

Язык гранично-скелетного представления бинарных изображений

Иван Рейер, Мария Петровцева
Вычислительный центр им. А.А. Дородницына РАН
Москва, Россия
reyer@forecsys.ru
maria@hnservice.ru

Аннотация

В работе приводится описание XML-производного языка, предназначенного для описания непрерывного гранично-скелетного представления дискретных бинарных изображений.

Данное представление основано на аппроксимации дискретного бинарного изображения непрерывным бинарным изображением и использовании границы и скелета полученного непрерывного изображения. Такой подход обеспечивает возможность комплексного исследования граничного и скелетного представлений, необходимость которого часто возникает при обработке и анализе бинарных изображений.

Предлагаемый язык является универсальным средством описания гранично-скелетного представления, обеспечивает возможности сохранения представления на диске и обмена информацией между различными приложениями.

В работе также представлено программное обеспечение, позволяющее создавать гранично-скелетные описания бинарных изображений на предлагаемом языке.

Ключевые слова: дискретное бинарное изображение, непрерывное гранично-скелетное представление, XML.

1. ВВЕДЕНИЕ

Дискретное бинарное изображение – разбиение множества всех точек евклидовой плоскости с целочисленными координатами на два подмножества: конечное, называемое объектом, и бесконечное, называемое фоном. Эта модель удобна для представления большого класса реальных изображений, полученных в результате сканирования и бинаризации. В частности, изображений печатных и рукописных документов [1].

Для анализа формы дискретного бинарного изображения широко используются граничное и скелетное представления изображения.

Традиционный подход к построению таких представлений основан на применении определенных алгоритмов непосредственно к дискретному изображению [2, 3]. И граница, и скелет при этом представляются также в виде дискретного изображения. Достоинством такого подхода являются относительные простота и эффективность алгоритмов построения скелета. Однако в этом случае приходится адаптировать границу и скелет, изначально непрерывные конструкции, определяемые для непрерывных областей, к геометрии регулярных точечных пространств, в терминах которой формулируются растровые алгоритмы.

Был предложен подход [4] к обработке дискретного бинарного изображения, основанный на аппроксимации

дискретного изображения непрерывным без использования растровых преобразований.

Непрерывное бинарное изображение – это подразбиение евклидовой плоскости, индуцированное конечным множеством непересекающихся замкнутых ограниченных областей. Объектом в этом случае является множество точек, принадлежащих этим областям, а фоном – все остальные точки плоскости.

В предложенном подходе дискретное изображение аппроксимируется многоугольной фигурой (граница этой фигуры используются в качестве граничного представления изображения), затем на основе этой фигуры строится непрерывное скелетное представление дискретного бинарного изображения.

Данный подход дает возможность эффективного комплексного анализа границы и скелета изображения. Необходимость такого анализа часто возникает при решении прикладных задач, например задачи штриховой сегментации изображения рукописного слова с учетом гладкости границы [5, 6].

Для работы с таким гранично-скелетным представлением изображения была разработана [7] структура данных и динамическая библиотека, предназначенная для построения и получения информации о различных элементах представления.

Однако при работе с данной библиотекой гранично-скелетное представление изображения хранится во внутреннем формате приложения, что затрудняет возможности сохранения полученного гранично-скелетного представления, а также его использования в других приложениях и на других платформах.

Существующие форматы векторной графики [8, 13] ориентированы прежде всего на визуализацию объектов, а не на описание структуры и взаимосвязей между элементами.

Таким образом, возникает задача разработки универсального способа описания гранично-скелетного представления, обеспечивающего возможность хранения информации в памяти компьютера или передачи между различными приложениями.

В данной работе представлено описание XML-производного языка, который соответствует предъявляемым требованиям. Также авторами разработано программное обеспечение, генерирующее гранично-скелетные описания бинарных растровых изображений на предлагаемом языке.

Работа выполнена при поддержке Российского Фонда фундаментальных исследований (РФФИ), проект 02-01-00667.

2. НЕПРЕРЫВНОЕ ГРАНИЧНО-СКЕЛЕТНОЕ ПРЕДСТАВЛЕНИЕ ДИСКРЕТНЫХ БИНАРНЫХ ИЗОБРАЖЕНИЙ

Рассмотрим подробнее выше упомянутое гранично-скелетное представление дискретного бинарного изображения, а также структуру данных для реализации этого представления.

