

**Ордена Ленина  
ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ  
имени М.В.Келдыша  
Российской академии наук**

**С.В.Андреев, А.В.Бондаренко, В.И.Горемычкин,  
А.В.Ермаков, С.Ю.Желтов**

**Алгоритмическое обеспечение  
прототипа устройства считывания  
машиночитаемых документов.**

**Москва 2003**

**УДК 519.68**

***С.В.Андреев, А.В.Бондаренко, В.И.Горемычкин, А.В.Ермаков, С.Ю.Желтов*** Алгоритмическое обеспечение прототипа устройства считывания паспортов и виз

Данная работа посвящена разработке прототипа алгоритмического и программного обеспечения для устройства считывания машиночитаемых документов. В качестве примера рассмотрены основные принципы построения системы автоматического считывания паспортов и виз. Предложены оригинальные алгоритмы обнаружения текстовых строк, определения знако-мест и адаптивной бинаризации изображений. Для распознавания символов OCR-B-кода реализованы метод зон и метод пересечений.

***S.V.Andreev, A.V.Bondarenko, V.I.Goremychkin, A.V.Yermakov, S.Y.Zgeltov***  
**Algorithmic support of the device prototype for passports and visas reading.**

The given work is devoted to development of the prototype of algorithms and software for the device of reading of machine-readable documents. As an example the basic principles of construction of system of automatic reading of the passports and visas are considered. The original algorithms of detection of textual lines, definition of symbol places and adaptive binarization of the images are offered. Method of zones and method of crossings are implemented for recognition of OCR-B-code symbols.

## **Введение.**

Разработка прототипа алгоритмического и программного обеспечения устройства считывания документов с использованием шрифта "OCR-B" является актуальной задачей с точки зрения автоматизации обработки машиночитаемых документов. В качестве примера могут быть рассмотрены машиночитаемые визовый талон или паспорт, выполненные в соответствии с рекомендациями ICAO.

Визовый талон разделен на две зоны: (А) визуальную, предназначенную для визуальной проверки документа, и (Б) машиночитаемую - для оперативного автоматизированного контроля. В структуру двух строк машиночитаемой зоны бланка визового талона включена информация из визуальной зоны бланка визового талона. Машиночитаемая зона состоит из двух строк по 44 символа, заполненных с использованием шрифта OCR-B type 1 (Стандарт ИСО 1073/II).

Авторами в процессе работы решались следующие задачи:

1. Разработка алгоритмического и программного обеспечения, позволяющего сегментировать на изображении документа поля, подлежащие считыванию.
2. Разработка алгоритмического и программного обеспечения, позволяющего считывать текстовую информацию заданного типа (OCR-B-коды).
3. Разработка алгоритмического и программного обеспечения для считывания цифровых изображений из соответствующих полей документа. Исследование эффективных алгоритмов сжатия фотографических изображений конкретного типа.

В работе описывается алгоритмическое обеспечение прототипа устройства считывания машиночитаемых документов, разрабатывается общая модульная структура алгоритмического обеспечения. Предлагаются алгоритмы обнаружения текстовых строк, определения знако-мест и адаптивной бинаризации изображений. Описываются реализованные методы распознавания символов OCR-B-кода, а также способы постпроверки результатов распознавания.

## **1. Постановка задачи и общая структура алгоритмического обеспечения.**

На этапе разработки алгоритмического обеспечения прототипа устройства считывания рассматривалась следующая постановка задачи.

Стенд-макет, состоящий из портативной черно-белой телевизионной камеры, фреймграбера и персонального компьютера, должен считывать коды OCR-B с документов, находящихся перед камерой. В тот момент, когда объект находится перед камерой, видеоданные фиксируются и помещаются в память компьютера. Полученное изображение является объектом последующей обработки.

Каждый пиксел этого изображения имеет значение, описываемое восьмью битами (256 полутонов). Размеры получаемого изображения составляют не менее 750x582 пикселей. Необходимо декодировать информацию, записанную в машиночитаемой зоне документа с использованием кода OCR-B.

Предполагается, что изображение находится в памяти как последовательный массив байт, начиная с некоторого известного адреса. Каждый байт изображения содержит значение одного пиксела. Первые восемь бит массива содержат значение яркости левого верхнего пиксела изображения. Порядок расположения пикселей в памяти - "слева направо по

каждой строке; сверху вниз - по строкам".

Символы кода обязательно должны быть темнее фона (обратный контраст не допускается), однако не обязательно совершенно черными.

Минимальный допустимый размер символов кода на цифровом изображении - 10x15 пикселей.

При разработке алгоритмического обеспечения необходимо решить задачи:

1. Обнаружить машиночитаемую зону на изображении при возможном развороте документа до 10 град. относительно рамки кадра. При этом фоном являются другие информационные поля документа, содержащие элементы графической и/или текстовой информации, сходной по структуре с машиночитаемой зоной, а также наносимый на бумагу защитный "узор".
2. Осуществить разбиение обнаруженной машиночитаемой зоны на знакоместа по числу символов OCR-B-кода.
3. Декодировать обнаруженный OCR-B- код и выдать информацию в ASCII формате.

В качестве тестовых задач была определена серия изображений паспортно-визовых документов. Она состояла из реальных изображений, полученных при помощи телевизионной камеры и фреймграбера AVER-2000. Размер тестовых изображений был выбран 688x512 пикселей.

Данная задача была разбита на два этапа. На первом этапе, включающем задачи 1 и 2, определяются координаты минимальных описанных прямоугольников для каждого символа OCR-B-кода в машиночитаемой зоне. На втором этапе формируется декодированная информация в виде двух текстовых строк в ASCII-коде.

Такое деление на этапы обусловлено следующими соображениями. Задачи 1 и 2 являются классическими задачами сегментации объектов на изображениях и могут быть решены различными методами обработки изображений. В то же время задача 3 является специфической задачей

проблемно-ориентированного распознавания печатных символов и тесно связана с конкретными особенностями данной кодировки и другими характеристиками анализируемого типа документов.

