

МЕТОДЫ ОБРАБОТКИ ИЗОБРАЖЕНИЯ, СОДЕРЖАЩЕГО МАТЕМАТИЧЕСКИЕ ФОРМУЛЫ

Кирыушина А.Е.

ФГБУН «Институт программных систем имени А.К. Айламазяна» Российской академии наук,
Ярославская область, Переславский район, Веськово, e-mail: annkiryushina@gmail.com

Статья посвящена методам обработки сканированных изображений, содержащих математические формулы, для формализации их описания. Приводится классификация формул на простые и сложные, представлен способ упрощения структуры формулы путем ее разбиения на простые выражения относительно определенных математических символов (операторы с переменным диапазоном, скобки, знаки корня, равенства, интеграла и дроби). Поиск знаков «равно» и дроби основан на выделении горизонтальных линий на изображении. Основным инструментом распознавания выбрана сверточная нейронная сеть, которая дополнительно использована для распознавания букв греческого алфавита. В статье представлена архитектура сверточной нейронной сети, описан способ структурного анализа математических формул и символов, приведены результаты экспериментального разбора формулы и распознавания выделенных символов тестовой выборки, даны оценки точности и полноты распознавания.

Ключевые слова: обработка изображений, математическая формула, поиск объектов на изображении, скелетизация, распознавание, сверточная нейронная сеть

THE METHODS OF PROCESSING OF THE IMAGE WHICH CONTAINS MATHEMATIC FORMULAS

Kiryushina A.E.

Program Systems Institute of Russian Academy of Science, Yaroslavl region,
Pereslavl region, Veskovo, e-mail: annkiryushina@gmail.com

The paper describes methods of processing of images which contains mathematical formulas. There are methods in the paper that classifies formulas into simple and complicated ones. The method cutting a formula into simple ones searches for any variable range sing (a summation, an integral, a set-theoretic union and an intersection, a product and a coproduct), a square root sign, a rational line, an equals sign and brackets. An equals sign and a rational line are detected by horizontal lines searching. Convolutional neural network (CNN) is used for mathematical sings and Greek letters recognition. The architecture of CNN and recognition results are shown in the paper.

Keywords: image processing, mathematic formulae, object searching, skeletonization, convolution neuron network, object recognition

Для конвертации книг и документов в электронный вид обычно используют только их текстовую часть, что позволяет хранить информацию в компактной форме и добавлять в электронные библиотеки. В то же время практически не обрабатывается часть документа, содержащая математические и химические формулы, что препятствует получению полноценного документа в электронном виде. Сложность задачи формализации описания математических формул состоит в необходимости выделения формулы среди нематематического текста, анализа ее структуры, распознавания математических символов и букв.

В настоящее время зарубежные и отечественные ученые ведут исследования в области распознавания формул, на их основе созданы системы и сервисы, решающие задачи распознавания формул, в зависимости от их вида, полностью или частично [3–5, 11, 13]. Однако, несмотря на имеющиеся

успехи, следует отметить и имеющиеся недостатки. Так, в приложении InftyReader [3] не решена задача распознавания «многоэтажных формул» с индексами и степенями; в системе Freehand Formula Entry System (FFES) [13] (on-line распознавание рукописных формул) нет возможности распознавания сканированного изображения; в приложении EXTRAFOR [11] (выделение формул из текста) достигается только частичная обработка математического документа. В отечественных работах [4, 5] затронут лишь начальный этап исследований в области описания формул с помощью регулярных выражений и проведено моделирование формул с помощью графов, но нет практических результатов.

В настоящей статье представлены методы анализа математических формул произвольной сложности. Цель описанных методов заключается в выявлении и анализе межсимвольных связей, распознавании выделенных специальных математических

символов и букв. Для достижения поставленной цели разработаны и реализованы следующие методы:

– метод скелетизации (позволяет производить дальнейшую обработку символа вне зависимости от толщины символа, шрифта, наклона);

– метод поиска линий и выявление отдельных символов (используется для выделения на изображении таких математических знаков, как «дробь» и разновидности знака «равно», и некоторых других математических символов (знаки интеграла, скобки, знаки арифметических и логических операций) для разбиения формулы на более простые (подформулы).

