Список літератури

1. Кузнецов С.П. Динамический хаос: монография / С.П. Кузнецов. – М.: Физматлит, 2002. – 296 с.
2. Шустер Г. Детерминированный хаос. Введение : пер. с нем. / Генрих Шустер. – М.: Мир, 1988. – 256 с.
3. Круглов В.В. Нечеткая логика и искусственные нейронные сети / В.В. Круглов, М.И. Дли, Р.Ю. Голунов. – М.: Солон, 1996. – 348 с.
4. Корнієнко В.І. Ідентифікація нелінійних процесів по часових реалізаціях / В.І. Корнієнко, Д.Ю. Скриль // Науковий вісник Національного гірничого університету – 2009. – №3. – С. 85–89.
5. Ивахненко А.Г. Долгосрочное прогнозирование и управление сложными системами / А.Г. Ивахненко. – К.: Техніка, 1975. – 312 с.
6. Сычев В. Фрактальный анализ. Программа Fractan 4.4 / Вячеслав Сычев. – http:// impb.ru/~sychyov/.

Разработана модифицированная методика идентификации сложных природных процессов по временным реализациям, которая включает определение режима работы порождающей системы и его характеристик, синтез математической модели. На примере прогнозирования расходов рек Днепра, Немана и Северского Донца показана эффективность этой методики.

Ключевые слова: *идентификация, прогнозирование, стохастический процесс, часовая реализация, нелинейная система*

It has been designed the modified methodology for identification of stochastic natural processes on temporal realizations, which includes determination of the mode of operation of the originative system, its descriptions and synthesis of mathematical model. The efficiency of this method is shown on the example of prediction of flow of the rivers Dniper, Neman and Siverskyi Donets.

Keywords: *identification, prediction, stochastic process, temporal realization, nonlinear system*

Рекомендовано до публікації д.т.н. Г.В. Кузнецовим. Дата надходження рукопису 15.09.10

УДК 681.3

Т.А. Рузова

УСОВЕРШЕНСТВОВАНИЕ АЛГОРИТМА ГРЭХЕМА ДЛЯ ПОСТРОЕНИЯ ВЫПУКЛЫХ ОБОЛОЧЕК ОБЪЕКТОВ РАСТРОВОЙ ГРАФИКИ

T.A. Ruzova

GRAHAM ALGORITHM DEVELOPMENT FOR CONSTRUCTING CONVEX HULLS OF BITMAPPED OBJECTS

Проведен анализ проблем, возникающих при использовании алгоритма Грэхема для построения выпуклых оболочек к объектам растровой графики. Предложено усовершенствование алгоритма для построения выпуклых оболочек объектов с целочисленными координатами. Модификация алгоритма позволяет избежать погрешностей, связанных с дискретным представлением контура, при построении оболочек объектов сложной конфигурации. Предложенный подход может быть использован при разработке компьютерных систем для решения широкого круга задач.

Ключевые слова: *алгоритм Грэхема, выпуклая оболочка, растровое изображение, граница объекта, полярный угол*

Введение. Понятие выпуклой оболочки широко используется при решении различных задач вычислительной геометрии [1, 2]. Таких, на первый взгляд, далеких от нее как задача о максимумах множества точек, а также для широкого круга задач статистики. Если многомерные статистические данные рассматривать как точки в евклидовом пространстве, то многие статистические задачи (такие как задача о робастном оценивании, монотонной регрессии и др.) можно рассматривать с точки зрения вычислительной геометрии. Во многих задачах построение выпуклой оболочки является определяющим моментом. Выпуклые оболочки находят применение и в задачах распознавания образов. Один из часто используемых способов описания формы объекта основан на построении его выпуклой оболочки и определении дефицита выпуклости [3, 4]. Понятие дефицита выпуклости используется также при построении деревьев вогнутости, в задачах о восстановлении формы объектов на изображении [5]. Выпуклые оболочки применяются также для сегментации агрегированных объектов [6]. Построение выпуклых оболочек является существенным этапом решения задач компоновки и раскроя материалов [7]. Поэтому методам построения выпуклых оболочек посвящено большое число публикаций [1, 8–12]. К числу наиболее известных относят алгоритмы Грэхема, Джарвиса, Чена.

Постановка задачи. Алгоритм Грэхема [11] является одним из наиболее простых и позволяет построить выпуклую оболочку N точек на плоскости за время *O(N log N)* при памяти O(N) с использованием только арифметических операций и сравнений [1]. Однако, применение этого алгоритма в дискретных системах, к которым относятся изображения растровой графики, сопряжено с рядом проблем, связанных с использованием полярных координат. При переходе из непрерывного пространства вещественных чисел к дискретным координатам объектов растровой графики возникают погрешности, которые приводят к ошибкам в работе алгоритма.

Целью данной работы является усовершенствование алгоритма Грэхема для построения выпуклых оболочек объектов с целочисленными координатами.

