

Распознавание символов на основе инвариантных моментов графических изображений*

Д.Ю.Васин¹, А.В.Аратский²

dm04@list.ru

¹НИИ Прикладной математики и кибернетики

Нижегородский Государственный Университет им.Н.И.Лобачевского;

²Факультет ВМК Нижегородского Государственного Университета им. Н.И.Лобачевского

В докладе рассмотрен алгоритм формирования эффективных признаков и решающих правил распознавания символов на основе инвариантных относительно масштабирования моментов на базе низкоуровневой структурированной модели описания бинарного растрового изображения.

Ключевые слова: распознавание с самообучением, оценка сходства, алгоритм распознавания.

The character recognition based on invariant moments graphics*

D. Yu. Vasin¹, A. V. Aratsky²

¹Research Institute of Applied Mathematics and Cybernetics,

Lobachevsky State University of Nizhni Novgorod

²Student of the Faculty of Lobachevsky State University of Nizhni Novgorod

The report considered the algorithm of formation of effective signs and decision rules based on character recognition are invariant with respect to scale points on the basis of a low-level structured model describing binary bitmap.

Keywords: recognition with self-learning, assessment of the similarity, algorithm recognizers.

Одной из основных проблем, возникающих при решении задачи распознавания образов, и существенно влияющих как на эффективность распознавания, с одной стороны, так и на его качество, с другой, является выбор системы эффективных признаков классификации. В работе рассматривается формирование решающих правил в задаче распознавания символов, находящихся в стандартной горизонтальной ориентации, но допускающих растяжение/сжатие по горизонтали/вертикали отдельных символов, на базе низкоуровневой структурированной модели описания бинарного растрового изображения (БРИ) [1]. Пример изображения текста для распознавания приведен на рисунке 1.

Пример текста для
алгоритма
распознавания

Рис. 1

Моментные характеристики изображения.

Моментные характеристики нашли широкое при-

Работа выполнена и опубликована при финансовой поддержке РФФИ, гранты 13-07-00521, 13-07-12211, 15-07-20347.

менение во многих задачах обработки цифровых изображений [1]. Рассчитываемые на их основе моментные инварианты стали важнейшим инструментом для распознавания образов независимо от их особого положения, ориентации, угла обзора и других изменений. Основным достоинством моментных инвариантов являются нечувствительность их к различным геометрическим преобразованиям (сдвиг, поворот, масштабирование) [1]. В данной работе рассматриваются нормированные моменты – инвариантные относительно масштабирования. Момент — это характеристика контура изображения, объединённая (суммированная) со всеми пикселями контура. Момент $m_{p,q}$ для непрерывного изображения $f(x, y)$ определяется как:

$$m_{p,q} = \iint x^p y^q f(x, y) dx dy \quad (1)$$

в случае дискретного растрового изображения

$$m_{p,q} = \sum \sum x^p y^q f(x, y), \quad (2)$$

где p, q — мощность, на которой соответствующий компонент взят в сумме с другими отображенными. Если функция $f(x, y)$ рассматривается как бинарное изображение, тогда момент m_{00} вычисляет площадь чёрных пикселей этого изображения. Для вычисления характеристик инвариантных относительно сдвига, необходимо центрировать относительно средних значений $\bar{x} = \frac{m_{10}}{m_{00}}, \bar{y} = \frac{m_{01}}{m_{00}}$ — центр тяжести фигуры.

Центральный момент определяется как:

$$\mu_{p,q} = \sum \sum (x - \bar{x})^p (y - \bar{y})^q f(x, y).$$

Доказано, что все центральные моменты выражаются через обычные, по формуле:

$$\mu_{p,q} = \sum_{i=0}^p \sum_{j=0}^q c_p^i c_s^j (-\bar{x})^{(p-i)} (-\bar{y})^{(q-j)} m_{ij}, \quad (3)$$

где $C_n^k = \frac{n!}{k!(n-k)!}$.

Для реализации инвариантности относительно масштабирования центральные моменты нормируют. Нормированные моменты вычисляются следующим образом:

$$\eta_{p,q} = \frac{\mu_{pq}}{\mu_{00}^{1+\frac{p+q}{2}}}, \quad \eta_{p,q} = \frac{m_{pq}}{m_{00}^{1+\frac{p+q}{2}}}. \quad (4)$$

В первом случае нормированный момент независим относительно масштабирования и сдвига, во втором – только относительно масштабирования.

