

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

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

А.А. Федотов, С.А. Акулов, А.С. Акулова

МЕТОДЫ КОМПЬЮТЕРНОЙ ОБРАБОТКИ БИОМЕДИЦИНСКИХ ИЗОБРАЖЕНИЙ В СРЕДЕ MATLAB

Рекомендовано редакционно-издательским советом федерального государственного автономного образовательного учреждения высшего образования «Самарский государственный аэрокосмический университет имени академика С.П. Королева (национальный исследовательский университет)» в качестве учебного пособия для студентов, обучающихся по программе высшего образования по направлению подготовки магистров 12.04.04 Биотехнические системы и технологии

Самара
Издательство СГАУ
2015

УДК 004.9(075)
ББК 32.973я7
Ф 342

Рецензенты: д-р техн. наук, доц. В. С. Л и м а н о в а
канд. техн. наук, доц. И. А. К у д р я в ц е в

Федотов А.А.

Ф342 **Методы компьютерной обработки биомедицинских изображений в среде MATLAB:** учеб. пособие / *А.А. Федотов, С.А. Акулов, А.С. Акулова.* – Самара: Изд-во СГАУ, 2015. – 88 с.

ISBN 978-5-7883-1011-4

Учебное пособие содержит информацию об основных методах получения биомедицинских изображений, общие сведения о цифровой обработке изображений, включает в себя подробный анализ существующих методик пространственной и частотной фильтрации изображений, а также обзор актуальных методов восстановления изображений.

Предназначено для студентов, обучающихся по направлению 12.04.04 Биотехнические системы и технологии.

Разработано на кафедре лазерных и биотехнических систем.

УДК 004.9(075)
ББК 32.973я7

ISBN 978-5-7883-1011-4

© СГАУ, 2015

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	5
ГЛАВА 1 Основы цифровой обработки медицинских изображений в среде MATLAB	6
1.1 Общие сведения о цифровых изображениях.....	6
1.2 Методы улучшения изображений.....	8
1.3 Основные сведения о системе компьютерных вычислений MATLAB.....	12
1.4 Основы цифровой обработки изображений в среде MATLAB.....	22
1.5 Практическое задание № 1.....	26
ГЛАВА 2 Пространственная фильтрация медицинских изображений в среде MATLAB	28
2.1 Линейная пространственная фильтрация изображений.....	28
2.2 Сглаживающие пространственные фильтры.....	29
2.3 Пространственные фильтры повышения резкости изображений.....	31
2.4 Нерезкое маскирование и фильтрация с подъемом высоких частот.....	34
2.5 Комбинированные методы пространственного улучшения изображений.....	37
2.6 Реализация методов пространственной фильтрации в среде MATLAB.....	37
2.7 Практическое задание № 2.....	41
ГЛАВА 3 Частотная фильтрация медицинских изображений в среде MATLAB	43
3.1 Введение в Фурье-анализ.....	43
3.2 Основы частотной фильтрации.....	44
3.3 Основные частотные фильтры и их свойства.....	46
3.4 Сглаживающие частотные фильтры.....	47
3.5 Частотные фильтры повышения резкости.....	53
3.6 Нерезкое маскирование. Высокочастотная фильтрация с подъемом частотной характеристики.....	56
3.7 Реализация методов частотной фильтрации в среде MATLAB.....	58
3.8 Практическое задание № 3.....	60
ГЛАВА 4 Восстановление медицинских изображений в среде MATLAB	62
4.1 Модель процесса искажения/восстановления изображения.....	62
4.2 Модели шума.....	63
4.3 Построение оценок для параметров шумов.....	68
4.4 Подавление шумов методами пространственной фильтрации.....	69
4.5 Инверсная фильтрация.....	72

4.6 Винеровская фильтрация.....	73
4.7 Восстановление изображений в среде MATLAB.....	75
4.8 Практическое задание № 4.....	77
КОНТРОЛЬНЫЕ ВОПРОСЫ ДЛЯ САМОСТОЯТЕЛЬНОЙ АТТЕСТАЦИИ.....	79
ЗАКЛЮЧЕНИЕ.....	85
БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....	86

ВВЕДЕНИЕ

Стремительное развитие новых технологий и инструментальных средств диагностической визуализации обусловлено современными потребностями человечества в появлении новых систем и методов, расширяющих возможности клинического мониторинга и улучшающих качество жизни людей.

Значительное повышение технического уровня развития современных неинвазивных диагностических систем за счет совершенствования аппаратной реализации и технологий производства делает системы диагностической визуализации незаменимыми в повседневной клинической практике. При этом наряду с прогрессом развития инструментальных средств, весьма существенную роль в настоящее время начинают играть компьютерные методы обработки графической информации. Современные методики компьютерной обработки биомедицинских изображений обеспечивают улучшение изображений для их наилучшего визуального восприятия врачом-диагностом, эффективное сжатие изображений – для надежного хранения и быстрой передачи данных по каналам связи.

Учебное пособие содержит информацию о практической реализации основных методов цифровой обработки биомедицинских изображений в среде компьютерных вычислений MATLAB, общие теоретические сведения о цифровом представлении изображений, описание основных методов пространственной и частотной фильтрации биомедицинских изображений, а также обзор актуальных методов восстановления изображений. Каждая глава учебного пособия содержит практическое задание, посвященное цифровой обработке реальных биомедицинских изображений, а также список контрольных вопросов для самопроверки.

ГЛАВА 1 ОСНОВЫ ЦИФРОВОЙ ОБРАБОТКИ МЕДИЦИНСКИХ ИЗОБРАЖЕНИЙ В СРЕДЕ MATLAB

1.1 Общие сведения о цифровых изображениях

Изображение представляет собой двумерную функцию вида $f(x, y)$, значение функции f в точке с пространственными координатами (x, y) является положительной скалярной величиной, физический смысл которой определяется источником изображения. Большинство медицинских изображения являются монохромными (черно-белыми), и их значения находятся в некотором диапазоне яркостей.

Цифровое изображение формируется на основе данных, воспринимаемых чувствительным элементом (ЧЭ). Выходной сигнал ЧЭ чаще всего представляет собой непрерывно меняющееся напряжение. Чтобы получить цифровое изображение, необходимо преобразовать непрерывно меняющийся сигнал в цифровую форму. Эта операция включает в себя два процесса – дискретизацию и квантование. Представление координат в виде конечного множества отсчетов называется *дискретизацией*, а представление амплитуды (то есть яркости) дискретными значениями из конечного множества – *квантованием*.

В результате операций квантования и дискретизации координаты (x, y) становятся дискретными значениями, само изображение представлено в виде матрицы размером $M \times N$, за начало координат принимают верхний левый угол. Каждый элемент такой матрицы называется пикселем. На рис. 1.1 размещена система координат для представления цифровых изображений.

Для выполнения процесса оцифровки изображения необходимо принять решения относительно значений M и N , а также числа уровней (градаций) яркости L , разрешенных для каждого пикселя. Значение L по соображениям удобства построения оборудования для обработки, хранения и дискретизации принято выбирать равным целочисленной степени двойки $L = 2^k$. Величина k определяет так называемую «битность» изображения. Принято говорить о 8-битном, 16-битном или 32-битном изображениях.

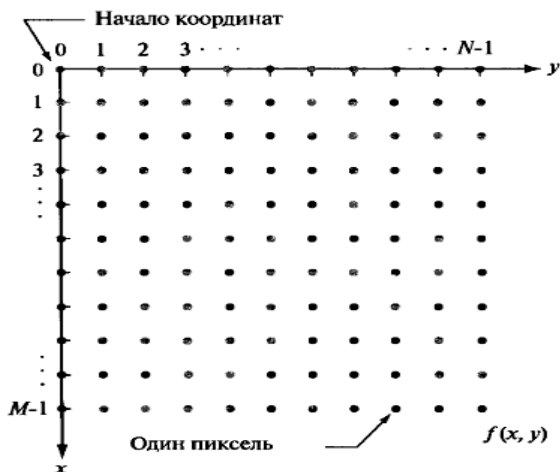


Рис. 1.1. Система представления цифровых изображений

Дискретные уровни яркости расположены с постоянным шагом, то есть используется равномерное квантование, и принимают целые значения в интервале от 0 до $L - 1$, при этом значение $L = 0$ принимается за уровень черного, а $L = L - 1$ — за уровень белого. Интервал значений яркости иногда называют динамическим диапазоном изображения. Если в изображении заметная доля пикселей занимает значительную часть всего диапазона уровней серого, то данное изображение обладает высоким контрастом. Изображения с малым динамическим диапазоном выглядят тусклыми, размытыми и серыми.

Принято разделять пространственное и яркостное разрешения изображений. Пространственное разрешение определяется дискретизацией и определяет размер мельчайших различимых деталей на изображении. Яркостным или полутоновым разрешением называется мельчайшее различимое изменение яркости, что по определению является субъективным. При выборе числа градаций яркости учитывают особенности аппаратуры, наиболее часто выбирают 8-битное или 16-битное представление. Оптимальный выбор уровня градаций яркости и пространственного разрешения изображения обусловлен уровнем детализации самого изображения, областью применения и дальнейшей обработки изображения.

1.2 Методы улучшения изображений

Под методами улучшения изображения подразумевается осуществление таких преобразований над исходным изображением, которые приводят к получению результата, более подходящего с точки зрения конкретного применения. Визуальное оценивание качества изображения представляет собой крайне субъективный процесс. В том случае, когда целью обработки изображения является их дальнейшее использование в системах машинного восприятия, критерием эффективности обработки исходного изображения является получение более точных результатов машинного распознавания.

Множество подходов к улучшению изображения распадается на 2 категории: методы обработки в пространственной области и методы обработки в частотной области. Термин пространственная область относится к плоскости изображения как таковой, и данная категория объединяет подходы, основанные на прямом манипулировании пикселями изображения.

Пространственные методы представляют собой процедуры, оперирующие непосредственно значениями пикселей, и описываются уравнением:

$$g(x, y) = T[f(x, y)],$$

где $f(x, y)$ – входное изображение, $g(x, y)$ – обработанное изображение, T – оператор над f , определенный в некоторой окрестности точки (x, y) . Под окрестностью точки понимается квадратная или прямоугольная область, являющаяся подмножеством изображения и центрированная относительно данной точки.