В статье выполнены экспериментальные исследования и дана оценка качества полученных результатов.

Структурный анализ математических формул и символов

Формулой будем называть выражение, имеющее следующий вид:

$$\langle \text{левая часть} \rangle \langle \text{оператор} \rangle \langle \text{правая часть} \rangle, \quad (1)$$

где $\langle \text{оператор} \rangle$ – это арифметическая или логическая операция или знак отношений; $\langle \text{левая часть} \rangle$ и $\langle \text{правая часть} \rangle$ – выражения (подформулы). В свою очередь подформулы, как и формула, могут иметь вид (1).

Формула состоит из символов, к которым относятся цифры, буквы латинского и греческого алфавитов, знаки арифметических и логических операций, стрелки, скобки, специальные знаки, использующиеся в математических формулах (операторы

с переменным диапазоном: $\sum \int \prod \cap \cup$, символы, использующиеся в теории множеств: $\forall, \exists, \in, \notin, \subset, \supset, \subseteq, \supseteq, \varnothing$).

1. Структурный анализ формулы

По своему расположению в тексте формулы делятся на два типа: однострочные, встроенные в нематематический текст (например, «...уравнение прямой имеет вид $y = kx + b$, где...»), и многострочные, нахо-

дящиеся изолированно между частями нематематического текста, например:

$$y(x) = \frac{e^x - e^{-x}}{4}.$$

В дальнейшем ограничимся рассмотрением сканированных изображений с одной формулой, предварительно выделенной из электронного документа. Первоначальным этапом анализа формулы является ее разбиение на отдельные символы. Эта задача решается при помощи программной системы ПС ИНС [7], в которой реализована схема разделения исходного изображения на изображения с отдельными символами. Система имеет развитый графический интерфейс, в котором каждый блок представляет отдельный программно реализованный модуль, который можно соединять с другими модулями посредством входных и выходных каналов. На рис. 1 представлена схема, реализующая алгоритм сегментации изображения, в результате которого будет получен набор изображений с отдельными символами.

Процесс получения на выходе серии изображений с отдельным символом формулы состоит из следующих шагов:

1) чтение (ReadImages) исходного изображения из памяти;

2) удаление фона (HistDelete) на входном цветном изображении путем анализа гистограммы. Полученное цветное изображение, в котором фон представлен alpha-каналом, подается на сегментацию;

3) сегментация (FindObjects, GetObjects) основана на маркировке связных компонент на цветном изображении. Этот метод эффективно работает на символах, состоящих из одной компоненты (например, 1, 2, A, B), а для символов из нескольких составляющих ($j, i, =, \%$) требуется дополнительная обработка для объединения, прежде чем символы будут поданы на распознавание;

4) передача (ImageWriter) полученных сегментов формулы на следующий этап обработки, использующий на входе серию изображений.

Рассмотрим последующие шаги обработки изображения, которые были реализованы в системе ПС ИНС непосредственно автором настоящей статьи.



Рис. 1. Схема сегментации отдельных элементов формулы

2. Морфология изображений

2.1. Скелетизация

Для описания символов, представленных матрицей точек, выполняется процедура скелетизации. Скелет представляет собой плоский граф, отражающий особенности формы объекта [12]. Вершины скелета – это центры окружностей, касающихся границы фигуры в трех или более точках, а ребра – линии, состоящие из точек-центров вписанных окружностей, касающихся в двух или более точках. Для нахождения скелетов предварительно бинаризованного после сегментации изображения использован алгоритм Зонга-Суня [14]. При сканировании символа применена 8-связная структура соседства пикселей. «Перекрашиваются» в белый цвет те черные граничные точки, которые не нарушают связности оставшейся фигуры. Благодаря 8-связной области происходит последовательное закрашивание пикселей краев «север-запад» и «юг-восток». Процесс повторяется до тех пор, пока не останется пикселей, которые можно переокрасить.

