

Устойчивое выделение прямолинейных участков замкнутых точечных кривых обладающих поворотной симметрией

К.Б. Мариничев, В.П. Вежнев

Лаборатория компьютерной графики и мультимедиа факультета ВМиК,
Московский государственный университет им. М.В.Ломоносова, Москва, Россия

kirill.marinichev@gmail.com, vvp@graphics.cs.msu.su

Аннотация

В задачах анализа изображений часто требуется выделение прямолинейных участков точечных кривых. Основной трудностью в решении этой задачи является наличие шума и выбросов в данных. В статье приведен новый алгоритм выделения прямолинейных участков, основанный на алгоритме Дугласа-Пекера и наличии поворотной симметрии в данных, что позволяет сильно повысить устойчивость его работы к шуму и выбросам.

Ключевые слова: Упрощение ломаных, задача min-#, автокорреляция, алгоритм Дугласа-Пекера.

1. ВВЕДЕНИЕ

Одной из важнейших задач компьютерного зрения является задача анализа точечных кривых. Зачастую ей предшествует подзадача выделения прямолинейных участков в этих точечных кривых для облегчения анализа. В ходе анализа физических данных, в подавляющем большинстве случаев приходится сталкиваться с их **зашумленностью** и наличием **выбросов**. Для борьбы с ними необходима некая дополнительная информация. В данной статье речь пойдет о точечных кривых, имеющих избыток информации в виде *поворотной симметрии* (см. рис. 1). Кривая обладает поворотной симметрией, если при повороте вокруг некоторой внутренней точки она отображается на себя.

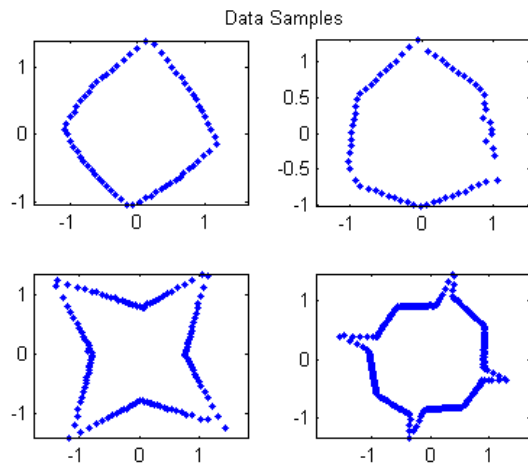


Рис. 1. Примеры входных данных

2. ПОСТАНОВКА ЗАДАЧИ

На вход алгоритму даются:

- замкнутая точечная кривая (ломаная) $(X_i, Y_i), i = 1..n$
- требуемая точность ε

На выходе:

- индексы ключевых точек $J_i, i = 1..k, k \leq n$

При этом ошибка упрощения не должна превышать заданной величины ε . Вводится она, как и в задаче min-# [1] с одной лишь поправкой: при подсчете ошибки – не должны учитываться точки выбросов (см. рис. 2). Если же выбросов в данных нет, то ошибка считается обычным образом.

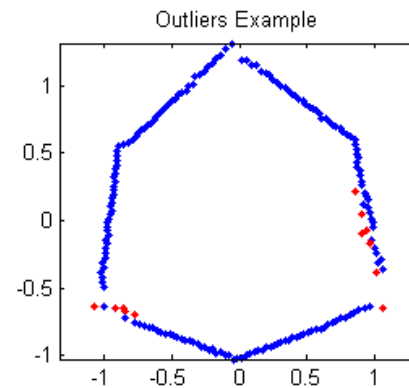


Рис. 2. В подсчете ошибки упрощения должны участвовать только синие точки, красные - выбросы

3. ПРЕДЛОЖЕННЫЙ АЛГОРИТМ

3.1 Основная идея

В статье [1] проводится обзор и сравнение более чем 10 существующих подходов и алгоритмов решения задачи упрощения точечных кривых. Но поскольку ни один из них не призван бороться с шумами и выбросами в данных, основной нашей задачей было исключить эти шумные данные и/или выбросы за счет некоторой дополнительной информации. При условии наличия поворотной симметрии данных, эта дополнительная информация для каждой точки берется от симметричных ей точек.

3.2 Этапы алгоритма

Таким образом, можно выделить 2-е основные фазы алгоритма:

1. Определение степени и периода поворотной симметрии точек
2. Выделение прямолинейных участков данных с учетом поворотной симметрии

3.3 Определение периода

3.3.1 Подготовка данных

Прежде чем определять степень поворотной симметрии данных, нужно их подготовить. Переведя точки в полярные координаты ($R_i, \varphi_i, i = 1..n$, см. рис. 3) сделаем следующее:

1. упорядочим точки по углу.
2. путем добавления новых точек, сделаем так, чтобы приращение угла φ было не более δ (т.е. $\varphi_i - \varphi_{i-1} \approx const$). При этом радиус R_i добавленных точек получим линейной интерполяцией.

