

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ
УНИВЕРСИТЕТ имени академика С.П.КОРОЛЕВА
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)»

ЛОКАЛЬНЫЕ МЕТОДЫ ВЫДЕЛЕНИЯ КОНТУРОВ

*Электронные методические указания
к лабораторной работе № 2*

САМАРА

2010

Составители: СЕРГЕЕВ Владислав Викторович
ГАШНИКОВ Михаил Валерьевич
МЯСНИКОВ Владислав Валерьевич

В лабораторной работе № 2 по дисциплине «Математические методы обработки изображений» изучаются методы выделения контуров на изображении с помощью линейной обработки “окном”; приведены примеры их применения при работе с автоматизированной системой обработки изображений в режиме диалога.

Методические указания предназначены для магистров направления 010400.68 “Прикладная математика и информатика”, обучающихся по программе «Математические и компьютерные методы обработки изображений и геоинформатики».

СОДЕРЖАНИЕ

1. Теоретические основы лабораторной работы	4
1.1 Линейная локальная обработка изображений	4
1.2 Выделение контуров	7
1.2.1. Постановка задачи выделения контуров	7
1.2.2. Градиентный метод	9
1.2.3. Метод оператора Лапласа	12
1.2.4. Метод согласования	14
1.3 Алгоритм выделения контуров.	17
2. Описание используемых прикладных задач.....	19
2.1 Линейная обработка скользящим окном 3×3	19
2.2 Градиентное выделение контуров окном 3×3	20
2.3 Построение гистограммы отсчетов изображения	21
2.4 Формирование таблицы кусочно-линейного преобразования.....	21
2.5 Поэлементное преобразование изображений	21
2.6 Вычисление значения переменной	23
2.7 Визуализация изображений, гистограмм и таблиц преобразования.....	23
3. Выполнение лабораторной работы.....	24
3.1 Общий план выполнения работы	24
3.2 Порядок проведения обработки.....	24
3.3 Содержание отчета.....	25
4. Контрольные вопросы	26
Библиографический список	27

Цель работы - изучение и экспериментальное исследование методов выделения контуров на изображениях с помощью линейной обработки “окном”; получение навыков работы с автоматизированной системой обработки изображений.

1. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ЛАБОРАТОРНОЙ РАБОТЫ

1.1 Линейная локальная обработка изображений

Введенные в ЭВМ изображения часто имеют низкое качество, что затрудняет визуальный анализ и определение параметров изображенных объектов. Ухудшение качества изображений связано со многими причинами: малым динамическим диапазоном видеодатчика, несовершенством оптической части системы регистрации, плохим освещением объекта и др. Визуально такие искажения воспринимаются как дефокусировка, “размытость”, ухудшение резкости, снижение контрастности, появление мелкоструктурных помех (шумов).

Простейший метод повышения качества изображений заключается в пространственно-инвариантном поэлементном преобразовании изображений (см. лабораторную работу №1). При поэлементном преобразовании каждый элемент изображения обрабатывается независимо от других (соседних) элементов. Однако поэлементная обработка не позволяет определять на изображении **контур**ы (границы объектов), которые играют первостепенную роль в восприятии изображенных объектов и, в конечном счете, позволяют классифицировать эти объекты.

В настоящей лабораторной работе рассматривается метод локальной обработки изображений. Локальную обработку проще всего представить себе, рассмотрев перемещающееся по полю изображения “окно” (маску) небольшого размера (его варианты представлены на рис. 1), которое занимает все возможные положения.

При каждом положении “окна” формируется один отсчет (элемент яркости) выходного изображения (обычно соответствующий центру “окна”), который является функцией отсчетов исходного изображения, попавших в “окно”.

Пусть элементы яркости исходного изображения - $f(m,n)$, $0 \leq m \leq M-1$, $0 \leq n \leq N-1$. Тогда отсчеты получаемого изображения

$$e(m,n) = E[f(m+k, n+l) : (k,l) \in W], \quad (1)$$

где E - функция преобразования,

а W - область “окна” (точка $(k,l)=(0,0)$ соответствует его центру).

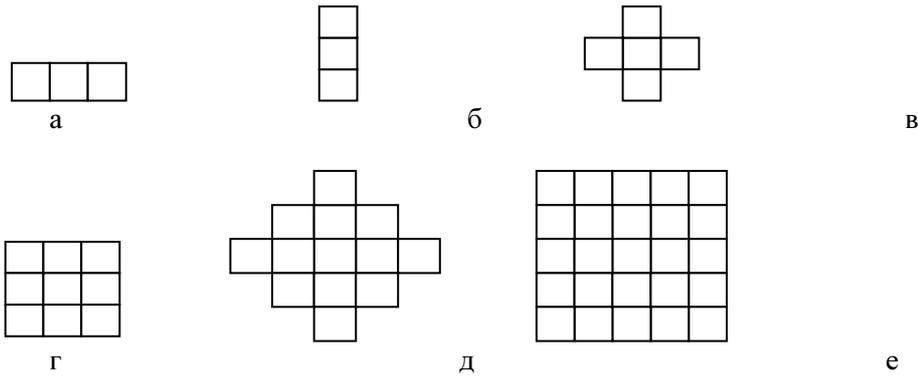


Рис. 1

Например, для окна, показанного на рис. 1г, имеем: $k=-1,0,1$ и $l=-1,0,1$, т.е.

$$e(m,n) = E \left[\begin{aligned} &f(m-1,n-1), f(m-1,n), f(m-1,n+1), \\ &f(m,n-1), f(m,n), f(m,n+1), \\ &f(m+1,n-1), f(m+1,n), f(m+1,n+1) \end{aligned} \right].$$

Из (1) следует, что поэлементное преобразование является частным случаем локальной обработки изображений, соответствующей размерам “окна” 1×1 . В общем случае функция E в (1) является нелинейной и может иметь весьма сложный вид. Ясно, что для ее реализации нельзя использовать табличный метод (как при поэлементной обработке), поскольку это потребовало бы неприемлемо больших затрат памяти. Однако для большого числа практических задач локальную обработку можно представить в виде суперпозиции одной или нескольких операций линейной обработки “окном” и поэлементного преобразования. Указанная процедура имеет вид:

$$s_i(m,n) = \sum_{(k,l) \in W} a_i(k,l) f(m-k, n-l) + C_i; i=1, \dots, I, \quad (2)$$

$$e(m,n) = H[s_1(m,n), s_2(m,n), \dots, s_I(m,n)], \quad (3)$$

где $s_i(m,n)$ - промежуточные изображения;

$a_i(k,l)$ - коэффициенты линейных масок;

C_i - коэффициенты постоянного смещения;

H - функция совместного поэлементного преобразования изображения $s_i(m,n)$;

I - число параллельных операций линейной обработки “окном”.