Алгоритм скелетизации упрощает дальнейшую работу с изображением

и устойчив к шуму. К недостаткам такого преобразования символа относятся вероятность нарушения связности и относительно большие временные затраты.

Следующий шаг состоит в поиске линий на скелетном изображении для извлечения информации о содержащихся в формуле символов дроби, отрицаний, знаков «=», «-». Если данная информация подтверждается, то выполняется исследование окружающих эти знаки символов.

2.2. Выделение линий

Для выделения линий на изображении используется преобразование Хафа [8]. Прямые описываются с помощью уравнения $r = x \cos \theta + y \sin \theta$ при условии, что $\theta \in [0, \pi]$ и $r \in R$, где r – величина вектора от начала координат до ближайшей точки на прямой; θ – угол между этим вектором и осью координат.

Схема на рис. 2 демонстрирует в комплексе: реализацию сегментации входного изображения (рис. 3) на символы, построение бинарного изображения, приведение к скелетному изображению и выделение прямых. На выходе схемы получаем скелет входного изображения с выделенными линиями (рис. 4).

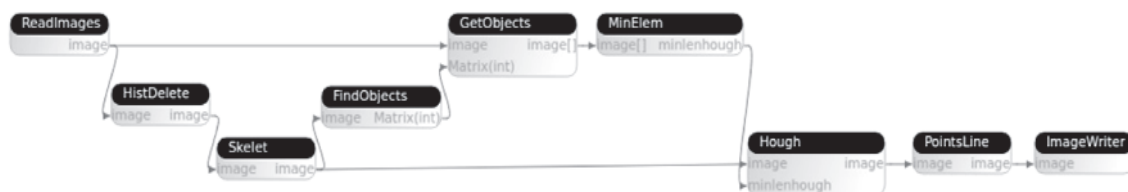


Рис. 2. Схема выделения линий на изображении

$$b_n = \frac{1}{L} \int_{-L}^L f(x) \sin\left(\frac{n\pi x}{L}\right) dx$$

Рис. 3. Пример входного изображения

$$b_n = \frac{1}{L} \int_{-L}^L f(x) \sin\left(\frac{n\pi x}{L}\right) dx$$

Рис. 4. Результат выделения линий на скелетизированном изображении (отмечены серым)

Распознавание символов на основе метода базовых локальных признаков

1. Структурный анализ формулы. Выделение подформул

Следующим шагом структурного анализа формулы является разбиение ее на подформулы. Разбиение формулы происходит до тех пор, пока не останется делимого на символы выражения. Каждый символ, соединяющий две подформулы, имеет свой приоритет, согласно которому определяется последовательность поиска символа в формуле (приоритет «1» – для =, ≠, ≠, приоритет «2» – для стрелок, приоритет «3» – для ≤, ≥, <, >, приоритет «4» – для знаков арифметических и логических операций). Дополнительными символами соединения подформул являются: операторы с переменным диапазоном (∑ ∫ ∏ ∩ ∪), знак корня, горизонтальной черты и скобки.

При помощи метода базовых локальных признаков [8] формируется множество признаков распознаваемых объектов, характерных для определенного символа. В качестве множества признаков будут выступать угловые точки, найденные на символе. Распознавание объекта происходит путем сопоставления признаков модели и признаков изображения. В статье будет освещено распознавание символов с приоритетом «1» и операторов с переменным диапазоном, корня и знака горизонтальной черты.

2. Обнаружение параллельных прямых

Пусть на изображении имеется набор прямых, найденных с помощью метода Хафа [8]. Прямая L_k представлена координатами начальной точки (x_{k1}, y_{k1}) и конечной точкой (x_{k2}, y_{k2}) . Для обнаружения объектов с приоритетом «1» из набора выбираются те прямые L_k, L_j , которые соответствуют следующим признакам:

