Об искажении символов при сканировании

Ю. В. Титов

В статье обсуждаются вопросы различия отсканированных символов кириллицы и латиницы от их идеальных прообразов. Оцениваются пределы вариации границ символов, соответствующих одному прообразу. Описана модель искажения образа при сканировании. Приведены результаты экспериментов для нескольких сканеров.

Введение

Системы оптического распознавания текстов (OCR), преобразующие графический образ документа в текстовый формат [1], используют в своей работе различные методы и алгоритмы, такие как бинаризация, сегментация (поиск текстовых блоков, таблиц, иллюстраций и иных объектов), распознавание текстовых строк и полей таблиц, адаптацию к особенностям шрифтов документа, лингвистические средства. Многие из них базируются на алгоритмах распознавания образов отдельных символов, которые оперируют либо представлением образа символа в виде набора признаков, либо оригинальным отсканированным образом. Предельные характеристики качества распознавания, распознающих набор признаков, определяются не только свойствами собственно алгоритма, но также искажениями сигнала и возможностями представления образа символа. Масштабирование образов к одному размеру приводит к невозможности различения одинаковых по написанию прописных и строчных букв, таких как **WwUuOo**. Алгоритмы распознавания символов, использующие оригинальные образы, лишены этого недостатка.

Однако возникает вопрос о границах возможностей распознавания символов, использующих оригинальные образы. Нам интересно, насколько могут отличаться образы отсканированных символов от их идеальных прототипов, а также, насколько могут различаться экземпляры одной буквы.

В данной работе мы будем придерживаться ниже описанных терминов и ограничений.

1. Определения

Опишем некоторые детали процесса получения изображения текста — *сканирования*.

Как правило, основным инструментом оцифровки печатного текста с бумажных носителей является *сканер*. *Сканер* — это устройство для ввода в компьютер графических изображений с плоских объектов: бумажные страницы и пр. [2].

Основными компонентами современного планшетного CCD-сканера являются лампа подсветки, система отклоняющих зеркал, двигатель и светочувствительный сенсор с системой фокусирующих линз. Свет от лампы подсветки попадает на сканируемый объект и отражается от него. Система зеркал направляет свет в систему линз. Пройдя через систему линз, свет попадает на CCD-сенсор (светочувствительный элемент).

Назовем некоторые технические характеристики сканера в наибольшей степени влияющие на результат сканирования серых изображений.

Яркость — фактически это яркость лампы подсветки сканера. Чем больше яркость, тем более светлым получается изображение, тоньше черные линии, светлее фон. При пониженной яркости фон текста будет более серым, буквы будут более жирными. Изменять яркость уже отсканированного изображения можно программным путем.

Контрастность — это параметр, регулирующий различие яркости темных и светлых участков изображения. При увеличении *контрастности,* темные участки изображения становятся более темными, а светлые — более светлыми. При уменьшении — наоборот — различие между темными и светлыми участками сглаживается.

В данной статье мы не будем рассматривать цветные изображения. Объектом наших исследований являются только серые изображения, и, как частный случай, черно-белые, поскольку большинство современных программ распознавания текста используют для распознавания исключительно черно-белые или серые изображения.

Понятие *порога* возникает при черно-белом сканировании, равно как и при приведении серого изображения к черно-белому. Этот параметр задает уровень снятия фона в сером изображении. Всем участкам изображения, насыщенность которых черным меньше принятого порога, присваивается значение, соответствующее белому, а участкам темнее порога — соответствующее черному.

Сетка сканера — это виртуальная прямоугольная сетка, на которую попадает сканируемое изображение. Количество ячеек в горизонтальной строке этой сетки равно количеству ячеек CCD-сенсора сканера участвующих в сканировании. На самом деле устройство цветного (другие почти не

встречаются в настоящее время) сканера несколько сложнее, но для наших целей вполне достаточно данного грубого приближения. Количество же строк сетки равно количеству отсканированных полосок, т. е. количеству смещений шагового двигателя. Поскольку мы предполагаем, что сканируемый объект целиком помещается в области сканирования, то можно считать, что сетка сканера бесконечная — не ограничена какими-либо рамками. Сетка сканера — это нечто вроде клетчатой бумаги, на которой мы закрашиваем клетки — каждую только в 1 оттенок. Описанные ячейки сетки сканера соответствуют пикселями получаемого изображения.

Пиксель — неделимая ячейка в графическом изображении; наименьший адресуемый элемент растрового изображения. *Пиксель* характеризуется прямоугольной (в нашем случае квадратной) формой и размерами, определяющими пространственное разрешение изображения. Один пиксель полученного при сканировании изображения — это ячейка сетки сканера, с присвоенным значением оттенка серого. В ч/б изображениях пиксель имеет значение 0 или 1, в серых — от 0 до 255; последнюю величину будем называть *полутоном*.

Интенсивность — это величина равная $255 - p$, где p — это значение полутона данного пикселя. Обычно интенсивность применяется для сравнения нескольких участков с большим значением полутонов. Говорят, что более черный участок имеет большую интенсивность.

Бинаризацией называют преобразование серого изображения к черно-белому, т. е. к бинарному.

Поскольку мы исследуем символы и нам интересно «что закрасилось черным», то будем измерять не «светлоту», а наоборот — «черноту». То есть во всех численных данных уровень серого равный 0 будет соответствовать абсолютно белому цвету, а уровень 255 — абсолютно черному. Данная шкала является противоположной по отношению к шкале принятой в графических форматах данных (обычно черному цвету соответствует 0, а белому 255).

2. Постановка задачи

Проиллюстрируем, что происходит с символом при сканировании. Жизненный цикл черно-белого образа буквы при сканировании состоит из следующих этапов:

1) непрерывный черно-белый оригинал (на бумажном носителе);
2) непрерывный серый оригинал — виртуальный образ (после отражения света);
3) дискретный серый отсканированный образ (после укладки на сетку сканера);
4) дискретный черно-белый образ (после бинаризации).

Четвертый пункт отсутствует в случае, если мы сканируем серое изображение. В первом пункте непрерывность и бинарность изображения на бумаге — условные, так как на самом деле типографская краска наносится неровно, а бумага не идеально белая.

Из идеальной буквы А, которая была напечатана на бумаге, получается «нечто» ворсистое, состоящее из квадратиков (рис. 1). Ступенчатая структура возникает из-за наложения символа на сетку сканера, а всевозможные пупырышки — из-за аппаратных погрешностей и некоторых других факторов:

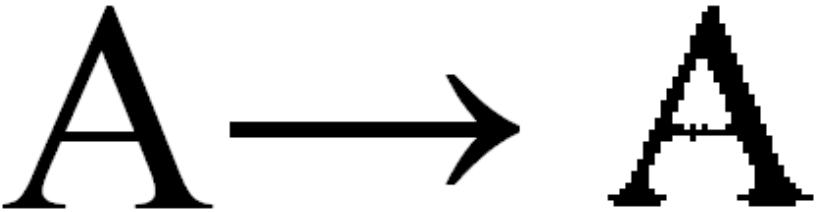

Рис. 1

То есть из одинаковых символов при сканировании мы получаем множество различных образов, при этом их количество достаточно велико. При таком положении дел, задача распознавания усложняется: если бы образы одного символа были бы абсолютно одинаковыми, то достаточно было бы объединить их в группы и распознать одного представителя этой группы. При этом мы были бы абсолютно уверены, что все символы в этой группе — это именно определенный при распознавании символ. В случае большого многообразия изображений символов часто возникают ошибки, когда один символ ошибочно распознан как другой.

Если бы по нескольким отсканированным изображениям можно было бы однозначно сказать, один ли и тот же символ у них в качестве прообраза или нет, то различие образов уже не было бы столь серьезным препятствием при распознавании. В частности, если мы будем знать вероятность, с которой данные образы совпадают, то сможем выбрать значение, при превышении которого сравниваемые символы будут считаться равными.