Простейшая форма оператора T достигается в случае, когда окрестность имеет размер в один пиксель, в этом случае значение g зависит только от значения f в точке (x, y) и T становится *функцией градиационного преобразования*. Оператор T часто называют также функцией преобразования интенсивностей или функцией отображения и записывают в виде: $s = T(r)$. Переменные r и s представляют собой значения яркостей изображений $f(x, y)$ и $g(x, y)$ в точке (x, y) . Если $T(r)$ имеет вид как на рис. 1.2а, то эффект от такого преобразования заключается в получении изображения более высокого кон-

траста по сравнению с оригиналом. Данное преобразование известно как *усиление контраста* и относится к группе методов поэлементной обработки изображений.

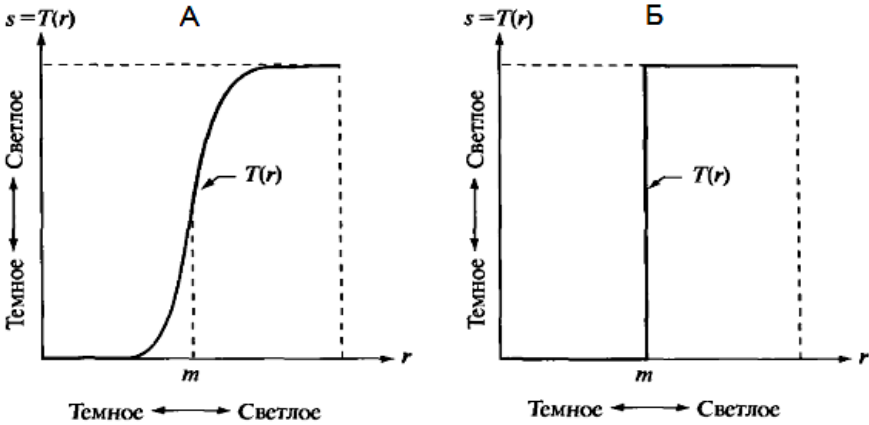


Рис. 1.2. Градационное преобразование для улучшения контраста

В предельном случае, показанном на рис. 1.2б, оператор $T(r)$ обеспечивает двухградационное или бинарное изображение. Отображение такой формы называют пороговой функцией.

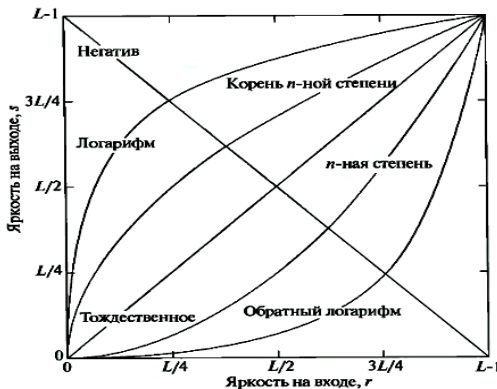


Рис. 1.3. Основные функции градационных преобразований

Градационные преобразования классифицируются на следующие группы методов: линейные (негатив и тождественное преобразование), логарифмические и степенные (рис. 1.3).

Преобразование изображения в негатив

Преобразование изображения в негатив с яркостями в диапазоне $[0, L - 1]$ осуществляется с использованием негативного преобразования, определяемого выражением:

$$S = L - I - r.$$

Данный тип обработки подходит для усиления белых или серых деталей на фоне темных областей изображения, особенно когда темные области имеют преобладающие размеры.

Логарифмические преобразования

Общий вид логарифмического преобразования выражается формулой:

$$S = c \cdot \log(I+r).$$

Данный вид преобразований отображает узкий диапазон малых значений яркостей на исходном изображении в более широкий диапазон выходных значений. Для больших значений входного сигнала верно обратное утверждение.

Степенные преобразования

Степенные преобразования имеют вид:

$$S = c \cdot r^\gamma,$$

где c и γ – положительные константы.

Заметим, что амплитудная характеристика многих устройств, используемых для ввода, печати или визуализации изображений, соответствует степенному закону. Процедура, используемая для коррекции такой степенной характеристики, называется гамма-коррекцией. Правильное воспроизведение цветов также требует применения методов гамма-коррекции. Также степенные преобразования могут применяться для улучшения контрастов: уменьшения или увеличения яркости изображений. При $\gamma > 1$ яркость изображения уменьшается, степенные преобразования в этом случае применяются для устранения слишком светлых участков изображения, при $\gamma < 1$ яркость изображения увеличивается.

Гистограмма цифрового изображения

Гистограммой цифрового изображения с уровнями яркости в диапазоне $[0, L - 1]$ называется дискретная функция $h(r_k) = n_k$, где r_k

есть k -й уровень яркости, а n_k – число пикселей на изображении, имеющих яркость r_k . Обычно гистограмма нормализуется путем деления каждого из ее значений на общее число пикселей в изображении, обозначаемое n . Таким образом, значения нормализованной гистограммы будут определяться как: $p(r_k) = n_k/n$, для $k = 0, 1 \dots L - 1$, при этом $p(r_k)$ представляет собой вероятность появления пикселя со значением яркости r_k , сумма всех значений нормализованной гистограммы равна единице.

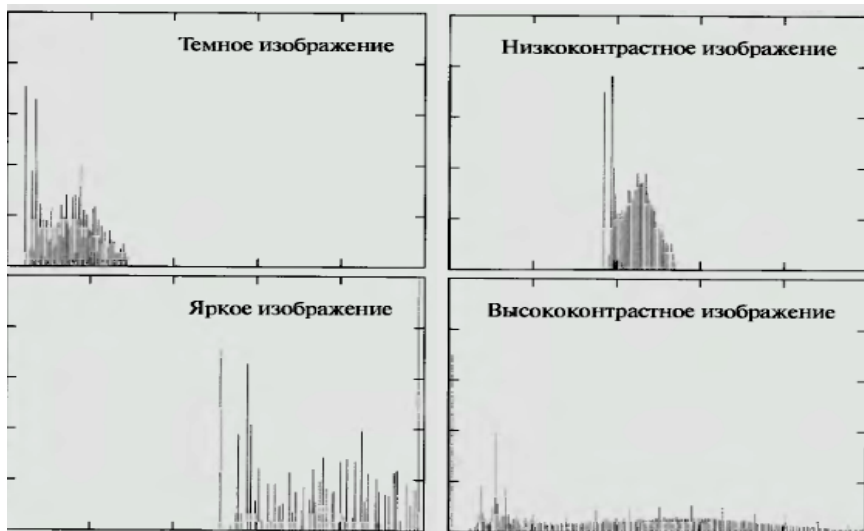


Рис. 1.4. Типичные гистограммы для 4 основных типов изображения

Эквализация гистограммы

Эквализацией или линеаризацией гистограммы цифрового изображения называется такое преобразование исходного изображения, при котором гистограмма эквализованного изображения перекрывает более широкий диапазон уровней яркости, чем гистограмма исходного изображения. Функция преобразования задается уравнением вида:

$$s_k = T(r_k) = \sum_{j=0}^k p_r(r_j) = \sum_{j=0}^k \frac{n_j}{n} \quad k = 0, 1, 2, \dots, L-1.$$

Таким образом, суть метода эквализации гистограммы заключается в нахождении функции преобразования, которая стремится сформировать выходное изображение с равномерной гистограммой. Эквализация гистограммы изображения приводит к существенному усилению контраста. К достоинствам метода относится простота его реализации и предсказуемость получаемых результатов.

Тем не менее в некоторых случаях улучшение изображения, основанное на модели равномерной гистограммы, не является наилучшим подходом. В некоторых случаях необходимо задавать желаемую форму гистограммы для обрабатываемого изображения. Метод, позволяющий получить обработанное изображение с заданной формой гистограммы, называется методом приведения или задания гистограммы. Данный метод представляет собой итеративную процедуру, заключающуюся в подборе оптимальной формы гистограммы.

1.3 Основные сведения о системе компьютерных вычислений MATLAB

Среда программирования MATLAB представляет собой высокоуровневый технический вычислительный язык и интерактивную среду для разработки алгоритмов, визуализации и анализа данных, числовых расчетов.

MATLAB как язык программирования был разработан в конце 1970-х годов. Цель – дать студентам возможность использования программных библиотек Linpack и EISPACK без необходимости изучения Фортрана. Вскоре новый язык распространился среди университетов и был с большим интересом встречен учёными, работающими в области прикладной математики. В дальнейшем создатели языка переписали MATLAB на C и основали в 1984 компанию The MathWorks для дальнейшего развития. Первоначально MATLAB предназначался для проектирования систем управления, но быстро завоевал популярность во многих других научных и инженерных областях. Он также широко использовался в образовании, в частности для преподавания линейной алгебры и численных методов.

Язык MATLAB является высокоуровневым языком программирования, включающим основанные на матрицах структуры данных, широкий спектр функций, интегрированную среду разработки, объектно-ориентированные возможности и интерфейсы к программам, написанным на других языках программирования.

MATLAB предоставляет удобные средства для разработки алгоритмов, включая высокоуровневые с использованием концепций объектно-ориентированного программирования. В нём имеются все необходимые средства интегрированной среды разработки, включая отладчик и профайлер. Функции для работы с целыми типами данных облегчают создание алгоритмов для микроконтроллеров и других приложений, где это необходимо.

В системе MATLAB имеется возможность создавать специальные наборы инструментов – toolbox, расширяющих его базовую функциональность. Данные наборы инструментов представляют собой коллекции функций, написанных на языке программирования MATLAB для решения определённого класса задач. Компания Mathworks поставляет наборы инструментов, которые используются во многих областях, включая решение таких инженерных задач, как обработка цифровых сигналов и изображений, сбор и анализ экспериментальных данных и множество других.

В частности, система MATLAB предоставляет пользователю многофункциональную и эффективную среду цифровой обработки двумерных изображений Image Processing Toolbox. Это специальный набор инструментов, расширяющий базовую функциональность среды MATLAB. Image Processing Toolbox представляет собой коллекции функций, написанных на языке MATLAB для решения широкого круга задач цифровой обработки изображений.

