

УДК 004

Б.А. Залесский

КОМБИНАТОРНЫЙ АЛГОРИТМ ВЫДЕЛЕНИЯ КОНТУРОВ ОБЪЕКТОВ НА ЦИФРОВЫХ ИЗОБРАЖЕНИЯХ

Рассматривается алгоритм выделения контуров на изображениях, который основывается на использовании комбинаторных методов, применяемых для кластеризации ориентированного градиента. Алгоритм позволяет оценить с достаточно высокой точностью положение и углы наклона контуров, а также получить удобное для анализа формы объектов векторное представление контуров ломаными или гладкими кривыми.

Введение

Задача выделения контуров хорошо известна в обработке изображений. Контур изображения достаточно широко используется для интерактивного и автоматического выделения объектов, улучшения качества изображений, решения различных задач распознавания, регистрации изменений, компьютерного зрения и т. д.

В данной работе контур определяется как связанное в заданной системе окрестностей множество пикселей изображения, в которых перепад яркости больше установленного порога [1]. Граница связанной области понимается как замкнутый контур.

В настоящее время существует множество различных подходов к выделению контуров. Наиболее ранние – оконные, появившиеся в 1980-е гг., основанные на использовании дифференциальных операторов, активно применяются до сих пор. Обычно авторы делят их на две группы в зависимости от типа используемого дифференциального оператора. К первой группе относятся алгоритмы, основанные на поиске больших значений откликов оконных операторов, вычисляющих первые разностные производные. Среди них алгоритмы Робертса, Собеля, Превитта, Кирша, Кани и др. [1, 2]. Алгоритмы второй группы для нахождения контуров используют различные разностные аналоги вторых производных в виде различных версий оператора Лапласа (включая LOG, DOG и т. д.) [1, 2]. Более поздние алгоритмы выделяют контуры с помощью преобразования Хафа, методов математической морфологии, теории графов, активных контуров, вариационных методов [1, 3]. Достаточно подробное описание известных алгоритмов и обширную библиографию по данной тематике можно найти в [4].

Большинство известных алгоритмов находят контуры в растровом виде, удобном для решения широкого круга задач обработки и распознавания изображений. Однако для некоторых задач, например связанных с анализом и распознаванием формы объектов или описанием чертежей и планов, проще использовать векторное представление контуров. Такое представление несложно получить в случае, когда на изображении присутствуют только контуры, состоящие из отрезков прямых или кусочно-гладких кривых, форма которых заранее известна. Если же форма контуров заранее неизвестна, построение их векторного представления превращается в трудоемкую задачу.

В качестве примера рассмотрим задачу выделения зданий, изображенных на аэрофотоснимке городского ландшафта (рис. 1, а), на основе анализа формы крыш. Для ее решения используются контуры изображения (рис. 1, б), на которых нужно найти участки, близкие по форме к отрезкам прямых линий, общие вершины таких участков и углы между линиями. В данном случае применение известных методов, большинство из которых использует растровое представление контуров, сопряжено с трудностями, вызванными целым рядом причин. Во-первых, на аэрофотоснимках в подавляющем числе случаев контуры не содержат идеально прямолинейных участков (тем более в присутствии растровых искажений), поэтому требуются операции, надежно выделяющие отрезки, близкие по форме к прямолинейным. Во-вторых, нередко в одном и том же месте сходятся несколько участков контуров, часть из которых порождена крышами, а другая – посторонними

объектами. В-третьих, часто края крыш порождают двойные или тройные контуры. Использование надежного векторного приближения контуров значительно облегчает решение задачи.