Итак, можно сформулировать основную задачу: построить алгоритм, дающий ответ на вопрос «какова вероятность того, что два серых изображения имеют один и тот же прообраз». Отдельно хотелось бы получить количественные оценки колебаний отличий одинаковых символов друг от друга и между различными символами.

Посмотрим, что нам для этого потребуется. Прежде всего, необходимо знать, характер распределения серого образа, т. е., как именно искажается

(размывается) черно-белая буква на бумаге при сканировании. Затем построить меру, по которой сравнивать серые изображения. В итоге, с помощью этой меры построить функцию, вычисляющую необходимую вероятность.

Видится необходимым проверка всех теоретических гипотез и выводов экспериментально, равно как и проведение некоторых предварительных экспериментальных исследований.

3. Влияние аппаратной функции

Как уже было замечено выше, любое черно-белое изображение при сканировании с бумажного носителя, будь то книга, газета, распечатанный на принтере текст или просто набор крестиков-ноликов для сканера выглядит как серая картинка.

Это легко объяснить хотя бы тем, что на светочувствительный элемент сканера может попасть как образ полностью черной области, так и образ границы объекта. У последнего некоторые падающие на него лучи света отражаются от черной поверхности, а некоторые от белой. Черная поверхность эффективнее поглощает световые лучи, и вследствие этого образ такого пограничного участка получается более темный, чем образ белого участка, и в то же время более светлый, чем образ полностью черного участка.

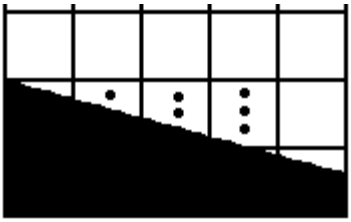

Рис. 2

При этом у областей исходного изображения соответствующих различным светочувствительным элементам (различным положениям одного элемента) может оказаться разное соотношение белого и черного. Как результат получаем разнообразные оттенки серого. На рис. 2 приведен пример наложения на сетку сканера наклонной линии. При сканировании в сером режиме в ячейках сетки отмеченных точками будем наблюдать различные оттенки серого, более того, очевидно, что мы можем заранее сказать, что чем больше точек, тем более светлым получится пиксель.

В реальности же все еще сложнее. Дело в том, что, во-первых, буквы не бывают идеально черными, а бумага не бывает идеально белой. Осо-

бенно это хорошо заметно при сканировании газетных текстов или пожелтевших от времени страниц старых книг. Во-вторых, стекло сканера редко бывает идеально чистым — всегда присутствуют различные пылинки, ворсинки и прочие загрязнения. И, наконец, в-третьих, свет при прохождении сквозь оптическую систему рассеивается, и на сенсор сканера попадает уже искаженное отражение.

Искажения первого типа удается в значительной мере нейтрализовать различными нормировками (например, увеличив контрастность) и несложными преобразованиями (к примеру, снятие фона).

Искажения второго типа — добавочный шум — в эту категорию входят аппаратные искажения сканера.

С третьим типом искажений бороться, наверное, сложнее всего. Рассмотрим этот случай подробнее. В процессе пути отраженные от сканируемого объекта лучи света до попадания на сенсор отражаются несколько раз от зеркал и проходят через несколько линз. Как известно, подобное искажение — дефокусировка — математически описывается гауссовской сверткой. Ниже мы дополнительно проверим это экспериментально, получив попутно ряд интересных результатов.

3.1. Одномерный случай

Проведем несколько экспериментов. Для этой серии экспериментов использовался сканер Fujitsu модель Fi4220C2.

Возьмем неширокую вертикальную черную линию, напечатанную для чистоты эксперимента на белой офисной бумаге и на хорошем лазерном принтере. Максимально точно выровняем её, так, чтобы направление вертикальной оси линии совпадало с вертикальными линиями сетки сканера. Отсканируем с разрешением 600 dpi в сером режиме, и разрежем полученную серую вертикальную линию на горизонтальные полоски шириной в 1 пиксель. Получается примерно следующее (рис. 3, увеличено):

Рис. 3

Рассмотрим изменение численного значения полутонов пикселей при перемещении от левого края к правому. В каждой полоске мы наблюдаем

постепенное увеличение интенсивности до некоторого максимума с последующим уменьшением до уровня фона.

Изобразим данные на графике. Отложим по вертикальной оси численное значение оттенка, а по горизонтальной — порядковый номер пикселя в полоске. На рис. 4 на одном графике одновременно изображено 250 линий. Видно, что все линии расположены в окрестности некоторого среднего значения.

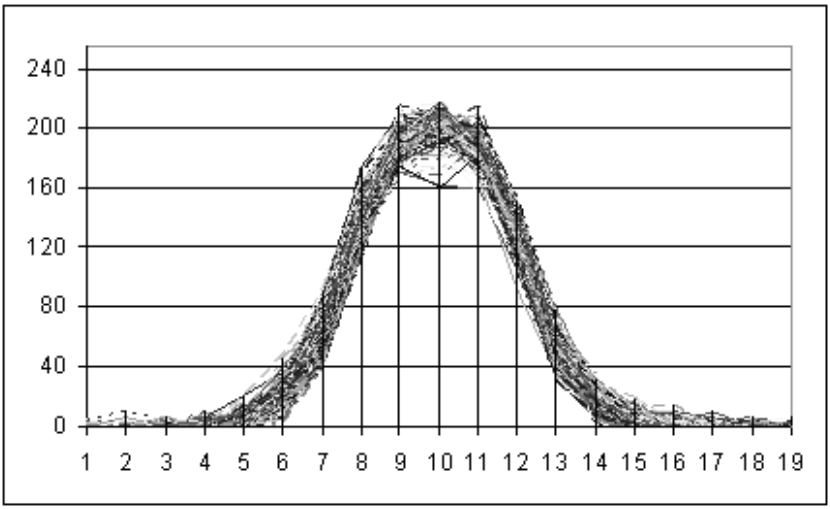

Рис. 4

Нельзя не заметить сходство среднего значения графиков с графиком плотности нормального распределения $f(x) = Ae^{-\frac{x^2}{\sigma^2}}$ с точностью до выбранного масштаба по осям и сдвига по горизонтальной оси. За счет подбора параметра A и σ можно сделать гауссиан таких же размеров, как и наш колокол.

Более точно было бы сказать, что форма гауссиана наблюдается на склонах получившегося колокола. Однако, это первое впечатление обманчиво — уравнение кривой склона колокола отличается от приведенной функции $f(x)$.

Прежде всего, приведем результаты аналогичных экспериментов с линиями различной ширины и изобразим на едином графике результат усреднения всех полученных колоколов. Опишем технологию, по которой проводился эксперимент.

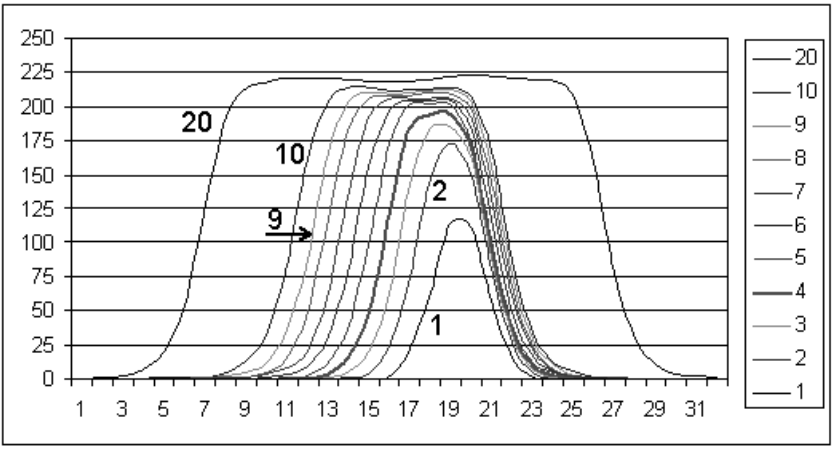

Рис. 5

Вначале напечатали на принтере с разрешением печати 600 dpi полоски шириной 1, 2, …, 9, 10 и 20 пикселей. Печать проводилась так, чтобы ширина полоски в 1 пиксель была бы равна 1/600 дюйма. После этого проведено сканирование с разрешением 600 dpi, и по уже описанной выше методике (в усреднении участвовало 250 полосок для каждой линии) получены графики колоколов для всех линий. Точное центрирование колоколов не производилось — только небольшой сдвиг, необходимый чтобы меньший колокол был внутри большего. На рис. 5 приведены слегка сглаженные, для большей наглядности, графики.

