УДК 681.327.12.001.362

МОДЕЛЬ ЦЕПОЧКИ СИМВОЛОВ В ЗАДАЧАХ РАСПОЗНАВАНИЯ НА ОСНОВЕ СТРУКТУРНЫХ МЕТОДОВ

В.А. ПРЫТКОВ

Белорусский государственный университет информатики и радиоэлектроники
П.Бровки, 6, Минск, 220013, Беларусь

Поступила в редакцию 23 ноября 2012

Предложена модель цепочки символов, которая позволяет определить степень соответствия формальной грамматике. Модель может применяться в задачах распознавания образов и обработки изображений на основе синтаксических методов в случае, когда цепочка символов не порождается грамматикой.

Ключевые слова: формальная грамматика, цепочка символов, распознавание образов, классификация, мера сходства, структурное описание.

Введение

На сегодняшний день одним из наиболее динамично развивающихся направлений в распознавании образов и обработке изображений является текстурный анализ. Толчком к развитию направления сегментации и распознавания текстурных изображений послужили работы Харалика и Лавса [1, 2]. Тем не менее, хотя и ведутся активные работы по определению текстурных признаков, выявлению текстурирующего элемента, описанию текстур и т.д., до сих пор не существует четкого формального определения текстуры [3–6]. В случае текстурной сегментации используют, как правило, фильтры, статистические признаки, нейронные сети, скрытые марковские модели, фрактальный анализ и др.

В данной работе будем придерживаться следующего определения текстуры: текстура – связная область элементов цифрового изображения с различными яркостями, визуально передающая характер поверхности объекта исходной сцены [7]. В работах [3, 5] отмечается, что текстура является свойством соседства. Наименьшую область изображения текстурного типа, передающую характер и основные особенности текстуры, принято называть текстурирующим элементом (текстоном, текселем).

Одной из основных работ по применению теории формальных грамматик в распознавании образов является [8]. Однако грамматики здесь используются в основном для описания геометрических свойств объектов и описания сцен. Теория формальных грамматик к распознаванию текстурных изображений применяется достаточно редко. В первую очередь, это связано со сложностью передачи свойств двумерного объекта (цифрового изображения) с помощью аппарата, предназначенного для работы с одномерными объектами (цепочками символов). Тем не менее, общие идеи такого подхода и некоторые классы используемых грамматик предложены в [9–11].

В работах [12–14] предложен метод для решения задач текстурного анализа изображений, основанный на синтаксическом описании текстур. Описание строится в виде правил формальной грамматики, определяющих отношения соседства для однородных элементов изображения. Текстурирующий элемент рассматривается в качестве множества таких однородных элементов, а правила грамматики образуются путем обхода контуров однородных областей и анализа границ. Задача распознавания сводится к задаче о принадлежности входной цепочки заданной грамматике, решаемой достаточно тривиально для классов контекстно-свободных и регулярных грамматик. Такой подход позволяет использовать достаточно простые критерии

однородности при выполнении сегментации и строить иерархические текстурные модели, однако обладает повышенной чувствительностью к ошибкам на этапах сегментации и кластеризации. Отметим, что при построении грамматики фактически используются кольцевые (циклические) строки. Это свойство позволяет использовать данный алгоритм и в генетике, при сравнении циклических ДНК [15]. Использование стохастических грамматик, детально описанных в [8], позволяет снизить влияние шумовых эффектов на результат.

В том случае, когда цепочка символов не соответствует грамматике, задачу распознавания и классификации будет значительно проще решить при наличии меры соответствия цепочки символов данной грамматике. В то же время метод построения грамматики путем обхода контуров, так же как и стохастические грамматики, не позволяет определить степень соответствия цепочки символов правилам грамматики в случае неполного соответствия. В [14] предлагается учитывать степень соответствия через количество верно распознанных нетерминалов правой части правила грамматики либо через длину контура и верно свернутых терминалов. В настоящей работе предлагается модель на основе кусочно-постоянной функции, позволяющая определить степень соответствия входной цепочки символов заданной грамматике.

Теоретический анализ

Формальная грамматика G определяется через множество терминальных символов (терминальный словарь) T, множество нетерминальных символов (нетерминальный словарь) N, множество правил P вида $\alpha\rightarrow\beta$, где α и β – цепочки символов, $\alpha, \beta \in (N\cup T)^*$ и целевого символа грамматики S. Аппарат формальных грамматик имеет хорошо проработанную математическую базу и для классов контекстно-свободных и регулярных грамматик позволяет достаточно тривиально решать задачу о принадлежности входной цепочки заданной грамматике. Таким образом, если описать текстуру с помощью регулярной либо контекстно-свободной грамматики, задачи распознавания и классификации изображений текстур переходят в разряд задач о принадлежности входной цепочки заданной грамматике.