2.1. Общая информация

2.1.1. Граничное представление

Для построения граничного представления дискретное бинарное изображение аппроксимируется непрерывным – многоугольной фигурой, разделяющей все точки объекта и фона. В предложенном подходе использовалась аппроксимация многоугольником [9], имеющим *минимальный* периметр (рисунок 1). Заметим, что такое решение не является единственным и возможно использование любого алгоритма полигональной аппроксимации.

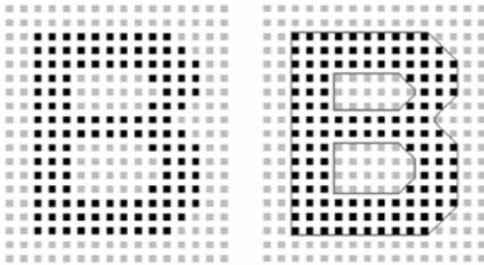


Рисунок 1: Дискретное бинарное изображение и его аппроксимация многоугольной фигурой.

Граница аппроксимирующей фигуры разбивает плоскость дискретного изображения на совокупность связанных компонент: фигурных (содержащих все черные точки) и фоновых (содержащих все белые точки). Полученная многоугольная фигура используется в качестве граничного представления дискретного бинарного изображения.

Границу, определяемую многоугольной фигурой, удобно задавать множеством вершин и открытых сегментов (сегментов без концевых точек), являющихся вершинами и сторонами фигуры. Такие точки и отрезки будем называть *сайтами*: *сайтами-точками* и *сайтами-сегментами* (рисунок 2).

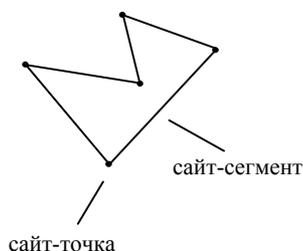


Рисунок 2: Представление границы в виде множества сайтов

2.1.2. Скелетное представление

Далее с использованием многоугольной фигуры строится так называемое скелетное представление непрерывного бинарного изображения.

Скелетом [10] плоской замкнутой области называется множество ее внутренних точек, имеющих не менее двух ближайших граничных точек. Скелет также можно определить на основе понятия пустого круга.

Пустой круг плоской замкнутой области – это круг на евклидовой плоскости, не содержащий внутри себя точек границы фигуры.

Максимальный пустой круг – это пустой круг, не содержащийся ни в одном другом пустом круге, не равном себе самому.

Скелет плоской замкнутой области представляет собой геометрическое место точек, являющихся центрами максимальных пустых кругов. На рисунке 3 приведен пример скелета многоугольной фигуры.

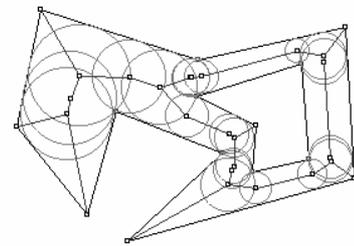


Рисунок 3: Скелет многоугольной фигуры

Скелет многоугольной фигуры можно рассматривать как плоский граф [11]. Этот граф образуют вершины – центры максимальных пустых кругов, касающихся трех или более сайтов (или одного сайта-вершины), соединенные ребрами – серединные осями некоторых пар сайтов. *Серединной осью* пары сайтов мы называем максимальное связанное множество центров максимальных пустых кругов, касающихся этих сайтов (то есть множество точек многоугольной фигуры, равноудаленных от двух сайтов). Серединная ось может быть прямолинейным сегментом для двух сайтов-сегментов или двух сайтов-точек или участком параболы для сайта-точки и сайта-сегмента (рисунок 4).

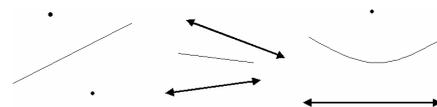


Рисунок 4: Серединная ось разных пар сайтов

Множества вершин и ребер скелета аппроксимирующей многоугольной фигуры, а также множество максимальных пустых кругов задают скелетное представление бинарного изображения.

2.2. Структура данных гранично-скелетного представления

В библиотеке [7] информация о гранично-скелетном представлении организована следующим образом. Основу

структуры представления составляют два списка, в одном из которых хранятся все фигурные компоненты, а в другом – все фоновые.

Для каждой фигурной и фоновой компоненты создается скелет, ссылка на внешний контур ее границы и список ссылок на внутренние контуры.