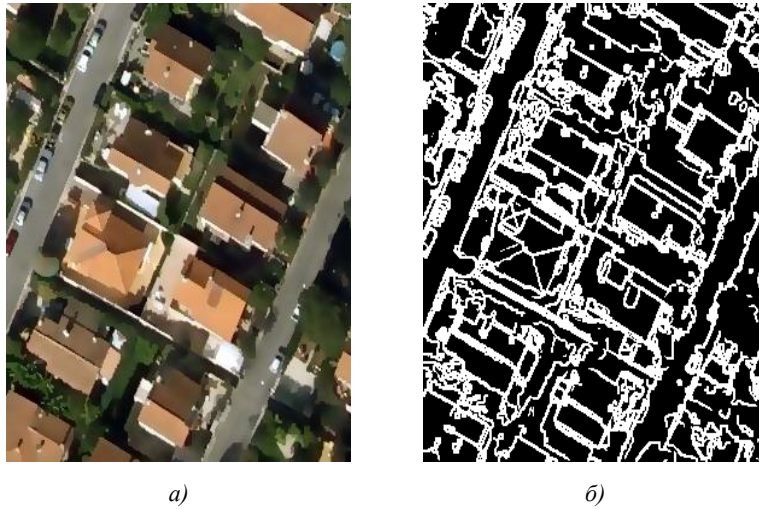


Рис. 1. Изображение городского ландшафта и его контуры: *а)* часть оригинального аэрокосмического снимка городского ландшафта; *б)* бинаризованный градиент снимка

сформулированной задачи [5–7]. Кластеры oG содержат пиксели, в которых значения углов наклона oG достаточно близки друг к другу, вследствие чего эти кластеры представляют собой сильно вытянутые множества, слабо искривляющиеся в пространстве (рис. 2). Их нетрудно аппроксимировать отрезками ломаных или кривых линий, которые можно использовать в качестве векторного представления контуров. При кажущейся простоте приведенного примера получить кластерное представление границы такой точности известными алгоритмами непросто.

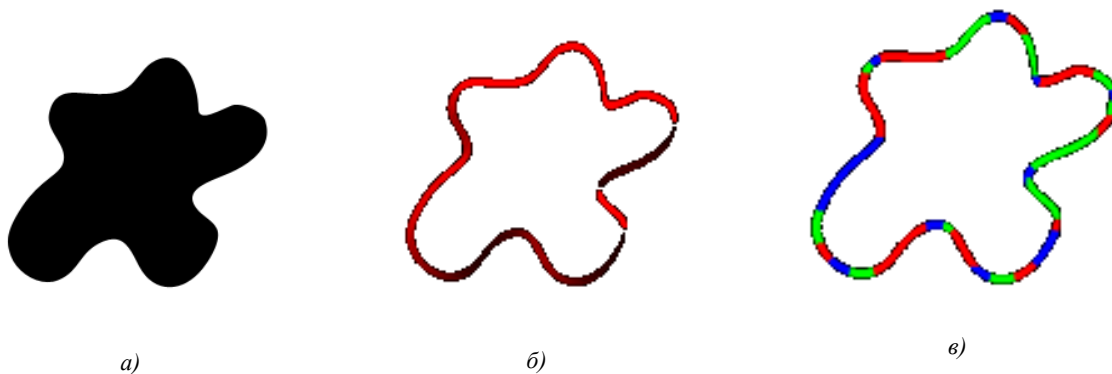


Рис. 2. Пример кластеризованного изображения oG : *а)* оригинальное модельное изображение; *б)* его ориентированный градиент. Яркостям красного цвета соответствуют углы наклона oG к оси OX , масштабированные к диапазону $[0, 255]$; *в)* кластеры oG , полученные после применения предложенного алгоритма, окрашенные для наглядности в три цвета

Алгоритм позволяет управлять кривизной получаемых кластеров. Полученные кластеры легко аппроксимируются отрезками ломаных или гладких кривых, с помощью которых и строится векторное представление контуров. Используя полученное векторное представление, можно выделять длинные участки контуров заданной гладкости и использовать полученные кривые для анализа формы объектов, их обнаружения и распознавания.

В предлагаемом алгоритме выделения контуров кластеризация ориентированного градиента oG используется для получения его достаточно точного векторного приближения на основе кластерного представления. Его особенность заключается в том, что полутоновое изображение oG , яркости которого соответствуют углам наклона вектора первой производной к оси OX , кластеризуется с помощью предложенного ранее автором статьи комбинаторного алгоритма, специально модифицированного для решения

1. Описание алгоритма кластеризации контура

Предложенный алгоритм предназначен для построения векторного приближения контуров изображения путем кластерного представления oG с помощью разработанных ранее комбинаторных алгоритмов кластеризации [5–7], приближения кластеров отрезками прямых или гладких кривых, объединения отрезков в кусочно-гладкие кривые заданной кривизны.

