

УДК 004.93'12

DOI: 10.22213/2410-9304-2026-2-71-79

Распознавание рукописного церковнославянского текста на примере Остромирова Евангелия (РНБ, Ф.п.1.5)

В. И. Терещенко, ИжГТУ имени М. Т. Калашникова, Ижевск, Россия

М. А. Комышев, студент, ИжГТУ имени М. Т. Калашникова, Ижевск, Россия

С. В. Вологдин, доктор технических наук, доцент, ИжГТУ имени М. Т. Калашникова, Ижевск, Россия

В. В. Сяктерева, кандидат технических наук, доцент, ИжГТУ имени М. Т. Калашникова, Ижевск, Россия

В данной статье рассматривается разработка инструмента для распознавания текстов церковнославянских рукописей на материале Остромирова Евангелия XI в. (РНБ, Ф.п.1.5). Для решения задачи была выбрана нейронная сеть архитектуры YOLOv8. Объектом распознавания являются 42 кириллические буквы и простое титло.

Был разработан эвристический алгоритм для построения текста из неупорядоченного множества объектов, определяемых YOLO. Алгоритм построчно собирает текст снизу вверх на основании координат ограничивающих рамок и выводит его пользователю в соответствии со стандартом Юникод.

Финальная модель обучалась в течение двух эпох на 1680 изображениях, составленных мозаичным способом из символов различных страниц рукописи. Для сокращения разрыва между числом экземпляров символов также размещались искусственно воссозданные варианты редких букв, например ферт. На первых 200 страницах Остромирова Евангелия получена совокупная метрика CER (CharacterErrorRate) на уровне 4,09 % (не учитываются пунктуация, пробелы и диакритика за исключением титл).

Модель научилась хорошо распознавать большинство символов основного текста, но встречаются проблемы с распознаванием мелкого текста (например, в заголовках глав), похожих букв и сочетаний букв (в том числе у при наличии отдельного класса для ѿ). Кроме того, алгоритм формирования текста оказывает неустойчивым к наклону страницы и требует усовершенствования.

На данный момент нельзя утверждать о возможности использования разработанного инструмента для массовой оцифровки славянских рукописей, но полученный прототип выглядит обещающим с перспективой дальнейшего развития.

Ключевые слова: распознавание рукописного текста, алгоритм формирования текста, славянские рукописи, искусственные нейронные сети, YOLO.

Введение

В библиотеках и хранилищах России и других стран содержится большое количество старославянских рукописей, часть из которых до сих пор остается неоцифрованной и неизученной. Ручная оцифровка требует больших вложений времени и усилий, в то же время развитие современных технологий дает надежду на то, что сложные некогда задачи – в том числе в сфере гуманитарных наук, к которым относится славистика, – можно будет решать автоматическими методами.

Целью данного исследования является определение методики распознавания текста церковнославянской рукописи XI века на растровых изображениях с помощью искусственной нейронной сети, извлечения текстовой информации и ее перевода в машиночитаемый формат.

Материалом исследования стал текст Остромирова Евангелия (1056–1057 гг., Российская национальная библиотека (РНБ), Ф.п.1.5). В качестве исходных данных были взяты цифровые изображения листов рукописи, представленные на сайте РНБ, а также цифровая транскрипция портала «Манускрипт»¹.

Эксперименты по созданию модели распознавания текста на этом же материале проводил студент ИжГТУ М. Музыкантов на платформе Transkribus² и смог достичь доли ошибок (CER) на уровне 4,19 %. Платформа предоставляет методы разметки обучающих данных в полуавтоматическом режиме и мощности для обучения нейронных сетей. Однако Transkribus является зарубежным коммерческим сервисом с ограничением на число обрабатываемых документов в зависимости от тарифа.

©Терещенко В. И., Комышев М. А., Вологдин С. В., Сяктерева В. В., 2026

¹ Манускрипт: славянское письменное наследие. URL: <http://manuscripts.ru> (дата обращения: 06.03.2026).

² Ostromir v1.0 // Transkribus. URL: <https://app.transkribus.org/models/text/ostromir-v10> (дата обращения: 06.03.2026).

Так как имеется потребность в том, чтобы поставить на поток распознавание большого количества старославянских рукописей, дошедших до наших дней, решение на основе Transkribus не подходит для этой задачи, и в данной статье рассматривается возможность создания собственного независимого инструмента.

Были поставлены следующие задачи:

- 1) сбор и подготовка исходных данных,
- 2) определение архитектуры искусственной нейросети,
- 3) подбор гиперпараметров,
- 4) обучение нейросети и корректировка данных при необходимости,
- 5) создание алгоритма формирования текста.

Обзор литературы

Задача распознавания текста определяется такими факторами, как тип письма и язык, количество и качество доступных изображений.