В частном случае может быть $I=1$. В дальнейших рассуждениях для краткости

обозначений индекс i будем опускать.

Соотношение (2) описывает двумерную свертку функции яркости исходного изображения $f(m,n)$ с импульсной характеристикой $a(k,l)$ некоторой двумерной КИХ-системы. Выбирая определенным образом размеры и коэффициенты маски $a(k,l)$, можно выполнить различные операции по улучшению качества изображения: повышение резкости, сглаживание шумов, подчеркивание контуров, низкочастотную или высокочастотную фильтрацию и т.д. Из соображений простоты обычно применяют “окна” малых размеров: 3×3 или 5×5 .

Коэффициент C в (2) не имеет принципиального значения для выполняемых операций. Он необходим для “компенсации” получающихся отрицательных значений отсчетов яркости. (Вспомним, что отсчеты изображения, как правило, кодируются в памяти ЭВМ одним байтом; функция яркости принимает целочисленные значения в диапазоне от 0 до 255).

Иногда требуется, чтобы при обработке изображения линейной маской (2) не менялось среднее значение яркости изображения (его постоянная составляющая). Для выполнения этого условия нужно, чтобы

$$\sum_{(k,l) \in W} a(k,l) = 1, \quad C = 0. \quad (4)$$

Если же нужно, чтобы среднее значение яркости после обработки произвольного изображения маской всегда было равно E , следует выбрать

$$\sum_{(k,l) \in W} a(k,l) = 0, \quad C = E. \quad (5)$$

При обработке изображения с использованием линейной маски следует учитывать возникающие “краевые эффекты”. Действительно, с практической точки зрения исходное и обработанное изображение удобно считать имеющими одинаковые размеры $M \times N$. Однако из (2) следует, что в этом случае получающееся изображение не определено вблизи краев, так как “окно” выходит за границу изображения при некоторых значениях индексов m и n . В этом случае обычно считают, что неопределенные значения элементов яркости вне поля изображения (при $m < 0$, или $n < 0$, или $m \geq M$, или $n \geq N$) равны константе (например, нулю) или значениям ближайших определенных точек.

1.2 Выделение контуров

1.2.1. Постановка задачи выделения контуров

Любое изображение в идеальном случае представляет собой набор однородных областей различной яркости (или, по меньшей мере, допускают такое представление с некоторой погрешностью). В простейшем случае бинарного (двухградационного) изображения имеем области, содержащие отсчеты двух значений яркости, которые соответствуют объектам и фону. Исследованиями психологов установлено, что с точки зрения распознавания и анализа объектов на изображении наиболее информативными являются не значения яркостей объектов, а характеристики их границ - **контуров**. Другими словами, основная информация заключена не в яркости (или цвете) отдельных областей, а в их очертаниях. Задача выделения контуров состоит в построении изображения именно границ объектов и очертаний однородных областей.

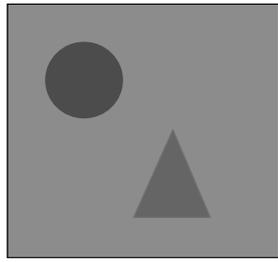
На рис. 2а,б показаны соответственно исходное изображение, состоящее из областей различной яркости, и его графический вариант, состоящий только из границ этих областей.

Будем называть контуром изображения совокупность его отсчетов, в окрестности которых наблюдается скачкообразное изменение функции яркости. Так как при цифровой обработке изображение представлено как функция целочисленных аргументов, то контуры представляются линиями шириной как минимум в один элемент. При этом может возникнуть неоднозначность в определении линии контура, как то показано на рис. 2г,д, для исходного изображения с перепадом яркости (рис. 2в).

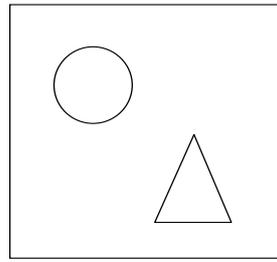
Если исходное изображение кроме областей постоянной яркости содержит участки с плавно меняющейся яркостью, то введенное определение контура остается справедливым, однако при этом не гарантируется непрерывность контурных линий: разрывы контуров будут наблюдаться в тех местах, где изменение функции яркости не является достаточно резким (этот эффект иллюстрируется на рис. 3).

С другой стороны, если на “кусочно-постоянном” изображении присутствует шум, то, возможно, будут обнаружены “лишние” контуры в точках, которые не являются границами областей.

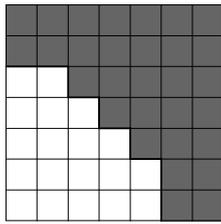
При разработке алгоритмов выделения контуров нужно учитывать указанные особенности поведения контурных линий. Специальная дополнительная обработка выделенных контуров позволяет устранять разрывы и подавлять ложные контурные линии.



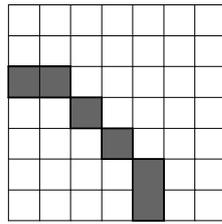
а



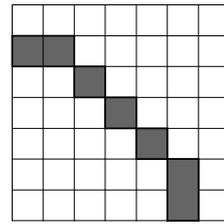
б



в



г



д

Рис.2

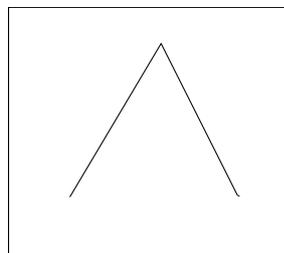
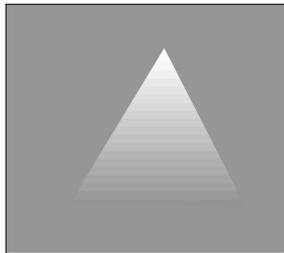


Рис.3

Общую процедуру построения бинарного изображения границ объектов иллюстрирует блок-схема, представленная на рис. 4.