– расстояние между точками (x_{k1}, y_{k1}) и (x_{j2}, y_{j2}) меньше или равно $\min(L_k, L_j)$, где L_k, L_j – длины k -й и j -й прямых;

$$-\frac{L_k}{L_j} \leq \varepsilon, \quad \varepsilon \leq \min(L_k, L_j).$$

Дополнительно определяются понятия «главное равно» и «второстепенное равно». «Главное равно» соединяет две подформулы, «второстепенное равно» входит в состав подформулы. Входными данными для обнаружения параллельных прямых является предварительно промаркированное, скелетизированное бинарное изображение, на котором горизонтальным прямым присвоены метки. На рис. 5 представлен результат

детектирования «главного равно», отмеченного цифрой 1, «второстепенного равно» – цифрой 2.

$$I_3(A, B) \stackrel{(1)}{=} \sum_{i=1 \dots 7}^{(2)} \frac{|m_i^A \stackrel{(4)}{=} m_i^B|}{|m_i^A|} \stackrel{(3)}$$

Рис. 5. Результат выделения знака «главное равно» (1), «второстепенное равно» (2), горизонтальных линий (3, 4)

3. Угловые точки как вершины графа

Поиск угловых точек как вершин графа является способом, инвариантным к аффинным преобразованиям, основан на направлении градиентов соседних пикселей. Детектор Харриса проигрывает в скорости работы по сравнению с другими алгоритмами [12], но дает точные результаты.

На вход алгоритма Харриса подается полутоновое изображение. Проход по изображению осуществляется с использованием дифференциальной маски. Определяется изменение яркости соседних пикселей в различных направлениях относительно текущего пикселя. Точка является угловой, если изменение градиента наибольшее, а сдвиг окна (относительно текущего пикселя) мал.

Каждый символ можно представить в виде графа $G = (X, V)$, где $X = (point_1, point_2, \dots, point_n)$ – множество вершин графа; $V = (v_1, v_2, \dots, v_n)$ – множество ребер графа. На рис. 6 представлен результат работы метода Харриса: помечены угловые точки.

В качестве примера в табл. 1 представлена матрица смежности для символа, изображенного на рис. 6. Значение «1» на пересечении i -го столбца и j -й строки означает наличие ребра между вершинами p_i и p_j , «0» – отсутствие ребра между соответствующими вершинами. Программным средством выбрана библиотека OpenCV [2], которая содержит реализацию метода поиска углов по Харрису [10]. На основе метода выделяются угловые точки.



Рис. 6. Узловые точки графа для знака корня

Таблица 1

Матрица смежности вершин графа для знака корня

	P_0	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}
P_0		1	0	0	0	0	0	0	0	0	0
P_1	1		1	0	0	0	0	0	0	0	0
P_2	0	1		1	0	0	0	0	0	0	0
P_3	0	0	1		1	0	0	0	0	0	0
P_4	0	0	0	1		1	0	0	0	0	0
P_5	0	0	0	0	1		1	0	0	0	0
P_6	0	0	0	0	0	1		1	0	0	0
P_7	0	0	0	0	0	0	1		1	0	0
P_8	0	0	0	0	0	0	0	1		1	0
P_9	0	0	0	0	0	0	0	0	1		1
P_{10}	0	0	0	0	0	0	0	0	0	1	

4. Изменение размера изображения

Все изображения с выделенными символами должны быть приведены к одному размеру 32×32 . Число 32 выбрано не случайно: во-первых, все изображения, подаваемые на вход сверточной нейронной сети (далее СНС) (о которой будет рассказано ниже), должны быть одного размера, во-вторых, на первом и втором слоях СНС размер карт признаков является целым числом [6]. Входными изображениями являются изображения, полученные после сегментации изображения.

Если размер изображения меньше 32×32 , то оно наращивается до 32×32 . Далее изображения поступают на блок распознавания.

Распознавание математических символов и букв греческого алфавита с использованием сверточной нейронной сети

1. Архитектура СНС для распознавания греческих букв

СНС – нейронная сеть, имеющая многослойную архитектуру. Слои предназначены для возможности быстрого обучения и уменьшения общего значения функции ошибки [9].

На вход сети и каждого последующего слоя поступают изображения, представленные в виде двумерных массивов. Модель СНС представляет собой последовательность сверточных и вычислительных слоев.

