

Э. А. МНАЦАКАНЯН, С. В. НЕФЕДОВ

*(Куйбышев)***ОБ ОДНОМ АЛГОРИТМЕ СКЕЛЕТИЗАЦИИ  
ПОЛОСОВЫХ ИЗОБРАЖЕНИЙ**

Нахождение линий центров максимумов и минимумов полос является в ряде случаев необходимым этапом автоматизированного получения количественных данных об объектах, исследуемых интерферометрическими, муаровыми и другими методами [1]. Как правило, для получения таких линий исходное полосовое изображение фильтруют с целью подавления шумов, затем путем порогового усечения образуют бинарные изображения разрывов (и желательным типом связности). Так, предложенный впервые метод скелетизации, описанный в [2] и развитый затем в [3—5], заключающийся в растровом сканировании бинарного кадра некоторым набором масок  $2 \times 2$ ,  $3 \times 3$  или их комбинаций, приводит к неоднозначности нахождения скелета, которая определяется порядком применения масок и ориентацией скелетизируемого объекта. На наш взгляд, наиболее эффективный метод для скелетизации широкого класса изображений — метод преобразования к срединным (медианным) осям [6, 7]. В [8] продемонстрирована реализация этого метода для сильно зашумленных бинарных изображений. Преимуществами его являются инвариантность к ориентации скелетизируемых объектов, независимость от порядка применения масок и направления сканирования изображения, хотя и он не свободен от некоторых недостатков, в частности количество разрывов линий скелета после утоньшения для объектов сложной формы может быть значительным.

Указанный метод положен в основу предлагаемого в настоящей работе алгоритма, поскольку в ряде случаев, в частности в голографической и спекл-интерферометрии, даже после фильтрации получающиеся изображения зашумлены, а полосы могут иметь достаточно сложный вид. Цель данной работы — разработка алгоритма и соответствующего программного обеспечения, позволяющего обрабатывать широкий класс бинарных изображений, обладающего требуемыми характеристиками и ориентированного на конечного пользователя.

Алгоритм скелетизации состоит из шести последовательно выполняемых этапов.

1-й этап. Итерационное выделение локальных максимумов бинарного изображения (точек, принадлежащих медианным осям бинарного объекта). Этот этап состоит из трех фаз и требует трех бинарных кадров. Исходное бинарное (утоњшаемое) изображение загружается в кадр 1, а кадры 2 и 3 предварительно очищаются.

1-я фаза. Выделение границы объектов (полос). Исходное бинарное изображение, подвергаемое утоньшению и расположенное в кадре 1, сканируется на предмет обнаружения «1» («1» — принадлежит объекту, «0» — не принадлежит). В случае появления «1» проверяется, лежит пиксел на границе объекта или нет. Принадлежность границе определяется по наличию хотя бы одного «0»-пиксела, связанного с данным «1»-пикселом по 4-связности. Найденные таким образом точки границы в процессе сканирования кадра 1 «выставляются» по тем же координатам в кадр 2.

объект в кадре 1 и его границу в кадре 2.

3-я фаза. Нахождение локальных максимумов и пересылка их в бинарный кадр 3. На этом этапе сканируется кадр 2 на предмет обнаружения «1», т. е. сканируется граница объекта, подвергнутого утоньшению во 2-й фазе. Для каждого пиксела границы осуществляется проверка, связан ли он с «1»-пикселами в кадре 1, полученном в результате утоньшения, по восьми направлениям. Если связности не обнаружено, то данная точка границы является локальным максимумом. Обнаруженные таким образом локальные максимумы «выставляются» в бинарный кадр 3, который перед 1-й фазой был заполнен нулями. По завершении сканирования кадр 2 очищается, и осуществляется следующая итерация с 1-й фазы.