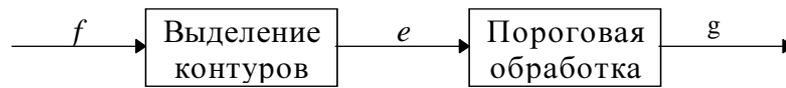


Рис. 4

Исходное изображение f подвергается линейной или нелинейной обработке, с тем чтобы выделить перепады яркости. В результате этой операции формируется изображение e , функция яркости которого существенно отличается от нуля только в областях резких изменений изображения f . Затем в результате пороговой обработки из изображения e формируется графический (контурный) препарат g . Правильный выбор порога на втором этапе должен производиться из следующих соображений:

- при слишком высоком пороге могут появиться разрывы контуров, а слабые перепады яркости не будут обнаружены;
- при слишком низком пороге из-за шумов и неоднородности областей могут появиться ложные контуры.

Других особенностей пороговая обработка не имеет. Поэтому обратим основное внимание на первую операцию - **выделение перепадов яркости (контуров)** - и рассмотрим основные методы выполнения этой операции.

1.2.2. Градиентный метод

Одним из наиболее простых способов выделения границ является пространственное дифференцирование функции яркости. Для одномерной непрерывной функции яркости $f(x)$ этот способ иллюстрирует рис. 5.

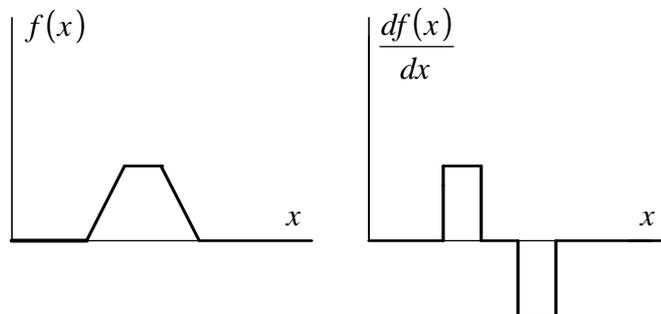


Рис. 5

Для двумерной функции яркости $f(x,y)$ (считаем пока, что координаты x, y

изменяются непрерывно) перепады в направлениях x и y регистрируются частными производными $\frac{\partial f(x,y)}{\partial x}$ и $\frac{\partial f(x,y)}{\partial y}$, которые пропорциональны скоростям изменения яркости в соответствующих направлениях. Выделение перепадов яркости в двумерном случае иллюстрирует рис. 6 (на нем серый цвет соответствует нулевому значению функции, черный - отрицательному, а отсутствие цвета - положительному). Видим, что подчеркивание контуров, перпендикулярных к оси x обеспечивает $\frac{\partial f(x,y)}{\partial x}$ (рис. 6,б), а подчеркивание контуров, перпендикулярных к оси y - $\frac{\partial f(x,y)}{\partial y}$ (рис. 6,в).

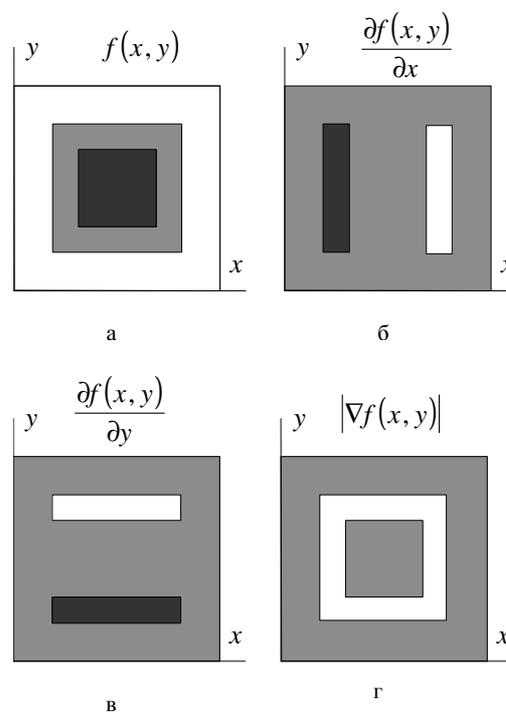


Рис.6

В практических задачах требуется выделить контуры, направление которых является произвольным. Для этих целей можно использовать **модуль градиента функции яркости**

$$|\nabla f(x,y)| = \sqrt{\left(\frac{\partial f(x,y)}{\partial x}\right)^2 + \left(\frac{\partial f(x,y)}{\partial y}\right)^2}, \quad (6)$$

который пропорционален максимальной (по направлению) скорости изменения функции яркости в данной точке и не зависит от направления контура.

Модуль градиента в отличие от частных производных принимает только

неотрицательные значения, поэтому на получающемся изображении (рисунок б,г) точки, соответствующие контурам, имеют повышенный уровень яркости. Для цифровых изображений аналогами частных производных и модуля градиента являются функции, содержащие дискретные разности, например:

$$\frac{\partial f(x,y)}{\partial x} \rightarrow \Delta_x f(m,n) = f(m,n) - f(m-1,n);$$

$$\frac{\partial f(x,y)}{\partial y} \rightarrow \Delta_y f(m,n) = f(m,n) - f(m,n-1);$$

$$\begin{aligned} |\nabla f(x,y)| &\rightarrow |\nabla f(m,n)| = \\ &= \sqrt{[f(m,n) - f(m-1,n)]^2 + [f(m,n) - f(m,n-1)]^2} \end{aligned}$$

Таким образом, операция выделения контуров заключается в выполнении нелинейной совместной поэлементной обработке изображений s_1 и s_2 (см. формулу (3)) следующего вида:

$$e(m,n) = \sqrt{s_1^2 + s_2^2}, \quad (7)$$

каждое из которых получается в результате локальной обработки исходного изображения “окнами” 2×1 и 1×2 :

$$\begin{aligned} s_1 &= f(m,n) - f(m-1,n), \\ s_2 &= f(m,n) - f(m,n-1). \end{aligned} \quad (8)$$

Для вычисления модуля градиента вместо частных производных $\frac{\partial f}{\partial x}$ и $\frac{\partial f}{\partial y}$ можно использовать производные по любой паре перпендикулярных направлений. Если в качестве этих направлений взять диагональные и применить соответствующие дискретные разности следующего вида:

$$\begin{aligned} s_1 &= f(m,n) - f(m-1,n-1), \\ s_2 &= f(m,n-1) - f(m-1,n). \end{aligned} \quad (9)$$

то формула (7) превратится в так называемый **оператор Робертса**, который также является приближением модуля непрерывного градиента (6).