Как видно из приведенного графика, характер поведения боковой поверхности колокола достаточно постоянен. Высота же колокола меняется в зависимости от толщины сканируемой линии. Это вполне объяснимо тем, что ячейки соответствующие центральной части толстой линии практически не засвечиваются лучами рассеивания от краев этой линии. У тонких же линий за счет противоположных эффектов — сильной засветки с двух сторон — наблюдается значительное увеличение яркости (уменьшение интенсивности черного).

3.2. Численное значение оттенков

Основной тип искажений вносимый оптической системой сканера — дефокусировка. Преобразование дефокусировки математически описывается гауссовской сверткой. Предположим, что каждый пучок отраженного света обеспечивает на изображении небольшой холмик в форме гауссиана

и задается функцией $f(x) = Ae^{-\frac{x^2}{\sigma^2}}$ — в положительные параметры A и σ мы включили все коэффициенты — для упрощения расчетов.

На схеме (рис. 6а) изображено, как каждый участок отражаемого изображения обеспечивает соответствующий холм — график гауссиана. Жирной кривой линией отмечен график суммы получившихся гауссианов. При уменьшении размеров участков отражения, высота холмиков и, соответственно, вклад каждого участка уменьшается, а их количество на единицу длины увеличивается. При этом происходит быстрое сглаживание графика суммы (рис. 6б).

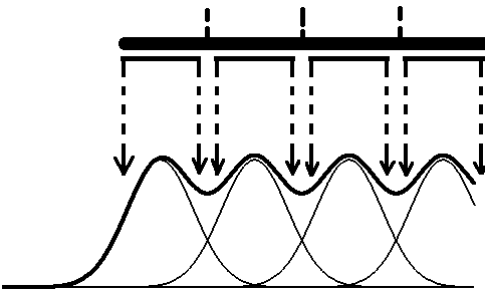

Рис. 6а

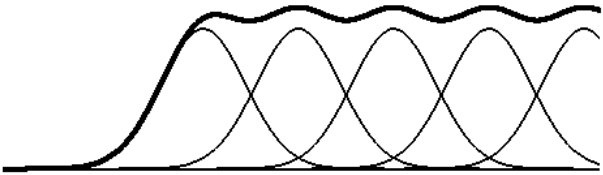

Рис. 6б

При дальнейшем уменьшении шага сканирования, описанные эффекты усиливаются, суммарный график получается совсем ровным. При стремлении шага суммирования к нулю, получаем непрерывный случай, и сумма гауссианов вычисляется по следующей формуле:

$$A\sigma\left(\frac{\sqrt{\pi}}{2}+\int_0^{\frac{x}{\sigma}} e^{-t^2}\,dt\right). \qquad (1)$$

Последняя функция, с подобранными параметрами A и σ, как раз и задает форму склона колокола, который мы получили экспериментально.

Точной функцией, задающей форму колокола, является интеграл свертки (погрешностью, возникающей за счет белого шума, мы пренебрегаем):

$$\varphi(x) = \int_{-\infty}^{+\infty} f(x-y)g(y)dy,$$

где функция $g(x)$ — индикаторная функция черно-белой полоски исходного изображения. Функция $f(x)$ — это аппаратная функция искажения сканера — как мы показали выше можно считать, что это функция плотности нормального распределения с коэффициентами.

Значения параметров A и σ можно получить методом регрессии по имеющимся численным данным. Так, в серии проведенных экспериментов для линий, толщина которых не менее 4 пикселей, искомые значения были вычислены: диапазон значений составил от 1,7 до 1,9 для σ и от 62 до 66 для параметра A. Значения параметра A для более тонких линий будет несколько меньше. Регрессию производили с помощью функции leastsquare() математического пакета Maple8 и функции nlinfit() пакета MATLAB6.5.

Влияние разрешения сканирования

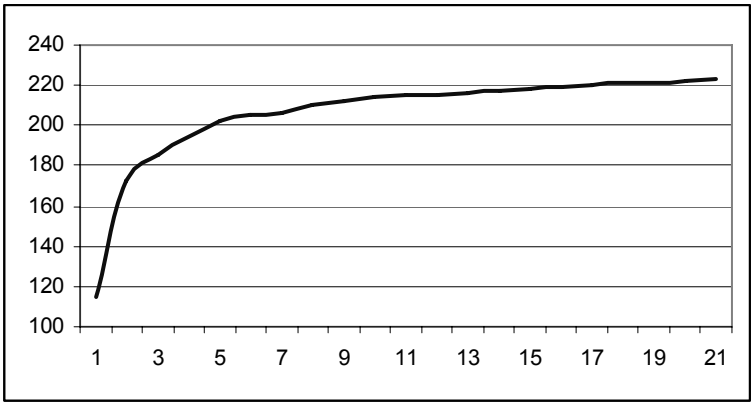

Рис. 7

Значение максимальной по высоте точки колокола ведет себя примерно как логарифм — с увеличением ширины линии скорость увеличения максимума быстро уменьшается (рис. 7). По горизонтальной оси отложена

ширина линии, по вертикальной значение наиболее черных пикселей в колоколе соответствующем линии с данной шириной. Напомним, что абсолютно черному цвету соответствует значение 255.

Так же был проведен ряд аналогичных экспериментов при меньшем разрешении сканирования, в которых были получены аналогичные результаты.

3.4. Двумерное распределение

Чтобы исследовать возможную анизотропию сканера — различие свойств в зависимости от направления полосок, все описанные выше эксперименты были проведены как для вертикальных линий, так и для горизонтальных линий. Выяснилось, что при сканировании горизонтальных линий картина не отличается от описанной выше. То есть для используемой модели сканера этот вопрос не актуален. Однако не исключается, что у других моделей сканеров других производителей может присутствовать некоторая анизотропия.

Тем самым и в горизонтальном направлении и в вертикальном рассеивание имеет нормальное распределение. Из теории вероятности известно, что двумерным распределением, удовлетворяющим указанным свойствам, является *двумерное нормальное распределение*. Делаем вывод, что искажения рассеивания в цельном (не порезанном на полоски) плоском образе можно смоделировать сверткой *функции образа* с функцией:

$$f(x,y) = Ae^{-\frac{x^2+y^2}{\sigma^2}}.$$

Функция образа — это функция-индикатор g(x, y) — равна 1, если образ в данной точке (x, y) черный, и равен 0 в противном случае. Свёртка функций $f(x, y)$ и $g(x, y)$ — это функция:

$$\varphi(x,y) = \int\limits_{a\in A}\int\limits_{b\in B} f(x-a, y-b)g(a,b)\,da\,db.$$

Отрезок прямой линии (в пространстве представляется прямоугольником) при свертке преобразуется в вытянутый объемный колокол. Прямоугольная рамка в середине колокола показывает размеры оригинального отрезка линии.

Приведем для большей наглядности пару иллюстраций: фигура до применения свертки, и после (см. рис. 8).