Текстурирующий элемент, в общем случае, может быть разбит на множество областей, к которым применимы более строгие критерии однородности по сравнению с текстоном в целом. Для таких однородных областей достаточно просто может быть построено пространство признаков, по которому их можно разбить на классы.

Обозначим текстуру через нетерминал I, и поставим в соответствие каждому классу однородных областей текстуры свой нетерминал A_i, где i – номер класса. Тогда текстура в первом приближении может быть описана следующим правилом: $I\rightarrow A_1I \mid A_2I \mid \ldots \mid A_nI \mid A_1 \mid A_2 \mid \ldots \mid A_n$, где n – количество классов однородных областей. Однако этого правила недостаточно для описания отношения соседства. Построим соответствующие правила следующим образом: выполним обход границы каждой из областей, например, по часовой стрелке, и в порядке прохождения смежных областей впишем соответствующие им нетерминалы в правую часть правила: $A_j\rightarrow B^1_{k1}B^2_{k2}\ldots B^m_{km}$, где m – количество смежных областей, $j = 1, 2, \ldots, n$, нетерминал B^i_k соответствует границе области k-класса, i – порядковый номер смежной области в последовательности, полученной при обходе границы, и, соответственно, $k1, k2, \ldots, km \in \{1, 2, \ldots, n\}$.

Учитывая, что в общем случае начало обхода может находиться в любой точке контура, последнее правило модифицируется следующим образом: $A_j\rightarrow B^1_{k1}B^2_{k2}\ldots B^m_{km} \mid B^1_{k1}B^2_{k2}\ldots B^m_{km}B^1_{k1} \mid B^2_{k2}\ldots B^m_{km}B^1_{k1} \mid B^2_{k2}\ldots B^m_{km}B^1_{k1}B^2_{k2} \mid \ldots \mid B^m_{km}B^1_{k1}B^2_{k2}\ldots \mid B^m_{km}B^1_{k1}B^2_{k2}\ldots B^m_{km}$. Подобное изменение позволяет учесть и шумовые эффекты вблизи граничных пикселей, а также построить инвариантное к повороту описание.

Для полноты описания в грамматику добавляются правила, позволяющие построить конечную цепочку терминальных символов: $B_k\rightarrow b_k \mid B_kb_k$. Здесь b_k – терминальный символ, соответствующий контурному пикселю, принадлежащему однородной области k-класса, $k = 1, 2, \ldots, n$.

В результате обучения для каждой текстуры будет построена своя грамматика, которая позволит выполнять на этапе распознавания проверку на точное соответствие входной цепочки заданной грамматике.

Предложенный подход не позволяет распознать цепочку в случае, если нет полного ее соответствия грамматике. Однако ошибки сегментации и классификации приводят к формиро-

ванию цепочек, которые либо не будут распознаны, либо будут распознаны неверно. Такие же цепочки формируются на границах текстурных областей, когда контур включает области других текстур, а также на границе изображения, когда часть контура вообще не имеет смежных областей.

Для обнаружения частичного совпадения правила грамматики требуют дополнения правилами вида $B_k \rightarrow E' \mid B_k E'$, $E' \rightarrow b_1 \mid b_2 \mid \ldots \mid b_n \mid e \mid \lambda$, где λ – пустая цепочка. Такое дополнение позволяет учесть как возможные включения любых других классов областей в контур класса B_k, так и полное отсутствие соседних областей. Отметим, что такие правила грамматики фактически приводят к полному перебору возможных сочетаний нетерминалов в правой части правил и к значительному увеличению объема вычислений и замедлению обработки.

Использовать для этой цели известные меры неполного сходства строк [15, 16], например, меры Хэмминга или Левенштейна, не представляется возможным. Мера Хэмминга основана на подсчете количества позиций, в которых символы различаются. Ее можно использовать только для строк одинаковой длины, и любой шумовой эффект во входной цепочке значительно повлияет на результат: так, добавление лишнего символа в начале строки и одновременно пропадание символа в конце строки может привести к нулевому результату. Мера Левенштейна (либо его модификация Дамерау-Левенштейна) рассчитывается как количество операций вставки, удаления и замены символов, необходимых для получения из одной строки второй. Вычислительная сложность этого алгоритма оценивается как $O(mn)$. Меры n-грамм основаны на подсчете количества совпадающих подстрок фиксированной длины (n-грамм).