Среди подходов к этой задаче в современной литературе встречается применение сверточных нейронных сетей [1] или производных архитектур на их основе [2].

Имеются примеры использования нейросетей семейства YOLO [3–5], которые в последние годы стали основной архитектурой для различных задач распознавания и обнаружения объектов [6, 7].

В случае сложных документов отдельной подзадачей является распознавание макета страниц: заголовков, основного текста, а при анализе древних рукописей – часто и различных декоративных элементов [8]. Для сегментации страниц могут применяться нейросети архитектуры U-Net или производные от нее [8–10], где в качестве результата генерируются растровые маски.

Для увеличения точности распознавания могут применяться дополнительные этапы. Например, в работе [11] используются сети кодирования-декодирования с добавлением слоя квантования. В работе, посвященной анализу рукописного текста на арабском языке [12], предлагается моделирование языковых зависимостей текста с помощью механизма внимания.

В Сибирском федеральном университете сейчас активно разрабатывается инструмент Manuscript OCR³, направленный на оцифровку русских дореволюционных рукописных текстов.

Он представляет собой полноценное решение, включающее компоненты сегментации, распознавания и коррекции.

Распознавание церковно-славянской рукописи

Остромирово Евангелие является примером устава – типом письма, присущего ранним кириллическим текстам [13, 14]. Его особенностью является четкая и угловатая форма букв и их обособленность друг от друга. Ранние кириллические тексты не обозначали границы слов и не имели развитой системы пунктуации, присущей современным языкам.

Особенности уставного письма дают возможность предполагать, что эффективным может быть распознавание отдельных символов.

Этот подход предлагают авторы [14]. Аналогичный подход успешно использован в работе [3], посвященной распознаванию греческих манускриптов со сходным рисунком.

В качестве архитектуры нейросети была выбрана YOLOv8 (актуальная версия на момент начала работ) для задачи обнаружения объектов. Как было указано выше, YOLO сегодня является одним из ведущих решений в области компьютерного зрения.

В качестве видов объектов, которые необходимо найти (классов), были выделены 42 буквы старославянской кириллицы, используемые в Остромировом Евангелии, а также надстрочный знак титло (простое), обозначающий сокращения.

В качестве отдельного класса выделяется диграф **ук** (ѡ) наряду с **жицей** (ѣ). Не выделялись отдельно широкий вариант буквы **есть**, монографный **ук** (Ѣ) и некоторые другие графемы, появляющиеся в рукописи sporadически, а также буква **коппа** (ѣ), используемая только для записи чисел.

Не все классы встречаются с сопоставимой частотой в рукописи, что негативно влияет на качество распознавания.

Разброс в количестве экземпляров различных букв был сокращен с помощью размещения мозаичным способом на изображениях символов, в том числе искусственно воссозданных вариантов редко встречающихся букв (например, *ферт*). Пример созданного текста представлен на рис. 1.

³ Манускрипт. Распознавание дореформенных рукописей. URL: <https://manuscript.sfu-kras.ru/> (дата обращения: 06.03.2026).



Рис. 1. Пример искусственно созданных букв в выборке

Fig. 1. Example of artificially produced letters inside the sample

Для увеличения количества данных в обучающей и валидационной выборках⁴ были использованы процедуры модификации изображений из программных пакетов PIL и OpenCV для Python. В табл. 1 приведены параметры изображений, подвергавшиеся изменениям, и соответствующие коэффициенты. Полученная таким образом обучающая выборка составила 1680 изображений, валидационная – 474.

Таблица 1. Модификации, применяемые к изображениям

Table 1. Modifications applied to the sample images

Процедура	Коэффициенты		
	Обучающая выборка	Валидационная выборка	
Grayscale	—		
Hue	0,70, 0,75, 0,80,	0,98, 1,02	
Saturation	0,85, 0,90, 1,10,		
Brightness	1,15, 1,20, 1,25,		
Contrast	1,30	0,96, 0,98,	
Sharpness	0,70, 0,75, 0,80,		1,02, 1,04
	0,85, 0,90, 0,95,		
S&P Noise	1,10, 1,15, 1,20,	0,001, 0,003,	
	1,25, 1,30, 1,35,		
	1,40, 1,45, 1,50		
	0,005, 0,007,	0,001, 0,003	
	0,009, 0,011		

Обучение модели YOLOv8 размера X на полученной выборке осуществлялось в течении трёх эпох. Размер подаваемых изображений составлял 1376×1376 , параметр batch равнялся 1.

Алгоритм формирования текста

Так как результатом работы YOLO является неупорядоченное множество распознанных объектов-символов, необходим алгоритм, который сформирует из них осмысленную последовательность – финальный текст. Нами был разработан базовый эвристический алгоритм, основанный на принципах организации текста Остромирова Евангелия.