Для поворота также есть соответствующие формулы разных порядков, вычисляющийся по нормированным моментам.

Инварианты бинарного изображения, представленного в штриховом виде. Основной упор в работе был направлен на нахождение инвариантных характеристик изображений символов в штриховом виде относительно растяжения/сжатия. В работе рассматривались следующие инварианты: по количеству горизонтальных и вертикальных штрихов. Рассмотрим эталонное изображение буквы Б (слева) и его же растянутое в $n = 2$ раз по горизонтали см. рисунок 2.

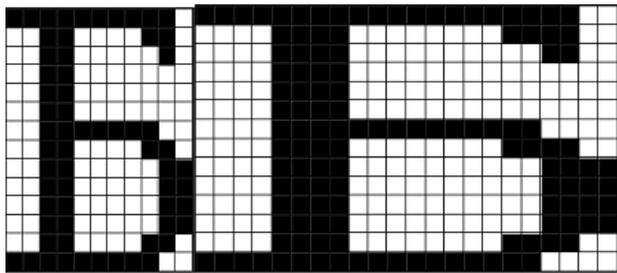


Рис. 2

Пусть w и h – соответственно ширина и высота оригинального изображения, а W и H – соответственно ширина и высота растянутого изображения. Очевидно, что между ними есть связь вида: $W = n \times w$, $H = h$. Так же обозначим dw и dh (dW и dH) – количество горизонтальных и вертикальных штрихов соответствующего изображения.

Можно заметить, что при растяжении по горизонтали сохранилась не только высота, но и количество штрихов по горизонтали. В тоже время при растяжении по ширине в n раз, у изображения образуется $(n - 1) \times w$ пустых столбцов пикселей которые надо заполнить, исходя из уже заполненных.

Очевидно, что вертикальные штрихи будут дублированы n раз. При не целых значениях n имеет смысл говорить о примерных соотношениях. Эти рассуждения аналогичны при масштабировании по вертикали. Соответственно инвариантные характеристики масштабированного штрихового изображения: $w/dh = a = const$, $h/d = b = const$. Однако на практике такая характеристика может довольно сильно колебаться. Не говоря о том, что одной такой характеристики недостаточно для полного отделения классов друг от друга.

Инвариант штриха относительно масштабирования. У штриха можно найти характеристику, не меняющуюся при масштабировании изображения. Рассмотрим горизонтальный штрих $d = \{i_d, s_d, f_d\}$ изображения размером w на h . Ясно, что при растяжении изображения по горизонтали, штрих также растягивается и отношение его длины к ширине сохраняется. Рассматривая растяжение по вертикали можно заметить, что отношение длины отрезка $[0, i_d]$ к высоте также сохраняется. Приведём эти нормированные значения к одному в виде угла Γ , см. рисунок 3.

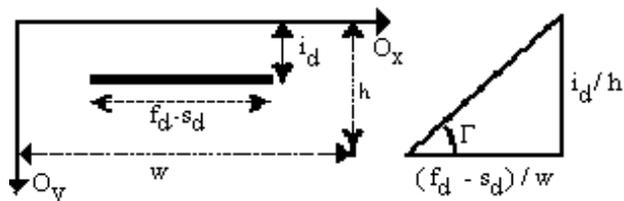


Рис. 3

Соответственно для штриха инвариантный коэффициент угла γ_d вычисляется по формуле: $\gamma_d = \arctan\{(i_d/h)(w/(f_d - s_d))\}$. Для получения инвариантной характеристики целого изображения D_n можно взять среднее арифметическое для γ по всем n его штрихам: $\bar{\gamma} = \frac{1}{n} \sum_{d \in D_n} \gamma_d$. Эмпирические наблюдения показали, что даже в большом растяжении по вертикали (в 5 раз и более) погрешность изменения не превышает сотых долей радиана. Однако эта характеристика как дискриминантный признак использоваться не может, т.к. на разных буквах она принимает очень близкие значения.

Приведённые примеры обладают слишком большой вычислительной сложностью. Они требуют полной обработки изображений, при этом не давая существенного выигрыша в выходном качестве распознавания.