Отличительной особенностью языка MATLAB является его широкие возможности по работе с матрицами. Основным объектом системы MATLAB является прямоугольный числовой массив, который допускает комплексные элементы и ввод матриц без явного указания их размеров. MATLAB предоставляет пользователю большое количество (несколько сотен) функций для анализа данных, покрывающих практически все области математики. Среда MATLAB по-

зволяет решать многие вычислительные задачи за значительно меньшее время, нежели то, которое необходимо для написания соответствующих программ на классических языках программирования.

Работа в среде MATLAB может осуществляться в двух режимах:

- в режиме калькулятора, когда вычисления производятся непосредственно после набора очередного оператора или команды MATLAB; при этом значения результатов вычисления могут присваиваться некоторым переменным либо результаты получаются непосредственно, без присваивания (как в обычных калькуляторах);
- программном режиме – путем вызова программы, составленной и записанной на языке MATLAB, которая содержит все необходимые команды, обеспечивающие ввод данных, организацию вычислений и вывод результатов на экран.

В обоих режимах пользователю доступны практически все вычислительные возможности системы, в том числе по выводу информации в графической форме. Программный режим позволяет сохранять разработанные вычислительные алгоритмы и таким образом повторять вычисления при других исходных данных.

Программы, написанные на MATLAB, бывают двух типов — функции и скрипты. Функции имеют входные и выходные аргументы, а также собственное рабочее пространство для хранения промежуточных результатов вычислений и переменных. Скриптовые файлы используют общее рабочее пространство. Как скрипты, так и функции не интерпретируются в машинный код и сохраняются в виде текстовых файлов.

Рабочий стол MATLAB

После вызова MATLAB из среды операционной системы на экране появляется командное окно среды MATLAB. На рис. 1.5 приведен рабочий стол в среде MATLAB, содержащий четыре подокна: окно команд (Command Window), окно рабочего пространства (Workspace Browser), окно текущей папки (Current Directory Window), окно совершенных команд (Command History Window).

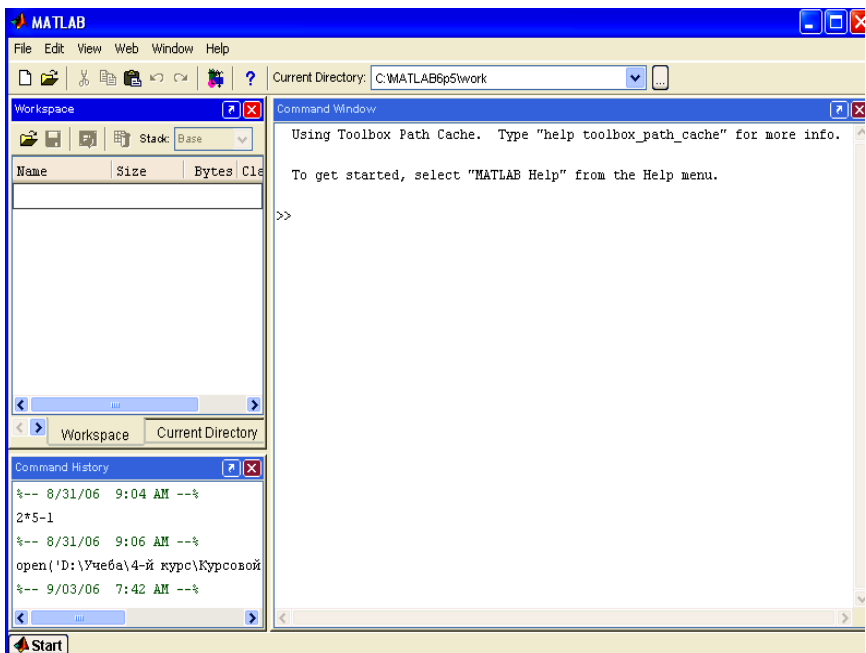


Рис. 1.5. Рабочий стол системы MATLAB и его основные компоненты

Окно команд (Command Window) – это область, где пользователь вводит команды и выражения на языке программирования среды MATLAB и где система размещает свои ответы на пользовательские команды. При каждом сеансе работы среда MATLAB формирует рабочее пространство, т.е. множество переменных, создаваемых пользователем. Команды вводятся после знака “>>”.

Окно совершенных команд (Command History Window) содержит записи всех команд, которые пользователь вводил в окне команд, включая все предыдущие сеансы работы в среде MATLAB. Любую команду можно вызвать еще раз двойным щелчком мыши на ней.

Окно рабочего пространства (Workspace Browser) отображает переменные, с которыми пользователь работает в данный момент, а также основную информацию о них – размер и тип переменной (рис. 1.6).

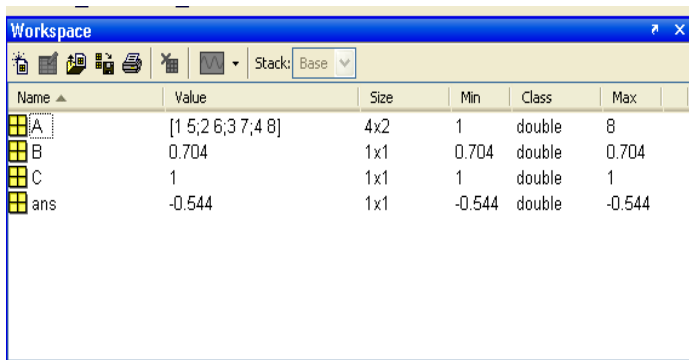


Рис. 1.6. Вид окна рабочего пространства (Workspace Browser)

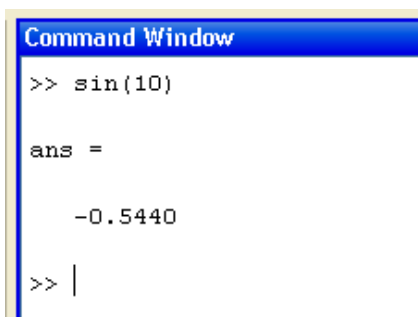


Рис. 1.7. Окно результата выполнения одиночной команды

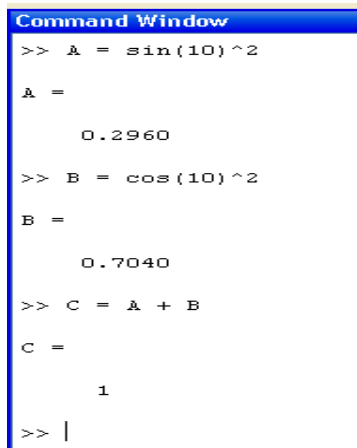
Работа с MATLAB в режиме командного окна

Язык программирования MATLAB имеет свой синтаксис, если в командном окне ввести команду **sin(10)**. На рис. 1.7 приведено окно результата.

Система MATLAB выдает результат путем создания новой переменной **ans** (это автоматически создаваемая переменная, которая содержит результат вычислений). Переменная **ans** также появляется в окне Workspace, где можно увидеть ее значение, размерность и т.д.

Очистить рабочее пространство Workspace можно командой **clear**. Для удаления одной или нескольких переменных надо указать их имена после команды **clear** (например, **clear a b c**). Для очистки командного окна используется команда **clc** (очистка не влияет на результаты).

Для присвоения переменной определенного значения используется обычный знак равенства “=”. На рис. 1.8 приведено окно Command Window, демонстрирующее данный вид записи.



```
Command Window
>> A = sin(10)^2
A =
    0.2960
>> B = cos(10)^2
B =
    0.7040
>> C = A + B
C =
    1
>> |
```

Рис. 1.8. Окно результата выполнения последовательности команд

В данном случае можно заметить, что переменная **ans** не была создана, поскольку результаты вычислений сохранились в переменных A, B и C.

Окно Command Window является основным в системе MATLAB. В нем отображаются символы команд, которые набираются пользователем, результаты выполнения этих команд, текст исполняемой программы, а также информация об ошибках выполнения программы, распознанных системой.

Признаком того, что система готова к восприятию и выполнению очередной команды, является наличие в последней строке текстового поля окна знака приглашения (>>), после которого возможен ввод символов или команд.

Вызов справки в среде MATLAB

Вызов окна справки в среде MATLAB осуществляется нажатием мышью на знак вопроса в строке инструментов рабочего стола или набором команды **helpbrowser** в командном окне. Справочная

система организована в виде веб-обозревателя, интегрированного в рабочий стол MATLAB, который способен отображать документы, написанные на языке HTML.

Существует другой путь получения справки по конкретной функции среды MATLAB. Для этого достаточно задать в командной строке команду `doc`, за которой следует поместить имя нужной функции. Справочная система среды MATLAB предоставляет пользователю исчерпывающую информацию по каждой имплементированной функции и содержит наглядные примеры использования данной функции для решения разного рода вычислительных задач.

Создание M-файлов (скриптов) в редакторе MATLAB

Работа в режиме калькулятора (командного окна) в среде MATLAB, несмотря на довольно значительные возможности, имеет существенные недостатки. Например, невозможно повторить все предыдущие вычисления и действия при новых значениях исходных данных без повторного набора всех предыдущих операторов; невозможно вернуться назад и повторить некоторые действия или по некоторому условию перейти к выполнению другой последовательности операторов. При большом количестве операторов возникает проблема отладки программы из-за неизбежных ошибок при наборе команд.

Создание программ позволяет значительно упростить и сократить процесс подготовки повторяемых вычислений, сделать процесс вычислений более наглядным и прозрачным, а благодаря этому уменьшить вероятность появления принципиальных ошибок при разработке программ. Кроме того, в программах появляется возможность автоматизировать процесс изменения значений входных параметров в диалоговом режиме.

Программирование в системе MATLAB является эффективным средством ее расширения и адаптации к решению специфических задач пользователя и реализуется с помощью входного языка системы, который является языком высокого уровня и содержит сложные операторы и функции.