Каждый обнаруженный YOLO объект имеет ограничивающую рамку в виде прямоугольника на двумерной плоскости (см. рис. 2 на примере буквы *ять*) и класс, обозначающий конкретный распознанный символ. Начало координат находится в левом верхнем углу изображения, ось *y* направлена вниз.

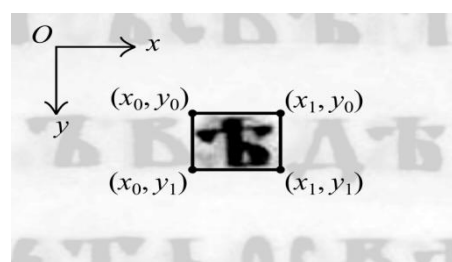


Рис. 2. Представление распознанного символа

Fig. 2. Representation of are cognized character

Помимо указанных координат для символов также высчитывается значение вертикальной координаты центра:

$$y_{mid} = \frac{y_0 + y_1}{2}.$$

В качестве первого шага все распознанные символы сортируются по убыванию y_0 (т. е. снизу вверх). Такой порядок обхода позволяет избежать нетривиальных проверок на границы строк, которые могут возникать из-за большей высоты отдельных литер (например, *рцы*) и наличия надстрочных знаков. Алгоритм проходит последовательно по каждому символу и пытается присоединить его к текущей формируемой строке. Обозначим i – номер первого символа, j – номер последнего символа, добавленного к строке, $j \geq i$.

Решение о присоединении очередного символа делается на основе усреднённых верхней и нижней границ текущей строки, обновляемых по ходу анализа:

$$\bar{y}_0 = \frac{1}{j-i+1} \sum_i^j y_0, \quad \bar{y}_1 = \frac{1}{j-i+1} \sum_i^j y_1.$$

Также вычисляется высота строки $H = \bar{y}_1 - \bar{y}_0$.

⁴ Обучающая выборка (trainingsample) используется для подбора и корректировки внутренних параметров модели во время обучения. Валидационная выборка (validation sample) используется для оценки производительности модели по результатам обучения на каждом этапе.

Если выполняется условие $y_{mid} \geq \bar{y}_0$, то символ уверенно считается частью строки и проверка переходит к следующему символу. В противном случае алгоритм начинает анализ потенциального разрыва строки. В качестве дополнительной проверки вводится предполагаемая верхняя граница строки E , которая была предварительно вычислена при сборке предыдущей строки по формуле

$$E = \bar{y}_0 - cH,$$

где $c = 1,15$ для первой строки и $c = 1,1$ для всех последующих, а также контрольная точка на уровне верхней трети текущей строки:

$$T = \bar{y}_0 + \frac{1}{3}H.$$

Алгоритм проверяет условие $T < E$, и если оно верно, то обнаружен явный разрыв строки. Рассматривается «хвост» из последних отложенных (еще не добавленных к строке) символов, и если среди них встречаются титла, то они прикрепляются к текущей строке. Последующие символы считаются частью уже новой строки, и анализ переходит к ней.

Наконец, внутри каждой найденной строки символы сортируются по возрастанию x_1 (слева направо) и определяется граница между левой и правой колонками. За нее принимается первый разрыв между двумя символами, ширина которого больше ширины предыдущего символа, умноженной на 2,5. По завершении алгоритма сформированный текст выводится пользователю построчно в соответствии со стандартом Юникод, в каждой строке колонки отделены символом табуляции (U+0009).

Метрики

Общепринятой метрикой оценки производительности моделей в контексте задачи обнаружения объектов в целом стал показатель mAP (MeanAverage Precision) [15]. Подробное описание процедуры вычисления приведено в [6]. Инструментарий YOLO автоматически подсчитывает mAP в процессе валидации модели. Также подсчитываются метрики **precision** (точность) и **recall** (полнота), позволяющие судить о количестве ложноположительных и ложноотрицательных обнаружений соответственно.

Однако при рассмотрении задачи распознавания текста часто используется мера CER (CharacterError Rate) [16] – текстовая мера, основанная на расстоянии Левенштейна (суммарном количестве вставленных, удаленных и замененных символов) и рассчитываемая по формуле:

$$CER = \frac{\sum \text{dist}(h_i, g_i)}{\sum \text{len}(g_i)},$$

где g_i – корректная транскрипция i -го текста (groundtruth), h_i – распознанный текст (гипотеза), dist – расстояние Левенштейна между двумя текстами, len – длина текста. CER измеряется в процентах, и более низкие значения указывают на более высокое качество распознавания. А. Рабус [17] приводит 5 % как реалистичное значение CER, достижимое современными методами при распознавании рукописей.