Итерации прекращаются, и 1-й этап завершается, когда в кадре 1 не остается ни одного пиксела, содержащего «1». После данного итерационного процесса в кадре 3 находятся все локальные максимумы бинарного объекта. Конечный результат утоньшения такой же, как и в [8]. Однако в [8] поиск локальных максимумов осуществлялся с помощью метода преобразования расстояний, когда каждая последующая граница, начиная с 1-й, пронумеровывалась числами 1, 2, 3, ..., а выделение локальных максимумов осуществлялось по 8-несвязности границы с данным номером с пикселами на единицу меньше. Преимущество данной реализации утоньшения заключается в использовании 3-битовых кадров, вместо двух 8-битовых, что дает выигрыш в памяти более чем в 5 раз и позволяет увеличить размеры обрабатываемых изображений непосредственно в оперативной памяти ЭВМ.

Модифицирована и стратегия поиска локальных максимумов. Указанная стратегия должна удовлетворять следующим условиям: совпадение выделенных линий с серединными осями полос (отсутствие ложных максимумов), минимальное число и длина разрывов. Разрывы линий скелета являются следствием изрезанности за счет шумов границ бинарных объектов и изменений их кривизны. Длины этих разрывов могут быть достаточно велики в областях, связанных с большой кривизной полос, их разветвлением и сильной зашумленностью объектов.

В [8] поиск локальных максимумов осуществлялся с использованием только 8-связности, что удовлетворяет лишь первому условию. При использовании только 4-связности в 3-й фазе линии скелета оказываются сильно зашумлены единичными и связными группами пикселов, отстоящих на различных расстояниях от линий скелета, а также шумовыми короткими ответвлениями, хотя разрывов самого скелета существенно меньше. По указанным причинам в данной работе использовалась 8-связность на начальных итерациях и 4-связность — на конечных. Такое сочетание при правильном выборе числа итераций с использованием 8-связности приводит к значительно большему соответствию вышеперечисленным трем условиям. Для оценок числа итераций по 8-связности можно воспользоваться полученным эмпирически простым правилом. Если концы полос закруглены, заострены или полосы замкнуты, число итераций по 8-связности определяется максимальной высотой шумовых неровностей полосы. Если концы полос широки и имеют резкие края, число таких итераций выбирается приблизительно равным половине максимальной ширины конца полосы.

2-й этап. Получение определенного типа связности. Множество локальных максимумов, получаемых с помощью 1-го этапа, как и в [8], не является отрезками 8-связных линий. Требование сохранения 8-связности для всех точек скелета, кроме точек разветвления полос, необходимо для алгоритмов сканирования скелета и автоматического счета полос. Эти алгоритмы могут использоваться на последующих этапах расшифровки интерферограмм [9, 10]. Для получения непрерывных 8-связных ли-

ний бинарный кадр на данном этапе подвергается обработке определенным набором масок:

1. Кадр сканируется на предмет обнаружения «1». Если «1» обнаружена, то проверяется наличие «1» в окрестности, задаваемой последовательностью четырех масок, приведенных на рис. 1. «.» обозначен «1»-пиксел, обнаруженный при сканировании кадра. Если любая из последовательности конфигураций, заданных рис. 1, обнаружена, то пиксел, помеченный «.», устанавливается в «0», после чего продолжается сканирование. Подобная обработка приводит к тому, что большая часть участков линий двойной толщины устраняется.

2. Аналогичная операция проводится с помощью масок, вид которых в соответствующей последовательности приведен на рис. 2: четыре маски  $3 \times 3$ ; восемь масок  $3 \times 3$ ; четыре маски  $3 \times 3$ ; восемь масок  $3 \times 3$ .

На рис. 1 и 2 штриховкой обозначены пикселы, установленные в «1», без штриховки — в «0». Последовательное применение масок из п. 2 приводит к устранению оставшихся нарушений 8-связности.

3-й этап. Соединение разрывов в один пиксел. Эта процедура выполняется эффективнее с помощью трех типов масок  $3 \times 3$  в указанной ниже последовательности, чем формированием стека концевых точек, как предложено в [8].

