Hu X., Yu D., Ma Y.* Pp-ocrv3: More attempts for the improvement of ultra lightweight ocr system. ArXiv abs/2206.03001 (2022), https://api.semanticscholar.org/CorpusID:249431435.
16. *Layek A.K., Mandal S., Ghosh S.* (2020). A Fast Approach for Text Region Detection from Images on Online Social Media. In: Das, A., Nayak, J., Naik, B., Pati, S., Pelusi, D. (eds) Computational Intelligence in Pattern Recognition. Advances in Intelligent Systems and Computing, vol 999. Springer, Singapore. https://doi.org/10.1007/978-981-13-9042-5_31.
17. *Gayer A.V., Sheshkus A.V. and Chernyshova Y.S.* "Augmentation on the fly for the neural networks learning," Trudy ISA RAN (Proceedings of ISA RAS), vol. 68, Спецвыпуск № S1. P. 150-157, 2018, DOI: 10.14357/20790279180517.
18. *Trusov A.V., Limonova E.E., Nikolaev D.P. and Arlazarov V.V.* "p-im2col: Simple Yet Efficient Convolution Algorithm with Flexibly Controlled Memory Overhead," IEEE Access, vol. 9. P. 168162-168184, 2021. DOI: 10.1109/ACCESS.2021.3135690.

Гайер Александр Вячеславович. Федеральный исследовательский центр «Информатика и управление» Российской академии наук, г. Москва, Россия. Младший научный сотрудник. ООО «Смарт Энджинс Сервис», г. Москва, Россия. Научный сотрудник-программист. Область научных интересов: глубокое обучение, детекция объектов. E-mail: agayer@smartengines.com

Context-independent fast text detection method for recognizing phone numbers

A.V. Gayer[I,II]

[I] Smart Engines Service LLC, Moscow, Russia

[II] Federal Research Center "Computer Science and Control" of Russian Academy of Sciences, Moscow, Russia

Abstract. Modern methods for detecting text in images are based on computationally expensive deep learning models and require a large amount of training data, including real data. In the case of text retrieval in arbitrary scenarios, the process of collecting and annotating real data for training is extremely labor-intensive and expensive due to the high variability of possible scenes. This paper presents a new method for detecting text in arbitrary images, which does not require photographs of text in real scenes to be trained and can be trained on simple synthetic data in the form of strings. The proposed neural network model is 42 times smaller than the text detector in one of the best text recognition systems in terms of quality and speed, PaddleOCR (84 KB versus 3.6 MB), which makes it an excellent choice for mobile devices. The model was tested as part of a phone number recognition system, where with its help it was possible to achieve 80.35% of correctly recognized numbers.

Keywords: *deep learning, object detection, image segmentation, text detection*
DOI: 10.14357/20790279240305 **EDN: HREWAU**