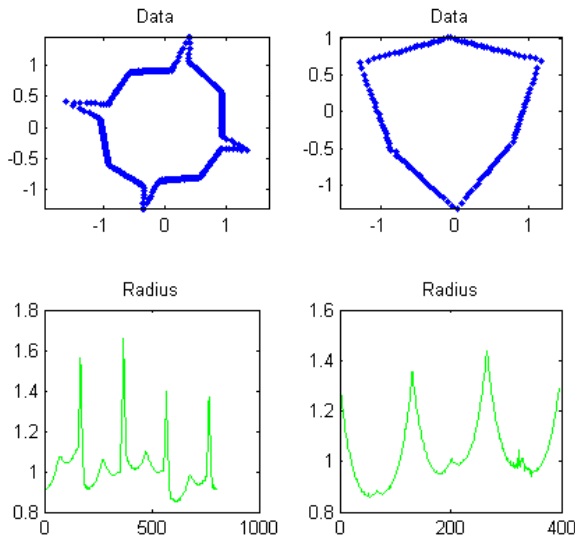


Рис. 3. сверху изображены входные данные, снизу – соответствующие им графики радиусов (по оси абсцисс идут индексы точек)

3.3.2 Автокорреляция

Для определения периода R_i , используется автокорреляция [2]. В каждой точке j она считается как:

$$Acorr_j = \sum_{i=1..n-j} R_i * R_{i+j} + \sum_{i=n-j+1..n} R_i * R_{i-n+j}, j = 0.. \left\lfloor \frac{n}{2} \right\rfloor$$

т.е. R_i , циклически сдвигается на j и скалярно умножается на R_{i+j} . По свойству автокорреляции, её локальные максимумы достигаются при индексах равных периодам сигнала R_i . А значит, найдя ближайший к 0 максимум автокорреляции, мы найдем период **Period** (см. рис. 4).

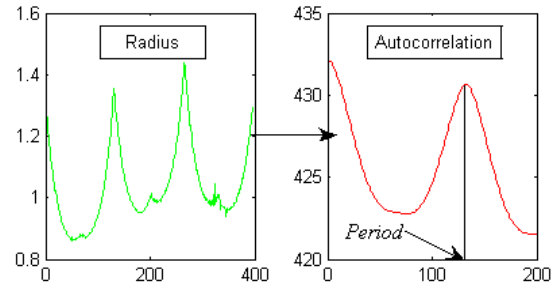


Рис. 4. для графика радиуса(слева) приведен график автокорреляции (справа).

3.4 Модифицированное упрощение

3.4.1 Инициализация

Теперь зная период нашего сигнала, применим к нему модифицированный алгоритм Дугласа-Пекера. Прежде всего, алгоритму необходимо передать индексы первой и последней точки, с которых он начнет работу.

Поскольку кривая является замкнутой, нам не известно в какой точке следует её разомкнуть. В статье [3] предлагается в качестве начальной точки выбрать максимально удаленную от центра массы точку (в нашем случае $\max(R_i)$). Однако, при наличии выбросов в данных велика вероятность выбрать именно точку сильного выброса. Что бы сделать алгоритм выбора более устойчивым, используем знание о поворотной симметрии – определим индекс точки по следующей формуле (см. рис. 5):

$$StartInd = \arg \max_{i=1..Period} median \left(\bigcup_{k=0, \lfloor \frac{n}{Period} \rfloor - 1} R_{i+k*Period} \right)$$

где функция $median(X)$ - возвращает медиану значений набора X . Рис. Что стоит отметить:

- точка ищется внутри периода
- даже если конкретная точка i является сильным выбросом, то соответствующие точки из других периодов (т.е. $i + Period, i + 2*Period, \dots$) не позволят выбрать её в качестве точки разрыва.

Далее, для простоты изложения, будем предполагать, что точка разрыва имеет индекс 1 – этого можно добиться простой переиндексацией точек.

Определив точку начала ($i = 1$), мы на самом деле определили $n/Period$ точек ($1, Period, 2*Period, \dots$). Все они являются точками начала в силу поворотной симметричности данных и добавляются в выходной массив J_i .