Методы решения. Алгоритм Грэхема состоит из следующих операций. Переносим начало координат в некоторую внутреннюю точку исследуемого множества. Затем упорядочиваем точки по возрастанию полярного угла. В качестве начальной выбираем точку, заведомо являющуюся вершиной выпуклой оболочки. При совпадении значений полярного угла у нескольких точек проводим их упорядочивание по расстоянию от начала координат. Таким образом получена связанная упорядоченная последовательность точек. Суть метода состоит в анализе этой последовательности с целью устранения внутренних точек.

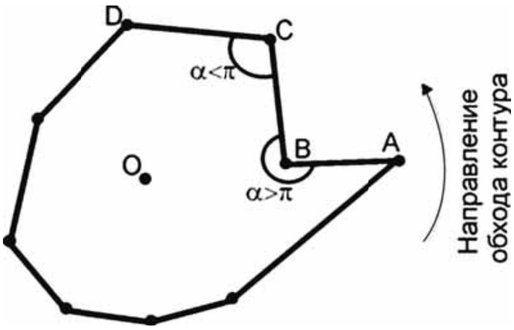


Рис. 1. Схема обхода контура методом Грэхема

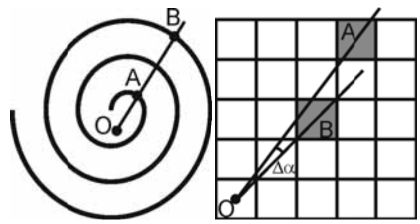
Для этого вводится понятие правого и левого поворотов. Точки образуют левый поворот, если при обходе контура против часовой стрелки внутренний угол, образованный ими, меньше π, в противном случае точки образуют правый поворот. Так, точки BCD контура, изображенного на рис. 1, образуют левый поворот, а точки ABC – правый.

При обходе выпуклого многоугольника могут выполняться только левые повороты. В случае правого поворота ABC точка B является внутренней и при построении оболочки должна быть исключена, так как не может быть крайней точкой (она является внутренней точкой треугольника OAC). Таким образом, при переборе точек контура возможны два варианта:

1. Если точки образуют левый поворот (BCD), вносим рассматриваемую точку (C) в список узлов выпуклой оболочки и переходим к анализу следующей тройки.

2. Если точки образуют правый поворот (ABC), то рассматриваемую точку (B) отклоняем.

При построении выпуклых оболочек к простым фигурам применение этого алгоритма не вызывает затруднений. Однако при обходе фигур со сложным контуром возникают трудности, связанные с погрешностями перехода от целочисленных координат рассматриваемого объекта растрового изображения в пространство вещественных чисел. Например, у абстрактной спирали, изображенной на рис. 2 *а*, точки *A* и *B* характеризуются одинаковым значением полярного угла, лежат на одном луче, исходящем из точки *O* – полюса системы координат. Однако, растровое представление этого объекта содержит погрешности. Так, точки *A* и *B* этой спирали, представленной в растровом формате, находятся на разных лучах. Лучи, проведенные из полюса *O* в центры пикселов, соответствующих этим точкам, различаются на угол $\Delta\alpha$.



а б

Рис. 2. Погрешность растрового представления объекта

Кроме того, погрешность $\Delta\alpha$ определения полярного угла не является постоянной для одного и того же объекта. Шаг изменения полярного угла при последовательном переборе точек контура не является постоянным, а зависит также от кривизны контура. На участках контура меньшей кривизны изменение угла больше, чем на участках с большей кривизной. На рис. 3 $\Delta\alpha_1<\Delta\alpha_2$. Такое непостоянство шага не позволяет ввести пороговое значение для погрешности определения угла.

Таким образом, точки *A* и *B* (рис. 2) с одинаковым значением полярного угла будут неверно внесены в упорядоченный список точек контура, что приведет к ошибке в работе алгоритма.

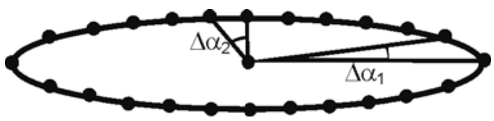


Рис. 3. Изменение полярного угла на участках контура различной кривизны

Вследствие этого автором предложено внести изменения, которые позволяют избежать указанного недостатка. Работа модифицированного алгоритма строится следующим образом:

1. Выбираем внутреннюю точку *O* множества, например центр тяжести фигуры.

2. В качестве стартовой точки выбираем самую верхнюю точку контура с наименьшей абсциссой.

3. Осуществляем обход контура и проводим последовательный анализ троек вершин на правый и левый поворот. Если тройка *ABC* образует правый поворот, точку *B* отклоняем, а в случае левого поворота осуществляем ревизию строящейся оболочки следующим образом (рис. 4). Для каждой точки *P* построенного участка выпуклой оболочки проверяем, не пересекает ли отрезок *BP* построенный участок выпуклой оболочки в некоторой точке *Q*, отличной от *P* и *B*. При наличии такого пересечения вносим точку *Q* в список узлов оболочки, а участок *PQ* ис-

ключаем из оболочки (на рис. 4 исключаемый участок представлен пунктирной линией).

Шаг 3 повторяем до получения оболочки, не содержащей правых поворотов и точек пересечения.

