УДК 004.932.75’1

# РАСПОЗНАВАНИЕ ТЕКСТА В ЗАШУМЛЕННЫХ ИЗОБРАЖЕНИЯХ СКАНИРОВАННЫХ ДОКУМЕНТОВ

**© 2016.** ***Т.В. Шарий, Р.О. Лялин, А.Е. Гукай, В.Н. Котенко***

В настоящее время по-прежнему актуальна проблема перевода сканированных документов относительно низкого качества в текстовый формат. В статье предлагается робастный метод распознавания текстовых символов в зашумленных изображениях. Описаны алгоритмы, применяемые для очистки изображения от шумов, выделения контуров текстовых областей, сегментации текста по буквам. Приведены результаты экспериментов по распознаванию текста на основе статистической модели машин опорных векторов.

***Ключевые слова:*** OCR, машинное обучение, медианный фильтр, бинаризация, сегментация.

**Введение.** Оптическое распознавание символов (Optical Character Recognition, OCR) [1–4] остается на протяжении многих лет актуальной задачей с точки зрения как бизнеса, так и развития научно-технического направления компьютерного зрения. Суть OCR-процесса состоит, прежде всего, в переводе цифровых изображений в другие цифровые форматы, более подходящие для редактирования, поиска, реферирования информации, а именно: непосредственно текст и метаданные о документе (количество слов, абзацев, таблиц, подписей и т.д.). С каждым годом число изображений на локальных компьютерах и в сети интернет стремительно растет, и все больше документов нуждается в оцифровке и постобработке. В данном контексте OCR является элементом передового направления разработок в современной IT-сфере – «больших данных» (Big Data and Data Science), в рамках которого большие объемы данных должны эффективно храниться и подготавливаться для оперативного анализа. Также к вариантам применения OCR можно отнести: автоматический ввод данных из бланков разного рода в компьютер, автоматическое распознавание автомобильных номеров, программы-помощники для лиц с нарушениями зрения и др.

В настоящее время существуют программные решения для распознавания текста в изображениях и преобразования форматов файлов, среди которых следует отметить коммерческие приложения [5]: OmniPage, Adobe Acrobat, ABBYY FineReader, ReadIris, PowerPDF, SodaPDF. Точность распознавания текста в лучших из них достигает 97–99 % для качественных сканов документов. Тем не менее, ряд проблем значительно ухудшает качество OCR даже самых эффективных программ. Эти проблемы вызваны искажением изображений фоновым шумом (пятна от чашек с кофе, завернутые уголки, следы от скрепок и т.д.), возможным поворотом текста, вариабельностью цветов и начертаний шрифтов, языков. Таким образом, предварительная обработка зашумленного сканированного изображения выходит на первый план в современных информационных технологиях хранения и обработки цифровых документов.

**Постановка задачи.** Целью данной работы является разработка и исследование робастного метода распознавания текста в изображениях отсканированных документов, содержащих шумы. Для визуализации и автоматизации проведения исследований необходимо создать соответствующий программный инструментарий. В статье акцент делается на предварительной цифровой обработке изображения. Этап распознавания также рассмотрен, однако подробный анализ и разработка статистических моделей машинного обучения, применяемых на этом этапе, является предметом дальнейших исследований.

**Цифровая обработка зашумленного сканированного изображения.** В статье за основу взята типовая схема процесса OCR, включающая блок предварительной обработки скана документа и блок распознавания символов. В первом блоке решаются следующие задачи (рис.1):

1) фильтрация шумов и удаление фона;
2) бинаризация изображения;
3) идентификация текстовой области документа;
4) поворот текстовой области при необходимости (выравнивание документа);
5) сегментация (горизонтальная сегментация по строкам текста и вертикальная сегментация по символам).


