УДК 004.93’1

ИСПОЛЬЗОВАНИЕ РАСТЕРИЗАЦИИ МЕТОДОМ БРЕЗЕНХЭМА ДЛЯ МЕТОДА ПЕРЕСЕЧЕНИЙ ПРИ ОПТИЧЕСКОМ РАСПОЗНАВАНИИ ПЕЧАТНЫХ СИМВОЛОВ

П.А. Хаустов, Е.И. Максимова

Томский политехнический университет
E-mail: exceibot@sibmail.com; yelenamaksimova@yandex.ru

Поиск решения задачи оптического распознавания символов актуален при разработке алгоритмов индексации и анализа оцифрованных документов. Предложен алгоритм для оптического распознавания печатных и рукописных символов на основе метода пересечения. Для представления отрезков, пересекающих изображение символа, было предложено использовать алгоритм растеризации Брезенхэма, который отличается своей простотой в реализации и отсутствием необходимости использовать вычисления с плавающей точкой. Метод был апробирован на наборе изображений печатных символов латинского алфавита с пиксельным шумом, извлеченных из реальных отсканированных документов. В результате работы алгоритма было верно распознано 91 % символов.

Ключевые слова:
Оптическое распознавание символов, растеризация, метод Брезенхэма, целочисленная арифметика, рукописные символы, печатные символы, обработка изображений.

Хаустов Павел Александрович, аспирант кафедры вычислительной техники Института кибернетики ТПУ.
E-mail: exceibot@sibmail.com
Область научных интересов: абстрактные типы данных, решения задачи оптического распознавания рукописных и печатных символов, алгоритмы обработки изображений.

Максимова Елена Ивановна, студент кафедры вычислительной техники Института кибернетики ТПУ.
E-mail: yelenamaksimova@yandex.ru
Область научных интересов: исследование задач и алгоритмов теории графов, алгоритмы обработки изображений.

Введение

Сегодня все чаще создаются базы, в которых хранятся отсканированные копии документов. Для того чтобы осуществлять контекстный поиск по таким документам и их классификацию, необходимо решить задачу перевода подобных изображений из графического формата в текстовый. Для реализации такого перевода требуется решить задачу оптического распознавания символов.

Сама по себе задача оптического распознавания символов является трудноформализуемой и имеет множество приближенных методов решений, каждый из которых обладает своими достоинствами и недостатками.

Большинство методов, которые применяются для решения задачи оптического распознавания символов, аналогичны методам, используемым для других задач классификации изображений. Как правило, такие методы реализуют представление каждого символа в виде некоторого вектора в пространстве признаков. Классификация по вектору признаков зачастую осуществляется с использованием искусственных нейронных сетей или аналогичных классификаторов, которые не позволяют отследить логику принятия решения о принадлежности символа тому или иному классу [1].

Отдельно стоит отметить методы, специально разработанные для бинаризированных изображений, и, в частности, изображений печатного текста. В таком случае задачу оптического распознавания символов можно решать, используя априорную информацию о том, какой из пикселов принадлежит объекту, а какой – фону. Особую роль среди подобных подходов к решению задачи оптического распознавания символов играет группа методов, основанных на идее анализа пересечений графического представления символа с некоторым набором отрезков. Подобный подход в литературе получил название «метод пересечений» [2].

Идею метода пересечений можно интерпретировать огромным количеством способов: можно варьировать набор отрезков, способ геометрического представления символа, правила учета количества и общей длины пересечений каждого из отрезков с этим представлением.

Особое внимание стоит уделить способу геометрического представления символа. Так как речь идет об отсканированном изображении, то необходимо осуществлять анализ растрового фрагмента некоторого графического документа. Наиболее простым вариантом можно назвать анализ пересечений символа именно в исходном виде, но несложно заметить некоторые существенные недостатки подобного подхода. Для полного понимания недостатков такого подхода можно рассмотреть следующий пример (рис. 1).