Бинарный кадр со скелетом сканируется с целью обнаружения «0». Если «0» обнаружен, то в окрестности  $3 \times 3$  проверяется наличие конфигураций, приведенных на рис. 3. Если подобная конфигурация из «1»-пикселов обнаружена, то пиксел сканирования, обозначенный «.», заменяется на «1». Очевидно, что существуют восемь масок такого же типа, как последняя маска на рис. 3, и сканирование кадра проводится всеми.

Выполнение описанных выше методик в строгой их последовательности приводит к тому, что скелет принимает 8-связный непрерывный вид с небольшим числом разрывов длиной более 1 и нарушением 8-связности только в точках ветвления линий.

4-й этап. Обработка точек ветвления. Точка ветвления — это пиксел на скелете, где должна нарушаться 8-связность линий. Если в точке ветвления полоса разветвляется на две, то при описанной выше обработке в области ветвления могут возникнуть дополнительные пикселы с нарушением 8-связности.

Для того чтобы остался один единственный пиксел с нарушением 8-связности, кадр обрабатывается четырьмя масками такого типа, как на рис. 4. Кадр сканируется с целью обнаружения «1». Если «1»-пиксел найден, то проверяется только наличие конфигурации из «1»-пикселов, помеченных «+». Величины остальных пикселов в окрестности точки сканирования не имеют значения. Если подобная конфигурация «1»-пикселов найдена, то пиксел, помеченный «.», устанавливается в «0». Затем к кадру применяется последовательность масок, описанная в третьем этапе. Подобная обработка приводит к тому, что нарушение 8-связности при разветвлении полос на две сохраняется только в одном пикселе.

В случае более сложных систем полос, когда возможны разветвления на четыре полосы, можно требовать нарушения 8-связности в одном пикселе или в двух соседних.

5-й этап. Соединение разрывов более чем в один пиксел. Соединение разрывов осуществляется отрезками прямых, проходящих от одной концевой точки до другой. Так как получаемые с использованием 4-связности на последних итерациях разрывы линий скелета не имеют большой длины, то подобный способ не сильно искажает «реальный» скелет. Рассматриваемое соединение осуществляется в несколько шагов:

1. Кадр, содержащий скелет, сканируется с целью обнаружения концевых точек линий. Концевыми точками считаются также изолированные «1»-пикселы, которые могут встречаться на линиях скелета. Концевые точки обнаруживаются с помощью соответствующего полного набора масок. Из координат концевых точек формируют стек концевых точек следующим образом: если концевая точка найдена, то в стеке ре-