Разработка в среде MATLAB сложных программ с прерываниями, переходами по определенным условиям, с часто повторяе-

мыми однотипными действиями, которые необходимо проводить неоднократно при измененных входных данных, требуют специального оформления программного кода в виде так называемых М-файлов. Все М-файлы имеют специальное расширение *.m. М-файл представляет собой созданный пользователем последовательный список исполняемых команд MATLAB, сохраненный на диске. Для создания нового М-файла необходимо с помощью меню редактора MATLAB выполнить следующую последовательность действий:

File->New->M-file

В среде программирования MATLAB существует возможность создавать программы 2 типов: скриптовые файл-программы (управляющие программы) и файл-функции (процедуры). При помощи скриптовых файлов оформляются основные программы, управляющие от начала до конца организацией вычислительного процесса. Скриптовые файл-программы не имеют входных и выходных аргументов, все переменные, объявленные в файле, становятся доступными в рабочей среде сразу после выполнения. В скриптовых файл-программах все используемые переменные образуют «рабочее пространство»; их значения сохраняются в течение всего сеанса работы с системой.

В файл-функции оформляются отдельные процедуры и разработанные пользователем специальные функции (т.е. такие части программы, которые рассчитаны на неоднократное использование при изменяемых входных параметрах). Функции могут быть полезны при программировании собственных приложений, функции производят необходимые действия со входными аргументами согласно составленной пользователем последовательности операций и возвращают результат в выходных аргументах. В файл-функциях все имена переменных внутри файла, а также имена переменных, указанные в заголовке, воспринимаются как локальные, т. е. все значения этих переменных после завершения работы процедуры исчезают, и область оперативной памяти компьютера, которая была отведена под запись значений этих переменных, освобождается для записи в нее значений других переменных.

Для исполнения скриптового М-Файла (файл-программы) в системе MATLAB необходимо выполнить одно из приведенных действий:

- 1) набрать имя файла в командной строке и нажать кнопку Enter (при этом необходимо помнить, что система MATLAB чувствительна к регистру);
- 2) нажать кнопку F5 на клавиатуре;
- 3) вызвать команду **Debug->Run** из меню редактора М-файлов.

Важной особенностью работы с файл-функциями в среде MATLAB является то, что файл-функции не оперируют переменными из рабочего пространства Workspace.

Для создания файл-функций в среде MATLAB необходимо создать новый М-файл, первая строка которого представляет собой заголовок файл-функции и должна содержать структуру следующего вида:

function f=funname(x)

где **x** – входной параметр (аргумент функции), **f** – выходной параметр (значение функции, определяемое значением аргумента **x**), **funname** – имя функции, **function** – обязательное наименование в заголовке m-файла, показывающее, что данный m-файл содержит файл-функцию.

После данного заголовка следует тело файл-функции, содержащее последовательность исполняемых команд, написанных на языке программирования среды MATLAB.

Название файла, содержащего файл-функцию, должно **обязательно** совпадать с названием m-файла, хранящегося в локальной директории. Для вызова и исполнения файл-функции в рабочем пространстве среды MATLAB необходимо поместить созданный m-файл в текущий каталог Current Directory, в том случае если пользователь использует каталог, отличный от текущего, необходимо добавить его в путь поиска среды MATLAB, в противном случае система не сможет использовать созданную файл-функцию.

Для вызова созданной пользователем файл-функции достаточно набрать в командной строке название файл-функции с указанием значения входного аргумента:

>> funname(x)

Вызов файл-функций может также осуществляться из скриптовых файл-программ или из других файл-функций в среде MATLAB. Синтаксис обращения к созданной пользователем файл-функции аналогичен синтаксису использования стандартных функций среды программирования MATLAB.

Файл-функции, создаваемые в среде MATLAB, могут иметь несколько входных аргументов, в этом случае все входные аргументы размещаются в списке и отделяются друг от друга запятой:

function f=funname(x, y, z)

Файл-функции могут также иметь несколько выходных аргументов, в этом случае выходные аргументы добавляются через запятую в список выходных параметров, который заключается в квадратные скобки:

function [f, y, s]=funname(x, y, z)

В среде MATLAB можно создавать файл-функции с переменным количеством входных параметров, в этом случае в качестве входного параметра будет задействован массив ячеек, позволяющий хранить разнородные данные. Все входные параметры сохраняются в виде единственного параметра массива ячеек **varargin**. С помощью функции **length(varargin)** можно будет вычислить количество указанных пользователем входных параметров, а с помощью конструкции вида **varargin(i)** обратиться к **i**-му из них:

function [f, y, s]=funname(varargin)

В системе MATLAB, начиная с версии MATLAB 7.0, разрешается использование вложенных файл-функций, структура основной файл-функции в этом случае имеет следующий вид:

```
function [f, y, s]=funname (x, y, z) % заголовок основной  
функции  
function [f1, y1, s1]=funname (x1, y1, z1) % заголовок первой  
подфункции  
<операторы 1> % операторы первой подфункции  
end % конец первой подфункции  
function [f2, y2, s2]=funname2 (x2, y2, z2) % заголовок вто-  
рой подфункции  
<операторы 2> % операторы второй подфункции  
end % конец второй подфункции  
.....
```

<операторы> *% операторы основной функции*
end *% конец основной функции*

Основная функция обменивается информацией с подфункциями при помощи входных и выходных параметров. Переменные, определенные в подфункциях и в основной функции, являются локальными и доступны только в пределах функции. Один из возможных вариантов использования переменных, которые являются общими для всех файл-функций, состоит в объявлении данных переменных в начале основной функции и начале подфункции как глобальных с помощью оператора **global**, содержащего список переменных, разделенных пробелом:

global x y z

В среде MATLAB есть возможность передавать имя функции как входной параметр файл-функции, что существенно расширяет возможности программирования. В данном случае имя функции передается как строка, а вычисление функции осуществляется с помощью оператора **feval**:

feval('funname', x, y, z)

где **'funname'** – строка с именем вызываемой функции, которая может быть как стандартной встроенной функцией, так и функцией, определенной пользователем, **x, y, z** – значения аргументов функции **funname**, разделенные запятой.

Одним из преимуществ языка программирования среды MATLAB является возможность использования рекурсивных функций. Под рекурсией в программировании понимается вызов функции из ее тела. Классическими рекурсивными функциями является возведение числа в целую положительную степень, вычисление факториала. В рекурсивных алгоритмах функция вызывает саму себя до выполнения какого-либо условия.

1.4 Основы цифровой обработки изображений в среде MATLAB

Загрузка изображений в среде MATLAB

Для загрузки (чтения) изображений в рабочее пространство MATLAB используется функция **imread** со следующим синтаксисом:

imread('filename')

filename – строка символов, образующая полное имя загружаемого файла изображения, включающее расширение файла и путь к файлу на компьютере.

Следующая команда присваивает файл изображения с названием **chestxray** формата JPEG, расположенного в папке **images** на локальном диске **C** матричной переменной **f**:

f=imread('C:\images\chestxray.jpg')

Среда программирования MATLAB, начиная с версии 6.5, поддерживает следующие графические форматы: TIFF, JPEG, GIF, BMP, PNG, XWD.

Функция **size** возвращает размер изображения, т.е. число строк и столбцов матрицы, автоматическое определение размера изображения **f** можно осуществить с помощью следующей операции:

[M, N]=size(f);

Переменной **M** будет присвоено значение, равное числу строк изображения, переменной **N** – числу столбцов изображения.

Для получения информации о графическом файле используется следующая команда:

imfinfo filename

filename – полное имя файла, хранящегося на диске.

Вывод изображения на дисплей

Изображения можно вывести на дисплей компьютера с помощью функции **imshow**, которая имеет следующий синтаксис:

imshow(f, G)

f – матрица изображения, **G** – число уровней яркости, используемое при отображении данного изображения. Если пользователь не определил переменную **G**, то по умолчанию используется 256 уровней яркости.

Следующая команда обозначает, что все пиксели со значениями яркости не более величины **low** надо показывать черными, все пиксели со значениями яркости не меньше числа **high** надо показывать белыми, все промежуточные значения остаются без изменения.

imshow(f, [low high])

Запись последней команды в следующем виде задает значение переменной **low**, равной минимальному значению в массиве **f**, переменной **high** – максимальному значению в массиве **f**.

imshow(f, [])

Последняя форма записи команды **imshow** бывает полезной при показе изображений, имеющих узкий динамический диапазон значений пикселей, или когда среди пикселей имеются положительные и отрицательные значения.

Для того чтобы вывести два изображения одновременно на экран в разных окнах, необходимо использовать команду **figure**, например, следующим образом:

```
imshow(f)  
figure  
imshow(g)
```

Сохранение изображений

Сохранение изображений на локальный диск или внешний носитель осуществляется функцией **imwrite**, которая имеет следующий синтаксис:

imwrite(f, 'filename')

Строка **filename** должна содержать расширение, поддерживаемое системой MATLAB. Если строка **filename** не содержит информацию о полном пути хранения файла, то изображение записывается в файл в текущую рабочую папку, по умолчанию находящуюся в директории:

C:\Program Files\Mathworks\work.

В зависимости от формата графического файла функция **imwrite** может содержать дополнительный набор входных аргументов и обеспечивать большую функциональность. Более подробно с особенностями работы функции можно познакомиться в справочной системе MATLAB с помощью команды вызова справки **doc imwrite**.

Преобразование яркости изображений

Функция **imadjust** является базовым инструментом в пакете MATLAB для преобразования яркости полутоновых изображений. Данная функция имеет следующий синтаксис:

g=imadjust(f, [low_in, high_in], [low_out, high_out], gamma)

На рис. 1.9 проиллюстрировано отображение с помощью этой функции исходного изображения **f** в новое изображение **g**, при котором значения яркости в интервале **[low_in, high_in]** переходят в значение интервала **[low_out, high_out]**, а значения меньше порога **low_in** или больше порога **high_in** обрезаются.