Данные меры используют символ в качестве атомарного элемента цепочки, что для рассматриваемой задачи понижает точность решения, поскольку нетерминалы правой части правил представляют терминальные цепочки различной длины. Кроме того, эти подходы не учитывают циклический характер цепочек. Обобщение с учетом цикличности требует применения данных алгоритмов m раз для вычисления максимального совпадения каждой цепочки. Соответственно увеличится и их сложность. Так, сложность алгоритма на основе меры Левенштейна возрастет до $O(m^2 n)$.

Методика

Обратим внимание на тот факт, что построение цепочки символов выполняется путем обхода замкнутого контура однородной области, представляющей нетерминал левой части правила. Данный контур состоит из множества сегментов, граничащих с соседними однородными областями и каждому такому сегменту соответствует свой нетерминал правой части правила грамматики.

Пусть n – количество классов однородных областей, $j = 1, 2, \ldots, n$ – порядковый номер класса, m – количество смежных областей у текущей однородной области, $i = 1, 2, \ldots, m$ – порядковый номер смежной области в последовательности, нетерминал B^i_k соответствует i смежной области k-класса. Если каждому нетерминалу правой части правила поставить в соответствие числовое значение, равное порядковому номеру определяемого им класса, то каждое подмножество правил грамматики вида $A_j \rightarrow B^1_{k1}B^2_{k2}\ldots B^m_{km} \mid B^1_{k1}B^2_{k2}\ldots B^m_{km}B^1_{k1} \mid B^2_{k2}\ldots B^m_{km}B^1_{k1} \mid B^2_{k2}\ldots B^m_{km}B^1_{k1}B^2_{k2} \mid \ldots \mid B^m_{km}B^1_{k1}B^2_{k2}\ldots \mid B^m_{km}B^1_{k1}B^2_{k2}\ldots B^m_{km}$ можно описать кусочно-постоянной функцией

$$A_j(x) = \begin{cases} k_1, \text{если } 0 \le x < x_1, \\ k_2, \text{если } x_1 \le x < x_2, \\ \ldots \\ k_m, \text{если } x_{m-1} \le x < x_m, \end{cases} \qquad (1)$$

где $x_1, x_2, \ldots, x_{m-1}$ – границы смежных областей, и x_m – длина контура.

Для возможности сравнения цепочек нормализуем функцию по длине контура:

$$A_j(x) = \begin{cases} k_1, \text{если } 0 \le x < \dfrac{x_1}{x_m}, \\ k_2, \text{если } \dfrac{x_1}{x_m} \le x < \dfrac{x_2}{x_m}, \\ \dots \\ k_m, \text{если } \dfrac{x_{m-1}}{x_m} \le x < 1, \end{cases} \tag{2}$$

где x = [0, 1). Подобная нормализация делает модель инвариантной к масштабу. Определим функцию $E_{i,j}$ двух цепочек A_i и A_j как

$$E_{i,j}(A_i, A_j) = \begin{cases} 1, \text{ если } A_i = A_j, \\ 0, \text{если иначе.} \end{cases} \tag{3}$$

Тогда в качестве меры соответствия можно использовать $\varepsilon_{i,j} = \int_{x=0}^{1} E_{i,j}(A_i(x), A_j(x))\, dx$.

Такая модель не инвариантна к повороту. Инвариантности можно достичь, вычисляя меру соответствия с учетом сдвига A_i и A_j относительно друг друга. Будем рассматривать функцию A_j как периодическую:

$$A_j(x) = \begin{cases} k_1, \text{если } t \le x < \dfrac{x_1}{x_m} + t \\ k_2, \text{если } \dfrac{x_1}{x_m} + t \le x < \dfrac{x_2}{x_m} + t \\ \dots \\ k_m, \text{если } \dfrac{x_{m-1}}{x_m} + t \le x < t + 1. \end{cases}, \; t = 0, \; \pm 1, \; \pm 2, \; \dots, \tag{4}$$

Пусть u – сдвиг цепочки A_j относительно A_i , u = [0, 1). Определим функцию соответствия $M_{i,j}$ цепочек следующим образом: $M_{i,j}(u) = \int_{x=0}^{1} E_{i,j}(A_i(x), A_j(x+u))\, dx$, тогда мера соответствия $\varepsilon_{i,j} = \max(M_{i,j}(u))$.

В таком виде модель уже можно использовать, однако точность вычислений при этом зависит от шага дискретизации.