С учетом вышеизложенного была разработана следующая общая структура алгоритмического обеспечения задачи автоматического считывания информации, состоящая из ряда процедур анализа изображения, выполняемых последовательно:

1. Обнаружение текстовых строк.
2. Анализ изображений строк и определение знакомест.
3. Бинаризация изображений символов.
4. Распознавание символов OCR-B-кода.
5. Постпроверка результатов распознавания символов.

Следует отметить, что важной отличительной особенностью данной модульной схемы обработки является то, что определение знакомест производится по полутоновому изображению, и лишь затем происходит бинаризация изображения, причем параметры пороговой бинаризации определяются локально - отдельно для каждого знакоместа. В отличие от схемы, в которой изображение сначала бинаризуется, это позволяет наиболее полно использовать полутоновую информацию для борьбы с яркостными искажениями, имеющими место в процессе получения изображения документа при помощи телевизионной камеры. В следующих подразделах каждый из этих этапов будет описан более подробно.

## **2. Обнаружение текстовых строк.**

На этапе обнаружения кодосодержащих строк основным отличительным свойством изображения символа является то, что его граничные пикселы обладают высоким модулем градиента яркости. Процедура обнаружения кодосодержащих строк основывается на

обнаружении и анализе точек изображения, обладающих этим свойством.

Процедура обнаружения кодосодержащих строк содержит следующие основные этапы:

1. Определение положения машиночитаемой зоны.
2. Определение параметров положения строк.
3. Нормализация положения строки.

Рассмотрим каждый этап подробнее.

### **2.1. Определение положения машиночитаемой зоны.**

Цель данного этапа - определение минимальной области изображения, содержащего кодовые строки, и определение порога модуля градиента граничных точек символов. Это делается для повышения скорости работы алгоритма (последующая обработка использует процедуру голосования точек в пространстве Нough, обеспечивающую хорошую точность в определении параметров, но сложную в вычислительном плане), и повышения точности («голосуют» только точки, гарантированно принадлежащие символам, а не фону).

Для достижения этой цели делается следующее. В середине изображения берется вертикальная полоса шириной в две ожидаемых ширины символа (тогда в эту полосу обязательно попадут хотя бы по одному символу из обеих текстовых строк). В пределах этой полосы сканируется каждая строка изображения, и в соответствующем элементе гистограммы запоминается максимальное значение алгебраической суммы абсолютных значений компонент вектора градиента (эта сумма берется вместо значения модуля вектора градиента для увеличения скорости вычислений) в этой строке. Полученная гистограмма сглаживается путем усреднения по отрезку, длина которого равна высоте текстовой строки. Построенная таким образом гистограмма имеет ярко выраженные пики в элементах, соответствующих строкам изображения, в которых вертикальная

полоса имеет точки высокого контраста. Такие пики будут в местах расположения краев визы или паспорта и там, где расположены какие-либо строки символов. В частности, две искомые строки дадут два пика примерно одинаковой высоты и ширины (равной высоте строки), расположенных на известном расстоянии друг от друга (это расстояние известно из стандарта на текстовые строки), и имеющие справа и слева "плато" низких значений градиента известных длин.

По этим свойствам искомых пиков строится функция, подлежащая максимизации на данной гистограмме, причем для удобства она строится так, что ее максимум располагается посередине между искомыми пиками. Эта функция строится следующим образом. Для каждого элемента гистограммы рассматриваются четыре участка длиной в высоту строки каждый. Про первый предполагается, что он расположен на "плато" низких значений градиента выше текстовых строк; про второй - что он попадает на верхнюю строку; про третий - что он расположен на "плато" низких значений градиента между текстовых строк; про четвертый - что он попадает на нижнюю строку. Из этих участков образуются три разности: первая - между точками второго и первого участков; вторая - между точками второго и третьего участков; третья - между точками четвертого и третьего участков соответственно. При вычислении этих разностей для каждой из них подсчитывается число точек, в которых эта разность оказалась положительной. Если хотя бы для одной разности оказалось, что число таких точек меньше заданного порога, то это означает, что в рассматриваемой окрестности центральной точки либо нет максимумов гистограммы, либо что по крайней мере один из них уже, чем высота текстовой строки, и, следовательно, скорее всего мы не обнаружили искомые строки; следовательно, значение функции в такой точке будет равно нулю.

Если же число точек с положительными значениями больше порога, то вычисляется отношение меньшей из второй и третьей разностей к

большей из них, и если это отношение меньше другого заданного порога, то это значит, что один из найденных пиков значительно больше другого, и, следовательно, по крайней мере один пик вызван не строкой, а имеет другую природу, и функция в такой точке также будет равна нулю. Но если удовлетворены оба ограничения, то функция принимает значение, равное сумме всех трех разностей. В силу выбора участков для формирования разностей и использования характерных размеров машиносчитываемых строк, понятно, что один из максимумов (в идеале - единственный) этой функции попадет на элемент гистограммы, соответствующий середине промежутка между строками. Теперь предположим, что изображение ниже искомым строк других таких же пар строк не имеет. Тогда достаточно найти первый максимум функции, соответствующий нижней части картинке и дающий местоположение середины промежутка между строками в рассматриваемой вертикальной полосе. Теперь, зная длину строки и возможный угол поворота документа относительно камеры, легко вычисляются координаты углов прямоугольника на изображении, гарантированно содержащего обе кодовые строки. (По сути, изложенный метод есть разновидность согласованной фильтрации.)

Проведенные вычисления дают возможность оценить величину модуля градиента яркости (точнее, той величины, которая его заменяет) на точках фона и граничных точках символов кодовых строк в предположении, что при получении изображения освещенность вдоль строки меняется слабо. Как было указано, при вычислении разностей, второй и четвертый участки характеризуют среднее значение этой величины на граничных точках символов, а третий участок - на точках фона между строками. Поэтому, если к среднему значению этой величины на третьем участке прибавить некоторую часть меньшего из средних значений этой величины на втором и четвертом участке, и полученную величину принять за пороговую, то можно быть уверенным в том, что точки со значением модуля градиента яркости выше порога являются граничными точками

символов обеих кодовых строк, и почти точно не являются точками фона. Таким образом, мы имеем возможность выставить адаптивный порог, важный для следующего этапа обнаружения кодовых строк.