Все контуры хранятся в отдельном списке. Каждый контур хранится в виде списка сайтов-точек (т.е. вершин многоугольника) и списка сайтов-сегментов (соединяющих вершины открытых сегментов). Вершины задаются своими координатами, а ребра – ссылками на инцидентные вершины.

Скелет компоненты хранится в виде списка вершин и списка ребер так называемого скелетного графа.

Для каждой вершины скелета (рисунок 5) хранятся соответствующий ей максимальный пустой круг (а именно, координаты его центра и радиус), список ссылок на сайты, которых касается данный круг, и список ссылок на инцидентные вершине ребра.

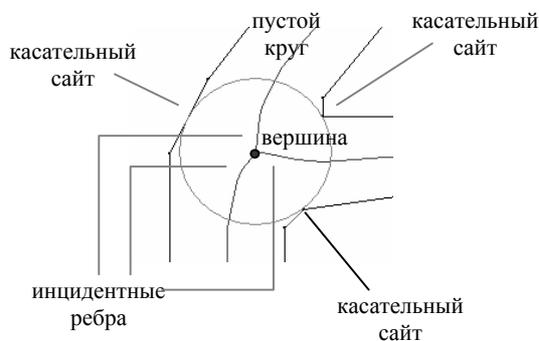


Рисунок 5: Вершина, соответствующие ей пустой круг, касательные сайты и инцидентные ребра

Для каждого ребра хранятся ссылки на инцидентные ему вершины. Отметим, что, зная касательные сайты вершин ребра, можно легко найти ту пару сайтов, для которых данное ребро является серединой осью. Тем самым определяется тип ребра (прямолинейный сегмент или участок параболы). По координатам этой пары сайтов и концевых точек ребра можно в явном виде получить уравнение ребра. А, зная радиусы и координаты центров пустых кругов вершин, можно определить радиус пустого круга с центром в любой точке ребра. Таким образом, описываемая структура данных является достаточно информативной.

Кроме того, каждому элементу представления присваивается уникальный идентификатор.

3. ИСПОЛЬЗОВАНИЕ XML ДЛЯ ОПИСАНИЯ ГРАНИЧНО-СКЕЛЕТНОГО ПРЕДСТАВЛЕНИЯ ДИСКРЕТНОГО ИЗОБРАЖЕНИЯ

При выборе средства хранения гранично-скелетного представления авторы руководствовались следующими критериями:

1. возможность описания структуры представления, взаимосвязей между элементами;
2. независимость от приложений;

3. простота переноса информации (возможность использования на разных аппаратных платформах);
4. простота реализации визуализации.

Всем этим требованиям удовлетворяет XML.

XML (*EX*tensible Markup Language, "расширяемый язык разметки") – рекомендованный W3C (World Wide Web Consortium) язык разметки [12], применяемый не только для разработки Интернет-приложений, но и в обычных приложениях, в том числе для хранения и обработки структурированных данных в едином формате.

XML обеспечивает эффективный поиск нужных элементов, изменение, дополнение данных, а также простоту проверки корректности записи данных. Это позволяет устанавливать единый стандарт на структуру документов, содержимым которых могут быть самые различные данные. Таким образом, его можно использовать при построении сложных информационных систем, в которых важен обмен информацией между различными приложениями.

XML-документ представляет собой обычный текстовый файл. Это обеспечивает возможность использования на различных аппаратных платформах а также создания и модификации с помощью простых средств редактирования текстов.

Многие объектно-ориентированные языки программирования имеют множество классов и библиотек функций для работы с XML-документами, что облегчает написание генератора и анализатора документа с описанием гранично-скелетного представления.

Также XML предоставляет возможность простой реализации визуализации гранично-скелетного представления за счет трансформации полученного XML файла в формат SVG (стандартный XML-производный язык описания векторной графики) [13, 14].

3.1. Общие сведения

XML – это язык разметки документов, описывающий целый класс объектов данных, называемых XML-документами. Основой разметки документа являются теги.

Тэги XML не заданы в этом языке с самого начала, мы можем ввести произвольное множество своих тегов, тем самым, сформировав *новый язык разметки*. XML используется в качестве средства для описания грамматики XML-производных языков и контроля правильности составления документов.

В XML-документе обычно присутствуют следующие составляющие: элементы, атрибуты, информационные объекты и комментарии.