Кластером в рассматриваемом случае называется максимальное по размеру связное множество пикселей изображения, обладающих общим для всех них свойством (все соседние пиксели кластера этим свойством не обладают).

Реализация алгоритма сводится к последовательному выполнению следующих тематических блоков:

- 1) вычисления oG изображения;
- 2) получения кластерного представления ориентированного градиента;
- 3) приближения кластеров oG отрезками прямых или гладких кривых;
- 4) объединения отрезков прямых или гладких кривых, приближающих кластеры, в кусочно-гладкие кривые заданной кривизны.

Дадим краткое описание блоков. Для этого обозначим через S множество пикселей $j = (x_1, y_2)$ полутонового или цветного $n_1 \times n_2$ изображения I с яркостями I_j , $0 \leq I_j \leq 255$, или значениями цвета $I_j = (R_j, G_j, B_j)$, $0 \leq R_j, G_j, B_j \leq 255$.

Опишем кратко блок 1 для полутонового изображения I , так как в случае цветного изображения производятся аналогичные действия для каждого цветового канала либо оно сначала преобразуется в полутоновое. Для окна наперед заданного вида вычисляется оконный градиент G изображения I , представляющий собой $n_1 \times n_2$ -матрицу с элементами – векторами $G_j = (g_{гор,j}, g_{верт,j})$. Для тестов и вычислительных экспериментов использовались окна, состоящие из двух прямоугольников, симметрично расположенных относительно пикселей j , в которых определялся градиент. Числовые значения коэффициентов прямоугольников окна полагались +1 для одной половины окна и –1 для другой. Размеры одной половины окна выбирались от 1×1 до 11×11 пикселей (хотя теоретически здесь нет каких-либо ограничений).

Ориентированный градиент представляет собой $n_1 \times n_2$ -матрицу oG , элементы которой равны углам наклона вектора градиента G_j (измеренного с точностью 1°) к оси OX в случае, если длина вектора градиента $|G_j|$ больше наперед заданного порога $\tau > 0$, и равны какому-либо числу, например –1, в случае отсутствия градиента в пикселе j при $|G_j| \leq \tau$. Для ускорения вычисления oG может быть применен метод бегущей суммы [8] или интегральное изображение [9].

Описание блока 2 наиболее сложно с теоретической точки зрения. Детальное описание комбинаторного алгоритма кластеризации полутоновых изображений приведено в [5–7]. Суть алгоритма формулируется следующим образом: для целого $0 < k \leq 255$ фиксируется произвольный набор целых чисел $0 = m_0 < m_1 < \dots < m_k = 255$, в котором значения m_i выбираются исходя из поставленной задачи. Например, при сегментации они могут быть взяты на равном удалении от соседних значений, тогда $m_i = \left\lfloor \frac{255}{k} \right\rfloor i$ при i от 0 до $k-1$, или в соответствии с яркостной гистограммой I .

Рассматривается множество \mathcal{U} изображений U размера $n_1 \times n_2$, яркости которых U_j могут быть равными только m_i , $1 \leq i \leq k$. Тогда для произвольного полутонового изображения I его кластерное приближение $U^* = U^*(I)$ строится как решение оптимизационной задачи

$$U^* = U^*(I) = \arg \min_{U \in \mathcal{U}} \left(\sum_{j \in S} \lambda_j |I_j - U_j| + \sum_{\substack{j_1, j_2 \in S \\ j_1 \sim j_2}} \beta_{j_1, j_2} |U_{j_1} - U_{j_2}| \right), \quad (1)$$

где $\lambda_j, \beta_{j_1, j_2} \geq 0$, а выражение $j_1 \sim j_2$ обозначает соседство пикселей в наперед заданной системе окрестностей. Чаще всего в обработке изображений используются классические 4- или 8-точечные системы окрестностей, хотя теоретически они могут быть любыми, нужно только помнить, что от вида окрестности зависит время поиска решения U^* . Второе слагаемое в (1) играет роль

сглаживающей штрафной функции: меняя значения β_{j_1, j_2} , можно управлять размером и гладкостью границы кластеров на U^* .