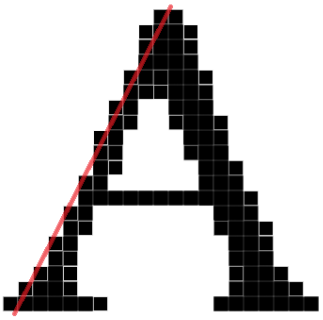

Рис. 1. Пример нежелательного многократного пересечения отрезка и графического представления символа

Для случая, изображенного на рис. 1, характерно многократное пересечение отрезком графического представления символа. Фактически группа пересекаемых пикселов до растеризации представляла собой отрезок, который не может иметь более одного интервала пересечения. Таким образом, можно заметить, что представление символа предложенным способом существенно искажает информацию о пересечениях изображения символа с большим количеством отрезков.

В таком случае можно сделать вывод о том, что наиболее точно описывает геометрию символа лишь его векторное представление. В общем случае символ будет представлен набором графических примитивов, которые будут ограничивать некоторые области изображения. Одна часть таких областей будут принадлежать символу, а другая – фону.

При подобном представлении символа интервалы пересечения любого отрезка с начертанием этого символа можно определить максимально точно. Но в таком случае возникает проблема высокой вычислительной стоимости анализа интервалов пересечения. Для нахождения каждого из интервалов необходимо анализировать взаимное положение графических примитивов и замкнутые области, ограниченные этими примитивами и границами изображения. Подобного рода алгоритмы обладают низким быстродействием и требуют выполнения большого количества операций с плавающей точкой. Также стоит отметить достаточно высокую сложность программной реализации при подобном представлении графической формы начертания символа.

Для того чтобы избежать высокой вычислительной сложности и трудной программной реализации, необходимо некоторым образом перевести каждый из отрезков в растровый формат, что приведет к существенному упрощению анализа пересечений. Остается лишь выбрать способ перевода отрезков из векторного формата в растровый.

Алгоритм Брезенхэма

Несложно заметить, что существует множество способов перевести векторный отрезок в растровый формат изображения. Для того чтобы определить, какие пикселы двумерного растра необходимо отнести к изображению отрезка, требуется анализировать их расположение относительно этого отрезка. Джек Е. Брезенхэм в 1962 году предложил способ определения принадлежности каждого из пикселов двумерного растра отрезку с использованием лишь целочисленной арифметики. Такой подход получил название «алгоритм Брезенхэма» [3].

В ходе алгоритма предполагается проход по каждому значению координат одной из осей и определение значения координаты на другой оси для размещения очередного пиксела.

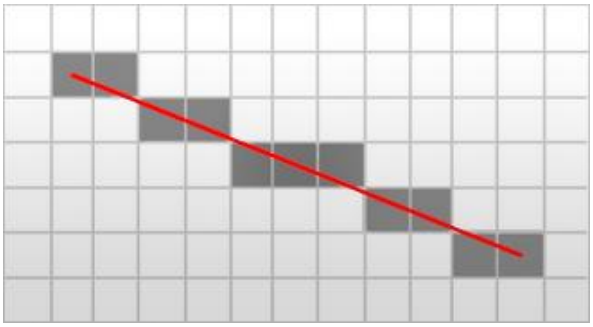

Рис. 2. Иллюстрация работы алгоритма Брезенхэма

Для того чтобы определить наиболее подходящее значение на другой оси координат, необходимо выбрать наиболее близкий к отрезку пиксел двумерного растра. Такую задачу можно решить, используя только целочисленную арифметику, с помощью уравнения прямой в отрезках.

Метод пересечений с использованием алгоритма Брезенхэма

При использовании растрового представления отрезков в методе пересечений анализ интервалов наложения изображения отрезка на изображение символа существенно упрощается. Наиболее простым из допустимых вариантов оценки меры пересечения является тривиальный подсчет количества общих пикселов у двух растровых представлений. В таком случае геометрические потери при растеризации гораздо меньше влияют на результат распознавания. Для более точного описания геометрических потерь можно воспользоваться иллюстрацией.