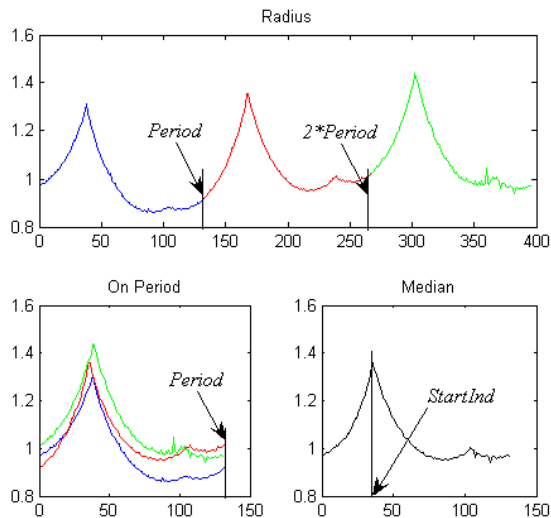


Рис. 5. Пример поиска точки начала. График радиуса (сверху) разбивается на периоды, и все периоды совмещаются в один (снизу слева). Далее они усредняются с помощью $median(X)$ (снизу справа).

3.4.2 Рекурсия

Следующим шагом производится упрощение ломаной алгоритмом Дугласа-Пекера ([4]), только работа ведется на одном периоде ($[1..Period]$), и её результаты отображаются на все периоды данных.

Поскольку алгоритм является рекурсивным, рассмотрим подробно один шаг.

1. Пусть нам необходимо упростить точечную кривую с индексами $i:(a \leq i \leq b)$, ($1 \leq a < b \leq Period$).
2. Составим функцию усредненного по периодам расстояния:

$$E(j) = median\left(\bigcup_{k=0, \lfloor \frac{n}{Period} \rfloor - 1} \rho\left(\begin{matrix} a + k * Period, \\ b + k * Period, \\ j + k * Period \end{matrix}\right)\right)$$

Здесь функция $\rho(a, b, j)$ – определяет расстояние от точки j до отрезка построенного на точках a и b .

Еще стоит отметить, что используется та же функция $median(X)$. Т.е. даже если некоторая точка l оказалась выбросом, то точки из других периодов ($l + Period, l + 2*Period, \dots$) компенсируют ошибку, и в результате функция $E(l)$ будет иметь адекватное значение.

3. Если максимум функции $E(l)$ меньше ε (входной порог алгоритма), то дальнейшее упрощение не требуется. В ином случае переходим в п.4.
4. Найдем точку, в которой достигается максимум расстояния:

$$c = \arg \max_{j=a..b} E(j)$$

5. Индексы $c, c + Period, c + 2*Period, \dots$ не всегда определяются точно, поскольку количество точек

не всегда делится нацело на период ($n/Period$ – не целое), а периоды могут отличаться. Поэтому, на каждом периоде, производится свое локальное уточнение индекса в небольшой δ :

$$c_k = \arg \max_{j=c-\delta..c+\delta} \rho\left(\begin{matrix} a + k * Period, \\ b + k * Period, \\ j + k * Period \end{matrix}\right), k = 0.. \left\lfloor \frac{n}{Period} \right\rfloor$$

6. Индексы c_k добавляем в выходной массив J_i .
7. Далее применяем вышеописанную последовательность действий для отрезков $[a, c]$ и $[c, b]$.

4. РЕЗУЛЬТАТЫ

Вышеописанный алгоритм был реализован в виде программной библиотеки и успешно применяется на одном из шагов системы по построению 3D моделей многогранников по силуэтам. В рамках этой задачи было произведено тестирование алгоритма на более чем 500 примерах (примеры см. рис. 6, 7).

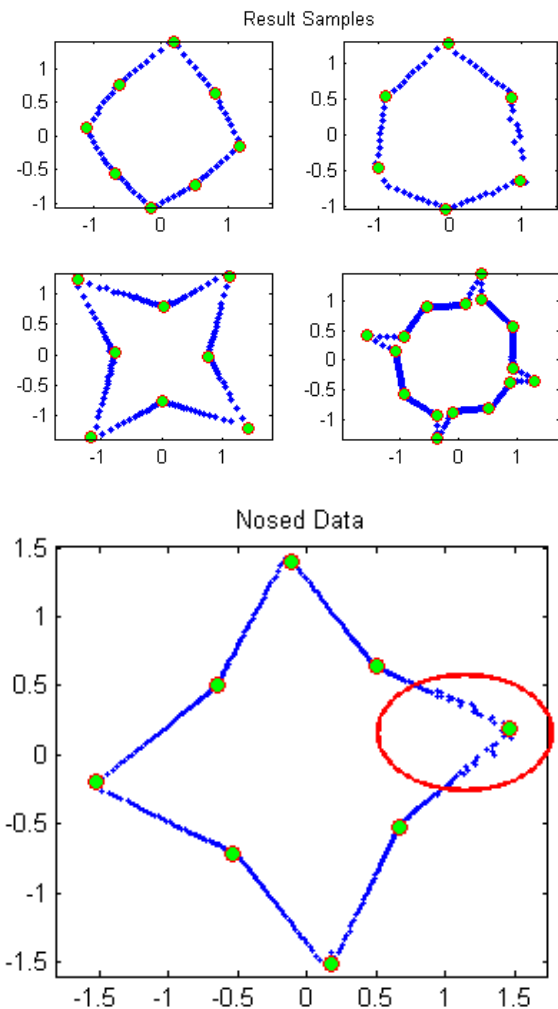


Рис. 6. Примеры работы алгоритма. Синим – отмечены входные точки, зеленым – выходные точки.