Однако, используя моментные характеристики изображений, можно получить эффективные в вычислительном плане признаки, которые образуют достаточное признаковое пространство, для отделения классов друг от друга.

Использование ключевых строк и столбцов. Очевидно, что средний столбец пикселей растянутого относительно центра изображения остается без изменений. То же справедливо для центральной строки изображения при растяжении по вертикали. Это значит, что центральный столбец/строку изображения можно рассматривать как некоторую постоянную величину. Для этого необходимо найти некоторую числовую характеристику, которая однозначно ставится в соответствие соответствующей полоске пикселей или строке штрихов. Для этого воспользуется формулой *нормированных моментов*. Рассмотрим общий подход для центральной строки, аналогичные рассуждения для центрального столбца см. рисунок 4.

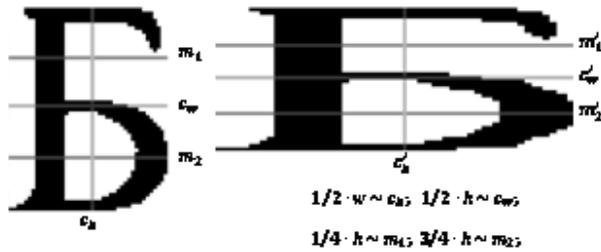


Рис. 4

Пусть высота и ширина изображения буквы соответственно h и w . Рассмотрим строку с номером $h/2$ и вычислим для неё числовую характеристику C_w^1 , которую можно считать хэш-функцией от строки. Аналогично можно вычислить значение C_w^2 , у растянутого изображения. Для столбцов аналогичные характеристики обозначим C_h^1, C_h^2 . Если вычисляемая характеристика инвариантна относительно масштабирования, то имеет смысл их сравнивать. По получившимся характеристикам получим признаковое описание объекта, в виде вектора вещественных чисел (c_w, c_h) . Тогда объект можно рассматривать как точку в признаковом пространстве \mathbf{R}^2 . Однако, не для всех букв достаточно двух значений. Например, у букв Y и X центральные полосы по горизонтали и вертикали будут похожими. В таком случае можно провести дополнительные полосы m_1 и m_2 , выбирая строки, как $1/4$ и $3/4$ высоты. Тем самым расширить признаковое пространство до \mathbf{R}^4 , и увеличить точность распознавания. Главный плюс такого подхода в том, что вычисляется момент не всего изображения, а лишь его небольшой части, а получаемые характеристики, для разных букв разные, что обеспечивает пересекающиеся множества классов. Рассмотрим подробнее, как производится расчёт моментов.

Нормированный момент штриховой строки. При работе со штриховым изображением также можно использовать центральные и нормированные мо-

менты. Заметим, что вычисление всех инвариантов опирается на вычисления простых моментов. Тогда, создав формулу для вычисления моментов штрихового изображения эквивалентную (2), формулы (3), (4) для центральных и нормированных моментов останутся справедливыми.

Разберём несколько случаев для монохромного штрихового изображения. Рассмотрим случай горизонтальных штрихов вида $d = \{i_d, s_d, f_d\}$ где i_d — номер строки по x , s_d, f_d — начало и конец штриха по y .

1. m_{00} — есть площадь чёрных пикселей. Для штрихового формата, очевидно, нужно подсчитать площадь всех штрихов D_n . Высота горизонтального штриха 1 пиксель, длина $f_d - s_d$:

$$m_{00} = \sum_{d \in D_n} (f_d - s_d).$$

2. m_{p0} — в данном случае каждый пиксель в штрихе умножается на x^p , что есть константные значения для горизонтального штриха равное i_d :

$$m_{00} = \sum_{d \in D} (f_d - s_d) i_d^p.$$

3. m_{0q} — в данном случае каждый пиксель в штрихе умножается на y^q , что есть изменяющееся от s_d к f_d значение для горизонтального штриха:

$$m_{0q} = \sum_{d \in D} \sum_{j=s_d}^{f_d} j^q.$$

4. m_{pq} — объединение рассмотренных случаев:

$$m_{pq} = \sum_{d \in D} (i_d^p \sum_{j=s_d}^{f_d} j^q) = \sum_{d \in D} \sum_{j=s_d}^{f_d} i_d^p j^q. \quad (5)$$

Поскольку штрих — одномерный кластер связанных чёрных пикселей бинарного изображения, для которых $f(x, y) = 1$, то данная формула эквивалентна дискретному моменту обычного растрового изображения. Следовательно, инвариантные моменты справедливы и для штрихового формата.