При реализации процедуры выделения контуров стремятся избежать наиболее

трудоемких операций (умножение, извлечение корня), поскольку их приходится выполнять для каждой точки изображения. Для приближенного вычисления модуля градиента часто используют упрощенные выражения, например (используя формулу (8)):

$$e(m,n)=|s_1|+|s_2|; \quad (10)$$

$$e(m,n)=\max \{ |s_1|; |s_2| \}; \quad (11)$$

Однако приближения (10)-(11) уже не являются одинаково чувствительными к границам при всех их ориентациях: (10) более чувствительно к диагонально ориентированным границам, а (11) - к вертикальным и горизонтальным.

Иногда используются некоторые другие приближения градиента, например:

$$e(m,n)=\sum_{(k,l)\in W} |f(m,n)-f(m+k,n+l)|, \quad (12)$$

$$e(m,n)=\max_{(k,l)\in W} |f(m,n)-f(m+k,n+l)|, \quad (13)$$

где W - область “окна” (целесообразно использовать “окна”, показанные на рис. 1,в и 1,г).

1.2.3. Метод оператора Лапласа

В отличие от градиентного метода выделения границ, базирующегося на использовании первой производной по направлению, **метод оператора Лапласа** использует производные второго порядка. Если функция яркости $f(m,n)$ является функцией непрерывных аргументов, то **лапласиан** (оператор Лапласа) определяется следующим образом:

$$\nabla^2 f(x,y)=\frac{\partial^2 f(x,y)}{\partial x^2}+\frac{\partial^2 f(x,y)}{\partial y^2}. \quad (14)$$

В отличие от градиента лапласиан является скалярной функцией. Его применение для выделения границ основано на том, что лапласиан принимает максимальное (по абсолютной величине) значение на участках “перегибов” функции яркости.

По сравнению с градиентом лапласиан обладает недостатком, заключающимся в раздваивании границы, если она не является достаточно резкой, т.е. если функция яркости содержит участки постоянного наклона. В этом случае необходимо применять специальную дополнительную обработку полученного графического препарата для устранения раздваивания линий.

В дискретном случае оператор Лапласа может быть реализован в виде процедуры локальной линейной обработки по “окну” 3x3:

$$\begin{aligned}\frac{\partial^2 f(x,y)}{\partial x^2} &\rightarrow f(m+1,n)-2f(m,n)+f(m-1,n); \\ \frac{\partial^2 f(x,y)}{\partial y^2} &\rightarrow f(m,n+1)-2f(m,n)+f(m,n-1); \\ \nabla^2 f(x,y) &\rightarrow f(m+1,n)+f(m-1,n)+ \\ &\quad +f(m,n+1)+f(m,n-1)-4f(m,n)\end{aligned}\quad (15)$$

Так как лапласиан может принимать как положительные, так и отрицательные значения, в операторе выделения контуров следует взять его абсолютное значение:

$$\begin{aligned}e(m,n) &\rightarrow |f(m+1,n)+f(m-1,n)+ \\ &\quad +f(m,n+1)+f(m,n-1)-4f(m,n)|\end{aligned}\quad (16)$$

Другой вариант дискретного лапласиана получается, если рассмотреть взаимно перпендикулярные диагональные направления для аппроксимации вторых производных:

$$\begin{aligned}\nabla^2 f(x,y) &\rightarrow \frac{f(m+1,n+1)-f(m,n)}{\sqrt{2}} - \frac{f(m,n)-f(m-1,n-1)}{\sqrt{2}} + \\ &\quad + \frac{f(m+1,n-1)-f(m,n)}{\sqrt{2}} - \frac{f(m,n)-f(m-1,n+1)}{\sqrt{2}}.\end{aligned}\quad (17)$$

В этом случае оператор выделения контуров принимает вид:

$$\begin{aligned}e(m,n) &= \frac{1}{2} |f(m+1,n+1)+f(m+1,n-1)+ \\ &\quad +f(m-1,n+1)+f(m-1,n-1)-4f(m,n)|\end{aligned}\quad (18)$$

Множество различных дискретных лапласианов получается суммированием аппроксимаций (15) и (17) с разными весами. Например, взяв весовые коэффициенты равными 1/3 и 2/3, получаем еще один вариант оператора выделения контуров:

$$\begin{aligned}
e(m,n) = & \frac{1}{3} | f(m+1,n) + f(m-1,n) + f(m,n+1) + \\
& + f(m,n-1) + f(m+1,n+1) + f(m+1,n-1) + \\
& + f(m-1,n+1) + f(m-1,n-1) - 8f(m,n) |.
\end{aligned} \tag{19}$$

Операторам (16), (18), (19) соответствуют следующие линейные маски:

$$\begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \frac{1}{2} \begin{pmatrix} 1 & 0 & 1 \\ 0 & -4 & 0 \\ 1 & 0 & 1 \end{pmatrix}, \frac{1}{3} \begin{pmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{pmatrix}. \tag{20}$$

Заметим, что операторы Лапласа (16), (18), (19) являются практически нечувствительными к ориентации выделяемых границ, что является преимуществом по сравнению с аппроксимациями градиента (10), (11).

1.2.4. Метод согласования

Общим недостатком рассмотренных выше методов выделения границ является высокая чувствительность к шуму. Это объясняется тем, что действие дифференциальных (разностных) операторов состоит в вычислении и комбинировании разностей отсчетов в пределах “окна” малых размеров. Из теории цифровой обработки сигналов известно, что операция дифференцирования приводит к ослаблению низкочастотных составляющих пространственной функции яркости и к усилению высокочастотных составляющих. В то же время высокочастотная составляющая функции яркости в большой степени обусловлена именно действием шума. Поэтому операторы градиента и лапласиана усиливают шум, что ведет к появлению ложных контуров.

