

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

ИЗВЕСТИЯ ВЫСШИХ УЧЕБНЫХ ЗАВЕДЕНИЙ

# ПРИБОРОСТРОЕНИЕ

ИЗДАНИЕ ГОСУДАРСТВЕННОГО ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ  
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ»

Журнал издается с января 1958 г.

ТОМ 52

ФЕВРАЛЬ 2009

№ 2

СПЕЦИАЛЬНЫЙ ВЫПУСК

## МЕТОДЫ И СРЕДСТВА ОБРАБОТКИ ИЗОБРАЖЕНИЙ

*Под редакцией доктора технических наук, профессора С. Г. Емельянова,  
доктора технических наук, профессора В. С. Титова*

### СОДЕРЖАНИЕ

ПРЕДИСЛОВИЕ .....	5
<b>МЕТОДЫ ЦИФРОВОЙ ОБРАБОТКИ ИЗОБРАЖЕНИЙ</b>	
Гридин В. Н., Бугаенко Е. И. Метод автоматического формирования панорамных изображений на основе аппарата нечеткой логики .....	7
Садыков С. С., Орлов А. А., Ермаков А. А. Методика обработки линейчатых образов на дефектоскопических снимках .....	11
Гордиенко В. В., Довгаль В. М., Пузына Р. А. Распознавание изображений с использованием хаотической трансформации эталонов и объектов.....	16
Гостев И. М., Подгорбунский А. Г. Распределение потоков графической информации в системах распознавания образов в реальном масштабе времени.....	20
Андрианов Д. Е., Садыков С. С., Фролов В. В. Формальное представление характеристик пространственно-распределенных объектов .....	25
Огнев А. В., Типикин А. П. Алгоритм инвариантного распознавания отпечатков пальцев по ключевым точкам .....	31
<b>ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА</b>	
Ватутин Э. И., Зотов И. В., Титов В. С. Выявление изоморфных вхождений R-выражений при построении параллельных алгоритмов логического управления.....	37
Мирошниченко С. Ю., Титов В. С. Параллельно-конвейерное устройство для векторизации аэрокосмических изображений земной поверхности.....	45
Титенко Е. А. Метод реконфигурации операционной части мультипроцессора структурного распознавания образов.....	52
Борзов Д. Б. Размещение параллельных подпрограмм в мультимикроконтроллерах с учетом отказов .....	55
Борзов Д. Б., Дюбрюкс С. А., Титов В. С. Метод объединения и разделения циклических участков последовательных наследуемых программ .....	60

## ОПТИКО-ЭЛЕКТРОННЫЕ УСТРОЙСТВА И СИСТЕМЫ

<b>Труфанов М. И.</b> Распознающие оптико-электронные устройства медицинской экспресс-диагностики.....	66
<b>Стрелкова А. Н., Труфанов М. И., Степченко А. А.</b> Математическая модель восстановления эндоскопических изображений .....	70
<b>Титов Д. В., Ширабакина Т. А.</b> Модуль цифровой коррекции дисторсии изображения .....	74
<b>Бехтин Ю. С., Соляков В. Н., Титов Д. В.</b> Сопряжение многоэлементного фотоприемного устройства с персональным компьютером измерительного стенда .....	78
SUMMARY .....	83

## SPECIAL ISSUE

# METHODS AND MAENS OF IMAGES PROCESSING

*By Edition S. G. Emeljanov, Doctor of Technical Sciences, Professor,  
V. S. Titov, Doctor of Technical Sciences, Professor*

## CONTENTS

INTRODUCTION .....	5
<b>DIGITAL IMAGES PROCESSING METHODS</b>	
<b>Gridin V. N., Bugaenko E. I.</b> Method of Automatic Forming of Panoramic Images Based On Fuzzy Logic .....	7
<b>Sadykov S. S., Orlov A. A., Ermakov A. A.</b> Method of Processing of the Band Images on Defectoscopy Scenes .....	11
<b>Gordienko V. V., Dovgal V. M., Puzyna R. A.</b> Image Recognition Method Based on Chaotic Transformation of Patterns and Images .....	16
<b>Gostev I. M., Podgorbunsky A. G.</b> Distribution of Flow of Graphik Information in Systems of Images Recognition in Real Time.....	20
<b>Andrianov D. E., Sadykov S. S., Frolov V. V.</b> Formal Representation of Characteristics of Spatially-Distributed Objects .....	25
<b>Ognev A. V., Tipikin A. P.</b> Invariant Recognition of Fingerprints Using Key Points Algorithm .....	31
<b>COMPUTING DEVICES</b>	
<b>Vatutin E. I., Zotov I. V., Titov V. S.</b> Checking up Isomorphic Inclusions of $R$ -Expressions in the Construction of a Set of Sections for Parallel Logic Control Algorithms.....	37
<b>Miroshnichenko S. Yu., Titov V. S.</b> Parallel-Pipeline Device for Earth Surfaces Aerospace Images Vectorization .....	45
<b>Titenko E. A.</b> Reconfiguration Method of Operational Part of Multiprocessor for Structural Recognition of Patterns.....	52
<b>Borzov D. B.</b> Parallel Subprograms Assignment in Multimicrocontrollers Taking into Account Rejection .....	55
<b>Borzov D. B., Dyubryuks S. A., Titov V. S.</b> Conflation and Splitting Method of Cyclic Parts of Sequential Inherit Programms.....	60

## OPTO-ELECTRONIC DEVICES AND SYSTEMS

<b>Truphanov M. I.</b> Recognition Opto-Electronic Device for Medical Fast Diagnostics .....	66
<b>Strelkova A. N., Truphanov M. I., Stepchenko A. A.</b> Mathematical Model of Endoscopic Images Restoration .....	70
<b>Titov D. V., Shirabakina T. A.</b> Correction Device of Image Distortion .....	74
<b>Bekhtin Yu. S., Solyakov V. N., Titov D. V.</b> Connecting the Multielement Photoreceivers to the Testing Table Based on Computer .....	78
SUMMARY .....	83

*Editor-in-Chief E. B. Yakovlev*

## ПРЕДИСЛОВИЕ

В настоящем специальном выпуске журнала приведены результаты исследований, проводимых в течение многих лет сотрудниками Курского государственного технического университета в содружестве с другими вузами России.

В соответствии с тематикой статей сборник разбит на три части.

В первом разделе „Методы цифровой обработки изображений“ рассмотрены метод автоматического формирования панорамных изображений, методика обработки линейчатых образов на дефектоскопических снимках, алгоритм инвариантного распознавания отпечатков пальцев, формальное представление характеристик пространственно-распределенных объектов.

Во втором разделе „Вычислительная техника“ рассматриваются вопросы объединения и разделения циклических участков последовательных наследуемых программ, представлен метод реконфигурации операционной части мультипроцессора, исследована проблема размещения параллельных подпрограмм в мультимикроконтроллерах с учетом возможных отказов.

В третьем разделе „Оптико-электронные устройства и системы“ наибольшее внимание уделено применению оптико-электронных устройств в медицинской экспресс-диагностике, для векторизации аэрокосмических изображений земной поверхности, цифровой коррекции дисторсии изображения. Практический интерес представляет статья, посвященная исследованию многоэлементных фотоприемников с использованием персонального компьютера.

Результаты исследований, представленные в выпуске, обсуждались и получили положительную оценку на Международной научно-технической конференции „Оптико-электронные приборы и устройства в системах распознавания образов, обработки изображений и символической информации (Распознавание—2008)“.

*Ректор Курского государственного технического университета,  
доктор технических наук, профессор  
С. Г. ЕМЕЛЬЯНОВ*

*Заведующий кафедрой вычислительной техники,  
доктор технических наук, профессор  
В. С. ТИТОВ*

## INTRODUCTION

This special journal issue acquaints readers with the works of scientists and teachers of Kursk State Technical University and other Universities of Russia.

According to subjects of articles the issue content is divided into three parts.

In a first part „Digital image processing methods“ there are articles devoted to a method of method of receiving of panoramic images, a method of processing of the band images on the defectoscopy scenes, an invariant recognition of fingerprints algorithm, formal representation of characteristics of the spatially-distributed objects.

The second part „Computer engineering“ introduces reader to problems of a merging and splitting method of cyclic sites of the consecutive inherited programs, a reconfiguration method of the operational part of multiunit, a parallel subprograms assignment in multimicrocontrolles into account rejection.

In the third part „Opto-electronic devices and systems“ the major attention is given to opto-electronic devices for medical fast diagnostics, earth surface's aerospace images vectorization. The article, describing the multielement photoreceivers based on computer, can be mentioned as especially interesting.

Investigation results represented in the issue were discussed and achieved the favorable reception at International scientific conference „Opto-electronic device in pattern recognition systems, image processing and symbolic information (Recognition—2008)“.

*Rector of Kursk State Technical University,  
Doctor of Technical Sciences,  
Professor S. G. EMELJANOV*

*Chief of Computational Sciences Department  
of Kursk State Technical University,  
Doctor of Technical Sciences,  
Professor V. S. TITOV*

---

---

# МЕТОДЫ ЦИФРОВОЙ ОБРАБОТКИ ИЗОБРАЖЕНИЙ

---

---

УДК 680.5.01:621.384

В. Н. Гридин, Е. И. Бугаенко

## МЕТОД АВТОМАТИЧЕСКОГО ФОРМИРОВАНИЯ ПАНОРАМНЫХ ИЗОБРАЖЕНИЙ НА ОСНОВЕ АППАРАТА НЕЧЕТКОЙ ЛОГИКИ

Представлен метод получения панорамных изображений из множества цифровых изображений, имеющих одинаковые области, с использованием математического аппарата нечеткой логики.

*Ключевые слова:* панорамное изображение, обработка изображений, радиальная дисторсия, нечеткая логика.

В настоящее время панорамные изображения получают в основном двумя способами: с помощью специализированных устройств и с помощью программ, позволяющих объединять несколько изображений в одно, которое будет иметь более широкий угол обзора, чем исходные. Однако существующие подходы имеют такие недостатки, как низкая точность сформированных панорамных изображений, обусловленная влиянием аберраций оптических систем, высокая стоимость специализированного оборудования, а также сложность в использовании этого оборудования. Программные методы позволяют создавать панорамные изображения из видеопоследовательности или из множества изображений с одинаковыми областями.

Например, один из существующих методов создания панорамных изображений [1] включает следующие этапы:

- получение нескольких пересекающихся изображений рабочей сцены;
- определение положения каждого последующего изображения относительно предыдущего;
- коррекция искажений перспективы на всех изображениях;
- перебор возможных вариантов соединения изображений и выбор такого варианта, при котором совпадение одинаковых участков изображений будет наилучшим;
- определение значения маски для каждого пиксела изображений;
- получение панорамного изображения путем наложения изображений друг на друга с использованием вычисленных значений маски.

Таким образом, описанный метод требует участия пользователя, и поэтому не может быть применен в автоматических системах обработки изображений.

В методе получения панорамных изображений, представленном в работах [2, 3], добавлен этап коррекции геометрических искажений, в ходе которого минимизируются искажения положения объектов на одном изображении относительно положения этих же объектов на других изображениях. Серия фотографий выполняется одним фотоаппаратом, который поворачивают горизонтально на определенные углы, т.е. одно изображение получается развернутым относительно второго на некоторый угол. При совмещении этих изображений расстояния

между объектами могут быть различными, что приводит к искажению панорамных изображений. Коррекция заключается в проведении аффинных преобразований изображения на плоскости [4]. Их использование ведет к упрощению объединения перекрывающихся изображений и повышению качества полученной панорамы, но искажения, вызванные аберрациями оптических систем, данным методом не корректируются.

Существует также подход к формированию панорамных изображений из видеопоследовательности [5]. Метод основан на анализе изображений, которые извлекаются из видеопоследовательности, по ним определяется вектор движения; используя определенный вектор, а также другие параметры участков кадров, получают панорамное изображение. Использовать этот метод очень удобно для случаев, когда из видеопоследовательности необходимо получить фотографию с широким углом обзора.

Итак, при создании панорамных изображений возможно использовать несколько способов коррекции искажений перспективы, или геометрических, т.е. искажений, возникающих на одном изображении по отношению к другим, однако существующие способы не обеспечивают компенсации оптических аберраций, возникающих в изображениях из-за несовершенства оптических узлов. На качество формируемых панорамных изображений особенно большое влияние оказывает радиальная дисторсия. Это затрудняет выделение в нескольких изображениях одинаковых участков, а также создает определенные трудности при соединении нескольких перекрывающихся изображений. Кроме того, ослаблением влияния радиальной дисторсии можно добиться повышения качества панорамных изображений.

Метод автоматического формирования панорамных изображений, представленный в настоящей статье, включает в себя:

- коррекцию радиальной дисторсии;
- выделение контуров объектов на множестве изображений;
- распознавание идентичных объектов на различных изображениях с использованием математического аппарата нечеткой логики по определенной контурной информации с последующим объединением частей изображений для получения панорамного изображения.

Входными данными метода являются множества цифровых изображений с наличием одинаковых областей. После ввода множества изображений производится коррекция радиальной дисторсии с использованием методов [6, 7]. Далее пары изображений обрабатываются в порядке следования. Рассмотрим порядок действий, производимых для первой пары.

*Этап 1. Выделение контуров объектов и формирование массива контуров для каждого изображения.*

На двух изображениях выделяют контуры и получают два множества контуров  $Q_1$  и  $Q_2$  соответственно. Множество контуров  $Q_1$  состоит из элементов (контуров)  $q_{lm}$ :

$$Q_1 = \{q_{lm}\}_{l=1, \overline{N_k}},$$

где  $N_k$  — количество контуров.

Из множеств  $Q_1$  и  $Q_2$  исключаются контуры, характеризующиеся следующими признаками:

- часть контура расположена у края изображения;
- контур имеет маленькую протяженность (для различных размеров изображений протяженность различна);
- контур полностью располагается на краю изображения.

Если после исключения не удовлетворяющих условиям контуров количество оставшихся контуров (множества  $Q_{1\text{ост}}$ ,  $Q_{2\text{ост}}$ ) превышает 10, то рассматривают первые 10 из них.

После того как множества контуров на изображениях получены, необходимо определить множество идентичных контуров на паре исследуемых изображений.



*Этап 2. Определение идентичных контуров.*

Для каждого контура  $q_{1i}$ , где  $i$  — порядковый номер элемента множества  $Q_{1\text{ост}}$ , выполняется последовательность действий, основанная на использовании аппарата нечеткой логики, в результате которой будет найден „идентичный“ контур  $q_{2j}$ , где  $j$  — порядковый номер элемента множества  $Q_{2\text{ост}}$ . Экспериментальные результаты показали, что не для всех элементов множества  $Q_{1\text{ост}}$  будет найден „идентичный“ в массиве  $Q_{2\text{ост}}$ , но часть „идентичных“ контуров будет определена, и из них необходимо будет выбрать наиболее информативные.

Введем лингвистическую переменную (ЛП) „идентичный контур“. Функция принадлежности ЛП „идентичный контур“ имеет вид

$$\mu_{\text{ид}} = \mu_{\text{пл}} \wedge \mu_{\text{пр}} \wedge \mu_{\text{цв}} \wedge \mu_{\text{п}}, \quad (1)$$

где  $\mu_{\text{пл}}$  — функция принадлежности терму „площадь“,  $\mu_{\text{пр}}$  — функция принадлежности терму „протяженность“,  $\mu_{\text{цв}}$  — функция принадлежности лингвистической переменной „цвет“,  $\mu_{\text{п}}$  — функция принадлежности терму „положение“.

Терм „площадь“ характеризует степень схожести площадей прямоугольников, в которые вписаны образованные контурами фигуры. Функция принадлежности терму „площадь“ принимает вид:

$$\mu_{\text{пл}}(x, a_{\text{пл}}, b_{\text{пл}}, c_{\text{пл}}) = \begin{cases} \mu_1(x, a_{\text{пл}}, b_{\text{пл}}), & x < b_{\text{пл}}, \\ 1, & b_{\text{пл}} \leq x \leq c_{\text{пл}}, \\ 1 - \mu_1(x, c_{\text{пл}}, c_{\text{пл}} + b_{\text{пл}} - a_{\text{пл}}), & x > c_{\text{пл}}; \end{cases} \quad (2)$$

$$\mu_1(x, a_{\text{пл}}, b_{\text{пл}}) = \begin{cases} 0, & x \leq a_{\text{пл}}, \\ \frac{2(x - a_{\text{пл}})^2}{(b_{\text{пл}} - a_{\text{пл}})^2}, & a_{\text{пл}} < x \leq \frac{a_{\text{пл}} + b_{\text{пл}}}{2}, \\ 1 - \frac{2(x - a_{\text{пл}})^2}{(b_{\text{пл}} - a_{\text{пл}})^2}, & \frac{a_{\text{пл}} + b_{\text{пл}}}{2} < x < b_{\text{пл}}, \\ 1, & x \geq b_{\text{пл}}; \end{cases} \quad (3)$$

где  $a_{\text{пл}}, b_{\text{пл}}, c_{\text{пл}}$  — параметры.

Параметры  $a_{\text{пл}}, b_{\text{пл}}, c_{\text{пл}}$  вычисляются для каждого контура на первом изображении, соответственно идентифицироваться будут нижним индексом, т.е. для контура  $q_{1i}$  параметры именуются  $a_{\text{пл}1i}, b_{\text{пл}1i}, c_{\text{пл}1i}$  и так далее.

Терм „протяженность“ характеризует степень идентичности контуров по длине. Под длиной или протяженностью в данном случае понимают количество единичных векторов, описывающих контур. Функция принадлежности терму „протяженность“ принимает вид:

$$\mu_{\text{пр}}(x, a_{\text{пр}}, b_{\text{пр}}, c_{\text{пр}}) = \begin{cases} 0, & x \leq a_{\text{пр}}, \\ \frac{x - a_{\text{пр}}}{c_{\text{пр}} - a_{\text{пр}}}, & a_{\text{пр}} < x \leq c_{\text{пр}}, \\ \frac{b_{\text{пр}} - x}{b_{\text{пр}} - c_{\text{пр}}}, & c_{\text{пр}} < x < b_{\text{пр}}, \\ 0, & x \geq b_{\text{пр}}. \end{cases} \quad (4)$$

Функция принадлежности ЛП „цвет“ принимает следующий вид:

$$\mu_{\text{цв}} = \mu_R \wedge \mu_B \wedge \mu_G, \quad (5)$$

где  $\mu_R, \mu_B, \mu_G$  — функция принадлежности терму „красный“, „синий“, „зеленый“ соответственно.

Терм „положение“ описывает схожесть расстояний от центра тяжести контура [8] до отрезка, соединяющего его начало и конец. Функцию принадлежности для термы „положение“ представим следующим образом:

$$\mu_{\text{п}}(x, a_{\text{п}}, b_{\text{п}}, c_{\text{п}}) = \begin{cases} \mu_1(x, a_{\text{п}}, b_{\text{п}}), & x < b_{\text{п}}, \\ 1, & b_{\text{п}} \leq x \leq c_{\text{п}}, \\ 1 - \mu_1(x, c_{\text{п}}, c_{\text{п}} + b_{\text{п}} - a_{\text{п}}), & x > c_{\text{п}}. \end{cases} \quad (6)$$

Параметры определяются экспериментально в зависимости от рассчитанных расстояний на изображениях. Параметры, используемые в функциях принадлежности, будут рассчитываться динамически, и для каждого контура на каждом изображении будет создан свой набор параметров.

В результате выполнения расчетов для каждой пары контуров, например, одного на первом изображении и одного на втором, ставится в соответствие число  $\mu$  от нуля до единицы, характеризующее степень „идентичности“ этой пары контуров.

Из множества рассчитанных значений коэффициентов  $\mu$  выбирают максимальное, и соответствующие контуры, входящие в пару, считаются „идентичными“. Если есть несколько пар контуров с близкими по значению коэффициентами, то выбирается та, что ближе расположена к краю изображения, которым они будут соединяться.

Информация о найденных „идентичных“ контурах и их характеристиках сохраняется в памяти.

Далее этапы идентификации одинаковых областей повторяются для следующей пары изображений — 2 и 3 и т.д. Результаты обработки всех изображений сохраняются для дальнейшего использования.

Заключительным этапом формирования панорамного изображения является соединение частей изображений на основе контурной информации.

Экспериментальные исследования показали, что разработанный метод превосходит существующие подходы к формированию панорам за счет возможности его использования для работы с неограниченным множеством изображений с одинаковыми областями, при этом размеры изображений могут быть различными. Получение панорам разработанным методом не требует знания параметров камеры, с помощью которой изображения были получены. Точность сформированного панорамного изображения повышается за счет коррекции радиальной дисторсии.

Работа выполнена при поддержке фонда содействия развитию малых форм предприятий в научно-технической сфере „У.М.Н.И.К.“ (договор № 0806, 2008 г., проект № 8555).

#### СПИСОК ЛИТЕРАТУРЫ

1. Pat. 7386188 USA. Merging images to form a panoramic image / J. Peterson. 10.06.2008.
2. Pat. 7095905 USA. Merging images to form a panoramic image / J. Peterson. 22.08.2006.
3. Pat. 20070159527 USA. Method and apparatus for providing panoramic view with geometric correction / K. H. Kim. 12.07.2007.
4. Шишкин Е. В., Боресков А. В. Компьютерная графика. М.: Диалог-МИФИ, 1995. 288 с.
5. Pat. 2007/122584. Method and device for generating a panoramic image from a video sequence / S. Auberger. 01.11.2007.

6. Бугаенко Е. И., Труфанов М. И. Способ автоматической коррекции радиальной дисторсии на цифровых изображениях // Тез. докл. XXXV Межвуз. науч.-технич. конф. студентов и аспирантов в области научных исследований „Молодежь и XXI век“. Ч. 1. Курск: КурскГТУ, 2007. С. 16—17.
7. Бугаенко Е. И., Труфанов М. И. Способ калибровки радиальной дисторсии оптической системы цифровой камеры // Сб. мат. X Междунар. науч.-технич. конф. „Медико-экологические технологии — 2007“. Курск: КурскГТУ, 2007. С. 173 — 176.
8. Дегтярев С. В., Садыков С. С., Тевс С. С., Ширабакина Т. А. Методы цифровой обработки изображений: уч. пос. Ч. 1. Курск: КурскГТУ, 2001. 167 с.

**Сведения об авторах**

- Владимир Николаевич Гридин** — д-р техн. наук, профессор; ЦИТП РАН, Московская обл., Одинцово; директор
- Елена Ивановна Бугаенко** — аспирант; Курский государственный технический университет, кафедра вычислительной техники; E-mail: Elena.bugaenko@gmail.com

Рекомендована кафедрой  
вычислительной техники

Поступила в редакцию  
12.09.08 г.

УДК 004.936

С. С. САДЫКОВ, А. А. ОРЛОВ, А. А. ЕРМАКОВ

**МЕТОДИКА ОБРАБОТКИ ЛИНЕЙЧАТЫХ ОБРАЗОВ  
НА ДЕФЕКТОСКОПИЧЕСКИХ СНИМКАХ**

Дается определение полосового (линейчатого) образа и строится его модель. Разрабатываются метод и алгоритм многомасштабной фильтрации полосовых образов. Представлена методика обработки дефектоскопических снимков, содержащих линейчатые образы дефектов. Приводится пример обработки реального изображения.

**Ключевые слова:** пространственно-распределенный объект, топологическое отношение, муниципальные геоинформационные системы, матрица шестнадцати пересечений.

В настоящее время прослеживается тенденция развития систем технического зрения и методов обработки изображений в цифровой дефектоскопии. Цифровой анализ позволяет быстро обнаруживать, измерять и классифицировать характер дефектов для обеспечения неразрушающего контроля различных материалов и изделий. Существует множество публикаций, связанных с анализом дефектоскопических снимков различной природы путем компьютерной обработки изображений. Основные подходы к решению таких задач представлены в статьях [1, 2].

Многие дефекты имеют специфические сложные образы. Например, очень распространены дефекты (трещины, риски, царапины), имеющие протяженную линейчатую форму (полосы). В существующих системах технического зрения, применяемых в дефектоскопии, в основном используются классические методы цифровой обработки изображений [3], не позволяющие выполнить более детальный разбор сложных по форме дефектов. Поэтому актуальным является создание, исследование и применение новых, более эффективных, специальных методов обработки изображений полосовых дефектов.

Из теории сигналов известен так называемый фильтр воспроизведения полезного сигнала [3]. Этот фильтр позволяет выделить из шума и помех сигнал заданной формы. Однако такой фильтр используется в основном для обработки одномерных сигналов. В нашем случае

к исходным данным для анализа относится изображение, которое рассматривается как двумерный сигнал. Предполагается использовать специфику формы образов — их полосовую структуру (производить фильтрацию вдоль полос).

Цель настоящей работы заключается в разработке специальной технологии обработки дефектоскопических снимков, содержащих линейчатые образы дефектов.

**Математическая модель полосового образа.** Пусть  $x = x(t)$ ,  $y = y(t)$  — параметрические уравнения некоторой кривой (рис. 1, пунктирная кривая) на плоскости  $R^2$ . Множество точек, расположенных вдоль кривой  $x = x(t)$ ,  $y = y(t)$  на расстоянии, не превышающем значения  $\sigma$ ,

$$S = \{(x_0, y_0) \mid (x_0 - x(t))^2 + (y_0 - y(t))^2 \leq \sigma^2\},$$

назовем областью полосы, заданной этой кривой.

Кривую  $x = x(t)$ ,  $y = y(t)$  будем называть образующей полосы  $S$ .

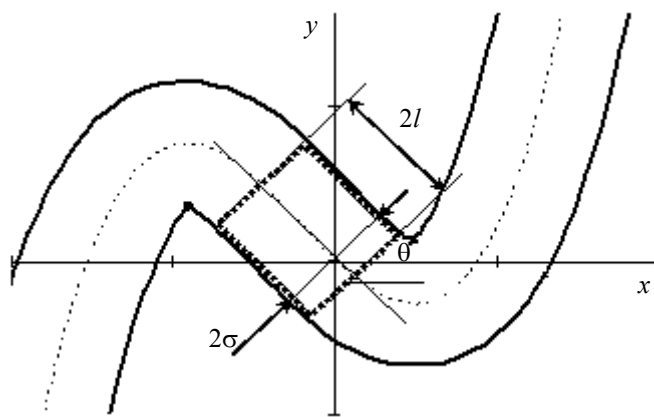


Рис. 1

Если значение  $\sigma$  (полуширина сегмента) является функцией  $\sigma = \sigma(t)$ , то полосу  $S$  назовем полосой с переменной шириной, так что  $2\sigma(t)$  — ширина этой полосы, изменяющаяся в соответствии с образующей кривой  $x = x(t)$ ,  $y = y(t)$ .

Сцена (изображение) полосы задается характеристической функцией

$$\chi(x_0, y_0) = \begin{cases} 1, & (x_0, y_0) \in S, \\ 0. & \end{cases}$$

Профилем полосы  $S$  в точке  $(x, y)$  будем называть сечение поверхности  $z = \chi(x, y)$  нормальной плоскостью в пространстве  $R^3$  к образующей кривой  $x = x(t)$ ,  $y = y(t)$  в точке  $(x, y)$ . Профиль изображения полосы  $S$  будет иметь прямоугольную форму.

Обобщим понятие образа полосы, видоизменив ее профиль. Для этого будем полагать, что полосовое изображение состоит (точнее, интегрируется) из бесконечного множества профилей, расположенных вдоль кривой  $x = x(t)$ ,  $y = y(t)$ .

Пусть  $A(x)$  — профиль полосы, так что

$$\lim_{x \rightarrow \pm\infty} A(x) = 0,$$

тогда характеристическая функция изображения полосы профиля  $A(x)$  определится как криволинейный интеграл:

$$\chi(x_0, y_0) = \int_{\substack{x=x(t), \\ y=y(t)}} A(|(x, y) - (x_0, y_0)|) ds.$$

Примеры изображения кривой (а), полосы с прямоугольным профилем (б) и полосы, где  $A(x)$  — гауссиан (в), приведены на рис. 2.



Рис. 2

Прямоугольник, аппроксимирующий часть полосы, будем называть сегментом полосы. Обозначим полудлину сегмента как  $l$ , а угол наклона сегмента (равный углу наклона нормали к образующей полосы) —  $\theta$  (см. рис. 1).

**Воспроизведение одномерных сигналов.** Рассмотрим одномерный случай. Фурье-спектр фильтра воспроизведения полезного сигнала определяется как

$$\Psi(\omega) = \frac{W_s(\omega)}{W_s(\omega) + W_n(\omega)},$$

где  $W_s(\omega)$  и  $W_n(\omega)$  — энергетический спектр (плотность мощности) полезного сигнала и помех.

Под помехами (шумами) будем понимать все сигналы вида, отличного от  $s(x)$ .

Энергетические спектры определяются как

$$W_s(\omega) = |S(\omega)|^2 \text{ и } W_n(\omega) = |N(\omega)|^2,$$

где  $S(\omega)$ ,  $N(\omega)$  — фурье-спектры полезного сигнала  $s(x)$  и помех  $n(x)$  соответственно.

Полезный сигнал (в нашем случае это функция  $s(x)$ ) содержится в исходном сигнале  $f(x)$  в сумме с шумами и помехами  $n(x)$ :

$$f(x) = s(x) + n(x).$$

Пусть  $\psi(x)$  — фильтр воспроизведения полезного сигнала,  $g(x)$  — функция восстановленного сигнала ( $g(x) = f(x) \otimes \psi(x)$ ). С помощью фильтра  $\psi(x)$  производится минимизация средней квадратической ошибки, которая характеризует степень отклонения функции восстановленного сигнала  $g(x)$  от полезного  $s(x)$ :

$$\varepsilon = \mathbf{M}[(g(x) - s(x))^2] = \mathbf{M}[(f(x) \otimes \psi(x) - s(x))^2] \rightarrow \min,$$

где  $\mathbf{M}$  — оператор среднего.

Пусть теперь  $s(x)$  — многомасштабная функция, т.е.  $s = s(x/\sigma)$ , так что  $\sigma \in [\sigma_1, \sigma_2]$  ( $\sigma_1$ ,  $\sigma_2$  — минимальное и максимальное значения масштаба соответственно). Для фильтрации сигналов различного масштаба в равной степени необходимо, чтобы средняя квадратическая ошибка была одинаковой для каждого значения  $\sigma$  из  $[\sigma_1, \sigma_2]$ . В качестве критерия оценки точности функционирования фильтра примем среднее значение ошибки для различных масштабов:

$$\varepsilon = \mathbf{M}_\sigma \{ \mathbf{M}[(f(x) \otimes \psi(x) - s(x/\sigma))^2] \} \rightarrow \min,$$

где  $\mathbf{M}_\sigma$  — оператор среднего по масштабу  $\sigma$ .

Можно доказать, что такой критерий обеспечивается фильтром при

$$\Psi(\omega) = \frac{\mathbf{M}_\sigma[W_s(\sigma\omega)]}{\mathbf{M}_\sigma[W_s(\sigma\omega)] + W_n(\omega)},$$

так что

$$\mathbf{M}_\sigma[W_s(\sigma\omega)] = \frac{1}{\sigma_2 - \sigma_1} \int_{\sigma_1}^{\sigma_2} W_s(\sigma\omega) d\sigma.$$

**Двумерная фильтрация вдоль полосы.** Функцию  $s(x, y)$ , которая максимально соответствует изображению полосы  $\chi(x, y)$  в каждой точке области ее сегмента, назовем детектором сегмента полосы (или детектором полосы).

В данном случае детектор сегмента полосы будет определять полезный двумерный сигнал. Можно показать, что если полоса имеет прямоугольный профиль, то

$$s(x, y) = \begin{cases} e^{-y^2/2}, & -1 \leq x \leq 1, \\ 0. & \end{cases}$$

Пусть  $S(\omega, \nu)$ ,  $N(\omega, \nu)$  — фурье-спектры детектора сегмента полосы и функции шума соответственно.

Назовем воспроизводящим интегральным преобразованием по сегменту полосы (ВИПСП) преобразование  $\mathbf{H}_\theta[\cdot]$ , выполняющее фильтрацию в пределах  $[\sigma_1, \sigma_2]$  функции  $f(x, y)$  вдоль каждого направления  $\theta$ . Спектральная функция  $g(x, y, \theta)$  формируется по правилу

$$G(\omega, \nu, \theta) = F(\omega, \nu) \frac{M_s}{M_s + |N(\omega, \nu)|^2},$$

где  $F(\omega, \nu)$ ,  $F(\omega, \nu, \theta)$  — фурье-спектры функций  $f(x, y)$  и  $g(x, y, \theta)$  соответственно,

$$M_s = \frac{1}{\sigma_2 - \sigma_1} \int_{\sigma_1}^{\sigma_2} \left| S(\mathbf{Rot}(\sigma\omega, l\nu)) \right|_\theta^2 d\sigma.$$

Преобразование исходного изображения  $f(x, y)$  на основе ВИПСП в изображение  $g(x, y)$ , при котором выполняется условие

$$g(x, y) = \max_\theta \mathbf{H}_\theta[f(x, y)],$$

назовем воспроизведением полосы (ВП).

С помощью ВП можно найти максимумы в откликах ВИПСП, тем самым среди сегментов разной ориентации определив сегмент, который наилучшим образом соответствует части полосы.

Приведем алгоритм воспроизведения полосы:

- 1) формирование фурье-спектра  $F(\omega, \nu)$  исходного изображения  $f(x, y)$ ;
- 2) вычисление спектра шума  $N(\omega, \nu)$  по области изображения, которая не содержит полезного сигнала;

- 3) выполнение воспроизводящего интегрального преобразования

$$G(\omega, \nu, \theta) = \mathbf{H}_\theta[F(\omega, \nu)];$$

- 4) обратное преобразование фурье-спектра  $G(\omega, \nu, \theta)$  в  $g(x, y, \theta)$ ;

- 5) поиск максимумов в функции  $g(x, y, \theta)$

$$g(x, y) = \max_\theta g(x, y, \theta).$$

**Методика обработки дефектоскопических изображений.** Методика обработки и анализа дефектоскопических изображений, содержащих полосовые образы, включает следующие этапы:

- 1) воспроизведение полосовых образов с помощью разработанного алгоритма;
- 2) бинаризация фильтрованного изображения;
- 3) вычисление площади выделенных областей;
- 4) устранение малых (незначачих) областей;

- 5) определение признаков (характеристик) выделенных дефектных областей;
- 6) классификация дефектов.

Фильтрация позволяет выделить образы протяженной формы, которые, вероятнее всего, являются дефектными, и подавить фоновые шумы и помехи, вызванные неоднородной структурой оцениваемой поверхности.

Пороговая бинаризация изображения необходима для отделения дефектных областей от фона. В результате бинаризации возможно выделение малых зашумленных областей, которые не являются дефектными. Для устранения из рассмотрения таких областей выполняется их отсечение по площади. Порог бинаризации необходимо выбирать в зависимости от уровня значимости обнаруживаемых дефектов.