Именно так выглядит отсканированный объект, если его нарисовать в координатах (x, y, z), где z — это численное значение серого оттенка пикселя с координатами (x, y) на сетке сканера.

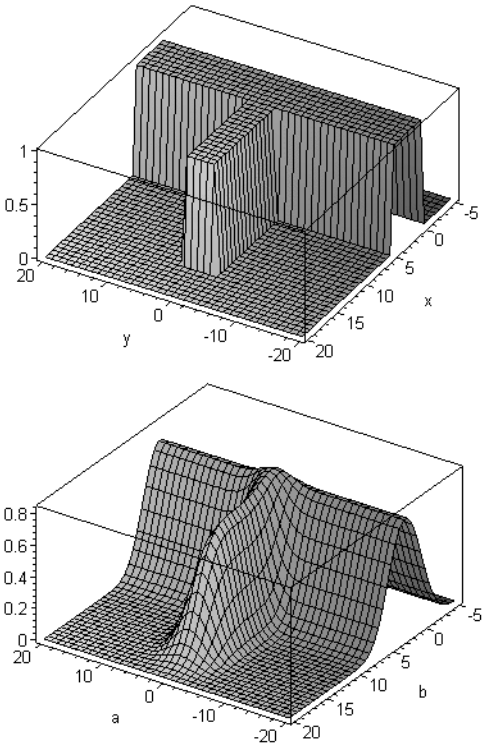

Рис. 8

4. Размеры прообраза

Интересным является вопрос определения размеров оригинального изображения символа. Как уже говорилось выше, наиболее простым способом приведения серого изображения отсканированного текста в задачах распознавания является бинаризация.

Рассмотрим рис. 9 — на нем изображены колокола линий известной ширины 1, 2, …, 9, 10 и 20 пикселей. Проведем у всех колоколов горизонтальные хорды, длины которых были бы равны ширине отсканированной линии.

Как видим, все хорды попадают в диапазон 100–150 по полутоновой шкале. Наибольшая их концентрация наблюдается в интервале 115–140, а, если взять среднее значение, то получим 125–130.

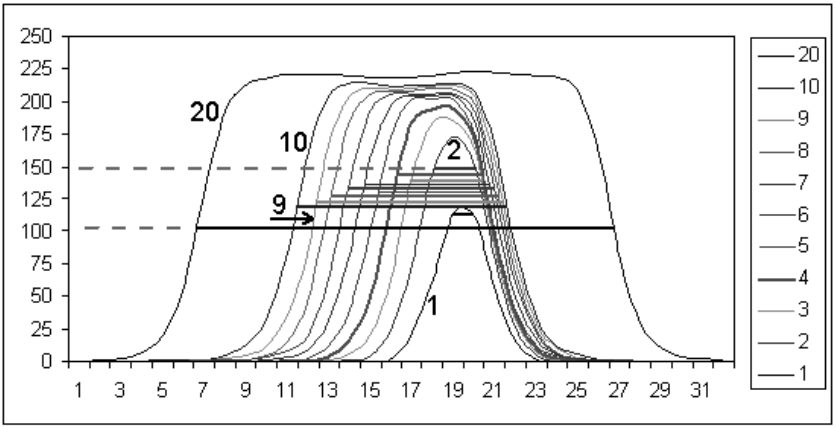

Рис. 9

Самое время вспомнить, что у сканеров сканирующих в черно-белом режиме по умолчанию предлагается порог равный 128. Что очень хорошо вписывается в полученные нами данные. То есть порог, равный 128 обеспечивает в среднем наибольшую приближенность размеров полученного ч/б изображения размерам оригинального образа.

Эксперимент только подтверждает теорию: функция Лапласа — нечетная, т. е. имеет центр симметрии, ордината которого равна полусумме проекций её асимптот на вертикальную ось. При применении преобразования свертки именно середина «холма» почти не изменяется, а другие значения отклоняются в более светлую или более темную сторону. На рис. 10 ширина оригинального изображения была 10 единиц; по вертикальной оси отложены условные единицы.

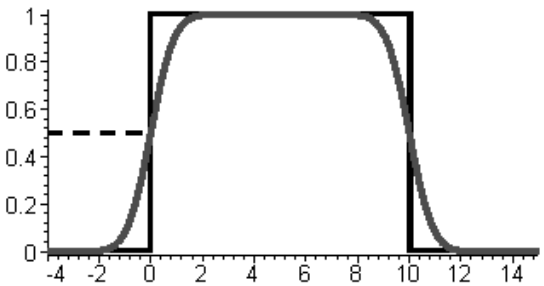

Рис. 10

Был проведен дополнительный эксперимент. Сканировались изображения букв в сером режиме, и в черно-белом. После этого серое изображение бинаризовывалось с порогом 128 и результат сравнивался с черно-белым. Наблюдаемое отличие было не бóльшим, чем отличаются два ч/б изображения, отсканированных друг за другом без сдвига бумажного носителя.

Стоит заметить, что высота колокола соответствующего линии шириной в 1 пиксель как раз находиться в этом диапазоне, причем в его середине. То есть совсем небольшое изменение порога может отрезать тонкие линии в результатах бинаризации.

4.1. Поведение тонких линий

Тонкие линии, при сером сканировании дают не тонкую контрастную линию, а бледно-серое изображение, причем не самой маленькой ширины. То есть две тонкие линии, отличающиеся в 1,5 раза по толщине, будут давать серые растры, отличающиеся по ширине незначительно. Отличие же их друг от друга по насыщенности черного цвета будет хорошо заметно. Поэтому обычная операция взятия порога может оказаться критической — исчезнут детали, на тонкой длинной линии появятся множественные разрывы.

Именно таким образом, ведут себя острые засечки у серифных шрифтов и прочие элементы букв — сканируются в заметные серые выступы, а после стандартного взятия порога — исчезают, скругляются:

Рис. 11

На рисунке показано, как размываются при сканировании, в частности, тонкие линии у буквы F, и что получается после взятия порога (уровень 128). У четвертой буквы рамкой помечены границы исходного оригинального символа. Иллюстрации сделаны искусственно, и аппаратная функция размытия изображения специально бралась чуть «более сильная», чтобы нагляднее показать процесс исчезновения деталей — остроконечных засечек у символа.

Толстые же линии — основной стержень буквы F, совершенно не изменился.

5. Наклонные линии

До настоящего момента все описанные эксперименты проводились исключительно на линиях, параллельных к тому же линиям сетки сканера, поскольку такие объекты — наиболее просты для исследования. Как будет видно впоследствии, полученный результат уже является достаточно содержательным.

Сейчас будет описан ряд экспериментов, целью которых было проверить корректность сделанных выше утверждений, а так же ряд попутно полученных интересных фактов.

5.1. Наклонные линии на сетке

Известна простая задача по программированию для начинающих — нарисовать попиксельно на экране монитора отрезок, соединяющий две точки с заданными координатами. Сложность задачи состоит в том, что отрезок получается при этом ступенчатым и необходимо понять, каким образом формировать длину каждой ступеньки.

Решением же является такая зависимость длины ступенек, при которой длина ступенек отличается попарно не более чем на 1 пиксель (рис. 12), и ступеньки некоторым несложным образом чередуются, так что их длина равна последовательно: k, $k + 1$, k, $k + 1$, k… Или — уже другая схема — k, k, $k + 1$, k, k, $k + 1$, k, k, и т. п. То есть наблюдается регулярность.