Рис. 1. Схема OCR

Вся описываемая далее цифровая обработка производится над изображением документа в шкале серого цвета с нормированными значениями пикселей. На первом этапе в работе производится медианная фильтрация. Медианный фильтр является распространенным простым средством удаления импульсных шумов в сигнале. Данный фильтр представляет собой нелинейный низкочастотный фильтр, заменяющий значение каждого пикселя изображения медианным значением пикселей из его окрестности, подавляя, таким образом, выбросы:

$$y[m,n] = \text{median}\{x[i,j], (i,j) \in \omega_N\},$$

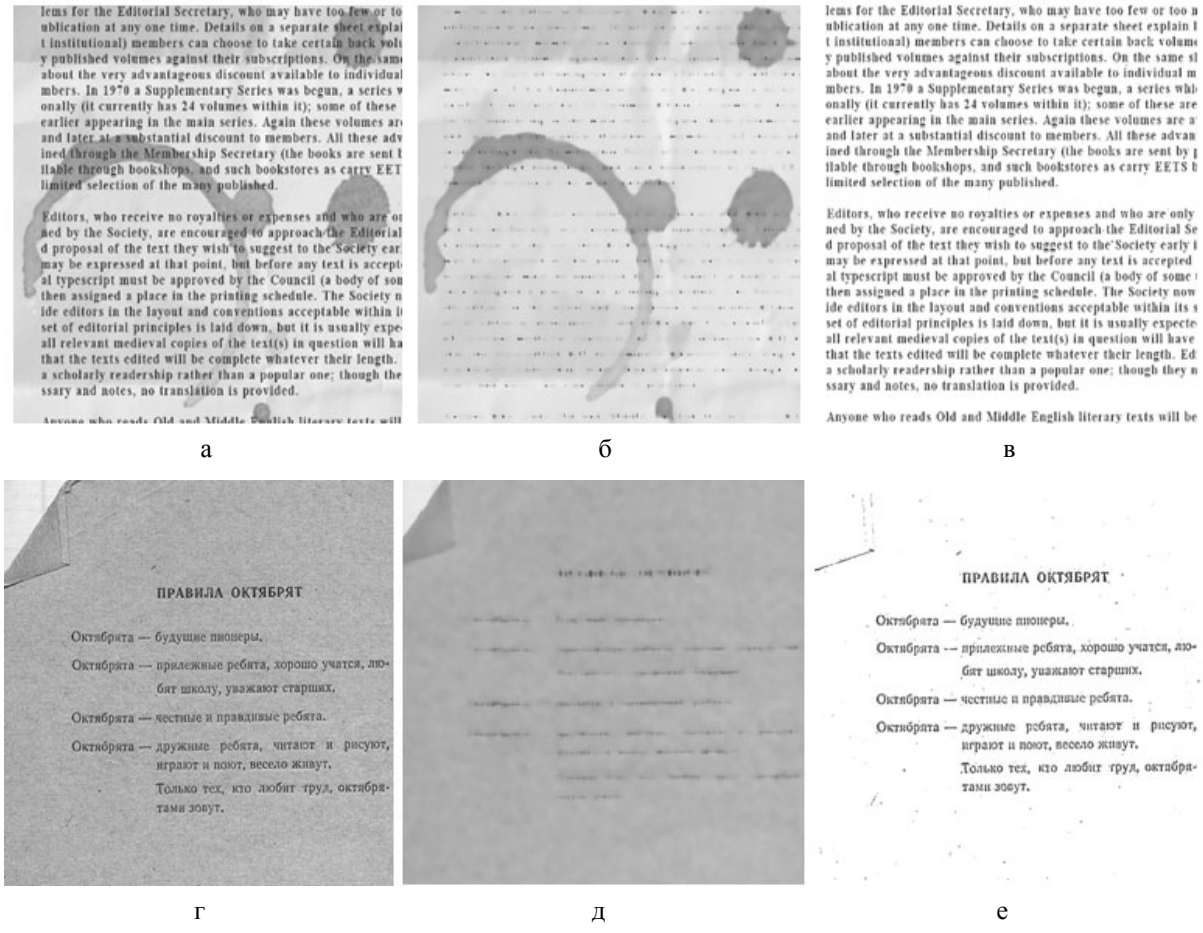
где $x$ – исходное изображение; $y$ – отфильтрованное изображение; $m$, $n$ – координаты текущего центрального пиксела; $\omega_N$ – окрестность пиксела размера $N$.