Инструментарий YOLO не включает подсчет CER, поэтому он был реализован нами самостоятельно для оценки производительности финальной модели. Корректные транскрипции, полученные с портала «Манускрипт», перед вычислением расстояния нормализуются следующим образом:

1. Все символы приводятся к нижнему регистру⁵.
2. Различные визуальные варианты простого титла приводятся к единой кодовой позиции U+0483.
3. Пробелы, пунктуация и диакритика (кроме простых и буквенных титл) очищаются.
4. Буквенные титло заменяются на соответствующие им строчные буквы (например, гѣа рассматривается как гѣа).
5. Следующие в одной строке друг за другом символы о и у заменяются на диграф оу (например, в слове моуѣси).

В качестве гипотезы передается распознанный текст, полученный с помощью описанного ранее алгоритма формирования текста.

Рабус [17] отмечает, что для уверенной оценки качества распознавания достаточно набора 20–30 тестовых страниц. Нами была вычислена метрика CER для первых 200 страниц Остромирова Евангелия за исключением пустых страниц – итого 195 изображений.

Отметим, что из-за особенностей изначального построения обучающего набора данных нельзя сказать, какие страницы рукописи были положены в его основу, и, вероятно, часть экземпляров букв на этих 195 изображениях модель уже видела при обучении. Тем не менее, так как обучение производилось на уровне отдельных символов, а CER оценивает производительность модели на уровне текста, такая проверка позволит сделать некоторые выводы.

⁵ Транскрипция текста на портале «Манускрипт» использует заглавные кодовые позиции для обозначения инициалов и других визуально больших букв

Обсуждение результатов

В табл. 2 приведены значения основных метрик во время обучения модели. Во вторую эпоху было достигнуто максимальное значение **precision**, поэтому она была автоматически выбрана как наиболее лучшая.

В табл. 3 приведены значения финальных метрик отдельно по каждому классу.

Таблица 2. Метрики обучения модели

Table 2. Model training metrics

Эпоха	Precision	Recall	mAP 50	mAP 50–95
1	0,998	0,585	0,792	0,692
2	1,000	0,590	0,795	0,705
3	0,999	0,595	0,797	0,704

Таблица 3. Метрики точности распознавания по классам

Table 3. Recognition accuracy metrics per class

Символ	Precision	Recall	mAP 50	mAP 50–95
все	1,000	0,590	0,795	0,705
титло	1,000	0,038	0,519	0,382
а	1,000	0,259	0,629	0,536
б	1,000	0,948	0,974	0,859
в	1,000	0,807	0,903	0,800
г	1,000	0,510	0,755	0,661
д	1,000	0,850	0,925	0,836
е	1,000	0,045	0,522	0,389
ж	1,000	0,282	0,641	0,585
з	1,000	0,747	0,874	0,810
з	1,000	0,638	0,819	0,720
и	1,000	0,725	0,862	0,735
і	1,000	0,137	0,569	0,478
к	1,000	0,932	0,966	0,853
л	1,000	0,228	0,614	0,539
м	1,000	0,467	0,734	0,663
н	1,000	0,862	0,931	0,806
о	1,000	0,048	0,524	0,424
п	1,000	0,842	0,921	0,816
р	0,999	0,285	0,642	0,561
с	1,000	0,151	0,575	0,465
т	1,000	0,094	0,547	0,497

Значения метрик mAP позволяют утверждать о приемлемом уровне распознавания символов Остромирова Евангелия. Видно, что доля ложноположительно распознанных объектов крайне мала.

Вместе с тем полнота распознавания (**recall**, доля правильно распознанных объектов от всего числа объектов данного класса) для некоторых символов невысока, особенно это характерно

Символ	Precision	Recall	mAP 50	mAP 50–95
γ	1,000	0,297	0,648	0,603
ϕ	1,000	0,890	0,945	0,888
ϕ	0,999	0,161	0,580	0,556
χ	1,000	0,955	0,977	0,885
w	1,000	0,559	0,779	0,692
ϖ	1,000	0,554	0,777	0,705
ц	1,000	0,771	0,885	0,810
ϣ	1,000	0,936	0,968	0,853
ш	1,000	0,919	0,959	0,866
щ	1,000	0,223	0,612	0,571
ъ	1,000	0,484	0,742	0,646
ы	1,000	0,829	0,914	0,809
ь	1,000	0,559	0,780	0,675
ѣ	1,000	0,790	0,895	0,813
ю	1,000	0,933	0,966	0,855
ѡ	1,000	0,950	0,975	0,876
ѣ	1,000	0,829	0,915	0,808
ѧ	1,000	0,479	0,740	0,667
Ѩ	1,000	0,906	0,953	0,848
ѩ	1,000	0,806	0,903	0,814
Ѫ	1,000	0,949	0,975	0,888
ѫ	0,988	0,707	0,847	0,765

для букв, похожих на другие, например **есть, он и слово**.