Этап вычисления геометрических признаков объектов основан на использовании известных алгоритмов обработки монохромных (бинарных) изображений, вычисляются их площадь, длина, ширина и др.

На этапе классификации образов выполняются распознавание и классификация областей. В зависимости от их ширины определяется значимость дефекта. В результате получается характеристическое (маркированное) изображение с отмеченными областями значимых дефектов.

**Экспериментальные результаты.** На рис. 3 приведен пример воспроизведения образа трещины как полосы прямоугольного профиля на дефектоскопическом изображении (здесь *a* — исходное изображение, *б* — результат фильтрации; *в*, *г* — их бинарные образы соответственно). Обработка выполнена при параметрах, которые соответствуют выделяемому объекту: угол наклона сегмента полосы  $\theta \in [0, \pi)$ , полуширина полосы  $\sigma \in [0, 5, 3]$ , полудлина сегмента полосы  $l = 3$ . Спектр помех  $N(\omega, \nu)$  определен по области изображения, которая не содержит трещины.

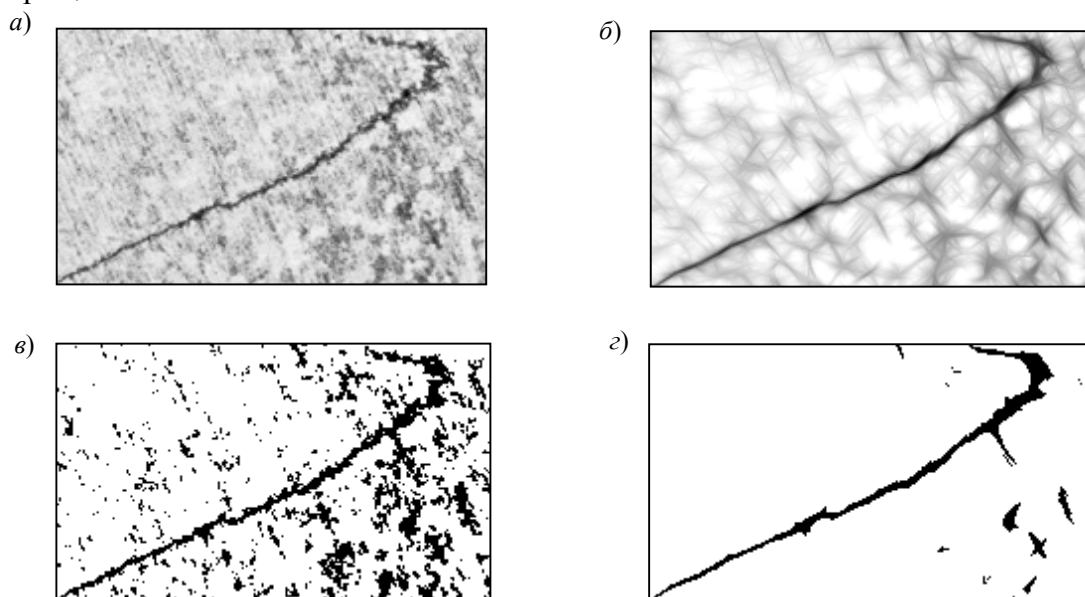


Рис. 3

Проведена бинаризация исходного и обработанного изображений. Порог бинаризации выбран таким образом, чтобы образ трещины не содержал разрывов. Из рис. 3, *в*, *г* видно, что с помощью фильтрации возможно довольно четко отделить фон от изображения, следовательно, возможно выделять изображения дефектов такого рода для дальнейшей автоматической оценки качества поверхности анализируемого изделия.

**Заключение.** Таким образом, в настоящей работе построена математическая модель обработки дефектоскопических изображений, учитывающая множество структурных особенностей полосовых объектов. Это обеспечивает качественный результат обработки изображений,

который подтверждается экспериментальными исследованиями. Обработка образов полос на основе предлагаемой методики обеспечивает их непрерывность на малоконтрастных изображениях. Появляется возможность дальнейшего автоматического анализа.

Следует отметить возможность и целесообразность применения метода в промышленности для автоматического анализа различного рода сцен, которые содержат полосовые образы.

#### СПИСОК ЛИТЕРАТУРЫ

1. Mery D., Berti M. A. Automatic Detection of Welding Defects Using Texture Features // *Insight*. 2003. Vol. 45, N 10. P. 676—681.
2. Silva R. R., Caloba L. P., Siqueira M. H. S. Evaluation of the Relevant Characteristic Parameters of Welding Defects and Probability of Correct Classification Using Linear Classifiers // *Insight*. 2002. Vol. 44, N 10. P. 197—218.
3. Методы компьютерной обработки изображений / Под ред. В. А. Сойфера. М.: Физматлит, 2003. 784 с.

#### Сведения об авторах

- Султан Садыкович Садыков** — д-р техн. наук, профессор; Муромский институт (филиал) Владимирского государственного университета, кафедра информационных систем
- Алексей Александрович Орлов** — канд. техн. наук, доцент; Муромский институт (филиал) Владимирского государственного университета, кафедра информационных систем; E-mail: AlexeyAlexOrlov@rambler.ru
- Александр Андреевич Ермаков** — аспирант; Муромский институт (филиал) Владимирского государственного университета, кафедра информационных систем

Рекомендована кафедрой  
информационных систем

Поступила в редакцию  
12.09.08 г.

УДК 681.3

В. В. ГОРДИЕНКО, В. М. ДОВГАЛЬ, Р. А. ПУЗЫНА

### РАСПОЗНАВАНИЕ ИЗОБРАЖЕНИЙ С ИСПОЛЬЗОВАНИЕМ ХАОТИЧЕСКОЙ ТРАНСФОРМАЦИИ ЭТАЛОНОВ И ОБЪЕКТОВ

Приведен метод распознавания изображений объектов путем использования дискретных хаотических отображений для трансформации эталонов и объектов распознавания изображения, а также для выбора их информативных фрагментов с целью восстановления и распознавания.

**Ключевые слова:** распознавание изображения, хаотический, отображение, эталон, объект, трансформация, таксон восстановления, отказ, матрица.

Настоящая статья посвящена разработке метода распознавания искаженных изображений (объектов), поступающих в графическую базу данных, в которой содержится определенное количество различающихся по Хеммингу эталонных изображений.

Очевидно, что прямое сопоставление эталона и искаженного изображения, выступающего в роли объекта распознавания, с использованием расстояния по Хеммингу обычно неприемлемо. Это обстоятельство послужило отправной точкой для разработки нового метода распознавания. В течение ряда лет авторы предлагаемого метода, имея опыт в области распознавания образов, разрабатывали методы компьютерной стеганографии на основе теории хаотических систем.



При исследовании хаотического рассеивания сообщения по пространству изображения-контейнера с последующим восстановлением был обнаружен эффект восстановления исходного изображения с визуально не воспринимаемыми отличиями (изменениями). При этом восстановление выполнялось не по всему изображению, а по случайным образом выбранному малому участку с предварительным хаотическим рассеиванием (трансформацией) исходного изображения.

Представим основные термины и введем аббревиатуры. Все изображения являются черно-белыми, они адекватно замещаются 0,1-матрицами размером  $N \times N$ . Эталонные изображения, хранящиеся в графической базе данных (ГБД), калиброваны по центру размещения и углу поворота. Такие изображения будем называть форматированными и обозначать „ЭФИ“. Поступающие в ГБД и полученные, например, в результате сканирования, фотографирования и т.д., как правило искаженные, изображения, являющиеся объектами распознавания, будем называть входными „ВИ“.

Основное существенное отличие предлагаемого метода распознавания ВИ заключается в том, что в нем используются положения теории хаотических систем [см. лит.]. Сегодня в достаточной степени изучены одномерные дискретные отображения, являющиеся генераторами хаотических числовых рядов (ХЧР). В качестве генератора ХЧР будем использовать отображение Э. Лоренца, имеющее минимальную вычислительную сложность по числу операций следующего вида:

$$X_{t+1} = 1 - 2|X_t|. \quad (1)$$

Начальное (стартовое) значение ХЧР  $X_0$  задается в виде правильной десятичной дроби. Как известно, каждому стартовому значению соответствует один и только один ХЧР. Установленным фактом является то, что все ХЧР при различных стартовых значениях не повторяются и не имеют совпадающих элементов как внутри себя, так и между собой. При этом необходимо иметь представительную разрядную сетку не менее 32 двоичных разрядов, тогда в рядах будет содержаться  $R = 2^{32}$  чисел. Очевидно, что строгое упорядочение не может быть осуществлено для числового ряда, имеющего одинаковые по значимости элементы, поэтому никакие генераторы, кроме хаотических, не могут быть использованы в предлагаемом методе.

На первом этапе распознавания выполняется предварительная обработка ЭФИ. С этой целью для всех ЭФИ из ГБД выполним операцию хаотического рассеивания в пространстве формата. Операция хаотического рассеивания содержит следующие процедуры.

1. Все элементы формата каждого ЭФИ нумеруются от 1 до  $N^2$ .
2. Запускается генератор ХЧР (1) с заданного стартового значения и формируется  $N^2$  чисел, которые ставятся во взаимно однозначное соответствие номерам позиций элементов формата.
3. Выполняется быстрая сортировка полученного ХЧР с алгоритмической сложностью  $O = N^2 \log(N^2)$ .
4. Все значения (0 или 1) в каждой имеющей свой порядковый номер позиции (от 1 до  $N^2$ ) для каждого ЭФИ переносятся в новые позиции, расположение которых теперь однозначно определяется строго равными значениями элементов упорядоченного ХЧР и исходного ХЧР.

Полученные в результате формы представления ЭФИ будем называть хаотическими трансформантами эталонов и обозначать „ХТЭ“.

На втором этапе распознавания при поступлении ВИ в ГБД оно вначале калибруется по центру и углу поворота, а затем над ним также выполняется операция хаотического рассеивания. Полученное ВИ будем называть частичным трансформантом и обозначать „ХТВИ“.

На третьем этапе предлагаемого метода выполняется операция хаотического выбора подмножества мощностью  $P$  элементов изображений ХТЭ или ХТВИ, которые условимся

называть таксонами восстановления и обозначать „ТВ“. Механизм хаотического выбора подмножества из формата изображения реализуется также с помощью отображения (1) с некоторым заданным значением  $X_0$ , одинаковым для всех изображений. Полученный ХЧР преобразуется в битовую строку, из которой сохраняются биты с номерами позиций с 1 до  $P \leq N^2$ , но при  $P$ , не меньшем значения  $N^2/\alpha$ . Параметр  $\alpha$  через  $P$  задает ту минимальную часть трансформанта, которая после восстановления изображения (см. ниже) позволяет по нему осуществить визуальное распознавание исходного изображения.

Формирование ТВ для всех ХТЭ осуществляется один раз после завершения загрузки ГБД. Для каждого изображения вычленяется множество  $ТВ_j$  при  $j = 1, 2, \dots, S$ , где  $S = N^2/\alpha$ . После формирования ТВ все многообразие ЭФИ при реализации предлагаемого метода распознавания в общем объеме не используется. На заключительном этапе метода обрабатывается только малая часть изображения. Это обстоятельство существенно снижает емкостную сложность (затраты памяти) метода и соответственно его алгоритмической реализации.

На четвертом этапе метода распознавания выполняется сопоставление по расстоянию Хемминга между выбранным таксонами из ХТВИ и всеми — из ХТЭ. Если осуществляется выбор одного или нескольких (до пяти)  $W$  ХТЭ по минимуму расстояния Хемминга, то реализуется следующий этап метода, в противном случае выполняются присоединение к выбранному одному, любого, ТВ для всех ХТВИ и ХТЭ. При невыполнении условий перехода к следующему этапу метода выполняется очередное присоединение ТВ.

Сформулируем важное утверждение: „Поскольку ТВ являются высокоспецифичными для каждого изображения и строго различаются на основании свойства ХЧР не повторять значения собственных элементов, а также в силу исходного различия по расстоянию Хемминга эталонов ГБД между собой, то итерация присоединения очередного ТВ завершается за конечное число шагов на основании того, что объединение разных ТВ из их множества в совокупности, например при  $S = N^2$ , приводит к его полному совпадению с исходным трансформантом изображения“.

На пятом этапе метода распознавания выполняются процедура восстановления ВИ и его классификация на основании нового множества  $ТВ_j$ , которое формируется для каждой пары  $ТВ_j$  из ХТВИ и  $ТВ_j$  из ХТЭ для каждого из выбранного на предыдущем этапе метода применением операции конъюнкции всех пар элементов (каждая определяется одинаковым порядковым номером от 1 до  $N^2$ ). Формируется новый трансформант промежуточного изображения (ТПИ), в который строго на свои позиции переносятся только единичные значения из нового  $ТВ_j$  в строгом соответствии с их размещением в ХТВИ. После этого выполняется процедура восстановления ВИ по полученному ТПИ.

Процедура восстановления (обратная операции хаотического рассеивания, с сохранением используемых для него исходного и упорядоченного ХЧР) выполняется в следующей последовательности.

1. Каждому элементу из упорядоченного ХЧР, имеющего свой порядковый номер, ставится во взаимно однозначное соответствие каждая позиция значения из ТПИ, имеющая одинаковый номер от 1 до  $N^2$ .

2. Из исходного ХЧР позиции элементов изображения переносятся из ТПИ в позиции, строго соответствующие совпадающим значениям элементов упорядоченного ХЧР. Перенос осуществляется для всех элементов изображения.

3. Восстановление осуществляется для всех полученных ТПИ. В результате получают новые изображения, число которых равно  $PW$ .

На шестом этапе выполняется сопоставление каждого из  $P$  восстановленных изображений по их ТПИ с одним и только одним ЭФИ. С этой целью определяется расстояние по Хеммингу между первым восстановленным ТПИ и ЭФИ. Затем выбирается второй ТВ из ХТВИ и повторяется построение ТПИ, а также выполняется процедура восстановления ВИ. После

этого выполняется операция дизъюнкции для всех пар элементов, имеющих равные порядковые номера в двух восстановленных изображениях, и формируется третье изображение. Это изображение назовем аккумулятором. Вновь вычисляется расстояние по Хеммингу между полученным аккумулятором и выбранным ЭФИ. Осуществляется построение всех  $P$  восстановленных ВИ с промежуточным и обязательным формированием аккумуляторов с определением расстояния по Хеммингу от него до выбранного ЭФИ. Аналогичным образом выполняются все построения для всех  $W$  ЭФИ.

Если хотя бы для одного фиксированного ЭФИ из множества  $W$  расстояние по Хеммингу от него до последовательно формирующихся  $P$  аккумуляторов убывает, то рассматриваемое ВИ относится к этому эталону. Но если для всех ЭФИ из множества  $W$  указанное расстояние возрастает, то процесс распознавания прекращается. В этом случае пользователь или выбраковывает ВИ, или включает его в исходном виде в качестве эталона, пополняя графическую базу данных.

Представленный метод распознавания алгоритмизирован и был разработан программный продукт „Haos-DGP“, использование которого показало, что при распознавании черно-белых изображений размером  $500 \times 500$  при  $\alpha = 100$  все искаженные до 40 % тестовые входные изображения, полученные из эталонов (в количестве 1000), распознавались. При этом количество  $W = 2$ , а максимальное наблюдаемое значение — 24. Коэффициент качества распознавания для общего случая имеет оценку не ниже 0,94.

Вместе с тем затраты времени в некоторых случаях достигали 5 с при среднем показателе затрат времени 0,8 с. Основным источником завышенных затрат времени является сортировка, поэтому для ее реализации рекомендуется использовать многоядерные вычислители или эффективно использовать для сортировки и слияния процессоры видеокарты (пиксельные и шейдерные), не занятые при реализации разработанного метода распознавания.

В настоящее время авторский коллектив завершает разработку программного продукта для распознавания изображений в формате .bmp с тремя байтами на пиксел.

#### ЛИТЕРАТУРА

Паркер Т. С., Чжуа Л. О. Введение в теорию хаотических систем для инженеров // ТИИЭР. 1987. Т. 75, № 8. С. 6—40.

#### *Сведения об авторах*

- Виктория Викторовна Гордиенко** — канд. техн. наук, доцент; Курский государственный технический университет, кафедра уголовного права
- Виктор Митрофанович Довгаль** — д-р техн. наук, профессор; Курский государственный технический университет, кафедра программного обеспечения вычислительной техники; заведующий кафедрой; E-mail: vmdovgal@yandex.ru
- Роман Андреевич Пузына** — аспирант; Курский государственный технический университет, кафедра программного обеспечения вычислительной техники

Рекомендована кафедрой  
программного обеспечения  
вычислительной техники

Поступила в редакцию  
12.09.08 г.

И. М. ГОСТЕВ, А. Г. ПОДГОРБУНСКИЙ

## РАСПРЕДЕЛЕНИЕ ПОТОКОВ ГРАФИЧЕСКОЙ ИНФОРМАЦИИ В СИСТЕМАХ РАСПОЗНАВАНИЯ ОБРАЗОВ В РЕАЛЬНОМ МАСШТАБЕ ВРЕМЕНИ

Рассматриваются два алгоритма управления видеопотоками в распределенной системе по обработке изображений и распознаванию образов на основе теории графов. Представлен алгоритм управления распределенной системой посредством контроллера потока. Приведено математическое описание статического (граф определен до начала работы системы) и динамического (граф может быть перестроен в процессе работы системы в зависимости от особенностей видеопотока) распределения информационных потоков.

*Ключевые слова:* алгоритм управления, графическая информация, распознавание объектов, реальный масштаб времени.

**Введение.** Несмотря на постоянное повышение быстродействия компьютеров построение систем обработки изображений и распознавания образов в реальном масштабе времени сопряжено с преодолением ряда проблем. В работе [1] показано, что быстродействие вычислительной системы для обработки видеопотока должно быть пропорциональным размеру изображения, для телевизионного формата оно превышает мощность одного компьютера примерно в 25—30 раз. Для решения этой проблемы были разработаны параллельные алгоритмы, позволяющие реализовать такую систему в распределенной среде, при этом каждый из элементов системы выполняет свою операцию по обработке изображения и распознавания образов. Для управления всей системой обработки видеопотока и распознавания объектов используется контроллер, который распределяет задания по вычислительным узлам системы, осуществляет контроль и синхронизацию их выполнения.

В работах [1, 2] была предложена и развита идея построения распределенной системы обработки и распознавания изображений в режиме реального времени, основанной на принципах конвейера и разбиения потока на участки. В настоящей работе рассматривается способ моделирования параллельных вычислений, основанный на применении ориентированных графов [3].

**Математическая модель процессов.** Разделим методы распределения потока на статические и динамические. В первом случае маршрут потоков информации определяется на этапе начальной инициализации системы и далее остается неизменным; размер фрагмента потока, его маршрут и методы обработки заранее определены. При динамическом распределении маршрут и размеры фрагментов определяются на стадии обработки.

Рассмотрим односвязный ориентированный ациклический граф, имеющий один исток и один сток, вершинами которого являются узлы вычислительной сети, реализованные в виде экземпляров метода, а дугами — каналы передачи потока. Методом назовем процедуру обработки изображения, реализованную в виде библиотечной функции. Входными данными метода будут фрагменты изображения и его параметры, необходимые для процедуры обработки. В результате работы метода получаются выходное изображение и/или некоторые другие данные, полученные в результате применения процедуры, формат которых зависит от конкретного метода. С помощью методов осуществляются разделение исходного потока на фрагменты, их параллельная обработка и последующее объединение нескольких потоков в один — результирующий. Всем процессом управляет компонент, называемый „контроллером потока“.

Поскольку процесс обработки изображения представляет собой последовательность (цепочку) нескольких методов, то в вычислительной сети каждому методу будет сопоставлен один или несколько узлов, его реализующих — экземпляров. Совокупность одинаковых экземпляров образует *ярус* сети, соответствующий данному методу цепочки. Входная информация поступает на некоторый узел только с узлов яруса, соответствующего предыдущему методу цепочки. Все выходы узла направлены только на узлы яруса, соответствующего следующему методу в цепочке. Первый и последний ярусы содержат по одному узлу. Каждый метод характеризуется относительной вычислительной сложностью процедуры, началом и концом смещения фрагмента изображения по высоте  $(y^-, y^+)$  и начальными условиями. Например, смещение по высоте  $(y^-, y^+)$  означает, что для метода  $l$  с параметрами начала и конца фрагмента  $[s_1, s_2]$  необходим фрагмент  $l[s_1 - y^-, s_2 + y^+]$ .

**Алгоритм построения вычислительной среды.** *Статическое распределение потока.* Рассмотрим алгоритм статического распределения потока. Пусть исходный поток представляет собой одно длинное изображение — „ленту“. Фрагмент потока — непрерывный участок изображения, содержащий целое число строк. Промежуточные потоки состоят из последовательности фрагментов. Если поток состоит из фрагментов одинаковой длины, следующих через равные промежутки, то он может быть описан схемой потока.

Схема фрагмента потока представляет собой сочетание  $\langle S, L, P \rangle$  где  $S$  — начальная строка первого фрагмента,  $L$  — длина каждого фрагмента в строках,  $P$  — промежуток от конца предыдущего фрагмента до начала следующего. Таким образом,  $\langle S, L, P \rangle$  определяет последовательность фрагментов

$$[S + (L + P)i, S + (L + P)i + L], i \in N.$$

Две схемы называются *смежными*, когда при объединении двух соседних фрагментов их описание (состоящее из всех фрагментов первого и второго потока) также определяется схемой потока. Результирующая схема потока носит название *объединенной*. Символом  $\Xi_i(\langle S, L, P \rangle)$  обозначается схема потока, скорректированная в соответствии со смещениями по  $y$  для  $i$ -го метода (необходимого методу на входе, чтобы производить поток  $\langle S, L, P \rangle$  на выходе):

$$\Xi_i(\langle S, L, P \rangle) = \langle S - y_i^-, L + y_i^- + y_i^+, P - y_i^- - y_i^+ \rangle.$$

Алгоритм состоит из двух шагов — прямого и обратного. На первом происходят определение количества вершин в каждом ярусе и построение связей между ними путем равномерного последовательного разделения потока. На втором шаге происходит коррекция соединений и разметки с целью исключения каналов, по которым передаются очень малые относительно среднего потока на ярусе фрагменты.

**Входом алгоритма** являются цепочка методов  $\langle M_1, \dots, M_N \rangle$ , относительная вычислительная сложность  $c_1, \dots, c_N$ , пропускная способность эталонного метода  $q_e$ , требуемая пропускная способность цепочки  $q$ ; **выходом** — построенный граф вычислительной сети  $G = \{V, E\}$ , ребра и вершины которого размечены схемами потоков  $M \rightarrow \{\langle S, L, P \rangle\}$ ,  $M' \rightarrow \{\langle S, L, P \rangle\}$ .

**Шаг 1.** Для каждого  $i \in [1, N]$  определить ярус  $G_i = \{V_i, \emptyset\}$ , где

$$V_i = \left\{ v_i^k \mid k = 1 \dots \left\lceil \frac{q_e c}{q} \right\rceil \right\}$$

— набор упорядоченных узлов методов, являющихся экземплярами метода  $M_i$ .

Положить  $V = \bigcup_{i \in [1, n]} V_i$ ;  $E = \emptyset$ . Приписать истоку схему потока  $\langle 0, L, 0 \rangle$ , где  $L$  — исходная длина блока, задаваемая контроллером потока на этапе конфигурации. Положить  $M = \emptyset, M' = \left\{ \left( v_1^1, \langle 0, L, 0 \rangle \right) \right\}$ , для каждого узла  $v_i^k$  разметить его следующим образом:

$$\left\langle \left\lfloor \frac{k-1}{|V_i|} \right\rfloor, \left\lfloor \frac{k}{|V_i|} \right\rfloor - \left\lfloor \frac{k+1}{|V_i|} \right\rfloor, L - \left\lfloor \frac{k}{|V_i|} \right\rfloor + \left\lfloor \frac{k+1}{|V_i|} \right\rfloor \right\rangle.$$

Для каждой пары узлов  $v_i^k$  и  $v_{i+1}^{k'}$ ,  $i \in [1, N-1]$ ,  $k \in [1, |V_i|]$ ,  $k' \in [1, |V_{i+1}|]$  в случае если  $M'(v_i^k) \cap \Xi_i(M'(v_{i+1}^{k'})) \neq \emptyset$ , добавить дугу между узлами  $E = E \cup (v_i^k, v_{i+1}^{k'})$  и пометить ее схемой-пересечением:

$$M = M \cup \left\{ \left( (v_i^k, v_{i+1}^{k'}), M'(v_i^k) \cap \Xi_i(M'(v_{i+1}^{k'})) \right) \right\}.$$

**Шаг 2.** Для каждого яруса  $V_i$ ,  $i = N, \dots, 2$  для каждого узла  $v_i^k \in V_i$ ,  $k \in [1, |V_i|]$ .

**а)** Убедиться, что объединенная схема выходов совпадает со схемой узла:

$$\bigcup_{(v_i^k, v) \in E} \left\{ M(v_i^k, v) \right\} = M'(v_i^k).$$

В случае, если это не так и объединенная схема включает схему узла, то вычислить множество недостающих схем

$$\bigcup_{(v_i^k, v) \in E} \left\{ M(v_i^k, v) \right\} \setminus \Xi_i(M'(v_i^k))$$

и каждый недостающий элемент этого множества прибавить к одной из схем входящих ребер узла, имеющих схему, смежную с недостающей, и наибольшую длину.

**б)** Удалить входные дуги, длина схемы которых меньше определенного порога, и схемы этих дуг приписать дугам наибольшей длины, имеющим смежные схемы.

*Динамическое распределение потока.* При динамическом распределении потока между обрабатываемыми узлами маршрут, по которому пройдет конкретный фрагмент изображения, определяется на стадии выполнения. При этом из-за смещений по высоте для обработки данного фрагмента может потребоваться несколько строк предыдущего или последующего фрагмента, непосредственно прилегающего к данному. А так как заранее неизвестно, какие именно фрагменты пройдут через данный экземпляр метода, невозможно сказать, требуется сохранять этот фрагмент или нет. Кроме того, чтобы обеспечить обработку данного фрагмента с учетом смещений по высоте, требуется разместить предыдущий и последующий фрагменты непосредственно до и после него. Указанные факторы усложняют процедуру управления памятью и требуют реализации дополнительных условий маршрутизации на этапе обработки потока, включающих взаимодействие с контроллером потока, который координирует в этом случае маршрутизацию. Это противоречит одной из основных концепций, положенных в основу проектирования системы, согласно которой взаимодействие контроллера потока с методами на этапе обработки должно быть минимальным. Однако обе проблемы могут быть решены с помощью передачи избыточного количества фрагментов, так чтобы каждый фрагмент на каждом ярусе сети мог быть обработан независимо. В этом случае освободить память, занимаемую фрагментом, можно сразу после его обработки. Единственным и очевидным недостатком такого метода является повышение общего объема пересылаемой по сети

информации, однако при больших значениях длины пересылаемых фрагментов и небольшом количестве ярусов эти издержки сравнительно малы.

Приведем алгоритм построения сети с динамическим избыточным распределением потока. При этом в отличие от статического метода, где схема потока, приписанная конкретному узлу или дуге сети, определяет, какие именно фрагменты изображения и в каком порядке пройдут через данную дугу или узел, эта схема определяет, какие фрагменты могут пройти через узел или дугу. Все узлы одного яруса являются равнозначными в том смысле, что любой узел предыдущего яруса может передать обработанный фрагмент любому из узлов следующего яруса.

**На вход** алгоритма поступает цепочка методов  $\langle M_1, \dots, M_N \rangle$  с относительной вычислительной сложностью  $c_1, \dots, c_N$ , пропускной способностью эталонного метода  $q_e$ , требуемой пропускной способностью цепочки  $q$ . **На выходе** получаем построенный граф вычислительной сети  $G = \{V, E\}$ , ребра и вершины которого размечены схемами потоков  $M \rightarrow \{\langle S, L, P \rangle\}$ ,  $M' \rightarrow \{\langle S, L, P \rangle\}$ .

Для каждого  $i \in [1, N]$  определить ярус  $G_i = \{V_i, \emptyset\}$ .

Положить  $V = \bigcup_{i \in [1, n]} V_i$ ;  $E = \emptyset$ . Приписать стоку схему потока  $\langle 0, L, 0 \rangle$ , где  $L$  — исходная длина блока, задаваемая контроллером потока на этапе конфигурации. Положить разметку  $M = \emptyset, M' = \{\langle v_N^1, \langle 0, L, 0 \rangle \rangle\}$ .

Для каждого яруса  $i \in [N - 1, 1]$  разметить все его вершины схемами

$$\left\langle -\sum_{j=i}^N y_j^-, L + \sum_{j=i}^N (y_j^- + y_j^+), -\sum_{j=i}^N (y_j^- + y_j^+) \right\rangle.$$

Для каждой пары узлов  $v_i^k$  и  $v_{i+1}^{k'}$ ,  $i \in [1, N - 1]$ ,  $k \in [1, |V_i|]$ ,  $k' \in [1, |V_{i+1}|]$  добавить дугу между ними

$$E = E \cup (v_i^k, v_{i+1}^{k'})$$

и пометить ее схемой  $\langle S, L, P \rangle$ :

$$\left\langle -\sum_{j=i+1}^N y_j^-, L + \sum_{j=i+1}^N (y_j^- + y_j^+), -\sum_{j=i+1}^N (y_j^- + y_j^+) \right\rangle.$$

В построенной таким образом сети каждый узел каждого яруса соединен дугой с каждым узлом последующего яруса. При большом количестве узлов в соседних ярусах это может привести к значительной дополнительной нагрузке на сетевые ресурсы вычислительной системы (каждая дуга соответствует двум дополнительным потокам в процессах отправляющего и принимающего узлов). Для снижения нагрузки при вычислении множества дуг можно использовать первый алгоритм, а для разметки дуг и узлов — второй. Тогда каждый узел может послать обработанный фрагмент любому из узлов следующего яруса, и простейшим способом маршрутизации обработанных фрагментов из данного узла станет чередующаяся рассылка фрагментов последовательно каждому из узлов следующего яруса. При этом будет достигаться равномерное распределение загрузки между всеми узлами каждого яруса в предположении, что каждый из них обладает равным количеством ресурсов.

**Обсуждение и выводы.** Подводя итог, можно выделить следующие преимущества и недостатки статического и динамического способа распределения потока.

Распределенная система обладает высокой отказоустойчивостью — при выходе из строя одного из узлов сети его функции перераспределяются между остальными узлами того же яруса. Поэтому при динамическом распределении потоков неисправный узел будет заменен. При статическом методе распределения отказ узла приведет к появлению в выходном изображении необработанных участков.

Динамическое распределение дает возможность более гибкой балансировки загрузки узлов вычислительной сети. Если какой-то узел перегружен, его работу можно распределить на другие узлы того же яруса. Статическое распределение не позволяет этого делать. При динамическом распределении суммарный объем пересылаемых данных возрастает вследствие избыточной передачи строк, появляется дополнительная нагрузка на узлы, связанная с маршрутизацией и балансировкой исходящих фрагментов. По сравнению со статическим методом возрастают число потоков приема и передачи фрагментов и число сетевых соединений.

В целом преимущества динамического метода превосходят его недостатки.

### СПИСОК ЛИТЕРАТУРЫ

1. *Gostev I. M., Sevastianov L. A.* About construction distributed calculation in real time system of image processing and pattern recognition // Proc. Int. Conf. „Distributed computation and GRID in science and education“. Dubna JINR. June, 2004. P. 79—85.
2. *Gostev I. M., Sevastianov L. A.* Internal Architecture of Distributed Real-time System of Image Processing and Pattern Recognition // Proc. XX Int. Symp. on Nuclear Electronics & Computing (NEC 2005). Varna, Bulgaria. September, 2005.
3. *Мелихов А. Н.* Ориентированные графы и конечные автоматы. М.: Наука, 1971. 416 с.

#### *Сведения об авторах*

**Иван Михайлович Гостев**

— д-р техн. наук; доцент; Московский государственный институт электроники и математики (Технический университет), кафедра кибернетики; E-mail: igostev@gmail.com

**Александр Глебович Подгорбунский**

— аспирант; Московский государственный институт электроники и математики (Технический университет), кафедра кибернетики; E-mail: shurikp@gmail.com

Рекомендована кафедрой  
вычислительной техники

Поступила в редакцию  
12.09.08 г.



Д. Е. Андрианов, С. С. Садыков, В. В. Фролов

## ФОРМАЛЬНОЕ ПРЕДСТАВЛЕНИЕ ХАРАКТЕРИСТИК ПРОСТРАНСТВЕННО-РАСПРЕДЕЛЕННЫХ ОБЪЕКТОВ

Рассматривается подход к формальному описанию данных о пространственно-распределенных объектах муниципальных геоинформационных систем. Предложено характеризовать пространственное положение объекта в трехмерном пространстве на основе семи топологических отношений, которые позволяют сформировать структуру из шестнадцати бинарных элементов. На основе этих формализаций возможно описать топологическое положение объекта в виде структур, позволяющих классифицировать пространственный объект.

**Ключевые слова:** пространственно-распределенный объект, топологическое отношение, муниципальные геоинформационные системы, матрица шестнадцати пересечений.

Для эффективного управления муниципальными образованиями и динамично развивающимися регионами необходимо иметь достоверные данные об объектах и протекающих на их территории процессах, а также передовые технологии накопления, обработки и представления информации. Современные географические информационные системы (ГИС) позволяют наглядно отобразить и осмыслить информацию о конкретных объектах, процессах и явлениях в их совокупности. Геоинформационные системы позволяют выявлять взаимосвязи и пространственные отношения, поддерживать коллективное использование данных и их объединение в единый информационный массив.

Определим виды пространственно-распределенных объектов и их взаимоотношения [1].

Точечный объект — это малоразмерный объект, который характеризуется координатами

$$p = (x, y) \in R^2 \quad p = (x, y) \in R^2, \quad (1)$$

линия описывается уравнением

$$f(x, y) = 0, \quad (2)$$

линейный сегмент задается координатами

$$ls = \{(p_1, p_2)\}, \quad (3)$$

где  $p_1 = (x_1, y_1)$ ,  $p_2 = (x_2, y_2)$  — точечные объекты.

Линейный объект представляет собой последовательность линейных сегментов:

$$t = \{(p_1, p_2), (p_2, p_3), \dots, (p_{k-1}, p_k)\} \quad (k=1, 2, \dots). \quad (4)$$

Полигональный объект — это двумерный объект, образованный замкнутой последовательностью линейных сегментов, т.е.

$$h = \{(p_1, p_2), (p_2, p_3), \dots, (p_k, p_1)\} \quad (k=1, 2, \dots). \quad (5)$$

На основании полученных выражений возможно описать тематический слой карты. Слоем карты назовем сочетание картографических объектов, выделенных из карты  $K$  по заданным условиям так, что

$$s = \{X_i \in K \mid X_i \text{ удовлетворяет заданным условиям}\} \quad (i=1, 2, \dots, n). \quad (6)$$

Например, из всех объектов карты можно выделить слои дорог, школ, жилых помещений, стадионов, водоемов, электросетей и т.д.