Результаты тестирования показали, что приведенный выше алгоритм определения положения объемлющего прямоугольника устойчиво работает даже в случае, когда виза или паспорт лежат на газете или другой печатной странице (размер символов и расстояние между строками текста имеют другие значения).

## **2.2. Определение параметров положения.**

Цель этого этапа - определение точного местоположения кодосодержащих строк на изображении путем нахождения линии, проходящей по середине ее вдоль каждой строки.

Согласно методике Hough ([1]-[4]) вводится параметризация этих линий путем перехода в полярную систему координат, у которой полюс находится в левом верхнем углу изображения, нулевой угол соответствует верхнему краю изображения, увеличение угла происходит по часовой стрелке, под расстоянием  $\rho$  понимается длина перпендикуляра, опущенного из полюса на линию. В пространстве этих параметров выбирается область, обусловленная по  $\rho$  размерами изображения, а по  $\varphi$  - допустимым углом поворота паспорта или визы относительно камеры. Затем производится дискретизация выбранной области путем выбора разумных значений дискретов по обеим осям с тем, чтобы обеспечить требуемую точность определения соответствующего параметра при разумных затратах времени на вычисления. Полученное пространство параметров  $\rho$ - $\varphi$  есть пространство Hough, или аккумулятор. Перед началом работы ячейки этого аккумулятора обнуляются.

Для повышения скорости вычислений в соответствии с выбранными дискретом и размером области аккумулятора, по  $\varphi$  табулируются значения

функций  $\cos(x)$  и  $\sin(x)$ . Далее производится процедура голосования точек внутри ранее найденного прямоугольника, содержащего искомые строки, со значениями модуля градиента яркости выше установленного ранее порога.

Для уменьшения числа операций (и времени счета) без потери точности результатов опрашиваются не все точки внутри прямоугольника, а только крайние правая и левая четверти ширины этого прямоугольника.

Из общих соображений ясно, что символы двух искомых строк дадут два высоких и широких максимума в аккумуляторе, центры которых соответствуют средним линиям этих строк. Поскольку эти максимумы недостаточно узки, то нельзя сразу воспользоваться значениями глобальных максимумов для определения параметров средних линий строк, а необходимо построить процедуру нахождения этих параметров. Построение такой процедуры производится, исходя из соображений и свойств кодосодержащих строк, использованных на предыдущем этапе алгоритма обнаружения строк: строится та же самая функция для максимизации, как и на первом этапе, с той только разницей, что отсутствуют условные ограничения на ширину пиков (не ведется подсчет точек, в которых разности отрицательны) и различие их по высоте. Поскольку искомые строки параллельны, то понятно, что эта функция есть функция одной переменной  $\rho$  при заданном значении параметра  $\varphi$ . Эта функция применяется последовательно ко всем строкам аккумулятора (т.е. элементам, имеющим одно и то же значение параметра  $\varphi$ ). Точка глобального (т.е. по всем строкам) максимума этой функции принимается за параметры линии, идущей параллельно искомым строкам посередине между ними, и таким образом мы точно находим значение параметра  $\varphi$  для средних линий строк. Значения параметра  $\rho$  для каждой из них находим, исходя из найденного экстремального значения и известного расстояния между строками. Зная высоту строки, определяем две полосы, содержащие в себе каждую из искомых строк.

### **2.3. Нормализация положения строки.**

Для удобства дальнейшей работы со строками и символами в них нужно устранить поворот этих строк относительно горизонтали и переписать их изображения в два буфера для ускорения доступа к ним и уменьшения объема требуемой памяти (исходное изображение теперь больше не нужно, и его можно выгрузить из памяти). Это и делается на третьем этапе алгоритма. Поскольку после выполнения предыдущего этапа параметры преобразования системы координат исходного изображения в систему координат, в которой строки горизонтальны, известны, то для каждой точки буфера можно единственным образом определить соответствующую точку изображения строки.

Для практической реализации был выбран алгоритм обратного пересчета, при котором по координатам точек результирующего буфера определяются соответствующие координаты точек исходного изображения, т.е. цикл пересчета организован как цикл построчного сканирования буфера. Это позволяет избежать появления незаполненных элементов буфера, которые в противном случае могут возникать из-за ошибок округления при прямом пересчете.

После завершения этапа обнаружения текстовых строк предполагается, что в системе имеется два горизонтальных изображения текстовых строк, каждое из которых на этапе выделения знако-мест анализируется отдельно.

### **3. Разбиение строк и определение знакомест.**

Разработанная процедура определения знакомест по цифровому

изображению одной горизонтальной строки символов включает следующие основные этапы, выполняемые последовательно:

1. Сбор латеральных диаграмм.
2. Анализ латеральных диаграмм.
3. Формирование первичного списка объемлющих прямоугольников.

Под латеральными диаграммами цифрового изображения [5] понимаются дискретные массивы, длина которых равна горизонтальному размеру изображения, а в каждой ячейке диаграммы содержится некоторое интегральное значение, которое тем или иным образом характеризует соответствующий столбец этого изображения. На этапах 1-2 анализа изображения строки используются две различные латеральные диаграммы - латеральная гистограмма яркости и латеральная диаграмма размаха яркости. Каждая ячейка латеральной гистограммы яркости содержит суммарную яркость пикселей соответствующего столбца изображения. Каждая ячейка латеральной диаграммы размаха яркости содержит значение разности между максимальной и минимальной яркостью пикселей соответствующего столбца изображения.

Исследование яркостных особенностей тестовых изображений позволило определить следующие связи между содержанием изображения символьной строки и формой указанных латеральных диаграмм:

- Локальным минимумам латеральной гистограммы яркости соответствует положение вертикальных штрихов в начертании текстовых символов.
- Локальным максимумам латеральной гистограммы яркости соответствуют промежутки между текстовыми символами (первичные максимумы), а также промежутки между двумя ярко выраженными вертикальными штрихами в середине символа (вторичные максимумы); вторичные максимумы имеют несколько меньшие абсолютные значения.
- Локальным минимумам латеральной диаграммы размаха также

соответствуют промежутки между текстовыми символами.

Для того, чтобы обнаружить промежутки, разделяющий два соседних символа текста, достаточно найти такой локальный максимум латеральной гистограммы яркости, которому соответствует локальный минимум диаграммы размаха яркости.