Элементы данных - это структурные единицы XML-документа. Например, заключая слова «описание контура» в тэги <contour>, </contour>, мы определяем непустой элемент, называемый <contour>, содержимым которого является «описание контура»:

<contour>описание контура</contour>

В общем случае в качестве содержимого элементов могут выступать как просто какой-то текст, так и другие, вложенные, элементы документа, секции CDATA, инструкции по обработке, комментарии, - т.е. практически любые части XML- документа.

Любой элемент должен состоять из начального, конечного тэгов и данных, заключенных между ними. В случае если

элемент не имеет содержимого, он называется пустым, и начальный и конечные тэги могут быть объединены в один, например, <component/>.

Набором всех элементов, содержащихся в документе, задается его структура, и определяются все иерархическое соотношения. Плоская модель данных превращается с использованием элементов в сложную иерархическую систему с множеством возможных связей между элементами, например:

```
<picture>
  <background_components>
    <component>...</component>
    <component>...</component>
    <component>...</component>
  </background_components>
  <figure_components>
    <component>...</component>
    <component>...</component>
  </figure_components>
</picture>
```

Атрибуты Если при определении элементов необходимо задать какие-либо параметры, уточняющие его характеристики, то имеется возможность использовать атрибуты элемента. Атрибут - это пара «название» = «значение», которую надо задавать при определении элемента в начальном тэге. Пример:

```
<point id="p0" x="11" y="11"/>
<site_vertex id="v0" point="p0"/>
<site_segment id="e0" point1="p0" point2="p1"/>
```

3.2. Определение структуры данных гранично-скелетного представления в терминах XML

Для корректного и более эффективного использования разработанного XML-производного языка нам необходимо точно задать все правила построения документа с описанием представления.

Для описания структуры XML документа и данных входящих в него будем использовать *DTD (Document Type Definition)*.

3.2.1. Описание языка гранично-скелетного представления с помощью DTD

В основу структуры языка положим описанную в п.2.2. структуру гранично-скелетного представления.

Для описания атрибутов элементов будем использовать следующие типы: CDATA – символьные данные; ID – определяет уникальный идентификатор элемента в документе.

Также заметим, что ссылки на какие-либо элементы в XML можно осуществлять с помощью атрибутов типа IDREF, IDREFS (один или несколько, разделенных пробелами, уникальных идентификаторов определенных в этом документе элементов), например:

```
<!ATTLIST component
  id ID #REQUIRED
  ext_contour IDREF #IMPLIED
  int_contours IDREFS #IMPLIED>
```

#REQUIRED, #IMPLIED – параметры, определяющие, обязательно ли должен присутствовать данный атрибут или его можно опустить.

Определение типа документа (DTD) будет выглядеть следующим образом:

- Объявление XML: указывается язык разметки документа XML, номер его версии 1.0 и используемая в документе кодировка UTF-8.

```
<?xml version="1.0" encoding="UTF-8"?>
```

- Описание корневого элемента picture. Дочерние элементы background_components, figure_components и contours обязательные, должны присутствовать один раз.

```
<!ELEMENT picture (background_components,
  figure_components, contours)>
```

- Элемент background_components должен содержать один и более элементов component, элемент figure_components может быть пустым или содержать элементы component. Элемент contour может быть пустым или содержать элементы contour.

```
<!ELEMENT background_components (component+)>
<!ELEMENT figure_components (component*)>
<!ELEMENT contours (contour*)>
```

- Для элемента component обязательно должен быть указан идентификатор id типа ID, могут быть указатель на внешний контур (ext_contour) и список указателей на внутренние контуры (int_contours). Дочерние элементы: nodes, bones.

```
<!ELEMENT component (nodes, bones)>
<!ATTLIST component
  id ID #REQUIRED
  ext_contour IDREF #IMPLIED
  int_contours IDREFS #IMPLIED>
```

- Элемент nodes (bones) может быть пустым или содержать любое число элементов node (bone).

```
<!ELEMENT nodes (node*)>
<!ELEMENT bones (bone*)>
```

- Элемент node – пустой. Все атрибуты обязательные.