Из определения функции $E_{i,j}$ следует, что она является кусочно-постоянной. Поскольку интеграл от постоянной величины является линейной функцией, то функция соответствия $M_{i,j}$ (u) будет являться кусочно-линейной функцией, при этом можно показать, что она непрерывна. Следовательно, функция соответствия $M_{i,j}$ (u) достигает максимума в точках изменения угловых коэффициентов на концах соответствующих интервалов. Определим эти точки. Рассмотрим функции A_1 и A_2, определенные следующим образом:

$$A_1 = \begin{cases} k_1, \text{ если } t \le x < x_1 + t, \\ k_2, \text{ если } x_1 + t \le x < x_2 + t, \\ k_3, \text{ если } x_2 + t \le x < t + 1, \end{cases} A_2 = \begin{cases} k_4, \text{ если } t \le x < x_3 + t, \\ k_2, \text{ если } x_3 + t \le x < x_4 + t, \\ k_5, \text{ если } x_4 + t \le x < t + 1, \end{cases} \begin{matrix} 0 \le x_1, x_2, x_3, x_4 \le 1, \\ t = 0, \; \pm 1, \; \pm 2, \; \dots \end{matrix}. \tag{5}$$

Допустим, что $x_1 < x_2 < x_3 < x_4$ и $x_2 - x_1 < x_4 - x_3$. Тогда функции имеют вид:

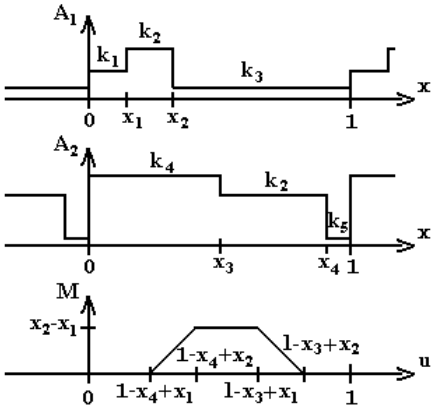

Рис. 1. Результат сравнения двух цепочек

Очевидно, что $M_{i,j}(u)$ начнет возрастать в точке $1 - x_4 + x_1$, достигнет максимума $x_2 - x_1$ в точке $1 - x_4 + x_1$, начнет убывать в точке $1 - x_3 + x_1$ и вновь станет равной 0 в точке $1 - x_3 + x_2$. Рассмотренный пример касается случая, когда функции имеют по одному интервалу с одинаковым значением. Результаты легко обобщаются. При этом обратим внимание на тот факт, что существенными являются границы только тех пар интервалов, значения функций на которых совпадают.

Пусть X – множество точек разрыва функции A:

$$X(A) = \{x \mid \frac{x_1}{x_m} + t, \frac{x_2}{x_m} + t, \ldots, \frac{x_{m-1}}{x_m} + t\},\ t = 0,\ \pm 1,\ \pm 2,\ \ldots \tag{6}$$

Тогда множество $U_{i,j}$ точек изменения угловых коэффициентов функции $M_{i,j}(u)$ определяется следующим образом:

$$U_{i,j} = \{u_{i,j} \mid x_i - x_j,\ x_i > x_j, \mid x_i - x_j \mid < 1, x_i \in X(A_i), x_j \in X(A_j), A_i(x_i) = A_j(x_j)$$
$$\text{либо } A_i(x_{i-1}) = A_j(x_j) \text{ либо } A_i(x_i) = A_j(x_{j-1})\}. \tag{7}$$

Альтернативы здесь учитывают прерывность функции в точке разрыва, а ограничение единичным диапазоном не приводит к потере общности, так как функции A_i и A_j периодичные с периодом 1. В этом случае мера соответствия будет вычисляться следующим образом: $\varepsilon_{i,j} = \max(M_{i,j}(U_{i,j}))$.

Полученная модель инвариантна к масштабу и повороту, и дает точное решение. Оценим ее вычислительную сложность. Алгоритм включает в себя вычисление множества U и, в соответствии с количеством элементов данного множества, расчета функции M и выбора максимального значения: $f(m) = 2(m+n) + |U|k(m+n) + |U|$.

Здесь m и n – количество символов в сравниваемых цепочках, k – некоторый коэффициент, учитывающий количество операций по вычислению частичной суммы при расчете значения функции M. Мощность множества U не превысит mn для произвольных цепочек и $mn/2$, если соседние символы не могут быть одинаковыми. Последнее условие следует из метода построения соответствующей грамматики. Таким образом, вычислительная сложность предложенного алгоритма не выше сложности алгоритма на основе меры Левенштейна $O(m^2n)$, что является хорошим показателем.

Экспериментальная часть

Рассмотрим результаты использования модели на текстуре, однородные области которой представляют собой квадраты с длиной стороны, равной k пикселей. Допустим, что входное изображение полностью идентично эталонному за исключением одной области, которая не соответствует исходной текстуре.