Выбранная в настоящей работе модель сети состоит из пяти слоев: входного, трех

скрытых слоев свертки и классификационного выходного. Локальное рецептивное поле сверточных нейронов [9] имеет смещение на два пикселя и размер 5×5 . Подробная архитектура сети представлена на рис. 7.

Входными данными нейронной сети являются сегментированные изображения греческих букв из математической формулы. Таким образом, входной слой содержит 1024 нейрона. В эксперименте размер обучающей выборки составил 362 изображения, размер тестирующей выборки – 196 символов.

Второй слой СНС (первый слой свертки) состоит из шести карт признаков размером 14×14 . В данном слое находится $14 \times 14 \times 6 = 1176$ нейронов. Каждый элемент признаков соединен с рецептивным полем размера 5×5 из входного слоя. Из этого следует, что каждый элемент карты имеет 26 обучаемых весов. Таким образом, в первом сверточном слое находится $1176 \times 26 = 30576$ синаптических связей и 156 обучаемых весов. Перейдем к описанию третьего сверточного слоя.

Третий слой, второй слой свертки, состоит из 50 карт признаков размером 5×5 и содержит $5 \times 5 \times 50 = 1250$ нейронов. Каждый элемент карты имеет связь с шестью рецептивными полями размером 5×5 из предыдущего слоя. Таким образом, во втором сверточном слое находится 7550 весовых коэффициентов и 188750 синаптических связей. Данные сверточные слои используются для извлечения признаков из изображения. Каждый нейрон соединен со всеми нейронами из предыдущего слоя.

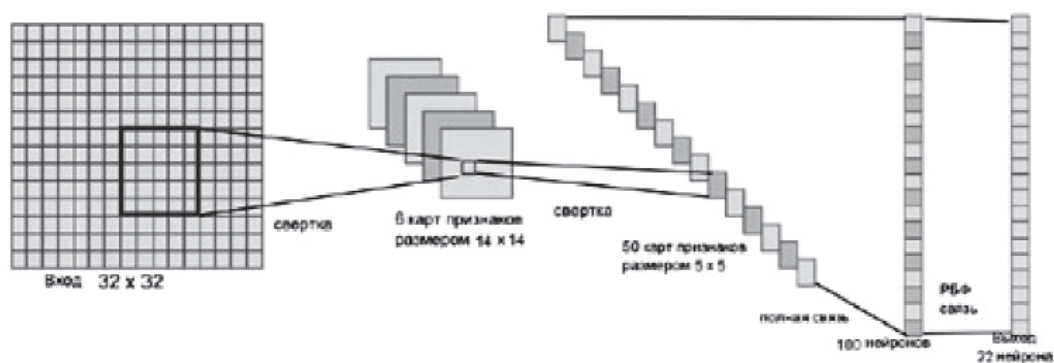


Рис. 7. Архитектура СНС для распознавания букв греческого алфавита

Четвертый слой, открытый, состоит из 100 нейронов. В этом слое содержится 125100 связей и столько же параметров для обучения.

Пятый слой, выходной, состоит из 22 нейронов, так как распознается 22 символа из алфавита. В последнем слое содержится 2222 связей и столько же весов.

Обучение построенной СНС производится с помощью метода обратного распространения ошибки [14]. В данном случае учитывается архитектура сети, а именно то, что веса одной карты признаков используются совместно.

2. Результаты распознавания

Выбранных значений параметров настройки сети оказалось достаточно, что-

бы получить довольно высокий процент распознавания. В табл. 2 представлены результаты распознавания тестовой выборки. Табл. 3 демонстрирует точность распознавания обучающей и тестовой выборки.

3. Архитектура СНС для распознавания математических символов

Для распознавания операторов с переменным диапазоном и скобок использовалась СНС с 11 выходными классами. Результаты распознавания символов представлены в табл. 4. Обучающая выборка составила 330 изображений, тестовая – 110. Средняя точность распознавания составила 97,9%.