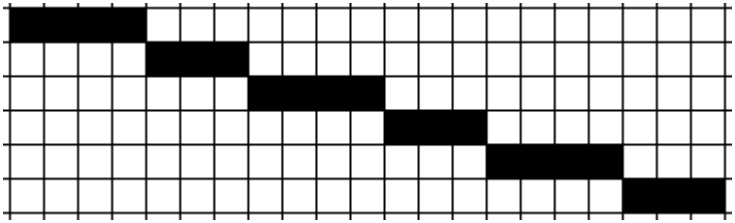

Рис. 12

Именно такому закону формирования удовлетворяют ступенчатые структуры, образующиеся при сканировании в ч/б режиме наклонных прямых линий. На рис. 13 изображен фрагмент реальной отсканированной наклонной линии (шахматная сетка — добавлена только для удобства счета). Длины ступенек сверху равны 5, 4, 5, 4… А длины ступенек снизу — 4, 5, 4, 4, 4. Незначительные нарушения строгости чередования длин ступенек происходит вследствие неизбежности искажений при реальном сканировании.

Рассмотрим несколько подробнее, что же происходит «на границе», т. е. как себя ведут пиксели, значения оттенков которых, близко к порогу.

Были проведены эксперименты для изучения изменения размеров и формы наклонной линии в результатах сканирования при изменении параметров яркости/порога у сканера. Мы не зря объединили эффекты от изменения яркости и порога. Сделано это потому, что изменения, получающиеся вследствие изменения указанных параметров, имеют один и тот же характер.

Общее поведение следующее: чем больше яркость, и чем меньше значение порога, тем тоньше получается образ линии. Причем уменьшение толщины происходит без существенного изменения формы линии — т. е. угол наклона практически не изменяется, и соотношение длин отрезков в ступенчатой структуре сохраняется.

5.2. Определение толщины ч/б оригинала по имеющемуся ч/б изображению

Будем называть ч/б оригиналом тот объект, который изначально печатался на бумажном носителе, т. е. его векторную сущность. Решим задачу получения точных размеров и угла наклона относительно сетки сканера ч/б оригинала.

Чтобы отличать прямую бесконечную линию от сканируемых объектов, а так же от таких математических объектов как «отрезок» и «прямая», будем называть результат сканирования отрезка прямой некоторой толщины *колонной* [3].

Для начала посчитаем площадь колонны, и разделим площадь на её длину, которую в свою очередь вычислим по теореме Пифагора, как гипотенузу треугольника составленного из сторон минимального прямоугольника включающего нашу колонну (стороны прямоугольника параллельны линиям сетки сканера).

Рассмотрим общую схему на примере (рис. 13). Имеем ч/б растр (на рисунке шахматный фон добавлен для удобства подсчета пикселей): высота прямоугольника 11 пикселей, ширина 42, площадь колонны 96.

После вычислений для предложенных данных мы получили, что угол наклона равен 21,82° (оценочная погрешность ±1 градус), длина колонны равна 42,9 пикселей, а ширина 2,2 пикселя.

Допустим, нам удалось построить *ч/б оригинал* (зная, что это колонна). Тогда определить его местоположение относительно *ч/б растра* колонны можно совмещением изображений, выбором такого положения, чтобы сумма площадей выступающих и недостающих частей ч/б растра была бы минимальна.

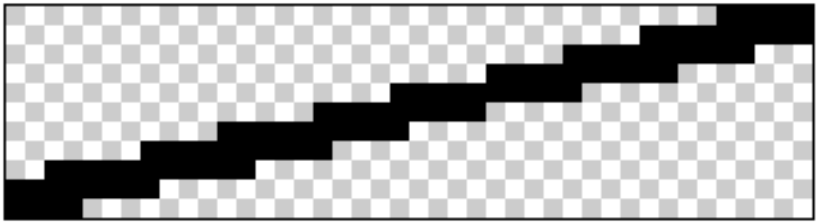

Рис. 13

Замечание. При дискретном способе поиска наилучшего сдвига, возникают вопросы о наложении идеальной буквы (в нашем случае колонны) на прямоугольную сетку. Действительно, что такое «подвигать чуть-чуть» при наложении — это подвигать с некоторым шагом. А что такое выступающие и недостающие части? Это количество мелких пикселей на новой сетке, которые покрыты одним образом и не покрыты другим.

Размер новой мелкой сетки нужно брать тем меньше, чем большую точность мы хотим достигнуть. Как самый грубый вариант — сетка в 2 раза меньшая, и совпадающая с имеющейся сеткой по основным линиям. На приведенных рисунках новая сетка имеет размер ячеек в 8 раз меньше, чем у исходного ч/б растра.

Не смотря на предложенный выше достаточно простой способ вычисления угла наклона прямой линии, остается, в большей степени из академического интереса, открытым вопрос — как теоретически рассчитать угол наклона, зная, к примеру, функцию, описывающую зависимость значения полутонов пикселей, в горизонтальных или вертикальных сечениях линии (аппаратная функция считается известной).

Для начала решим обратную задачу: при известной аппаратной функции рассеивания и известном угле наклона прямой линии, определить функцию зависимости оттенков точек образа в горизонтальных и вертикальных сечениях рассматриваемой линии.

Решение, как ни странно, лежит на поверхности. Достаточно рассмотреть наклонные сечения вытянутого колокола (рис. 14), и вычислить функцию образующейся при этом кривой. Пусть угол между плоскостями $\boldsymbol{m}$ и $\boldsymbol{l}$ равен $\boldsymbol{\alpha}$, тогда точки колокола в плоскости $\boldsymbol{m}$ будут отстоять от оси колокола в $cos(\boldsymbol{\alpha})$ раз дальше, чем соответствующие точки колокола в плоскости $\boldsymbol{l}$, поскольку колокол на плоскости $\boldsymbol{l}$ — это проекция колокола на плоскость $\boldsymbol{m}$. В свою очередь это равносильно растяжению графика m-колокола в $cos(\boldsymbol{\alpha})$ раз относительно вертикальной оси. То есть, если функция, задающая колокол в перпендикулярном сечении (плоскость $\boldsymbol{l}$) равна (1), то аналогичная функция в плоскости $\boldsymbol{l}$ будет равна

$$\Phi(x) = A\sigma\left(\frac{\sqrt{\pi}}{2} + \int_0^{\frac{x}{\sigma\cdot\cos\alpha}} e^{-t^2}\,dt\right).$$

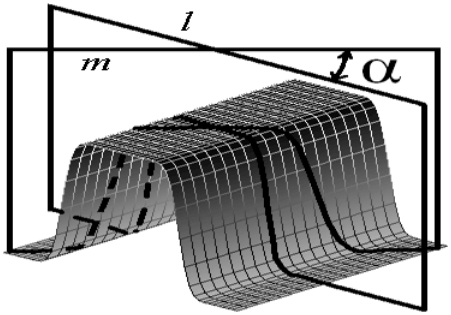

Рис. 14

Возвращаясь к первоначальному вопросу, остается заметить, что, зная аппаратную функцию и используя полученную формулу, можно подобрать параметр α такой, чтобы функция распределения совпала и имеющимися экспериментальными данными, т. е. найти угол наклона. Стоит, однако, заметить, что погрешность этого метода на практике будет слишком велика.

6. Непрерывная модель. Количественные характеристики

Выше, мы предположили, что аппаратная функция описывается гауссовским распределением

$$f(x) = Ae^{-\frac{x^2}{\sigma^2}} \qquad \text{и} \qquad f(x,y) = Ae^{-\frac{x^2+y^2}{\sigma^2}}$$

для одномерного и двумерного случая соответственно. После этого теоретически была выведена функция, описывающая закон изменения оттенка на границе отсканированного символа. Найденной функцией $\boldsymbol{\Phi(x)}$ является сдвинутая и растянутая относительно координатных осей функция Лапласа. Ряд проведенных экспериментов хорошо согласуется с этим выводом.

Отсканированный образ — есть свертка оригинала изображения с аппаратной функцией. Параметры последней были нами подобраны для нескольких моделей сканеров различных производителей.

Озаботимся вопросами распределения полутонов на плоскости в количественном отношении:

- Какой диапазон оттенков приходится на единицу длины/площади?
- Какую длину/площадь занимает заданный диапазон оттенков?