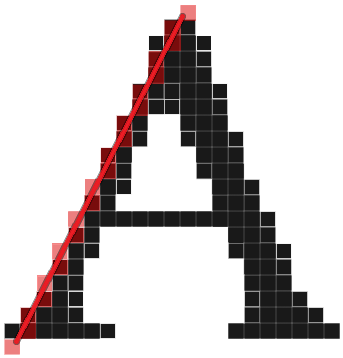

Рис. 3. Иллюстрация отрезка, его растрового представления и его наложения на графическое представление символа

На рис. 3 оттенками красного цвета показаны пикселы, принадлежащие растеризации изображенного отрезка. Темный оттенок соответствует пикселам, которые относятся к графическому представлению символа, светлый – остальным пикселам растрового представления отрезка. Как можно заметить, темно-красные пикселы ввиду некоторой погрешности растрового представления образуют множество непрерывных интервалов наложения, что при точном геометрическом представлении соответствовало бы лишь одному интервалу наложения. При статистическом подсчете количества общих точек такая погрешность не выглядит столь существенной, какой может оказаться при более детальном геометрическом анализе. Для приведенного на рис. 3 примера вместо одного интервала наложения можно увидеть сразу пять интервалов, что может привести к заблуждению анализ количества непрерывных пересечений.

Таким образом, можно сформулировать идею алгоритма оптического распознавания символов на основе метода пересечений и алгоритма Брезенхэма.

Шаг 1. Сформировать некоторый набор отрезков.

Шаг 2. Получить растровое представление каждого из отрезков, используя алгоритм Брезенхэма.

Шаг 3. Сформировать набор изображений символов – шаблонов, с которыми будет осуществляться сравнение.

Шаг 4. Для каждого изображения-шаблона произвести наложение каждого из отрезков набора. Для каждого из таких наложений O_i запомнить количество общих пикселей C_i. В дальнейшем вектор значений $V = \{C_1, C_2, \ldots, C_K\}$ будет использоваться для представления данного изображения-шаблона.

Шаг 5. Для каждого из изображений, класс которого необходимо определить, описанным в шаге 4 образом получить вектор $V = \{C_1, C_2, \ldots, C_K\}$. Найти наиболее похожий вектор в наборе шаблонов, определить класс, к которому относится символ, соответствующий этому вектору. Отнести к этому классу данное изображение.

Результаты апробации алгоритма

Для тестирования предложенного алгоритма было реализовано консольное приложение на языке C++, которое случайным образом генерирует набор отрезков и реализует описанный ранее алгоритм. Для оценки качества распознавания был выбран набор изображений символов латинского алфавита, полученных из реальных отсканированных документов. Пример таких изображений можно увидеть на рис. 4.

Рис. 4. Примеры изображений символов в тестовом наборе

В качестве шаблонов было выбрано около 20 изображений символов каждого класса. Для тестирования было выбрано 100 изображений символов каждого класса.

В результате апробации алгоритма было установлено, что предложенная вариация метода пересечений верно распознает 91 % символов тестовой выборки.

Таким образом, можно сделать вывод, что был предложен алгоритм, который не требует ни одной операции с плавающей точкой, способный корректно распознавать более 90 % символов в отсканированных документах.

Стоит отметить, что метод пересечений дает некоторое преимущество при создании каскада классификаторов с его использованием. После того, как первичный классификатор определил группу классов, к которым с наибольшей долей достоверности можно отнести текущий символ, рационально использовать далее дифференциальный классификатор, который позволит выбрать из выделенных классов наиболее подходящий. В дифференциальном классификаторе можно использовать более узкоспециализированный набор отрезков, который будет предназначен специально для поиска различий выбранной группы классов. В дальнейшем планируется реализация каскадной системы классификаторов на основе метода пересечений с целью улучшения качества распознавания.

СПИСОК ЛИТЕРАТУРЫ

1. Schantz, Herbert F. The history of OCR, optical character recognition / «Recognition Technologies Users Association», 1982. – 213 p.
2. Роджерс Д. Алгоритмические основы машинной графики. – М.: Мир, 1989. – С. 54–63.
3. Bresenham J. E. Algorithm for computer control of a digital plotter // IBM Systems Journal. – 1965. – V. 4. – P. 25–30.

Поступила 06.10.2014