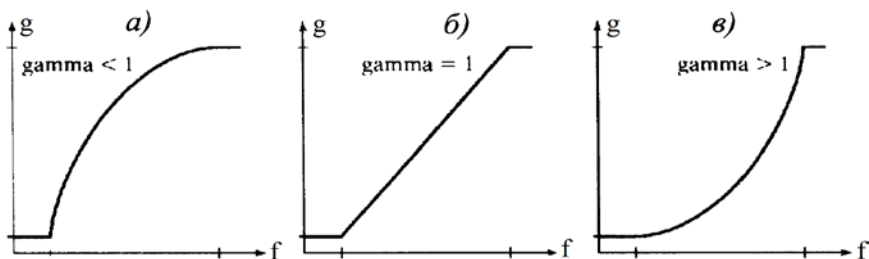


Рис. 1.9. Различные отображения, допустимые в функции **imadjust**

Все входные параметры функции **imadjust** за исключением **f** являются вещественными числами в диапазоне от 0 до 1. Параметр **gamma** служит для задания формы кривой, отображающей яркость **f** в яркость **g**. По умолчанию параметр **gamma = 1**, что соответствует линейному отображению.

Степенное преобразование яркости изображений, или гамма-коррекция, определяется с помощью следующего выражения:

$$S = c \cdot r^\gamma,$$

где r – входная (исходная) яркость, S – выходная яркость, c и γ – положительные константы.

В системе MATLAB операции возведения в степень, умножения и деления матриц при обработке соответствующих изображений необходимо выполнять поэлементно, для этого используется следующая запись команд (соответственно):

$$g=f.^{0.5} \quad g=f.*h \quad g=f./h$$

При умножении или делении матрицы на константу (скаляр) можно использовать обычную запись арифметических операций: **g=f*2**, **g=f/2**.

Обработка гистограмм изображения

Для построения гистограмм в системе MATLAB используется функция **imhist** со следующим синтаксисом:

h=imhist(f, b)

f – входное изображение, **h** – гистограмма изображения, **b** – число уровней разбиения шкалы яркости, использованных при формировании гистограмм.

По умолчанию параметр **b** принимается равным 256. Например, если при работе с 8-битным изображением пользователь задает **b**, равное 2, то шкала яркости делится на 2 области: от 0 до 127 и от 128 до 255. Итоговая гистограмма будет иметь два значения: $h(1)$, равное числу пикселей изображения, величины яркости которых находятся в диапазоне $[0, 127]$, и $h(2)$, которое равно числу пикселей, значения яркости которых находятся в диапазоне $[127, 255]$.

Для получения нормированной гистограммы (оггибающая гистограммы будет являться оценкой функции распределения вероятностей) необходимо выполнить следующее действие:

p=imhist(f, b)/numel(f)

где **numel(f)** – общее число пикселей изображения.

Эквализация гистограммы в пакете MATLAB реализуется с помощью функции **histeq**, имеющей синтаксис:

g=histeq(f, nlev)

f – входное изображение, **nlev** – число уровней интенсивности, установленное для выходного изображения.

1.5 Практическое задание № 1

1. Загрузите тестовое изображение рентгенограммы молочной железы **mammary.jpg**, получите полную информацию о данном графическом файле, сохраните исходное изображение в формат **tif** без сжатия, присвойте файлу свое имя, сократите размер изображения до $1,5 \times 1,5$ дюйма, сохранив при этом его пиксельный размер 482×571 . Используйте справочную информацию пакета MATLAB о функции **imwrite** с помощью следующей команды: **doc imwrite**.

2. Загрузите снова исходное изображение рентгенограммы молочной железы **mammary.jpg**. Сформируйте с помощью команды **imadjust** негативное изображение, а также получите изображения с

преобразованной яркостью при различных значениях входных порогов и параметра **gamma** функции **imadjust**. Выберите такие параметры функции **imadjust**, которые, на ваш взгляд, обеспечивают получение наиболее информативной и наглядной информации о содержащейся опухоли.

3. Загрузите изображение позвоночника человека **vertebra.jpg**, полученное с помощью ЯМР-томографа. На основе применения методов степенного преобразования яркости (гамма-коррекция) сформируйте изображение с большей контрастностью, выбор оптимальных параметров гамма-коррекции осуществляется на основе визуального контроля.

4. Постройте гистограммы для изображений **vertebra.jpg** и **mammary.jpg**.

5. Выполните процедуру эквализации гистограммы для изображений **vertebra.jpg** и **mammary.jpg**. Постройте гистограммы эквализованных изображений и сравните их с исходными гистограммами. Сравните метод эквализации гистограммы с рассмотренными ранее методами градиентного преобразования яркости. Сделайте выводы о полученных результатах.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Представление о цифровом изображении.
2. Квантование и дискретизация изображений.
2. Пространственное и яркостное разрешение изображения.
3. Пространственные методы улучшения изображений.
4. Гистограмма цифрового изображения.
5. Эквализация гистограмм изображения как метод улучшения изображения.

ГЛАВА 2 ПРОСТРАНСТВЕННАЯ ФИЛЬТРАЦИЯ МЕДИЦИНСКИХ ИЗОБРАЖЕНИЙ В СРЕДЕ MATLAB

2.1 Линейная пространственная фильтрация изображений

Локальные преобразования оперируют одновременно как со значениями пикселей в окрестности, так и с соответствующими им значениями некоторой матрицы, имеющей те же размеры, что и окрестность. Такую матрицу принято называть фильтром, маской, ядром, шаблоном или окном. Схема пространственной фильтрации проиллюстрирована на рис. 2.1, процесс основан на простом перемещении маски фильтра от точки к точке изображения, в каждой точке (x, y) отклик фильтра вычисляется с использованием предварительно заданных связей.

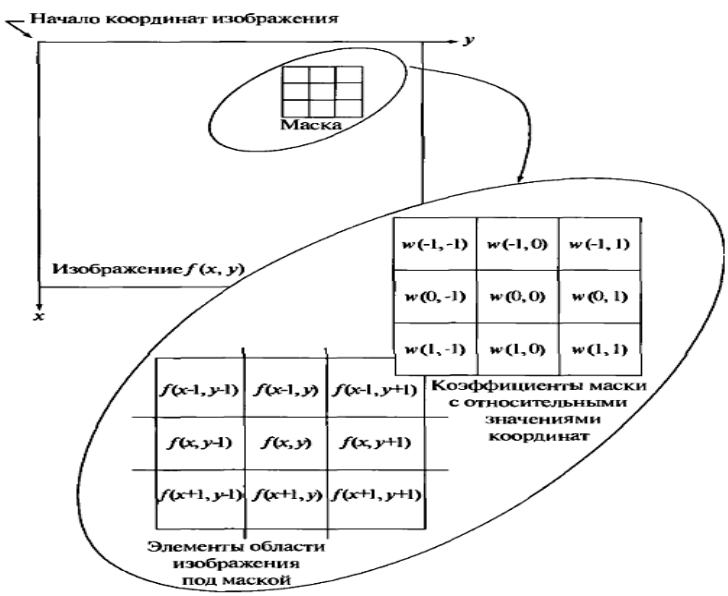


Рис. 2.1. Схема пространственной фильтрации

Фильтрация изображения f , имеющего размеры $M \times N$, с помощью фильтра размерами $m \times n$ задается выражением общего вида:

$$g(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x + s, y + t),$$

где $a = (m - 1)/2$, $b = (n - 1)/2$, $x = 0, 1, 2 \dots M - 1$; $y = 0, 1, 2 \dots N - 1$.

Для маски размером элемента 3×3 результат линейной фильтрации изображения в точке (x, y) представлен выражением:

$$R = w(-1, -1) f(x - 1, y - 1) + w(-1, 0) f(x - 1, y) + \dots \\ + w(0, 0) f(x, y) + \dots + w(1, 0) f(x + 1, y) + w(1, 1) f(x + 1, y + 1).$$

Таким образом, процедура линейной фильтрации представляет собой процедуру свертки коэффициентов фильтра (маски) с отсчетами исходного изображения. В случае, когда интерес представляет только значение отклика R по маске $m \times n$, в точке (x, y) используется следующее выражение:

$$R = w_1 z_1 + w_2 z_2 + \dots + w_{mn} z_m n = \sum_{i=1}^{mn} w_i z_i,$$

где w_i – коэффициент маски, z_i – значение пикселя, соответствующего данному коэффициенту, $m \times n$ – общее число коэффициентов в маске.

2.2 Сглаживающие пространственные фильтры

Сглаживающие фильтры применяются для расфокусировки изображения и подавления шума. Расфокусировка применяется как предварительный шаг обработки изображения, например, для удаления мелких деталей перед обнаружением больших объектов, для устранения разрывов в линиях или деталях.

Отклик или выходной сигнал линейного сглаживающего фильтра представляет собой среднее значение элементов по окрестности, покрытой маской фильтра, что обеспечивает сглаживающий эффект. Такие фильтры иногда также называют низкочастотными фильтрами.

Заменой исходных значений элементов изображения на средние значения по маске фильтра достигается уменьшение резких переходов уровней яркости. Поскольку случайный шум как раз характеризуется резкими скачками яркости, наиболее очевидным применением сглаживания является подавление шума. Однако негативной

стороной применения сглаживающих фильтров является расфокусировка контуров изображения, которые также характеризуются резкими перепадами яркостей. Главное использование сглаживающих фильтров состоит в подавлении несущественных деталей на изображении, под которыми понимается совокупность пикселей, которые малы по сравнению с размерами маски фильтра.

$$R = \frac{1}{9} \sum_{i=1}^9 z_i,$$

$\frac{1}{9} \times$	1	1	1
	1	1	1
	1	1	1

$\frac{1}{16} \times$	1	2	1
	2	4	2
	1	2	1

Рис. 2.2. Маски сглаживающих фильтров

Постоянный множитель перед маской равен единице, деленной на сумму значений коэффициентов, множитель необходим для нормировки. Первый фильтр имеет одинаковые коэффициенты и называется однородным усредняющим фильтром, выходное значение данного фильтра представляет собой обычное среднее значение по маске. Вторая маска обеспечивает на выходе фильтра взвешенное среднее, коэффициент в центре маски имеет самое большое значение, значения остальных коэффициентов уменьшаются по мере удаления от центра маски. Основная стратегия применения подобного рода фильтров – уменьшение расфокусировки при сглаживании.