Использование грамматик приведет к нулевому результату как для собственно искаженной области, так и для смежных с ней областей, поскольку для них входная цепочка будет

содержать нетерминал *J*, отсутствующий в правой части правил эталонной грамматики. Предложенная методика не учитывает соответствие нетерминала левой части правила, поэтому центральный элемент будет иметь меру соответствия, равную 1. Для диагональных элементов длина несовпадающего участка равна 1 пикселю, соответственно мера равна $1–1/4\cdot(k+1)$. Для остальных элементов длина несовпадающего участка равна *k*, соответственно мера $1–k/4\cdot(k+1)$.

Рис. 2. Пример расчета меры соответствия

Результаты и их обсуждение

В ходе работы предложена модель, которая позволяет вычислять меру неполного сходства двух цепочек, при этом в отличие от типовых методов вычисления меры сходства строк в текстах, она учитывает циклический характер сравниваемых цепочек и может использоваться в алгоритмах распознавания текстур на основе синтаксических методов. Предложенная модель инвариантна к повороту и масштабированию.

В реальных задачах на слабоподобных участках количество элементов множества *U* существенно снижается по сравнению с максимальной оценкой, а на близких фрагментах – приближается к ней, поскольку данный параметр зависит от количества одинаковых символов в строке (нетерминалов в правой части правила). Алгоритм Левенштейна ведет себя противоположным образом. Соответственно, предложенную модель лучше использовать для малоподобных цепочек.

Модель не учитывает значение нетерминала левой части правила, что в некоторых случаях может привести к снижению точности результата. Аналогично, модель не учитывает меру подобия несовпадающих сегментов, соответствующих нетерминалам правой части правил. Разработка соответствующих методов повысит точность результата, однако вычислительная сложность алгоритма очевидным образом возрастет.

После незначительной адаптации, заключающейся в использовании непериодической функции, модель можно использовать для сравнения строк текста.

Заключение

В работе предложена модель циклической цепочки символов, которая позволяет определить меру сходства двух цепочек. Модель оптимизирована для использования в задачах распознавания образов и обработки изображений на основе синтаксических методов и повышает точность распознавания на границах изображения и на участках с шумовыми эффектами.

MODEL OF CHARACTER STRING IN PATTERN RECOGNITION BASED ON STRUCTURAL METHODS

V.A. PRYTKOV

Abstract

The proposed model of the character string allows to determine its degree of compliance of the formal grammar. The model can be used for pattern recognition and image processing based on syntactic techniques when the character string is not derived by the grammar.

Список литературы

1. *Haralick R.M., Shanmugan K., Dinstein I.* // IEEE Trans. Syst. Man. Cybern. 1973. Vol. 3. P. 610–621.
2. *Laws K.L.* Textured Image Segmentation. PhD thesis. Los Angeles, 1980.
3. *Jain A.K., Karu K.* // Lecture notes in computer science. 1995. Vol. 974. P. 3–10.
4. *Noriega L., Westland S.* // Proc. of 6th Intern. Conf. «Pattern Recognition and Information Processing». Minsk, 2001. P. 121–125.
5. *Zhou F., Feng J., Shi Q.* // Proc. of 6th Intern. Conf. «Pattern Recognition and Information Processing». Minsk, 2001. P. 41–45
6. *Абламейко С.В., Лагуновский Д.М.* Обработка изображений: технология, методы, применение. Минск, 2000.
7. *Старовойтов В.В.* Локальные геометрические методы цифровой обработки и анализа изображений. Минск, 1997.
8. *Фу К.* Структурные методы в распознавании образов. М., 1977
9. *Гонсалес Р., Вудс Р.* Цифровая обработка изображений. М., 2006.
10. *Форсайт Д., Понс Ж.* Компьютерное зрение. Современный подход. М., 2004.
11. *Сулейменов Е.Р.* // Докл. 9 Всеросс. конф. М., 1999. С. 230–231
12. *Прытков В.А.* // Доклады БГУИР. 2008. № 4. С. 115–120.
13. *Yarmolik A.P., Bartashevich Y.A., Prytkov V.A.* // Proc. of the 10th Intern. Conf. «Pattern Recognition and Information Processing». Minsk, 2009. P. 112–114
14. *Прытков В.А., Барташевич Ю.А., Лукашевич М.М.* // Матер. V междунар. конф.-форума «Информационные системы и технологии». Минск, 2009. С. 172–175
15. *Гасфилд Д.* Строки, деревья и последовательности в алгоритмах. СПб, 2003.
16. *Gonzalo Navarro* // ACM Computing Surveys. 2001. Vol. 33, № 1. P. 31–88.