- id – идентификатор;
- dischalfplain – определяет, является ли круг полуплоскостью;
- disc_x, disc_y, disc_r – если круг не полуплоскость, то это координаты центра и радиус круга; если полуплоскость, то коэффициенты уравнения $Ax+Bx+C=0$, задающего границу полуплоскости;
- incidental_bones – ссылок на инцидентные кости;
- tangent_sites – список ссылок на сайты, которых касается данный пустой круг.

```
<!ELEMENT node EMPTY>
<!ATTLIST node
  id ID #REQUIRED
  dischalfplain (yes | no) #REQUIRED
  disc_x CDATA #REQUIRED
  disc_y CDATA #REQUIRED
  disc_r CDATA #REQUIRED
  incidental_bones IDREFS #REQUIRED
  tangent_sites IDREFS #REQUIRED>
```

- Элемент bone – пустой. Все атрибуты обязательные.

- id – идентификатор;
- finite – определяет, является ли кость конечной;
- node1, node2 – ссылки на инцидентные узлы

```

<!ELEMENT bone EMPTY>
<!ATTLIST bone
  id ID #REQUIRED
  finite (yes | no) #REQUIRED
  node1 IDREF #REQUIRED
  node2 IDREF #REQUIRED>

```

У элемента `contour` должны быть указаны `id`; `intorext`: атрибут определяет, является ли контур внутренним (значение атрибута `int`) или внешним (`ext`) для соответствующей фигурной компоненты. Дочерние элементы – пара `site_vertex`, `site_segment`, повторяющаяся не менее одного раза.

```

<!ELEMENT contour ((site_vertex, site_segment)+)>
<!ATTLIST contour
  id ID #REQUIRED
  intorext (int | ext) #REQUIRED>

```

Элемент `site_vertex` пустой, необходимые атрибуты: `id` (идентификатор), `x`, `y` – координаты точки. Элемент `site_segment` также пустой, необходимые атрибуты: `id` и `point1`, `point2` – ссылки на соответствующие концевые точки.

```

<!ELEMENT site_vertex EMPTY>
<!ATTLIST site_vertex
  id ID #REQUIRED
  x CDATA #REQUIRED
  y CDATA #REQUIRED>
<!ELEMENT site_segment EMPTY>
<!ATTLIST site_segment
  id ID #REQUIRED
  point1 IDREF #REQUIRED
  point2 IDREF #REQUIRED>

```

3.2.2. Пример XML-документа

Итак, используя данный язык, мы можем описать гранично-скелетное представление.

Рассмотрим пример XML-документа для следующего дискретного бинарного изображения (рисунок 6).

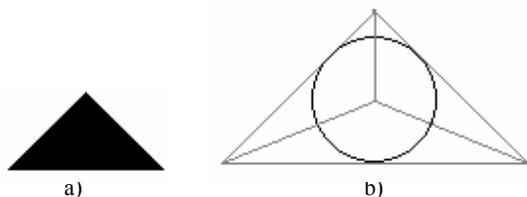


Рисунок 6: Дискретное бинарное изображение (а) и соответствующее ему гранично-скелетное представление (b, увеличено в 2 раза)

Для данного изображения описание структуры гранично-скелетного представления в терминах XML будет выглядеть следующим образом (для простоты будем подробно описывать только фигурные компоненты, а фоновые перечислим).

```

<?xml version="1.0" encoding="UTF-8"?>
<picture>

<background_components>
  <component id="c0" int_contours="k0">...</component>
</background_components>

```

```

<figure_components>
  <component id="c1" ext_contour="k0">
    <nodes>
      <node id="n4" dischalfplain="no" disc_x="53"
disc_y="82" disc_r="0" incidental_bones="b5"
tangent_sites="e1"/>
      <node id="n8" dischalfplain="no" disc_x="53"
disc_y="59,1" disc_r="16,1" incidental_bones="b5 b7 b8"
tangent_sites="e0 e1 e2"/>
      <node id="n7" dischalfplain="no" disc_x="14"
disc_y="43" disc_r="0" incidental_bones="b7"
tangent_sites="v0"/>
      <node id="n9" dischalfplain="no" disc_x="92"
disc_y="43" disc_r="0" incidental_bones="b8"
tangent_sites="e0"/>
    </nodes>
    <bones>
      <bone id="b5" finite="yes" node1="n4" node2="n8"/>
      <bone id="b7" finite="yes" node1="n7" node2="n8"/>
      <bone id="b8" finite="yes" node1="n8" node2="n9"/>
    </bones>
  </component>
</figure_components>

```

```

<contours>
  <contour id="k0" intorext="ext">
    <site_vertex id="v0" x="14" y="43"/>
    <site_segment id="e0" point1="v0" point2="v1"/>
    <site_vertex id="v1" x="92" y="43"/>
    <site_segment id="e1" point1="v1" point2="v2"/>
    <site_vertex id="v2" x="53" y="82"/>
    <site_segment id="e2" point1="v2" point2="v0"/>
  </contour>
</contours>
</picture>

```

Листинг 1: Пример XML-документа, содержащего описание гранично-скелетного представления для изображения, представленного на рисунке 6.