Общая формула фильтрации изображения размерами $M \times N$ с помощью фильтра взвешенного среднего по окрестности $m \times n$ определяется выражением:

$$g(x, y) = \frac{\sum_{s=-a}^a \sum_{t=-b}^b w(s, t) \cdot f(x + s, y + t)}{\sum_{s=-a}^a \sum_{t=-b}^b w(s, t)}.$$

Подбор оптимального размера маски требует проведения нескольких экспериментов и определяется как размерами исходного изображения, так и основным назначением выполнения процедуры сглаживания изображения.

2.3 Пространственные фильтры повышения резкости изображений

Главная цель повышения резкости заключается в том, чтобы подчеркнуть мелкие детали изображения или улучшить те детали, которые оказались расфокусированы вследствие ошибок или несовершенства самого метода регистрации изображений.

Ранее было показано, что расфокусировка изображения осуществлялась путем усреднения значений пикселей по окрестности. Повышение резкости изображения может быть достигнуто с помощью обратной процедуры – пространственного дифференцирования.

Дифференцирование изображения позволяет усилить перепады и другие разрывы на изображении, но также неизбежно приводит и к усилению шумов на изображении. Величина отклика оператора производной в точке изображения пропорциональна степени разрывности изображения в данной точке.

Производные дискретной функции определяются в терминах разности.

Первая производная должна быть:

- 1) равной нулю на участках с постоянным уровнем яркости;
- 2) ненулевой в начале и в конце ступеньки или склона яркости;
- 3) ненулевой на склонах яркости.

Вторая производная должна быть:

- 1) равной нулю на участках с постоянным уровнем яркости;
- 2) ненулевой в начале и в конце ступеньки или склона яркости;
- 3) равной нулю на склонах постоянной крутизны.

Первая производная одномерной функции определяется как разность значений соседних элементов:

$$\frac{\partial f}{\partial x} = f(x+1) - f(x).$$

Аналогично вторая производная определяется как разность соседних значений первой производной:

$$\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x).$$

Отметим, что вторая производная является намного более эффективной при решении задач усиления резких переходов на изображении, чем первая производная, однако вторая производная обеспечивает намного большее усиление мелких деталей, включая шум.

Улучшение изображений с использованием вторых производных

В данном случае рассматривается использование оператора двумерной второй производной в задачах улучшения изображения. Подход сводится к выбору дискретной формулировки второй производной и к последующему построению маски фильтра, основанной на предложенной формулировке. В данных задачах рассматриваются исключительно изотропные фильтры, отклик которых не зависит от направления неоднородностей на обрабатываемом изображении. Изотропные фильтры обладают свойством инвариантности к повороту изображения, то есть поворот изображения и последующее применение фильтра дает тот же результат, что и первоначальное применение фильтра с последующим поворотом результата.

Простейшим оператором, основанным на производных, является оператор Лапласа – лапласиан, который для функции двух переменных определяется как:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}.$$

Для применения оператора Лапласа в цифровой обработке изображений его необходимо выразить в дискретном виде. Наиболее часто используется следующее представление:

$$\begin{aligned}\frac{\partial^2 f}{\partial x^2} &= f(x+1, y) + f(x-1, y) - 2f(x, y), \\ \frac{\partial^2 f}{\partial y^2} &= f(x, y+1) + f(x, y-1) - 2f(x, y)\end{aligned}$$

Дискретная формулировка двумерного лапласиана получается объединением этих двух составляющих:

$$\nabla^2 f = [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] - 4f(x, y).$$

Данное уравнение может быть реализовано с помощью маски, представленной на рис. 2.3.

0	1	0
1	-4	1
0	1	0

Рис. 2.3. Маска фильтра, используемая для реализации дискретного лапласиана

Данная маска дает изотропный результат для поворотов на углы, кратные 90° . В формулу дискретного лапласиана могут быть включены диагональные направления добавлением еще двух членов – по одному для каждого из диагональных направлений. Маска, соответствующая такому определению, представлена на рис. 2.4.

1	1	1
1	-8	1
1	1	1

Рис. 2.4. Маска фильтра, используемая для реализации дискретного лапласиана, полученная с учетом диагональных направлений

Данная маска является изотропной для поворотов на углы, кратные 45° .

В силу того, что оператор Лапласа по сути является второй производной, его применение подчеркивает разрывы уровней яркостей на изображении и подавляет области со слабыми изменениями яркостей. Это приводит к получению изображения, содержащего сероватые линии на месте контуров и других разрывов, наложенные на темный фон без особенностей. Для восстановления прежнего фона при сохранении эффекта повышения резкости, достигаемого использованием лапласиана, достаточно сложить исходное изображение и изображение, получаемое на выходе оператора Лапласа.

Необходимо отметить, что результаты, получаемые с применением маски, содержащей диагональные элементы (рис. 2.4), являются более резкими, чем с применением первичной маски, приведенной на рис. 2.3.

2.4 Нерезкое маскирование и фильтрация с подъемом высоких частот

Процедура, используемая для повышения резкости изображения, заключающаяся в вычитании из исходного изображения его расфокусированной копии, называется *нерезким маскированием* и выражается следующей формулой:

$$f_s(x, y) = f(x, y) - \bar{f}(x, y),$$

где $f_s(x, y)$ – изображение с повышенной резкостью, полученное методом нерезкого маскирования, $\bar{f}(x, y)$ – расфокусированная копия исходного изображения $f(x, y)$.

Обобщенной операцией нерезкого маскирования является фильтрация с подъемом высоких частот, которая задается в следующем виде:

$$f_{hb}(x, y) = Af(x, y) - \bar{f}(x, y),$$

где $A \geq 1$. При увеличении A больше 1 эффект увеличения резкости становится меньше. Типичным применением фильтрации изображений с подъемом высоких частот является случай, когда исходное изображение темнее, чем это требуется. Варьированием постоянного коэффициента A удается добиться общего увеличения яркости изображения и получить желаемый результат.

Улучшение изображений с использованием первых производных

В обработке изображений первые производные реализуются через модуль градиента. Для функции $f(x, y)$ градиент f в точке (x, y) определяется как двумерный вектор-столбец:

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}.$$

Модуль этого вектора часто называют градиентом, который определяется следующим образом:

$$\nabla f = |\nabla f| = [G_x^2 + G_y^2]^{0,5} = \left[\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2 \right]^{0,5}.$$

В силу большого объема вычислений, необходимых для обработки всего изображения оператором градиента, на практике часто используют приближенное значение в виде:

$$\nabla f \approx |G_x| + |G_y|.$$

Определим дискретные приближения приведенных уравнений. Для указания пикселей в окрестности 3×3 будем использовать обозначения, представленные на рис. 2.5.

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

Рис. 2.5. Область изображения элемента размером 3×3

Простейшими приближениями первой производной являются следующие выражения: $G_x = z_8 - z_5$ и $G_y = z_6 - z_5$. Робертс предложил использовать перекрестные направления: $G_x = z_9 - z_5$ и $G_y = z_8 - z_6$. В этом случае градиент будет определяться следующим выражением:

$$\nabla f = [(z_9 - z_5)^2 + (z_8 - z_6)^2]^{0,5}.$$

В случае использования приближенной формулы для определения градиента:

$$\nabla f \approx |z_9 - z_5| + |z_8 - z_6|.$$

Приведенное уравнение может быть реализовано с помощью масок, приведенных на рис. 2.6.

-1	0
0	1

0	-1
1	0

Рис. 2.6. Маски 2×2, используемые для вычисления градиента

Данные маски называют перекрестным градиентным оператором Робертса. Однако маски четного размера неудобны в своей реализации, поэтому более предпочтительным является применение масок с минимальным размером окрестности 3×3, в этом случае градиентный оператор может быть реализован с помощью следующего выражения:

$$\nabla f \approx |z_7 + 2z_8 + z_9| - |z_1 + 2z_2 + z_3| + |z_3 + 2z_6 + z_9| - |z_1 + 2z_4 + z_7|.$$

Соответствующие маски приведены на рис. 2.7. Представленные маски называют оператором Собела.

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

Рис. 2.7. Маски 3×3, используемые для вычисления градиента

Использование весовых коэффициентов со значением 2 основано на стремлении присвоить центральным точкам большую значимость и тем самым достичь большей гладкости. Заметим, что сумма коэффициентов каждой из масок равна 0, что обеспечивает нулевой отклик операторов первой производной на участках изображения с постоянным уровнем яркости, что соответствует требованиям к оператору первой производной. Градиентные операторы применяются для улучшения контуров, в задачах улучшения видимости дефектов и для удаления слабо меняющихся характеристик фона.

2.5 Комбинированные методы пространственного улучшения изображений

При решении конкретных задач обработки биомедицинских изображений для достижения приемлемого результата может потребоваться применение нескольких дополняющих друг друга пространственных методов улучшения изображения.

В клинической диагностике используются полные снимки скелета, полученные с помощью гамма-лучей, для обнаружения таких заболеваний, как костные инфекции и опухоли. Одной из актуальных задач обработки такого рода изображений является повышение резкости изображения для более подробного выявления деталей скелета. Зачастую такие изображения характеризуются малым динамическим диапазоном уровней яркости и высоким содержанием шума, что затрудняет реализацию поставленных задач.

Стратегия обработки данного вида изображений состоит в использовании лапласиана для выделения мелких деталей скелета и градиента для улучшения выступающих краев, а также градиентного преобразования для увеличения динамического диапазона яркостей.

Альтернативным подходом является использование маски, формируемой путем сглаживания градиента исходного изображения. Оператор второй производной (лапласиан) обеспечивает значительное улучшение отображения мелких деталей изображения по сравнению с градиентом, однако при этом лапласиан усиливает шум в большей степени, чем градиент. Градиент по сравнению с лапласианом дает более сильный отклик в областях со значительными изменениями яркости (яркостные переходы и ступеньки), при этом отклик градиента на шум и мелкие детали слабее, чем у лапласиана, и может быть в дальнейшем дополнительно снижен путем сглаживания градиента усредняющим фильтром.

