УДК 004.8.81
DOI: 10.17586/0021-3454-2024-67-9-767-775

РАСПОЗНАВАНИЕ РУКОПИСНОГО ТЕКСТА ИСТОРИЧЕСКИХ ДОКУМЕНТОВ С ПРИМЕНЕНИЕМ ТЕХНОЛОГИЙ ГЛУБОКИХ НЕЙРОННЫХ СЕТЕЙ

А. М. Унтерберг*, А. В. Пятаева, С. С. Замыслова, Е. Д. Рукосуева, К. В. Богданов

Сибирский федеральный университет, Красноярск, Россия
** unterberg2012@gmail.com*

Аннотация. Рассматривается задача распознавания рукописного текста на дореформенном русском языке с применением технологий глубоких нейронных сетей. В качестве исходных данных использованы отсканированные JPG-снимки исторических документов, в частности XIX века, содержащие различные шумы и помехи, что затрудняет работу алгоритма распознавания. Распознавание текста выполнено в три этапа: устранение шумов, сегментация (выделение) строк текста на изображении, так как входными данными для работы глубокой нейронной сети являются именно строки, и затем распознавание текста выделенных срок с помощью дообученной модели Tesseract OCR, осуществляющей электронный перевод изображений рукописного или печатного текста в текстовые данные. В качестве модели использована сверточно-рекуррентная нейронная сеть; модель представляет собой комбинацию сверточной нейронной сети для извлечения локальных признаков из изображения и рекуррентной нейронной сети, представленной двумя слоями двунаправленных сетей LSTM для обработки последовательности. Использование именно такой модели позволяет достоверно распознавать рукописный текст.

Ключевые слова: *нейронные сети, обработка естественного языка, исторические документы, глубокое обучение, библиотека Tesseract OCR*

Ссылка для цитирования: *Унтерберг А. М., Пятаева А. В., Замыслова С. С., Рукосуева Е. Д., Богданов К. В.* Распознавание рукописного текста исторических документов с применением технологий глубоких нейронных сетей // Изв. вузов. Приборостроение. 2024. Т. 67, № 9. С. 767–775. DOI: 10.17586/0021-3454-2024-67-9-767-775.

HANDWRITTEN TEXT RECOGNITION OF HISTORICAL DOCUMENTS USING DEEP NEURAL NETWORK TECHNOLOGIES

A. M. Unterberg*, A. V. Pyataeva, S. S. Zamyslova, E. D. Rukosueva, K. V. Bogdanov

Siberian Federal University, Krasnoyarsk, Russia
** unterberg2012@gmail.com*

Abstract. The application of deep neural network technologies to the problem of handwriting recognition in pre-reform Russian is considered. The initial data used are scanned JPG images of historical documents from the 19th century, in particular containing various noises and interference, which complicates the work of the recognition algorithm. Text recognition is performed in three stages: noise removal, segmentation (highlighting) of text lines in the image, since the input data for the deep neural network are precisely the lines, and then recognition of the text of the highlighted lines using the pre-trained Tesseract OCR model, which performs electronic translation of images of handwritten or printed text into text data. The model used is a convolutional recurrent neural network; the model is a combination of a convolutional neural network for extracting local features from an image and a recurrent neural network represented by two layers of bidirectional LSTM networks for processing the sequence. Using this model allows for reliable recognition of handwritten text.

Keywords: neural networks, natural language processing, historical documents, deep learning, Tesseract — OCR Library

For citation: Unterberg A. M., Pyataeva A. V., Zamyslova S. S., Rukosueva E. D., Bogdanov K. V. Handwritten text recognition of historical documents using deep neural network technologies. *Journal of Instrument Engineering*. 2024. Vol. 67, N 9. P. 767–775 (in Russian). DOI: 10.17586/0021-3454-2024-67-9-767-775.

Введение. Сохранение для потомков информации о различных исторических событиях является важной задачей не только для ученых-историков, но и для человечества в целом. Бумага и чернила, которые использовались при написании исторических документов, разрушаются со

временем, и хронология событий тех лет может быть навсегда утеряна. Преобразование таких документов из бумажного вида в цифровой формат позволит избежать утраты информации о важных исторических фактах.