Таким образом, анализ латеральных диаграмм предполагает выполнение следующих этапов обработки изображения:

1. Поиск значимых локальных экстремумов латеральных диаграмм.
2. Удаление вторичных максимумов латеральной гистограммы яркости.
3. Слияние пар близких максимумов.
4. Определение локальных максимумов латеральной гистограммы яркости, имеющих соответственные минимумы в диаграмме размаха яркости.

Значимыми здесь считаются локальные экстремумы, характеризующиеся разницей с обоими соседними локальными экстремумами противоположного знака не менее чем на 10% амплитуды диаграммы. Удаление вторичных максимумов осуществляется на основе простейшего признака - "значимый локальный максимум, который по крайней мере на 5% амплитуды ниже, чем оба соседних к нему значимых локальных максимума". Пары близких максимумов латеральной гистограммы яркости, соответствующие двум краям промежутка между соседними символами, определяются по признаку "пара соседних значимых максимумов, между которыми нет значимого минимума". Определение соответственных минимумов в диаграмме размаха яркости производилось для каждого локального максимума латеральной гистограммы яркости в некоторой зоне поиска, размер которой определяется как 20% от предполагаемой ширины символа.

Выполняемая далее процедура формирования первичного списка объемлющих прямоугольников опирается на полученные списки

максимумов латеральной гистограммы и минимумов диаграммы яркостного размаха, и заключается в следующем:

1. Оценка горизонтального яркостного тренда изображения.
2. Восстановление недостающих промежуточных максимумов.
3. Восстановление недостающих краевых максимумов.
4. Формирование списка прямоугольников.

Под "горизонтальным яркостным трендом" здесь понимается оценка "среднего наклона" функции локальной средней яркости изображения в горизонтальном направлении. Восстановление недостающих максимумов предполагает: сбор гистограммы расстояний между найденными максимумами, оценку ожидаемой длины символа по этой гистограмме, определение ожидаемых координат положения недостающих разделителей символов и уточнение этих значений путем поиска локальных максимумов гистограммы яркости в малой окрестности их ожидаемого положения.

Полученные оценки яркостного тренда используются при определении недостающих знакомест на концах символьной строки. При формировании первичного списка прямоугольников в качестве левых и правых границ знакомест используются последовательно все соседние пары точек из финального списка локальных максимумов, а в качестве верхней и нижней границы знакоместа - соответственно верхняя и нижняя границы анализируемого изображения текстовой строки. В результате выполнения данного этапа обработки изображение текстовой строки оказывается разбитым на знакоместа, причем между этими знакоместами нет промежутков.

#### **4. Бинаризация изображений символов и формирование списка минимальных объемлющих прямоугольников.**

Следующий этап анализа изображения строки предполагает

выполнение трех основных процедур:

1. Бинаризация каждого знакоместа.
2. Логическая фильтрация бинарного изображения.
3. Формирование списка минимальных прямоугольников, объемлющих символы.

Под бинаризацией здесь понимается разбиение изображения на две области - "объект" и "фон" - на основе анализа яркостной гистограммы изображения. Порогом  $T$  при этом называется такой уровень яркости, что "фон" содержит все пиксели со значением яркости пикселей больше  $T$ , а "объект" - все остальные пиксели изображения.

Однако эксперименты на тестовых изображениях показали наличие ярко выраженного неравномерного горизонтального тренда, то есть средняя яркость столбцов существенно неодинакова по ширине изображения. Поэтому использование глобальной пороговой сегментации всего изображения одним порогом (как это часто делается в стандартных алгоритмах автоматического распознавания символов) в данной задаче невозможно. В данной системе алгоритм бинаризации используется не глобально, а локально - для каждого знакоместа определяется свой порог сегментации.

Исследования показали, что при правильном выборе порога сегментация обеспечивает удовлетворительное разбиение знакоместа, такое, что большинство пикселей текстового символа принадлежат "объекту". При этом, однако, возникает и значительное число ложных пикселей "объекта", чем и вызвана необходимость применения к результатам бинаризации дополнительной логической пространственной фильтрации.

Для определения оптимального порога бинаризации использовался метод [6], который предполагает не только определение оптимального порога бинаризации, но и вычисление некоторого критерия бимодальности, т.е. оценку того, действительно ли исследуемая гистограмма содержит именно две моды (два выраженных пика). Идея данного подхода

заключается в следующем.

Пусть изображение знакоместа представляет собой двумерный массив размером  $X \times Y$ , причем его элементы (пиксели) принимают значения на отрезке  $[0..255]$ . Тогда яркостная гистограмма этого изображения представляет собой одномерный массив  $Hist [0..255]$ , в каждой ячейке которого  $Hist[i]$  содержится число пикселей изображения, имеющих значение, равное  $i$ . Рассмотрим теперь "подгистограмму"  $Hist1[k..l]$ ,  $0 \leq k, l \leq 255$ .

Для любой такой подгистограммы (окна)  $[k, l]$  можно получить оценку математического ожидания  $M_0(k, l)$  и дисперсии  $DISP(k, l)$  яркости пикселей.

Пусть дан порог  $t$ :  $0 \leq t \leq 255$ . Тогда для данного порога  $t$  можно вычислить "критерий разделимости"  $SC$  вида

$$SC(t) = 1 - (DISP(0, t) + DISP(t+1, 255)) / DISP(0, 255).$$

Критерий  $SC(t)$  всегда принимает значение на отрезке  $[0, 1]$ , причем значение его тем больше, чем лучше разделимость яркостного распределения на два класса относительно порога  $t$ . Рассматриваемый алгоритм предполагает вычисление  $SC(t)$  для всех  $t \in [0..255]$ , после чего оптимальный порог определяется как:

$$T = \arg \max SC(t), \quad t \in [0..255].$$

Этот алгоритм имеет ясный статистический смысл и, как показывает практика, является эффективным и устойчивым способом определения адаптивного порога для бинаризации бимодальных изображений.

Как уже отмечалось, после бинаризации на изображении символов появляются ошибки двух типов: небольшие выступы и впадины по длине линий, маленькие отростки и "хвостики" (т.н. "бахрома"), а также пустоты, т.е. группы незачерненных элементов внутри линий. Кроме того, на белом "фоне" возможно появление изолированных черных зон, не имеющих отношения к искомому символу.