Результатом медианной фильтрации является изображение, которое можно считать фоном, т.к. данная операция сохраняет медленно меняющиеся признаки и удаляет высокочастотные компоненты изображения, представленные самим текстом. Таким образом, для получения текста можно вычесть изображение фона из исходного изображения скана или оставить только те пиксели, цвет которых темнее фона. При этом, в работе, по аналогии с предложенным в [6] алгоритмом, используется также некоторый порог для уменьшения влияния шумов и отбрасывания незначащих для текста пикселей:

$$t[m,n] = \begin{cases} x[m,n], & \textit{если } x[m,n] < y[m,n] - \delta_S \\ 1.0, & \textit{иначе} \end{cases},$$

где $t$ – результирующее изображение, содержащее текст; $m$, $n$ – координаты текущего пикселя; $\delta_S$ – порог вычитания. Экспериментально подобраны значения параметров $N = 7$ и $\delta_S = 0.1$, при которых текст на тестовых сканах документов субъективно воспринимается наиболее отчетливо.

Результаты для двух изображений с текстом приведены на рис. 2. На рис. 2, а и 2, г показаны сканы документов, на рис. 2, б и 2, д – фон как результат работы медианного фильтра, на рис. 2, в и 2, е – разность изображения и фона, являющаяся искомым текстом.




а б в

г д е

Рис. 2. Примеры результатов медианной фильтрации сканированных изображений:
а) тестовое изображение №1; б) выделенный фон в изображении №1; в) выделенный текст в изображении №1; г) тестовое изображение №2; д) выделенный фон в изображении №2; е) выделенный текст в изображении №2

Следующим шагом при цифровой обработке изображения документа является его бинаризация. В работе используется следующий алгоритм бинаризации:

1. Изображение фильтруется с помощью гауссовского фильтра с ядром размером $L = 11$ пикселей и стандартным отклонением $\sigma = 2$:

$$G[m,n] = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{m^2+n^2}{2\sigma^2}\right).$$

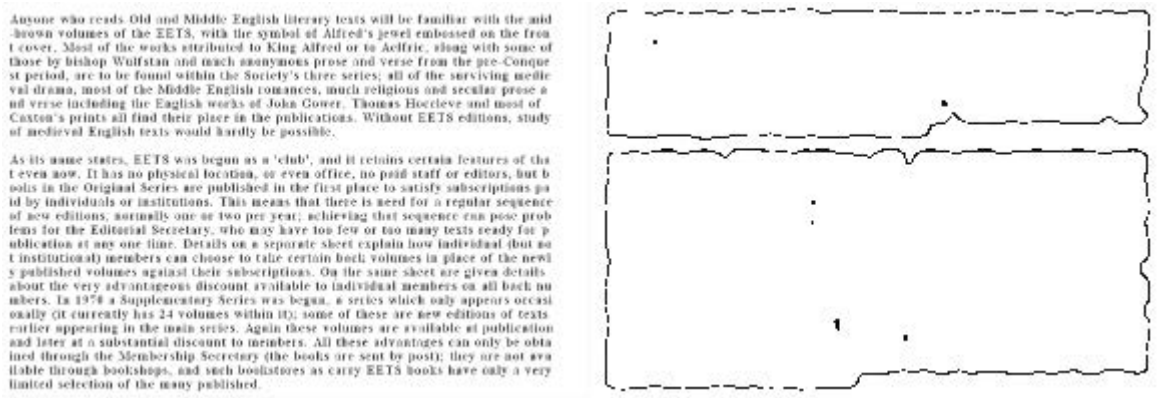
2. В отфильтрованном изображении рассчитывается порог для бинаризации на основе алгоритма Оцу [7]. Алгоритм позволяет вычислить на основе гистограммы

изображения оптимальный глобальный порог для разделения всех пикселей на бинарные группы.