Выполнение перевода текста с физического носителя на цифровой — процесс, требующий значительных человеческих затрат. Отсканированные исторические документы могут содержать различные артефакты, шумы либо информацию, тяжело воспринимаемую современным читателем; например, в книге „Акты, собранные в библиотеках и архивах Российской империи. Том I“, где собраны документы конца XIX века, встречается и рукописный текст на дореформенном русском языке, и машинописный текст; в отсканированных письмах Петра I встречается текст, написанный трудночитаемым почерком, а также текст с частично выцветшими чернилами. Автоматизация процесса распознавания текста, а затем и преобразование его в вид, аналогичный оригинальному, позволит решить задачу сохранения информации.

В настоящей статье эта задача решается на примере текста отчетов губернаторов Енисейской губернии XVIII–XX веков. Эти документы ценны для изучения истории региона и для понимания политических, экономических, социальных и культурных процессов, происходивших здесь в определенный период времени. Отчеты состоят в основном из текстовых сообщений и таблиц. Ранние отчеты — рукописные, а более поздние содержат машинописный текст. Особую трудность при распознавании документов представляет рукописный текст, так как требуется настройка алгоритма на распознавание особенностей почерка в представленных текстах.

Литературный обзор. Задача распознавания рукописных текстов исторических документов может быть решена различными способами. Современное решение заключается в применении технологий глубокого обучения для обработки текста, написанного на естественном языке. Так, в работе [1] исследуется сложность обработки текста, размещенного на разных частях страницы и имеющего различные направления чтения, представлена новая нейронная модель, которая может быть эффективно использована для локализации текста, его транскрипции и распознавания символов в документах. Сравнение рекуррентной (RNN) и сверточной (CNN) нейронных сетей для задачи распознавания рукописного текста выполнено в работе [2], показано, что CNN-модель имеет преимущество по скорости. В [3] представлена прогрессивная методика обучения для распознавания рукописного текста, имеющая ограниченные ресурсы. Авторы исследуют проблему ограниченного объема доступных обучающих данных и предлагают метод, который может эффективно обучаться при малом количестве данных. В [4] выполнены исследования построчного распознавания текста, описываются особенности, возможности и преимущества нерекуррентных моделей. Стратегия аугментации данных под названием “CycleAugment” для распознавания рукописного текста в изображениях исторических документов представлена в [5]. Методика “CycleAugment” основана на циклической генерации различных вариантов аугментированных изображений с помощью преобразований, таких как поворот, масштабирование, сдвиг и изменение контрастности. В [6] представлен новый подход к улучшению автономного распознавания рукописного текста исторических документов при наличии ограниченного числа размеченных строк; описывается методика, основанная на комбинировании двух подходов: “Transfer Learning” и “Active Learning”, и по результатам исследований показана эффекттивность методики. В работе [7] представлены результаты исследования, в ходе которого авторы использовали большой набор данных, состоящий из цифровых изображений исторических книг, созданных с помощью различных технологий печати. На основе этого набора с применением методов обучения, в частности CNN, были созданы модели, которые могут быть использованы для автоматической классификации технологий печати исторических книг на основе цифровых изображений, что способствует сохранению и изучению культурного наследия. Рассматриваемые в [8] различные методы и подходы, используемые для извлечения текста из изображений, включают как классические методы, например шаблонное сопоставление, статистические модели и правила, так и современные методы на основе глубокого обучения — RNN и CNN. Также в работе освещаются некоторые проблемы,

связанные с технологией OCR (Optical Character Recognition — оптическое распознавание символов), такие как неточность распознавания нестандартных символов и шум на изображениях, предлагаются некоторые пути решения этих проблем: например, применение ансамблей моделей, комбинирование классических и глубоких методов, использование контекстуальной информации для повышения точности распознавания. В [9] представлен обзор основанных на глубоком обучении методов и подходов, используемых для анализа и распознавания исторических документов. Методы включают в себя различные архитектуры нейронных сетей, такие как CNN, RNN, сети долгой краткосрочной памяти (LSTM). Также приведены рекомендации по подготовке и созданию наборов данных для использования в исследованиях. В работе [10] описывается метод повышения точности программы для распознавания текста на изображения — OCR-движка Tesseract 4.0 — с использованием предварительной обработки на основе свертки. В [11] представлены результаты исследования, показывающие эффективность применения различных архитектур нейронных сетей — CNN, RNN и CRNN — для распознавания текста на изображениях.

Таким образом, глубокие нейронные сети являются современным и наиболее эффективным способом распознавания рукописных текстов исторических документов.