References

1. *Arlazarov V.L., Slavin O.A.* Issues of recognition and verification of text documents. ITiVS 3. P. 55–61. 2023. DOI: 10.14357/20718632230306.
2. *Bulatov K.B., Emelyanova E.V., Tropin D.V., Skoryukina N.S., Chernyshova Y.S., Sheshkus A.V., Usilin S.A., Ming Z., Burie J.C., Luqman M.M., Arlazarov V.V.* Midv-2020: A comprehensive benchmark dataset for identity document analysis. Computer Optics 46(2), 252–270 (2022). DOI: 10.18287/2412-6179-CO-1006.
3. *Okun O., Yan Y., Pietikainen M.* Robust text detection from binarized document images. In: 2002 International Conference on Pattern Recognition. Vol. 3. P. 61–64 vol.3 (2002). https://doi.org/10.1109/ICPR.2002.1047795.
4. *Diem M., Kleber F., Sablatnig R.* Text line detection for heterogeneous documents. In: 2013 12th International Conference on Document Analysis and Recognition. P. 743–747 (2013). https://doi.org/10.1109/ICDAR.2013.152.
5. *dos Santos R.P., Clemente G.S., Ren T.I., Cavalcanti G.D.* Text line segmentation based on morphology and histogram projection. In: 2009 10th International Conference on Document Analysis and Recognition. P. 651–655 (2009). https://doi.org/10.1109/ICDAR.2009.183.
6. *Gatos B., Papamarkos N., Chamzas C.* Skew detection and text line position de- termination in digitized documents. Pattern Recognition 30(9), 1505–1519 (1997). https://doi.org/ https://doi.org/10.1016/S0031-3203(96)00157-4.
7. *Lin T.Y., Dollár P., Girshick R., He K., Hariharan B. and Belongie S.* "Feature Pyramid Networks for Object Detection," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017. P. 936-944. DOI: 10.1109/CVPR.2017.106.
8. *Baek Y., Lee B., Han D., Yun S., Lee H.* Character region awareness for text detection. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). P. 9357–9366 (06 2019), DOI: 10.1109/CVPR.2019.00959.
9. *Chen Z., Wang J., Wang W., Chen G., Xie E., Luo P., Lu T.* Fast: Faster arbitrarily-shaped text detector with minimalist kernel representation. In: arXiv (2021), 2111.02394.
10. *Liao M., Wan Z., Yao C., Chen K., Bai X.* Real-time scene text detection with differentiable binarization. Proceedings of the AAAI Conference on Artificial Intelligence 34(07), 11474–11481 (Apr 2020). https://doi.org/10.1609/aaai.v34i07.6812.
11. *Liao M., Zou, Z., Wan Z., Yao C., Bai X.* Real-time scene text detection with differentiable binarization and adaptive scale fusion. arXiv (2022), 2202.10304.
12. *Zhang S.X., Zhu X., Yang C., Yin X.C.* Arbitrary shape text detection via boundary transformer. IEEE Transactions on Multimedia 26, 1747–1760 (2022), https://api.semanticscholar.org/CorpusID:248693243.
13. *Bu Q., Park S., Khang M. & Cheng Y.* (2024). SRFormer: Text Detection Transformer with Incorporated Segmentation and Regression. Proceedings of the AAAI Conference on Artificial Intelligence, 38(2), 855-863. https://doi.org/10.1609/aaai.v38i2.27844.
14. *Ye M., Zhang J., Zhao S., Liu J., Du B., Tao D.* Dptext-detr: towards better scene text detection with dynamic points in transformer. In: Proceedings of the AAAI Conference on Artificial Intelligence. AAAI'23/IAAI'23/EAAI'23, AAAI Press (2023). https://doi.org/10.1609/aaai.v37i3.25430, https://doi.org/10.1609/aaai.v37i3.25430.
15. *Li C., Liu W., Guo R., Yin X., Jiang K., Du Y., Du Y., Zhu L., Lai B., Hu X., Yu D., Ma Y.* Pp-ocrv3: More attempts for the improvement of ultra lightweight ocr system. ArXiv abs/2206.03001 (2022), https://api.semanticscholar.org/CorpusID:249431435.
16. *Layek A.K., Mandal S., Ghosh S.* (2020). A Fast Approach for Text Region Detection from Images on Online Social Media. In: Das, A., Nayak, J., Naik, B., Pati, S., Pelusi, D. (eds) Computational Intelligence in Pattern Recognition. Advances in Intelligent Systems and Computing, vol 999. Springer, Singapore. https://doi.org/10.1007/978-981-13-9042-5_31.
17. *Gayer A.V., Sheshkus A.V. and Chernyshova Y.S.* "Augmentation on the fly for the neural networks learning," Trudy ISA RAN (Proceedings of ISA RAS), vol. 68, Спецвыпуск № S1. P. 150-157, 2018, DOI: 10.14357/20790279180517.
18. *Trusov A.V., Limonova E.E., Nikolaev D.P. and Arlazarov V.V.* "p-im2col: Simple Yet Efficient Convolution Algorithm with Flexibly Controlled Memory Overhead," IEEE Access, vol. 9. P. 168162-168184, 2021. DOI: 10.1109/ACCESS.2021.3135690.

Gayer A.V. Junior researcher at the Federal Research Center "Computer Science and Control" of Russian Academy of Sciences, Moscow, Russia. Researcher-programmer at Smart Engines Service LLC, Moscow, Russia. Number of publications: 13. Research interests: deep learning, object detection. E-mail: agayer@smartengines.com