Ответим вначале на первый вопрос.

Заметим, что при бесконечном количестве допустимых оттенков (т. е. в непрерывном случае), речь может идти только о диапазонах оттенков или об их мере, поскольку слово «количество» плохо применимо к континуальным множествам. В случае же ограниченного количества цветов (дискретный случай), вопрос «количество» приобретает уже вполне ясное и естественное значение. Аналогично ситуация обстоит с дискретизацией по координатам.

Термин «длина» мы будем употреблять применительно к одномерным ситуациям, а так же при рассмотрении одномерных сечений.

Чтобы не выходить за рамки непрерывного случая, вычислим диапазон различных оттенков встречающихся на некоторой полоске фиксированной ширины.

Введем понятие меры. Если в данной полоске встречаются (все) оттенки от 230 до 240, то будем говорить, что мера количества цветов в этой области равна 240 – 230 = 10. Предполагается, что значение полутонов меняется плавно, т. е. непрерывно.

Для начала рассмотрим самый простой случай, когда в рассматриваемой области находится отсканированный участок прямолинейной границы между черной и белой полуплоскостью, при этом линии сетки сканера параллельны этой границе.

Пусть для определенности граница вертикальна. В идеальном непрерывном случае, при введенных допущениях, горизонтальным сечением склона «колокольного холма» будет являться одна и та же линия (рис. 15). График изменения оттенков от черного к белому будет описываться функцией $\boldsymbol{\Phi(x)}$ (1). По горизонтальной оси отложена координата точки, а по вертикальной — значение полутона этой точки (от 0 до 255).

Из приведенного рисунка видно, что мера оттенков попавших в полосу от $\boldsymbol{x_k}$ до $\boldsymbol{x_{k+1}}$ равна:

$$\Delta f = f(x_{k+1}) - f(x_k).$$

При этом ширина диапазона по оси $\boldsymbol{x}$ равняется $\Delta x = x_{k+1} - x_k$.

Другими словами, отношение количества оттенков на единицу длины в данной точке есть предел отношения приращения аргумента к прираще-

нию функции, т. е. её производная. Полученное вполне адекватно: чем быстрее убывает значение функции — гауссиана, тем большее оттенков вместится в заданную полоску.

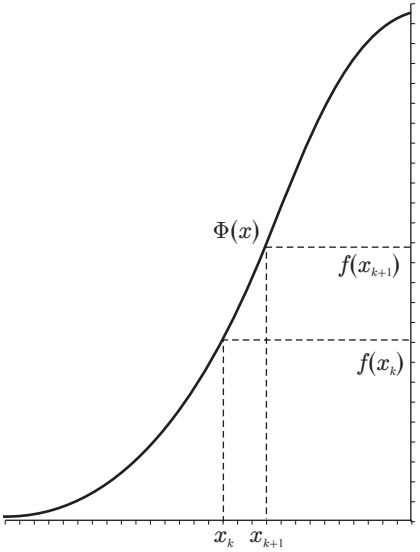

Рис. 15

Для второго вопроса — какую длину/площадь занимает заданный полутоновый диапазон — ответом является противоположная величина, и при предельном переходе для произвольной точки x_0 имеем:

$$\lim_{x\to x_0}\frac{x-x_0}{f(x)-f(x_0)}=\frac{1}{f'(x_0)}=\left(f^{-1}\right)'(y_0),$$

где последнее равенство следует из теоремы о производной обратной функции. Так как $f(x)$ — это функция Лапласа (график № 1 на рис. 16), то находим для нее обратную функцию (график № 2). Для последней функции дифференцированием находим производную (график № 3).

Итак, мы получили зависимость меры оттенков на единицу длины для одномерного сечения в непрерывном случае. График № 3 показан ниже уже с указанием масштаба по осям (рис. 17, линия № 1), при этом коэффициенты A и σ выбраны из расчета, что значение полутонов меняется от 0 до 255, а рассеивание — среднее для сканеров. По горизонтальной

оси отложено полутоновое значение. По вертикальной оси — значения производной обратной функции — их можно использовать, если мы хотим вычислить долю занимаемую оттенками заданного диапазона.

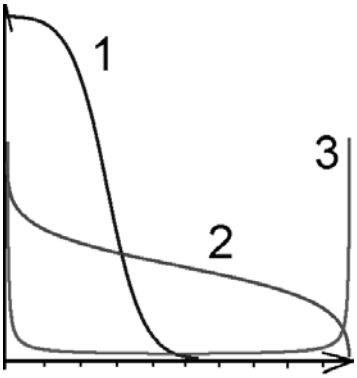

Рис. 16

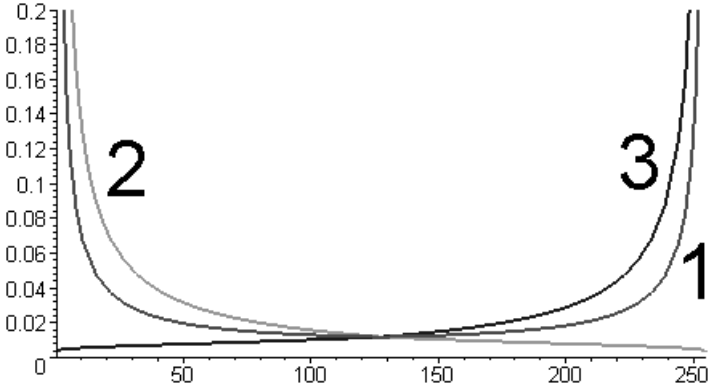

Рис. 17

Анализ графика № 1 (рис. 17) позволяет сделать вывод, что оттенков в диапазоне от 120 до 170 практически не будет, а оттенки со значением от 0 до 10 (почти белые) и от 250 до 255 (самые черные) будут занимать существенную долю отсканированного изображения. Заметим, что на практике абсолютно черных пикселей почти не получается. Как правило, у реальных сканеров есть некоторые средний «самый черный» уровень, принимающий обычно значение 200–220. То есть именно самые темные

и самые светлые пиксели будут занимать большую часть отсканированного изображения.

Заметим, что поскольку найти функцию обратную для функции Лапласа аналитически не удается, для построения графиков применялись численные методы.

Рассмотрим, как будет меняться найденная зависимость при усложнении фигуры — сканируемого образа. Рассмотрим вначале всюду выпуклый объект — черный круг. Так как длина окружности прямо пропорциональна радиусу, то отношение «количество светлых пикселей» / «количество темных пикселей» для отсканированного черного круга будет больше, чем аналогичное отношение для соответствующих оттенков прямолинейной границы — первой модели текущего этюда.

Изображенный на рис. 17 график 1 для распределения полутонов изменится — количество темных оттенков на единицу длины уменьшится по отношению к количеству светлых оттенков. Формула для этой зависимости умножится на некоторый двучлен вида $kx + b$, где $b > 0$, $k < 0$, и интеграл от 0 до 255 от $kx + b$ будет равен 255 — т. е. площадь под графиком этого двучлена равна площади на этом же интервале под графиком функции $y = 1$.

На рис. 17 изображены три графика, соответствующие прямолинейной границе (№ 1), всюду выпуклой фигуры — черного кружка малого радиуса (№ 2), и всюду вогнутой фигуры — белое круглое отверстие малого радиуса на черном фоне (№ 3).

Замечание про малый радиус сделано из тех соображений, что при достаточно большом радиусе черного кружка, при применении свертки середина этого кружка свой цвет практически не изменит, и резкое увеличение значения функции при аргументе 254–255 (соответствует абсолютно черному) будет сохраняться. Изменения же будут состоять только в некотором увеличении значений функции при аргументе 0–100 относительно значений при аргументе 150–250. Абсолютные же значения — увеличатся повсеместно, так как с увеличением радиуса увеличится длина окружности сканируемого круга (рис. 18).