В случае кластеризации самого изображения I выбором значения β_{j_1, j_2} в зависимости от величины вектора градиента $|G_j|$ можно добиться того, чтобы полученные кластеры не пересекались с контурами. В [5–7] получены комбинаторные решения задачи (1), позволяющие находить кластерное представление U^* даже для больших изображений I путем сведения задачи к решению k независимых задач поиска минимального разреза k сетей (под сетью понимается ориентированный граф с источником и стоком [10]), которые строятся по I . Там же подробно описано, как строятся сети.

Новизна настоящей работы заключается в том, что впервые предложенный метод кластеризации применяется для сегментации и сглаживания не самих изображений, а их контуров. Формально это означает, что в качестве исходного изображения рассматривается не само I , а матрица его ориентированного градиента oG . Кластерное представление $V^* = V^*(oG)$ ориентированного градиента принимает значения m_i из множества целых чисел $0 = m_0 < m_1 < \dots < m_k = 359$ (если шаг дискретизации угла наклона градиента выбран равным 1°) и вычисляется как решение оптимизационной задачи

$$V^* = V^*(oG) = \arg \min_V \left(\sum_{j \in S, oG_j \geq 0} \lambda_j |oG_j - V_j| + \sum_{\substack{j_1, j_2 \in S, oG \geq 0 \\ j_1 \sim j_2}} \beta_{j_1, j_2} |V_{j_1} - V_{j_2}| \right), \quad (2)$$

в которой, как и ранее, $\lambda_j, \beta_{j_1, j_2} \geq 0$, выражение $j_1 \sim j_2$ обозначает соседство пикселей в наперед заданной системе окрестностей, но суммирование берется только по пикселям, в которых присутствует градиент, т. е. в которых $oG \geq 0$. Решение задачи (2) с помощью предложенных методов комбинаторной оптимизации [5–7] строится аналогично решению задачи (1).

Следует заметить, что если для фиксированных λ_j увеличивать β_{j_1, j_2} , размеры кластеров возрастают, а их количество уменьшается. В пределе может получиться так, что каждый контур превратится в отдельный кластер. Однако это происходит при очень больших значениях β_{j_1, j_2} . Практические вычисления могут производиться в широком диапазоне значений параметров $\lambda_j, \beta_{j_1, j_2}$. Для упрощения процедуры выбора параметров можно положить все λ_j равными одному и тому же положительному числу, например $\lambda_j = 1$, тогда при четырехточечной системе окрестностей качественные с практической точки зрения результаты кластеризации могут быть получены при $0,25 \leq \beta_{j_1, j_2} \leq 20$, а при восьмиточечной – при $0,125 \leq \beta_{j_1, j_2} \leq 10$. Кривизной получаемых кластеров можно также управлять следующим образом: положить для некоторого числа $\rho \geq 1$

$$\beta_{j_1, j_2} = \begin{cases} \beta, & \text{если } j_1 \sim j_2, oG_{j_1} \geq 0, oG_{j_2} \geq 0 \text{ и } |oG_{j_1} - oG_{j_2}| \leq \rho; \\ 0 & \text{в противном случае.} \end{cases}$$

Это приведет к тому, что в кластер попадут только те соседние пиксели j_1 и j_2 , в которых присутствуют градиенты G_{j_1} и G_{j_2} , образующие угол, не превосходящий ρ . Простейший пример кластеризации контуров приведен на рис. 2. Если использовать описанные выше стратегии выбора параметров, полученные кластеры oG будут содержать пиксели, в которых значения углов наклона oG близки друг к другу, вследствие чего большинство кластеров будут представлять собой вытянутые множества, слабо искривляющиеся в пространстве. Исключение могут составить лишь кластеры, возникшие в местах сильного перегиба контуров. Такие кластеры имеют небольшой размер (см. рис. 2), но опять-таки незначительную кривизну.

Описание блока 3. Свойства полученных кластеров позволяют использовать самые разнообразные методы для приближения кластеров отрезками прямых или гладких кривых, начиная от простого вычисления диаметра кластера или осей инерции или применения метода наименьших квадратов до использования аппроксимирующих сплайнов и других современных методов. Нужно только не забывать о том, что у коротких кластеров, образовавшихся в местах

перегиба, диаметр или большая ось инерции могут оказаться перпендикулярными к направлению контура в данном месте.