3. По результатам сравнения с порогом, вычисленным на шаге 2, пикселям изображения присваивается значение 0 или 1 (черный или белый цвет).

Приведенный алгоритм позволяет улучшить качество бинаризации по сравнению со стандартным глобальным алгоритмом Оцу за счет удаления лишних теней.

Далее в бинарном изображении осуществляется идентификация области, содержащей текст. В большинстве случаев этой областью является абзац. Для этих целей применяется алгоритм выделения контуров, предложенный в [8]. Алгоритм относится к классу алгоритмов следования границам (Border following), позволяет отслеживать как внешние, так и внутренние границы, и может быть использован для широкого круга задач топологического анализа изображений. Об эффективности алгоритма говорит, в частности, тот факт, что он реализован в функции findContours популярной библиотеки компьютерного зрения OpenCV [9]. Пример результата выделения границ текстовых областей приведен на рис. 3.



а б

Рис. 3. Пример результатов выделения текстовых областей:
а) исходное изображение; б) выделенные контуры текстовых областей

Важной частью предлагаемого метода распознавания текста в зашумленных сканах документов является обнаружение наклона и поворот выделенной текстовой области. Для этого в работе на основе координат левого верхнего, правого верхнего, левого нижнего и правого нижнего пикселей контура текстовой области определяется угол наклона и производится поворот области на найденный угол (преобразование на основе стандартной матрицы поворота).

Завершающим шагом цифровой обработки изображения сканированного документа перед распознаванием символов является сегментация. В качестве единицы распознавания выбрана буква как алфавитная единица любого естественного языка, а также двухбуквенные сочетания, которые в данной статье не рассматриваются и являются предметом дальнейших исследований. В связи с этим, необходимо разделять абзацы сначала на строки (горизонтальная сегментация), а затем на буквы (вертикальная сегментация). При горизонтальной сегментации в каждой строке пикселей изображения вычисляется среднее значение пикселей, и если оно превышает пороговое значение, то полагается, что текущая линия принадлежит пустой области, разделяющей строки текста. Термин «строка пикселей» необходимо отличать от термина «строка текста». Первый относится к изображению, второй – к распознаваемому тексту на изображении. На рис. 4 приведен при-

мер результата алгоритма построчной сегментации (слева расположены графики средних значений пикселей по строкам пикселей; белый цвет имеет максимальное значение).

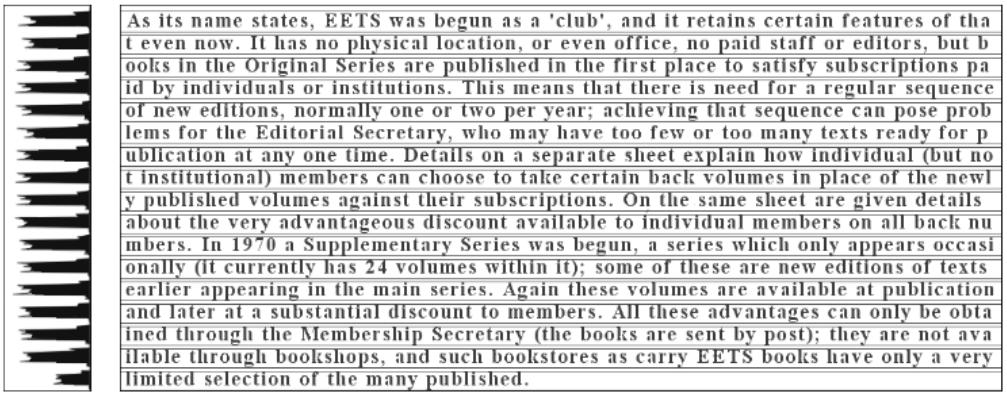



Рис. 4. Пример результата алгоритма построчной сегментации

При вертикальной (побуквенной) сегментации вычисляется среднее значение пикселей столбцов в каждом столбце участка строки текста, выделенной ранее. На рис. 5 приведен пример результата такой побуквенной сегментации (вверху указан график средних значений пикселей по столбцам пикселей).