Так как основная идея состоит в использовании лишь опорных строк или столбцов изображения, то высота/ширина строки/столбца будет всегда равна 1 пикселю. И вычисляемый момент будет зависеть только от одного порядка, т.к. другая координата равна константе (при $q = 0$, это будет 1). Пусть задана строка штрихов D , тогда строковый момент вычисляем по формуле:

$$M_p(D) = \sum_{d \in D} \left(\int_0^1 \int_{s_d}^{f_d} x^p y^q dx dy \right) \Rightarrow \frac{1}{p+1} \sum_{d \in D} (f_d^{p+1} - s_d^{p+1}), \text{ при } q = 0. \quad (6)$$

Далее будет подразумевать под моментом строки (или столбца) именно выражение (6). Момент строки назывём горизонтальным, а столбца вертикальным.

Соответственно через моменты выражаем центр тяжести строки (столбца): $\bar{x} = M_p/M_0$. Справедлива и формула для нормированного момента. Однако тестирование показало, при сильном масштабировании значение нормированного момента также начинает отклоняться. Это связано с тем, что расчёт моментов ведётся в абсолютных величинах (координатах пикселей). Расчёт моментов в относительных координатах (отношение координат начала и конца штриха к общей длине полосы), даёт минимальные колебания.

$$\tilde{M}_p(D) = \frac{1}{p+1} \sum_{d \in D} \left(\left(\frac{f_d}{|D|} \right)^{p+1} - \left(\frac{s_d}{|D|} \right)^{p+1} \right)$$

В (2) $|D|$ — это длина изображения для горизонтального момента, и высота — для вертикального. Нормированный момент:

$$N_p(D) = \left(\frac{M_p}{M_0^{p+1}} \right)^{\frac{1}{p+1}}$$

На рисунке 5. представлены все нормированные вертикальные (слева) и горизонтальные (справа) моменты изображения эталонной буквы Б.

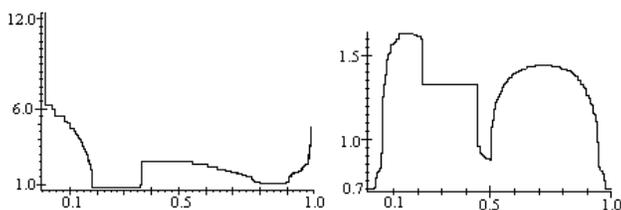


Рис. 5

Пространство признаков. В качестве набора признаков выберем нормированные вертикальные и горизонтальные моменты. Оставшийся вопрос в их количестве и номере строк (столбца) для подсчёта может быть очень вариативным. Будем исследовать некоторые выбранные фиксированные случаи. Определим для случая вертикальных моментов (аналогично для горизонтальных) основные:

Центральный момент изображения c_w — нормированный момент строки, взятой по номеру $h/2$.

Двойной момент изображения $\{a_w^1, a_w^2\}$ — нормированные моменты строк, взятых по номерам $h/3$ и $2/3$.

Тройной момент изображения $\{b_w^1, b_w^2 = c_w, b_w^3\}$ — нормированные моменты строк взятых по номерам $h/4, h/2$ и $3h/4$.

Исходя из сказанного, пара множеств и будет определять признаковое описание объектов и эталонов классов. Всего будет от 2 ($\{c_w, c_h\}$) до 6 ($\{b_w^1, c_w, b_w^3, b_h^1, c_h, b_h^3\}$) моментов. Поскольку признаковое пространство есть не что иное, как множество $R_n = \{x \in \mathbf{R}^n : x_i \geq 0, i = 1, 2, \dots, n\}$ каждый класс представляется точкой в пространстве. Алгоритм классификации объекта, в таком случае сводится к поиску ближайшего эталона по выбранной метрике.

Чем больше размерность признакового описания, там больше вероятность разнесения классов друг от друга. Тестирования с разными векторами моментов показало, что при 6-мерном векторе классы в достаточной степени отдаляются друг от друга. Однако, такое отдаление очень сильно зависит от используемого шрифта, размера и качества исходного документа. В случае с двумерным вектором моментов довольно часто ситуация когда несколько классов оказываются в большой близости друг от друга. На рисунке 6 представлено множество эталонов в двумерном признаковом пространстве.