Чтобы повысить помехоустойчивость рассмотренных методов, перед выполнением операции дифференцирования производят **сглаживание** функции яркости в пределах “окна”, например, “подгоняют” или согласуют с функцией яркости выбранную поверхность первого или второго порядков. Такой подход приводит к методу согласования.

Рассмотрим принцип метода согласования на примере “окна” 3x3. По наблюдаемым значениям функции яркости в пределах этого “окна”

$$\{f(m,n), f(m,n-1), f(m-1,n), f(m-1,n-1)\}$$

построим аппроксимирующую плоскость

$$\bar{f}(x,y) = \alpha x + \beta y + \gamma. \tag{21}$$

Определив коэффициенты α, β, γ , можно вычислить частные производные и

искомый модуль градиента:

$$\frac{\partial \bar{f}(x, y)}{\partial x} = \alpha, \quad \frac{\partial \bar{f}(x, y)}{\partial y} = \beta, \quad |\nabla \bar{f}(x, y)| = \sqrt{\alpha^2 + \beta^2}.$$

Неизвестные коэффициенты можно определить по методу наименьших квадратов, найдя минимум функционала:

$$\bar{\varepsilon}^2 = \sum_{k=0}^1 \sum_{l=0}^1 [\bar{f}(m-k, n-l) - f(m-k, n-l)]^2 \rightarrow \min_{\alpha, \beta, \gamma}.$$

В результате этой минимизации получаем:

$$\alpha = \frac{f(m, n) + f(m, n-1)}{2} - \frac{f(m-1, n) + f(m-1, n-1)}{2}, \quad (22)$$

$$\beta = \frac{f(m, n) + f(m-1, n)}{2} - \frac{f(m, n-1) + f(m-1, n-1)}{2}. \quad (23)$$

Несложно видеть, что коэффициенты α и β формируются при линейной обработке изображения в “окне” 2x2 соответственно масками

$$\frac{1}{2} \begin{pmatrix} -1 & -1 \\ 1 & 1 \end{pmatrix}, \quad \frac{1}{2} \begin{pmatrix} -1 & 1 \\ -1 & 1 \end{pmatrix}, \quad (24)$$

то есть могут быть интерпретированы как промежуточные изображения s_1 и s_2 в формуле (3).

Частные дискретные разности, рассчитанные по (22) и (23), являются менее чувствительными к помехам, так как эти разности считаются по усредненным (по горизонтали или вертикали) значениям двух соседних отсчетов изображения. Такое усреднение (сглаживание) перед вычислением дискретной разности приводит к уменьшению шума.

Рассмотренный метод построения аппроксимирующей плоскости допускает очевидное обобщение на “окно” произвольных размеров $K \times L$. Следует, однако, заметить, что размеры “окна” должны быть меньше размеров любой области постоянной яркости на изображении. Иначе вместе со сглаживанием шумов будут “сглаживаться” (и исчезать!) мелкие объекты на изображении.

При аппроксимации плоскостью (21) отсчетов в “окне” размером 3x3, по аналогии с предыдущим случаем можно получить следующие маски для формирования

изображений s_1 и s_2 :

$$\frac{1}{6} \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}, \frac{1}{6} \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix}. \quad (25)$$

Вычисление модуля градиента $\sqrt{s_1^2 + s_2^2}$ с использованием масок (25) определяет так называемый **оператор Превитт**, который часто используется на практике. Здесь также можно применить описанные ранее аппроксимации модуля градиента (10) или (11).

На “окне” размером 3×3 можно построить и аппроксимирующую поверхность второго порядка:

$$\bar{f}(x, y) = \alpha x^2 + \beta y^2 + \gamma xy + \delta x + \zeta y + \xi. \quad (26)$$

Тогда величина лапласиана по методу согласования будет равна

$$\nabla^2 \bar{f}(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2} = 2\alpha + 2\beta,$$

где коэффициенты α и β формируются масками:

$$\frac{1}{6} \begin{pmatrix} 1 & 1 & 1 \\ -2 & -2 & -2 \\ 1 & 1 & 1 \end{pmatrix}, \frac{1}{6} \begin{pmatrix} 1 & -2 & 1 \\ 1 & -2 & 1 \\ 1 & -2 & 1 \end{pmatrix}.$$

Очевидно, дискретный лапласиан может быть вычислен при помощи единственной маски, являющейся удвоенной суммой указанных масок. Эта маска имеет вид:

$$\frac{1}{3} \begin{pmatrix} 2 & -1 & 2 \\ -1 & -4 & -1 \\ 2 & -1 & 2 \end{pmatrix}. \quad (27)$$

На примере лапласиана продемонстрируем, что метод согласования обеспечивает большую помехоустойчивость процедуры выделения контуров по сравнению с методами непосредственного вычисления лапласиана с использованием масок (20). В качестве критерия помехоустойчивости будем использовать коэффициент Q усиления мощности белого шума, который показывает, во сколько раз увеличивается дисперсия аддитивного белого шума при обработке изображения “окном” с линейной маской. Несложно получить, что

$$Q = \sum_{(k,l) \in W} a^2(k,l),$$

где $a(k,l)$ - коэффициенты линейной маски (2).

$$\text{Для "традиционного" лапласиана с маской } \begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix} Q=20.$$

$$\text{Для "диагонального" лапласиана с маской } \frac{1}{2} \begin{pmatrix} 1 & 0 & 1 \\ 0 & -4 & 0 \\ 1 & 0 & 1 \end{pmatrix} Q=5.$$

$$\text{Для "комбинированного" лапласиана с маской } \frac{1}{3} \begin{pmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{pmatrix} Q=8.$$

Для "согласованного" лапласиана с маской (27) $Q=4$, то есть по сравнению с "традиционным" лапласианам, помехоустойчивость повышается в 5 раз.

1.3 Алгоритм выделения контуров.