Между пространственными объектами существуют сложные взаимосвязи. Для формального описания связей будем использовать топологические отношения, так как они наиболее полно отражают взаимодействие объектов в пространстве [1].

Топологическим отношением между любыми двумя множествами  $X$  и  $Y$  называется такое отношение, которое при аффинных или топологических преобразованиях будет сохраняться.

Введем определения для формального описания следующих топологических отношений:

- 1) между частями пространственных объектов;
- 2) между пространственными объектами;
- 3) между слоями пространственных объектов.

1. Отношение, сохраняющееся между элементами  $x_1 \in X$  и  $x_2 \in X$  картографического объекта  $X$  при различных аффинных и топологических преобразованиях, назовем топологическим  $\varphi_{эл}$ . Топологическое отношение между элементами обозначается следующим образом:  $x_1 \varphi_{эл} x_2$ , т.е. элемент  $x_1$  имеет топологическое отношение  $\varphi_{эл}$  с элементом  $x_2$ . Следует указать, что противоположное отношение может и не иметь места.

2. Топологическим отношением  $\varphi_{об}$  между картографическими объектами  $X_1 \in K$  и  $X_2 \in K$  карты  $K$  будем называть отношение, которое сохраняется между данными объектами при различных аффинных и топологических преобразованиях. Топологическое отношение между объектами одного слоя обозначается следующим образом:  $X_1 \varphi_{об} X_2$  (картографический объект  $X_1$  имеет топологическое отношение  $\varphi_{об}$  с объектом  $X_2$ ). Противоположное отношение может и не иметь места.

3. Топологическим отношением  $\varphi_{сл}$  между слоями  $s_1 \in S$  и  $s_2 \in S$  назовем отношение, которое сохраняется между данными слоями при различных аффинных и топологических преобразованиях между объектами, находящимися в слоях  $s_1$  и  $s_2$  соответственно:  $s_1 \varphi_{сл} s_2$  (слой  $s_1$  имеет топологическое отношение  $\varphi_{сл}$  со слоем  $s_2$ ). Обратная запись может и не иметь места.

Рассмотрим следующие типы топологических отношений между картографическими объектами: соседство, изолированность, близость, вложенность и др. Эти топологические отношения наиболее полно отражают взаимодействие пространственных объектов.

*Отношение „Соседство“.* Картографический объект  $X_1 \in K$  находится в соседстве с объектом  $X_2 \in K$ , или между  $X_1$  и  $X_2$  установлено топологическое отношение „Соседство“  $\alpha$ , т.е.  $X_1 \alpha X_2$  тогда и только тогда, когда картографические объекты  $X_1$  и  $X_2$  имеют общую граничную точку или линию, т.е. если есть совпадение координат точек обоих объектов.

Отношение „Соседство“ выполняется ( $X_1 \alpha X_2$ ), если оно антирефлексивно, симметрично, транзитивно и пересечение картографических объектов  $X_1$  и  $X_2$  есть непустое множество, т.е.  $X_1 \cap X_2 = A$ ,  $A \neq \emptyset$  и непустое множество  $A$  должно состоять только из граничных точек картографических объектов  $X_1$  и  $X_2$ .

*Отношение „Изолированность“.* Картографический объект  $X_1 \in K$  изолирован от объекта  $X_2 \in K$ , или между  $X_1$  и  $X_2$  установлено топологическое отношение изолированности  $\delta$ , т.е.  $X_1 \delta X_2$  тогда и только тогда, когда объекты  $X_1$  и  $X_2$  не пересекаются друг с другом.

Отношение „Изолированность“ выполняется ( $X_1 \delta X_2$ ), если оно антирефлексивно, симметрично, транзитивно и при этом выполняется условие  $X_1 \cap X_2 = \emptyset$ .

*Отношение „Близость“.* Картографический объект  $X_1 \in K$  находится в близости к картографическому объекту  $X_2 \in K$ , или между  $X_1$  и  $X_2$  установлено топологическое отношение „Близость“  $\beta$ , т.е.  $X_1 \beta X_2$  тогда и только тогда, когда картографический объект  $X_1$  расположен на заданном расстоянии от картографического объекта  $X_2$ .

Отношение „Близость“ выполняется ( $X_1\beta X_2$ ), если оно антирефлексивно, симметрично, транзитивно и при этом выполняется следующее условие: картографические объекты  $X_1$  и  $X_2$  не должны пересекаться между собой, т.е.  $X_1 \cap X_2 = \emptyset$ . Минимальное расстояние между граничными точками объектов  $X_1$  и  $X_2$  не превышает заданного расстояния  $\rho_3$ , т.е.  $\min \rho(a, b) \leq \rho_3$ , где  $a, b$  — граничные точки картографических объектов  $X_1$  и  $X_2$  соответственно;  $\rho(a, b)$  — расстояние между граничными точками  $a$  и  $b$ .

Отношение „Вложенность“. Картографический объект  $X_1 \in K$  вложен в картографический объект  $X_2 \in K$ , или между  $X_1$  и  $X_2$  установлено топологическое отношение „Вложенность“  $\gamma$ , т.е.  $X_1\gamma X_2$  тогда и только тогда, когда все элементы объекта  $X_1$  находятся внутри объекта  $X_2$ .

Отношение „Вложенность“ выполняется ( $X_1\gamma X_2$ ), если оно рефлексивно, асимметрично, нетранзитивно и при этом  $X_2 \subseteq X_1$ .

Отношение „Пересечение“. Картографический объект  $X_1 \in K$  пересекается с картографическим объектом  $X_2 \in K$ , или между  $X_1$  и  $X_2$  установлено топологическое отношение „Пересечение“  $\chi$ , т.е.  $X_1\chi X_2$ .

Отношение „Пересечение“ выполняется ( $X_1\chi X_2$ ), если оно рефлексивно, симметрично, транзитивно и  $X_1 \cap X_2 = \emptyset$ .

Для более полного описания положения объекта городской инфраструктуры введем отношения, которые характеризуют геометрическое расположение пространственных объектов: параллельность и перпендикулярность.

Картографический объект  $X_1 \in K$  параллелен объекту  $X_2 \in K$ , или между  $X_1$  и  $X_2$  установлено геометрическое отношение „Параллельность“  $\eta$ , т.е.  $X_1\eta X_2$  тогда и только тогда, когда в объектах  $X_1$  и  $X_2$  можно найти по одному линейному сегменту, которые параллельны между собой, т.е.

$$(\exists x_k \in X_1, \exists x_l \in X_2): x_k \parallel x_l, \quad (7)$$

где  $x_k, x_l$  — линейные сегменты.

Объект  $X_1 \in K$  перпендикулярен картографическому объекту  $X_2 \in K$ , или между  $X_1$  и  $X_2$  установлено геометрическое отношение „Перпендикулярность“  $\mu$ , т.е.  $X_1\mu X_2$  тогда и только тогда, когда в объектах  $X_1$  и  $X_2$  можно найти по одному линейному сегменту, которые перпендикулярны друг другу, т.е.

$$(\exists x_k \in X_1, \exists x_l \in X_2): x_k \perp x_l. \quad (8)$$

При анализе карты города было выявлено, что часть пространственных объектов по различным признакам сгруппирована в отдельные структуры. Следует отметить, что в структуру могут входить объекты не только из одного, но и из разных слоев. В результате если формально описать такие структуры, то карта будет представлять собой совокупность структур с учетом взаимосвязей входящих в них объектов.

Топологическая структура — это совокупность картографических объектов, связанных между собой топологическими и геометрическими отношениями, она задается следующим образом:

$$T = (W, \Pi, \Phi), \quad (9)$$

где  $W = \{X_1, X_2, \dots, X_n\}$  — подсистема картографических объектов из разных слоев,  $n$  — количество объектов подсистемы;  $\Pi = \{\rho_k\}$  ( $k=1, 2, \dots, h$ ) — множество типов топологических и геометрических отношений в подсистеме  $W$ ,  $h$  — количество отношений;  $\Phi = [\varphi_{ij}^{(k)}]$  ( $i, j=1, 2, \dots, n$ ) — матрица топологических отношений между объектами из подсистемы  $W$ .

Элемент  $\varphi_{ij}^{(k)} = 0$ , если не существует взаимодействия между объектами  $X_i$  и  $X_j$ . Элемент  $\varphi_{ij}^{(k)}$  есть топологическое отношение  $k$ -го типа, если существует связь между слоями  $X_i$  и  $X_j$ .

Рассмотрим простейший пример расположения объектов в двумерном пространстве (рис. 1). Здесь показаны отношения объектов  $A$  и  $B1, B2, B3, B4$  (объекты  $A$  и  $B1$  не вложены друг в друга,  $A$  и  $B1$  — соприкасаются,  $A$  и  $B3$  — пересекаются,  $A$  и  $B4$  — один объект включает в себя другой).

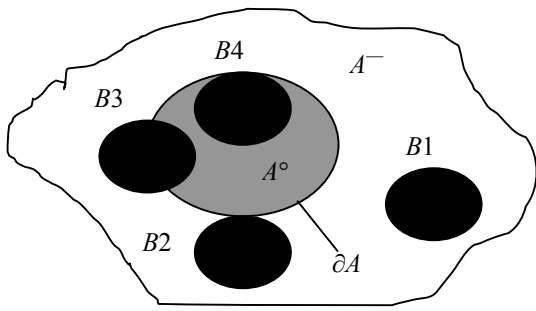


Рис. 1

Когда рассматривается отношение объектов трехмерном пространстве, необходимо анализировать и высотную составляющую объектов. На рис. 2 показаны возможные комбинации расположения объектов  $A$  и  $Bn$  по высоте, где  $n$  — один из возможных вариантов расположения объекта  $B$  (первый вариант — один объект включает в себя другой,

второй и седьмой — один объект находится на некотором удалении по высоте от другого, третий и шестой — касание границы, четвертый и пятый — пересечение границы [2]).

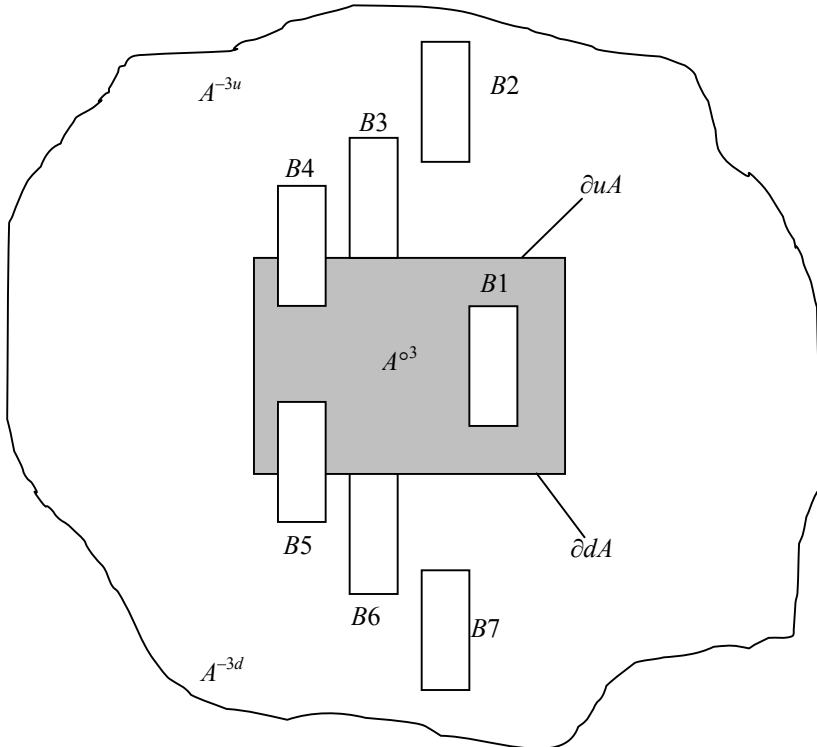


Рис. 2

В основу метода описания трехмерных топологических отношений положим математическую модель „девяти пересечений“ Эгенгофера [3], которая использует пересечения различных составных элементов объектов.

Часть точек пространства в множестве  $A$ , расположенная во внутренней части объекта и обозначенная  $A^\circ$ , является объединением всех открытых множеств в  $A$ . Замкнутое выражение  $A$ , обозначенное  $A^-$ , является пересечением всех замкнутых множеств  $A$ . Внешняя часть  $A$ , относящаяся к вложенному пространству  $R^2$ , обозначенному  $A^-$ , является множеством всех точек  $R^2$ , не содержащихся в  $A$ . Граница  $A$ , обозначенная  $\partial A$ , является пересечением замкнутого выражения  $A$  и замкнутого выражения внешней части  $A$ . Тем самым выполняется формальная характеристика расположения объектов на плоскости.

Для описания положения объекта в трехмерном пространстве введем дополнительные обозначения. Часть точек трехмерного пространства, расположенную во внутренней части объекта, обозначим как  $A^{\circ 3}$ . Внешняя часть объекта будет представлять собой две непересекающиеся области  $A^{-3u}$  (над объектом) и  $A^{-3d}$  (под объектом). Верхняя граница  $A$ , обозначенная  $\partial uA$ , является пересечением замкнутого выражения  $A$  и замкнутого выражения верхней внешней части  $A$ . И соответственно нижняя граница  $A$ , обозначенная  $\partial dA$ , является пересечением замкнутого выражения  $A$  и замкнутого выражения нижней внешней части  $A$ .

Таким образом, пространственная область определена как трехмерное множество точек, которое является гомеоморфным к пяти областям, т.е. каждая из восьми объектных частей области в трехмерном пространстве (внутренняя часть, граница и внешняя часть) — непустая и связанная.

Бинарное топологическое отношение между двумя областями  $A$  и  $B$  характеризуется как сравнение границы ( $\partial A$ ), внутренней части ( $A^\circ$ ), внешней части ( $A^-$ ) на горизонтальной плоскости и верхней, нижней границ ( $\partial uA, \partial dA$ ), внутренней части ( $A^{\circ 3}$ ), верхней и нижней внешних частей ( $A^{-3u}, A^{-3d}$ ) в вертикальной плоскости объекта  $A$  с границей ( $\partial B$ ), внутренней частью ( $B^\circ$ ), и внешней частью ( $B^-$ ) на горизонтальной плоскости и верхней, нижней границ ( $\partial uB, \partial dB$ ) внутренней части ( $B^{\circ 3}$ ), верхней и нижней внешних частей ( $B^{-3u}, B^{-3d}$ ) в вертикальной плоскости объекта  $B$ . Эти четырнадцать объектных частей объединены так, что между ними существует шестнадцать пересечений, которые представляют топологические отношения между этими двумя объектами.

Естественно, это не все возможные комбинации топологических характеристик, но рассмотрев рис. 2, будем считать, что этими отношениями можно описать взаимодействие объектов в дополнение к плоскости в трехмерном пространстве.

Топологическое отношение между областями  $A$  и  $B$  кратко может быть представлено как матрица  $4 \times 4$ , назовем ее „16 пересечений“.

$$R_{16}(A, B) = \begin{bmatrix} \partial A \cap \partial B & \partial A \cap B^\circ & \partial A \cap B^- & A^{-3d} \cap B^{\circ 3} \\ A^\circ \cap \partial B & A^\circ \cap B^\circ & A^\circ \cap B^- & A^{-3u} \cap B^{\circ 3} \\ A^- \cap \partial B & A^- \cap B^\circ & A^- \cap B^- & \partial uA \cap B^{\circ 3} \\ \partial uA \cap \partial dB & \partial dA \cap \partial uB & \partial dA \cap B^{\circ 3} & A^{\circ 3} \cap B^{\circ 3} \end{bmatrix}.$$

Топологические отношения характеризуются топологическими инвариантами „16 пересечений“, т.е. свойствами, которые сохраняются при топологических преобразованиях. Сохранение этих шестнадцати пересечений было идентифицировано как самый простой и общий топологический инвариант, хотя другие могут быть полезны как компоненты пересечения и их измерения. Инвариант содержания характеризует каждое из этих шестнадцати пересечений значением „пустое“ ( $\emptyset$ ) или „непустое“ ( $\neg \emptyset$ ). С различием „пустое/непустое“ этих шестнадцати пересечений потенциально можно получить 216 различных топологических отношений. Два из этих топологических отношений сохраняются между любыми двумя областями. Шестнадцать пересечений „пустое/непустое“ описывают набор отношений, которые обеспечивают полный охват — три объектные части: граница, внутренняя часть и внешняя часть.

Содержания их пересечений такие, что любое множество является или пустым, или непустым. Фактическое число возможных отношений зависит от измерения пространства относительно объектов и от топологических свойств объектов, вложенных в это пространство. Например, граница линии (нециклической), т.е. множество ее начальных и конечных точек, является разделенной, тогда как граница области без отверстий является соединенной, и различие в этих топологических свойствах влияет на то, какие топологические отношения могут быть реализованы.

Для хранения в памяти одного топологического отношения между картографическими объектами достаточно шестнадцати бит. Формализованная выше структура будет состоять из набора бинарных таблиц, достаточно простых в хранении и анализе.

#### СПИСОК ЛИТЕРАТУРЫ

1. Еремеев С. В., Садыков С. С. Автоматический контроль размещения пространственных объектов на цифровой карте с использованием топологических отношений // Информационные технологии. 2005. № 8. С. 6—9.
2. Андрианов Д. Е. Создание метода представления топологических отношений в трехмерном пространстве для задач городских ГИС // Геоинформатика. М.: ВНИИгеосистем, 2007. № 2. С. 1—3.
3. Egenhofer M. J. and Franzosa R. D. Point-set topological spatial relations // Int. J. of Geographical Inf. Systems. 1991. Vol. 5, N 2. P. 161—176.

#### *Сведения об авторах*

- Дмитрий Евгеньевич Андрианов** — канд. техн. наук, доцент; Муромский институт (филиал) Владимирского государственного университета, кафедра информационных систем; E-mail: AndrianovDE@inbox.ru
- Султан Садыкович Садыков** — д-р техн. наук, профессор; Муромский институт (филиал) Владимирского государственного университета, кафедра информационных систем; E-mail: SadykovSS@yandex.ru
- Владислав Валерьевич Фролов** — аспирант; Муромский институт (филиал) Владимирского государственного университета, кафедра информационных систем; E-mail: AndrianovDE@inbox.ru

Рекомендована кафедрой  
информационных систем

Поступила в редакцию  
12.09.08 г.

А. В. ОГНЕВ, А. П. ТИПИКИН

## АЛГОРИТМ ИНВАРИАНТНОГО РАСПОЗНАВАНИЯ ОТПЕЧАТКОВ ПАЛЬЦЕВ ПО КЛЮЧЕВЫМ ТОЧКАМ

Рассматривается алгоритм, позволяющий существенно снизить зависимость вероятности распознавания отпечатков пальцев (ОП) от смещений, поворотов, фрагментации и дефектов поверхности папиллярного узора. Скорость распознавания увеличивается за счет введения этапа центрирования ОП с поиском наиболее достоверных пар базовых отрезков в сопоставляемых ОП, а также за счет сравнительной простоты распараллеливания алгоритмов распознавания, основанных на метрике Хаусдорфа, позволяющей находить расстояния между неравномоными множествами признаков.

**Ключевые слова:** идентификация отпечатка пальца, инвариантность, оцифровка, центрирование.

В современной криминалистике широко используются программы распознавания отпечатков пальцев (ОП). Для достижения высокой степени вероятности распознавания как на начальных, так и на завершающих стадиях работы программы требуется участие человека, и чем больше база данных ОП и выше вероятность повреждения отпечатков, тем чаще. Рассматриваемый алгоритм позволяет повысить степень автоматизации процесса распознавания, а также скорость распознавания на основе привлечения больших вычислительных ресурсов и ускорения отдельных процедур путем параллельной обработки информации.

Разработка алгоритмов распознавания ОП идет очень активно [1], но любой из существующих имеет свои недостатки: возможны ошибки false-reject rate (ошибочный отказ) и false-alarm rate (ошибочное подтверждение), возникающие из-за образующихся при сканировании смещений, поворотов, фрагментации, а также дефектов поверхности папиллярного узора. Избежать этих недостатков можно, если применять алгоритмы, инвариантные к названным негативным факторам. При решении задачи верификации (подтверждения личности) требуются мобильные датчики и вычислительные устройства (ВУ) со сравнительно небольшой тактовой частотой процессоров и объемом памяти. При решении задачи идентификации (определения личности) используются большие базы данных (БД), содержащие информацию об ОП и их обладателях, многократно повторяется процедура распознавания, поэтому возрастают требования к повышению производительности ВУ. В последнем случае целесообразно использовать параллельные алгоритмы сравнения и выполнять их на многопроцессорных ВУ. Так как в обоих случаях время принятия решения ограничено, вычислительная сложность инвариантного алгоритма не должна быть слишком большой, а для снижения требуемой емкости памяти желательно не хранить полное описание изображения ОП, а оперировать только с его моделями (множествами ключевых точек). Этого требуют соображения безопасности для того, чтобы при хищении из БД отпечатков ими нельзя было воспользоваться.

Известны два основополагающих алгоритма распознавания отпечатков пальцев: по отдельным деталям (ключевым точкам, КТ — развилкам и окончаниям, называемым минуциями) и по рельефу всей поверхности пальца [2]. В первом случае устройство регистрирует только некоторые точки, уникальные, свойственные конкретному отпечатку, и определяет их взаимное расположение. Во втором случае обрабатывается изображение всего отпечатка.

Алгоритм инвариантного распознавания отпечатков пальцев по ключевым точкам состоит из трех этапов: предварительная обработка изображения, центрирование, распознавание.

На этапе предварительной обработки каждое изображение ОП должно быть бинаризовано и отфильтровано. На этом этапе в черно-белом пиксельном изображении ОП устраняется часть ложных пропечаток, папиллярные линии утончаются до значения в один пиксел, логически устраняются некоторые их короткие ложные ответвления и небольшие разрывы. В этом изображении должны быть найдены граничные точки, определяющие контур информативной области ОП, его центр, а также выделены и описаны КТ в первичной (декартовой) системе координат относительно центра.

В настоящее время при использовании многих известных алгоритмов сравнения ОП сталкиваются с проблемой не вполне точного центрирования отпечатков [3]. Центрирование необходимо для ускорения процесса сравнения. Известно множество методов центрирования, но все они уязвимы к аффинным трансформациям (поворотам и смещениям) и дефектам папиллярного рисунка (фрагментации, порезам, повреждениям и т.д.) и не обеспечивают высокой точности [4]. Можно выделить два основных подхода: нахождение пиксельного центра тяжести и определение центра информативной области ОП. Если за центры сравниваемых ОП принимаются центры тяжести всех пикселей папилляров каждого отпечатка, то вследствие неправильного расположения пальца на считывающем устройстве при снятии отпечатка возникают неточности при фрагментации ОП.

Выделение КТ и вычисление их координат выполняется в первичной системе координат относительно неточного центра ОП путем просмотра каждого пикселя отпечатка и его ближайшей окрестности. Если в этом пикселе папилляр разветвляется или оканчивается, он принимается за КТ.

Широко применяемым инвариантным алгоритмом распознавания ОП является метод корреляционного сравнения [4]. Сопоставление двух ОП основано на многократных сравнениях множеств КТ и выполняется со скоростью 200—3000 сравнений в секунду [5], что приводит к значительным затратам времени. При сопоставлении перебираются пары КТ по принципу „каждая с каждой“. В окрестности каждой КТ проводится поиск ближайшей КТ другого отпечатка. Если она расположена на допустимом удалении, то эти две точки считаются совпавшими. В качестве меры близости двух ОП принимается число пар их КТ, признанных совпавшими. При сопоставлении двух ОП за достоверный центр одного из ОП принимается поочередно каждый пиксел его центральной области [6], размер которой определяется величиной возможного смещения ОП при сканировании и может составить до 10 % от длины рамки изображения ОП.

Перед каждым сопоставлением вносятся небольшой угол поворота одного из ОП и смещение его центра в другой пиксел центральной области и вновь вычисляется мера близости двух ОП. Повторные сопоставления двух ОП продолжаются до тех пор, пока не будут перебраны все возможные сочетания пикселей центральной области и возможных углов поворота одного из ОП, например, с шагом в  $1^\circ$  в диапазоне  $\pm 15^\circ$ . Решение об идентификации принимается по наибольшему значению меры близости из всех вариантов сопоставлений, если оно не меньше заданного порога. Число парных сравнений КТ двух ОП, требуемое для принятия решения, равно

$$N_{\text{ср}} = N_d N_{\text{ЦО}} N_{\text{КТ}}^2,$$

где  $N_d$  — число шагов поворота одного из ОП;  $N_{\text{ЦО}}$  — число пикселей центральной области;  $N_{\text{КТ}}$  — число КТ в одном из ОП.  $N_d = 30$  при шаге  $1^\circ$  и диапазоне  $\pm 15^\circ$ ;  $N_{\text{ЦО}} = 60 \cdot 40 = 2400$  пикселей при растре  $600 \times 400$  пикселей;  $N_{\text{КТ}} = 70$  точек. Поэтому требуемое число парных сравнений точек  $N_{\text{ср}} \approx 3,5 \cdot 10^8$ . Это приводит к большим затратам времени при реализации данного способа сопоставления на современных компьютерах.

Рассматриваемый алгоритм позволяет существенно снизить зависимость вероятности распознавания ОП от смещений, поворотов, фрагментации и дефектов поверхности папиллярного узора, а также распараллелить их многовариантное сопоставление для повышения



скорости распознавания. Применяемые в алгоритме процедуры составления избыточных векторных описаний изображения ОП и параллельного их сопоставления позволяют автоматизировать формирование БД эталонов ОП и повысить вероятность их инвариантного распознавания. Инвариантность по отношению к смещениям, поворотам, фрагментации и наиболее вероятным дефектам изображений ОП достигается путем многовариантного сопоставления с использованием метрики Хаусдорфа [1] их избыточных векторных описаний в полярной системе координат относительно базовых отрезков (БО).

Скорость инвариантного распознавания ОП увеличивается за счет дополнительного предварительного этапа поиска наиболее достоверных пар базовых отрезков в сопоставляемых отпечатках. После ранжирования пар базовых отрезков можно сократить требуемое число вариантов сопоставления векторных описаний множеств минутий до трех, а число шагов попарной обработки минутий двух отпечатков — до  $10^4$ — $10^5$  [6]. Дополнительным резервом повышения скорости является сравнительная простота распараллеливания алгоритмов распознавания, основанных на метрике Хаусдорфа, позволяющей находить расстояния между неравномоощными множествами признаков [7].

Требуемое число вариантов сравнения ОП по ключевым точкам может быть уменьшено на основе следующей методики центрирования [6]. Для достижения желаемой инвариантности необходимо перейти от „грубой“ (декартовой) к „новой“ (полярной) системе координат, центром которой будет являться центр достоверного базового отрезка. Для повышения эффективности работы алгоритма из рассмотрения исключаются КТ, близкие к границе информативной области („зашумленность“ изображения ОП при приближении к его краям повышается), и к центру ОП (из-за неточности расположения центра ОП). БО формируется только из оставшихся внутренних точек окончаний и разветвлений папилляров.

Минимального числа БО с сохранением единообразия их выбора в паре сравниваемых ОП можно достичь по следующему правилу обхода точек. Вращая по часовой стрелке радиус-вектор из центра первичной системы координат, последовательно соединить отрезками прямых встречающиеся КТ. Если несколько КТ встречаются в одном направлении радиуса-вектора, то предыдущая точка соединяется с самой удаленной от центра. При этом остальные КТ соединяются между собой последовательно по мере приближения к центру. Таким образом, формируется замкнутый контур БО, представляющий собой ломаную с вершинами в КТ.

После составления контуров БО необходимо в обоих ОП выделить пары БО, близкие по длине и взаимной ориентации. Последняя определяется значениями углов с четырьмя соседними БО в контуре (см. рисунок). Так как могут выпадать соседние БО или появляться ложные отрезки, то определить достоверную пару БО можно следующим образом. Сначала определим, в каком из ОП имеется меньшее число БО, входящих в ломаные. Этот отпечаток  $A$  используется для минимизации числа проверяемых пар БО. Для каждого БО отпечатка  $A$  необходимо выбрать такой парный ему БО из второго отпечатка  $B$ , для которого будет минимальной мера  $C$ :

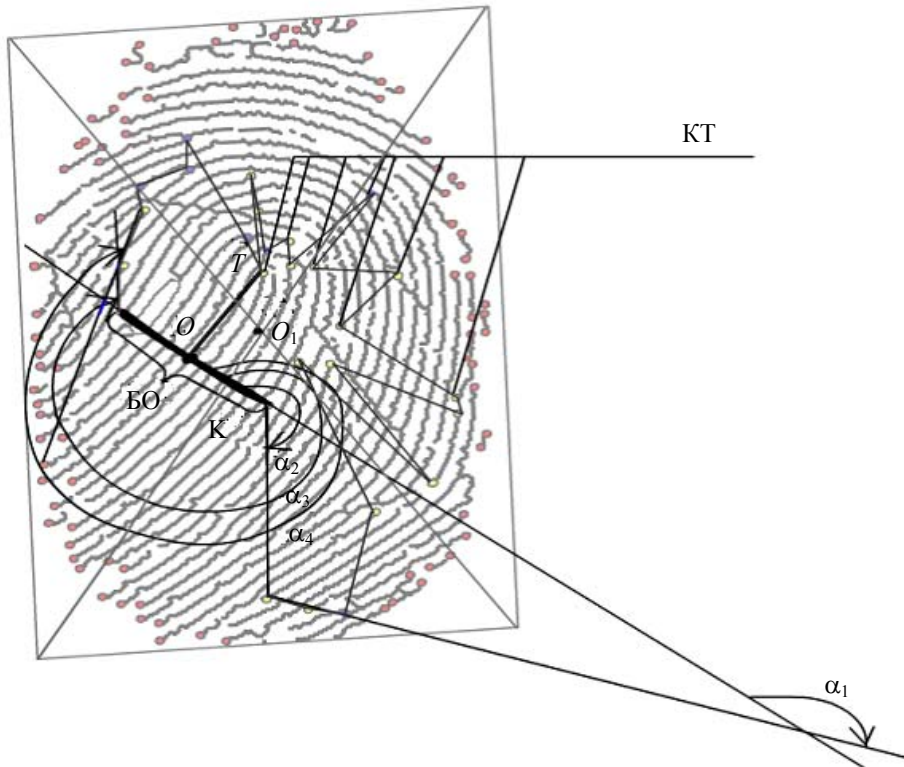
$$\begin{aligned}
 C_1 &= \frac{l_A - l_B}{L} + \frac{a_{A_1} - a_{B_1}}{\Lambda}, \\
 C_2 &= \frac{l_A - l_B}{L} + \frac{a_{A_2} - a_{B_2}}{\Lambda}, \\
 C_3 &= \frac{l_A - l_B}{L} + \frac{a_{A_3} - a_{B_3}}{\Lambda}, \\
 C_4 &= \frac{l_A - l_B}{L} + \frac{a_{A_4} - a_{B_4}}{\Lambda}, \\
 C &= \min\{C_1, C_2, C_3, C_4\},
 \end{aligned} \tag{1}$$

где  $l$  — длина БО;  $a$  — угол с соседним БО в пределах одного и того же ОП;  $L$  — максимальная длина БО в обоих ОП;  $\Lambda$  — максимально возможная величина угла, равная  $2\pi$ .

В качестве центра  $O$  новой полярной системы координат в одном из ОП принимается центр достоверного БО (см. рисунок) той пары БО, для которой будет значение  $C$  наименьшим. Достоверная пара БО выбирается по следующей формуле:

$$C_{\min} = \min_i \left\{ \min_j \left\{ \min_K \{C\}_{K=1}^{K=4} \right\}_{j=1}^{j=N_B} \right\}_{i=1}^{i=N_A},$$

где  $N_A, N_B$  — число БО в отпечатках  $A$  и  $B$  соответственно;  $K$  — число соседних БО, используемых при нахождении меры  $C$  (1), где  $K = 4$ .



Учитывая возможную близость между собой по значению  $C$  нескольких первых проанжированных пар БО, в дальнейшем величины метрики Хаусдорфа целесообразно вычислять для первых трех пар БО, а расстояния между ними выбирать минимальными.

Векторные описания множеств КТ формируются в полярной системе координат с учетом расстояния от центра  $O$  достоверного БО до данной КТ, например до точки  $T$  (см. рисунок), и угла между лучом, проведенным из центра  $O$  в первый ( $K$ ) конец БО, и лучом, проведенным из центра БО в данную КТ. Перевод из первичной (декартовой,  $O_1$  — ее центр) в новую полярную систему координат осуществляется по следующим формулам:

$$L = \sqrt{(X_T - X_O)^2 + (Y_T - Y_O)^2},$$

$$\alpha = \arccos \frac{(X_T - X_O)^2 + (Y_T - Y_O)^2 + (X_K - X_O)^2 + (Y_K - Y_O)^2 - (X_K - X_T)^2 - (Y_K - Y_T)^2}{2\sqrt{(X_T - X_O)^2 + (Y_T - Y_O)^2} \sqrt{(X_K - X_O)^2 + (Y_K - Y_O)^2}},$$

где  $X_T, Y_T$  — абсцисса и ордината данной КТ в первичной системе координат;  $X_O, Y_O$  — абсцисса и ордината центра достоверного БО в первичной системе координат.

После того как сформированы описания множеств КТ относительно центров достоверных БО в обоих ОП, для каждого элемента множеств КТ, соответствующих отпечаткам  $A$  и  $B$ , вычисляются минимальные расстояния от всех элементов другого множества:

$$e_k^A = \min_t \left\{ d \left( (\alpha_k^A, l_k^A), (\alpha_t^B, l_t^B) \right) \right\}_{t=1}^{t=N_B}, \quad k=1, \dots, N_A,$$

$$e_t^B = \min_k \left\{ d \left( (\alpha_t^B, l_t^B), (\alpha_k^A, l_k^A) \right) \right\}_{k=1}^{k=N_A}, \quad t=1, \dots, N_B, \quad (2)$$

где  $d(x, y)$  — расстояние между элементами множеств  $x$  и  $y$ ;  $t, k$  — номер КТ в множествах  $A$  и  $B$  соответственно.

Затем вычисляются максимальные значения из множеств полученных минимумов:

$$e_{\max}^A = \min_k \{ e_k^A \},$$

$$e_{\max}^B = \min_t \{ e_t^B \}. \quad (3)$$

После этого вычисляются максимум и минимум из двух полученных величин (3):

$$e_{\max i,j} = \max (e_{\max}^A, e_{\max}^B),$$

$$e_{\min i,j} = \min (e_{\max}^A, e_{\max}^B), \quad (4)$$

где  $i, j$  — номера выбранных достоверных БО в  $A$  и  $B$  соответственно.