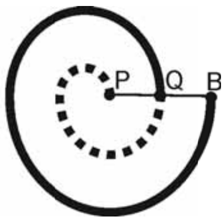


Рис. 4. Корректировка выпуклой оболочки

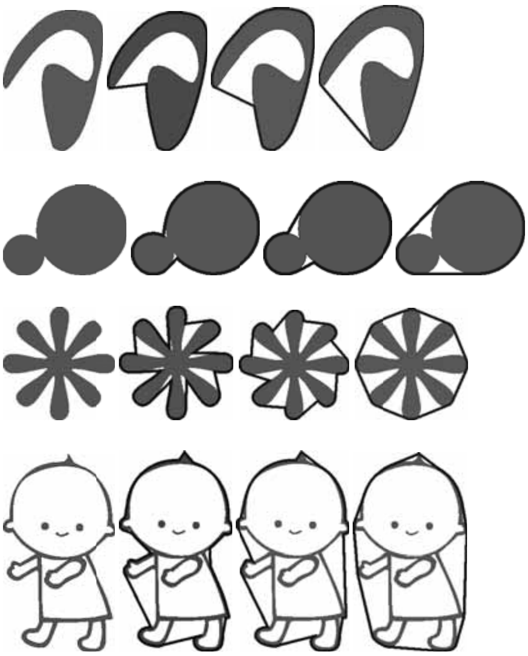


Рис. 5. Поэтапное построение выпуклой оболочки

На рис. 5 приведен пример поэтапного построения выпуклых оболочек для объектов различной степени сложности. Обход контура осуществлялся по часовой стрелке.

Выводы. В результате анализа проблем, возникающих при практическом использовании алгоритма Грэхем для построения выпуклых оболочек к объектам, представленным на изображениях растровой графики, предложено усовершенствование алгоритма, исключающее сортировку точек границы объекта по углу в полярной системе координат и позволяющее избежать погрешностей, связанных с дискретным представлением контура, в определении полярного угла, что, тем самым, позволяет избежать ошибок при построении оболочек объектов сложной конфигурации.

Предложенный подход может быть использован при разработке компьютерных систем, использующих выпуклые оболочки для решения различных задач вычислительной геометрии, распознавания образов, сегментации агрегированных объектов, компоновки и раскроя материалов и др.

Список литературы

1. Препарата Ф., Шеймос М. Вычислительная геометрия: Введение. – М.: Мир, 1989. – 478 с.
2. Рюэль Д. Статистическая механика. Строгие результаты. – М.: Мир, 1971. – 368.
3. Дуда Р., Харт П. Распознавание образов и анализ сцен. – М.: Мир, 1976. – 511.
4. Batchelor B.G., Whelan P.F. Intelligent vision systems for industry. – Springer-Verlag, 1997. – P. 457.
5. Badawy O., Kamel M. Shape Retrieval using Concavity Trees // Proceedings of the Pattern Recognition, 17th International Conference. – 2004. – Vol. 3. – P. 111–115.
6. Kutalik Z. Razaz M. Occluding convex image segmentation for e.coli microscopy images // XII European Signal Processing Conference EUSIPCO. – 2004. – P. 937–940.
7. Freeman H. Computer processing of line-drawing images // ACM Computing Surveys (CSUR). – 1974.– Vol. 6. – P. 57–97.
8. Kim C.E., Stojmenovic I. Sequential and parallel approximate convex hull algorithms// Computers and Artificial Intelligence. – 1995. – Vol. 14. – №6. – P. 597–610.
9. Badashian A.S., Razzazi M.R. Kinetic convex hull algorithm using spiral kinetic data structure // International Conference Computational Science and its Applications. – Kuala Lumpur. – 2007. – P. 216–226.
10. Brönnimann H., Iacono J. Space-efficient planar convex hull algorithms //Theoretical Computer Science.– 2004. – Vol. 31(1). – P. 25–40.
11. Graham R.L. An efficient algorithm for determining the convex hull of finite planar set // Information Processing Leters. – 1:132–133.– 1972.
12. An P. A modification of Graham's algorithm for determining the convex hull of a finite planar set // Annales Mathematicae et Informaticae. – 2007.– Vol. 34.– P. 3–8.

Проведено аналіз проблем, які виникають при застосуванні алгоритму Грехема для побудови опуклих оболонок до об'єктів растрової графіки. Запропоновано вдосконалення алгоритму для побудови опуклих оболонок об'єктів із цілочисельними координатами. Модифікація алгоритму дозволяє уникнути похибок, пов'язаних із дискретним представленням контуру, при побудові оболонок об'єктів складної конфігурації. Запропонований підхід може бути використаний при розробці комп'ютерних систем для рішення широкого кола задач.

Ключові слова: *алгоритм Грехема, опукла оболонка, растрове зображення, границя об'єкта, полярний кут*

It has been analysed the problems occuring while using Graham algorithm for constructing convex hulls of bitmapped objects. It is proposed the algorithm of making hulls for geometrically-complex objects improvement which allows avoiding errors caused by discrete representation of contour. Proposed approach can be used for designing software concerning wide range of problems.

Keywords: *Graham algorithm, convex hull, bitmapped image, object's contour, polar angle*

Рекомендовано до публікації д.т.н. В.І. Корсуном. Дата надходження рукопису 22.09.10