Совокупное значение CER на первых 200 страницах Остромирова Евангелия составило 4,09 %. Распределение CER на отдельных страницах представлено на рис. 3 в виде «ящика с усами». Как видно на графике, доля ошибок для большинства страниц не превышает 8,44%. Медианное значение CER составило 3,33 %.

Это говорит о сравнительно хорошем качестве распознавания, хотя нельзя не отметить, что существуют выбросы в виде нескольких страниц с высокой долей ошибок (отмечены крестиками на графике).

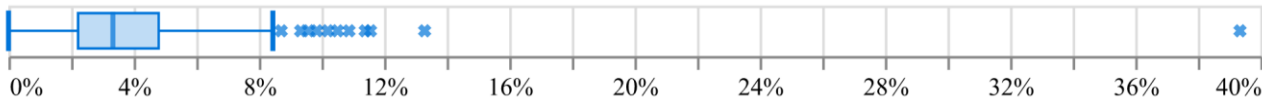


Рис. 3. Распределение постраничного CER

Fig. 3. Distribution of the page-based CER

На рис. 4 показан пример распознанного текста на странице 54 (лицевая сторона листа 28). CER для данной страницы составил 4,37 % и сопоставим со средним значением.

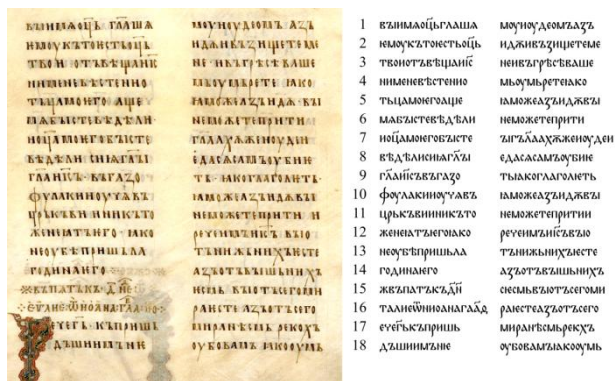


Рис. 4. Текст, распознанный на странице 54 Остромирова Евангелия

Fig. 4. Text recognized on the page 54 of the Ostromir Gospel

Подавляющая часть текста распознана корректно. Однако можно заметить, что вместо *у* в 10-й строке левой колонки в слове *газофулаки* модель распознала *оу*; аналогично происходит и в других случаях, где *ижица* (*у*) используется отдельно. Об этом же свидетельствует табл. 3, где метрика **recall** для *ижицы* достаточно мала. Можно сделать вывод, что выделение диграфа *оу* в отдельный класс не оптимально, так как он подавляет менее представленный класс *у*.

В строке 12 правой колонки сочетание букв *ы* и *о* было распознано как *ь* и *ю*. Также модель иногда распознает пунктуацию как буквы (например, в строке 15 левой колонки) или видит символы, которых на странице в этом месте нет (строки 7 и 15 правой колонки). В строке 17 правой колонки была потеряна буква *он* в слове *рекохъ*.

Также следует учитывать, что метрика CER для упомянутой ранее модели на платформе Transkribus (4,19 %) включает диакритические знаки и пробелы, которые в наших расчетах не учитываются.

Максимальный показатель CER (39,34 %) наблюдается на странице 127 (обратная сторона листа 64). Часть текста представлена на рис. 5.

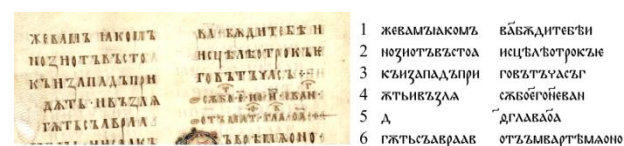


Рис. 5. Фрагмент текста, распознанного на странице 127

Fig. 5. Fragment of the text recognized on the page 127

Наиболее заметно то, что в строках 4–6 обеих колонок алгоритм неправильно определяет границы строк, разделяя и смешивая символы. Вероятно, это связано с наклоном страницы, т. к. при ручном повороте изображения на 3° данная проблема исчезает. Однако похожая проблема с переносом отдельных символов (особенно титл) на соседние строки встречается и на других изображениях. Кроме того, модель может ошибочно распознавать символы, просвечивающие из-под листа (см. строки 2 и 6 левой колонки на рис. 5).