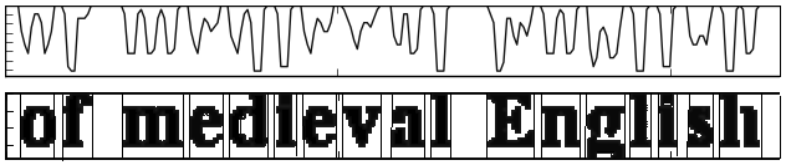



Рис. 5. Пример результата алгоритма побуквенной сегментации

В данном случае также осуществляется сравнение с порогом, которого в ряде случаев оказывается достаточно. Однако при артефактах изображения, не удаленных на предыдущих этапах предобработки, алгоритм может давать ложные срабатывания. В связи с этим, в работе вводится еще адаптивный порог разности между соседними темными пикселами в столбцах, благодаря которому отсеиваются случайные темные пикселы в выделенной строке, а сохраняются только относительно плотно прилегающие друг к другу символы.

**Распознавание текстовых символов.** На этапе распознавания текстовых символов традиционно применяются статистические модели и методы машинного обучения. Данные модели сначала обучаются на большом количестве изображений образцов символов, представляющих обучающую выборку, после чего могут быть использованы для решения задач классификации и регрессии, в частности, для распознавания текста. Также в последнее время наблюдается тенденция использования моделей машинного обучения и на этапе цифровой обработки сигнала. На текущем этапе работы детальный анализ моделей машинного обучения не производится. Тем не менее, проведены предварительные эксперименты по распознаванию текста на основе машин опорных векторов (Support Vector Machine, SVM) [10, 11].

Алгоритм обучения SVM находит среди элементов обучающей выборки векторы, лежащие на границе двух разделяемых подмножеств и строит между этими векторами гиперплоскость, максимально разделяющую входные образы в пространстве признаков. В терминах SVM это опорные векторы.

Решающая функция классификатора SVM задана формулой:

$$h(x) = \mathrm{sgn}(\sum_{i=1}^{m} \lambda_i y_i K(\mathbf{x_i}, \mathbf{x}) - \omega_0),$$

где $\lambda_i$ и $\omega_0$ – коэффициенты; $\mathbf{x_i}$ и $y_i$ – входной вектор и соответствующее ему значение (0 или 1) из обучающей выборки, соответственно; $K(\mathbf{x_i}, \mathbf{x}) = \varphi(\mathbf{x_i}, \mathbf{x})^T \varphi(\mathbf{x_i}, \mathbf{x})$ – ядро. Функция ядра служит для того, чтобы отображать входной вектор в пространство более высокой размерности, в котором, согласно теореме Ковера [11], вероятность разделимости образов повышается. В работе в качестве ядра применяются так называемые радиально-базисные функции:

$$\varphi(\mathbf{x_i}, \mathbf{x}) = \exp(-\gamma \left\| \mathbf{x} - \mathbf{x_i} \right\|^2), \quad \gamma > 0 \tag{1}$$

Обучение SVM заключается в нахождении коэффициентов $\lambda_i$ и $\omega_0$. Для этого решается задача квадратичной оптимизации с ограничениями:

$$\sum_{i=1}^{m} \lambda_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \lambda_i \lambda_j y_i y_j K(\mathbf{x_i}, \mathbf{x_j}) \to \max_{\lambda},$$

$$\begin{cases} \sum_{i=1}^{m} \lambda_i y_i = 0, \\ 0 \le \lambda_i \le C; \quad C > 0, \quad i = 1, 2, ..., m. \end{cases} \tag{2}$$

Константы $\gamma$ и $C$ в формулах (1) и (2), соответственно, являются свободными параметрами модели и задаются пользователем. Наиболее часто параметр $\gamma$ в случае радиально-базисных функций устанавливается в диапазоне [0.0001, 0.1], параметр $C$ – в диапазоне [1, 100].