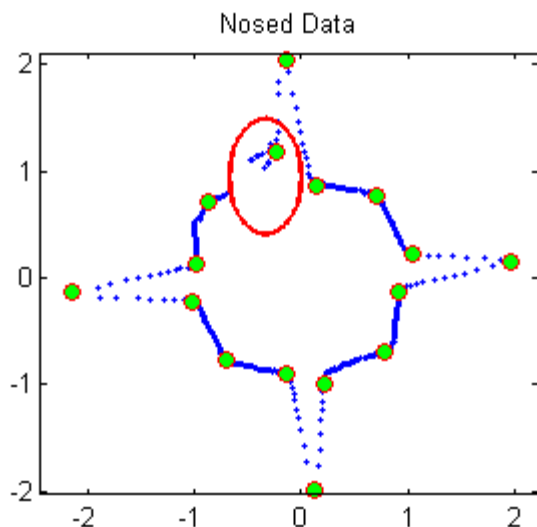


Рис. 7. Примеры работы алгоритма. Синим – отмечены входные точки, зеленым – выходные точки.

5. ОБСУЖДЕНИЕ

Основу устойчивости алгоритма составляет функция $median(X)$, которая применяется в инициализации и в основном цикле. Она производит усреднение по периодам, но если их меньше 3-х, то её применение практического смысла не имеет. Таким образом, алгоритм проявляет свою устойчивость при наличии более чем 2-х периодов, иначе он работает как обычный алгоритм Дугласа-Пекера.

Так же из свойств функции $median(X)$, следует, что алгоритм не будет устойчив в точке, если большая часть периодов в ней сильно зашумлены или имеют выбросы. Тогда эти выбросы справедливо рассматриваются как исходная структура.

6. ЗАКЛЮЧЕНИЕ

В статье был представлен алгоритм устойчивого выделения прямолинейных участков точечных кривых, имеющих поворотную симметрию. Этот алгоритм показал высокую устойчивость к шумам и выбросам в данных.

7. БИБЛИОГРАФИЯ

- [1] Ph.D. Alexander Kolesnikov, *PhD Thesis: Efficient Algorithms for Vectorization and Polygonal approximation*, http://cs.joensuu.fi/~koles/approximation/Ch3_0.html
- [2] Л. Рабинер и Б. Голд "Теория и практика цифровой обработки сигналов", МИР 1978
- [3] Sato Y., *Piecewise linear approximation of planar curves by perimeter optimization*, Pattern Recognition, 25(12): 1535-1543, 1992.
- [4] Douglas D.H., Peucker T.K., *Algorithm for the reduction of the number of points required to represent a line or its caricature*, The Canadian, Cartographer, 10 (2): 112-122, 1973.

Об авторах

Мариничев Кирилл Борисович – студент 5-ого курса Лаборатории Компьютерной Графики и Мультимедиа факультета Вычислительной Математики и Кибернетики Московского Государственного Университета им. М.В.Ломоносова.

Адрес: Москва, Воробьевы горы, МГУ, 2-й учебный корпус, факультет ВМиК, кафедра АСВК

Телефон: (495) 939-01-90

E-mail: kirill.marinichev@gmail.com

Вежневцев Владимир Петрович – к.ф.-м.н., научный сотрудник Лаборатории Компьютерной Графики и Мультимедиа факультета Вычислительной Математики и Кибернетики Московского Государственного Университета им. М.В.Ломоносова.

Адрес: Москва, Воробьевы горы, МГУ, 2-й учебный корпус, факультет ВМиК, кафедра АСВК

Телефон: (495) 939-01-90

E-mail: vvp@graphics.cs.msu.su

Robust simplification of rotation symmetrical polylines

Abstract

The problem of polyline simplification is the basic one in Computer Vision. It is often needed in image analyzing problems. But fundamental algorithms do not providing good results for noised data or data with outliers. In this paper is proposed new algorithm of polyline simplification based on Douglas- Peucker algorithm for self correlated data, so it became more robust to noise and outliers.

Keywords: *polyline simplification, min-# problem, autocorrelation, Douglas- Peucker algorithm.*

About the author(s)

Marinichev Kirill is a student at Moscow State University, Department of Mechanics and Mathematics. His contact email is kirill.marinichev@gmail.com

Dr. Vezhnevets Vladimir is a Researcher at Moscow State University, Department of Mechanics and Mathematics. His contact email is vvp@graphics.cs.msu.su