2.6 Реализация методов пространственной фильтрации в среде MATLAB

В пакете MATLAB линейная пространственная фильтрация реализуется с помощью функции **imfilter**, которая имеет следующий синтаксис:

g=imfilter(f, w, filtering_mode, boundary_options, size_options)

где f – входное изображение, w – фильтрующая маска, g – результат фильтрации. Остальные параметры приведены в табл. 2.1.

Таблица 2.1. Опции функции **imfilter**

Опции	Описание
<i>Мода фильтрации</i>	
'corr'	Фильтрация делается методом корреляции. Мода по умолчанию.
'conv'	Фильтрация делается методом свертки
<i>Граничные опции</i>	
P	Границы изображения расширяются значением P (без апострофов). По умолчанию P=0.
'replicate'	Размер изображения увеличивается повторением величин на его боковых границах.
'symmetric'	Размер изображения увеличивается путем зеркального отражения через границы.
'circular'	Размер изображения увеличивается периодическим повторением двумерной функции.
<i>Опции размера</i>	
'full'	Выход имеет те же размеры, что и расширенное входное изображение.
'same'	Выход имеет те же размеры, что и вход. Это достигается с помощью ограничения перемещения центра фильтрующей маски точками, принадлежащими исходному изображению. Опция по умолчанию.

Механизм линейной пространственной фильтрации был проиллюстрирован на рис. 2.1. Суть процесса заключается в перемещении центра фильтрующей маски w от точки к точке изображения f . В каждой точке (x, y) отклик фильтра определяется суммой произведений коэффициентов фильтра и соответствующих пикселей окрестности, покрываемой маской. Как правило, выбирают маски нечетных размеров в силу наличия выраженной центральной точки.

Механизм вычисления результатов фильтрации осуществляется двумя способами: корреляцией и сверткой. Корреляция состоит в прохождении маски w по изображению f , как показано на рис. 2.1. Свертка выполняется аналогично с тем отличием, что маску w необходимо повернуть на 180^0 перед прохождением по изображению f .

Наиболее часто функция **imfilter** применяется в виде следующей команды:

g=imfilter(f, w, 'replicate')

В пакете MATLAB имеются некоторые стандартные двумерные линейные пространственные фильтры, которые можно получить с помощью функции **fspecial**, генерирующей маску фильтра **w** при выполнении следующей команды:

w=fspecial('type', parameters)

где **type** обозначат тип фильтра, а **parameters** задают параметры выбранного фильтра.

Пространственные фильтры, получаемые данной командой, приведены в табл. 2.2 с соответствующими параметрами каждого фильтра. Необходимо заметить, что результаты фильтрации изображений с помощью применения некоторых видов масок могут содержать отрицательные значения пикселей. При выводе таких изображений на дисплее пиксели с отрицательными значениями яркостей будут приравняться к нулю, что приведет к сокращению динамического диапазона яркости и искажению получаемых результатов. Для преодоления указанных трудностей необходимо преобразовать исходное изображение в класс **double** перед фильтрацией с помощью следующей команды:

f2=im2double(f)

где **f** – исходное изображение класса **uint8**, **uint16** или **logical**.

Различные классы данных приведены в табл. 2.3.

Большинство медицинских изображений представляет собой полутоновые изображения, элементы которых относятся наиболее часто к следующим классам данных: **uint8**, **uint16** и **double**. Если элементы полутонового изображения относятся к классу **uint8** или **uint16**, то они представлены целыми числами в интервалах [0 – 255] и [0 – 65535] соответственно. Если изображение принадлежит классу **double**, то его пиксели представляют собой вещественные числа с плавающей точкой. По общему соглашению пиксели изображений класса **double** должны лежать в интервале [0 – 1].

Таблица 2.2. Пространственные фильтры функции **fspecial**

Тип	Синтаксис и параметры
'average'	<code>fspecial('average', [r c])</code> . Прямоугольный усредняющий фильтр размера $r \times c$. По умолчанию 3×3 . Одно число на месте $[r c]$ означает квадратный фильтр.
'disk'	<code>fspecial('disk', r)</code> . Круговой усредняющий фильтр (внутри квадрата со стороной $2r+1$) радиуса r . По умолчанию $r = 5$.
'gaussian'	<code>fspecial('gaussian', [r c], sig)</code> . Низкочастотный гауссов фильтр размера $r \times c$ со стандартным (положительным) отклонением <code>sig</code> . Значения по умолчанию 3×3 и 0.5 . Одно число на месте $[r c]$ означает квадратный фильтр.
'laplacian'	<code>fspecial('laplacian', alpha)</code> . Фильтр Лапласа 3×3 , форма которого задается параметром <code>alpha</code> из интервала $[0, 1]$. По умолчанию <code>alpha = 0.5</code> .
'log'	<code>fspecial('log', [r c], sig)</code> . Лаплас от гауссова фильтра (LoG) размера $r \times c$ со стандартным (положительным) отклонением <code>sig</code> . Значения по умолчанию 5×5 и 0.5 . Одно число на месте $[r c]$ означает квадратный фильтр.
'motion'	<code>fspecial('motion', len, theta)</code> . Выдает фильтр, который, будучи свернутым с изображением, приближает линейное перемещение (видеокамеры по отношению к изображению) на <code>len</code> пикселей. Направление перемещения задается углом <code>theta</code> , который измеряется в градусах от горизонтали против часовой стрелки. Значения по умолчанию 9 и 0 , что соответствует перемещению на 9 пикселей в горизонтальном направлении.
'prewitt'	<code>fspecial('prewitt')</code> . Выдает 3×3 маску Превитта <code>wv</code> , которая аппроксимирует вертикальный градиент. Маску горизонтального градиента можно получить, транспонировав результат: <code>wh = wv'</code> .
'sobel'	<code>fspecial('sobel')</code> . Выдает 3×3 маску Собела <code>sv</code> , которая аппроксимирует вертикальный градиент. Маску горизонтального градиента можно получить, транспонировав результат: <code>sh = sv'</code> .
'unsharp'	<code>fspecial('unsharp', alpha)</code> . Выдает 3×3 маску нечеткого фильтра. Параметр <code>alpha</code> контролирует форму, он должен быть не меньше 0 и не больше 1.0 . По умолчанию <code>alpha = 0.2</code> .

Таблица 2.3. Классы данных

Имя	Описание
<code>double</code>	Вещественные числа с плавающей запятой двойной точности в диапазоне, примерно, от -10^{308} до 10^{308} (8 байт на число).
<code>uint8</code>	Целые без знака в интервале $[0, 255]$ (1 байт на число).
<code>uint16</code>	Целые без знака в интервале $[0, 65535]$ (2 байта на число).
<code>uint32</code>	Целые без знака в интервале $[0, 4294967295]$ (4 байта на число).
<code>int8</code>	Целые со знаком в интервале $[-128, 127]$ (1 байт на число).
<code>int16</code>	Целые со знаком в интервале $[-32768, 32767]$ (2 байта на число).
<code>int32</code>	Целые со знаком в интервале $[-2147483648, 2147483647]$ (4 байта).
<code>single</code>	Вещественные числа с плавающей запятой обычной точности в диапазоне, примерно, от -10^{38} до 10^{38} (4 байта на число).
<code>char</code>	Символы (буквы и знаки) (2 байта на символ).
<code>logical</code>	Значения 0 или 1 (1 байт на элемент).

2.7 Практическое задание № 2

1. Загрузите тестовое изображение скелета человека **skeleton.jpg**. Методом фильтрации с подъемом высоких частот добейтесь максимального повышения резкости исходного изображения. Для получения расфокусированного изображения используйте линейный усредняющий фильтр.

2. Примените линейную пространственную фильтрацию с помощью оператора Лапласа к исходному изображению скелета человека **skeleton.jpg**.

3. Сложите исходное изображение скелета человека **skeleton.jpg** с изображением, полученным в результате выполнения второго пункта задания.

4. Примените оператор Собела к исходному изображению скелета человека **skeleton.jpg**.

5. Изображение, полученное в результате выполнения четвертого пункта задания, подвергните фильтрации с помощью сглаживающего усредняющего фильтра. С помощью масок различного размера добейтесь оптимальных результатов.

6. Получите изображение путем умножения изображения, полученного после выполнения пятого пункта задания, и изображения, полученного в результате выполнения третьего пункта задания. Необходимо помнить, что речь идет о поэлементном умножении матриц изображений.

7. Получите изображения с повышенной резкостью путем сложения изображения, полученного в результате выполнения шестого пункта задания, и исходного изображения скелета человека **skeleton.jpg**.

8. Преобразуйте изображение, полученное в результате выполнения седьмого пункта задания, с помощью метода градиционной коррекции по степенному закону.

9. Сравните различные методы пространственной фильтрации, использованные при выполнении всех пунктов задания. Сравните изображения, полученные в результате выполнения первого, третьего, пятого и восьмого пунктов задания. Сделайте выводы о полученных результатах.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Основной принцип метода линейной пространственной фильтрации изображений.
2. Сглаживающие линейные пространственные фильтры.
3. Пространственные фильтры повышения резкости.
4. Пространственные фильтры повышения резкости на основе второй производной.
5. Пространственные фильтры повышения резкости на основе первой производной.
6. Нерезкое маскирование и фильтрация с подъемом высоких частот.
7. Особенности применения комбинированных методов пространственной обработки биомедицинских изображений.

ГЛАВА 3 ЧАСТОТНАЯ ФИЛЬТРАЦИЯ МЕДИЦИНСКИХ ИЗОБРАЖЕНИЙ В СРЕДЕ MATLAB

3.1 Введение в Фурье-анализ

Частотные методы улучшения изображений основаны на выполнении преобразования Фурье над функцией двух переменных – функцией дискретного изображения. Прямое дискретное Фурье-преобразование функции $f(x, y)$, изображения размером $M \times N$, задается равенством:

$$F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-i2\pi\left(\frac{ux}{M} + \frac{vy}{N}\right)},$$

где $u = 0, 1, 2, \dots, M - 1$; $v = 0, 1, 2, \dots, N - 1$.