Распознавание текста отчетов. Исходные изображения исследуемых отчетов получены с использованием ресурсов библиотеки Президента РФ. Эти изображения представляют собой отсканированный разворот страниц или одну страницу оригинального документа. Пример изображения из отчета показан на рис. 1.

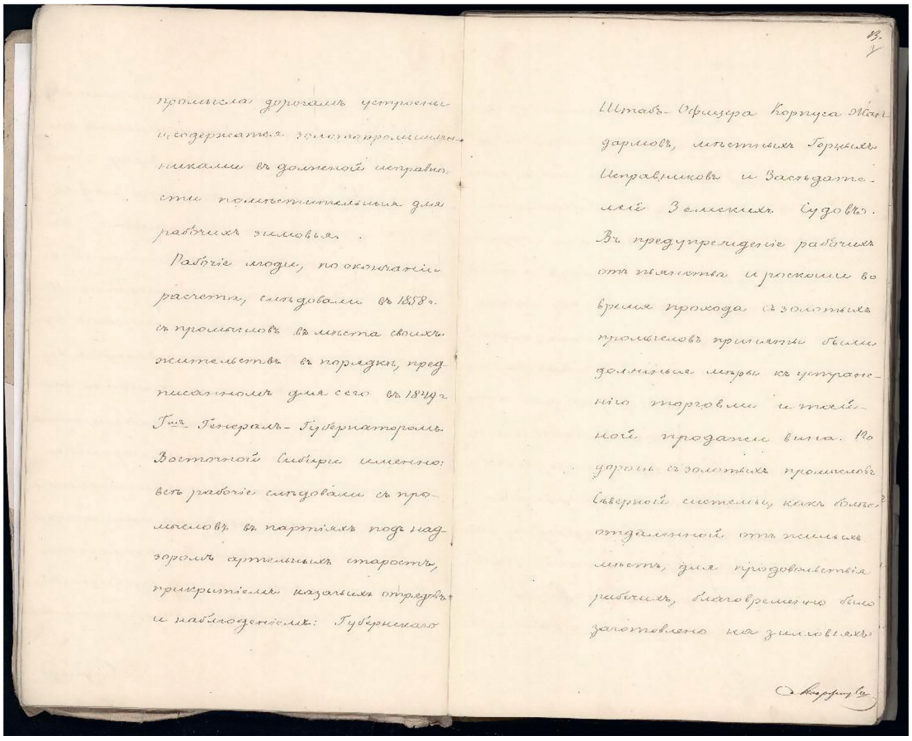


Рис. 1

Снимки документов, как уже упоминалось, могут содержать различные шумы: артефакты сканирования (по краям листа и от переплета по центру фотографии); следы от вмятин бумаги; помехи, связанные с тем, что в качестве письменного инструмента использовалось гусиное перо и капли чернил попадали на бумагу. Такие шумы могут значительно снижать качество алгоритма распознавания текста, поэтому их необходимо устранить. Таким образом, обработка изображений для распознавания текста включает в себя удаление шумов, выделение строк и применение глубоких нейронных сетей. Схема преобразования отчетов в цифровой вид, включающая перечисленные этапы, показана на рис. 2.

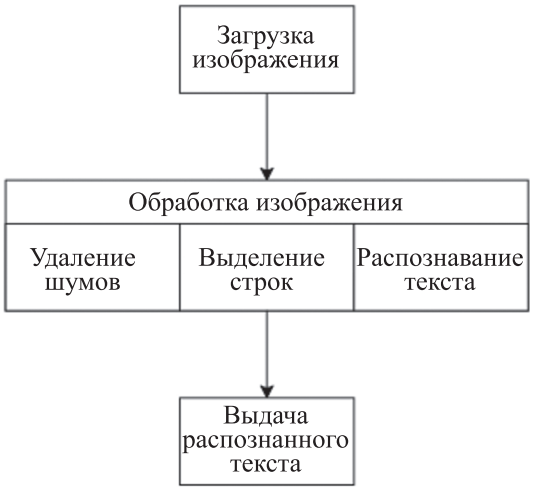


Рис. 2

Этап выделения строк текста на изображении необходим, так как входными данными для работы глубокой нейронной сети являются именно строки. Для того чтобы оставить только строки текста, необходимо „убрать“ с изображения все лишние элементы, что можно сделать с помощью полей серого цвета по краям изображения. Создание полей необходимо для отделения символов текста от различных шумов. Для выделения строк разработан алгоритм преобразования исходного изображения, содержащий следующие шаги (рис. 3).