Наконец, на внешних линиях символов возможно образование

"наростов" значительного размера, что приводит к искажению габаритных размеров символа или даже к слиянию соседних символов. Для исправления этих ошибок в данной системе используется логическая фильтрация трех видов:

- согласованная фильтрация с окаймлением для удаления изолированных черных зон;
- стирание бахромы - заполнение пустот;
- оригинальный алгоритм фильтрации рамки для стирания крупноразмерных ложных наростов.

После логической фильтрации осуществляется окончательное формирование списка знакомест данной строки как списка минимальных прямоугольников, объемлющих все оставшиеся "черные" пиксели данного знакоместа.

После того как данный этап завершен (применительно к изображениям обеих выделенных текстовых строк), считается, что мы имеем для каждой строки 44 бинарных изображения символов OCR-B-кода, подлежащих дальнейшей обработке.

## **5. Распознавание символов OCR-B.**

В системе реализовано два основных метода распознавания символов - метод зон и метод пересечений.

### **5.1. Метод зон.**

Как указывалось ранее, метод зон основан на сравнении *зонного представления* изображения символа с эталонным набором зонных представлений символов OCR-B-кода. Поэтому алгоритм распознавания состоит из трех основных процедур:

- *обучение* (выполняется один раз при разработке программного обеспечения);
- *загрузка набора эталонных зонных описаний* (выполняется один раз при каждой загрузке программы);
- *собственно распознавание* (выполняется для каждого распознаваемого символа);

На этапе обучения использовался полный эталонный набор бинарных изображений символов OCR-B-кода, имеющих размеры 100x135 пикселей. При этом для каждого эталонного изображения в наборе выполнялись следующие действия:

1. определение минимального прямоугольника, содержащего все "черные" пиксели изображения ("рамки");
2. равномерное разбиение рамки на  $N \times M$  прямоугольных зон;
3. подсчет числа пикселей, принадлежащих каждой зоне;
4. формирование вектора зонного описания, состоящего из значений числа "черных" пикселей для каждой зоны, нормированных путем деления на суммарное число "черных" пикселей всего изображения.

Подсчет числа пикселей, принадлежащих каждой зоне, осуществляется путем голосования "черных" пикселей. При этом каждый пиксел рассматривается как квадрат размером 1x1, положение границ зон рассчитывается с субпиксельной точностью (float), и каждый "черный" пиксел голосует в пользу тех зон, с которыми он пересекается, с весом, равным площади пересечения.

В результате этапа обучения формируется эталонный файл зонных описаний, представляющий собой текстовый файл, содержащий 37 текстовых строк, в каждой из которых (через пробел) записано имя символа в ASCII-коде и вектор зонного описания размера  $N \times M$  с элементами типа float. Вектор зонного описания записывается по строкам, слева направо сверху вниз.

При загрузке эталонных данных происходит считывание эталонного файла и формирование соответствующего списка эталонных векторов зонных описаний.

При анализе каждого символа производятся следующие операции:

1. формирование вектора зонного описания;
2. формирование вектора расстояний;
3. классификация символа на основе вектора расстояний.

Формирование вектора зонного описания осуществляется как было описано ранее для эталонных изображений.

Формирование вектора расстояний предполагает последовательное вычисление евклидовых расстояний между вектором зонного описания тестируемого символа и векторами зонных описаний каждого эталона в списке.

Классификация символа осуществляется путем анализа вектора расстояний и выбора "ближайшего соседа" среди эталонных зонных описаний. При этом принимается решение о том, что данный символ распознан, и ему присваивается ASCII-код, соответствующий ASCII-коду найденного "ближайшего соседа".

Степень доверия к результату распознавания методом зон тем выше, чем меньше значение расстояния до ближайшего соседа  $R_1$  по сравнению с расстоянием до следующего по близости соседа  $R_2$  ( $R_2 \geq R_1$ ). Поэтому его можно охарактеризовать соответствующим *коэффициентом доверия*

$$K = (R_2 - R_1) / R_2.$$

Определенный таким образом коэффициент доверия обладает следующими удобными свойствами:

- $1 \leq K \leq 0$ ;
- если  $R_1 = 0$ , то  $K = 1$ ;
- если  $R_1 = R_2$ , то  $K = 0$ .

То есть увеличение значения  $K$  соответствует росту доверия к результату классификации,  $K = 0$  означает полную неуверенность при выборе

между ближайшими соседями, а  $K=1$  - полную уверенность вследствие точного совпадения с зонным эталоном. Кроме того, степень доверия к результату классификации тем выше, чем меньше абсолютное значение расстояния до ближайшего соседа.

Проведенные исследования показали, что с учетом структуры штрихов начертания символов OCR-B-кода, для размера изображений символов порядка  $10 \times 15$  пикселей оптимальным числом зон будет  $N \times M$ , где  $N=5$  и  $M=5$ . При таком выборе зонного разрешения использованный способ нормирования вектора зонного описания позволил обеспечить устойчивость данного метода распознавания к изменению толщины штрихов символа вследствие нестабильности яркостных характеристик изображения.

На рис.1 показано эталонное изображение символа R, разбиение на зоны, пиксельные веса каждой зоны, а также вектор расстояний до эталонных векторов.

На рис.2 показано реальное изображение символа R, полученное для одного из изображений тестового набора, разбиение его на зоны, пиксельные веса каждой зоны, а также вектор расстояний до эталонных векторов.