При реализации алгоритмов выделения контуров необходимо учитывать некоторые особенности программного обеспечения автоматизированной системы обработки изображений. Эти особенности связаны прежде всего с форматом представления данных (отсчетов функции яркости). Большинство программ пакета ориентировано на работу с изображениями, каждый элемент которых может принимать целочисленные значения из диапазона 0...255. Однако значения лапласиана и проекций градиента принимают и положительные, и отрицательные значения. Для того чтобы не происходило обнуление отрицательных значений дифференциальных операторов, следует предпринять специальные меры.

С этой целью в операторе локальной линейной обработки (2) введен коэффициент постоянного смещения C . Если выбрать $C=128$, то отрицательные значения лапласиана и проекций градиента будут отображаться на отрезок $[0,127]$, а положительные - на $[129,255]$. Для получения абсолютных значений дифференциальных операторов можно затем воспользоваться соответствующим поэлементным преобразованием, причем при использовании градиентов метода в форме (10) или (11) указанную операцию нужно выполнить дважды - для каждой проекции градиента, а затем найти максимум или сумму полученных полей. Впрочем, в используемой программной системе имеется функция, осуществляющая преобразование входного поля сразу по формуле (7) и ее обобщением.

Следующая операция после выделения контуров - **пороговая обработка**. Чтобы получить светлые линии контуров на темном фоне, нужно использовать функцию поэлементного преобразования, изображенную на рис. 7,г. Операции нахождения абсолютной величины и пороговой обработки можно объединить, если используется метод оператора Лапласа. Для этого можно воспользоваться функцией поэлементного преобразования, представленной на рис. 7,б.

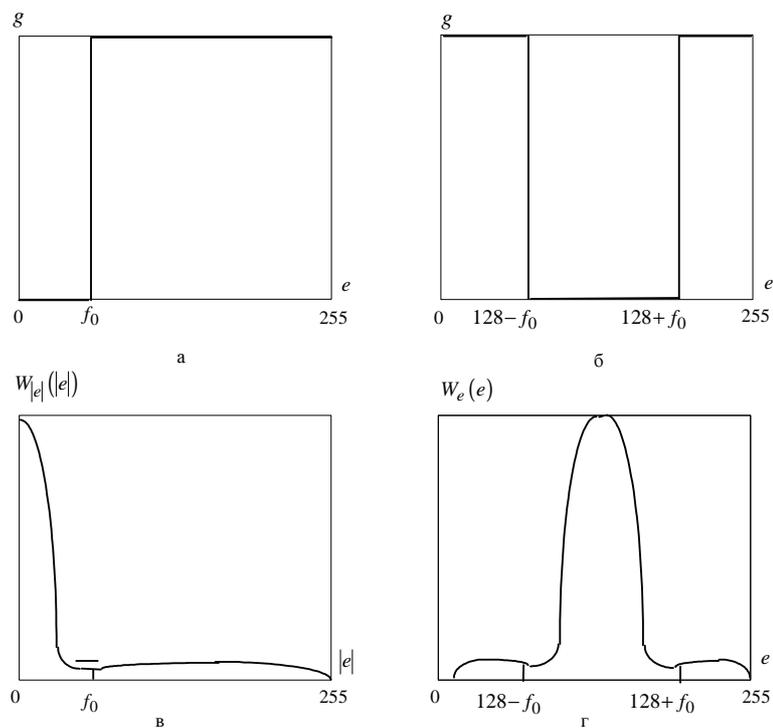


Рис.7

Проблема заключается в выборе порога f_0 . Его можно определить, проанализировав гистограмму изображения e , полученного в результате обработки исходного изображения с помощью градиентного метода или лапласиана (рисунок 7 в,г). Если на гистограмме нет явно выраженных максимумов, соответствующих границам и внутренним областям изображенных объектов, то необходимо выбрать порог экспериментально, визуально контролируя качество получающегося изображения. Заметим, что при использовании масок (20), (24), (25), (27) их постоянные множители можно взять единичными, так как они влияют лишь на масштаб гистограмм и, в конечном счете, только на величину порога f_0 .

2. ОПИСАНИЕ ИСПОЛЬЗУЕМЫХ ПРИКЛАДНЫХ ЗАДАЧ

В данной лабораторной работе используется система обработки изображений *ResLook*. Необходимые сведения о системе *ResLook* содержатся в описании системы *ResLook*. Далее приводится необходимая информация о некоторых прикладных модулях системы *ResLook*, необходимых для выполнения данной лабораторной работы. Все прикладные модули системы *ResLook* вызываются в режиме командной строки с помощью вспомогательной программы *RL.exe*. Формат вызова:

RL <Имя_DLL_Содержащей_Модуль> <Имя_Модуля>

2.1 Линейная обработка скользящим окном 3×3

Обращение к модулю:

RL Local3 LinFil3 *ImgIn* *w11* *w12* *w13* *w21* *w22* *w23* *w31* *w32* *w33* *ImgOut*
 [*MulConst*=<значение>] [*DivConst*=<значение>] [*AddConst*=<значение>]

Модуль осуществляет преобразование вида:

$$Y(i, j) = (X(i-1, j-1) \cdot w11 + X(i-1, j) \cdot w12 + X(i-1, j+1) \cdot w13 + \\ + X(i, j-1) \cdot w21 + X(i, j) \cdot w22 + X(i, j+1) \cdot w23 + \\ + X(i+1, j-1) \cdot w31 + X(i+1, j) \cdot w32 + X(i+1, j+1) \cdot w33) \cdot \frac{MulConst}{DivConst} + \\ + AddConst,$$

где $X(i, j)$, $Y(i, j)$ - отсчеты входного и выходного изображений,

$w11 \dots w33$, $MulConst$, $DivConst$, $AddConst$ – задаваемые параметры.