Минимум (см. выражение (4)) находится для того, чтобы определить степень включения одного множества в другое. Такая модификация алгоритма сопоставления неравномоощных множеств на основе метрики Хаусдорфа позволяет численно определить степень идентичности фрагментов пары ОП.

Решение об идентичности ОП принимается на основе следующего анализа значений  $e_{\max i,j}$  и  $e_{\min i,j}$  (4). Если  $e_{\max i,j} \leq h$  и  $e_{\min i,j} \leq h$ , то ОП идентичны. Если  $e_{\max i,j} > h$  и  $e_{\min i,j} > h$ , то ОП принадлежат разным людям. Если  $e_{\max i,j} > h$  и  $e_{\min i,j} \leq h$ , то фрагменты сопоставляемых ОП близки. Величина порога  $h$  выбирается пропорциональной среднему значению меры  $C$  (1) и может быть подобрана опытным путем на тестовых парах сопоставляемых ОП. Вычислительная сложность описанного алгоритма инвариантного распознавания ОП определяется в основном необходимостью трехкратного повторения расчетов по формулам (2)—(4). Затраты времени на остальные описанные выше процедуры составляют не более 4 %. Сопоставление двух ОП выполняется примерно за  $3,5 \cdot 10^4$  шагов, каждый из которых эквивалентен вычислению одного расстояния между двумя КТ и по длительности не превышает затрат времени на 10 парных сравнений КТ в корреляционном методе. Таким образом, скорость инвариантного распознавания возрастает практически в 1000 раз.

Преимуществами данного алгоритма являются повышение степени инвариантности к поворотам, смещениям, фрагментации и дефектам ОП, а также возможность ускорения процесса распознавания за счет сравнительной простоты распараллеливания вычислений при центрировании ОП (1) и нахождении расстояний между сравниваемыми ОП по формулам (2)—(4).

#### СПИСОК ЛИТЕРАТУРЫ

1. Биометрия. Отпечаток пальца [Электронный ресурс]: <<http://www.biometrics.ru>>.
2. Рябов Г. В. Современные технологии идентификации личности по отпечатку пальца с использованием емкостных датчиков [Электронный ресурс]: <[http://www.radioradar.net/articles/scientific\\_technical](http://www.radioradar.net/articles/scientific_technical)>.

3. Яковлев А. В., Румянцев Р. В. Разработка алгоритма идентификации личности по изображению отпечатка пальца // Методы и системы обработки информации. Сб. науч. ст. Ч. 2 / Под ред. С. С. Садыкова, Д. Е. Андрианова. М.: Горячая линия – Телеком, 2004. С. 41—48.
4. Кухарев Г. А. Биометрические системы: Методы и средства идентификации личности человека. СПб: Политехника, 2004.
5. Пат. № 2310910 РФ. Способ верификации и идентификации отпечатков папиллярных узоров / А. А. Борейшо, Д. А. Фокин, С. Я. Чакчир, А. Ю. Хозин, Е. М. Орлов. 16.05.2006.
6. Огнев А. В., Типикин А. П. Центрирование отпечатков пальцев при инвариантном распознавании на основе метрики Хаусдорфа // Опτικο-электронные приборы и устройства в системах распознавания образов, обработки изображений и символической информации. Сб. мат. VIII Междунар. конф. „Распознавание – 2008“. Ч. 2. Курск: КурскГТУ, 2008. С. 34—35.
7. Ognev A. V., Tipikin A. P. Method of invariable recognition of fingerprints based on metric of Hausdorff // Proc. 6th Int. Conf. „Information and telecommunication technologies in intelligent systems“. Crete, Greece. June 2—6, 2008. P. 78—82.

**Сведения об авторах**

**Антон Вадимович Огнев**

— магистрант; Курский государственный технический университет, кафедра вычислительной техники; E-mail: light@kursknet.ru

**Александр Петрович Типикин**

— д-р техн. наук, профессор; Курский государственный технический университет, кафедра вычислительной техники; E-mail: cct@cafct.kstu.kursk.ru

Рекомендована кафедрой  
вычислительной техники

Поступила в редакцию  
12.09.08 г.

Э. И. ВАТУТИН, И. В. ЗОТОВ, В. С. ТИТОВ

## ВЫЯВЛЕНИЕ ИЗОМОРФНЫХ ВХОЖДЕНИЙ $R$ -ВЫРАЖЕНИЙ ПРИ ПОСТРОЕНИИ ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ ЛОГИЧЕСКОГО УПРАВЛЕНИЯ

Представлен алгоритм выяснения отношения изоморфизма  $R$ -выражений (сечений параллельного алгоритма), основанный на ряде их особых свойств и ориентированный на аппаратную реализацию. Приведено описание устройства (акселератора) на его основе, позволяющего проводить проверку отношения изоморфизма за линейное время.

*Ключевые слова:* логический мультиконтроллер, синтез, параллельный алгоритм логического управления, разбиение, оптимизация, ориентированные деревья.

Реализация параллельных управляющих алгоритмов в базисе логических мультиконтроллеров (ЛМК) требует их декомпозиции на множество частных алгоритмов ограниченной сложности [1]. Получение оптимального набора частных алгоритмов (разбиения) — сложная комбинаторная задача. Качество ее решения существенно влияет на аппаратную сложность ЛМК и определяет, в конечном счете, время выполнения алгоритма. Один из наиболее эффективных путей решения данной задачи предлагает развиваемый авторами параллельно-последовательный метод декомпозиции [2—8]. Как было показано в статье [9], он позволяет формировать наиболее близкие к оптимальным варианты разбиения с учетом основных структурных и технологических ограничений базиса ЛМК.

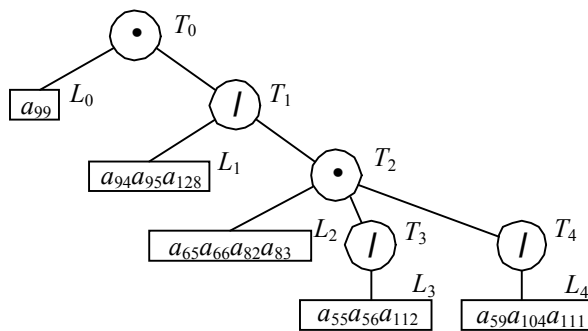
Один из ключевых этапов параллельно-последовательной декомпозиции — построение множества сечений, покрывающего все вершины исходного алгоритма. Формирование сечений осуществляется путем выполнения трудоемких операций подстановки над множеством так называемых  $R$ -выражений, описывающих алгоритм управления. Как показывают исследования, упрощение и ускорение этих операций возможно путем их сведения к действиям над деревьями, в частности, к проверке изоморфизма.

В настоящей статье предложен алгоритм определения изоморфных вхождений  $R$ -выражений, основанный на ряде их специфических свойств, не присущих графам или деревьям общего вида, и обеспечивающий проверку изоморфизма за полиномиальное время. (Как известно [10], для решения задачи распознавания изоморфизма графов общего вида до сих пор не только не придумано эффективного универсального алгоритма, но и не доказана ее принадлежность к классу  $P$  или  $NP$  [11].) Следует отметить, что рассматриваемый вид изоморфизма  $R$ -выражений определяется исходя из возможности проведения операции подстановки [1, 3, 4] и несколько отличается от „классического“ понятия изоморфизма [10], определяемого для графов, ввиду чего далее будем именовать его  $r$ -изоморфизмом.

При практической реализации операций над  $R$ -выражениями удобным является их представление в виде деревьев, допускающее преобразование в табличный вид (рис. 1). Каждый элемент дерева  $X$ , представленного совокупностью наборов листьев  $L_1^X, L_2^X, \dots, L_{n_L}^X(X)$ , узлов  $T_1^X, T_2^X, \dots, T_{n_T}^X(X)$  и связей между ними, кодируется набором полей. С учетом ряда особенностей обработки наборы листьев и узлы дерева кодируются отдельно. Узлы дерева представлены полями:

- 1) типа узла (ТУ) — параллельный или альтернативный;
- 2) ссылки на предка (СП) — номер узла-предка;
- 3) номера соответствия (НС) — номер изоморфного эквивалента в соседнем дереве;
- 4) типа соответствия (ТС) — может принимать значения „0\*“ — соответствие отсутствует, „10“ — неполное (частичное) соответствие, „11“ — полное соответствие.

$a_{99} \cdot (a_{94} | a_{95} | a_{128} | (a_{65} \cdot a_{66} \cdot a_{82} \cdot a_{83} \cdot (a_{55} | a_{56} | a_{112}) \cdot (a_{59} | a_{104} | a_{111})))$



Узлы	Т0	Тип узла	Ссылка на предка	Текущее количество узлов ( $N_T$ )
	T0	•	—	5
	T1		T0	
	T2	•	T1	
	T3		T2	
	T4		T2	
T5	—	—	Текущее количество наборов листьев ( $N_L$ )	
...	...	...		5
$N_{L_{max}}$	—	—		

Наборы листьев	№	Множество вершин	Ссылка на предка
	$L_0$	99	$T_0$
	$L_1$	94, 95, 128	$T_1$
	$L_2$	65, 66, 82, 83	$T_2$
	$L_3$	55, 56, 112	$T_3$
	$L_4$	59, 104, 111	$T_4$
	$L_5$	—	—
...	...	...	
$N_{L_{max}}$	—	—	

Рис. 1

Наборам листьев дерева при этом соответствуют поля множества вершин (МВ) — двоичный вектор с единичными битами в позициях, соответствующих номерам присутствующих в наборе вершин, а также поля СП, НС и ТС. Во избежание путаницы при обозначении одноименных полей, соответствующих узлам и наборам листьев, будем обозначать их с указанием принадлежности к элементам дерева (например, СП(у) и СП(л)).

Краткое описание свойств  $R$ -выражений, гарантирующих наличие единственно возможного изоморфного эквивалента поддерева и позволяющих исключить из рассмотрения проверку соответствия типов узлов деревьев, приведено ниже (ввиду ограниченного объема статьи доказательства лемм и теорем не приведены).

*Необходимое условие 1 отсутствия  $r$ -изоморфизма.* Если в дереве  $A$  найдется набор листьев  $L_i^A$ , не находящийся в отношении эквивалентности, обозначаемом как

$$L_1 [\sim] L_2 \Leftrightarrow (L_1 = L_2) \vee (L_1 \subset L_2) \vee (L_2 \subset L_1),$$

ни с одним набором листьев дерева  $B$ , то дерево  $A$  не является  $r$ -изоморфным дереву  $B$ :

$$\exists L_i^A \in A, \forall L_j^B \in B, L_i^A [\nexists] L_j^B \rightarrow A [\nexists] B.$$

*Необходимое условие 2 отсутствия  $r$ -изоморфизма.* Если в деревьях  $A$  и  $B$  присутствует более одной пары наборов листьев с частичным соответствием ( $L_i^A \subset L_j^B$ , где  $L_i^A$  —  $i$ -й набор листьев дерева  $A$ ), рассматриваемые деревья не  $r$ -изоморфны.

*Аксиома 1.* Невозможно найти такие два набора листьев  $L_i^A$  и  $L_j^A$ , предком которых являлся бы один и тот же узел.

*Аксиома 2.* При движении по узлам любой ветви дерева типы узлов строго чередуются.

*Лемма 1* (об ортогональности наборов листьев в составе дерева). В пределах дерева все наборы листьев ортогональны:  $L_i^A \cap L_j^A = \emptyset, \forall i \neq j$ .

*Лемма 2* (о совпадении типа узлов-предков для вершин в составе различных наборов листьев). Любые две вершины, входящие в состав одного набора листьев  $L_i^X$  дерева  $X$ , могут одновременно войти в состав набора листьев  $L_j^Y$  дерева  $Y$  только в том случае, если тип узла предков наборов листьев  $L_i^X$  и  $L_j^Y$  совпадает.

*Следствие.* У полностью или частично эквивалентных наборов листьев не может быть предков разного типа.

*Теорема 1* (о единственности  $r$ -изоморфной пары наборов листьев). Набору листьев  $L_i^A$  может соответствовать не более одного полностью или частично эквивалентного набора листьев  $L_j^B$ :  $L_i^A [\sim] L_j^B : \exists L_k^B, L_i^A [\sim] L_k^B, k \neq j$ .

*Теорема 2* (о единственности  $r$ -изоморфной пары поддеревьев). В деревьях  $A$  и  $B$  не может быть более одной пары совпадающих поддеревьев.

*Следствие.* Дереву  $A$  может быть изоморфно не более одного поддерева из дерева  $B$ .

Приведенные выше аксиомы, леммы, теоремы и необходимые условия позволяют сформулировать алгоритм выявления  $r$ -изоморфизма пары деревьев  $A$  и  $B$ , ориентированный на параллельную аппаратную реализацию, в следующем виде.

1. Установить значения полей ТС всех наборов листьев дерева  $A$  в „00“ (соответствия нет), значения полей НС всех наборов листьев дерева  $A$  в „11...1“ (ссылка на несуществующий элемент дерева  $B$ ), значения полей ТС всех узлов дерева  $A$  в „11“ (полное соответствие), значения полей НС всех узлов дерева  $A$  в „11...1“.

2. Если  $n_L(A) > n_L(B)$  (количество наборов листьев в подставляемом дереве больше количества наборов листьев в объемлющем), установить признак  $\varphi = 0$  отсутствия изоморфного дереву  $A$  поддерева в составе дерева  $B$ . Перейти к п. 7.

3. Поочередно выбирать все наборы листьев дерева  $B$ . Для выбранного набора листьев  $L_j^B$  дерева  $B$  осуществить параллельное во времени сравнение поля МВ с полями МВ всех наборов листьев дерева  $A$ , сформировать признаки полного  $\delta^+ = (L_i^A = L_j^B)$  и частичного  $\delta^- = (L_i^A \subseteq L_j^B)$  соответствия наборов листьев с их последующим сохранением в поле ТС набора листьев  $L_i^A$  в формате  $[\delta^- | \delta^+]$  (на позиции старшего бита — признак  $\delta^-$ , младшего —  $\delta^+$ ) и в поле ТС предка набора листьев  $L_i^A$ . В случае полного или частичного соответствия между наборами листьев  $L_i^A$  и  $L_j^B$  сохранить номер  $j$ , соответствующий предположительно изоморфному эквиваленту  $L_i^A$  в составе дерева  $B$ , в поле НС набора листьев  $L_i^A$ , а также значение

поля СП набора листьев  $L_j^B$  в поле НС предка набора листьев  $L_i^A$  (предположительно изоморфный эквивалент предка набора листьев  $L_i^A$ ).

4. Если хотя бы для одного набора листьев дерева  $A$  не нашлось полностью или частично эквивалентного предположительно изоморфного набора листьев в составе дерева  $B$ , т.е.

$$\left( \exists L_k^A : \delta_k^- = 0 \Leftrightarrow \lambda = \bigwedge_{i=0, N_L(A)-1} \delta^-(L_i^A) \neq 1 \right),$$

установить признак  $\varphi = 0$  отсутствия изоморфного дерева  $A$  поддерева в дереве  $B$ . Перейти к п. 7.

5. Просмотреть все узлы дерева  $A$ , кроме корня, в направлении от узлов с большим номером к узлам с меньшим номером. Откорректировать значения поля ТС предка каждого  $i$ -го узла как  $ТС^{\uparrow'} = f(ТС^{\downarrow}, ТС^{\uparrow})$  в случае наличия предка у предположительно изоморфного эквивалента текущего узла ( $\tilde{\gamma} = 0$ ), установить  $ТС^{\uparrow'} = 00$  в противном случае ( $\gamma = 1$ ). (Здесь  $f(x, y)$  — функция корректировки значения поля ТС узла-предка,  $x$  — значение поля ТС узла-предка,  $y$  — узла-потомка, эквивалентная функции минимума двух аргументов для всех случаев, кроме  $f(10, 10) = f(10, 11) = 00$ .) Задать значения поля НС предка рассматриваемого узла  $a[i]$  как  $НС^{\uparrow} = b[a[i].НС].СП$ , где  $НС^{\uparrow}$  — значение поля НС узла-предка текущего узла.

Если поля ТС всех узлов дерева  $A$  имеют значение „11“ или „10“, т.е.

$$\left( \varphi_0 = \bigwedge_{i=0, N_T(A)-1} \delta^-(T_i^A) = 1 \right),$$

в дереве  $B$  есть поддерево, изоморфное дереву  $A$ , причем подстановка изоморфизма определяется значениями полей НС ( $i$ -му элементу дерева  $A$  соответствует изоморфный эквивалент  $a[i].НС$  в дереве  $B$ ). В противном случае в составе дерева  $B$  нет поддерева, изоморфного дереву  $A$ . Установить признак  $\varphi = \varphi_0$ .

7. Конец алгоритма.

Рассмотрим пример работы описанного алгоритма. В качестве объемлющего дерева  $B$  возьмем дерево, приведенное на рис. 1. Пример (см. таблицу) показывает, что в состав дерева  $B$  (рис. 1) входит поддерево,  $r$ -изоморфное изображенному на рис. 2 дереву  $A$ .

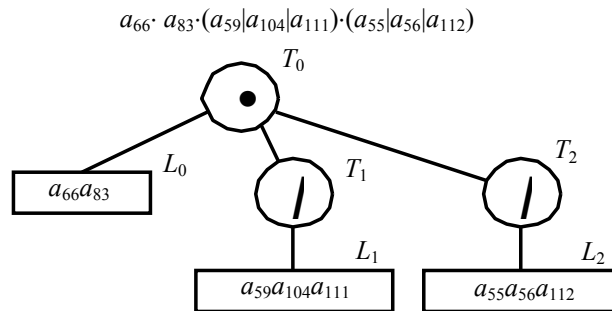


Рис. 2



Пример определения  $r$ -изоморфизма

№ итерации	Пояснение к действию	Значения полей элементов дерева $A$																	
		$L_0$			$L_1$			$L_2$			$T_0$			$T_1$			$T_2$		
		С	НС	ТС	НС	ТС	НС	ТС	НС	ТС	НС	ТС	НС	ТС	НС	ТС	НС	ТС	НС
1	-	00	-1	00	-1	00	-1	00	-1	11	-1	11	-1	11	-1	11	-1	11	-1
2	$N_L(A) = 3, N_L(B) = 5, n_L(A) \leq n_L(B)$	00	-1	00	-1	00	-1	00	-1	11	-1	11	-1	11	-1	11	-1	11	-1
3.1	$j = 0, B.L_0.MB = a_{99}$	00	-1	00	-1	00	-1	00	-1	11	-1	11	-1	11	-1	11	-1	11	-1
3.2	$j = 1, B.L_1.MB = a_{94}a_{95}a_{128}$	00	1	00	-1	00	-1	00	1	11	-1	11	-1	11	-1	11	-1	11	-1
3.3	$j = 2, B.L_2.MB = a_{65}a_{66}a_{82}a_{83}, i = 0, A : L_0 \rightarrow T_0, B : L_2 \rightarrow T_2$	10	2	00	-1	00	-1	00	1	10	2	11	-1	11	-1	11	-1	11	-1
3.4	$j = 3, B.L_3.MB = a_{55}a_{56}a_{112}, i = 2, A : L_2 \rightarrow T_2, B : L_3 \rightarrow T_3$	10	2	00	-1	11	3	10	2	11	3	10	2	11	-1	11	3	10	2
3.5	$j = 4, B.L_4.MB = a_{59}a_{104}a_{111}, i = 1, A : L_1 \rightarrow T_1, B : L_4 \rightarrow T_4$	10	2	11	4	11	3	10	2	11	4	11	3	10	2	11	4	11	3
4	$\lambda = 1 \& 1 = 1$	10	2	11	4	11	3	10	2	11	4	11	3	10	2	11	4	11	3
5.1	$i = 2, A : T_2 \rightarrow T_0, AT_2[\sim]BT_3$ $B : T_3 \rightarrow T_2, \gamma = 0, T_0.TC = f(11, 10) = 10$	10	2	11	4	11	3	10	2	11	4	11	3	10	2	11	4	11	3
5.2	$i = 1, A : T_1 \rightarrow y_0, AT_1[\sim]BT_4$ $B : T_4 \rightarrow T_2, \gamma = 0, T_0.TC = f(11, 10) = 10$	10	2	11	4	11	3	10	2	11	4	11	3	10	2	11	4	11	3
6	$\varphi = 1 \& 1 = 1$	10	2	11	4	11	3	10	2	11	4	11	3	10	2	11	4	11	3

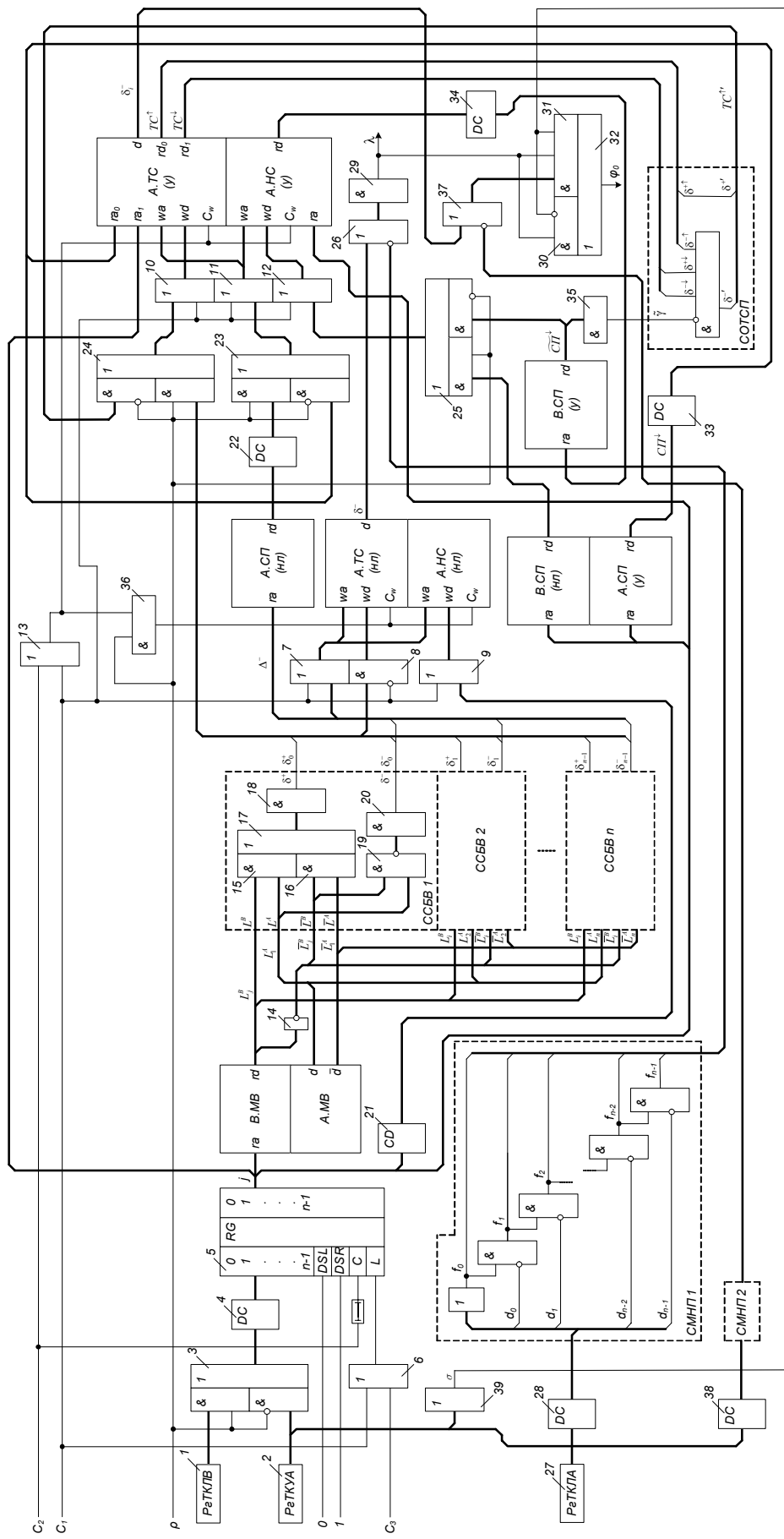


Рис. 3

На основании приведенного выше алгоритма синтезировано специализированное устройство, функциональная схема которого представлена на рис. 3. Регистры 1, 2 и 27 предназначены для хранения значений  $n_L(B)$ ,  $n_T(A)$  и  $n_L(A)$  соответственно, при этом значения полей обрабатываемых деревьев хранятся в элементах однородной среды электронной модели дерева (ОСЭМД) [12] В.МВ, А.МВ, А.СП(нл), А.ТС(нл), А.НС(нл), В.СП(нл), А.СП(у), В.СП(у), А.ТС(у) и А.НС(у), схема ячейки которой приведена на рис. 4.

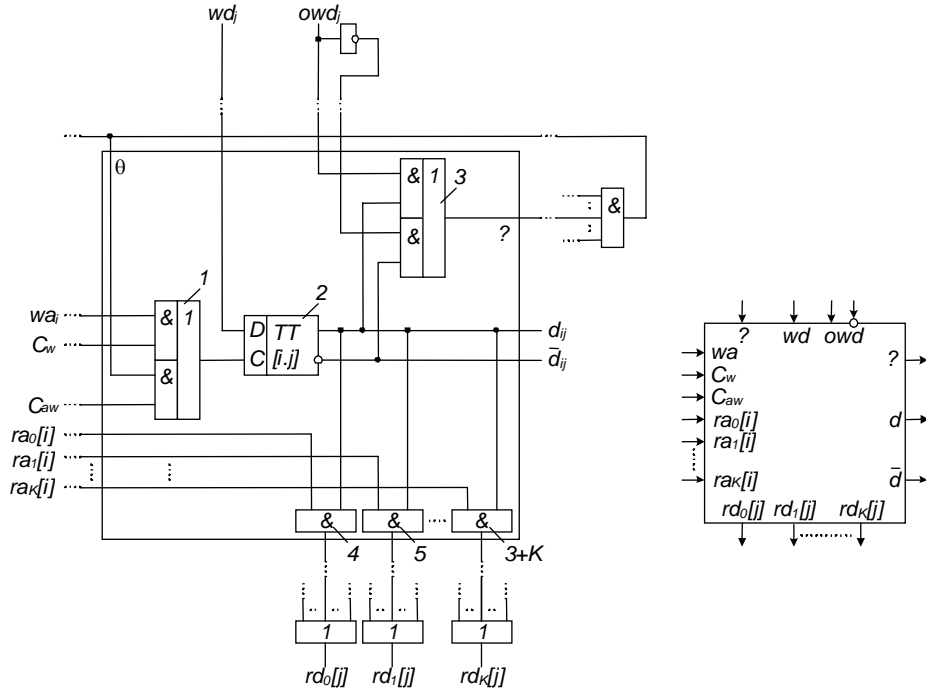


Рис. 4

Коммутаторы 3, 23—25 управляются сигналом  $\rho$  и предназначены для коммутации значений на разных этапах работы алгоритма. Дешифраторы 4, 22, 28, 33, 34, 38 используются для преобразования номеров позиций элементов дерева в табличном представлении из двоичного кода в унитарный, используемый на входах  $ra$  (сокр. от „read address“ — адрес чтения) элементов ОСЭМД. Шифратор 21 используется для обратного преобразования. Сдвиговый регистр 5 используется для поочередного выбора элементов обрабатываемых деревьев. Блоки элементов ИЛИ 7, запрета 8, ИЛИ 9—12 используются для формирования начальных значений полей на этапе инициализации. Элементы ИЛИ 6, ИЛИ 13 и И 36 используются для коммутации синхросигналов. Блок элементов запрета 26 в совокупности с элементом И 29 и схемой маскировки неиспользуемых позиций (СМНП) 1 используется для формирования значения признака  $\lambda$ . Схема СМНП 2 в совокупности с блоком элементов запрета 37, элементом ИЛИ 39, запрета 30, блоком элементов И 31 и элементом ИЛИ 32 используется для формирования результирующего признака  $r$ -изоморфизма  $\phi$ , при этом на выходе элемента ИЛИ 39 формируется значение признака  $\sigma$  отсутствия узлов у дерева  $A$ . Элемент И 35, на выходе которого формируется значение признака  $\tilde{\gamma}$ , в совокупности со схемой определения типа соответствия предка (СОТСП) отвечает за формирование обновленного значения поля типа соответствия узла-предка (фактически схема СОТСП реализует рассмотренную выше функцию  $f$ ). Блок элементов НЕ 14 в совокупности со схемами сравнения битовых векторов (ССБВ) 1— $n$ , образованными блоками элементов И 15 и 16, ИЛИ 17, И-НЕ 19 и элементами И 18 и 20, используется для формирования признаков  $\delta_i^+$  и  $\delta_i^-$  полного и частичного соответствия наборов листьев. Элемент задержки предназначен для задержки синхросигнала сдвига

содержимого регистра 5 на время, достаточное для формирования и записи обновленных значений полей на предыдущей итерации алгоритма.

По сравнению с программной реализацией предлагаемого алгоритма выигрыш в скорости достигается за счет параллельной инициализации значений полей ТС и НС узлов и наборов листьев дерева  $A$ ; параллельного сравнения выбранного  $j$ -го набора листьев дерева  $B$  со всеми наборами листьев дерева  $A$ ; параллельного сравнения компонентов битовых векторов, соответствующих наборам листьев; параллельного чтения и записи значений различных полей ( $A$ .ТС(нл),  $A$ .НС(нл),  $A$ .НС(у) и т.д.) из/в разные элементы ОСЭМД.

При программной реализации [3, 5, 6] имеют место дополнительные затраты на организацию стека возвратов при рекуррентном сравнении деревьев, сканирование наборов листьев при этом фактически является условием завершения рекурсии: обработка элементов наборов листьев производится последовательно. Просмотр узлов деревьев как при программной, так и при аппаратной реализации осуществляется последовательно.

Последовательная программная реализация предложенного в статье алгоритма проверки  $r$ -изоморфизма предполагает попарное сравнение наборов листьев деревьев  $A$  и  $B$  (каждое сравнение наборов листьев требует  $O(L_{\max})$  действий) с последующим просмотром узлов дерева  $A$  снизу вверх. Ее асимптотическая временная сложность составляет

$$O(N_{L_{\max}}^2 L_{\max} + N_T) \approx O(L_{\max}^3),$$

где  $N_{L_{\max}}$  — максимально возможное число наборов листьев в дереве,  $L_{\max}$  — размерность битовых векторов МВ наборов листьев (фактически число вершин в алгоритме управления). Предложенное аппаратное решение обладает асимптотической временной сложностью

$$O(N_{L_{\max}} + N_{T_{\max}}) \approx O(L_{\max}),$$

где  $N_{T_{\max}}$  — максимально возможное число узлов в дереве.

#### СПИСОК ЛИТЕРАТУРЫ

1. Организация и синтез микропрограммных мультимикроконтроллеров / И. В. Зотов, В. А. Колосков, В. С. Титов и др. Курск: КурскГТУ, 1999. 368 с.
2. Зотов И. В., Колосков В. А., Титов В. С. Выбор оптимальных разбиений алгоритмов при проектировании микроконтроллерных сетей // Автоматика и вычислительная техника. 1997. № 5. С. 5162.
3. Поиск базового сечения в задаче разбиения параллельных алгоритмов / Э. И. Ватутин, И. В. Зотов. Курск: КурскГТУ. 2003. 30 с.
4. Ватутин Э. И., Зотов И. В., Титов В. С. Построение множества сечений в задаче оптимального разбиения параллельных управляющих алгоритмов // Изв. ТулГУ. Вычислительная техника. Информационные технологии. Системы управления. Тула: ТулГУ, 2003. Т. 1, вып. 2. С. 70—77.
5. Ватутин Э. И., Зотов И. В. Метод формирования субоптимальных разбиений параллельных управляющих алгоритмов // Параллельные вычисления и задачи управления (РАСО'04). М.: ИПУ РАН, 2004. С. 884—917.
6. Ватутин Э. И., Зотов И. В. Параллельно-последовательный метод формирования субоптимальных разбиений параллельных управляющих алгоритмов. Свид-во об офиц. регистрации программы для ЭВМ № 2005613091 от 28.11.05.
7. Ватутин Э. И., Зотов И. В. Программная система для построения разбиений параллельных управляющих алгоритмов // Идентификация систем и задачи управления (SICPRO'06). М.: ИПУ РАН, 2006. С. 2239—2250.
8. Ватутин Э. И., Зотов И. В. Визуальная среда синтеза разбиений параллельных алгоритмов логического управления. Свид-во об офиц. регистрации программы для ЭВМ № 2007613222 от 30.07.07.
9. Ватутин Э. И., Волобуев С. В., Зотов И. В. Комплексная сравнительная оценка методов выбора разбиений при проектировании логических мультимикроконтроллеров // Идентификация систем и задачи управления (SICPRO'08). М.: ИПУ РАН, 2008. С. 1917—1940.

10. Зыков А. А. Основы теории графов. М.: Наука, 1987. 381 с.
11. Хопкрофт Дж., Мотвани Р., Ульман Дж. Введение в теорию автоматов, языков и вычислений: Пер. с англ. М.: Вильямс, 2002. 528 с.
12. Ватутин Э. И. Однородная среда электронной модели дерева для аппаратно-ориентированной обработки R-выражений // Оптико-электронные приборы и устройства в системах распознавания образов, обработки изображений и символической информации (Распознавание — 2008). Ч. 1. Курск: КурскГТУ, 2008. С. 90—92.

**Сведения об авторах**

- Эдуард Игоревич Ватутин** — аспирант; Курский государственный технический университет, кафедра вычислительной техники; E-mail: evatutin@rambler.ru
- Игорь Валерьевич Зотов** — д-р техн. наук, профессор; Курский государственный технический университет, кафедра вычислительной техники; E-mail: zotovigor@yandex.ru
- Виталий Семенович Титов** — д-р техн. наук, профессор; Курский государственный технический университет, кафедра вычислительной техники; зав. кафедрой; E-mail: titov@vt.kstu.kursk.ru

Рекомендована кафедрой  
вычислительной техники

Поступила в редакцию  
12.09.08 г.

УДК 681.3

С. Ю. МИРОШНИЧЕНКО, В. С. ТИТОВ

**ПАРАЛЛЕЛЬНО-КОНВЕЙЕРНОЕ УСТРОЙСТВО  
ДЛЯ ВЕКТОРИЗАЦИИ АЭРОКОСМИЧЕСКИХ ИЗОБРАЖЕНИЙ  
ЗЕМНОЙ ПОВЕРХНОСТИ**

Рассмотрены структурная схема и алгоритмы функционирования устройства автоматического построения тематического слоя зданий и строений для электронных карт геоинформационных систем на основе аэрокосмических изображений. Наличие специализированных параллельных графических модулей и конвейерный режим функционирования устройства позволяют повысить степень загрузки вычислительных блоков и увеличить его производительность.

**Ключевые слова:** геоинформационные системы, аэрокосмические изображения, автоматическая векторизация, слой зданий, выделение контуров, параллельно-конвейерное устройство.