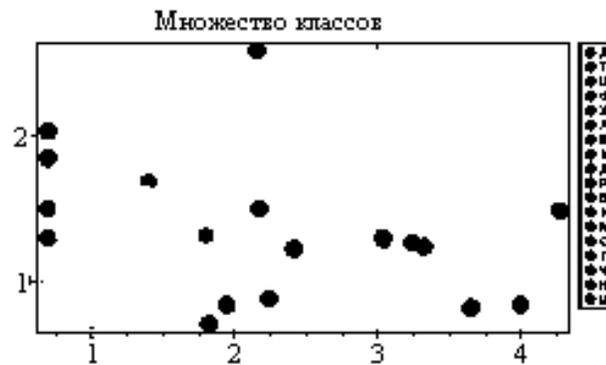


Рис. 6

В показанном распределении классы букв О и Г находятся в близости друг к другу. Такое распределение классов может

Расширения пространства признаков. К сожалению, нормированный строковый момент может давать похожие значения для разных полос. Это связано с тем, что по факту вычисляется площадь интеграла степенной функции на некотором промежутке, ограниченным рассматриваемым штрихом. Нетрудно подобрать разные по длине и положению штрихи, но одинаковые по значению нормированного момента. Что бы избежать такой ситуации можно помимо моментов хранить ещё дополнительную информацию по полосе. В частности можно хранить центр тяжести полосы, получаемые по формуле $X = N_p/N_0$, при этом размерность признакового описания возрастает в два раза.

Опишем поэтапно работу алгоритма распознавания.

Этап подготовки изображения. На данном этапе изображения разбиваемся на связные области чёрных пикселей, каждая из которых является буквой. Под выделением связных компонент понимаем присвоение уникальной метки каждому объекту изображения – букве. Каждую компоненту будем описывать меткой и описанной вокруг буквы прямоугольной областью. Отметим, что данная операция является наиболее времязатратной. Помимо выделения связных компонент, разобьем их по общей структуре текста на строки и слова.

Алгоритм организации структуры теста. На входе имеем общий список из связных компонент (в виде информации о расположении прямоугольной области на изображении и метки).

1. Отсортируем компоненты по возрастанию координаты у их центров прямоугольной области.
2. Выбираем первую компоненту из общего списка и смотрим пересечение горизонтальной линии, проходящей через центр компоненты с другими компонентами. Будем считать, что все попавшие компоненты расположены на одной строке. Для них:
 - (а) Отсортируем центра компонент по возрастанию координаты X .
 - (б) Пройдя по отсортированному списку, найдём расстояние между их крайними границами прямоугольников. Если оно больше некоторого заданного среднего значения, будем считать это пробелом.
 - (в) Запишем соответствующую структуру строки в список меток, хранящий информацию о структуре текста, где метка = 1 означает пробел, а метка = 2 — переход на новую строку.
3. Удалим попавшие в одну строку компоненты из общего списка и повторим шаг 2, пока не рассмотрим все компоненты.

Этап обучения. Обучение производится в интерактивном режиме, при этом определяется, какие моменты (центральный, двойной, тройной) будут вычислены по вертикали и горизонтали выбираются эталоны. Как и было обговорено выше, будем проводить вычисление признаков для выбранного эталона по горизонтали (вертикали) при $p = 2$:

Центральный $w = N_p(D)$ и его дополнение $\bar{X}_c = N_p/N_0$ в строке (столбце) D под номером $h/2(w/2)$.

Двойной, $a_w^i = N_p(D_i)$, $i = 1, 2$ и их дополнение $\bar{X}_a^i = N_p/N_0$ в строках (столбцах) с номерами $h/3(w/3)$ и $2h/3(2w/3)$.

Тройной, $b_w^i = N_p(D_i)$, $i = 1, 2, 3$ и их дополнение $\bar{X}_a^i = N_p/N_0$ в строках (столбцах) с номерами $h/4(w/4)$, $2h/4(2w/4)$ и $3h/4(3w/4)$.