Шаг 1. Исходное RGB-изображение (см. рис. 1) преобразуется в grayscale (оттенки серого)-

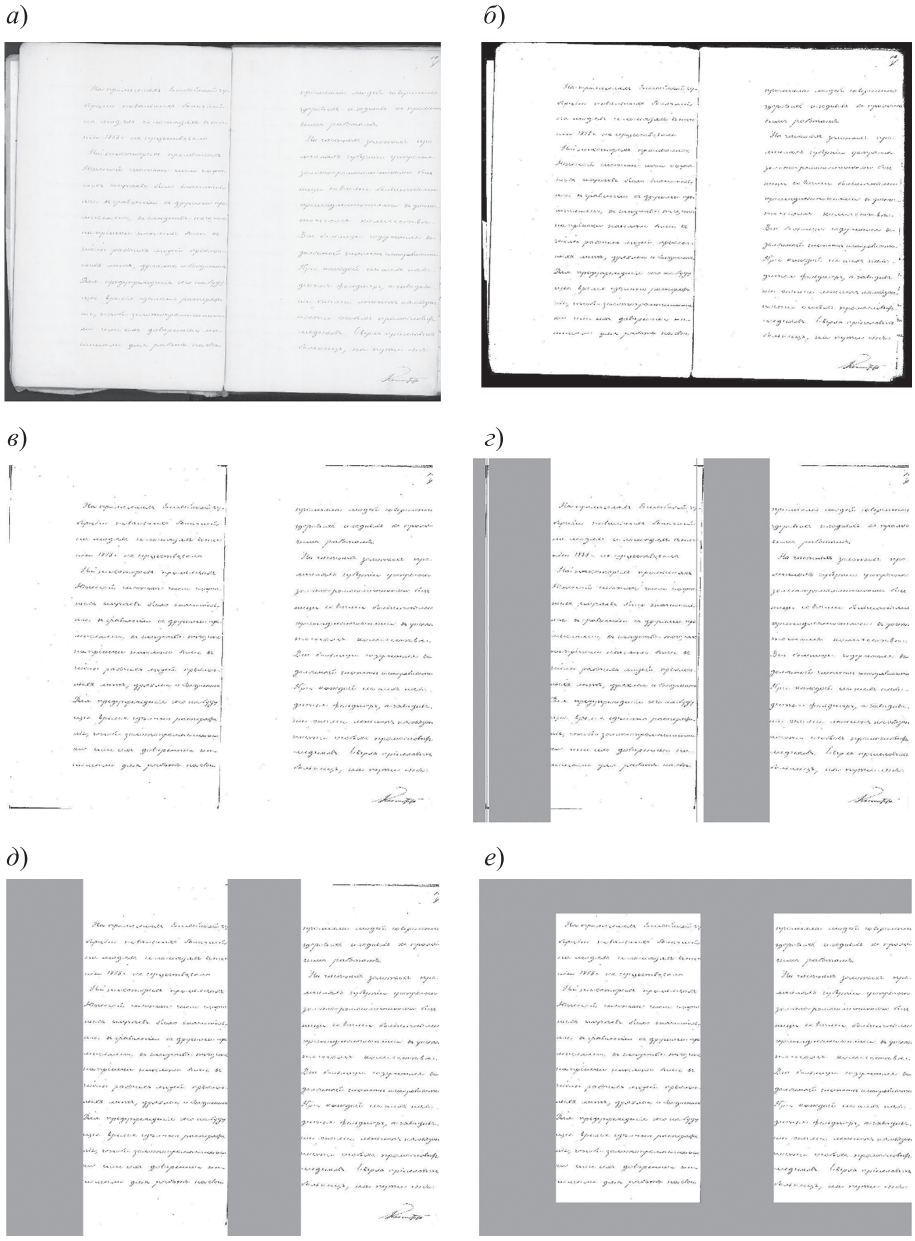


Рис. 3

изображение (рис. 3, *а*). Преобразование осуществляется с помощью функции библиотеки OpenCV — cv.cvtColor(image, cv.COLOR_BGR2GRAY).