В современном мире геоинформационные системы (ГИС) используются при решении широкого круга задач в области кадастрового учета и городского планирования, управления транспортом, землепользования и сельскохозяйственной деятельности, подведения коммуникаций и их обслуживания [1, 2]. Для эффективного решения поставленных задач пользователи ГИС должны получать актуальную информацию, полностью отражающую реальное состояние определенного региона на текущий момент времени. Следовательно, необходимо периодически производить актуализацию электронных карт ГИС с учетом изменений, произошедших со времени последнего обновления.

Наиболее актуальную информацию, необходимую для создания и обновления электронных карт ГИС, предоставляют цифровые аэрокосмические изображения. Перенос информации, содержащейся в аэрокосмическом изображении, в электронную карту выполняется путем формирования набора тематических слоев, каждый из которых содержит векторные представления принадлежащих данному слою объектов.

Процесс построения электронной карты на основе аэрокосмического изображения называется *векторизацией* и заключается в последовательном создании оператором векторных представлений объектов каждого из необходимых тематических слоев. Векторизация аэрокосмических изображений является трудоемким процессом, поскольку автоматизирована слабо и практически полностью выполняется оператором. Методы и программные продукты, применяемые для автоматической векторизации сканированных картографических материалов, практически неприменимы для аэрокосмических изображений, поскольку не учитывают их информационной избыточности и сложности внутренней структуры содержащихся в них объектов [3].

В настоящей статье предложен метод автоматического формирования слоя городской застройки по аэрокосмическому изображению, основанный на контурном подходе к решению задачи локализации объектов и предназначенный для автоматического построения „эскиза“ данного тематического слоя, что позволяет избавить оператора от рутинных операций по

„отрисовке“ объектов. Данный метод позволяет существенно сократить время, необходимое для создания или обновления электронной карты города, однако обладает высокой вычислительной сложностью (время обработки изображения  $10\,000 \times 10\,000$  точек параллельной реализацией метода на четырехъядерных процессорах Intel Core 2 Quad или AMD Phenom достигает нескольких часов).

Для снижения временных затрат на автоматическое построение тематического слоя городской застройки разработано параллельно-конвейерное устройство, содержащее процессорный модуль (ПМ), состоящий из управляющего процессора (УП) и блока векторизации (БВ), ОЗУ, контроллер ОЗУ (КЗУ), первый графический модуль (ГМ1), состоящий из первого графического процессора (ГП1) и первой буферной памяти (БП1), модуль (ГМ2), состоящий из ГП2 и БП2, третий графический модуль (ГМ3), состоящий из ГП3 и БП3, контроллер ввода—вывода изображения (КВВ), внешнее ЗУ (ВЗУ), информационный интерфейс (ИИ). Структурная схема устройства приведена на рис. 1.

Устройство функционирует в двух режимах: последовательном и конвейерном.

При запуске устройства обработка первого изображения выполняется в последовательном режиме (граф-схема алгоритма последовательного режима приведена на рис. 2).

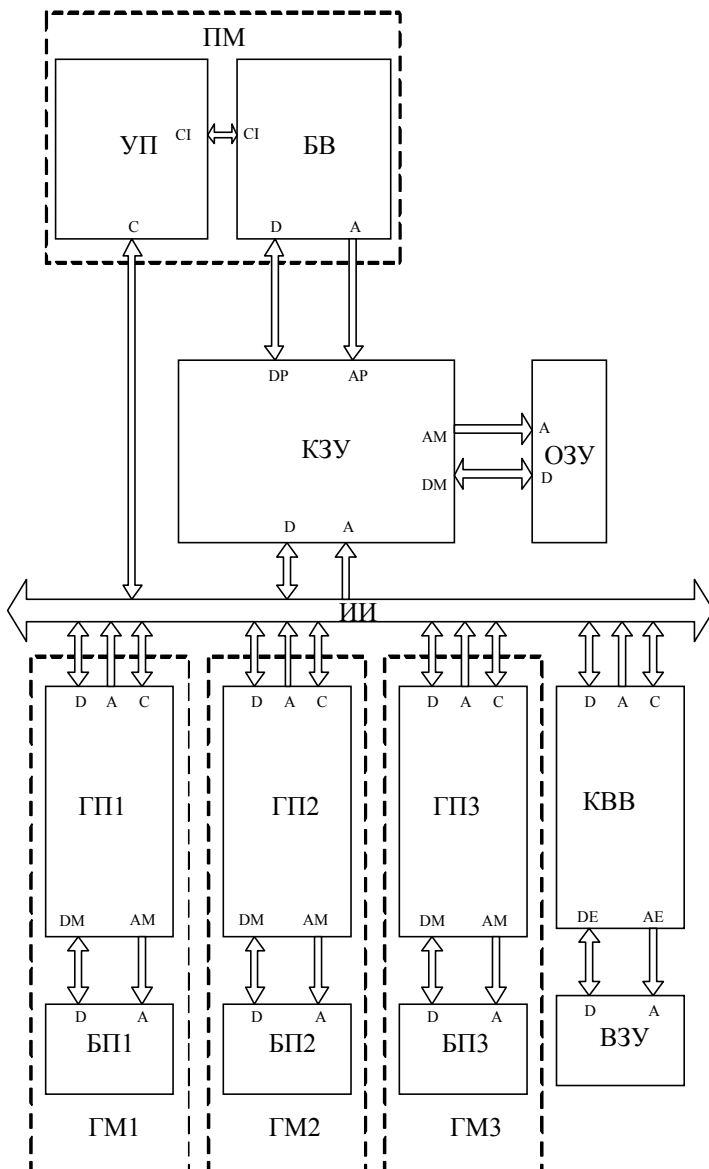


Рис. 1

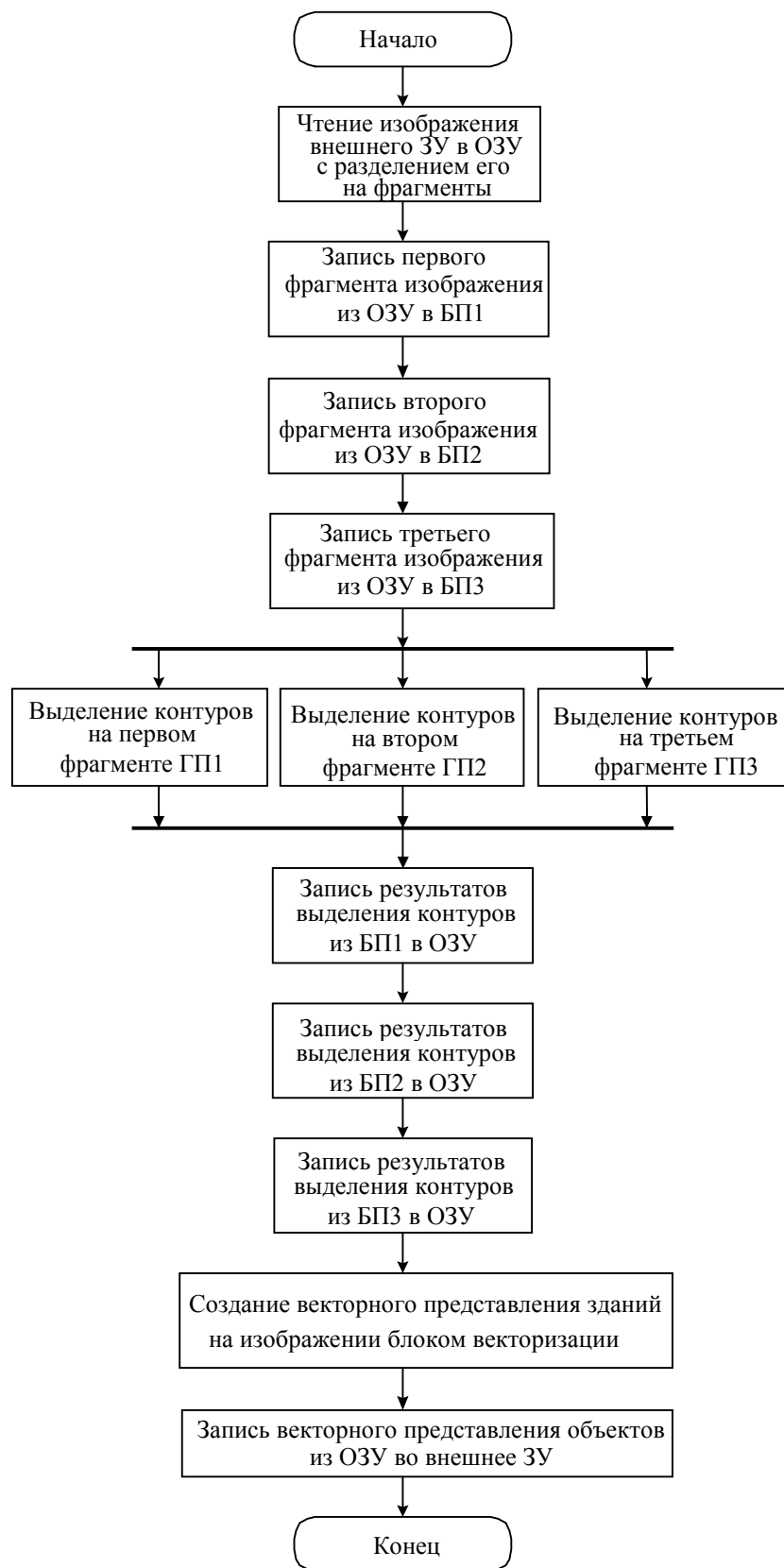


Рис. 2

Для этого по команде управляющего процессора контроллер ввода—вывода выполняет построчное считывание аэрокосмического изображения из внешнего ЗУ и его передачу через информационный интерфейс в ОЗУ под управлением контроллера ОЗУ. По окончании ввода изображения управляющий процессор выдает команду подготовки изображения к обработке, при получении которой контроллер ОЗУ по оси ординат разбивает изображение размером  $K \times M$  на три фрагмента (их количество может быть произвольным и определяется числом графических модулей). Размер каждого фрагмента увеличен на  $2N$  элементов в каждом измерении. В центре созданного фрагмента размещается часть исходного изображения, за счет чего по краям образуется „бордюр“ в  $N$  элементов (рис. 3), который позволяет обрабатывать фрагмент изображения дифференциальными масочными операторами, не изменяя алгоритма свертки на краях фрагмента. Ширина „бордюра“  $N$  вычисляется исходя из точности аппроксимации при построении масочных фильтров на основании первой производной функции Гаусса с максимальным значением масштаба  $\sigma_{\max}$  (как правило,  $N = 4\sigma_{\max}$ ).

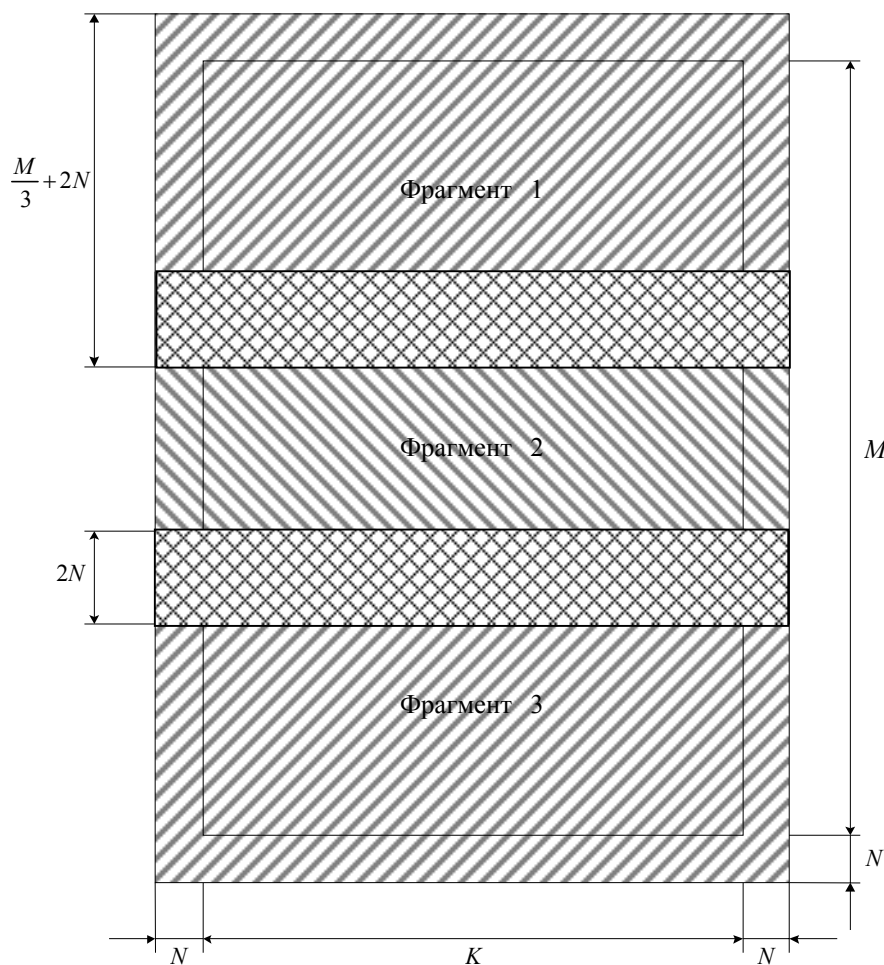


Рис. 3

По команде передачи данных на обработку, выдаваемой управляющим процессором, первый фрагмент из ОЗУ под управлением контроллера ОЗУ через информационный интерфейс передается в буферную память первого графического модуля. Таким же образом осуществляется передача второго и третьего фрагментов в буферную память второго и третьего графических модулей соответственно. По окончании передачи данных управляющий процессор выдает команду, которая инициирует обработку фрагментов изображения графическими процессорами.



Алгоритм обработки следующий.

1. Дифференцирование фрагментов исходного изображения. Производится свертка фрагмента с масками, представляющими дискретные аппроксимации первых частных производных функции Гаусса. Масштаб функции Гаусса  $\sigma$  вычисляется в каждой точке фрагмента с учетом функции яркости [4, 5]. В результате дифференцирования формируется карта модулей, описывающая контуры объектов на изображении.

2. Утончение контурных линий объектов на карте модулей градиента с помощью критерия максимума модуля градиента в направлении, перпендикулярном касательной к контурной линии в данной точке [6]. Утончение производится для приведения контурных линий к односвязному виду, обеспечивающему возможность их последующего векторного кодирования.

3. Бинаризация карты модулей градиента для отделения малоинформативных деталей изображения, которая выполняется с помощью порогового оператора с гистерезисом [6]. Пороговые значения рассчитываются автоматически с помощью сигма-метода [7] и метода Отсу [8].

По завершении обработки управляющий процессор инициирует передачу бинарных контурных представлений фрагментов исходного изображения из буферной памяти каждого из трех графических модулей через информационный интерфейс в ОЗУ с последующим их объединением в результирующее бинарное изображение.

Блок векторизации по команде управляющего процессора выполняет построение векторного представления объектов тематического слоя городской застройки. Построение включает следующие основные этапы.

1. Формирование векторных представлений контурных линий объектов с помощью алгоритма [9].

2. Восстановление разрывов в контурах объектов.

3. Идентификация контуров зданий и строений.

4. Геометрическая коррекция контуров зданий и приведение их к основанию (фундаменту).

По окончании векторизации сформированное множество векторных представлений зданий и строений по команде управляющего процессора передается из ОЗУ через информационный интерфейс во внешнее ЗУ.

Обработку второго и последующих изображений устройство выполняет в конвейерном режиме, что позволяет повысить загрузку вычислительных блоков (алгоритм конвейерного режима работы представлен на рис. 4). После копирования фрагментов текущего исходного аэрокосмического изображения из ОЗУ в буферную память каждого из трех графических модулей помимо построения бинарных контурных представлений фрагментов графическими процессорами также производится чтение следующего изображения из внешнего ЗУ в ОЗУ и создание блоком векторизации векторного представления зданий и строений по бинарному представлению предыдущего изображения, хранимому в ОЗУ. По окончании всех указанных операций бинарные представления фрагментов текущего изображения последовательно копируются из буферной памяти каждого графического модуля в ОЗУ с помощью контроллера ОЗУ, после чего векторное представление зданий на предыдущем изображении записывается из ОЗУ во внешнее ЗУ.

Процессорный модуль может быть реализован на двухъядерных процессорах Intel Core 2 Duo или AMD Athlon x2, причем первое ядро процессора выполняет функции управляющего процессора, второе — функции блока векторизации. Для реализации процессорного модуля также могут быть использованы трехъядерные процессоры AMD Phenom x3 и четырехъядерные процессоры Intel Core 2 Quad или AMD Phenom, где первое ядро выполняет функции управляющего процессора, а другие два (или три) ядра — функции блока векторизации.

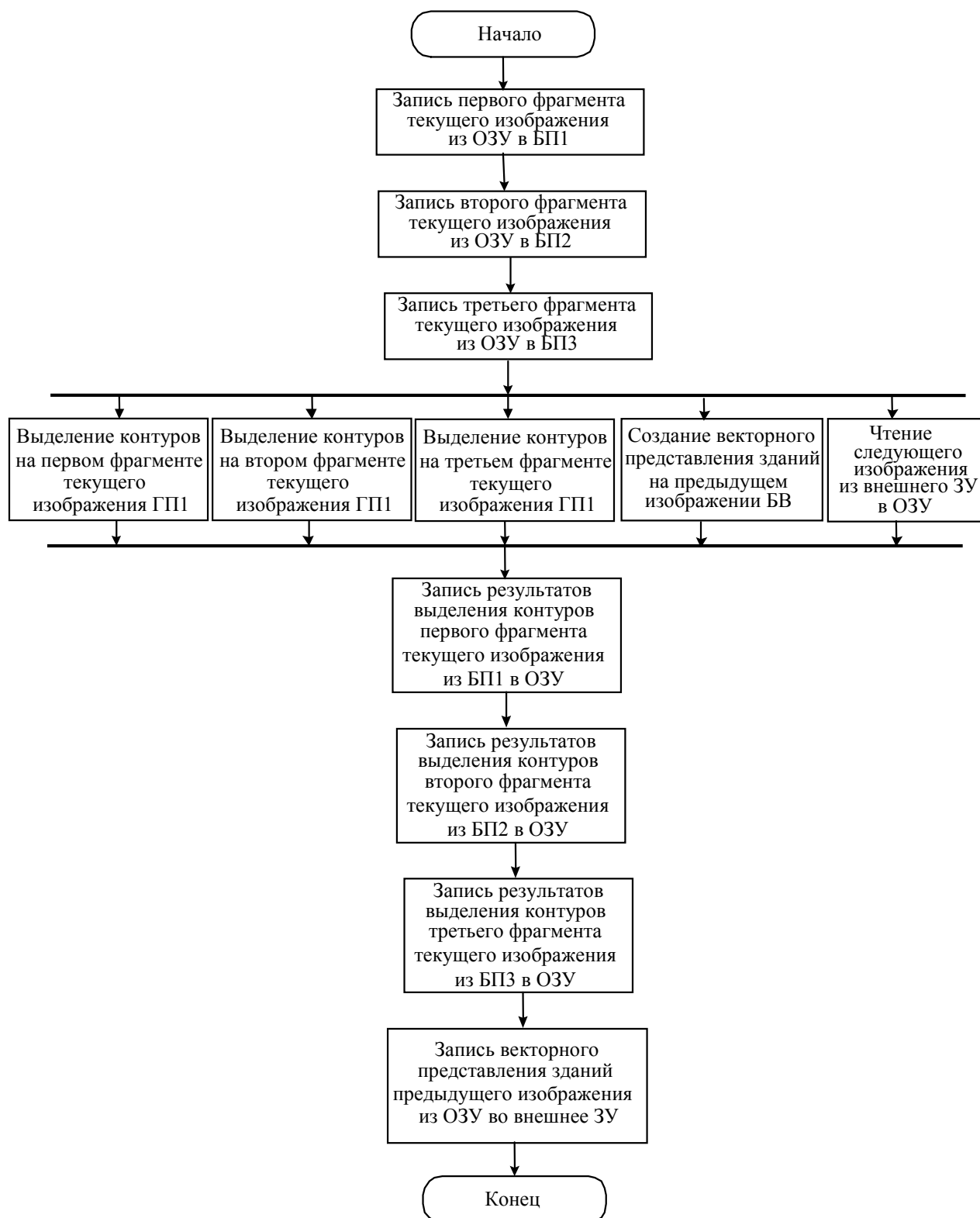


Рис. 4

Графические модули реализуются на трех графических ускорителях nVidia GeForce 8800/9600/9800/GT260/GT280, объединенных с использованием технологии nVidia SLI, с программной реализацией алгоритма выделения контуров объектов на изображении под платформу nVidia CUDA.

Таким образом, предложенное устройство позволяет снизить временные затраты на формирование векторного описания элементов городской застройки по аэрокосмическим изображениям земной поверхности за счет построения бинарного контурного представления изображения средствами специализированных графических модулей, а также за счет конвейерного режима работы, обеспечивающего увеличение производительности аппаратуры.

#### СПИСОК ЛИТЕРАТУРЫ

1. Андрианов Д. Е. Геоинформационные системы: исследование, анализ и разработка. М.: ГНЦ РФ ВНИИгеосистем, 2004. 184 с.
2. Мартыненко А. И. Информатика и электронная Земля: фундаментальные научные проблемы // Системы и средства информатики. М.: ИПИ РАН, 2001. Вып. 11. С. 94—112.
3. Easy Trace: Интеллектуальное программное обеспечение для векторизации картографических материалов [Электронный ресурс]: <[http://easytrace.com/site/russian/easytracepro/new\\_in\\_ET.html](http://easytrace.com/site/russian/easytracepro/new_in_ET.html)>.
4. Elder J., Zucker S. Local Scale Control for Edge Detection and Blur Estimation // IEEE Trans. Pattern Analysis and Machine Intelligence. 1998. Vol. 20, N 7. P. 699—716.
5. Miroschnichenko S. Yu. Buildings and Structures Preliminary Identification on Aerospace Images Using Edges Detection with Local Scale Estimation // Proc. 9th Int. Conf. „Pattern Recognition and Image Analysis: New Information Technologies“. Nizhni Novgorod, 2008. Vol. 2. P. 51—52.
6. Canny J. F. A Computational Approach to Edge Detection // IEEE Trans. Pattern Recognition and Machine Intelligence. 1986. Vol. 8, N 6. P. 679—698.
7. Miroschnichenko S. Yu., Degtyarev S. V., Rukavitsin Yu. A. Adaptive Choice of Global Threshold for Binarization of Gradient Images // Pattern Recognition and Image Analysis. 2005. Vol. 15, N 1. P. 249—251.
8. Otsu N. A Threshold Selection Method from Gray-Level Histograms // IEEE Trans. Syst. Man Cybern. 1979. Vol. 9, N 1. P. 62—66.
9. Мирошниченко С. Ю., Мишустин В. Н., Дегтярев С. В. Распознающий аппаратно-программный диагностирующий комплекс // Изв. вузов. Приборостроение. 2005. Т. 48, № 2. С. 22—27.

#### Сведения об авторах

- Сергей Юрьевич Мирошниченко** — аспирант; Курский государственный технический университет, кафедра вычислительной техники; E-mail: oldguy7@rambler.ru
- Виталий Семенович Титов** — д-р техн. наук, профессор; Курский государственный технический университет, кафедра вычислительной техники; зав. кафедрой; E-mail: titov@vt.kstu.kursk.ru

Рекомендована кафедрой  
вычислительной техники

Поступила в редакцию  
12.09.08 г.

Е. А. ТИТЕНКО

## МЕТОД РЕКОНФИГУРАЦИИ ОПЕРАЦИОННОЙ ЧАСТИ МУЛЬТИПРОЦЕССОРА СТРУКТУРНОГО РАСПОЗНАВАНИЯ ОБРАЗОВ

Представлен метод распознавания структурных дескрипторов для ветвящихся процессов, показана организация соединений между производственными устройствами в мультипроцессоре.

*Ключевые слова:* структурное распознавание, мультипроцессор, ветвящиеся вычислительные процессы, продукция.

Современные теоретические, программные и технические средства обработки разнородной информации все в большей степени ориентируются на организацию параллельных распределенных вычислений [см. лит.]. Такие инструментальные средства должны содержать блоки (модули) для упреждающего распознавания структурных особенностей взаимодействия параллельных процессов и последующей реконфигурации под граф решаемой задачи. Основная проблема заключается в том, что проектирование современных многопроцессорных устройств и машин не учитывает специфики взаимодействия вычислительных процессов решаемой задачи, что приводит к дополнительным затратам времени на обменные процессы между параллельно работающими устройствами или блоками при организации параллельных вычислений. Таким образом, семантический разрыв между аппаратными средствами проектировщика и алгоритмическими средствами пользователя порождает временную избыточность. Особенно остро эта проблема проявляется при организации параллельных вычислений с ветвящимися процессами, возникающими в задачах структурного распознавания образов в различных предметных областях: в динамических экспертных системах, в системах обработки знаний.

В настоящей статье представлен перспективный вариант реализации гибкой динамически реконфигурируемой коммутационной структуры мультипроцессора, являющейся основой его операционной части.

Метод реконфигурации соединений производственных устройств основан на упреждающем структурном распознавании символьных дескрипторов конфликтных ситуаций и динамической реконфигурации структуры операционной части мультипроцессора в однородную.

Был разработан инвариантный по возможностям метод прогнозирования соединений параллельно работающих устройств в соответствии с графом решаемой задачи на основе производственной парадигмы организации вычислений. Целесообразность выбора производственной парадигмы для организации параллельных вычислений определяется следующими факторами:

- модульность и однородность состава системы производств, что позволяет динамически реконфигурировать соответствующие аппаратные средства;
- гибкость реализации различных схем управления;
- потенциальный параллелизм и асинхронность производственных вычислений.

Целесообразно использовать вычислительную производственную систему. Одной из основных причин ограничения области применения многопроцессорных устройств является возникновение конфликтных ситуаций, существенно ограничивающих возможности таких систем в задачах распознавания реального уровня сложности. Вследствие конфликтной ситуации структура образца  $j$ -й продукции в обрабатываемом слове нарушается, что препятствует срабатыванию  $i$ -й продукции при пересечении образцов  $i, j$  ( $i \neq j$ ) производственной системы.

Попарное пересечение образцов  $i, j$  ( $i \neq j$ ) производственной системы является логическим условием отношения зависимости между производствами, вследствие чего возникают недетер-

минированность в процессах принятия решения и непродуктивные затраты времени на поиск и распознавание эталонных описаний в массивах разнородных данных большой размерности. Конфликтные ситуации сужают потенциальные возможности программных и аппаратных средств, что недопустимо в задачах распознавания динамично возникающих в графе решаемой задачи ситуаций конфликтов между параллельными процессами за общий ресурс.

В связи с этим важнейшей задачей проектирования мультипроцессорной организации системы является метод реконфигурации, основанный на упреждающем распознавании и нейтрализации конфликтных ситуаций. Основное преимущество предлагаемого метода распознавания структурных дескрипторов конфликтов заключается в извлечении дополнительной информации, основанной на рекурсивном обнаружении парных пересечений образцов.

Исследование вариантов пересечения слов и синтез на их основе конфликтных слов определили структуру аналитического модуля распознавания, входящего в состав мультипроцессора (рис. 1, где 1 — модуль проверки условия заикливания, 2 — модуль поиска пересечений слов, 3 — модуль построения дополнений, 4 — модуль объединения слов,  $n$  — количество продукций,  $O_1, \dots, O_n$  — входной список образцов,  $m$  — количество дескрипторов конфликтных ситуаций,  $v$  — количество промежуточных дескрипторов конфликтных ситуаций,  $K_1, \dots, K_m$  — выходной список дескрипторов конфликтных ситуаций,  $K_1, \dots, K_v$  — промежуточный список дескрипторов конфликтных ситуаций).

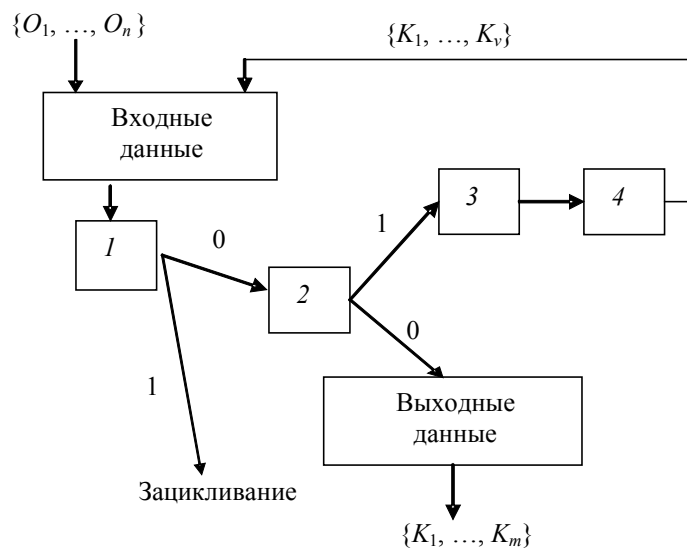


Рис. 1

На основе полученных дескрипторов конфликтных ситуаций  $K_1, \dots, K_m$  вычисляются настроечные значения соединений продукционных устройств в операционной части.

Рассмотрим общие принципы построения мультипроцессора с реконфигурируемыми соединениями. Под мультипроцессором понимается символьное устройство для аппаратной поддержки ветвящихся продукционных вычислений. Каждое продукционное процессорное устройство (ППУ) аппаратно реализует базовые операции поиска и преобразования строковых данных. Принципы однородности и модульной наращиваемости архитектуры символьного мультипроцессора предписывают ставить в соответствие каждой продукции вычислительной системы отдельное ППУ с собственной внутренней памятью. Тогда массовые параллельные вычисления в мультипроцессоре определяются сбалансированной подсистемой коммутации ППУ в составе операционной части в соответствии с графом решаемой задачи.

Исходя из общесистемных принципов организации параллельных вычислений в задачах с ветвящимися конструктивными процессами операционная часть символьного мультипроцессора будет содержать два настраиваемых яруса ППУ. Эти ярусы являются независимыми друг от друга, поэтому возможно их параллельное функционирование. Первый ярус ориентирован

на работу ППУ в соответствии с установленными соединениями и обеспечивает текущие вычисления по графу решаемой задачи. Второй ярус операционной части мультимикропроцессора осуществляет настройку новых соединений ППУ для работы на следующем шаге по графу решаемой задачи.

Попеременное переключение двух параллельно работающих ярусов ППУ является отличительной чертой предлагаемого технического решения. Это обстоятельство позволяет осуществлять расслоение общего графа решаемой задачи на самостоятельные подграфы, в которых параллельные вычисления комбинируются с последовательными. Независимость подграфов вычислений создает предпосылки для параллельного функционирования двух ярусов ППУ с децентрализованным управлением каждым ярусом. В связи с этим архитектура символьного мультимикропроцессора получается гибридной, так как в ней совмещаются детерминированные принципы синхронного управления конвейерных соединений ППУ и недетерминированные принципы асинхронного управления параллельных соединений ППУ.

На рис. 2 приведена структура яруса операционной части мультимикропроцессора для обеспечения ветвящихся вычислительных процессов.

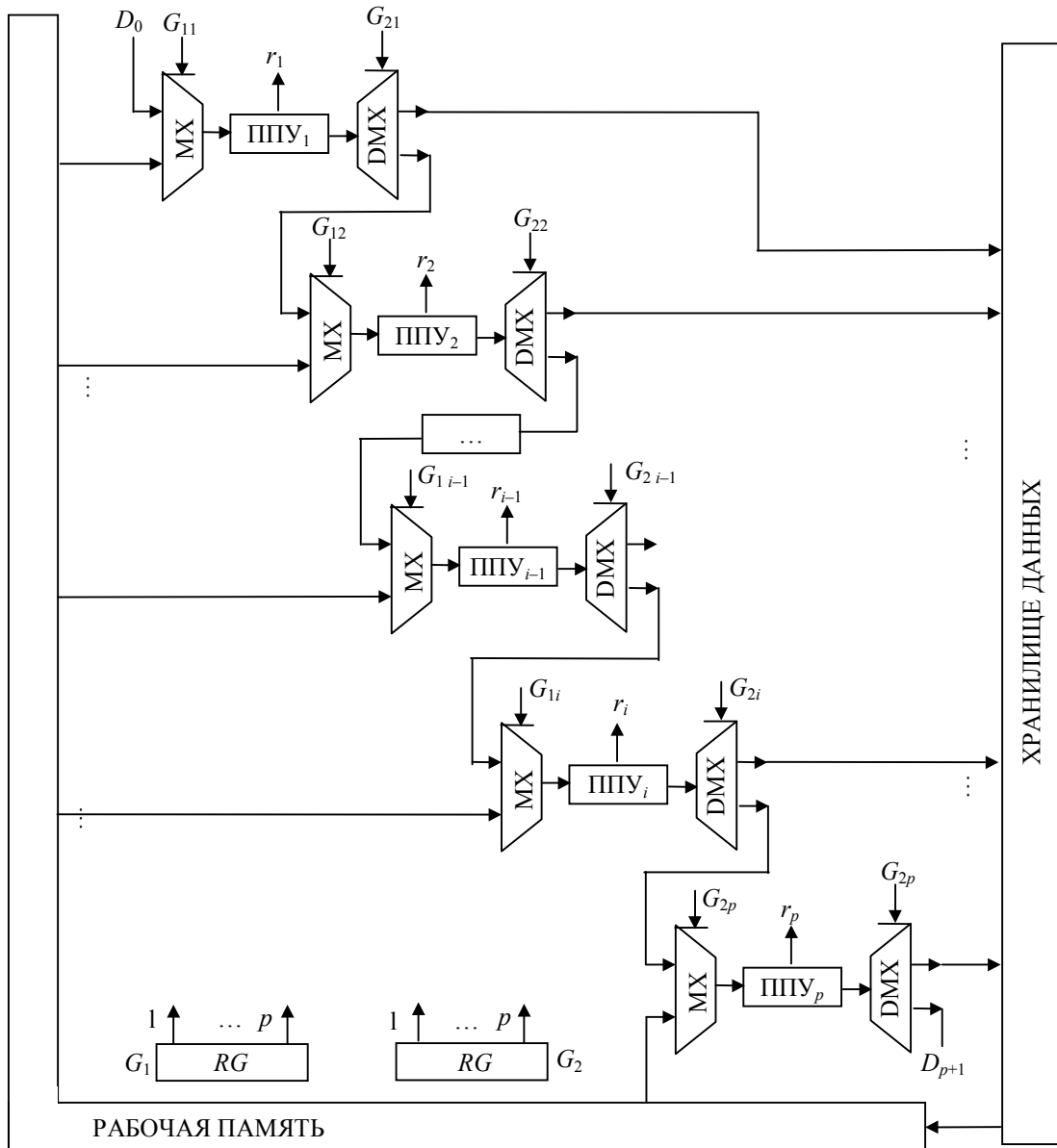


Рис. 2

Входные регистры  $G_1$  и  $G_2$  хранят настроечные векторы соединений ППУ между собой. Настроечные векторы вычисляются аналитическим модулем распознавания структурных дескрипторов конфликтов, сам модуль не входит в состав операционной части. Основу однородной структуры операционной части составляет вычислительно-коммутационный блок, содержащий собственно ППУ, входной мультиплексор (МХ) с двумя входами, выходной демultipлексор (ДМХ) с двумя выходами ( $r$  — индикатор номера производственного устройства). Введение двух коммутирующих узлов в вычислительно-коммутационный блок обеспечивает гибкую настройку соединений конвейерных или параллельных видов с объединением необходимого на текущем уровне количества ППУ.