Отдельная SVM решает задачу бинарной классификации, т.е. способна разделить образы по принципу «свой-чужой». В случае текстовых символов число выходных образов соответствует числу букв в тестовом алфавите, поэтому в работе модель SVM расширяется на случай мультиклассовой классификации путем композиций автономных SVM, принимающих для каждого символа решение по типу «один против остальных».

Для решения задачи, рассматриваемой в статье, SVM обучаются и тестируются на буквах из алфавита английского языка. В тестовых сканах документов используются шрифты со сходным начертанием символов. При формировании вектора признаков для SVM двумерный образ сегментированного символа масштабируется к размеру 20x20 пикселей, центрируется и преобразовывается в одномерный вектор размерности 400 пикселей. Отметим, что это простейший способ представления символа. В дальнейшем планируется исследовать более эффективные и компактные способы описания символа на основе специальных дескрипторов изображения: гистограмм ориентированных градиентов (HOG), признаков SURF и др.

**Подготовка эксперимента и анализ результатов.** Информационная технология распознавания текста в зашумленных сканах документов основывается на методе, описанном в статье, и специально разработанном инструментальном комплексе TextRecognizer. Программный комплекс представляет собой пакет модулей на языке Python: denoising.py (очистка изображения от шумов), segmentation.py (сегментация), train.py

(обучение SVM) и recognize.py (распознавание текста). Модули опираются на внешние зависимости: библиотеки numpy, scikit-image и cv2 для цифровой обработки изображений, scikit-learn для машинного обучения.

Методология работы с инструментальным комплексом предполагает следующий набор действий:

1) подготовка конфигурационного файла в формате json, в котором указываются настраиваемые параметры всех алгоритмов, применяемых в системе;
2) подготовка директории train с файлами изображений (формат png, jpg или bmp) для обучения, директории test с файлами изображений для тестов и файлами с соответствующим текстом из каждого документа;
3) обучение SVM: система в автоматическом режиме работает с каталогом train; после сегментации исследователь уточняет каждый полученнный символ для корректного обучения; обученная SVM сохраняется в служебном файле;
4) пакетное распознавание: система запускает SVM, загруженную из служебного файла, на тестовых данных, которые она извлекает из каталога test.

Для эксперимента была обучена модель SVM на выборке из 12274 букв, выделенных в 25 изображениях. Среди изображений присутствовали, в том числе, тексты, на которые накладывались шумы в графическом редакторе искусственным путем. Тестовая выборка включала 10 документов с 4106 буквами. Свободные параметры модели выбирались с помощью методологии Grid Search (перебор всех сочетаний коэффициентов из определенных вариантов): параметр $\gamma$ – из списка [0.0001, 0.001, 0.01, 0.1], параметр $C$ – из списка [1, 10, 25, 50, 100]. Результаты распознавания текста варьируются от 72 % до 98 % для отдельных документов из тестовой выборки. Средний процент точности распознавания составил 87 %. Лучшие результаты ожидаемо демонстрируются на документах с меньшим шумом, более разборчивым текстом и более крупными символами.

**Выводы**. Проблема распознавания текста в зашумленных изображениях остается актуальной и нуждается в решении. Как видно из приведенных в статье иллюстраций, медианная фильтрация позволяет достаточно хорошо определять фон некачественно отсканированного документа для дальнейшей очистки. Алгоритм сегментации, предложенный и описанный в работе, позволяет с высокой точностью выделять строки с текстом и, с меньшей точностью, фрагменты изображения, содержащие буквы. Вариабельность начертания шрифтов и размеров символов, «склеивание» букв, остаточные шумы после первых шагов предобработки изображения – все эти факторы затрудняют как сегментацию, так и распознавание отдельных символов. Смягчить влияние указанных факторов можно в следующих направлениях: использовать в качестве образца, помимо буквы еще двухбуквенные и, возможно, трехбуквенные сочетания (слова для этой цели не подходят ввиду слишком большого числа вариантов); выполнять дополнительную постобработку символа в виде расчета специальных дескрипторов (HOG, SURF и др.). На этапе распознавания символов использовались машины опорных векторов. Результаты распознавания колебались для разных параметров SVM, документов и шрифтов в пределах 72–98 %.