Рис. 1

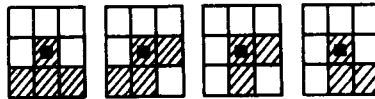


Рис. 2

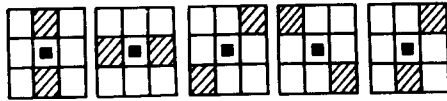


Рис. 3



Рис. 4

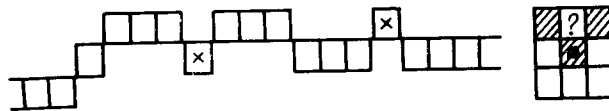


Рис. 5

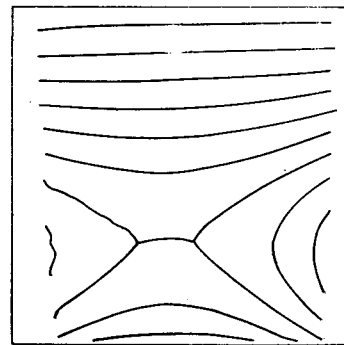


Рис. 6

зервируется место для указателя длины группы, вслед за которым заносятся координаты обнаруженной концевой точки. Затем отрезок скелета, начиная с этой точки, сканируется до следующей концевой точки. Найденная таким образом вторая концевая точка также заносится в стек следом за предыдущей. В процессе сканирования отрезок дуги уничтожается. Эта процедура должна выполняться во вспомогательном кадре, например 2, в который предварительно переписывается кадр 3 со скелетом. Найденные таким образом две концевые точки, принадлежащие отрезку скелета, образуют группу длиной в два элемента, что и помечается в указателе длины группы. В конце операции сканирования мы имеем стек концевых точек, организованный попарно из концевых точек, принадлежащих отрезкам дуг скелета. Единичные пиксели представляют пары с одинаковыми координатами. В случае разветвления полос, например на три, формируется группа длиной в три элемента следующим образом: когда в процессе сканирования обнаружена точка разветвления, координаты впервые обнаруженной концевой точки данного участка скелета помещаются вслед за указателем, в который заносится общее число концевых точек группы. Как только участок скелета просканирован и все концевые точки найдены, их число заносится в указатель.

После того как координаты всех концевых точек и их общее число определены, переходят к следующему шагу — соединению разрывов.

2. Сначала соединяют разрывы наименьшей длины, потом большей и т. д. до наперед заданной максимальной длины. Максимальное значение длины соединения выбирается таким, чтобы не соединялись концы соседних линий, принадлежащих разным полосам. За критерий длины удобно выбрать квадрат расстояния между пикселями  $l^2 = (x_i - x_j)^2 + (y_i - y_j)^2$ , где  $x_i$ ,  $x_j$  и  $y_i$ ,  $y_j$  — координаты пикселей, между которыми ищется расстояние. Тогда последовательный ряд нарастания величины  $l^2$  будет иметь вид  $l^2 = 9, 10, 13, 16, 17, 18, \dots$

Для каждого текущего значения  $l^2$  стек концевых точек сканируется последовательно с первой точки до последней. При этом вычисляются

значения  $l^2$  для точки сканирования и всех остальных точек, которые еще не пройдены, исключая точки из одной с ней группы. Когда найдены все точки, исключают соединение концов коротких отрезков и дуг с большой кривизной.

Когда соединения для заданного максимального  $l^2$  выполнены, кадр со скелетом повторно обрабатывается набором масок, описанным во 2-м этапе, так как соединение концевых точек отрезками прямых в некоторых случаях может привести к нарушению 8-связности.

Следует подчеркнуть, что если на интерферограмме присутствуют замкнутые полосы, то необходимо сканирование стека по парам для замыкания концов дуг. Эта операция должна проводиться после соединения всех разрывов вдоль линий.

6-й этап. Сглаживание единичных выбросов линий скелета. В результате описанной выше последовательности обработок скелета на линиях возникают единичные нарушения гладкости скелета, которые могут быть как следствием зашумленности утоньшаемого объекта, так и следствием метода восстановления 8-связности. Подобные пикселы, помеченные «х», приведены на рис. 5. Для сглаживания подобных выбросов кадр со скелетом сканируется на предмет обнаружения «1». Если «1» найдена, то проверяется наличие конфигураций, задаваемых четырьмя масками, тип которых также приведен на рис. 5. Если любая из подобных четырех конфигураций обнаружена, то пиксел, помеченный «.», устанавливается в «0», а пиксел, помеченный «?», — в «1».

Описанный выше алгоритм скелетизации, разработанный на основе метода приведения к срединным осям, обеспечивает автоматизированное получение скелетов полосовых изображений широкого класса, в том числе с различным уровнем шумов. На основе этих алгоритмов создано функционирующее на ЭВМ СМ 1420 программное обеспечение, которое использовалось для обработки голографических интерферограмм синусоидально вибрирующих объектов сложной формы. Экспериментальные результаты, приведенные на рис. 6, демонстрируют высокое качество получаемых скелетов. Опыт обработки интерферограмм позволяет сделать также вывод о хороших реализационных характеристиках созданного программного обеспечения.

Интерактивный режим работы программного обеспечения позволяет оператору не только быстро исправлять остающиеся разрывы и другие ошибки при получении скелетов, но и подбирать необходимые режимы при обработке конкретных полосовых изображений.