Хуже всего распознается мелкий текст, в частности золотые заголовки глав (пример представлен на рис. 6 со страницы 15). В них модель начинает распознавать несколько букв в одном месте и становится более склонной путать буквы между собой.

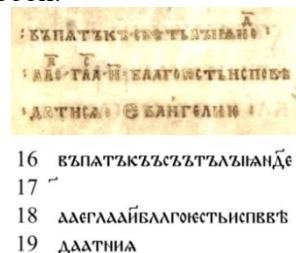


Рис. 6. Распознавание заголовков глав

Fig. 6. Recognition of chapter titles

В некоторых случаях высокий показатель CER был вызван ошибками в ручной транскрипции текста. В данной статье приведены результаты после исправления таких недочетов, которые удалось обнаружить по результатам валидации модели.

Выводы

В данной статье была рассмотрена разработка инструмента для распознавания церковнославянских рукописей с помощью нейросети архитектуры YOLO. Предложенная методика включает обнаружение отдельных символов на странице и формирование из них текста с помощью специального алгоритма.

Разработанный прототип позволяет хорошо распознавать символы основного текста рукописи с метрикой CER на уровне 4,09 %. При этом в случае отдельных символов модель пока не всегда дает качественные результаты распознавания, что на данный момент не позволяет уверенно использовать разработанный инструмент для массовой оцифровки. В целом же предложенный подход имеет большие перспективы, и авторы продолжают работы по повышению точности распознавания рукописного текста.

Одной из приоритетных задач исследования является оцифровка, распознавание и перепроверка большого набора манускриптов различных авторов и эпох, поэтому логичным продолжением работы будет включение в набор данных других рукописей со сходными почерками для создания так называемой генеральной модели.

Библиографические ссылки

1. Worddeepnet: handwritten gurmukhi word recognition using convolutional neural network / H. Kaur [et al.] // *Multimedia Tools and Applications*. 2023. Vol. 82. DOI: 10.1007/s11042-023-15527-2.

2. A Novel Approach for Vietnamese Handwritten Text Recognition / V. H. Duong [et al.] // *Automatic Control and Computer Sciences*. 2023. Vol. 57, no. 5. P. 534–541. DOI: 10.3103/S014641162305005X.

3. Turnbull R., Mannix E. Detecting and recognizing characters in Greek papyri with YOLOv8, DeiT and SimCLR // *arXiv*. 2024. DOI: 10.48550/arXiv.2401.12513.

4. Handwritten English word recognition using a deep learning based object detection architecture / R. Mondal [et al.] // *Multimedia Tools and Applications*. 2022. Vol. 81, no. 1. P. 975–1000. DOI: 10.1007/s11042-021-11425-7.

5. Bangla handwritten word recognition using YOLO V5 / M. A. Hossain [et al.] // *Bulletin of Electrical Engineering and Informatics*. 2024. Vol. 13, no. 3. P. 2175–2190. DOI: 10.11591/eei.v13i3.6953.

6. Terven J., Córdova-Esparza D.-M., Romero-González J.-A. A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS // *Machine Learning and Knowledge Extraction*. 2023. Vol. 5, no. 4. P. 1680–1716. DOI: 10.3390/make5040083.

7. Филичкин С. А., Вологдин С. В. Применение нейронной сети YOLOv5 для распознавания наличия средств индивидуальной защиты // *Интеллектуальные системы в производстве*. 2022. Т. 20, № 2. С. 61–67. DOI: 10.22213/2410-9304-2022-2-61-67. EDN: FIXGZD.

8. Rahal N., Vöglin L., Ingold R. Approximate ground truth generation for semantic labeling of historical documents with minimal human effort // *International Journal on Document Analysis and Recognition*. 2024. Vol. 27, no. 3. P. 335–347. DOI: 10.1007/s10032-024-00475-w.

9. Николаев П. Л. Сегментация неструктурированного текста на изображениях книжных обложек с помощью сверточной сети, основанной на архитектуре U-Net // *International Journal of Open Information Technologies*. 2023. Т. 11, № 10. С. 39–43. EDN: JSRFLB.

10. Лобанова В. А., Иванова Ю. А. Разработка нейросетевого алгоритма распознавания надписей на изображениях реальных сцен // *Компьютерная оптика*. 2022. Т. 46, № 5. С. 790–800. DOI: 10.18287/2412-6179-CO-1047. EDN: AKMCMG.

11. Davoudi H., Traviglia A. Discrete representation learning for handwritten text recognition // *Neural Computing and Applications*. 2023. Vol. 35, no. 21. P. 15759–15773. DOI: 10.1007/s00521-023-08445-9.

12. Momeni S., BabaAli B. A transformer-based approach for Arabic offline handwritten text recognition // *Signal, Image and Video Processing*. 2024. Vol. 18, no. 4. P. 3053–3062. DOI: 10.1007/s11760-023-02970-9.