Дальнейшая работа связана также с применением и детальным анализом в задаче распознавания текста других эффективных статистических моделей машинного обучения, помимо SVM, таких как: классификатор методом $k$ ближайших соседей, классификатор на основе случайного леса, сверточные нейронные сети и нейронные сети с глубоким обучением.

**СПИСОК ЛИТЕРАТУРЫ**


1. Pai N. Optical Character Recognition: An Encompassing Review / N. Pai, V.S. Kolkure // International Journal of Research in Engineering and Technology. – 2015. – Vol. 4. – P. 407-409.
2. Drinkwater R. The use of Optical Character Recognition (OCR) in the digitisation of herbarium specimen labels / R. Drinkwater, R. Hubey, E. Haston // PhytoKeys. – 2014. – Vol. 38. – P. 15-30.
3. Marinai S. Machine Learning in Document Analysis and Recognition / S. Marinai, H. Fujisawa. – Berlin: Springer Berlin Heidelberg, 2008. – 433 p.
4. Huang G. Bounding the Probability of Error for High Precision Optical Character Recognition / G. Huang, A. Kae, C. Doersch // The Journal of Machine Learning Research. – 2012. – Vol. 13 (1). – P. 363-387.
5. The Best OCR Software of 2016 / URL: http://ocr-software-review.toptenreviews.com / 19.02.2016.
6. Dokov R. Background Removal Script in Python / URL: https://www.kaggle.com/rdokov/denoising-dirty-documents/background-removal / 05.06.2015.
7. Xu C. Fast Algorithm for 2D Otsu Thresholding Algorithm / C. Xu, G. Peng // Journal of Computer Applications. – 2013. – Vol.32 (5). – P. 1258-1260.
8. Suzuki S. Topological Structural Analysis of Digitized Binary Images by Border Following / S. Suzuki, K. Abe // Computer Vision, Graphics and Image Processing. – 1985. – Vol. 30. – P. 32-46.
9. Minichino J. Learning OpenCV3 Computer Vision with Python / J. Minichino, J. Howse. – Packt. Publishing, 2015. – 266 p.
10. Steinwart I. Support Vector Machines / I. Steinwart, A. Christmann. – New York: Springer-Verlag, 2008. – 601 p.
11. Хайкин С. Нейронные сети: полный курс / С. Хайкин. – М.: Издательский дом «Вильямс», 2008. – 1102 с.


*Поступила в редакцию 23.03.2016 г.*

**TEXT RECOGNITION IN NOISY IMAGES OF SCANNED DOCUMENTS**

***T.V. Sharii, R.A. Lialin, A.Ye. Gukai, V.N. Kotenko***

The problem of transforming scanned documents of a relatively poor quality into text is still relevant today. The article offers a new robust method for recognizing text symbols in noisy images. The algorithms are described that have been applied for image de-noising, paragraph detection and letter segmentation. The results of text recognition experiments based on a Support Vector Machine statistical model are given.

***Keywords:*** OCR, machine learning, median filter, binarization, segmentation.

**Шарий Тимофей Вячеславович**
кандидат технических наук, доцент, tsphere@mail.ru

**Лялин Роман Олегович**
студент, bumagniypacket@yandex.ru

**Гукай Алексей Евгеньевич**
старший реподаватель, exxxar@gmail.com.

**Котенко Владислав Николаевич**
старший преподаватель, kotenko1967@gmail.com

ГОУ ВПО «Донецкий национальный университет», г. Донецк
Тел. для контактов: +38 (050) 769-61-79

**Sharii Timofei Viacheslavovich**
Candidate of Engineering Sciences, associate professor
**Lialin Roman Olegovich**
student
**Gukai Aleksei Yevgenievich**
senior lecturer
**Kotenko Vladislav Nikolaevich**
senior lecturer
Donetsk National University, Donetsk.