Шаг 2. Grayscale-изображение преобразуется в бинарное изображение (рис. 3, *б*). Преобразование осуществляется с помощью функции библиотеки OpenCV — cv.threshold(image,187,255,cv.THRESH_BINARY).

Шаг 3. Черные поля закрашиваются в белый (рис. 3, *в*). Для осуществления данного этапа необходимо найти все горизонтальные и вертикальные столбцы пикселов на изображении, которые полностью состоят из черных пикселов, после чего найденные столбцы необходимо перекрасить в белый путем присвоения пикселам нового значения, равного 255.

Шаг 4. Все белые вертикальные столбцы закрашиваются в серый (рис. 3, *г*).

Шаг 5. Закрашиваются промежутки между серыми столбцами (рис. 3, *д*); пп. 4 и 5 выполняются для отделения частей изображения, не относящихся к тексту, от частей изображения, содержащих текст.

Шаг 6. Крайние верхние и нижние горизонтальные столбцы изображения, не относящиеся к тексту (количество белых пикселов превышает количество черных), закрашиваются в серый (рис. 3, *е*).

Шаг 7. Части изображения, не серые, делятся на отдельные строки (рис. 4). С помощью вложенных циклов обрабатываются все пикселы изображения: внешний цикл проверяет строки пикселов сверху вниз, а внутренний цикл — пикселы в каждой строке слева направо. Определяются начальные и конечные координаты строк.

Шаг 8. Создается список из начальных и конечных координат полученных строк, необходимый для обрезания исходного изображения по координатам.

Шаг 9. Строки делятся на отдельные изображения, сохраняются в папке images line и передаются для распознавания. Деление строк осуществляется с помощь команды `image[y1:y2, x1:x2]`, где `y1` и `x1` — начальные координаты строки, а `y2` и `x2` — конечные.

Шаг 10. Выделенные строки сохраняются в отдельные изображения для загрузки каждого из них в Tesseract-OCR.

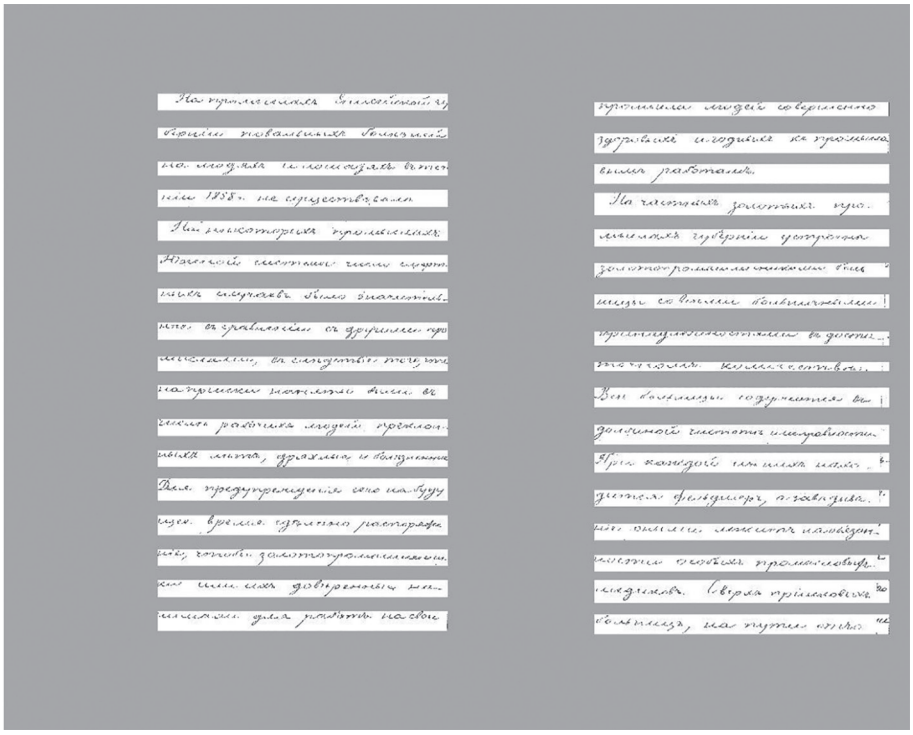


Рис. 4

В качестве модели использована сверточно-рекуррентная нейронная сеть CRNN, представляющая собой комбинацию CNN для извлечения локальных признаков из изображения и RNN, представленной двумя слоями двунаправленных LSTM для обработки последовательности. Использование именно такой модели позволяет достоверно распознавать рукописный текст.