Таким образом, разработана структура операционной части мультипроцессора, отличающаяся поддержкой ветвящихся вычислительных процессов и имеющая в своем составе два параллельно работающих яруса производственных процессорных устройств на основе структурного распознавания дескрипторов конфликтов и динамической упреждающей реконфигурации яруса ППУ под следующий шаг вычислений по структуре графа задачи.

#### ЛИТЕРАТУРА

Каляев А. В., Левин И. И. Модульно-наращиваемые многопроцессорные системы со структурно-процедурной организацией вычислений. М.: Изд-во „Янус-К“, 2003. 380 с.

#### *Сведения об авторе*

**Евгений Анатольевич Титенко** — канд. техн. наук, доцент; Курский государственный технический университет, кафедра программного обеспечения вычислительной техники; E-mail: johntit@mail.ru

Рекомендована кафедрой  
программного обеспечения  
вычислительной техники

Поступила в редакцию  
12.09.08 г.

УДК 681.3

Д. Б. БОРЗОВ

### РАЗМЕЩЕНИЕ ПАРАЛЛЕЛЬНЫХ ПОДПРОГРАММ В МУЛЬТИМИКРОКОНТРОЛЛЕРАХ С УЧЕТОМ ОТКАЗОВ

Рассмотрена проблема размещения параллельных подпрограмм в мультимикроконтроллере с учетом возможного отказа его процессорного модуля, сформулированы методика и алгоритм планирования размещения задач.

**Ключевые слова:** размещение, мультимикроконтроллер, отказ, реконфигурация, параллельная подпрограмма, процессор.

В настоящее время все большее распространение получают отказоустойчивые мультимикроконтроллеры [1], при этом повышаются требования к их быстродействию.

В случае отказа одного из процессоров мультимикроконтроллера необходимо быстро восстановить его функционирование путем реконфигурации структуры с отключением неисправного процессора и заменой его резервным, расположенным обычно вне поля обрабатываемых процессоров. В результате существенно изменяются конфигурации связей между процессорами мультимикроконтроллера и возрастает длина маршрутов передачи данных. Длина маршрута может быть уменьшена путем оперативного перераспределения задач. Процедуры планирования размещения являются комбинаторными, имеют большую вычислительную сложность и поэтому могут

привести к существенному увеличению времени восстановления системы и снижению коэффициента ее готовности. Отказываться из-за этого от перераспределения задач перед рестартом восстановленной системы нецелесообразно, так как возросшие коммуникационные задержки могут привести к потере системной производительности, превышающей ожидаемый выигрыш от применения параллельной многопроцессорной обработки комплекса взаимодействующих подпрограмм [2—4]. В связи с этим для уменьшения времени восстановления мультимикроконтроллера следует многократно снизить затраты времени на планирование размещения задач. Для этого необходимо создать метод сокращения вычислительной сложности процедур планирования размещения задач в процессорных модулях мультимикроконтроллера.

Множество обрабатываемых подпрограмм описывается графом взаимодействия задач [5]:  $G = \langle X, E \rangle$ , где  $X$  — множество вершин графа  $G$ , вершины  $x_i \in X$  которого соответствуют подпрограммам, а взвешенные дуги  $e_{ij} \in E$  определяют объемы данных (в байтах), передаваемых между задачами при следующих соотношениях между индексами:  $i = (q - 1)n + k$ . Граф  $G$  может быть описан матрицей обмена информации (МОИ)

$$M = \|m_{ij}\|_{N \times N},$$

где  $N = n^2 = |X|$ ;  $m_{ij}$  — объем передаваемых данных между  $i$ -м и  $j$ -м процессорными модулями [6].

Мультимикроконтроллерную систему будем представлять топологической моделью в виде графа  $H = \langle P_1, V \rangle$ , где  $P_1 = \{P \cup L\}$ , здесь  $P$  — множество процессорных модулей мультимикроконтроллера, организованных в матрицу  $|P|_{n \times n}$ , где  $|P| = N = n^2$  — число процессорных модулей мультимикроконтроллера,  $V$  — множество межмодульных связей, задаваемых матрицей смежности  $\|W\|_{N \times N}$  размером  $n^2 \times n^2$  [7];  $L$  — дополнительно введенные резервные модули мультимикроконтроллерной системы:

$$\left\{ \begin{array}{c} l_1 \\ l_2 \\ \dots \\ l_n \end{array} \right\}. \quad (1)$$

Дополнительные модули (1) образуют столбец, записываемый в крайнем правом столбце матрицы  $P$ .

В отсутствие отказов мультимикроконтроллерной системы размещение пакета подпрограмм (задач), описываемых графом  $G$ , может быть аналитически описано следующим образом:

$$\beta_s = X_s \rightarrow P, \text{ где } s = \overline{1, N!}, k = \overline{1, n}, q = \overline{1, n}.$$

Здесь  $s$  — номер очередной перестановки задач  $\{x_{qk}\}$  по процессорным модулям  $\{P_{qk}\}$ , соответствующий  $s$ -му варианту размещения. Мощность множества  $\psi = \{\beta_s\}$  всевозможных отображений  $\beta_s$  равна числу всевозможных перестановок задач  $\{x_{qk}\}$  в матрице  $X: |\psi| = N!$ . Резервные процессорные модули  $l_1, l_2, \dots, l_l$  в случае полной работоспособности всех процессоров мультимикроконтроллера остаются незадействованными. Для описания множества длин  $d_{ij}$  кратчайших маршрутов передачи данных в пределах мультимикроконтроллера введем матрицу минимальных расстояний (ММР)



$$D = \|d_{ij}\|_{N \times N}, \quad N = n^2 = |P|,$$

которую можно построить по матрице смежности.

Пусть  $\Psi$  — множество всевозможных отображений вида  $\beta_s$ . Тогда задачу размещения можно сформулировать как поиск такого отображения  $\beta^* \in \Psi$  [7], что

$$T_{\beta^*} = \min_{\Psi} \left\{ \max_{\beta \in \Psi} \left\{ T_{\beta}(p_{a,b}, p_{x,y}) \right\} \right\}, \quad (2)$$

где  $T_{\beta}(p_{a,b}, p_{x,y})$  — коммутационная задержка при передаче данных между процессорными модулями  $p_{a,b}$  и  $p_{x,y}$ , соответствующая отображению  $\beta$  и вычисляемая как произведение

$$T_{\beta}(p_{a,b}, p_{x,y}) = d_{ij} m_{ij}. \quad (3)$$

Процесс планирования размещения задач может быть выполнен, например, по методу, описанному в статье [6].

В случае отказа, например, процессора  $p_{r,t}$  ( $r = \overline{1, n}, t = \overline{1, n}$ ) мультимикроконтроллерной системы размещение пакета задач, описываемых графом  $G$ , может быть аналитически описано следующим образом:

$$X_s \rightarrow P_1 = \left\{ \begin{array}{cccccc} x_{s_{1,1}} & x_{s_{1,2}} & \dots & x_{s_{1,k}} & x_{s_{1,k+1}} & \dots & x_{s_{1,n}} \\ x_{s_{2,1}} & x_{s_{2,2}} & \dots & x_{s_{2,k}} & x_{s_{2,k+1}} & \dots & x_{s_{2,n}} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ x_{s_{q,1}} & x_{s_{q,1}} & \dots & x_{s_{q,k}} & x_{s_{q,k+1}} & \dots & x_{s_{q,n}} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ x_{s_{n,1}} & x_{s_{n,2}} & \dots & x_{s_{n,k}} & x_{s_{n,k+1}} & \dots & x_{s_{n,n}} \end{array} \right\} \rightarrow$$

$$\rightarrow \left\{ \begin{array}{cccccc|c} p_{1,1} & p_{1,2} & \dots & p_{1,k} & p_{1,k+1} & \dots & p_{1,n} & l_1 \\ p_{2,1} & p_{2,2} & \dots & p_{2,k} & p_{2,k+1} & \dots & p_{2,n} & l_2 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ p_{q,1} & p_{q,2} & \dots & \boxtimes & p_{q,k} & \dots & p_{q,n-1} & p_{q,n} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ p_{n,1} & p_{n,2} & \dots & p_{n,k} & p_{n,k+1} & \dots & p_{n,n} & l_t \end{array} \right\}, \quad (4)$$

где  $s = \overline{1, N!}, k = \overline{1, n}, q = \overline{1, n}, r = \overline{1, n}, t = \overline{1, n}$ ; символом  $\boxtimes$  обозначен возможный отказ процессора  $p_{r,t}$ .

В данном случае при отказе процессора  $p_{r,t}$  необходимо оперативно заменить его на один из ближайших свободных резервных процессоров  $l_1, l_2, \dots, l_t$ . Возможен вариант замены [1], при котором все подпрограммы  $x_{s_{q,k}}, x_{s_{q,k+1}}, \dots, x_{s_{q,n}}$ , назначенные ранее на процессоры  $p_{q,k}, p_{q,k+1}, \dots, p_{q,n}$ , переназначаются на один из ближайших свободных резервных процессоров, расположенных правее. В этом случае, например, задание  $x_{s_{q,n}}$ , назначенное ранее процессору  $p_{q,n}$ , назначается  $l_q$ ,  $x_{s_{q,n-1}}$  — процессору  $p_{q,n}$  и т.д. В то же время все задания  $x_{s_{q,1}}, x_{s_{q,2}}, \dots, x_{s_{q,k-1}}$  остаются без изменений.

Рассмотрим справедливость приведенного отображения (4) на примере. Пусть имеется гипотетический вариант размещения, заданный моделью в виде матрицы  $|P|_{2 \times 2}$ , где  $|P| = N = n^2 = 2^2 = 4$ :

$$P = \left\{ \begin{array}{cc|l} p_1 & p_2 & l_1 \\ p_3 & p_4 & l_2 \end{array} \right\}. \quad (5)$$

Выражению (5) соответствует ММР:

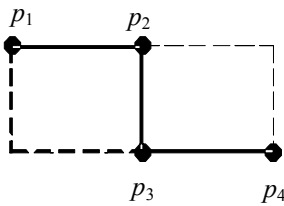
$$D = \begin{vmatrix} d_{1,1} & d_{1,2} & d_{1,3} & d_{1,4} \\ d_{2,1} & d_{2,2} & d_{2,3} & d_{2,4} \\ d_{3,1} & d_{3,2} & d_{3,3} & d_{3,4} \\ d_{4,1} & d_{4,2} & d_{4,3} & d_{4,4} \end{vmatrix}. \quad (6)$$

Пусть также задано множество межмодульных связей  $V$ , соответствующее отображению (4), описанное матрицей МОИ:  $M = \|m_{ij}\|_{4 \times 4}$ , которой ставится в соответствие ММР (6):

$$M = \begin{Bmatrix} 0 & 15 & 0 & 7 \\ 15 & 0 & 10 & 3 \\ 0 & 4 & 0 & 10 \\ 7 & 3 & 4 & 0 \end{Bmatrix} \rightarrow D = \begin{Bmatrix} 0 & 1 & 2 & 1 \\ 1 & 0 & 1 & 2 \\ 2 & 1 & 0 & 1 \\ 1 & 2 & 1 & 0 \end{Bmatrix}. \quad (7)$$

В выражении (7) левая матрица представляет собой МОИ гипотетического варианта размещения, правая матрица — соответствующая ей ММР (6).

В случае, например, отказа процессорного модуля  $p_3$ , на который назначена подпрограмма, необходимо провести оперативное переназначение подпрограмм с учетом ближайшего резервного процессорного модуля  $l_2$ . В данном случае предлагается вариант замены, когда подпрограмма, назначенная ранее на процессор  $p_3$ , переназначается (фактически — сдвигается) на один модуль правее с учетом резервного модуля  $l_2$  [1]. В результате получаем вариант топологической модели мультимикроконтроллера, представленный на рисунке.



Для топологической модели, представленной на рисунке, получаем модель мультимикроконтроллера, заданную следующей матрицей  $\tilde{P}$ :

$$\tilde{P} = \left\{ \begin{array}{ccc} p_1 & p_2 & 0 \\ 0 & p_3 & p_4 \end{array} \right\}. \quad (8)$$

Соответствующая матрица ММР с учетом (8) выглядит следующим образом:

$$D = \begin{Bmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 1 & 2 \\ 2 & 1 & 0 & 1 \\ 3 & 2 & 1 & 0 \end{Bmatrix}. \quad (9)$$

Тогда вариант размещения с учетом (8), (9) будет задаваться следующим отображением:

$$M = \begin{Bmatrix} 0 & 15 & 0 & 7 \\ 15 & 0 & 10 & 3 \\ 0 & 4 & 0 & 10 \\ 7 & 3 & 4 & 0 \end{Bmatrix} \rightarrow D = \begin{Bmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 1 & 2 \\ 2 & 1 & 0 & 1 \\ 3 & 2 & 1 & 0 \end{Bmatrix}. \quad (10)$$

В результате подпрограмма, ранее назначенная на процессорный модуль  $p_4$  с кратчайшим расстоянием 1, переназначается на резервный процессор с соответствующим расстоянием 3. Следовательно, как видно из выражения (10), после введения резервного процессорного модуля требуется новое оперативное перераспределение задач. Кроме того, в статье [6] было показано, что подпрограммы (подзадачи), которым соответствует наибольший объем передачи данных, предпочтительнее назначать на процессорные модули, которым соответствуют минимальные кратчайшие расстояния  $D$  для уменьшения общего времени решения задачи.

Как видно из рассмотренного примера, в результате отказа процессорных модулей мультимикроконтроллера из-за перераспределения подпрограмм на резервные процессоры происходят значительное изменение топологии мультимикроконтроллерной системы и увеличение кратчайших маршрутов, вследствие чего ухудшается общее качество размещения подпрограмм, и как следствие — увеличивается общее время решения задачи. Следовательно, необходимо проводить новое планирование размещения подпрограмм фактически в новой топологической модели мультимикроконтроллера. Эту задачу можно выполнить с помощью метода, представленного в работе [6].

Предложенный способ планирования размещения с учетом отказа мультимикроконтроллерных процессорных модулей позволит уменьшить время восстановления работоспособности мультимикроконтроллера, благодаря чему существенно уменьшается время простоя и повышается производительность системы.

#### СПИСОК ЛИТЕРАТУРЫ

1. *Зотов И. В.* Организация и синтез микропрограммных мультимикроконтроллеров. Курск: Изд-во „Курск“, 1999. 368 с.
2. *Воеводин В. В., Воеводин Вл. В.* Параллельные вычисления. СПб: БХВ-Петербург, 2002. 608 с.
3. *Tanenbaum A. S.* Distributed Operation Systems. Prentice-Hall, 1994. 648 p.
4. *Корнеев В. В.* Параллельные вычислительные системы. М.: Нолидж, 1999. 340 с.
5. *Оре О.* Теория графов. М.: Наука, 1968. 352 с.
6. *Борзов Д. Б.* Метод снижения коммуникационной задержки путем субоптимального размещения задач в матричных базовых блоках кластера // Телекоммуникации. 2008. № 4. С. 21—25.
7. *Борзов Д. Б., Аль-Мараят Б. И., Тишкин А. П.* Акселератор планирования размещения задач в кластерных вычислительных системах высокой готовности // Изв. вузов. Приборостроение. 2008. Т. 51, № 2. С. 29—33.

#### *Сведения об авторе*

*Дмитрий Борисович Борзов*

— канд. техн. наук; Курский государственный технический университет, кафедра вычислительной техники; E-mail: borzovdb@kursknet.ru

Рекомендована кафедрой  
вычислительной техники

Поступила в редакцию  
12.09.08 г.

Д. Б. БОРЗОВ, С. А. ДЮБРЮКС, В. С. ТИТОВ

## МЕТОД ОБЪЕДИНЕНИЯ И РАЗДЕЛЕНИЯ ЦИКЛИЧЕСКИХ УЧАСТКОВ ПОСЛЕДОВАТЕЛЬНЫХ НАСЛЕДУЕМЫХ ПРОГРАММ

Предложен метод объединения/разделения циклических участков последовательных наследуемых программ, позволяющий объединять тела циклов и получать более длинные линейные участки, подлежащие последующему распараллеливанию. На основе данных о внутренней структуре программы проанализированы возможности ее эквивалентного преобразования к виду, позволяющему выполнять параллельно часть задач, назначенных ранее для последовательного выполнения.

**Ключевые слова:** распараллеливание, программа, объединение, разделение, цикл, линейка, участок, метод.

В настоящее время все большее распространение получает использование большого количества процессоров (до нескольких сотен) для решения поставленных задач [1]. В то же время значительная часть задач остается последовательными [2], в связи с этим для повышения производительности необходимо назначать на разные процессоры отдельные части одной задачи. Очевидно, что различные подзадачи при этом должны оставаться независимыми без потери логики исходного алгоритма, в связи с чем необходимо выявлять независимые подалгоритмы внутри последовательных программ. Значительная часть современных алгоритмов состоит из циклических участков [3], на обработку которых используется ощутимая часть ресурсов вычислительной системы.

В настоящей статье предложен метод объединения и разделения циклических участков последовательных исполняемых программ, развиваются идеи, представленные в работах [4, 5].

Предлагаемый метод позволяет объединять циклы с целью получения более длинных линейных участков для их последующего распараллеливания, а также сокращения количества операций приращения счетчиков циклов и проверок условий выхода из них. Метод основан на объединении тел смежных циклов последовательной наследуемой программы, операторы которых имеют одинаковый уровень вложенности.

Введем исходные обозначения.

1. Множество операторов задается массивом  $M = \|m_{ij}\|_{N \times 5}$ , где  $i = N$  — порядковые номера операторов,  $N$  — общее число операторов.

2.  $\forall j = 1, E_{m_{i1}} = N_{op_N}$ , где  $N_{op}$  — порядковые номера элементов массива  $M$ ,  $i = \overline{1, N}$ .

3.  $\forall j = 2, E_{m_{i2}} = V_N$ , где  $V$  — уровень вложенности  $i$ -го оператора,  $i = \overline{1, N}$ .

4.  $\forall j = 3, E_{m_{i3}} = C_{beg_N}$ , где  $C_{beg}$  — начальное значение счетчика цикла, в который вложен  $i$ -й оператор,  $i = \overline{1, N}$ .

5.  $\forall j = 4, E_{m_{i4}} = C_{end_N}$ , где  $C_{end}$  — конечное значение счетчика цикла, в который вложен  $i$ -й оператор,  $i = \overline{1, N}$ .

6.  $\forall j = 5, E_{m_{i5}} = C_{n_N}$ , где  $C_n$  — порядковые номера циклов,  $i = \overline{1, N}$ .

На первом этапе при использовании метода осуществляются поиск и выделение на описанном массивом  $M$  фрагменте программы участка из циклов, подлежащих объедине-

нию. Операторы, составляющие тела этих циклов, должны иметь одинаковый уровень вложенности.

Рассмотрим более подробно данный этап на примере следующего алгоритма. Счетчик  $C_t$  используется для подсчета числа операторов, составляющих выделяемый участок программы, переменная  $Level$  используется для перебора возможных значений уровня вложенности от первого до максимального для всего фрагмента программы значения  $LevMax$ , переменная  $Copy2$  используется для копирования нижней границы искомого участка,  $i$  и  $k$  являются вспомогательными переменными.

1.  $Level=1$ .
2.  $i=1$ .
3. Если  $V(i)=Level$ , то  $C_t=1$ ;  $A(C_t)=M(i)$ ;  $LowSide=i$  (иначе п. 9).
4.  $k=i+1$ .
5. Если  $V(k)=Level$ , то  $C_t=C_t+1$ ;  $A(C_t)=M(k)$  (иначе п. 8).
6.  $k=k+1$ .
7. Если  $k=N$ , то п. 8 (иначе п. 5).
8. Если  $C_t>1$ , то  $Copy2=LowSide$ ; переход на 2 этап (иначе п. 10).
9.  $i=i+1$ .
10. Если  $i=N$ , то п. 11 (иначе п. 3).
11.  $Level=Level+1$ .
12. Если  $Level>LevMax$ , то конец (иначе п. 2).

Данные о выделенных участках заносятся во вспомогательный массив  $A = \|a_{ij}\|_{N \times 5}$ , имеющий одинаковую структуру с массивом  $M$ .

Опишем работу приведенного алгоритма на следующем примере. Пусть дан массив  $M$  с числом операторов  $N=9$ , где максимальное значение уровня вложенности  $V$  равно 2 (табл. 1).

Таблица 1

$N_{op}$	$V$	$C_{beg}$	$C_{end}$	$C_n$
1	0	1	12	0
2	1	1	15	1
3	1	1	20	2
4	1	1	20	2
5	1	1	12	3
6	0	13	15	0
7	1	13	15	4
8	2	16	20	5
9	0	16	20	0

В п. 1 алгоритма переменной  $Level$  присваивается значение 1, т.е. выполняются поиск и выделение групп операторов первого уровня вложенности. В п. 2 счетчику  $i$  операторов присваивается значение 1. В результате значение первого элемента вектора  $V$  меньше значения переменной  $Level$ , поэтому содержимое счетчика  $i$  увеличивается на единицу ( $i=2$ ). Так как значение второго элемента вектора  $V$  равно значению переменной  $Level$ , то счетчику  $C_t$  операторов образуемой группы присваивается значение 1, а нижней границе  $LowSide$  будущего участка — значение  $i$ :  $LowSide=2$ . Элемент 2 массива  $M$  копируется на место элемента 1 вспомогательного массива  $A$ . В п. 4 инициализируется вспомогательная переменная  $k=3$ . Значение третьего элемента вектора  $V$  равно значению переменной  $Level$ , поэтому содержание счетчика  $C_t$  увеличивается на единицу:  $C_t=2$ . В то же время элемент 3 массива  $M$  перемещается (копируется) на место второго элемента вспомогательного массива  $A$ . В п. 6  $k=4$ . Так как  $k$  меньше общего числа операторов, осуществляется переход к п. 5. Значение четвертого элемента вектора  $V$  равно значению переменной  $Level$ , следовательно,  $C_t$  увеличивается на единицу:  $C_t=3$ . При этом элемент 4 массива  $M$  копируется на место элемента 3 вспомогательного массива  $A$  и  $k=5$ . Так как  $k<9$ , то осуществляется переход к п. 5. Так как значение пятого элемента вектора  $V$  равно значению переменной  $Level$ , содержимое счетчика  $C_t$  увеличивается

на единицу:  $C_t = 4$ , а элемент 5 массива  $M$  копируется на место элемента 4 вспомогательного массива  $A$ , при этом и увеличивается на единицу переменная  $k$ :  $k = 6$ . Значение шестого элемента вектора  $V$  не равно значению переменной  $Level$ , следовательно, осуществляется переход к п. 8. Так как  $C_t > 1$ , то искомый участок выбран, значение его нижней границы (2) сохраняется в переменной  $Copy2$ , осуществляется переход к следующему этапу.

На втором этапе происходит объединение всех циклов выделенного участка в один с минимальным значением счетчика  $C_{beg}$  с последующим разделением оставшихся участков, затем — объединение по новому минимальному значению  $C_{beg}$  среди счетчиков конца цикла с дальнейшим разделением и т.д., пока количество циклов объединения/разделения не станет равным числу циклов на участке, уменьшенному на единицу.

Исходными данными этапа 2 являются массив  $A = \parallel a_{ij} \parallel_{N \times 5}$  (табл. 2), значение  $C_t$  счетчика количества операторов выделенного на первом этапе участка, суммарное количество циклов этого участка  $MaxN_c$ , и его нижняя граница  $LowSide$ .

Таблица 2

$N_{op}$	$V$	$C_{beg}$	$C_{end}$	$C_n$
2	1	1	15	1
3	1	1	20	2
4	1	1	20	2
5	1	1	12	3

Этап состоит из многоитерационной процедуры поиска минимума среди элементов  $C_{end}$  с последующей модификацией элементов  $C_{beg}$  и  $C_n$  при объединении циклов участка и копирования информации, характеризующей операторы, при разделении циклов. Процесс представляется следующим алгоритмом, в котором  $HighSide$  — верхняя граница участка,  $x$  — счетчик числа итераций объединения/разделения,  $Min$  — переменная для поиска минимума среди конечных значений счетчиков циклов,  $Copy1$  — переменная для копирования значений границ участка,  $z$  — вспомогательная переменная:

1.  $HighSide = C_t$ ;  $LowSide = 1$ .
2.  $x = 1$ .
3. Если  $x \neq 1$ , то  $HighSide = LowSide + 1$ .
4.  $Min = C_{end}(LowSide)$ .
5.  $z = LowSide + 1$ .
6. Если  $C_{end}(z) = Min$  then  $Min = C_{end}(z)$ .
7.  $z = z + 1$ .
8. Если  $z > HighSide$ , то п. 9 (иначе п. 6).
9.  $Copy1 = HighSide$ .
10.  $z = LowSide$ .
11. Если  $C_{end}(z) > Min$ , то  $C_{end}(z) = Min$ ,  $C_{beg}(z + C_t) = Min + 1$ ;  $N_{op}(z + C_t) = N_{op}(z) + C_t$ ;  $C_n(z) = x$ .
12.  $C_n(z + C_t) = C_n(z)$ ;  $Copy1 = Copy1 + 1$ .
13.  $z = z + 1$ .
14. Если  $z > HighSide$ , то п. 14 (иначе п. 11).
15.  $LowSide = HighSide + 1$ ;  $HighSide = Copy1$ .
16.  $x = x + 1$ .
17. Если  $x = MaxN_c - 1$ , то переход на этап 3 (иначе п. 4).

Рассмотрим работу данного алгоритма на примере выходных данных первого этапа, которые состоят из массива  $A$  (табл. 2), нижней границы  $LowSide = 2$ , суммарного количества циклов участка  $MaxN_c = 3$  и длины участка, равной значению счетчика  $C_t = 4$ .

В п. 1 значению начальной верхней границы участка присваивается значение  $HighSide = C_t = 4$  и нижней границы —  $LowSide = 1$ , т.е. на первой итерации обработка ведется среди элементов  $a_{1,1} — a_{4,1}$ . В п. 2 счетчику  $x$  числа итераций присваивается начальное значение 1. На первой итерации модификация границ участка поиска минимума не производится,

за минимум принимается значение  $C_{\text{end}}(1)=15$ , соответствующее первому элементу массива  $A$ . Значение нижней границы участка, увеличенное на единицу, присваивается счетчику  $z$  перебора участка:  $z=2$ . Значение  $C_{\text{end}}(2)$  больше минимума, поэтому  $z$  увеличивается на единицу:  $z=3$ . Так как значение  $z$  меньше верхней границы, элемент вектора  $C_{\text{end}}(3)$  сравнивается с минимумом. Значение  $C_{\text{end}}(3)$  больше минимума, поэтому содержание счетчика увеличивается и становится равным четырем. Так как значение  $z$  равно верхней границе, элемент вектора  $C_{\text{end}}(4)$  сравнивается с минимумом.  $C_{\text{end}}(4)$  больше минимума, следовательно, минимуму присваивается значение  $C_{\text{end}}(4)=12$ . Далее  $z = 5$  — что больше верхней границы, следовательно, осуществляется переход к п. 9 и копируется значение верхней границы  $\text{HighSide}$  в переменную  $\text{Copy1}$ . В п. 10 счетчику  $z$  нижней границы перебора участка присваивается значение 1. Так как  $C_{\text{end}}(1)$  больше минимума, то в п. 11 минимуму присваивается значение  $C_{\text{end}}(1)$ , а элемент массива  $A(1)$  копируется в элемент  $A(5)$ , при этом происходит модификация значений  $C_{\text{end}}(1)=\text{Min}=12$ ;  $C_{\text{beg}}(5)=\text{Min}+1=13$ ,  $C_n(1)=1$  и  $C_n(5)=4$ , а также увеличивается значение копии верхней границы:  $\text{Copy1}=5$ . В п. 12 осуществляется приращение значения  $z$ :  $z=2$ ; Так как  $z<4$ , то происходит переход к п. 11. Поскольку  $C_{\text{end}}(2)$  больше минимума, то в п. 11 элементу вектора  $C_{\text{end}}(2)$  присваивается значение минимума, при этом элемент массива  $A(2)$  копируется в элемент  $A(6)$ , осуществляются модификация значений  $C_{\text{end}}(1)=\text{Min}=12$ ,  $C_{\text{beg}}(5)=\text{Min}+1=13$ ,  $C_n(2)=1$ ,  $C_n(6)=5$  и увеличение значения копии нижней границы  $\text{Copy1}=\text{Copy1}+1=6$ . Далее значение  $z$  увеличивается на единицу:  $z=3$ . Так как  $z<4$ , то осуществляется переход к п. 11. Элемент вектора  $C_{\text{end}}(3)$  больше минимума, поэтому в п. 11 ему присваивается значение минимума, а элемент массива  $A(3)$  копируется в элемент  $A(7)$ , при этом происходит модификация следующих значений:  $C_{\text{end}}(1)=\text{Min}=12$ ,  $C_{\text{beg}}(5)=\text{Min}+1=13$ ,  $C_n(3)=1$ ,  $C_n(7)=6$ . Затем происходит увеличение на единицу значения копии нижней границы и счетчика  $z$ :  $\text{Copy1}=7$ ,  $z=4$ . Так как  $z=4$ , то происходит переход к п. 11. Значение  $C_{\text{end}}(4)$  равно минимуму, следовательно, осуществляются переход к п. 13 и приращение на единицу значения  $z$ :  $z=5$ . Так как  $z>4$ , то переходим к п. 14 и модифицируем границы участка поиска минимума:  $\text{LowSide}=\text{HighSide}+1=5$ ;  $\text{HighSide}=\text{Copy1}=7$ . В п. 15 содержимое счетчика итераций  $x$  увеличивается, поиск минимума и копирование элементов массива  $A$ , меньших минимума, повторяются. В данном примере необходимо выполнить три итерации, после чего массив  $A$  примет вид, представленный в табл. 3. После выполнения последней итерации управление передается третьему этапу.

Таблица 3

$N_{\text{оп}}$	$V$	$C_{\text{beg}}$	$C_{\text{end}}$	$C_n$
2	1	1	12	1
3	1	1	12	1
4	1	1	12	1
5	1	1	12	1
6	1	13	15	2
7	1	13	15	2
8	1	13	15	2
9	1	16	20	3
10	1	16	20	3

На третьем этапе происходит вставка в исходный массив  $M$  вспомогательного массива  $A$  с помощью нижеприведенного алгоритма, в котором значения переменных  $\text{Copy1}$  и  $\text{Copy2}$  заимствуются из предыдущих этапов,  $s$  и  $h$  — вспомогательные переменные для индексации массивов  $A$  и  $M$ .

1.  $h=\text{Copy1} - \text{Copy2}$ .
2.  $s=N$ .
3.  $M(s+h)=M(s)$ .
4.  $s=s-1$ .
5. Если  $s<\text{Copy2}$ , то п. 6 (иначе п. 3).

6.  $s=i$ .
7.  $M(s)=A(s)$ .
8.  $s=s+1$ .
9. Если  $s>C_{op}2$ , то  $N=N+h$ ;  $i=C_{op}1+1$ ; передача управления на этап 1 (иначе п. 7).

В приведенном алгоритме происходит нахождение длины  $h$  отрезка массива  $M$ , который необходимо освободить для вставки массива  $A$  (п. 1 алгоритма), затем перезапись элементов массива  $M$  в память на расстояние  $h$  (пп. 2—5 алгоритма), а также вставка элементов массива  $A$  в массив  $M$  (пп. 6—9 алгоритма), что при копировании всех данных, характеризующих операторы, приводит к эквивалентному преобразованию программы. В п. 9 число операторов программы увеличивается на  $h$ , счетчик операторов — на  $C_{op}1+1$ , и управление передается п. 10 первого этапа. Аналогичным образом осуществляются обработка оставшихся элементов вектора  $V$  и поиск участков, содержащих операторы уровня вложенности 1, после чего содержимое счетчика  $x$  уровня вложенности на первом этапе увеличивается на 1. Так как при  $x=2$  участков для объединения/разделения в рассматриваемом фрагменте программы нет, массив  $M$  примет окончательный вид (табл. 4).

Таблица 4

$N_{op}$	$V$	$C_{beg}$	$C_{end}$	$C_n$
1	0	1	12	0
2	1	1	12	1
3	1	1	12	1
4	1	1	12	1
5	1	1	12	1
6	1	13	15	2
7	1	13	15	2
8	1	13	15	2
9	1	16	20	3
10	1	16	20	3
11	0	13	15	0
12	1	13	15	4
13	2	16	20	5
14	0	16	20	0

Предложенный метод позволяет сократить число приращений счетчиков и проверок выполнения условий окончания циклов, а также выполнять некоторые циклы параллельно на определенном пространстве итераций, что ведет к повышению производительности многопроцессорных систем, обрабатывающих алгоритм. Данный метод имеет ряд недостатков, одним из которых является возможность его применения только к циклам, следующим друг за другом. Поэтому возникает проблема предварительной перегруппировки операторов программы, исследование которой планируется в дальнейшем.

## СПИСОК ЛИТЕРАТУРЫ

1. Трахтенгерц Э. А. Введение в теорию анализа и распараллеливания программ ЭВМ в процессе трансляции. М.: Наука, 1981. С. 184—187.
2. Воеводин В. В. Математические модели и методы в параллельных системах. М.: Наука, 1986.
3. Дюбрюкс С. А., Борзов Д. Б., Титов В. С. Выявление параллелизма внутри линейных участков последовательных программ со связями по управлению // Сб. тр. XIV Междунар. науч.-технич. конф. „Машиностроение и техносфера XXI века“. Т. 2. Донецк, 2007. С. 26—30.
4. Борзов Д. Б., Дюбрюкс С. А., Титов В. С. Метод выявления параллелизма внутри линейных участков последовательных программ и его аппаратная реализация // Изв. вузов. Приборостроение. 2008. Т. 51, № 2. С. 34—38.
5. Воеводин В. В., Воеводин Вл. В. Параллельные вычисления. СПб: БХВ-Петербург, 2002. 608 с.