График 1 на рис. 18 — это диаграмма для круга с небольшим радиусом, 2 — радиус больше, 3 — радиус кружка — еще больше — отличия в симметрии между правой и левой половиной почти не заметно на глаз.

Из общих закономерностей формирования функции распределения полутонов можно выделить следующие:

1) выпуклые во внешнюю сторону относительно символа участки увеличивают значения функции распределения при малых значениях аргументах (меньше 100);
2) выпуклые во внутреннюю сторону (вогнутые) относительно символа участки увеличивают значения функции распределения при больших значениях аргументах (больше 150);

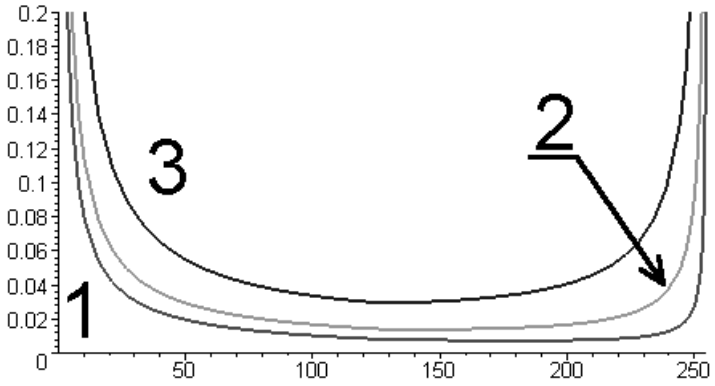

Рис. 18

3) прямолинейные участки (вроде колонн букв Т или П) не вносят качественных изменений;
4) общий уровень значения функции зависит от длины периметра символа — чем он больше, тем больше абсолютные значения функции;
5) увеличение площади оригинального символа (коррелирующей с площадью бинаризации отсканированного изображения) обеспечивает увеличение значений функции в области самых больших значений аргумента (240–255);
6) увеличение площади фона — границ рассматриваемой области обеспечивает увеличение значений функции в области самых маленьких значений аргумента (0–20).

Приведем на одних осях несколько реальных примеров, подтверждающих теорию, и иллюстрирующих вид диаграммы при реальных сканируемых символах. Стоит заметить, что в реальности мы имеем дискретный случай, как следствие — соответствующие изменения внешнего вида графиков — изломы, резкие пики и прочие артефакты. Имеющиеся закономерности лучше наблюдаются на сглаженных графиках (рис. 19).

По горизонтальной оси отложено числовое значение оттенка пикселей, а по вертикальной оси — количество пикселей данного оттенка.

Первое, что бросается в глаза, что правый максимум не прижат к краю диаграммы (значения полутонов 250–255), а несколько отделен от него. Это явление обуславливается тем, что сканер не сканирует черное изображение в абсолютно черный образ, и значения черного получаются на некотором среднем уровне (может отличаться для различных моделей сканера). Аналогично — для «фона», т. е. для самых светлых оттенков — после сканирования эти участки, как правило, не являются абсолютно белыми.

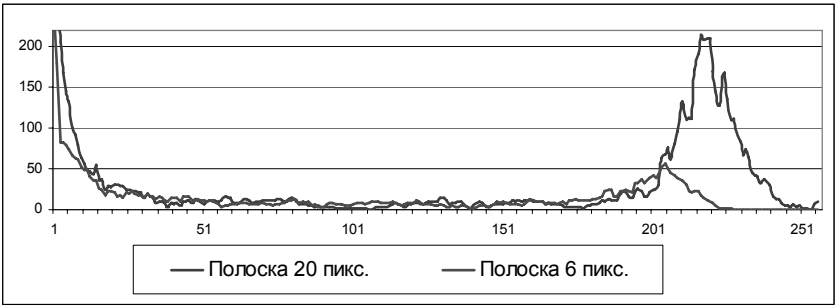

Рис. 19

7. Вероятности совпадения прообразов

В данном этюде мы построим функцию вероятности, которая будет давать ответ на вопрос «какова вероятность того, что два образа имеют один и тот же прообраз».

Существует несколько подходов для сравнения двух изображений, в том числе: метод наложения, сравнение наборов признаков и т. п. В зависимости от метода получается и результат: где-то ответ дискретный, где-то непрерывный.

В данной работе применялся достаточно простой и естественный способ равнения — метод простого наложения. Для подсчета меры различия двух изображений производился подсчет суммы модулей разностей полутоновых значений пикселей. При этом выбирается такое взаимное расположение объектов, что мера их различия минимальна.

$$\sum_j \left| a_j - b_j \right| \to \min .$$

Первая проблема, с которой приходится сталкиваться — необходимость определения размеров частей изображения подлежащих сравнению, и расположение внутри этих областей основной несущей информации.

Проиллюстрируем сказанное, рассмотрев две буквы «А» в прямоугольнике 100 на 50 пикселей. Если окажется, что в одном изображении буква «А» сдвинута влево, а в другом вправо, то даже если буквы полностью совпадают по форме и размерам, попиксельная сумма разностей модулей будет отличаться значительно.

Если не ограничивать время работы алгоритма, то можно перебрать всевозможные взаимные расположения двух изображений и выбрать минимальное значение меры.

Для поиска наилучшего положения, в данной работе был использовано следующий метод: производилась бинаризация по уровню 50 %, затем у черно-белого изображения определялось необходимое для центрирования смещение внутри своего прямоугольника, и это смещение применялось для исходного серого изображения. Со вторым изображением производили такую же операцию, и затем, после снятия фона (по уровню 10 %) их сравнивали. Аналогичные сдвиги осуществлялись в каждый из 4-х углов содержащего символ прямоугольника. При этом в каждом из пяти описанных положений (центр + 4 угла) производился сдвиг ±1 пиксель по всем направлениям. Указанный способ обеспечивает поиск взаимного расположения, при котором изображения совпадают на наибольшей площади, и сумма модулей разностей принимает наименьшее значение.

Во избежание сильного влияния размеров изображений на значения меры осуществлялась нормировка по площади сравниваемых участков, а для ограничения вклада белого шума в сумму пришлось прибегнуть к снятию фона на уровне 10 %.

Пороговое значение бинаризации при поиске необходимого сдвига, а так же границы подсчета суммы модулей разности в сером изображении брались на основе результатов описанных в предыдущих этюдах.

В качестве меры бралось ближайшее целое число, не превосходящее полученное значение (вообще говоря, построенная мера — непрерывная функция).

Рис. 20

Приведем на едином графике пример получившихся значений меры для пар различных и одинаковых символов (рис. 20). На графиках пред-

ставлены усредненные данные большого количества замеров для заглавных букв шрифта Arial. Были проведены эксперименты для различных шрифтов, жирных и курсивных модификаций, и так же для прописных букв. Результаты при этом были очень схожи с приведенными. Во всех экспериментах использовалось разрешения сканирования 600 dpi. Данные сравнивались для моделей сканеров различных производителей.

На рис. 20 по вертикальной оси отложено значение меры, а по горизонтальной оси — порядковый номер буквы в алфавите. Сравнения между одинаковыми символами представляют среднее значение между 32 замерами для каждого символа (всего 33×32 замера). Сравнения между различными символами — среднее значение двух замеров попарно для всех букв алфавита (всего 33×32 замера).

Для построения функции вероятности нам потребуется ввести несколько событий.

Пусть событие A = «два сравниваемых образа имеют один и тот же прообраз», т. е. это отсканированные образы одного и того же символа. Противоположное A событие $\tilde{A}$ = «два сравниваемых образа имеют различные прообразы».

Введем событие B_k = «мера различия двух образов равна k».