13. Гранстрем Е. Э. Описание русских и славянских пергаменных рукописей: Рукописи русские, болгарские, молдавлахийские, сербские / сост. канд. пед. наук Е. Э. Гранстрем; под ред. д-ра филол. наук Д. С. Лихачева. Л.: ГПБ им. М. Е. Салтыкова-Щедрина, 1953. 148 с.

14. Варламов А. С., Макарова О. Л. Проблемы распознавания старославянских манускриптов // *Информационные технологии в науке, промышленности и образовании: сборник трудов Всероссийской научно-технической конференции, Ижевск, 23–24 мая 2024 года*. 2024. С. 57–61. EDN: AKMCMG.

15. Diwan T., Anirudh G., Tembhurne J. V. Object detection using YOLO: challenges, architectural successors, datasets and applications // *Multimedia Tools and Applications*. 2023. Vol. 82, no. 6. P. 9243–9275. DOI: 10.1007/s11042-022-13644-y.

16. StackMix and Blot Augmentations for Handwritten Text Recognition / A. Shonenkov [et al.] // *arXiv*. 2021. DOI: 10.48550/arXiv.2108.11667.

17. Rabus A., Meindl M. The Digital Revolution in Slavic Manuscript Studies: HTR Technology and its Impact on Philological Research // *Studi Slavistici*. 2025.

Vol. XXII, Special Issue. P. 35–54. DOI: 10.36253/Studi_Slavis-17690

References

1. Kaur H., Bansal S., Kumar M., Mittal A., Kumar K. Worddeepnet: handwritten gurumukhi word recognition using convolutional neural network. *Multimedia Tools and Applications*, 2023, vol. 82, pp. 46763–46788. DOI 10.1007/s11042-023-15527-2.
2. Duong V. H., Nguyen H. T., Nakagawa M., Pham T. B. A Novel Approach for Vietnamese Handwritten Text Recognition. *Automatic Control and Computer Sciences*, 2023, vol. 57, no. 5, pp. 534–541. DOI 10.3103/S014641162305005X.
3. Turnbull R., Mannix E. Detecting and recognizing characters in Greek papyri with YOLOv8, DeiT and SimCLR. *arXiv*, 2024. DOI 10.48550/arXiv.2401.12513.
4. Mondal R., Malakar S., Barney Smith E. H., Sarkar R. Handwritten English word recognition using a deep learning based object detection architecture. *Multimedia Tools and Applications*, 2022, vol. 81, no. 1, pp. 975–1000. DOI 10.1007/s11042-021-11425-7.
5. Hossain M. A., Abadin A. Z., Faruk M. O., Ara I., Rashidul Hasan M. A., Fatta N., Asraful M., Hossen E. Bangla handwritten word recognition using YOLO V5. *Bulletin of Electrical Engineering and Informatics*, 2024, vol. 13, no. 3, pp. 2175–2190. DOI 10.11591/eei.v13i3.6953.
6. Terven J., Córdova-Esparza D.-M., Romero-González J.-A. A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS. *Machine Learning and Knowledge Extraction*, 2023, vol. 5, no. 4, pp. 1680–1716. DOI 10.3390/make5040083.
7. Filichkin S. A., Vologdin S. V. [Using the YOLOv5 Neural Network to Recognize the Presence of Personal Protective Equipment]. *Intellektual'nyesistemy v proizvodstve*, 2022, vol. 20, no. 2, pp. 61–67. DOI 10.22213/2410-9304-2022-2-61-67.
8. Rahal N., Vögtlin L., Ingold R. Approximate ground truth generation for semantic labeling of historical documents with minimal human effort. *International Journal on Document Analysis and Recognition*, 2024, vol. 27, no. 3, pp. 335–347. DOI 10.1007/s10032-024-00475-w.
9. Nikolaev P. L. [Segmentation of unstructured text on the book cover images using the convolutional network based on the U-Net architecture]. *International Journal of Open Information Technologies*, 2023, vol. 11, no. 10, pp. 39–43 (in Russ.).
10. Lobanova V. A., Ivanova Yu. A. [Development of software for the segmentation of text areas in real-scene images]. *Computer Optics*, 2022, vol. 46, no. 5, pp. 790–800 (in Russ.). DOI 10.18287/2412-6179-CO-1047.
11. Davoudi H., Traviglia A. Discrete representation learning for handwritten text recognition. *Neural Computing and Applications*, 2023, vol. 35, no. 21, pp. 15759–15773. DOI 10.1007/s00521-023-08445-9.
12. Momeni S., BabaAli B. A transformer-based approach for Arabic offline handwritten text recognition. *Signal, Image and Video Processing*, 2024, vol. 18, no. 4, pp. 3053–3062. DOI 10.1007/s11760-023-02970-9.
13. Granstrem E. E. *Opisanie russkikh i slavyanskikh pergamennykh rukopisei: Rukopisi russkie, bolgarskie, moldovlakhitskie, serbskie [Description of Russian and Slavic parchment manuscripts: Russian, Bulgarian, Moldovan, Serbian manuscripts]*. Leningrad, M. E. Saltykov-Shchedrin Public Library, 1953, 148 p. (in Russ.).
14. Varlamov A. S., Makarova O. L. Problemy raspoznavaniya staroslavjanskikh manuskriptov [Problems of recognizing of Old Church Slavonic manuscripts]. *Informatsionnye tekhnologii v nauke, promyshlennosti i obrazovanii : Sbornik trudov Vserossiiskoi nauchno-tekhnicheskoi konferentsii*. [Proc. Information technologies in science, industry and education: proceedings of the All-Russian Scientific and Technical Conference] (ed. Petukhov K. Yu.). Izhevsk, 2024, pp. 57–61 (in Russ.).
15. Diwan T., Anirudh G., Tembhurne J. V. Object detection using YOLO: challenges, architectural successors, datasets and applications. *Multimedia Tools and Applications*, 2023, vol. 82, no. 6, pp. 9243–9275. DOI 10.1007/s11042-022-13644-y.
16. Shonenkov A., Karachev D., Novopoltsev M., Potanin M., Dimitrov D. StackMix and Blot Augmentations for Handwritten Text Recognition. *arXiv*, 2021. DOI 10.48550/arXiv.2108.11667.
17. Rabus A., Meindl M. The Digital Revolution in Slavic Manuscript Studies: HTR Technology and its Impact on Philological Research. *Studi Slavistici*, 2025, vol. XXII, Special Issue, pp. 35–54. DOI 10.36253/Studi_Slavis-17690.