Результаты тестирования и практического применения предложенного алгоритма показали, что следует избегать соблазна использовать для приближения кластера oG отрезком прямой оценку его приближенного угла наклона m_i , так как в некоторых случаях реальный угол наклона кластера может существенно отличаться от этого значения. Ошибка может быть вызвана округлением значений угла наклона градиента до некоторого m_i , а также растровыми искажениями и шумами, присутствующими на изображении, или неточностями, возникшими при вычислении оконного градиента.

При реализации алгоритма параметры векторного представления кривых, приближающих кластеры, можно хранить в структуре, описывающей их свойства: количество кластеров ориентированного градиента обычно ограничено несколькими тысячами.

Блок 4 алгоритма заключается в построении векторного представления контуров путем объединения кривых, приближающих кластеры в более сложные и длинные линии. Здесь можно действовать несколькими способами. По-видимому, самый простой из них состоит в следующем: определить критерий соседства кластеров, например считать соседними только кластеры, имеющие общую границу, либо только кластеры, расположенные на расстоянии меньшем некоторого фиксированного числа. Последний критерий имеет смысл потому, что встречаются случаи, когда между кластерами контура расположена маленькая область, не принадлежащая исходному контуру (бинаризованному градиенту), либо маленький кластер в местах сильного перегиба контура, состоящий из одного или нескольких пикселей.

Далее следует построить граф, вершины которого соответствуют кластерам представления V^* , а ребра – соседним кластерам. На основе полученного графа необходимо построить подграф, которой и будет задавать векторное представление контуров, оставляя в нем вершины и ребра, подходящие для решения поставленной задачи. Например, при решении некоторых задач может потребоваться учитывать только кластеры (и, следовательно, оставлять только вершины графа, соответствующие этим кластерам), площадь, размеры или форма приближающих кривых которых удовлетворяют определенным условиям. В качестве критерия оставления ребер может быть использован, например, допустимый угол между кривыми, приближающими соседние кластеры. Так, при поиске гладких участков контуров можно потребовать, чтобы углы между соседними приближающими кластеры кривыми были меньше некоторой заданной величины. При выделении же элементов крыш зданий можно потребовать, чтобы угол между приближающими кривыми менялся в заданном диапазоне, например был близок к 90 или 45° . Примеры применения предложенного алгоритма приведены в следующем разделе.

2. Исследование свойств алгоритма

Исследование свойств алгоритма проводилось на рисунках геометрических фигур, созданных в графических редакторах, на снимках автомобилей и других транспортных средств. Снимки транспортных средств были выбраны для тестов потому, что большая часть их контуров имеет гладкую правильную форму, разную у разных моделей. В качестве линий, приближающих кластеры контуров, выбирались отрезки прямых, направление которых совпадает с направлением большей оси эллипса инерции кластера. Отрезки объединялись в кусочно-ломанные линии, если угол между соседними отрезками был меньше заданного.

Были также реализованы несколько версий алгоритма наращивания контуров, основанных на поиске направлений наименьшего изменения направления градиента. Однако их применение дало существенно худшие результаты векторизации. Одна из причин низкого качества векторизации – искажения значений ориентированного градиента, присутствующие даже на изображениях высокого качества. Например, значения ориентированного градиента каждой стороны многоугольника могут отличаться по значениям на 20° . Вторая причина худшего качества векторного представления границы, полученного алгоритмом наращивания областей, вызвана ошибками, появляющимися в местах пересечения нескольких контуров.

На рис. 3 приведены результаты векторизации (предложенным комбинаторным алгоритмом) границы объекта, расположенного в левом верхнем углу, и его искаженной гауссовским шумом копии. В одну линию объединялись соседние отрезки, угол между которыми не превосходил 45° . Объединение отрезков начиналось с верхнего правого в обе стороны, поэтому граница оригинального объекта оказалась представлена одной линией, окрашенной в красный цвет. Граница зашумленного объекта распалась на несколько линий потому, что в кластерном представлении зашумленного объекта появились маленькие кластеры, которые были исключены из рассмотрения.