Из экспериментальных данных нам известны условные вероятности $P(B_k|A)$ — вероятность того, что мера принимает значение B_k при условии совпадения прообразов; $P(B_k|\tilde{A})$ — вероятность того, что мера принимает значение B_k при условии различия прообразов. Указанные зависимости показаны на рисунках (обратите внимание на значения по осям).

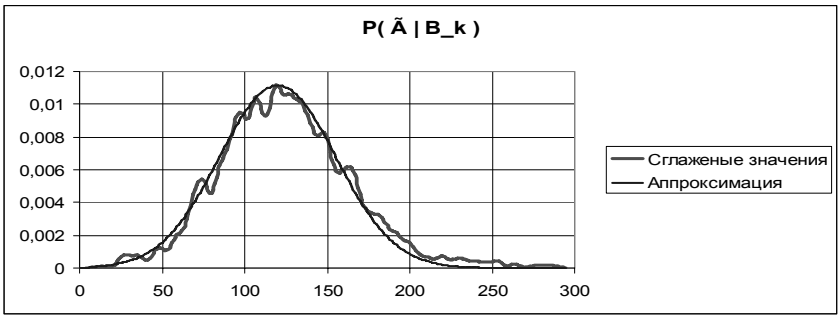

Рис. 21

На рис. 21 и 22 помимо экспериментальных значений условных вероятностей изображены их аппроксимации функцией Гаусса. Как будет видно ниже, для дальнейших вычислений нам важно знать значение $P(A|B_k)$ на

правом склоне и $P(\tilde{A}|B_k)$ на левом склоне, так как именно на этих участках будет происходить основное изменение итоговой функции вероятности. В свете сказанного аппроксимация гауссианом подбиралась соответствующим образом.

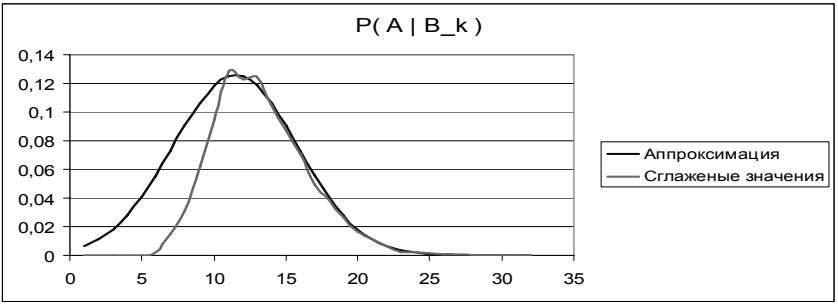

Рис. 22

Итак, нас интересуют условные вероятности $P(\tilde{A}|B_k)$ и $P(A|B_k)$, т. е. какова вероятность совпадения/различия прообразов при данном значении меры.

Воспользовавшись формулой Байеса

$$P(A|B_k) = P(A)^* P(B_k|A)/P(B_k)$$

и формулой полной вероятности

$$P(B_k) = P(A)^* P(B_k|A) + P(\tilde{A})^* P(B_k|\tilde{A}),$$

получаем:

$$P(A|B_k) = P(B_k|A)/[P(B_k|A) + P(\tilde{A})/P(A)^*P(B_k|\tilde{A})].$$

То есть искомая вероятность зависит только от $P(A)$, так как $P(A) = 1 - P(\tilde{A})$. Напомним, что $P(A)$ — это вероятность совпадения прообразов двух произвольно взятых образов. Очевидно, что эта величина зависит от количественного соотношения различных символов в выборке. Так, если все символы в выборке совпадают, то $P(A) = 1$. Если же различных символов в выборке N, и каждый из них встречается ровно 1 раз, то $P(A) = 1/N$.

Оглядываясь на практические задачи сканирования, заметим, что при больших текстовых массивах частотность букв используемого алфавита примерно постоянна, и приближенное значение $P(A)$ можно оценить. Для текстов на русском языке, при использовании стандартной таблицы частотности для алфавита из 33 букв, $P(A)$ оказывается примерно равной 1/18. Это значение получено в предположении, что все буквы имеют один и тот же

регистр (к примеру, все буквы прописные). В реальных текстах эта величина будет несколько меньше.

Дополнительным источником информации о количественном соотношении символов на распознаваемой странице в двухпроходных системах OCR могут служить данные первого прохода [3]. Эти же данные разумно использовать при небольшом количестве распознаваемых символов на странице, так как различия со стандартной таблицей частотности могут быть уже существенны.

В качестве предельного случая можно привести пример распознавания анкеты, в которой необходимо проставлять в заранее отведенные поля ограниченный набор символов — «А», «Б» и «В», или распознавание последовательностей. В этих случаях значение $P(A)$ может значительно приближаться к единице.

На рис. 23 показаны графики значений $P(A|B_k)$ для различных значений $P(A)$; по горизонтальной оси отложены значения k.

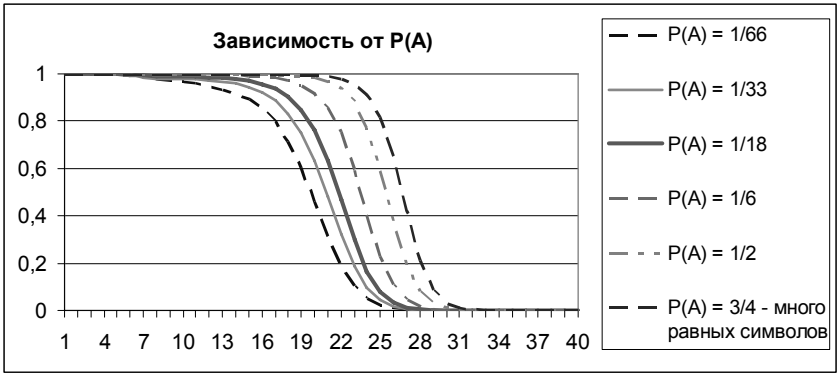

Рис. 23

В задачах распознавания больших массивов текста можно использовать значение $P(A) = 1/18 \ldots 1/70$. При распознавании печатных форм с небольшим количеством различных символов в пределах одной формы рекомендуемое значение $P(A)$ лежит в диапазоне $1/10 \ldots 1/40$.

Обсуждения и выводы

В настоящей работе на основе экспериментальных данных и теоретических расчетов построена модель рассеивания образа при сканировании для непрерывного случая. Предложена и проверена на практике функция

вероятности совпадения прообразов двух отсканированных образов символов. Попутно получены интересные и полезные факты искажения символов при сканировании. Описанные результаты вместе с использованием разработанного программного обеспечения являются инструментом для проверки гипотезы о совпадении или различии отсканированных символов, а так же для дальнейших исследований в этой области.

Полученную функцию вероятности можно использовать в прикладных задачах распознавания текстов кириллицы, латиницы и, вероятно, других алфавитов в равной степени; в частности в задаче кластеризации коллекции символов.

Дополнительный интерес вызывают исследования по нормировке изображения символов по яркости и контрастности внутри ограничивающего прямоугольника одного символа — такой прием позволил бы уменьшить искажающее влияние недорогих моделей сканеров, в частности неравномерная яркость в пределах одной страницы.

Перспективно проведение исследований поведения предложенной меры в случае наличия на странице с произвольными различными шрифтами — т. е. проверить, насколько хорошо разделяются одинаковые символы, напечатанные разными шрифтами.

Список литературы

1. *Арлазаров В. Л., Славин О. А.* Алгоритмы распознавания и технологии ввода текстов в ЭВМ // Информационные технологии и вычислительные системы. 1996. № 1
2. *Славин О. А.* Распознавание атрибутов текстовых символов // Сб. трудов ИСА РАН «Документооборот. Концепции и инструментарий». 2004. С. 142–150.
3. *Арлазаров В. Л., Славин О. А., Котович Н. В.* Адаптивное распознавание // Информационные технологии и вычислительные системы. 2002. № 4. С. 11–23.