## **5.2. Метод пересечений.**

Общая структура процедуры анализа изображения символа методом пересечений предполагает выполнение следующих основных операций:

1. определение минимального объемлющего прямоугольника;
2. сечение этого прямоугольника некоторым набором прямых;
3. подсчет характеристик пересечения символа с каждой секущей линией;
4. сравнение полученных наборов характеристик пересечений с

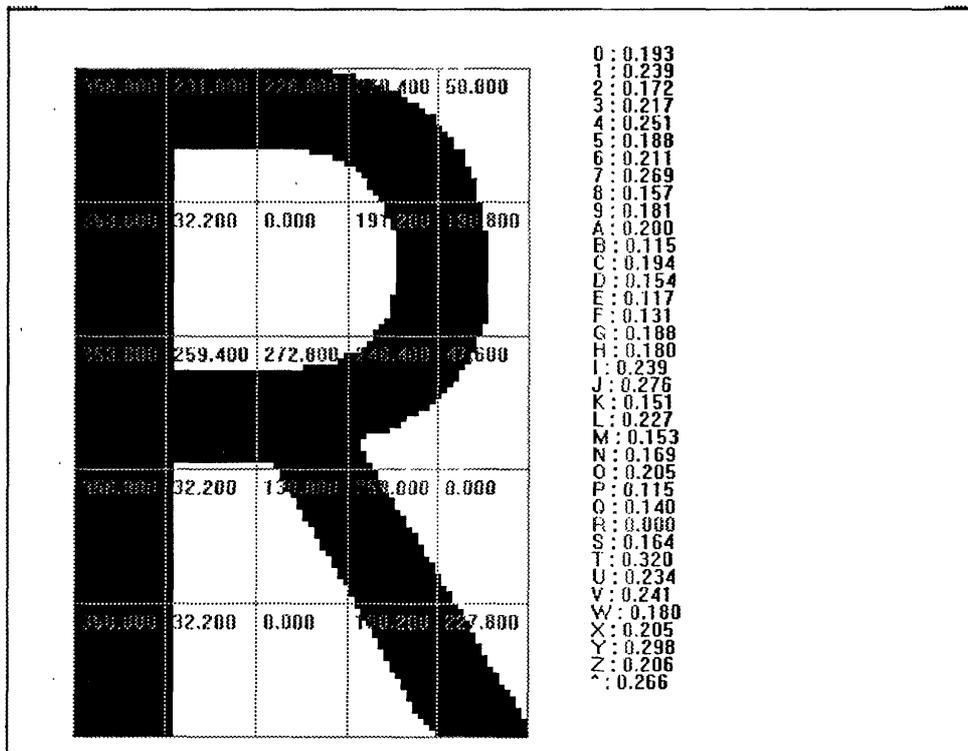


Рис.1. Эталонное изображение символа R.

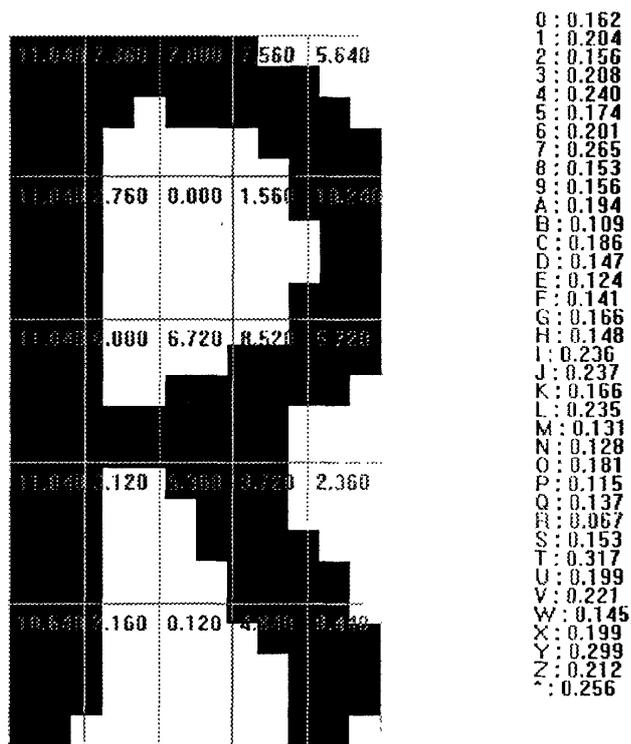


Рис.2. Реальное изображение символа R; CF==0.387.

аналогичными наборами характеристик для некоторого заранее заданного множества эталонов.

Следует отметить, что известные в настоящее время модификации данного метода существенно различаются в реализации следующих принципиальных моментов:

- выбор системы текущих линий;
- выбор типа характеристик сечений;
- выбор методов и критериев принятия решения о классификации символа.

В результате анализа проведенных экспериментов с несколькими вариантами данного метода вариант, описание которого приводится ниже, был признан наиболее перспективным и практически реализован.

Считается, что после сегментации строки на знакоместа и их бинаризации минимальный объемлющий прямоугольник (MBR) уже известен.

Внутренность этого прямоугольника сечется заранее выбранным набором прямых, одинаковым для всех распознаваемых символов. В этом наборе могут присутствовать линии, проведенные под любым углом к горизонтали.

Для каждой линии фиксируется общая длина отрезка внутри MBR, а также длины последовательных отрезков этой секущей с указанием атрибута: лежит этот отрезок на фоне или части символа (таким атрибутом выбран знак числа). При этом не делается никаких предположений о толщине линий символа. Каждый символ заданного алфавита представляется набором, состоящим из одинакового числа последовательностей целых чисел, меньших или равных 100 по абсолютной величине. Такой выбор диапазона значений позволяет выделять на машинах традиционной архитектуры по 1 байту на отрезок, что позволяет экономить память, не теряя важной информации о топологии символа.

Затем производится нормализация длин отрезков путем деления длины

каждого отрезка на общую длину отрезка секущей внутри MBR, причем для экономии памяти и времени, в последующих вычислениях результат домножается на 100%, округляется и запоминается получившаяся последовательность целых чисел, сумма абсолютных значений которых равна 100.

Важно, что предложенный способ описания символов обеспечивает инвариантность к изменению масштаба и позволяет сравнивать символы разных размеров.

Так же, как и в методе зон, процесс распознавания состоит из трех основных процедур:

- *обучение* (выполняется один раз при разработке программного обеспечения);
- *загрузка набора эталонных описаний* (выполняется один раз при каждой загрузке программы);
- *собственно распознавание* (выполняется для каждого распознаваемого символа).