Таблица 2

Результаты распознавания букв греческого алфавита

Символ	Процент распознавания	Символ	Процент распознавания
α	100%	π	100%
β	75%	ρ	100%
γ	100%	σ	100%
δ	100%	ζ	100%
ε	100%	τ	100%
ζ	100%	ν	100%
η	100%	φ	100%
λ	100%	χ	88,2353%
μ	100%	ψ	100%
ν	100%	ω	100%
ξ	100%	φ	100%

Таблица 3

Результат обучения сети

Точность распознавания обучающей выборки (362 изображения), %	Точность распознавания тестовой выборки (196 изображений), %	Количество эпох обучения
100	98,07	90

Таблица 4

Результат распознавания операторов с переменным диапазоном и скобок

Класс	Процент распознавания	Класс	Процент распознавания
U	100	(100
∩	100)	100
Π	100	{	85,7
Σ	100	}	88,9
∫	100		

Выводы

В статье представлены методы разделения формулы на подформулы с целью изучения структуры формулы и ее упрощенного представления; определены символы, которые выступают в качестве разделителей двух подформул и описаны способы выявления на изображении: знака « \Rightarrow », операторов с переменным диапазоном, линии дроби, скобок. С помощью метода базовых локальных признаков построена модель символа, представлены результаты распознавания букв греческого алфавита, операторов с переменным диапазоном и скобок. Инструментом для распознавания выбрана сверточная нейронная сеть. При этом средняя точность распознавания букв греческого алфавита составила 98,07%, операторов и скобок – 97,9%. Цель дальнейших исследований заключается в восстановлении межсимвольных связей и представлении формулы в удобном для пользователя виде.

Список литературы

1. Метод обратного распространения ошибки. Описание метода [Электронный ресурс]. – Режим доступа: <http://www.aiportal.ru/articles/neural-networks/back-propagation.html> (дата обращения: 18.06.2015).
2. Сайт библиотеки OpenCV. Детектор углов Харриса [Электронный ресурс]. – Режим доступа: http://docs.opencv.org/doc/tutorials/features2d/trackingmotion/harris_detector/harris_detector.html (дата обращения: 18.06.2015).
3. Сайт приложения InftyReader. Раздел «Examples» [Электронный ресурс]. – Режим доступа: <http://www.inftyreader.org/?p=61> (дата обращения: 18.06.2015).
4. Сапаров А.Ю., Бельтюков А.П. Математическое моделирование изображений формул с целью их распознавания // Вестник Удмуртского университета. Математика, механика, компьютерные науки. – 2013. – Вып. 1. – С. 153–167.
5. Сапаров А.Ю., Бельтюков А.П. Применение регулярных выражений в распознавании математических текстов // Вестник Удмуртского университета. Математика, механика, компьютерные науки. – 2012. – Вып. 2. – С. 63–73.
6. Солдатова О.П., Гаршин А.А. Применение сверточной нейронной сети для распознавания рукописных цифр // Компьютерная оптика. – 2010 – Т. 34. – С. 251–259.
7. Талалаев А.А., Фраленко В.П. Архитектура комплекса конвейерно-параллельной обработки данных в гетерогенной вычислительной среде // Вестник РУДН. Серия Математика. Информатика. Физика. – 2013. – № 3. – С. 113–117.
8. Шапиро Л., Стокман Д. Компьютерное зрение. – М: БИНОМ, Лаборатория знаний, 2006. – 752 с.
9. Backpropagation Applied to Handwritten Zip Code Recognition / Y. LeCun, B. Boser, J. Denker and others // Neural Computation. – 1989. – Vol. 1, № 4. – P. 541–551.
10. Harris Ch., Stephens M. A combined corner and edge detector // In Proc. of Fourth Alvey Vision Conference. – University of Manchester, 1988. – Session. 6. – P. 147–151.
11. Kacem A. EXTRAFOR: Automatic EXTRACTION of mathematical FORMulas / Belaid A., Ahmed M.B. // Proceedings

of the Fifth Intentional Conference on Document Analysis and Recognition ICDAR'99. – Bangalore-India, 1999. – P. 527–530.