Обязательные параметры

Имя	Описание	Тип	Пример значения
<i>ImgIn</i>	Входное изображение	Строка	FileImgIn
<i>w11</i>	Весовой коэффициент (1,1) окна 3×3	Вещественный	0.33
<i>w12</i>	Весовой коэффициент (1,2) окна 3×3	Вещественный	0.33
<i>w13</i>	Весовой коэффициент (1,3) окна 3×3	Вещественный	0.33
<i>w21</i>	Весовой коэффициент (2,1) окна 3×3	Вещественный	0.33
<i>w22</i>	Весовой коэффициент (2,2) окна 3×3	Вещественный	0.33
<i>w23</i>	Весовой коэффициент (2,3) окна 3×3	Вещественный	0.33
<i>w31</i>	Весовой коэффициент (3,1) окна 3×3	Вещественный	0.33
<i>w32</i>	Весовой коэффициент (3,2) окна 3×3	Вещественный	0.33
<i>w33</i>	Весовой коэффициент (3,3) окна 3×3	Вещественный	0.33
<i>ImgOut</i>	Выходное изображение	Строка	FileImgOut

Необязательные параметры

Имя	Описание	Тип	Атрибут	По умолчанию
<i>AddConst</i>	Аддитивная константа	Вещественный	Входной	0
<i>MulConst</i>	Мультипликативная константа	Вещественный	Входной	1
<i>DivConst</i>	Делитель	Вещественный	Входной	1

2.2 Градиентное выделение контуров окном 3×3

Обращение к модулю:

RL Local3 Gradi3 ImgIn w11 w12 w13 w21 w22 w23 w31 w32 w33 ImgOut

Обязательные параметры

Имя	Описание	Тип	Пример значения
<i>ImgIn</i>	Входное изображение	Строка	ImgInFile
<i>w11</i>	Весовой коэффициент (1,1) окна 3×3	Вещественный	0.33
<i>w12</i>	Весовой коэффициент (1,2) окна 3×3	Вещественный	0.33
<i>w13</i>	Весовой коэффициент (1,3) окна 3×3	Вещественный	0.33
<i>w21</i>	Весовой коэффициент (2,1) окна 3×3	Вещественный	0.33
<i>w22</i>	Весовой коэффициент (2,2) окна 3×3	Вещественный	0.33
<i>w23</i>	Весовой коэффициент (2,3) окна 3×3	Вещественный	0.33
<i>w31</i>	Весовой коэффициент (3,1) окна 3×3	Вещественный	0.33
<i>w32</i>	Весовой коэффициент (3,2) окна 3×3	Вещественный	0.33
<i>w33</i>	Весовой коэффициент (3,3) окна 3×3	Вещественный	0.33
<i>ImgOut</i>	Выходное изображение	Строка	ImgOutFile

Прикладной модуль осуществляет приближенное вычисление модуля градиента изображения по формуле:

$$Y(i, j) = \sqrt{A^2 + B^2},$$

где

$$A = (X(i-1, j-1) \cdot w11 + X(i-1, j) \cdot w12 + X(i-1, j+1) \cdot w13 + \\ + X(i, j-1) \cdot w21 + X(i, j) \cdot w22 + X(i, j+1) \cdot w23 + \\ + X(i+1, j-1) \cdot w31 + X(i+1, j) \cdot w32 + X(i+1, j+1) \cdot w33),$$

$$B = (X(i-1, j-1) \cdot w11 + X(i-1, j) \cdot w21 + X(i-1, j+1) \cdot w31 + \\ + X(i, j-1) \cdot w12 + X(i, j) \cdot w22 + X(i, j+1) \cdot w32 + \\ + X(i+1, j-1) \cdot w13 + X(i+1, j) \cdot w23 + X(i+1, j+1) \cdot w33),$$

где $X(i, j)$, $Y(i, j)$ - отсчеты входного и выходного изображений.

2.3 Построение гистограммы отсчетов изображения

Обращение к модулю:

RL Stat Hst ImgIn HstFile

Модуль обрабатывает только байтовые изображения, для них строится гистограмма в диапазоне значений пикселей 0÷255 и сохраняется в файл с расширением .HST, который потом используется модулями *TableEq*, *StatHst*, *Quantil..*

Обязательные параметры

Имя	Описание	Тип	Пример значения
<i>ImgIn</i>	Входное изображение	Строка	<i>ImgInFile</i>
<i>HstFile</i>	Имя файла, в который будет помещена гистограмма изображения	Строка	<i>FileHst</i>

2.4 Формирование таблицы кусочно-линейного преобразования

Обращение к модулю:

RL Table TablePL ArrayX_Y TabFile

Модуль формирует файл с расширением .TAB, содержащий таблицу кусочно-линейного преобразования для последующего использования модулем *CalcI*. Таблица строится по заданным узловым точкам.

Обязательные параметры

Имя	Описание	Тип	Пример значения
<i>ArrayX_Y</i>	Массив, содержащий узловые точки кусочно-линейного преобразования в виде $x_1, y_1, x_2, y_2, \dots$	Массив	20,0,200,255
<i>TabFile</i>	Имя файла, в который будет помещена таблица	Строка	<i>FileTab</i>

2.5 Поэлементное преобразование изображений

Обращение к модулю:

RL Clc CalcI Expression [BPSOut=<значение>]

Модуль формирует новое изображение из входных изображений и переменных в соответствии с заданным *аналитическим выражением (Expression)*.

При записи *аналитического выражения* можно использовать:

математические операции: "+" - сложение, "-" - вычитание, "*" - умножение, "/" - деление, ^ - возведение в степень;

функции: \sin – синус (аргумент считается заданным в радианах), \cos - косинус, \tan - тангенс, atan – арктангенс (результат выражается в радианах), sqrt – квадратный корень, \log – натуральный логарифм, exp – экспонента, $\text{rand}(x)$ – генерация случайной величины в диапазоне $0 \div 32767$ (" x " - начальное значение);

вспомогательные функции:

$$\text{bin}(x) = \begin{cases} 0, & \text{при } x \leq 0; \\ 1, & \text{при } x > 0, \end{cases} \quad \text{sign}(x) = \begin{cases} -1, & \text{при } x < 0; \\ 0, & \text{при } x = 0; \\ 1, & \text{при } x > 0, \end{cases}$$

$$\text{byte}(x) = \begin{cases} 0, & \text{при } x \leq 0; \\ [x], & \text{при } 0 < x < 255; \\ 255, & \text{при } x \geq 255, \end{cases}$$

где $[x]$ - целая часть числа x .