До стадии распознавания символ претерпевает ряд преобразований, и каждое из них по-своему искажает символ; нельзя рассчитывать на то, что символ и эталон этого символа, даже будучи приведенными к одному размеру, будут в точности совпадать. Компенсировать этот фактор путем учета модели вносимых искажений не представляется возможным, поскольку, во-первых, искажения имеют различную природу и, во-вторых, как правило, учет искажений приводит к алгоритмам ограниченного перебора, что сопряжено с существенными затратами вычислительных ресурсов. Поэтому требовалось найти критерий, который основывался бы не просто на сходстве полного набора характеристик всех сечений, а на выделении и сравнении только наиболее характерных особенностей. За счет этого достигается меньшая чувствительность алгоритма к имеющим место искажениям. В качестве таких характерных особенностей символа в работе выбраны превышающие некоторый заданный порог отрезки секущих,

лежащие целиком на фоне или символе.

При определении степени схожести символа и эталона производится сравнение последовательных отрезков соответствующих секущих распознаваемого символа и эталона.

Предложенная в работе модификация метода пересечений характеризуется следующими основными особенностями:

1. Пороговые значения величин определяются на стадии проектирования и тестирования программы.
2. Метод обладает устойчивостью к шумам типа "соль-перец" при не очень большом зашумлении.
3. Метод допускает наличие небольшого поворота символа внутри MBR.
4. Метод инвариантен к преобразованию масштаба.

На рис. 3 показано эталонное изображение символа R, система секущих прямых, а также вектор расстояний до эталонных векторов.

На рис. 4 показано реальное изображение символа R, полученное для одного из изображений тестового набора, система секущих прямых, а также вектор расстояний до эталонных векторов.

### **5.3. Постпроверка результатов распознавания.**

Проведенное тестирование показало, что при использовании обоих реализованных методов распознавания символов могут возникать ошибки, связанные с тем, что из-за случайных искажений формы символов, имеющих место в процессе получения цифровых бинарных изображений, иногда происходит перепутывание символов, близко расположенных в пространстве используемых признаков.

Простейший прием контроля результата классификации состоит в том, чтобы произвести независимую классификацию того же символа другим способом, а затем сравнить результаты и выбрать тот, который обеспечил более высокий показатель достоверности. При этом финальная

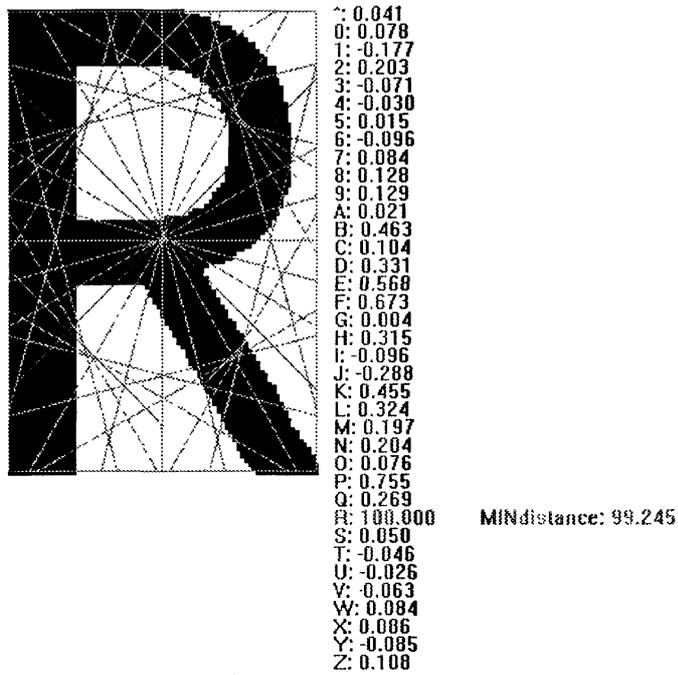


Рис.3. Эталонное изображение символа R.

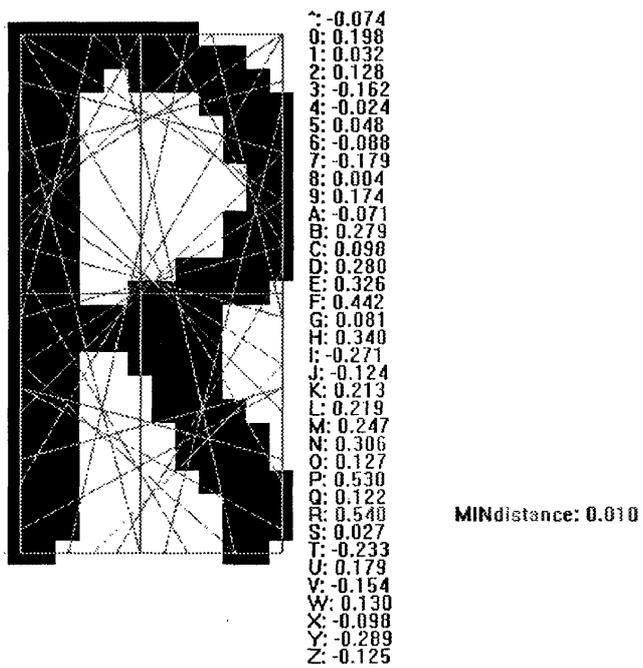


Рис.4. Реальное изображение символа R.

достоверность классификации определяется либо по полученной максимальной оценке, либо по методу Демпстера-Шафера. Такой способ повторного анализа изображения называется *кросс-проверкой* результатов классификации.

Метод зон и метод пересечений образуют удачную пару для кросс-проверки, так как они опираются на совершенно различные принципы анализа изображений символов.

Альтернативным методом постпроверки классификации символа является т.н. *логический анализ различий*. Идея этого метода заключается в следующем. Для используемого базового метода распознавания в пространстве признаков (на этапе обучения) производится кластеризация, то есть выделяются группы "близких", а значит, плохо различаемых символов. Для каждой такой группы находится минимальный набор признаков постпроверки, позволяющий уверенно различать между собой все символы внутри данной группы. Тогда на этапе распознавания необходимо:

1. первично классифицировать символ с использованием базового метода;
2. определить, к какой группе "близких" символов относится результат классификации;
3. вычислить значения признаков постпроверки для данной группы символов;
4. проанализировать набор признаков постпроверки и окончательно классифицировать данный символ.