#### СПИСОК ЛИТЕРАТУРЫ

1. Приборы для неразрушающего контроля материалов и изделий: Справочник/ Под ред. В. В. Клюева. — М.: Машиностроение, 1986.
2. Sherman H. A quasi-topological method for the recognition of line patterns // Information Process.— London: Butterworths, 1960.— P. 232.
3. Levaldi S. On shrinking binary picture patterns // Comm. Assoc. Comput.— 1972.— March 15(1).— P. 7.
4. Arcelli Cordella L., Levaldi S. Parallel thinning of binary pictures // Electron. Lett.— 1975.— 11.— P. 148.
5. Бутаков Е. А., Островский В. И., Фадеев И. Л. Обработка изображений на ЭВМ.— М.: Радио и связь, 1987.
6. Pat. 3339179 US. Pattern recognition preprocessing techniques/G. L. Shelton.— Publ. 1967.
7. Blum H. A transformation for extracting new descriptors of shape // Symposium Models for Perception of Speech and Visual Form/Ed. Weiant Whaten-Dunn.— Cambridge, Mass.: MIT Press, 1967.
8. Pasian F., Santin P. Shape information extraction in noisy environment // Patt Recogn. Latt.— 1983.— 2.— P. 103.

9. Mnatsakanyan E. A., Nefyodov S. V. Comparative analysis of two methods of interpolation at phase reconstruction in holographic interferometry // SPIE's Technical Symposium on High Power Lasers and Optical Computing, 14-19 January 1990.— Los Angeles, California, USA, 1990.
10. Mnatsakanyan E. A., Nefyodov S. V. Algorithms for automatic obtaining of preparations used for the counting of phase difference of holographic interferograms // Proc. of the 1-st Internat. Workshop on Automatic Processing of Fringe Patterns, Fringe 89 // Physical Research.— 10.— P. 124.

Поступила в редакцию 14 августа 1990 г.

## ГРАФИЧЕСКОГО ВВОДА

В настоящее время полуавтоматические планшетные устройства графического ввода, предназначенные для кодирования и ввода графической информации с заранее подготовленного графического документа (чертежа, схемы и т. п.) [1], после соответствующей схемно-конструктивной доработки находят применение в качестве устройств ввода в ЭВМ рукописного текста и графики в реальном времени [2].

Режим реального времени (РРВ) применительно к этим устройствам — это режим, при котором происходит кодирование координат графической линии, оставляемой на листе бумаги пишущим элементом электронного пера при перемещении последнего пользователем. Режим РВ необходим при идентификации лиц по стилю написания букв и слов [3], что очень важно в торговле, банковском деле, на контрольно-пропускных пунктах и т. п. Он также необходим при распознавании отдельных рукописных букв в человекомашинных системах, где требуется автоматическое преобразование рукописного текста в машинописный [4].

Достаточно широкое применение этот режим находит при передаче рукописного текста и графики по телефонному каналу [5], а также по радиоканалу [6] синхронно с речевой информацией. Известно, что этот режим используется в обучающих системах [7].

Таким образом, уже из этого краткого обзора можно сделать вывод, что проектирование, расчет и исследование устройств ввода рукописной графической информации (РГИ), имеющих режим реального времени, представляют актуальную научную и техническую задачу.

На рис. 1 изображена обобщенная блок-схема устройства графического ввода, обеспечивающая режим реального времени. Особенностью блок-схемы является наличие блока 11 временной дискретизации (БВД), задача этого блока — обеспечение циклического формирования кода ко-

Рис. 1. Обобщенная блок-схема устройства:

1 — планшет; 2 — координатные шины; 3 — токоограничивающие резисторы; 4X, Y — блок формирования кода грубого отсчета координатной оси X и Y соответственно; 5 — графическое индукционное перо; 6 — усилитель считывания; 7 — фазоимпульсный селектор; 8X, Y — блок формирования кода точного отсчета соответствующей координатной оси; 9 — блок управления; 10 — датчик касания; 11 — блок временной дискретизации; 12 — блок сжатия информации; 13 — блок связи с ЭВМ; коды ГО X, Y — коды грубого отсчета соответственно координатной оси X и Y