Кроме того, в качестве имени функции может выступать имя файла – таблицы, созданного при помощи модулей *TablePL*, *TableEq*. В качестве аргументов функций могут выступать константы, имена изображений и переменные, записанные в файл параметров "params.mem" (имена переменных нужно предварять символом "\$").

Пример аналитического выражения:

$$\text{ImgA} = 15 * \sin(\text{ImgB}) * \text{Tabl}(\text{ImgC}) - \$\text{VarName} * \text{sqrt}(\text{exp}(\$x^2 + \$y^2))$$

Параметр *BPSOut* задает количество бит на один отсчет одной цветовой компоненты выходного изображения. Возможные значения 8,16,32,0. Если указать 0, то количество бит будет выбрано автоматически и равно максимальному из встречающихся в правой части выражения *Expression*. Например, если в правой части есть 8-битное и 16-битное изображения, то результат будет 16-битным.

Обязательные параметры

Имя	Описание	Тип	Пример значения
<i>Expression</i>	Аналитическое выражение	Строка	$\text{img2} = 2 * \text{img1} + \var

Необязательные параметры

Имя	Описание	Тип	Атрибут	По умолчанию
<i>BPSOut</i>	Количество бит на отсчет для выходного изображения	Целый	Входной	0

2.6 Вычисление значения переменной

Обращение к модулю:

RL Clc CalcV Expression

Модуль вычисляет значение переменной, имя которой задано в левой части *аналитического выражения (Expression)*. Аналитическое выражение формируется по тем же правилам, что и для модуля *CalcI* (см. предыдущий раздел). Отличие заключается в том, что результатом *аналитического выражения* должно быть число, а не изображение.

Имя переменной в левой части *аналитического выражения* не предваряется знаком "\$", имена переменных в правой части *аналитического выражения* предваряются знаком "\$" (как и для модуля *CalcI*):

*ResultVariable=\$MyVariable*10*

Обязательные параметры

Имя	Описание	Тип	Пример значения
<i>Expression</i>	Аналитическое выражение	Строка	<i>var2=3.14*\$var1+10</i>

2.7 Визуализация изображений, гистограмм и таблиц преобразования

Обращение к модулю:

Vis Time Name1 Name2 Name3 ...

Параметры:

Time - время демонстрации на экране (в секундах),

Name1 Name2 Name3 ... - имена визуализируемых файлов.

Выведенная на монитор информация сохраняется в течение времени *Time* или до первого нажатия любой клавиши. Все файлы отображаются в окнах одинакового размера. Весь экран равномерно распределяется в соответствии с числом файлов так, чтобы изображения были близки к квадратным.

Имена файлов указываются без расширений. Если существуют файлы с одинаковыми именами и разными расширениями, то отображение производится в соответствии с приоритетом. Наибольший приоритет имеют файлы с расширением *.PIC*, затем - с расширением *.HST*, *.TAB* (в порядке убывания приоритетов). Если файл с заданным именем отсутствует, то соответствующее место на мониторе останется пустым.

3. ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

3.1 Общий план выполнения работы

1. Изучить методы локальной линейной обработки изображений, используемых для выделения контуров.
2. Ознакомиться с описанием программного обеспечения системы обработки изображений ResLook.
3. Составить последовательность команд с указанием значений параметров и с комментариями для решения задачи выделения контуров следующими методами:
 - 1) метод простого градиента (соотношение (7));
 - 2) метод аппроксимации лапласиана (маски (20));
 - 3) метод оператора Превитт (маски (25));
 - 4) метод согласования для лапласиана (маска (27));
4. Согласовать план проведения обработки с преподавателем, получить разрешение на практическую работу.
5. Выполнить обработку.
6. Составить отчет о выполненной работе.
7. Сдать отчет преподавателю, ответить на контрольные вопросы, получить зачет по работе.

3.2 Порядок проведения обработки

Для заданного изображения выполнить операцию выделения контуров при помощи обработки "окном" с маской, соответствующей выбранному методу. Вывести на экран промежуточное изображение, полученное в результате обработки "окном".

Рассчитать гистограмму промежуточного изображения с выделенными контурами, вывести ее на экран, произвести выбор порога по гистограмме.

Построить функцию поэлементного преобразования для пороговой обработки и вывести ее на экран.

Провести пороговую обработку промежуточного изображения и вывести на экран результат обработки.

При необходимости повторить выбор порога и пороговую обработку с новым порогом.

3.3 Содержание отчета

Отчет по работе должен содержать:

1. Последовательность команд обработки с комментариями и числовыми значениями параметров.
2. Перечень созданных изображений с указанием их смыслового содержания.
3. Зарисованные графики гистограмм и функций для поэлементных преобразований.
4. Выводы по результатам исследования.

4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Метод локальной линейной обработки. Виды “окон”.
2. Понятие контура. Этапы решения задачи выделения контуров.
3. Градиентный метод выделения контуров и его варианты.
4. Метод оператора Лапласа, его достоинства и недостатки.
5. Выделение контуров методом согласования.
6. Помехоустойчивость алгоритмов выделения контуров.
7. Как выбрать порог для получения изображения контуров?
8. Общая характеристика программного обеспечения

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Анисимов Б.В., Курганов В.Д., Злобин В.К. *Распознавание и цифровая обработка изображений* (Москва: Высшая школа, 1983)
2. Гонсалес Р., Вудс Р. *Цифровая обработка изображений* (Москва: Техносфера, 2005)
3. *Методы компьютерной обработки изображений* // Под редакцией В.А. Сойфера. (Москва: Физматлит, 2003)
4. Павлидис Т. *Алгоритмы машинной графики и обработка изображений* (Москва: Радио и связь, 1988)
5. Прэтт У.К. *Цифровая обработка изображений* (Москва: Мир, 1982, 2 т.)
6. Рабинер Р., Гоулд Б. *Теория и применение цифровой обработки сигналов* (Москва: Мир, 1978)
7. Ярославский Л.П. *Введение в цифровую обработку изображений* (Москва: Советское радио, 1979)
8. Ярославский Л.П. *Цифровая обработка сигналов в оптике и голографии: Введение в цифровую оптику* (Москва: Радио и связь, 1987)