Следует, однако, отметить, что такой подход является эффективным лишь в том случае, когда в пространстве признаков данного метода действительно наблюдаются несколько компактных кластеров, соответствующих группам из нескольких символов, близких между собой и достаточно удаленных от других. Проведенное исследование показало, что ни для метода зон, ни для метода пересечений это не так. В результате кластерного анализа по методу дальнего соседа [5] было установлено, что

структура эталонного множества содержит размытые пересекающиеся кластеры из 7-12 элементов, поэтому данный метод постпроверки классификации символов не использовался в данной работе.

Методы кросс-проверки и логического анализа различий объединяет то, что это методы посимвольного распознавания текста. То есть в этих методах каждое знакоместо анализируется отдельно, независимо от того, какие результаты распознавания получены для других знакомест. Между тем, если анализируемый набор символов представляет собой связный текст, а не случайную последовательность букв, то на возможные результаты распознавания символов всегда накладываются некоторые дополнительные условия, вытекающие из характера этого текста.

По сравнению с методами постпроверки, рассмотренными выше, *учет контекстной информации* имеет то важное преимущество, что не основывается на анализе изображения символа (т.е. той информации, которая уже была использована на этапе первичной классификации символа), а привлекает дополнительную, "внешнюю" по отношению к изображению, информацию. При этом используемая контекстная информация может быть глобальной или локальной, а также логической или вероятностной.

Глобальная контекстная информация предполагает некоторое априорное знание о том, какие значения может принимать символьная переменная в зависимости от номера знакоместа, на котором она находится. Простейшим примером глобальной контекстной информации является утверждение "в десятой позиции данной строки находится цифра". Это правило сразу "сужает" набор классов для распознавания в указанной позиции.

Локальная контекстная информация предполагает наличие априорных сведений о том, какие значения может принимать символьная переменная в данной позиции в зависимости от того, какие значения принимают символьные переменные в некотором заданном наборе других позиций (знакомест). Примером локальной информации о контексте является,

например, правило "последовательность символов, начинающаяся с буквы, содержащая только буквы". С учетом этого правила значение символьной переменной в первой позиции слова непосредственно влияет на возможные наборы значений символьных переменных в остальных позициях.

Логическая контекстная информация предполагает строгое задание правил, определяющих, какие символы или сочетания символов являются в некотором конкретном случае допустимыми, а какие - нет. При этом логическая контекстная информация может быть как глобальной, так и локальной.

Вероятностная контекстная информация предполагает некоторые знания о том, какие символы или сочетания символов являются в некотором конкретном случае более вероятными, а какие - менее вероятными. Вероятностная контекстная информация также может быть и глобальной, и локальной.

Чем сильнее контекстные ограничения, накладываемые на анализируемый текст, тем больший эффект может дать использование контекстной информации при постпроверке результатов первичной классификации.

В задаче автоматического считывания машиночитаемых строк OCR-В-кода с машиночитаемых документов определяемая стандартом записи структура текстовых строк позволяет задать существенный набор логических контекстных правил (как локальных, так и глобальных). Наиболее важная контекстная информация связана с разбиением символов на последовательные группы по номерам знакомест, а также с наличием контрольных цифр.

### **Заключение.**

Для решения задачи автоматического считывания машиночитаемых документов разработана модульная схема обработки изображения, предполагающая последовательное выполнение следующих шагов: обнаружение текстовых строк, анализ изображений строк и определение знакомест, бинаризация изображений символов, распознавание символов OCR-B-кода, постпроверка результатов распознавания символов. Отличительной особенностью данной модульной схемы обработки является то, что определение знакомест производится по полутоновому изображению. Это позволяет наиболее полно использовать полутоновую информацию для борьбы с яркостными искажениями, имеющими место в процессе получения изображения документа при помощи телевизионной камеры, и уменьшить тем самым величины ошибок чтения.

Разработанная оригинальная процедура обнаружения кодосодержащих строк включает три основных этапа:

- определение положения машиночитаемой зоны,
- определение параметров положения строк,
- нормализацию положения строки.

Разработанная оригинальная процедура определения знакомест по цифровому изображению одной горизонтальной строки символов включает следующие основные этапы, выполняемые последовательно: сбор латеральных диаграмм, анализ латеральных диаграмм, формирование первичного списка объемлющих прямоугольников. Используются латеральные диаграммы двух типов: латеральная гистограмма яркости и латеральная диаграмма размаха вариации яркости. Определение границ символов осуществляется путем анализа и сравнения локальных экстремумов латеральных диаграмм.

Этап бинаризации изображений символов и формирования списка минимальных объемлющих прямоугольников заключается в

последовательном выполнении трех основных процедур: отдельной бинаризации каждого знакоместа, логической фильтрации бинарного изображения и формировании окончательного списка минимальных объемлющих символы прямоугольников.

Предложены оригинальные реализации двух основных методов распознавания символов - метода зон и метода пересечений. Оба этих метода обеспечивают для тестовых изображений машиночитаемых документов качество распознавания порядка 90-95%. Кроме того, метод зон и метод пересечений образуют удачную пару методов для кросс-проверки результатов распознавания, так как они опираются на различные принципы анализа изображений символов и потому могут корректировать ошибки друг друга.

На основе анализа структуры записи данных в машиночитаемой зоне определен значительный набор глобальных и локальных контекстных правил постпроверки результатов распознавания символов.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Duda, R.O., and Hart, P.E. "Use of the Hough Transformation to Detect Lines and Curves in Pictures." *Comm. ACM*, vol. 15, no. 1, pp. 11-15, 1972.
2. Elliott, D.F., and Rao, K.R. *Fast Transforms: Algorithms and Applications*, Academic Press, New York, 1983.
3. Makoto Nagao, Shigeyoshi Nakajima. "On the relation between the Hough transformation and the projection curves of a rectangular window", *Pattern Recognition Letters* 6(1987), 185-188.
4. Визильтер Ю.В., Степанов А.А., Желтов С.Ю. "Новые методы обработки изображений", НЗНТ, Серия: Авиационные системы, №4; М: ГосНИИ АС, 1992.
5. Ramesh Jain, Rangachar Kasturi, Briane G. Schunck " *Machine vision* " , McGraw-Hill, Inc, 1995г.
6. Otsu N. A theshold selection method from gray-level histograms //IEEE Trans. on System, Man and Cybernetics - 1979, v.SMC-9, N 1.