*Сведения об авторах*

- Дмитрий Борисович Борзов** — канд. техн. наук, доцент; Курский государственный технический университет, кафедра вычислительной техники;  
E-mail: borzovdb@kursknet.ru
- Сергей Александрович Дюбрюкс** — аспирант; Курский государственный технический университет, кафедра вычислительной техники; E-mail: serhio5551@yandex.ru
- Виталий Семенович Титов** — д-р техн. наук, профессор; Курский государственный технический университет, кафедра вычислительной техники; зав. кафедрой;  
E-mail: titov@vt.kstu.kursk.ru

Рекомендована кафедрой  
вычислительной техники

Поступила в редакцию  
12.09.08 г.

М. И. ТРУФАНОВ

## РАСПОЗНАЮЩИЕ ОПТИКО-ЭЛЕКТРОННЫЕ УСТРОЙСТВА МЕДИЦИНСКОЙ ЭКСПРЕСС-ДИАГНОСТИКИ

Обоснована необходимость построения распознающих оптико-электронных устройств экспресс-диагностики заболеваний и определена область их применения. Представлены принципы построения устройств экспресс-диагностики и улучшения качества медицинских изображений для диагностики тромбозов голени и тромбозомболических осложнений, эндоскопии, диагностики глазодвигательного аппарата.

*Ключевые слова:* оптико-электронное устройство, медицинские изображения, экспресс-диагностика, распознавание.

Одной из проблем современного здравоохранения является низкий уровень оснащенности средствами диагностики медицинских учреждений небольших населенных пунктов и низкая квалификация медицинского персонала. Это зачастую обуславливает несвоевременное обнаружение заболеваний, лечение которых на более поздней стадии малоэффективно.

В этой связи актуальной задачей является создание и внедрение приборов экспресс-диагностики заболеваний, которые могут быть использованы низкоквалифицированным медицинским персоналом, что позволит выявлять заболевания уже при первичном медицинском осмотре, охватывающем широкие круги населения, и при необходимости направлять пациента на более детальное обследование.

Современный уровень развития вычислительной техники и систем обработки изображений позволяет рекомендовать для экспресс-диагностики заболеваний оптико-электронные устройства, характеризующиеся низкой стоимостью и габаритами, высоким уровнем автоматизации, позволяющие бесконтактно и быстро диагностировать признаки заболевания.

К серьезным заболеваниям, требующим своевременного обнаружения и лечения, относятся:

- заболевания, связанные с тромбозом вен нижних конечностей и тромбозомболией легочной артерии [1];
- заболевания желудка, двенадцатиперстной кишки, диагностируемые эндоскопическими приборами;
- заболевания глазодвигательного аппарата, например косоглазие, которое может развиваться у маленьких детей и вследствие сложности работы с детьми является сложно диагностируемым [2].

Важной характеристикой любой оптико-электронной системы является качество получаемого изображения. Получение четкого изображения особенно актуально при решении задач диагностики в медицине.

В частности, при эндоскопии повышение качества изображения позволяет существенно увеличить точность диагностики, что способствует своевременному лечению и выявлению

заболевания в начале его развития. Вследствие конструктивных ограничений оконечной части эндоскопа исходное изображение характеризуется рядом аберраций, а необходимость использования яркого источника света и влажная поверхность вызывают блики, что делает первоочередной задачей повышения качества изображения при эндоскопии. Для коррекции аберрационных искажений разработаны цифровые устройства обработки изображений [3—5].

При диагностике тромбоэмболических осложнений качество изображения не является основополагающим, однако при решении данной задачи возникает необходимость разработки новых методов калибровки оптико-электронных устройств [6, 7] и доработки методов анализа стереоскопических изображений [8].

Актуальность разработки методов и оптико-электронных средств диагностики офтальмологических заболеваний обусловлена в первую очередь двумя факторами: высокой стоимостью современных комплексных средств диагностики, которые могут приобрести только крупные клиники, и вынужденным применением устаревших методик, основанных на субъективном выявлении врачом заболевания [9]. Разработка новых диагностирующих оптико-электронных устройств для офтальмологии позволит оснастить различные медицинские учреждения и выявлять ранние стадии заболевания у широкого круга населения, что будет способствовать значительному повышению эффективности лечения.

Рассмотрим принципы построения оптико-электронных средств экспресс-диагностики.

Тромбоз глубоких вен нижних конечностей (ТГВНК) и связанная с ним тромбоэмболия легочной артерии (ТЭЛА) вызывают осложнения, которые могут приводить к смертельному исходу. Применяемые в настоящее время средства диагностики не в полной мере удовлетворяют требованиям по стоимости, качеству и своевременности диагностики этого заболевания [1].

Предлагаемый автоматизированный оптико-электронный комплекс диагностики тромбоэмболических осложнений предназначен для малоинвазивной диагностики тромбоза глубоких вен голени и прогнозирования риска возникновения ТЭЛА, а также ее исходов при окклюзивных и неокклюзивных формах венозных тромбозов [10]. В состав комплекса входят:

- три оптико-электронных датчика (ОЭД);
- компьютер;
- специальное программное обеспечение.

Компьютер с подключенными ОЭД предназначен для оптико-электронной диагностики тромбоза глубоких вен голени [8], основанной на получении с различных позиций изображений голени в свободном состоянии и после пережатия поверхностных вен компрессионной манжетой с последующим восстановлением трехмерной поверхности голени по полученным изображениям путем сопоставления одинаковых точек голени на различных изображениях, определением множества трехмерных координат точек голени, формированием трехмерной поверхности голени и определением объема голени до и после пережатия поверхностных вен.

Полученные в результате диагностики данные используются как для направления пациента на дальнейшее более детальное и глубокое обследование или назначения лечения, так и в качестве входных данных (полученных также другими методами) для прогнозирования развития ТГВНК и ТЭЛА созданным программным обеспечением [1].

Прогнозирование заболевания заключается в выявлении совокупности его признаков, полученной в результате опроса пациента, а также инструментальными средствами, и распознавании формы, типа ТЭЛА посредством расчета функций принадлежности входного описания заболевания известным классам и выбора функции принадлежности с максимальным значением, которая и будет соответствовать известному исходу заболевания (выздоровлению, смерти), типу ТЭЛА (массивной, мелких ветвей), форме ТЭЛА (острой, рецидивирующей, молниеносной), типу тромбоза (эмбологенному, неэмбологенному).

Новизна предложенных решений заключается в использовании аппарата нечеткой логики для прогнозирования тромбоэмболических осложнений, что обеспечивает автоматизацию

процесса принятия решения о наличии заболевания и его дальнейшем развитии при неполной или искаженной информации, тогда как известные методы, основанные на прогнозе по факторам риска, дают неточный результат или не могут быть использованы для прогнозирования и диагностики в таких условиях. Новизна предлагаемых решений заключается также в использовании оптико-электронных устройств для диагностики тромбоэмболических осложнений, что обеспечивает:

— существенное снижение стоимости комплекса диагностики тромбоэмболических осложнений по сравнению с системой ультразвуковой диагностики [10];

— диагностику тромбозов нижних конечностей на ранней стадии с большей по сравнению с известными методами достоверностью (за исключением метода ультразвуковой диагностики);

— автоматизацию процесса диагностики.

Для повышения качества эндоскопических изображений предложено устройство, обеспечивающее коррекцию бликов, возникающих вследствие отражения излучения яркого осветителя эндоскопа от слизистой оболочки и дисторсии оптической системы.

Принцип функционирования устройства заключается в коррекции дисторсии оптической системы специализированным устройством (или последующей программной обработкой) при получении изображения и последующем восстановлении искаженных бликами областей изображения двумя различными способами в зависимости от площади блика: медианным фильтром (при малой площади блика) и уменьшением чувствительности оптико-электронного датчика и последующей заменой искаженной области на исходном изображении на соответствующую область дополнительно полученного изображения (при большой площади) [11].

Предлагаемый подход к обработке эндоскопических изображений слизистой оболочки пищевода, желудка и двенадцатиперстной кишки позволит повысить точность выявления патологических изменений и правильность постановки диагноза, что будет способствовать оптимизации терапии, улучшению прогноза заболевания, сокращению сроков лечения, уменьшению экономических затрат.

Анализ известных технических решений и способов диагностики глазодвигательного аппарата, выявления косоглазия и других офтальмологических заболеваний позволил установить следующее [9]:

— в настоящее время на практике применяются в основном средства качественной, а не количественной диагностики косоглазия, которые не претерпели принципиальных изменений за последние десятилетия;

— создаются оптико-электронные (телевизионные) средства, анализирующие изображения глаз в процессе наблюдения человеком заданной точки, позволяющие выявить косоглазие на ранней стадии. Недостатками существующих средств являются сложность их практического применения и отсутствие функции измерения направлений наблюдения в трехмерном пространстве, что снижает их точность;

— актуальна разработка бинокулярной системы технического зрения, обеспечивающей отслеживание движений глаза в пространстве и на их основе диагностирующей косоглазие и отклонения глазодвигательного аппарата.

Предлагаемое бинокулярное оптико-электронное устройство [9] предназначено для диагностики заболеваний глазодвигательного аппарата и предположительно будет отличаться от известных инструментальных средств аналогичного назначения простотой практического применения, большей точностью, низкой стоимостью, малыми габаритами, высоким уровнем автоматизации.

Устройство реализует принцип диагностики косоглазия, основанный на измерении параметров движений зрачков глаз при фиксации взгляда человека на заданном врачом объекте по изображениям, поступающим с двух оптико-электронных датчиков, обеспечивающих трехмерное восприятие.

Устройство может быть применено при решении других медицинских задач, связанных с анализом движения зрачков, например, диагностике нистагма и состояния вестибулярного аппарата человека.

Таким образом, представлены подходы к обработке и распознаванию изображений, позволяющие создавать распознающие оптико-электронные устройства для ранней экспресс-диагностики заболеваний. Перспективный результат, являющийся развитием разрабатываемых методов и оптико-электронных средств, заключается в создании новых систем распознавания образов и анализа изображений, применяемых в медицине при построении эффективных диагностирующих систем различного назначения, характеризующихся высоким уровнем автоматизации, малой стоимостью и возможностью применения медицинским персоналом со средним образованием.

#### СПИСОК ЛИТЕРАТУРЫ

1. Прогнозирование тромбоэмболических осложнений при травме / В. А. Лазаренко, В. Н. Мишустин, В. С. Титов, М. И. Труфанов. Курск: КГМУ, 2004. 148 с.
2. Ковалевский Е. И. Глазные болезни. 2-е изд. М.: Медицина, 1980. 432 с.
3. Пат. № 2295153 РФ, МКИ G09K32/00. Корректирующее устройство ввода изображения в ЭВМ / М. И. Труфанов, Д. В. Титов. Заявл. 4.07.2005, опубл. 10.03.2007. Б.И. № 7. 6 с.
4. Пат. № 2292023 РФ, МКИ G01M11/02. Способ определения комы оптической системы / А. Н. Стрелкова, В. С. Титов, М. И. Труфанов. Заявл. 24.05.2005, опубл. 20.01.2007. Б.И. № 2. 5 с.
5. Пат. № 2295712 РФ, МКИ G01M11/02. Способ определения коэффициентов сферической аберрации / Е. И. Бугаенко, В. С. Титов, М. И. Труфанов. Заявл. 10.09.2006, опубл. 20.03.2007. Б.И. № 8. 6 с.
6. Пат. №2321888 РФ, МКИ G01M11/02, G06K9/32. Способ калибровки дисторсии оптических систем / А. Н. Стрелкова, Д. В. Титов, М. И. Труфанов. Заявл. 16.10.2006, опубл. 10.04.2008. Б.И. № 10. 5 с.
7. Пат. № 2286598 РФ, МКИ G09K32. Способ внешней калибровки бинокулярной системы технического зрения / С. В. Дегтярев, В. С. Титов, М. И. Труфанов, В. А. Денисюк. Заявл. 01.03.2005, опубл. 27.10.2006. Б.И. № 30. 15 с.
8. Пат. № 2314026 РФ, МКИ А61В5/103. Способ оптико-электронной диагностики тромбоза глубоких вен голени / В. Н. Мишустин, Н. Н. Мишустина, С. Ю. Мирошниченко, В. С. Титов, М. И. Труфанов. Заявл. 10.01.2006, опубл. 10.01.2008. Б.И. № 1. 11 с.
9. Принципы функционирования бинокулярного оптико-электронного устройства для диагностики отклонений глазодвигательного аппарата / В. Н. Гридин, В. С. Титов, М. И. Труфанов // Изв. вузов. Приборостроение. 2008. Т. 51, № 2. С. 48—53.
10. Автоматизированный аппаратно-программный комплекс диагностики тромбоэмболических осложнений / В. Н. Гридин, В. Н. Мишустин, О. Б. Тарасова, В. С. Титов, М. И. Труфанов // Информационные технологии в проектировании и производстве. 2008. № 2. С. 79—82.
11. Устройство повышения качества изображения при эндоскопии / А. Н. Стрелкова, А. А. Степченко // Изв. вузов. Приборостроение. 2008. Т. 51, № 2. С. 54—58.

#### *Сведения об авторе*

**Максим Игоревич Труфанов**

— канд. техн. наук; Курский государственный технический университет, кафедра вычислительной техники;  
E-mail: tmi@pub.sovtest.ru, temp1202@mail.ru

Рекомендована кафедрой  
вычислительной техники

Поступила в редакцию  
12.09.08 г.

А. Н. СТРЕЛКОВА, М. И. ТРУФАНОВ, А. А. СТЕПЧЕНКО

## МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ВОССТАНОВЛЕНИЯ ЭНДОСКОПИЧЕСКИХ ИЗОБРАЖЕНИЙ

Представлена математическая модель восстановления эндоскопических изображений, основанная на коррекции изображений двумя различными способами в зависимости от результата спектрального анализа и позволяющая повысить качество формирования эндоскопических образов.

**Ключевые слова:** эндоскопия, обработка медицинских изображений, блики, преобразование Фурье, спектральный анализ.

Разработка и широкое внедрение в клиническую практику современных эндоскопических методов исследования значительно расширили диагностические и лечебные возможности практически во всех областях медицины, в частности, в гастроэнтерологии.

Однако несмотря на постоянное усовершенствование эндоскопов в настоящее время перед их разработчиками остается нерешенной задача, связанная со значительной потерей яркости света в месте стыковки фиброволокна с узлом эндоскопа, через который осуществляется передача света. Во избежание потери яркости можно применять более мощные источники света, однако при увеличении интенсивности освещения существенно возрастает количество засвеченных областей и бликов на получаемом эндоскопом изображении, что значительно снижает информативность полученного изображения, закрывая от наблюдателя области исследуемого участка. Коррекция бликов на изображении позволит сделать доступными для наблюдения засвеченные области, тем самым повысив информативность изображения. В соответствии с этим актуальна задача повышения качества эндоскопического изображения посредством его цифровой обработки в режиме реального времени.

Способ определения засвеченных областей на эндоскопическом изображении основан на использовании априорной информации о том, что на исследуемом не искаженном бликами кадре наличие ярких (в частности, белых) точек исключено (цвет полости желудка естественный) [1].

Для упрощения распознавания на изображении засвеченных областей и бликов производится бинаризация эндоскопического изображения. На рис. 1 приведены эндоскопические изображения до (а) и после (б) бинаризации.

Для каждой точки изображения значение яркости усредняется по трем цветовым каналам:

$$I(x, y) = \frac{I_R(x, y) + I_G(x, y) + I_B(x, y)}{3}, \quad (1)$$

где  $(x, y)$  — координаты текущей точки,  $I(x, y)$  — усредненное значение яркости в текущей точке,  $I_R(x, y)$ ,  $I_G(x, y)$ ,  $I_B(x, y)$  — значение яркости красного, зеленого и синего канала соответственно. Таким образом, после усреднения по трем цветовым каналам получают значение яркости, соответствующее градациям серого.

После проводят бинаризацию [2]

$$I_b(x, y) = \begin{cases} 1 & \text{при } I(x, y) > tI_{im}, \\ 0 & \text{при } I(x, y) < tI_{im}, \end{cases} \quad (2)$$

где  $I_b(x, y)$  — значение яркости точки после бинаризации: 1 соответствует белому цвету, 0 — черному,  $t$  — пороговая величина яркости,  $I_{im}$  — среднее значение яркости точек всего изображения, определяемое следующим образом:

$$I_{im} = \frac{\sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} I(x, y)}{XY}, \quad (3)$$

где  $X, Y$  — количество точек изображения по горизонтали и вертикали соответственно.

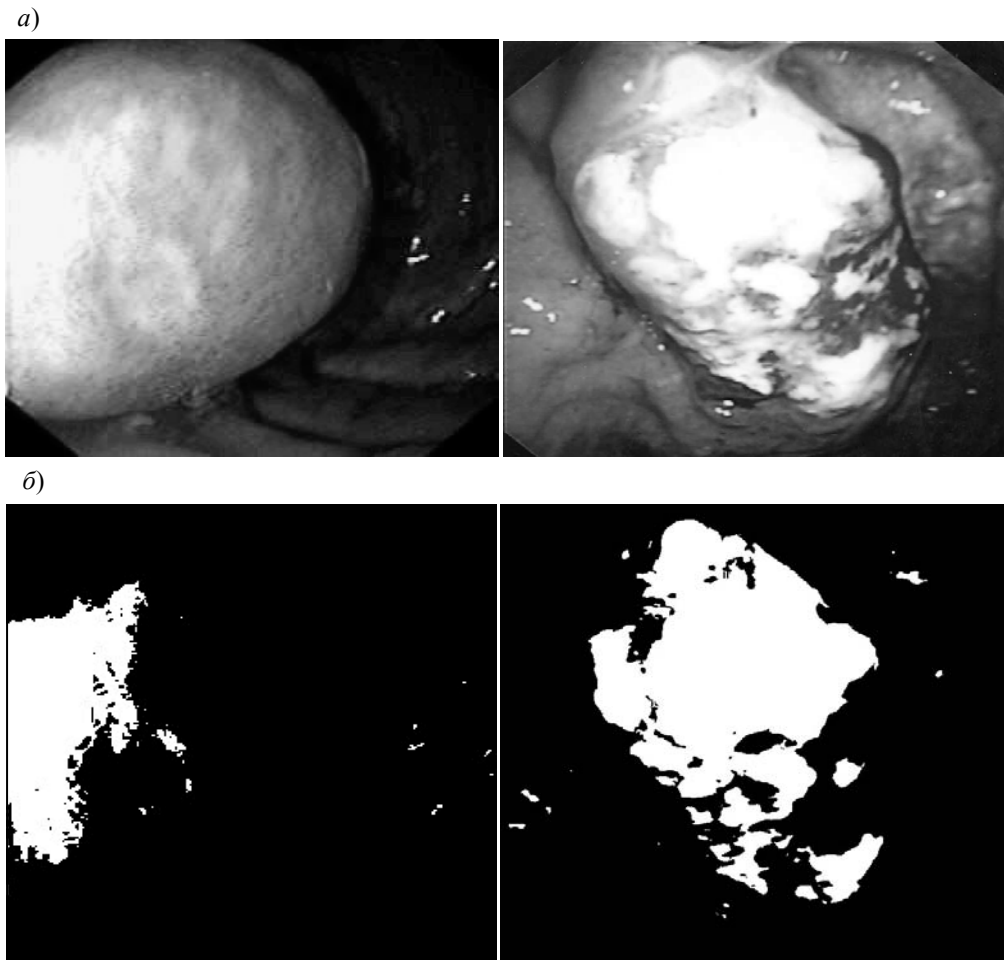


Рис. 1

После бинаризации полученного изображения яркость точек, составляющих засвеченные области и блики, примет значение „1“, в то время как значение яркости точек неискаженных областей эндоскопического изображения будет „0“.

При анализе полученного бинаризованного изображения его рассматривают как двумерный сигнал и для определения наличия на нем бликов и засвеченных областей, а также для анализа их площади применяют двумерное дискретное преобразование Фурье (ДПФ) [3]:

$$F(x', y') = \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} I_b(x, y) e^{-jxx' \frac{2\pi}{X}} e^{-jyy' \frac{2\pi}{Y}}, \quad (4)$$

где  $F(x', y')$  — спектр сигнала,  $x'$  и  $y'$  — номера базисных функций (коэффициентов двумерного ДПФ, при которых эти функции находятся), здесь  $x$  и  $y$  выполняют роль аргументов базисных функций,  $X \times Y$  — размерность исходного сигнала, соответствующая размерности спектра.

Для определения площади бликов и засвеченных областей производят анализ полученного спектра. Поскольку формула (4) содержит мнимую единицу  $j$ , каждое значение спектра является комплексным числом. Поэтому при анализе спектра рассматривают лишь значения действительных частей его элементов. Спектр может быть представлен в виде трехмерной поверхности (рис. 2, *а* — эндоскопическое изображение с усредненными значениями яркости точек, *б* — бинаризованное эндоскопическое изображение, *в* — трехмерная поверхность спектра).

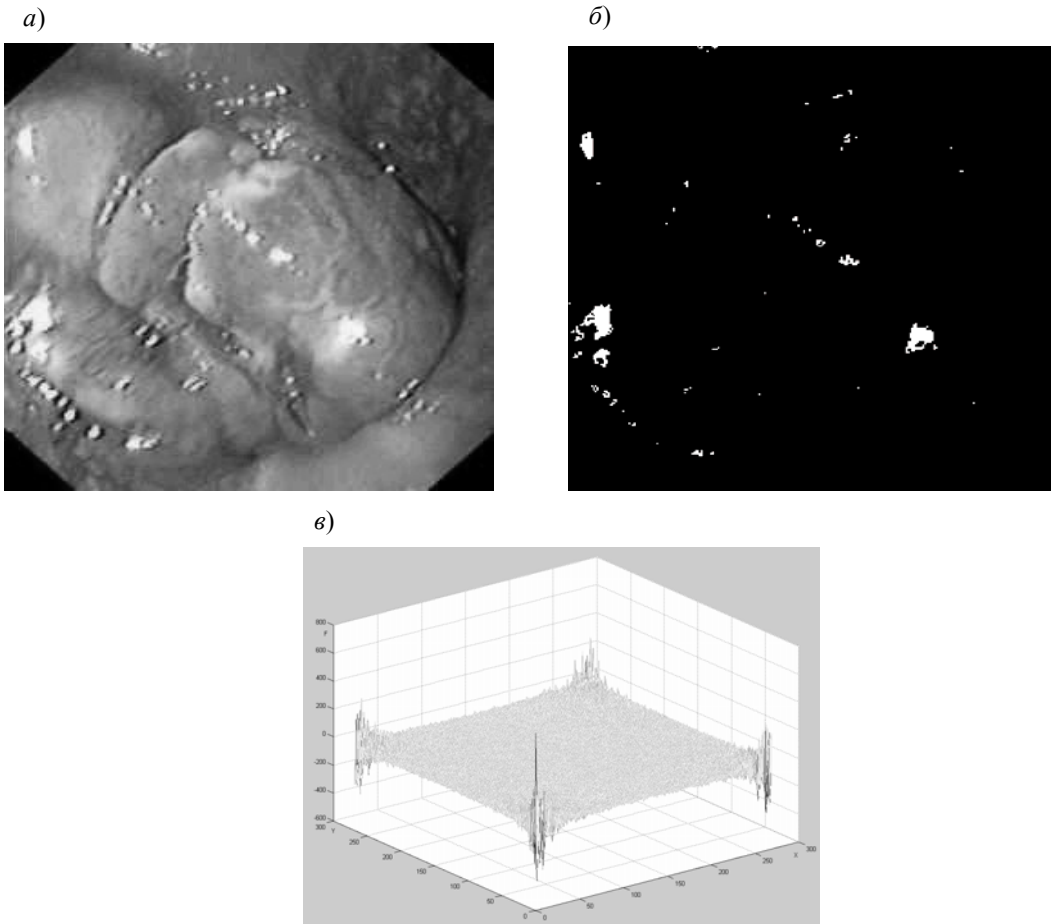


Рис. 2

При этом по осям абсцисс и ординат откладывают количество пикселей изображения по горизонтали и вертикали соответственно, а по оси аппликат — значение вещественной части спектра. Однако подобная форма представления не является удобной для последующего анализа, поэтому рассматривают проекцию поверхности спектра на плоскость  $OFX$  (рис. 3) [3].

Анализ проекции спектра заключается в выборе и оценке модуля максимального значения вещественной части спектра  $F_{\max}$ , и чем больше это значение, тем больше на бинаризованном изображении площади засвеченных областей [4].

Следовательно, перед началом анализа определяют пороговую величину  $T_F$ . Коррекция засвеченных областей и бликов выполняются в соответствии со следующими выражениями:

$$I'(x, y) = \begin{cases} \text{Ch}(I(x, y), I_b(x, y), I_{ad}(x, y)) & \text{при } F_{\max} > T_F, I_b(x, y) = 1, \\ MF(I(x, y), I_b(x, y)) & \text{при } F_{\max} < T_F, I_b(x, y) = 1, \\ I(x, y) & \text{при } I_b(x, y) = 0, \end{cases} \quad (5)$$



где  $I(x, y)$  — исходное изображение,  $I'(x, y)$  — изображение после коррекции,  $F_{\max}$  — модуль максимального значения спектра,  $\text{Ch}(I(x, y), I_b(x, y), I_{ad}(x, y))$  — функция замены точки исходного изображения  $I(x, y)$  точкой в той же позиции дополнительно полученного кадра при уменьшенной интенсивности источника освещения эндоскопа  $I_{ad}(x, y)$ ,  $MF(I(x, y), I_b(x, y))$  — функция медианного фильтра к текущей точке.

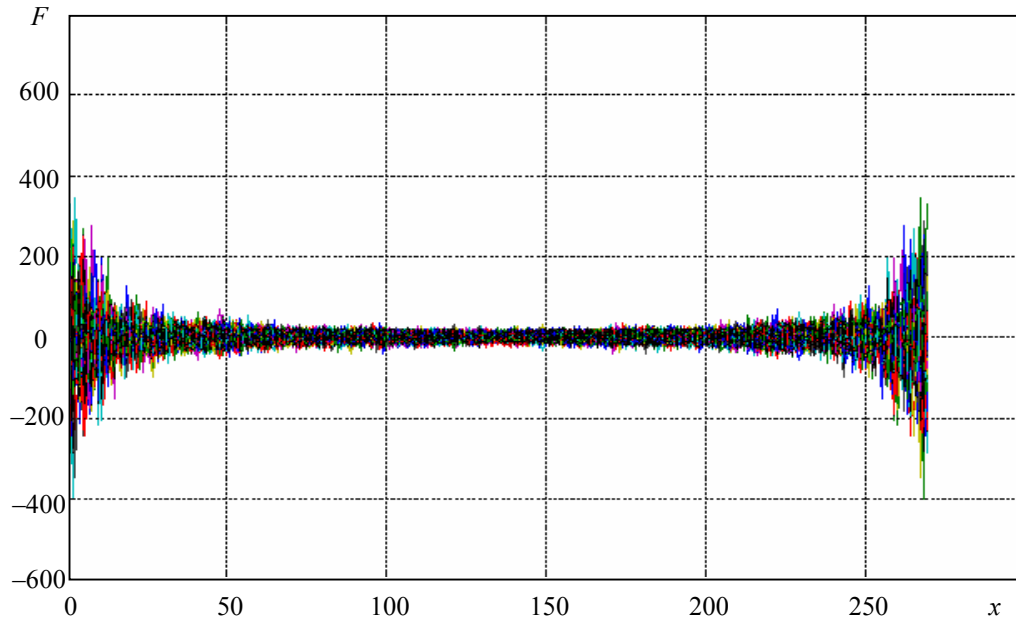


Рис. 3

Медианная фильтрация основана на замене значения яркости точки по всем трем цветовым каналам на медиану последовательности, составленной из значений окружающих ее точек. При медианной фильтрации изображений используется двумерное окно (апертура фильтра)  $A$ , имеющее центральную симметрию, при этом его центр располагается в текущей точке фильтрации  $p(x, y)$  [5].

При этом значение яркости в точке  $p(x, y)$  будет определено следующим образом:

$$I'(x, y) = M(p(x+r, y+c)), \quad r \in (-A; A), \quad c \in (-A, A), \quad (6)$$

где  $M(p(x+r, y+c))$  — медиана для текущей точки.

При этом медианой яркости множества описанных точек является средний по значению член ряда, получающегося при упорядочении последовательности по возрастанию.

Таким образом, в настоящей работе представлена математическая модель восстановления эндоскопических изображений, на основе которой возможно разработать способы и устройства, позволяющие повысить качество эндоскопических изображений в режиме реального времени.

Работа выполнена при поддержке Фонда содействия развитию малых форм предприятий в научно-технической сфере „У.М.Н.И.К.“ (договор № 0806, 2008 г., проект № 8555).

#### СПИСОК ЛИТЕРАТУРЫ

1. Прэтт У. Цифровая обработка изображений / Пер. с англ. М.: Мир, 1982. Кн. 1. 312 с.; Кн. 2. 493 с.
2. Tsai R. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses // IEEE Trans. Rob. Autom. RA-3 (4). 1987. P. 323—344.
3. Методы компьютерной обработки изображений / Под ред. В. А. Сойфера. М.: Физмалит, 2001. 784 с.

4. Пат. №2295153 РФ, МПК<sup>7</sup> G 06 K 9/32. Корректирующее устройство ввода изображения в ЭВМ / М. И. Труфанов, Д. В. Титов. Оpubл. 10.03. 2007. Б.И. № 7. 8 с.
5. Titov D. V., Shirabakina T. A. Correction Device of Distortion // „Information and Telecommunication Technologies in Intelligent Systems“. Proc. 5th Int. Conf. Spain, 2007. P. 122—124.

**Сведения об авторах**

- Александра Николаевна Стрелкова** — аспирант; Курский государственный технический университет, кафедра вычислительной техники; E-mail: strelkova@bk.ru
- Максим Игоревич Труфанов** — канд. техн. наук; Курский государственный технический университет, кафедра вычислительной техники; E-mail: temp1202@mail.ru
- Александр Александрович Степченко** — канд. мед. наук, доцент; Курский государственный медицинский университет, кафедра внутренних болезней

Рекомендована кафедрой  
вычислительной техники

Поступила в редакцию  
12.09.08 г.

УДК 680.5.01:621.384

Д. В. ТИТОВ, Т. А. ШИРАБАКИНА

## МОДУЛЬ ЦИФРОВОЙ КОРРЕКЦИИ ДИСТОРСИИ ИЗОБРАЖЕНИЯ

Представлен модуль цифровой коррекции дисторсии изображения, основанный на автоматическом выборе объектов на изображении, по которым выявляется дисторсия, определении коэффициента радиальной дисторсии с последующим расчетом истинного положения всех точек искаженного изображения, а также яркостной коррекции.

**Ключевые слова:** цифровое изображение, пиксел, калибровка, оптико-электронное устройство, дисторсия, искажение, яркость.

В настоящее время в различных областях науки и промышленности широко используются оптико-электронные устройства. Их важнейшей характеристикой является точность формирования изображения, которая зависит от абберационных погрешностей оптической системы [1]. Одна из аббераций — дисторсия, при которой прямые линии объекта проецируются в кривые.

При построении малогабаритных оптико-электронных приборов перспективно использовать цифровые устройства, обеспечивающие коррекцию дисторсии без изменения конструкции видеодатчиков оптико-электронных устройств [2].

Был проведен анализ существующих модулей цифровой коррекции дисторсии, в ходе которого выявлены такие недостатки, как низкая точность ввода изображения, а также малое быстродействие. Для устранения этих недостатков модули цифровой коррекции были доработаны.

В настоящей статье разработана математическая модель, на базе которой построен модуль цифровой коррекции дисторсии изображения, выполняемой в соответствии с теоретическими положениями, представленными в работе [3].

Искажения, вызванные дисторсией, определяются по формулам

$$\begin{aligned}\Delta x_r &= x(k_1 r^2 + k_2 r^4 + \dots + k_n r^{2n}), \\ \Delta y_r &= y(k_1 r^2 + k_2 r^4 + \dots + k_n r^{2n}),\end{aligned}\tag{1}$$

где  $(\Delta x_r, \Delta y_r)$  — смещение точки изображения относительно ее истинного положения;  $k_1, k_2, \dots, k_n$  — коэффициенты радиальной дисторсии [4];  $r = (x^2 + y^2)^{1/2}$  — расстояние от центра кадра до точки с координатами  $x, y$ .

Коррекция радиальной дисторсии осуществляется переносом точек  $x$  и  $y$  изображения в их истинные позиции

$$\begin{aligned} x' &= x + xk(x^2 + y^2), \\ y' &= y + yk(x^2 + y^2). \end{aligned} \quad (2)$$

Коррекция яркости текущего пиксела  $I_c$  осуществляется в соответствии с соотношением

$$I_c = k_c I_c + (1 - k_c) \sum_{i=1}^4 k_i I_i, \quad (3)$$

где  $k_c$  — весовой коэффициент, определяющий степень коррекции яркости текущего пиксела;  $k_i$  — весовой коэффициент, определяющий степень влияния яркости смежного пиксела на яркость текущего;  $I_i$  — яркость смежного пиксела.

С использованием представленной выше математической модели (1)—(3) разработан модуль МД1 цифровой коррекции дисторсии изображения [5]. Схема модуля представлена на рис. 1.

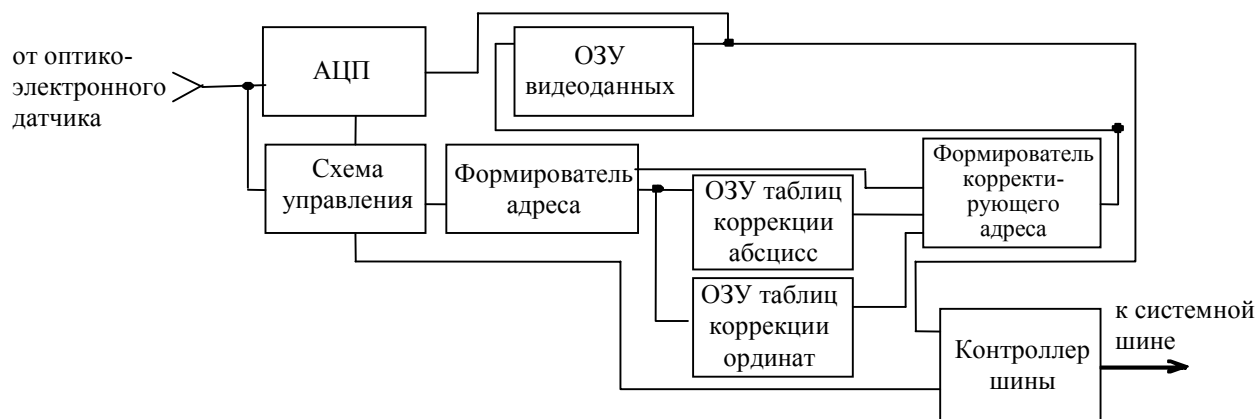


Рис. 1

С помощью схемы управления осуществляются синхронизация работы АЦП, формирование импульсных сигналов, соответствующих поступлению очередного пиксела, очередных строки и кадра. Схема управления работой ОЗУ видеоданных включает его синхронизацию при записи данных от АЦП и их передаче от компьютера. ОЗУ видеоданных предназначено для хранения кадров изображения.