Программная реализация и экспериментальные исследования. Для автоматизации процесса распознавания рукописных текстов отчетов было создано веб-приложение для работы с фотографиями документов. Приложение написано на языке Python, дизайн выполнен с помощью библиотеки TKinter. Программа, реализующая алгоритм распознавания текста, может принимать на вход файлы изображений форматов JPG, JPEG, TIFF, TIF, PNG.

Распознавание текста отчетов выполняется согласно разработанному алгоритму в несколько этапов: фильтрация изображения для устранения шумов, сегментация строк текста и применение нейронной сети для распознавания текста. Устранение шумов реализовано с помощью функции fastNIMeansDenoisingColored() библиотеки OpenCV. Эта функция сначала преобразует RGB-изображение в цветовое пространство CIELAB, а затем отдельно удаляет основные виды шума.

Для сегментации строк реализован свой алгоритм выделения их на изображении. Сначала изображение конвертируется в бинарное двумя функциями библиотеки OpenCV: cvtColor(), threshold(). Далее реализуется алгоритм, меняющий пикселы, находящиеся на определенном расстоянии от основного текста, с белого на серый цвет, после чего выделенные строки текста сохраняются в отдельный файл в формате JPG.

Программная реализация распознавания текста с помощью глубоких нейронных сетей создана посредством библиотеки Pytesseract. Основным компонентом библиотеки Pytesseract является Tesseract OCR Engine — мощный инструмент с открытым исходным кодом, который необходимо установить, чтобы использовать Pytesseract . Эта библиотека хорошо зарекомендовала себя для решения задачи распознавания машинописного текста. Для распознавания рукописного текста потребовалось дообучение модели с помощью встроенных утилит tesseract. Для этого был создан набор обучающих данных, состоящий из следующих пар: изображения текста, взятого из отчетов, и текстового файла, содержащего текст с изображения. Для корректной работы алгоритма имена файлов изображения и текстового файла должны быть одинаковыми.

Данные для обучения модели были взяты из исследуемых отчетов. Для эксперимента было создано три набора данных:

1) набор данных, содержащий строки, — состоит из 500 изображений строк из отчетов и 500 текстовых файлов с распознанным текстом для каждого изображения;

2) набор данных, состоящий из отдельных слов, — состоит из 1000 изображений, на каждом из которых представлено только одно слово; к каждому слову прилагается текстовый файл с распознанным словом; тестирование модели в дальнейшем выполнялось на фотографиях отчетов, содержащих только эти слова;

3) набор данных, содержащий буквы, — состоит из 1500 изображений, содержащих буквы из отчетов, а также 1500 текстовых файлов, содержащих текст с изображений.

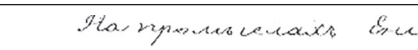
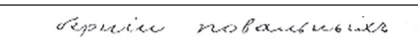
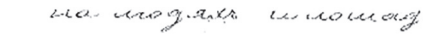
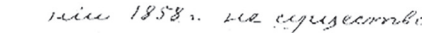
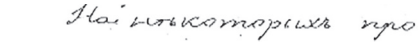
Тестирование модели выполнено с помощью 200 файлов (100 изображений и 100 текстовых файлов), взятых случайным образом из каждого из наборов данных. Изображения поступали на вход обученной модели, после чего генерировался ответ, который сравнивался с текстовым файлом из набора данных. Сравнение результатов выполнено с помощью двух метрик: CRR и WRR. Метрика WRR (Word Recognition Rate) показывает точность распознавания, которая определяется как процент правильно распознанных слов. Метрика CRR (Character Recognition Rate) показывает точность распознавания как отношение количества правильно распознанных символов к их общему количеству. Данные метрики показывают количество ошибок, допущенных нейронной сетью. В табл. 1 показаны результаты обучения модели на разных наборах данных с количеством итераций, равным 50 000.

Таблица 1

Метрика	Точность распознавания, %, на наборе данных		
	Строки	Слова	Буквы
CRR	99,8547	32,6979	16,9969
WRR	95,4640	14,9961	1,0699

Согласно результатам экспериментальных исследований качество распознавания текста моделью, обученной на отдельных строках, существенно выше, чем при ее обучении на словах или буквах, поэтому для построения системы распознавания текста была использована нейронная сеть, обученная именно на этом наборе данных. В табл. 2 приведены результаты работы алгоритма — примеры распознавания строк текста при обучении на наборах строк.