* * *

Church Slavonic Handwritten Text Recognition on the Example of the Ostromir Gospel (NRL, F.n.I.5)

V. I. Tereshchenko, Software Engineer, Kalashnikov Izhevsk State Technical University, Izhevsk, Russia

M. A. Komyshev, Student, Kalashnikov Izhevsk State Technical University, Izhevsk, Russia

S. V. Vologdin, DSc in Engineering, Associate Professor, Kalashnikov Izhevsk State Technical University, Izhevsk, Russia

V. V. Syaktereva, PhD in Engineering, Associate Professor, Kalashnikov Izhevsk State Technical University, Izhevsk, Russia

This paper discusses the development of a tool for Church Slavonic manuscript recognition based on the 11th-century Ostromir Gospel (National Library of Russia, F.n.I.5). The YOLOv8 neural network architecture was chosen for the task. The target objects for recognition are 42 Cyrillic letters and the simple titlo.

We developed a heuristic algorithm that reconstructs text from an unordered set of objects detected by YOLO. The algorithm accumulates the text line-by-line from bottom to top based on the bounding box coordinates and outputs the result according to the Unicode standard.

The final model was trained for 2 epochs on 1680 images containing mosaic compositions of characters taken from different pages of the manuscript. To reduce the imbalance in character instance counts, we inserted artificially created samples of rare letters such as fert. A Character Error Rate of 4.09% was achieved on the first 200 pages of the Ostromir Gospel (punctuation, spaces, and diacritics except for titlo were not included in the evaluation).

The model learned to recognize most characters in the main text well, yet it struggles with small writing (e.g. in chapter titles) and similar letters and their combinations (including y when having a separate class for oy). Moreover, the text reconstruction algorithm proves unstable when the page is tilted and thus requires improvement.

At this stage, we cannot claim that the tool is ready for a mass digitization of Slavic manuscripts, but the prototype shows promise for further development.

Keywords: handwriting recognition, text generation algorithm, Slavic manuscripts, artificial neural networks, YOLO.

Получено: 23.03.26

Образец цитирования

Распознавание рукописного церковнославянского текста на примере Остромирова евангелия (РНБ, Ф.п.1.5) / В. И. Терещенко, М. А. Комышев, С. В. Волгодин, В. В. Сяктерева // Интеллектуальные системы в производстве. 2026. Т. 24, № 2. С. 71–79. DOI: 10.22213/2410-9304-2026-2-71–79.

For Citation

Filipovich O.V., Balakina N.A., Balakin A.I. [Simulation Of Two-Parametric Selective Assembly Of Two Parts]. *Intellektual'nyesistemy v proizvodstve*. 2026, vol. 24, no. 2, pp. 71-79 (in Russ.). DOI: 10.22213/2410-9304-2026-2-71–79.