Формирователь адреса служит для преобразования адреса очередного поступающего пиксела в формате („номер строки“, „номер столбца“) по адресу, определяемому переменной, соответствующей линейному адресному пространству ОЗУ видеоданных.

ОЗУ таблиц коррекции абсцисс и ординат предназначены для хранения корректирующих таблиц, позволяющих определять смещение пиксела в горизонтальном и вертикальном направлении в зависимости от координат очередного пиксела.

Формирователь корректирующего адреса обеспечивает определение адреса очередного пиксела с учетом его текущих координат и смещений. Контроллер шины предназначен для управления обменом данными.

В ходе анализа работоспособности модуля был выявлен недостаток: невысокая точность ввода изображения. Для устранения указанного недостатка разработан усовершенствованный модуль цифровой коррекции дисторсии МД2 [6], схема которого представлена на рис. 2.

Модуль МД2 отличается от МД1 тем, что он корректирует не только геометрические, но и яркостные отклонения, что позволяет достичь большей точности формирования изображения.

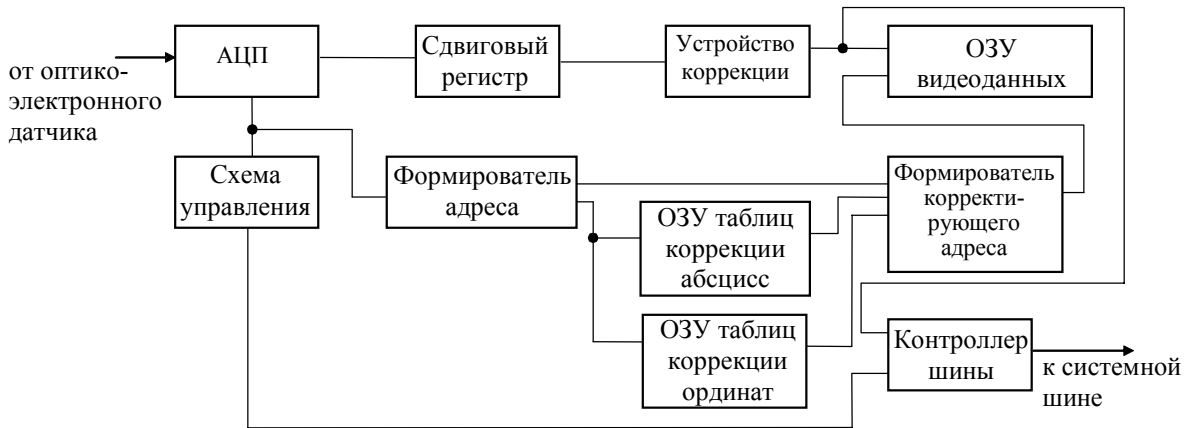


Рис. 2

На рис. 3 представлен алгоритм работы модуля коррекции МД2.



Рис. 3

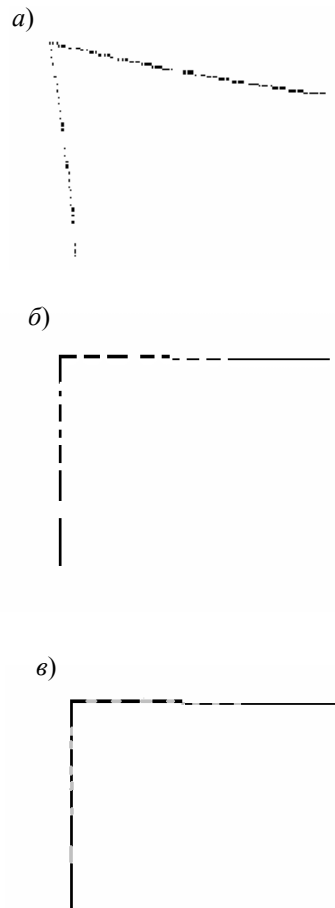


Рис. 4

Для экспериментальной проверки модуля была разработана программная модель. Исходными данными служит изображение, искаженное влиянием дисторсии (рис. 4, а). На рис. 4, б и в представлены изображения с откорректированными геометрическими и яркостными

отклонениями соответственно. Как видно из рис. 4, б, объект скорректирован геометрически, появились пробелы. Затем была выполнена яркостная коррекция (см. полутона на рис. 4, в), что позволило улучшить качество изображения.

Разработанный модуль цифровой коррекции дисторсии МД2 превосходит аналоги, что позволяет достичь большей точности ввода изображения, а также повысить его качество. Разработанные устройства могут найти применение при создании оптико-электронных устройств различного назначения, включая бытовую (фотоаппараты, видеокамеры) и медицинскую технику.

Работа выполнена при поддержке фонда содействия развитию малых форм предприятий в научно-технической сфере „У.М.Н.И.К.“ (договор № 0806, 2008 г., проект № 8555).

#### СПИСОК ЛИТЕРАТУРЫ

1. *Прэнтт У.* Цифровая обработка изображений / Пер. с англ. М.: Мир, 1982. Кн. 1. 312 с.; Кн. 2. 493 с.
2. *Tsai R. Y.* A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses // IEEE Trans. Rob. Autom. RA-3 (4). 1987. P. 323—344.
3. Методы компьютерной обработки изображений / Под ред. *В. А. Сойфера*. М.: Физмалит, 2001. 784 с.
4. *Titov D. V., Shirabakina T. A.* The Mathematical Model and the Device of Rise in a Precision of Image Forming // „Information and Telecommunication Technologies in Intelligent Systems“. Proc. 6th Int. Conf. Greece, 2008.
5. Пат. № 2295153 РФ, МПК<sup>7</sup> G 06 K 9/32. Корректирующее устройство ввода изображения в ЭВМ / *М. И. Труфанов, Д. В. Титов*. Оpubл. 10.03.2007. Б.И. № 7. 8 с.
6. *Titov D. V., Shirabakina T. A.* Correction Device of Distortion // „Information and Telecommunication Technologies in Intelligent Systems“. Proc. 5th Int. Conf. Spain, 2007. P. 122—124.

#### *Сведения об авторах*

**Дмитрий Витальевич Титов**

— магистрант; Курский государственный технический университет, кафедра вычислительной техники; E-mail: amazing2004@inbox.ru

**Тамара Александровна Ширабакина**

— канд. техн. наук, профессор; Курский государственный технический университет, кафедра конструирования и технологии ЭВС; E-mail: tas\_06@mail.ru

Рекомендована кафедрой  
вычислительной техники

Поступила в редакцию  
12.09.08 г.

Ю. С. БЕХТИН, В. Н. СОЛЯКОВ, Д. В. ТИТОВ

## СОПРЯЖЕНИЕ МНОГОЭЛЕМЕНТНОГО ФОТОПРИЕМНОГО УСТРОЙСТВА С ПЕРСОНАЛЬНЫМ КОМПЬЮТЕРОМ ИЗМЕРИТЕЛЬНОГО СТЕНДА

Рассматриваются варианты сопряжения многоэлементных фотоприемных устройств (МФПУ) с персональным компьютером измерительного стенда, выполненного на базе аналоговых и цифровых модулей ввода—вывода. Представлены структурные схемы и разработано программное обеспечение стенда, позволяющие эффективно измерять параметры МФПУ в реальном масштабе времени и визуализировать тепловые изображения.

**Ключевые слова:** многоэлементные фотоприемные устройства, персональный компьютер, измерительный стенд, модули ввода—вывода, сопряжение, тестирование, геометрический шум.

**Введение.** В настоящее время многоэлементные матричные фотоприемные устройства (МФПУ) находят широкое применение в системах технического зрения, охранно-пожарной сигнализации, видеокomпьютерных системах и т.п. Выпускаемые отечественной промышленностью МФПУ проходят тестирование и калибровку на измерительных стендах (рис. 1).



Рис. 1

Современные МФПУ представляют собой сложные изделия, которые содержат большое число (до  $10^6$ ) фоточувствительных элементов, а также схему считывания и предварительной обработки сигналов в фокальной плоскости. Выходной сигнал формируется в процессе последовательного опроса сигналов всех фоточувствительных элементов встроенным электронным коммутатором. Выходные сигналы МФПУ всегда содержат так называемый геометрический шум (*fixed pattern noise* [1]), в частности, геометрический шум чувствительности (вызванный неоднородностью коэффициентов передачи), темнового тока (вследствие неоднородности постоянной составляющей в выходном сигнале) и дефектов (из-за нерабочих элементов).

Для проверки изделия и оценки годности на стенде измеряют разброс чувствительности его фотоэлементов, записывают профиль их пьедестальных напряжений и определяют дефектные элементы. Для экспресс-оценки работоспособности МФПУ необходимо осуществить визуализацию тепловых изображений, регистрируемых им. С целью автоматизации измерений в последнее время в состав стенда включают персональный компьютер (ПК), в связи с чем воз-

никает задача сопряжения МФПУ с ПК, требующая разработки аппаратного и алгоритмического (программного) обеспечения, функционирующего в реальном масштабе времени.

**Функциональные схемы подключения МФПУ к ПК.** Технически решить данную задачу возможно несколькими способами. Относительно дорогостоящим до сих пор считается применение внешних по отношению к ПК устройств, содержащих мощные процессоры обработки сигналов (DSP), буферную память для записи кадров [2]. Другой подход заключается в применении плат сопряжения на ПЛИС [3], при этом вероятны существенные временные задержки при перепрограммировании ПЛИС в случае модификации методов измерения параметров МФПУ. Одним из относительно простых и недорогих вариантов [4] можно считать применение в качестве устройства сопряжения плат аналого-цифрового или цифрового ввода—вывода, устанавливаемых в ПК, которые не содержат встроенного DSP. Следовательно, алгоритмы обработки выходного сигнала МФПУ полностью реализуются через программное обеспечение в ПК, которое и управляет работой всей цепи „МФПУ—плата сопряжения—ПК“.

В настоящей статье рассматривается пример подключения МФПУ (форматом 240×320 фоточувствительных элементов) к ПК измерительного стенда с помощью базового модуля ввода—вывода AMBPCI и submodule семейства ADM фирмы „Инструментальные системы“ (Москва).

Если МФПУ содержит встроенный АЦП, то с помощью submodule ADM-DIO32 обеспечиваются ввод и отображение выходных оцифрованных отсчетов кадра МФПУ в реальном масштабе времени (рис. 2). Однако вследствие ограниченной разрядности АЦП МФПУ (8—10 разрядов) необходимо использовать дополнительные ЦАП и память (ОЗУ) для компенсации пьедестальных напряжений фотоэлементов, которые перекрывают диапазон АЦП. Наличие обратной связи через дополнительное устройство (ЦАП) делает управление стендом относительно неустойчивым. Очевидно, аппаратные затраты сокращаются при наращивании разрядности АЦП изделий.

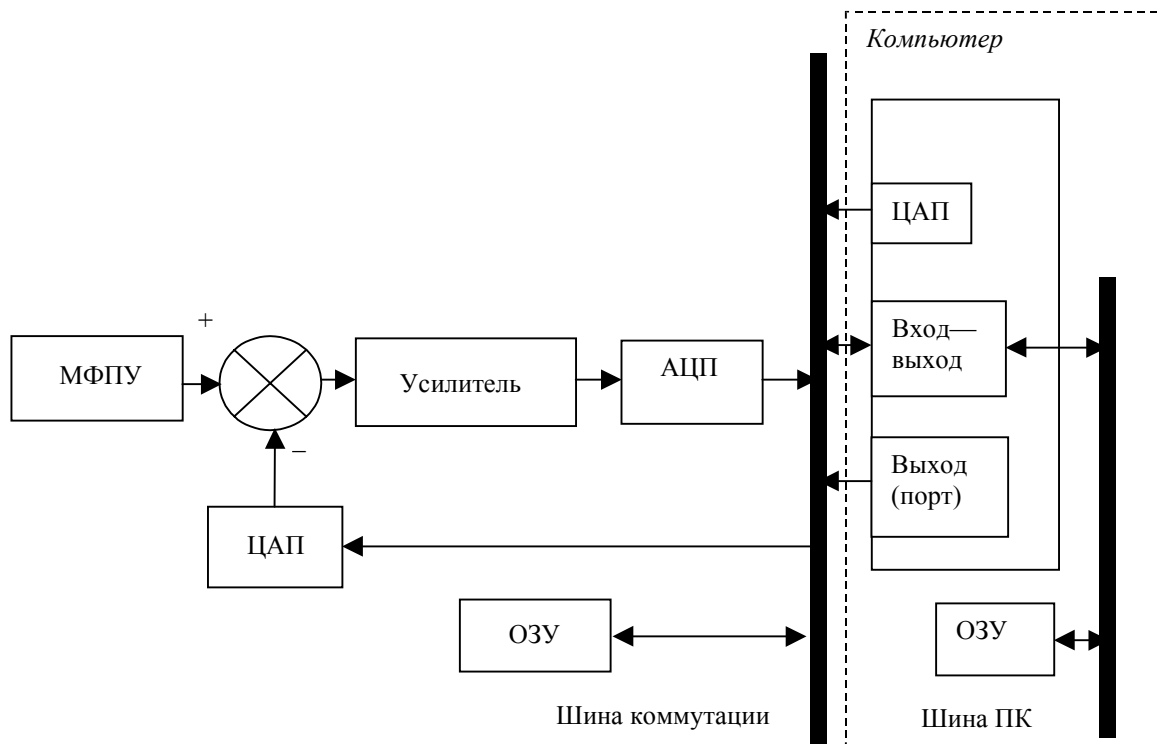


Рис. 2

Если МФПУ не содержит встроенного АЦП, то вместо submodule цифрового ввода ADM-DIO32 можно использовать submodule аналогового ввода ADM214x10M с двумя АЦП.

Большая разрядность АЦП субмодуля (14 разрядов) позволяет сразу вводить оцифрованные значения выходного дискретно-аналогового сигнала МФПУ через базовый модуль в оперативную память компьютера, откуда они извлекаются процессором (ЦПУ) ПК для обработки. Обработанные данные через видеоадаптер выводятся на дисплей и могут быть сохранены на жестком диске (рис. 3).

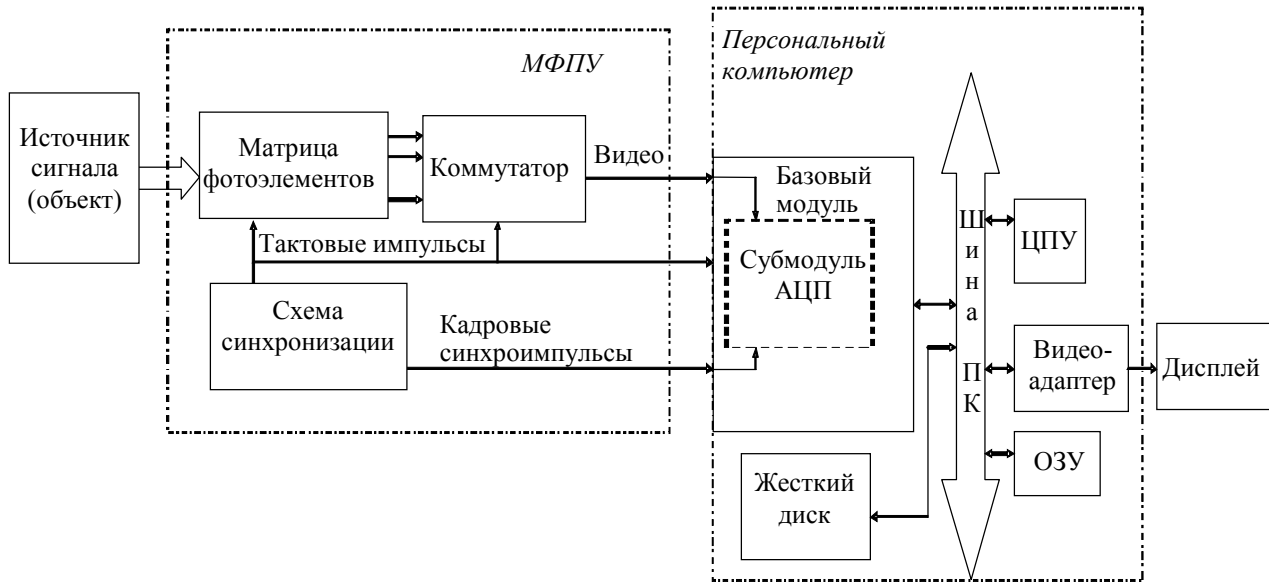


Рис. 3

Схема синхронизации МФПУ вырабатывает тактовые синхроимпульсы, по передним фронтам которых встроенный электронный коммутатор мультиплексирует электрические сигналы фотоэлементов на выход, а также кадровые синхроимпульсы, формируемые в начале каждого кадра. Оцифрованные данные поступают в базовый модуль, далее через буферную память FIFO на шину PCI ПК, где осуществляется преобразование 16-разрядных данных в 32-разрядные. Вводимые отсчеты АЦП занимают старшие 14 разрядов и представляются в дополнительном коде. В базовом модуле устанавливаются режим стартовой синхронизации АЦП и режим тактирования АЦП.

Тактовая частота персонального компьютера должна быть не ниже 2,4 ГГц с объемом ОЗУ не менее 256 Кб.

**Программное обеспечение.** Для программирования модуля и субмодуля используется драйвер DAQDRV. Он представляет собой набор динамических библиотек и виртуальных драйверов, которые должны находиться либо в текущем рабочем каталоге (из последнего производится запуск приложения, использующего драйвер), либо в системном каталоге Windows (daqdrv32.dll, adm214x.dll, ambpci32.dll, vambpcid.vxd, v5933d.vxd). С помощью драйвера осуществляется первоначальная настройка режима работы блока.

Программное обеспечение для операционной системы Windows 98, XP, 2000, выполняющее сопряжение ПК с AMBPCI, создано в среде разработки Visual C++ 6.0 с применением библиотеки классов MFC. В приложении реализованы однодокументный интерфейс (SDI), а также возможность непрерывного ввода изображений (так называемый видеорежим). Для разделения данных и их графического представления выбрана архитектура программы „Документ—представление“, так как она облегчает сопровождение программы, т.е. ее последующую модификацию. Поскольку исходные данные из АЦП поступают в память ПК по каналу DMA (т.е. без участия процессора ПК), а получение изображения требует значительных вычислительных затрат, то целесообразно разделить процессы ввода и обработки информации и организовать их в виде двух потоков. В результате в буфере памяти постоянно находятся данные, соответствующие текущему мо-



менту времени, а их обработка выполняется с максимально возможной скоростью, которая зависит от типа ПК. Такое решение позволяет улучшить визуальное представление обрабатываемых данных, отображаемых в режиме „видео“ на компьютерах с относительно малым быстродействием.

Для ввода данных из базового модуля и их обработки было создано программное обеспечение на языке высокого уровня C++, позволяющее осуществлять калибровку системы и обеспечивать корректную обработку данных с целью получения изображения высокого качества. Такое обеспечение позволяет не учитывать тип компьютера и операционной системы, а также, в случае изменения алгоритмов, необязательно изменять остальную часть кода приложения. Для вычислений в процессе калибровки МФПУ используются рекуррентные алгоритмы, что значительно сокращает расходы оперативной памяти персонального компьютера.

Работа стенда начинается с инициализации базового модуля АМВРСІ, при этом настраиваются режимы старта/останова АЦП. С помощью программы осуществляется выделение областей памяти, необходимой для хранения информации, поступающей с матрицы или линеек фотоэлементов, и преобразованного (откорректированного) изображения. Далее происходит заполнение калибровочных массивов. На этапе настройки (тестирования) МФПУ калибровка осуществляется вручную. Оператор переводит стенд в ручной режим калибровки и дает команду на выполнение первой калибровки при однородном низком уровне освещения фоточувствительных элементов МФПУ (например, при закрытии объектива холодной заслонкой). При калибровке вычисляются коэффициенты, необходимые для компенсации пьедестальных напряжений фотоэлементов, и выявляются дефектные элементы. Вторая калибровка происходит аналогичным образом при увеличенном уровне однородной засветки (например, при размещении перед объективом нагретой поверхности). При этом вычисляются коэффициенты передачи каналов МФПУ (фотоэлемент + электрические цепи), отражающие качество изделия и предназначенные для компенсации неоднородности чувствительности. После получения калибровочных коэффициентов программа готова к формированию тепловых изображений, регистрируемых МФПУ.

В программе предусмотрен интерфейс пользователя, предназначенный для измерения и отображения параметров МФПУ, а также для изменения параметров модуля и submodule непосредственно из программы без ее recompilation.

#### СПИСОК ЛИТЕРАТУРЫ

1. *Кругликов С. В.* Методы и средства подавления структурных помех многоэлементных фотоприемников // Аналитический обзор № 4628 за 1970—1987 гг. М., 1989.
2. *Алексеев Ю. В., Арзамасов Д. И., Вахитов М. А., Нигматуллина Р. Р., Сунцов В. В., Хисамов Р. Ш.* Стенд контроля параметров матричного ФПУ тепlopеленгационного устройства // Тез. докл. XIX МНТК по фотоэлектронике и приборам ночного видения. М.: ФГУП НПО „Орион“, 2006. С. 207.
3. *Бодров В. Н., Рыков А. Н.* Цифровой модуль многоканальной обработки сигналов изображений различных спектральных диапазонов // Там же. С.131.
4. *Бехтин Ю. С., Баранцев А.А., Соляков В. Н., Медведев А. С.* Аппаратно-программный комплекс цифровой обработки сигналов многорядных матричных фотоприемных устройств // Прикладная физика. 2007. № 3. С. 77—82.

#### *Сведения об авторах*

**Юрий Станиславович Бехтин**

— канд. техн. наук; Рязанский государственный радиотехнический университет, кафедра автоматизации и информационных технологий в управлении; E-mail: aitu@rgrtu.ryazan.ru

- Владимир Николаевич Соляков* — д-р техн. наук; НТЦ НПО „Орион“, Москва, кафедра физической электроники МФТИ; E-mail: [solyakov@mail.ru](mailto:solyakov@mail.ru)
- Дмитрий Витальевич Титов* — магистрант; Курский государственный технический университет, кафедра вычислительной техники; E-mail: [amazing2004@inbox.ru](mailto:amazing2004@inbox.ru)

Рекомендована кафедрой  
вычислительной техники

Поступила в редакцию  
12.09.08 г.

## SUMMARY

P. 7—11.

### METHOD OF AUTOMATIC FORMING OF PANORAMIC IMAGES BASED ON FUZZY LOGIC

The method of receiving of panoramic images from some set of digital images with identical areas, with use of a mathematical apparatus of the fuzzy logic.

**Keywords:** panoramic image, image processing, radial distortion, fuzzy logic.

#### *Data on authors*

- Vladimir N. Gridin* — Dr. Techn. Sci., Professor; Design Information Technologies Center of RAS; Director  
*Elena I. Bugaenko* — Post-Graduate Student; Kursk State Technical University, Department Computational Sciences; E-mail: Elena.bugaenko@gmail.com

P. 11—16.

### METHOD OF PROCESSING OF THE BAND IMAGES ON DEFECTOSCOPY SCENES

In article the approach to the formal description of the data on spatially-distributed objects municipal geoinformation systems is considered. For this purpose it is offered to characterize spatial position of object in three-dimensional space on the basis of seven topological relations which allow to generate structure from sixteen binary elements. On the basis of these formalizations there is an opportunity to describe topological position of object as the structures, allowing classifying spatial object.

**Keywords:** spatially-distributed object, the topological relation, municipal geoinformation systems, matrix of sixteen intersections.

#### *Data on authors*

- Sultan S. Sadykov* — Dr. Techn. Sci., Professor; Murom Institute (Branch) of Vladimir State University, Department Information Systems  
*Aleksey A. Orlov* — Cand. Techn. Sci.; Murom Institute (Branch) of Vladimir State University, Department Information Systems; E-mail: AlexeyAlexOrlov@rambler.ru  
*Aleksander A. Ermakov* — Post-Graduate Student; Murom Institute (Branch) of Vladimir State University, Department Information Systems

## P. 16—19.

**IMAGE RECOGNITION METHOD BASED ON CHAOTIC TRANSFORMATION OF PATTERNS AND IMAGES**

The paper describes image recognition method which performs pattern and image transformation based on discrete chaotic mapping in order to select informative subimages for image restoration and recognition.

**Keywords:** images recognition, chaotic, mapping, pattern, object, transformation, taxons restoration, failure, matrix.

*Data on authors*

- Victoria V. Gordienko* — Cand. Techn. Sci.; Kursk State Technical University, Department Criminal Law  
*Victor M. Dovgal* — Dr. Techn. Sci., Professor; Kursk State Technical University, Software Development for Computer Engineering and Computer-Based Systems; E-mail: vmdovgal@yandex.ru  
*Roman A. Puzyna* — Post-Graduate Student; Kursk State Technical University, Software Development for Computer Engineering and Computer-Based Systems

## P. 20—24.

**DISTRIBUTION OF FLOW OF GRAPHIK INFORMATION IN SYSTEMS OF IMAGES RECOGNITION IN REAL TIME**

Processing of the big files of the graphic information in real time, for example, a video stream, demands its processing in the distributed streams and their subsequent assemblage for recognition of required objects. In papers algorithms for parallel work and managements of such process are considered.

**Keywords:** control algorithm, graphic information, object recognition, real-time scale.

*Data on authors*

- Ivan M. Gostev* — Dr. Techn. Sci.; Moscow State Institute of Electronics and Mathematics (Technical University), Department Cybernetics; E-mail: igostev@gmail.com  
*Aleksandr G. Podgorbunsky* — Post-Graduate Student; Moscow State Institute of Electronics and Mathematics (Technical University), Department Cybernetics; E-mail: shurikp@gmail.com

## P. 25—30.

**FORMAL REPRESENTATION OF CHARACTERISTICS OF SPATIALLY-DISTRIBUTED OBJECTS**

In this article the approach to the formal description of the data on spatially-distributed objects municipal geoinformation systems is considered. On the basis of these formalizations there is an opportunity to describe the topological position of object as structures, allowing classifying spatial object.

**Keywords:** spatially-distributed object, topological relation, municipal geoinformation systems, matrix of sixteen intersections.

*Data on authors*

- Dmitry Y. Andrianov* — Cand. Techn. Sci.; Murom Institute (Branch) of Vladimir State University, Department Information Systems; E-mail: AndrianovDE@inbox.ru  
*Sultan S. Sadykov* — Dr. Techn. Sci., Professor; Murom Institute (Branch) of Vladimir State University, Department Information Systems; E-mail: SadykovSS@yandex.ru  
*Vladislav V. Frolov* — Post-Graduate Student; Murom Institute (Branch) of Vladimir State University, Department Information Systems; E-mail: AndrianovDE@inbox.ru

P. 31—36.

### INVARIANT RECOGNITION OF FINGERPRINTS USING KEY POINTS ALGORITHM

Speed of invariant to rotating, moving and fragmentation fingerprint identification increasing method is written. Reliable of identification on low quality digitalized fingerprint images are increased.

**Keywords:** fingerprint identification, invariant, digitalization, centering.

#### *Data on authors*

- Anton V. Ognev** — Student; Kursk State Technical University, Department Computational Sciences; E-mail: light@kursknet.ru
- Aleksander P. Tipikin** — Dr. Techn. Sci., Professor; Kursk State Technical University, Department Computational Sciences; E-mail: cct@cafct.kstu.kursk.ru

P. 37—45.

### CHECKING UP ISOMORPHIC INCLUSIONS OF *R*-EXPRESSIONS IN THE CONSTRUCTION OF A SET OF SECTIONS FOR PARALLEL LOGIC CONTROL ALGORITHMS

In the paper a hardware-level algorithm is proposed for checking up isomorphism (*r*-isomorphism) of *R*-expression trees (sections of parallel algorithm) based upon a number of *R*-expression tree features. A description for the corresponding hardware (accelerator) is presented to check up isomorphism of *R*-expressions in linear time.

**Keywords:** logic multicontroller, synthesis, parallel logic control algorithm, separation, optimization, oriented trees.

#### *Data on authors*

- Eduard I. Vatutin** — Post-Graduate Student; Kursk State Technical University, Department Computational Sciences; E-mail: evatutin@rambler.ru
- Igor V. Zotov** — Dr. Techn. Sci., Professor; Kursk State Technical University, Department Computational Sciences; E-mail: zotovigor@yandex.ru
- Vitaly S. Titov** — Dr. Techn. Sci., Professor; Kursk State Technical University, Department Computational Sciences; Managing Department; E-mail: titov@vt.kstu.kursk.ru

P. 45—51.

### PARALLEL-PIPELINE DEVICE FOR EARTH SURFACES AEROSPACE IMAGES VECTORIZATION

Structure char and functioning algorithms of a device designed for buildings thematic layer automatic creation based on aerospace images for digital maps of geographic information systems are considered on the presented paper. Dedicated parallel graphic units and pipeline functioning algorithm provide greater computational units capacity and higher performance.

**Keywords:** geoinformation systems, aerospace images, automatic vectorization, buildings layer, edge detection, parallel-pipeline device.

#### *Data on authors*

- Sergey Yu. Miroshnichenko** — Post-Graduate Student; Kursk State Technical University, Department Computational Sciences; E-mail: oldguy7@rambler.ru
- Vitaly S. Titov** — Dr. Techn. Sci., Professor; Kursk State Technical University, Department Computational Sciences; Managing Department; E-mail: titov@vt.kstu.kursk.ru

P. 52—55.

### RECONFIGURATION METHOD OF OPERATIONAL PART OF MULTIPROCESSOR FOR STRUCTURAL RECOGNITION OF PATTERNS

The method of recognition of structural descriptors for branching processes and the organization of connections between production devices in the multiprocessor are described.

**Keywords:** structural recognition, multiprocessor, branching constructive process, production.

#### *Data on author*

*Evgeny A. Titenko* — Cand. Techn. Sci.; Kursk State Technical University, Department Software of Computer Facilities; E-mail: johntit@mail.ru

P. 55—59.

### PARALLEL SUBPROGRAMS ASSIGNMENT IN MULTIMICROCONTROLLERS TAKING INTO ACCOUNT REJECTION

The planning assignment problem of the parallel subprograms in multimicrocontrollers is considered taking into account rejection of multimicrocontroller processor, is formulated a technique and algorithm of planning assignment of tasks in multimicrocontroller fault-tolerance.

**Keywords:** allocation, multimicrocontroller, rejection, reconfiguration, parallel program, processor.

#### *Data on author*

*Dmitry B. Borzov* — Cand. Techn. Sci.; Kursk State Technical University, Department Computational Sciences; E-mail: borzovdb@kursknet.ru

P. 60—65.

### CONFLATION AND SPLITTING METHOD OF CYCLIC PARTS OF SEQUENTIAL INHERIT PROGRAMMS

In the paper, a method of conflation and splitting of cyclic parts of sequential inherit programmes is proposed. The method permits to unite cycle bodies to get longest linear parts for consequent code parallelization. Possibility analysis of the equivalent transformation of the fragment program to type, allowing execute some parallel cycles on determined space of iterations, to reduce number an operation incrementations counter of these cycles and checking the conditions of the output of them.

**Keywords:** multisequencing, program, merging, splitting, cycle, line, site, method.

#### *Data on authors*

*Dmitry B. Borzov* — Cand. Techn. Sci.; Kursk State Technical University, Department Computational Sciences; E-mail: borzovdb@kursknet.ru

*Sergey A. Dyubryuks* — Post-Graduate Student; Kursk State Technical University, Department Computational Sciences; E-mail: serhio5551@yandex.ru

*Vitaly S. Titov* — Dr. Techn. Sci., Professor; Kursk State Technical University, Department Computational Sciences; Managing Department; E-mail: titov@vt.kstu.kursk.ru

P. 66—69.

### RECOGNITION OPTO-ELECTRONIC DEVICE FOR MEDICAL FAST DIAGNOSTICS

Image recognition devices for fast medical diagnostics are proposed. Devices are intended for diagnostics a deep vein thrombosis and pulmonary embolism predict, image enhancement for endoscope analysis, binocular vision and squint diagnostics.

**Keywords:** opto-electronic device, medical image processing, fast diagnostics, recognition.

#### *Data on author*

**Maxim I. Truphanov** — Cand. Techn. Sci.; Kursk State Technical University, Department Computational Sciences; E-mail: tmi@pub.sovtest.ru, temp1202@mail.ru

P. 70—74.

### MATHEMATICAL MODEL OF ENDOSCOPIC IMAGES RESTORATION

The mathematical model of the endoscopic images restoration, which helps to increase the quality of endoscopic image's forming, is presented.

**Keywords:** endoscopic image, image processing, catchlights, Fourier transform, spectrum analysis.

#### *Data on authors*

**Aleksandra N. Strelkova** — Post-Graduate Student; Kursk State Technical University, Department Computational Sciences; E-mail: strelkova@bk.ru

**Maxim I. Truphanov** — Cand. Techn. Sci.; Kursk State Technical University, Department Computational Sciences; E-mail: temp1202@mail.ru

**Aleksander A. Stepchenko** — Cand. Med. Sci.; Kursk State Medical University, Department Internal Diseases

P. 74—77.

### CORRECTION DEVICE OF IMAGE DISTORTION

The mathematical model of digital correction of distortion on the image, which helps to increase the quality of image forming, is created. The correction device of distortion on the image is developed.

**Keywords:** digital image, pixel, calibration, opto-electronic device, distortion, brightness.

#### *Data on authors*

**Dmitry V. Titov** — Kursk State Technical University, Department Computational Sciences; E-mail: amazing2004@inbox.ru

**Tamara A. Shirabakina** — Cand. Techn. Sci., Professor; Kursk State Technical University, Department Computer Construction and Technologies; E-mail: tas\_06@mail.ru

P. 78—82.

### CONNECTING THE MULTIELEMENT PHOTORECEIVERS TO THE TESTING TABLE BASED ON COMPUTER

Different schemes for connecting the multielement infrared sensors to the testing table based on IBM PC are considered. The hard- and software design of the testing table is described, too. It provides with the sensor parameters to be good estimated in real-time scale and the visual representation of infrared images.

**Keywords:** multielement infrared sensor, computer, testing table, input-output devices, connecting, testing, fixed pattern noise.

#### *Data on authors*

- Yury S. Bekhtin* — Cand. Techn. Sci.; Ryazan State Radio University, Department Automatics and Control Information Technologies; E-mail: [aitu@rgrtu.ryazan.ru](mailto:aitu@rgrtu.ryazan.ru)
- Vladimir N. Solyakov* — Dr. Techn. Sci., Professor; Head of Scientific Center R&D Enterprise “Orion”, Moscow Institute of Physics and Technology, Department Physical Electronics; E-mail: [solyakov@mail.ru](mailto:solyakov@mail.ru)
- Dmitry V. Titov* — Kursk State Technical University, Department Computational Sciences; E-mail: [amazing2004@inbox.ru](mailto:amazing2004@inbox.ru)