12. Trajkovich M. Fast corner detection // Image and Vision Computing. – 1998. – Vol. 16. – P. 75–87.

13. Zanibbi R. Recognizing Mathematical Expressions Using Tree Transformation / R. Zanibbi, D. Blostein, James R. Cordy // IEEE. – Transactions On Pattern Analysis and Machine Intelligence. – 2002. – Vol. 24. – № 11. – P. 1455–1467.

14. Zhang T.A fast parallel algorithm for thinning digital patterns / T. Zhang, C. Suen // Communications of the ACM. – 1984. – № 3, Vol. 27. – P. 236–239.

References

1. *Metod obratnogo rasprostraneniya oshibki (Opisanie metoda)*, Available at: <http://www.aiportal.ru/articles/neural-networks/back-propagation.html> (accessed 18 June 2015).
2. *Sajt biblioteki OpenCV (Detektor uglov Harris)*, Available at: http://docs.opencv.org/doc/tutorials/features2d/trackingmotion/harris_detector/harris_detector.html (accessed 18 June 2015).
3. *Sajt prilozhenija InftyReader (Razdel «Examples»)*, Available at: <http://www.inftyreader.org/?p=61> (accessed 18 June 2015).
4. Сапаров А.Ю., Бельтюков А.П. *Matematicheskoe modelirovanie izobrazhenij formul s celju ih raspoznavaniya – Vestnik Udmurtskogo universiteta. Matematika, mehanika, kompjuternye nauki*, 2013, Vyp. 1, pp. 153–167.
5. Сапаров А.Ю., Бельтюков А.П. *Primenenie reguljarnyh vyrazhenij v raspoznavanii matematicheskikh tekstov // Vestnik Udmurtskogo universiteta. Matematika, mehanika, kompjuternye nauki*, 2012, Vyp. 2, pp. 63–73.
6. Солдатова О.П., Гаршин А.А. *Primenenie svertochnoj nejronnoj seti dlja raspoznavaniya rukopisnyh cifr*, 2010, T. 34, pp. 251–259.
7. Талалаев А.А., Фраленко В.П. *Arhitektura kompleksa konvejerno-parallelnoj obrabotki dannyh v geterogennoj vychislitelnoj srede – Vestnik RUDN. Serija Matematika. Informatika. Fizika*, 2013, no. 3, pp. 113–117.
8. Shapiro L., Stokman D. *Kompjuternoje zrenie*. Moscow, BINOM, Laboratorija znaniy, 2006. 752 p.
9. Y. LeCun, B. Boser, J. Denker *Backpropagation Applied to Handwritten Zip Code Recognition – Neural Computation*, 1989, Vol. 1, no. 4, pp. 541–551.
10. Ch. Harris, M. Stephens *A combined corner and edge detector – In Proc. of Fourth Alvey Vision Conference. – University of Manchester*, 1988, Session. 6, pp. 147–151.
11. Kacem A., Belaid A., Ahmed M.B. *EXTRAFOR: Automatic EXTRACTION of mathematical FORMulas – Proceedings of the Fifth Intentional Conference on Document Analysis and Recognition ICDAR99*, Bangalore-India, 1999, pp. 527–530.
12. Trajkovich M., Handley M. *Fast corner detection – Image and Vision Computing*, 1998, Vol. 16, pp. 75–87.
13. Zanibbi R., Blostein D., James R. *Cordy Recognizing Mathematical Expressions Using Tree Transformation – IEEE, Transactions On Pattern Analysis and Machine Intelligence*, 2002, Vol. 24, no. 11, pp. 1455–1467.
14. Zhang T., Suen C. *A fast parallel algorithm for thinning digital patterns – Communications of the ACM*, 1984, no. 3, Vol. 27, pp. 236–239.

Рецензенты:

Хачумов В.М., д.т.н., профессор, заведующий лабораторией 0-4 ИСА РАН, г. Москва; Знаменский С.В., д.ф.-м.н., заведующий лабораторией информатизации образования Институт программных систем им. А.К. Айламазяна РАН, Ярославская обл., Переславский р-н, Веськово.