Различные варианты пар нормированных моментов по горизонтали и вертикали будут образовывать признаковое пространство.

Этап распознавания. Основной и наиболее быстрый этап в работе алгоритма. Каждый образец буквы есть точка в пространстве $R_n\{x \in \mathbf{R}^n: x_i \geq 0, i = 1, 2, \dots, n\}$. При этом для каждого образца имеется дополнительный вектор из центров тяжести, представимый точкой в пространстве $C_n\{x \in \mathbf{R}^n: 0 \leq x_i \leq 1, i = 1, 2, \dots, n\}$. Введём δ -окрестность вокруг точки отвечающей эталону.

Возможны два варианта действия:

- при попадании образца в δ -окрестность эталона считать, что образец однозначно классифицирован и использование дополнительного пространства C_n не требуется («притягивающий» вариант);
- при попадании образца в δ -окрестность эталона считать, что образец претендует на класс эталона и требовать дополнительной проверки, используя для этого пространство C_n («отталкивающий» вариант).

Использование дополнительного пространства C_n необходимо контролировать с помощью специального коэффициента нагрузки $\alpha > 0$. Это связано с тем, что компоненты в C_n принадлежат отрезку $[0, 1]$, а в R_n отрезку $[0, \infty]$.

Алгоритм распознавания символа. На входе алгоритма растровая связная компонента и набор эталонов.

1. Для каждого эталона найдём Δ -характеризующую близость к эталону.
 - (а) Для вектора момента найдём близость по метрике $\rho = \sqrt{\sum_{i=1}^n (m_i - m_i^E)^2}$, где $m_i = c, a^1, a^2, b^1, b^2, b^3$ в зависимости от выбранных моментов.
 - (б) Делаем проверку $\rho < \delta$ для «отталкивающего» варианта ($\rho > \delta$ для «притягивающего» варианта):
 - i. Если так, рассчитываем близость векторов пространства по метрике $\tau = \alpha \sqrt{\sum_{i=1}^n (C_i - C_i^E)^2}$.
 - ii. Иначе берём $\tau = 0$.
2. Находим $\Delta = \rho + \tau$ для каждого эталона и выбираем класс эталона с минимальным Δ и определяем символ, который принадлежит данному классу.

Тестирование показало, что притягивающий и отталкивающий подходы благодаря регулирования значений δ и α во многом эквиваленты. Следует заметить, что при большой размерности R_n дополнительная нагрузка и вовсе не нужна, так как признаковое пространство в достаточной степени разделяет классы друг от друга.

Алгоритм распознавания символа запускается на каждой связной компоненте после чего на основе полученных результатов формируется текст той структуры, которую определили на этапе обработки.

На базе разработанных алгоритмов создано оригинальное ПО. Практические тесты проводились на растровом бинарном изображении текста (шрифт Times New Roman, 16 кегль). Процент правильно распознанных знаков от 77 до 97. При этом, очевидно, что чем больше размерность пространства, тем более эффективен алгоритм, а центральные моменты дают лучший результат, чем двойные моменты. Так же тестирование показало, что при небольшом повороте около $\pm 10^\circ$ эффективность распознавания (при размерности \mathbf{R}^6) сохраняется на уровне 85-95%.

Считая, что допустимое количество ошибок не должно превышать 10–15% от общего числа символов текста, признаковое пространство \mathbf{R}^6 и более, с дополнительной нагрузкой в виде центров тяжести штрихов будет оптимальным решением. Одна-

ко получаемая эффективность относится к хорошо отфильтрованным изображениям. При тестировании на старых документах с плохой фильтрацией при попадании полосы на белые пятна нормированные моменты будут подсчитаны неверно, и эффективность распознавания по сравнению с качественным изображением упадет в 2 раза (с 97-100% до 50-55%).

Литература

- [1] Васин Д.Ю., Васин Ю.Г., Громов В.П. Структурное описание растровых данных // VI Всероссийская конференция «Методы и средства обработки сложной графической информации». Тезисы. Н.Новгород. 2001. С.21–23.
- [2] Wong R. Scene matching with invariant moments // AComputer Graphics and image processing, 1978.– Vol.8. – pp.16–24.
- [3] Васин Д.Ю. Image Analysis by Moments // The Department of Electrical and Computer Engineering The University of Manitoba, 1993.