Таблица 2

Изображение	Результат
	На промыслахъ Енисейской гу
	берніи повальныхъ болезней
	на людяхъ и лошадяхъ вътече
	ніи 1858 г. не существовало
	На некоторыхъ промыслахъ

Распознанные таким образом строки текста объединяются и помещаются в отдельный текстовый файл. При этом отсутствующие в современном письменном русском языке символы заменяются на их современные аналоги, кроме символа „і“. Например, символы „Ѣ, ѣ“ заменены на „е“.

Заключение. Представлены подробные результаты по распознаванию рукописных нечетких текстов. Распознавание выполнено в три этапа: устранение шумов, сегментация строк текста, распознавание текста выделенных строк. Удаление шумов реализовано с помощью библиотеки OpenCV, для выделения строк создан оригинальный алгоритм на языке Python, распознавание текста выполнено с помощью дообученной модели библиотеки Tesseract OCR. В ходе разработки создано приложение с веб-интерфейсом. В дальнейшим планируется создание алгоритма сохранения верстки оригинального документа и перевод текста с дореформенного русского языка на современный для популяризации истории и обеспечения доступности исторических документов.

СПИСОК ЛИТЕРАТУРЫ

1. *Carbonell M., Fornés A., Villegas M., Lladós J.* A neural model for text localization, transcription and named entity recognition in full pages // Pattern Recognition Letters. 2020. Vol. 136. P. 219–227. DOI: 10.1016/j.patrec.2020.05.001.
2. *Mestha P., Asif S., Mayekar M*. Handwritten Text Line Recognition Using Deep Learning // Lecture Notes in Networks and Systems. 2022. P. 567–580. DOI: 10.1007/978-3-030-84760-9_48.
3. *Souibgui M. A., Fornes A., Kessentini Y., Megyesi B.* Few shots are all you need: A progressive learning approach for low resource handwritten text recognition // Pattern Recognition Letters. 2022. Vol. 160. P. 43–49. DOI: 10.1016/j.patrec.2022.06.003.
4. *Kang L., Riba P., Rusinol M., Fornes A., Villegas M.* Pay attention to what you read: Non-recurrent handwritten text-Line recognition //Pattern Recognition. 2022. Vol. 129. P. 108766. DOI: 10.1016/j.patcog.2022.108766.
5. *Gonwirat S., Surinta O*. CycleAugment: Efficient data augmentation strategy for handwritten text recognition in historical document images // Engineering and Applied Science Research. 2022. Vol. 49, N. 4 P. 505–520. DOI: 10.14456/easr.2022.50.
6. *Aradillas J., Murillo-Fuentes J., Olmos P*. Boosting Offline Handwritten Text Recognition in Historical Documents with Few Labeled Lines // IEEE Access. 2021. Vol. 9. P. 76674–76688. DOI: 10.1109/ACCESS.2021.3082689.

7. *Im C., Kim Y., Mandl T.* Deep learning for historical books: classification of printing technology for digitized images // Multimedia Tools and Applications. 2022. Vol. 81. P. 5867–5888. DOI: 10.1007/s11042-021-11754-7.
8. *Jiju A., Tuscano S., Badgujar C.* OCR Text Extraction // Intern. Journal of Engineering and Management Research. 2021. Vol. 11. P. 83–86. DOI:10.31033/ijemr.11.2.11.
9. *Lombardi F., Marinai S.* Deep Learning for Historical Document Analysis and Recognition — A Survey // Journal of Imaging. 2020. Vol. 6, N. 10. P. 110 DOI: 10.3390/jimaging6100110.
10. *Sporici D., Cusnir E., Boiangiu C.* Improving the accuracy of Tesseract 4.0 OCR engine using convolution-based preprocessing // Symmetry. 2020. Vol. 12, N. 5. P. 715. DOI: 10.3390/sym12050715.
11. *Pyataeva A. V., Genza S. A.* Artificial neural network technology for text recognition // CEUR Workshop Proc. 2019. Vol. 2534. P. 248–252.