4. ВИЗУАЛИЗАЦИЯ С ПОМОЩЬЮ SVG

Как упоминалось в пункте 3, использование XML для описания гранично-скелетного представления позволяет реализовать визуализацию представления за счет преобразования полученного XML-документа в SVG, которую удобно использовать для наглядного просмотра информации об исходном изображении.

Рассмотрим, как будет выглядеть SVG-файл для примера, приведенного в пункте 3.2.2.

Будем отрисовывать не все элементы, а только контуры, ребра скелета и пустые круги для фигурных компонент. Полученный SVG-документ выглядит следующим образом (листинг 2).

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.0//EN"
  "http://www.w3.org/TR/2001/REC-SVG-
  20010904/DTD/svg10.dtd">
<svg>
  <title>Border-skeletal representation of an image</title>

```

```

<defs>
  <style type="text/css"><![CDATA[
    *{fill: none; stroke-width: 0.5}
    .white{stroke:white;}
    .bone{stroke:red;}
    .contour{stroke:green;}
    rect{fill:#DDDDDD}
  ]]></style>
</defs>
<rect x="0" y="0" width="100%" height="100%" />
<g id="int_component_c1" transform="scale(2,-2)
translate(0,-92)">
  <circle cx="53" cy="59" r="16" class="white" />
  <line x1="53" y1="82" x2="53" y2="59" class="bone" />
  <line x1="14" y1="43" x2="53" y2="59" class="bone" />
  <line x1="53" y1="59" x2="92" y2="43" class="bone" />
</g>
<g id="contour_k0" transform="scale(2,-2) translate(0,-92)">
  <polygon class="contour" points="14 43,92 43,82 59" />
</g>
</svg>

```

Листинг 2. Пример SVG-документа, реализующего визуализацию гранично-скелетного представления.

Результат визуализации в приложении, поддерживающем формат SVG, будет выглядеть следующим образом (рисунок 7, изображение увеличено в два раза):



Рисунок 7: Изображение гранично-скелетного представления в формате SVG

Кроме используемых в данном примере элементов формат SVG позволяет также изображать все остальные элементы (в том числе и идентификаторы), что предоставляет нам универсальное средство для просмотра гранично-скелетного представления изображения.

5. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ ГЕНЕРАЦИИ XML-ДОКУМЕНТОВ

Для генерации XML-документов, содержащих описание гранично-скелетного представления для заданного дискретного бинарного изображения, и дальнейшего преобразования их в формат SVG авторами было разработано программное обеспечение, использующее рассмотренную в п.2 динамическую библиотеку.

На рисунке 8 приведен пример работы программы. Представлены: выбранная бинарная картинка, полученное XML-описание гранично-скелетного представления и SVG-изображение этого представления.

6. ЗАКЛЮЧЕНИЕ

Предложенный в работе язык является эффективным инструментом описания гранично-скелетного

представления дискретного бинарного изображения. Отметим, что такое представление может быть получено с помощью алгоритмов полигональной аппроксимации и построения скелета многоугольника, отличных от используемых авторами. Также данный язык позволяет описать скелетное представление, полученное в результате "стрижки", то есть выделения в скелете аппроксимирующего многоугольника части, отражающей фундаментальные свойства структуры образа и удаления несущественных ветвей.

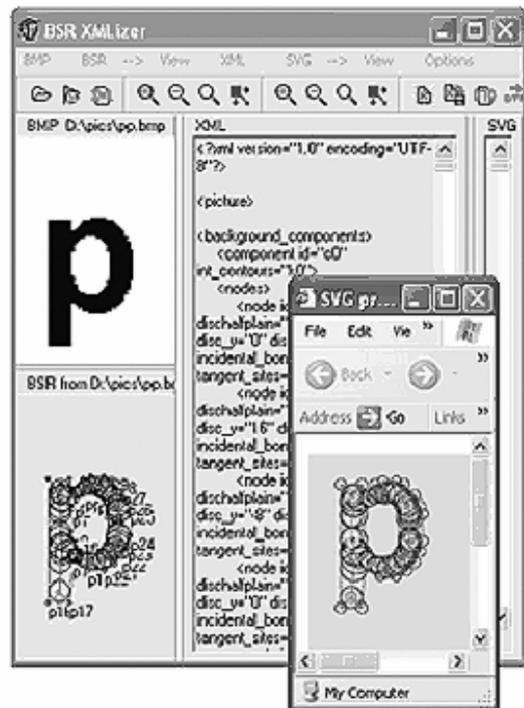


Рисунок 8. Пример работы программы

Дополнение программных средств построения гранично-скелетного представления бинарных изображений XML-производным языком описания такого представления предоставляет новые возможности: использование построенного гранично-скелетного представления в других приложениях, на других платформах, Интернет-приложениях; простую и эффективную визуализацию.