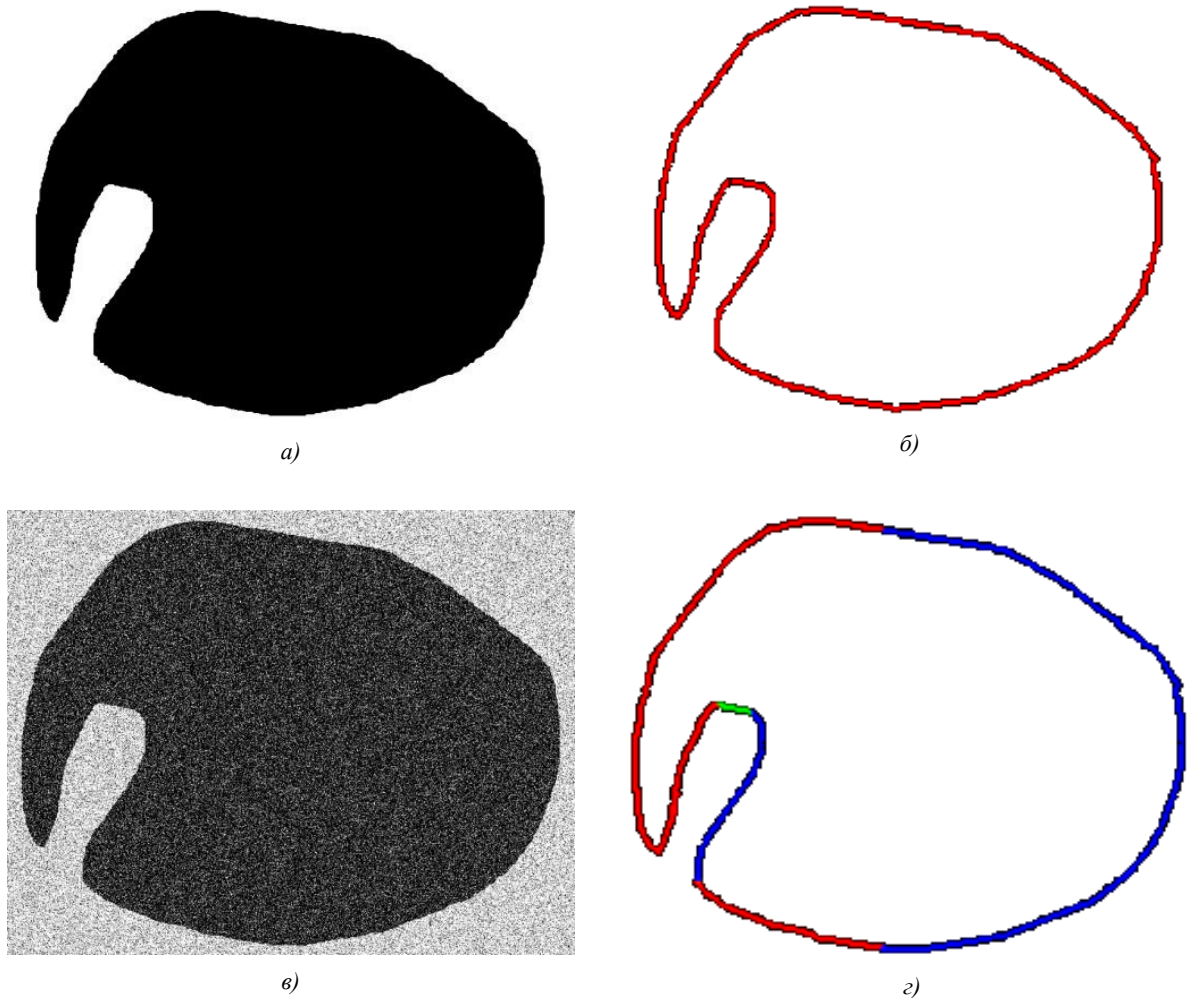


Рис. 3. Векторизованный градиент объекта: а) объект, созданный в графическом редакторе; б) векторное представление границы объекта с помощью отрезков прямых, направление которых задается большей осью эллипса инерции кластеров; в) тот же объект, искаженный гауссовским шумом; г) векторное представление границы искаженного шумом объекта с помощью отрезков прямых, направление которых задается большей осью эллипса инерции кластеров

Результаты векторного представления контуров автомобиля показаны на рис. 4. Приведены оригинальные изображения автомобилей размера 1600×1200 пикселей и векторных представлений их контуров, имеющих длину более 135 пикселей и представляющих собой кусочно-ломанные кривые, которые составлены из отрезков, образующих между собой угол меньше 90° .