СВЕДЕНИЯ ОБ АВТОРАХ

Александр Максимович Унтерберг — студент; Сибирский федеральный университет, Институт космических и информационных технологий, кафедра систем искусственного интеллекта; E-mail: unterberg2012@gmail.com

Анна Владимировна Пятаева — канд. техн. наук, доцент; Сибирский федеральный университет, Институт космических и информационных технологий, кафедра систем искусственного интеллекта; руководитель научно-учебной лаборатории систем искусственного интеллекта; E-mail: anna4u@list.ru

Светлана Сергеевна Замыслова — студентка; Сибирский федеральный университет, Институт космических и информационных технологий, кафедра систем искусственного интеллекта; E-mail: zamyslova_svetlana_17-05@mail.ru

Екатерина Дмитриевна Рукосуева — студентка; Сибирский федеральный университет, Институт космических и информационных технологий, кафедра систем искусственного интеллекта; E-mail: rukosuevakatya@gmail.com

Константин Валерьевич Богданов — канд. техн. наук, доцент; Сибирский федеральный университет, Институт космических и информационных технологий, кафедра программной инженерии; kbogdanov@sfu-kras.ru

Поступила в редакцию 30.03.2024; одобрена после рецензирования 11.04.2024; принята к публикации 23.07.2024.

REFERENCES

1. Carbonell M., Fornés A., Villegas M., Lladós J. *Pattern Recognition Letters*, 2020, vol. 136, pp. 219–227, DOI: 10.1016/j.patrec.2020.05.001.
2. Mestha P., Asif S., Mayekar M. *Lecture Notes in Networks and Systems*, 2022, pp. 567–580, DOI: 10.1007/978-3-030-84760-9_48.
3. Souibgui M.A., Fornes A., Kessentini Y., Megyesi B. *Pattern Recognition Letters*, 2022, vol. 160, pp. 43–49, DOI: 10.1016/j.patrec.2022.06.003.
4. Kang L., Riba P., Rusinol M., Fornes A., Villegas M. *Pattern Recognition*, 2022, vol. 129, art. no. 108766, DOI: 10.1016/j.patcog.2022.108766.
5. Gonwirat S., Surinta O. *Engineering and Applied Science Research*, 2022, no. 4(49), pp. pp. 505–520, DOI: 10.14456/easr.2022.50.
6. Aradillas J., Murillo-Fuentes J., Olmos P. *IEEE Access*, 2021, vol. 9, pp. 76674–76688, DOI: 10.1109/ACCESS.2021.3082689.
7. Im C., Kim Y., Mandl T. *Multimedia Tools and Applications*, 2022, vol. 81, pp. 5867–5888, DOI: 10.1007/s11042-021-11754-7.
8. Jiju A., Tuscano Sh., Badgujar Ch. *International Journal of Engineering and Management Research*, 2021, no. 2(11), pp. 83–86, DOI:10.31033/ijemr.11.2.11.
9. Lombardi F., Marinai S. *Journal of Imaging*, 2020, no. 6(10), pp. 110, DOI: 10.3390/jimaging6100110.
10. Sporici D., Cusnir E., Boiangiu C. *Symmetry*, 2020, no. 5(12), pp. 715, DOI: 10.3390/sym12050715.
11. Pyataeva A.V., Genza S.A. *CEUR Workshop Proceedings*, 2019, vol. 2534, pp. 248–252.

DATA ON AUTHORS

Aleksander M. Unterberg — Student; Siberian Federal University, Institute of Space and Information Technologies, Department of Artificial Intelligence Systems; E-mail: unterberg2012@gmail.com

Anna V. Pyataeva — PhD, Associate Professor; Siberian Federal University, Institute of Space and Information Technologies, Department of Artificial Intelligence Systems; E-mail: anna4u@list.ru

Svetlana S. Zamyslova — Student; Siberian Federal University, Institute of Space and Information Technologies, Department of Artificial Intelligence Systems; E-mail: zamyslova_ svetlana_17-05@mail.ru

Ekaterina D. Rukosueva — Student; Siberian Federal University, Institute of Space and Information Technologies, Department of Artificial Intelligence Systems; E-mail: rukosuevakatya@gmail.com

Konstantin V. Bogdanov — PhD, Associate Professor; Siberian Federal University, Institute of Space and Information Technologies, Department of Software Engineering; E-mail: kbogdanov@sfu-kras.ru

Received 30.03.2024; approved after reviewing 11.04.2024; accepted for publication 23.07.2024.