7. БЛАГОДАРНОСТИ

Авторы выражают благодарность Л.М. Местецкому, предложившему постановку задачи.

8. ЛИТЕРАТУРА

- [1] G. Nagy. *Twenty years of document image analysis in PAMI*. IEEE Trans. PAMI, 22(1), Jan 2000.
- [2] L. Lam, S.-W. Lee, C. Y. Suen. *Thinning methodologies: A comprehensive survey*. IEEE Trans. PAMI, 14(9), Sept 1992.
- [3] В. В. Старовойтов. *Локальные геометрические методы цифровой обработки и анализа изображений*. Минск, 1997.

[4] Л. М. Местецкий. «Векторизация бинарных растровых изображений на основе аппроксимации». Доклады X Всероссийской конференции «Математические методы распознавания образов» (ММРО-10), Москва, 2001, с. 249-252.

[5] И. А. Рейер. *Сегментация штрихов и их соединений при распознавании рукописного текста. Труды 9-й международной конференции по компьютерной графике и машинному зрению Графикон-99*, Москва, 1999, с. 151-155.

[6] L. M. Mestetskii, I. A. Reyer, T. W. Sederberg. *Continuous approach to segmentation of handwritten text. Proceedings of 8th International Workshop on Frontiers in Handwriting Recognition (WFHR-8)*, 2002, pp. 440-445.

[7] Л. М. Местецкий, И. А. Рейер. «Программное обеспечение для работы с непрерывным гранично-скелетным представлением дискретных изображений». *Интеллектуализация обработки информации: тезисы докладов международной конференции, Симферополь, 200*, стр. 137-138.

[8] Д. Мюррей, У. ван Райпер. *Энциклопедия форматов графических файлов*. – К.: Издательская группа BHV, 1997.

[9] Л. М. Местецкий. *Непрерывный скелет бинарного растрового изображения. Труды между. конф. "Графикон-98"*, Москва, 1998.

[10] H. Blum. *A transformation for extracting new descriptors of shape. Models for the Perception of Speech and Visual Form*, pp.362-380. MIT Press, 1967.

[11] H. I. Choi, S. W. Choi, H. P. Moon. *Mathematical theory of medial axis transform. Pacific. J. of Math.*, 181(1), 1997.

[12] *Extensible Markup Language (XML) 1.0 (Second Edition) W3C Recommendation 6 October 2000*, Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler:
<http://www.w3.org/TR/REC-xml>.

[13] *Scalable Vector Graphics (SVG) 1.1 Specification W3C Recommendation 14 January 2003*:
<http://www.w3.org/TR/2003/REC-SVG11-20030114/>.

[14] J. David Eisenberg "SVG Essentials". *O'Reilly, First Edition February 2002*.

Авторы

Рейер Иван Александрович – аспирант ВЦ РАН.
E-mail: rever@forecsys.ru

Петровцева Мария Александровна – студентка факультета ВМиК МГУ.
E-mail: maria@hnservice.ru

Abstract

In the paper an XML-based language for description of continuous boundary-skeletal representation of discrete binary images is suggested.

The boundary-skeletal representation is based on approximation of a discrete binary image with a continuous binary image and usage of a boundary and a skeleton of the obtained continuous

image. This approach makes possible complex research of boundary and skeletal representations which is frequently necessary at binary images processing and analysis.

The language suggested is a universal tool for description of boundary-skeletal representation. It provides ways of representation storage on disk, information exchange between various applications, and easy visualization.

We also present software that allows users to create boundary-skeletal descriptions of binary images in the suggested language.

Keywords: *discrete binary image, continuous boundary-skeletal representation, XML.*