Полученные векторные представления контуров обладают большей гладкостью по сравнению с контурами, полученными методами наращивания областей. Они могут быть использованы как для решения задачи обнаружения и распознавания объектов по форме их границ, так и для улучшения качества изображений. Собственно, алгоритм и разрабатывался для решения этих задач.



Рис. 4. Изображения автомобилей и векторные представления их контуров, полученные с помощью предложенного алгоритма

Заключение

В статье предложен новый подход к задаче выделения и векторизации контуров на изображениях. Его смысл заключается в использовании комбинаторных методов кластеризации, разработанных автором ранее [5–7], для кластеризации ориентированного градиента. Полученные кластеры ориентированного градиента приближаются отрезками гладких кривых, которые объединяются в более длинные кусочно-гладкие кривые в соответствии с выбранными критериями соседства отрезков и допустимыми углами между ними. Построенные таким способом векторные представления контуров обладают заданной гладкостью и достаточно высокой точностью. Они удобны для решения задач обнаружения и распознавания объектов по форме. В ближайшем будущем планируется разработать подходы к решению этой задачи на основе анализа векторного представления контуров.

Список литературы

1. Гонсалес, Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс. – М. : Техносфера, 2005. – 1070 с.
2. Acharya, T. Image processing. Principles and Applications / T. Acharya, K. Ray Ajoy. – John Wiley & Sons, 2005. – 449 p.
3. Chan, T.F. Image Processing and Analysis / T.F. Chan, J.H.G Shen. – Philadelphia : Society for Industrial and Applied Mathematics, 2005. – 423 p.

4. Park, J.M. Encyclopedia of Computer Science and Engineering / J.M. Park, M. Lu. – Wiley-interscience, 2008. – 2365 p.
5. Zalesky, B.A. Integer Programming Methods in Image Processing and Bayes Estimation / B.A. Zalesky // Soft Computing in Image Processing – Recent Advances Series: Studies in Fuzziness and Soft Computing. – 2007. – Vol. 210. – P. 417–446.
6. Zalesky, B.A. Gibbs Classifiers / B.A. Zalesky // Probability Theory and Mathematical Statistics. – 2004. – Vol. 70. – P. 36–46.
7. Залесский, Б.А. Фильтрация и кластеризация мультиспектральных изображений с помощью алгоритма максимального потока в сети на основе вычисления градиента / Б.А. Залесский, Д.В. Прадун // Информатика. – 2010. – № 27. – С. 73–80.
8. McDonnell, M. Box-filtering techniques / M. McDonnell // Computer Graphics and Image Processing. – 1981. – Vol. 17, no. 1. – P. 65–70.
9. Viola, P. Rapid Object Detection using a Boosted Cascade of Simple Features / P. Viola, M.J. Jones // Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR 2001). – 2001. – Vol. 1. – P. 511–518.
10. Пападимитриу, Х. Комбинаторная оптимизация / Х. Пападимитриу, К. Стайглиц. – М. : Мир, 1985. – 510 с.

Поступила 15.01.2013

*Объединенный институт проблем
информатики НАН Беларуси,
Минск, Сурганова, 6
e-mail: zalesky@newman.bas-net.by*

B.A. Zalesky

COMBINATORIAL ALGORITHM FOR OBJECT CONTOURS DETECTION OF DIGITAL IMAGES

An algorithm for the extraction of image contours is presented. The algorithm is based on combinatorial methods of clustering the oriented gradient of an image. It allows to estimate the position and the slope angles of the contours with a sufficient accuracy, and to obtain their vector representation by the broken lines or pieces of smooth curves, which is convenient for analyzing the shape of objects.