Обратное преобразование Фурье определяется выражением:

$$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{i2\pi\left(\frac{ux}{M} + \frac{vy}{N}\right)}.$$

Переменные u и v называются переменными преобразования или частотными переменными, а переменные x и y называются пространственными переменными или переменными изображения. Как правило, числа M и N являются четными для упрощения компьютерной реализации, а центр Фурье-образа находится в точке с координатами: $u = (M/2) + 1$, $v = (N/2) + 1$. Значение Фурье-преобразования в точке $(u, v) = (0, 0)$ равно:

$$F(0, 0) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y).$$

Таким образом, если $f(x, y)$ – изображение, то значение Фурье-преобразования в начале координат равно среднему значению яркости на изображении. Поскольку началу координат отвечают нулевые частоты, то величину $F(0, 0)$ часто называют постоянной составляющей спектра. В силу того, что функция $f(x, y)$ вещественная, то спектр Фурье-преобразования изображения обладает свойством симметрии. Справедливы следующие соотношения между отсчетами в пространственной и частотной области:

$$\Delta u = \frac{1}{M\Delta x}, \quad \Delta v = \frac{1}{N\Delta y}.$$

Фурье-спектр изображения состоит из пикселей, имеющих большой динамический диапазон яркости. Система воспроизведения изображений, как правило, не способна правильно отобразить столь большой диапазон значений интенсивности, что приводит к тому, что при обычном отображении спектра Фурье теряется значительное число деталей. В этой связи для улучшения зрительного восприятия полутонов изображение спектра подвергается логарифмическому преобразованию.

На рис. 3.1 приведено изображение, представляющее собой белый прямоугольник размерами 20×40 пикселей, наложенный на черный фон размерами 512×512 пикселей, а также представлен центрированный Фурье-спектр изображения. Для того чтобы центрировать спектр, необходимо исходное изображение умножить на $(-1)^{x+y}$ перед вычислением Фурье-преобразования.

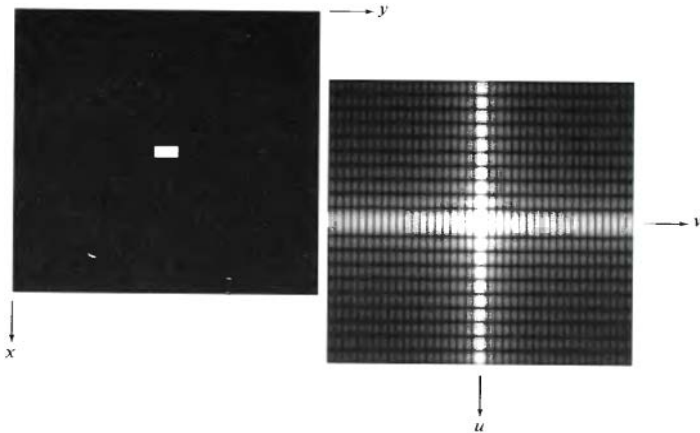


Рис. 3.1. Слева – изображение белого прямоугольника размерами 20×40 пикселей на черном фоне размерами 512×512 пикселей, справа – центрированный Фурье-спектр, представленный после применения логарифмического преобразования

3.2 Основы частотной фильтрации

Частотная область цифрового изображения представляет собой не что иное, как пространство, в котором принимают значения

переменные (u, v) Фурье-преобразования. Как известно, частота сигнала прямо связана со скоростью изменения сигнала, поэтому интуитивно понятно, что частоты в Фурье-преобразовании связаны с вариацией яркости на изображении. Наиболее медленно меняющаяся (постоянная) частотная составляющая ($u = 0, v = 0$) совпадает со средней яркостью изображения. Низкие частоты, соответствующие точкам вблизи начала координат Фурье-преобразования, определяют медленно меняющиеся компоненты изображения. По мере удаления от начала координат более высокие частоты начинают соответствовать все более и более быстрым изменениям яркости, представляющим собой границы объектов и другие детали изображения, характеризующиеся резкими изменениями яркости, например шум на изображении.

Процедура фильтрации изображения в частотной области состоит из следующих шагов:

1. Исходное изображение умножается на $(-1)^{x+y}$, чтобы его Фурье-преобразование оказалось центрированным.

2. Вычисляется прямое дискретное преобразование Фурье (ДПФ) $F(u, v)$ исходного изображения.

3. Функция $F(u, v)$ умножается на функцию фильтра $H(u, v)$.

4. Вычисляется обратное ДПФ от результата шага 3.

5. Выделяется вещественная часть результата шага 4.

6. Результат шага 5 умножается на $(-1)^{x+y}$.

Функция фильтра $H(u, v)$ или передаточная характеристика фильтра подавляет некоторые частоты преобразования, оставляя другие при этом без изменения. На рис. 3.2 представлена структурная схема, включающая основные этапы проведения фильтрации изображения в частотной области.

На стадии предварительной обработки, помимо умножения изображения на $(-1)^{x+y}$, могут применяться операции яркостного масштабирования, нормировка размеров исходного изображения, преобразование формата входных данных в формат с плавающей точкой и ряд других.

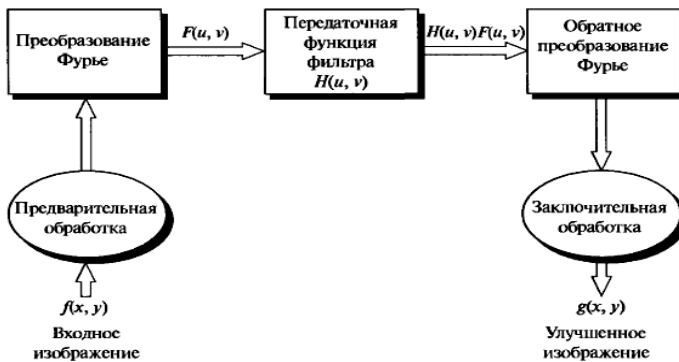


Рис. 3.2. Основные этапы фильтрации изображения в частотной области

3.3 Основные частотные фильтры и их свойства

Основными видами используемых фильтров являются: низкочастотный фильтр – ослабляет высокие частоты, одновременно пропуская низкие частоты; высокочастотный фильтр – фильтр, обладающий противоположными свойствами. Низкие частоты Фурье-преобразования отвечают за возникновение превалирующих значений яркости на гладких участках изображения, в то время как высокие частоты отвечают преимущественно за контуры и шум. После применения низкочастотной фильтрации изображение, по сравнению с исходным изображением, содержит меньше резких деталей. После применения высокочастотной фильтрации на изображении уменьшаются изменения яркости в пределах больших гладких областей и выделяются переходные зоны быстрого изменения яркости, то есть контуры изображения. Как правило, такое изображение обладает большей резкостью по сравнению с исходным.

В силу того, что высокочастотные фильтры почти полностью подавляют постоянную составляющую $F(0, 0)$, которая определяет среднюю яркость изображения, выходное изображение после обработки таким фильтром выглядит очень темным. Для устранения указанного недостатка к передаточной функции фильтра добавляют константу, равную половине высоты фильтра.

Другим видом частотного фильтра является фильтр-пробка, или режекторный фильтр, вырезающий определенное значение яркости из изображения. Как правило, это значение яркости в точке

начала координат – среднее значение яркости изображения. Среднее значение яркости изображения не может быть строго равно нулю, поскольку для этого некоторые элементы изображения должны содержать отрицательные значения, а средства отображения графической информации не могут оперировать с отрицательными значениями яркости. Для устранения указанного противоречия наименьшее отрицательное значение приравнивают к нулю (уровень черного), а остальные значения пропорционально увеличивают.

3.4 Сглаживающие частотные фильтры

Как отмечалось ранее, контуры и другие резкие перепады яркости на изображении, включая шумы, вносят значительный вклад в высокочастотные составляющие его Фурье-преобразования. Сглаживание изображения в частотной области достигается путем ослабления высокочастотных компонент определенного диапазона Фурье-образа данного изображения.

Модель фильтрации изображения в частотной области в обобщенном виде может быть описана следующим равенством:

$$G(u, v) = H(u, v) \cdot F(u, v),$$

где $F(u, v)$ – Фурье-образ изображения, которое подлежит операции фильтрации, $H(u, v)$ – передаточная функция фильтра, которая ослабит высокочастотные компоненты $F(u, v)$ и сформирует функцию $G(u, v)$.

Рассмотрим три вида низкочастотных фильтров: идеальный фильтр, фильтр Баттерворта и гауссов фильтр. Указанные фильтры по виду передаточной функции покрывают диапазон от очень резких фильтров (идеальный фильтр) до очень гладких фильтров (гауссов фильтр).

Фильтр Баттерворта является единственным из рассматриваемых, который характеризуется порядком фильтра – параметром, определяющим крутизну передаточной функции фильтра. При малых значениях порядка фильтра передаточная функция имеет гладкую форму, близкую по форме к передаточной характеристике гауссова фильтра; при высоких значениях порядок фильтра характеризуется крутой передаточной функцией и приближается по форме к характеристикам идеального фильтра.

Идеальные фильтры низких частот

Идеальный фильтр низких частот представляют собой такой низкочастотный фильтр, который обрезает все высокочастотные составляющие Фурье-образа, находящиеся на большем расстоянии от начала координат централизованного изображения, чем некоторое заданное расстояние D_0 . Такой тип фильтра еще называется двумерным идеальным низкочастотным фильтром и имеет передаточную функцию:

$$H(u, v) = \begin{cases} 1, & \text{при } D(u, v) \leq D_0 \\ 0, & \text{при } D(u, v) > D_0 \end{cases},$$

где D_0 – заданная неотрицательная величина, точка профиля фильтра, в которой совершается переход от значений $H(u, v) = 1$ к значениям $H(u, v) = 0$, называемая частотой среза; $D(u, v)$ – расстояние от точки (u, v) до начала координат – центра частотного прямоугольника. В силу того, что Фурье-преобразование централизовано, расстояние $D(u, v)$ определяется по формуле:

$$D(u, v) = [(u - m/2)^2 + (v - n/2)^2]^{0.5},$$

где m и n – размеры исходного изображения.

На рис. 3.3 представлен идеальный низкочастотный фильтр в виде изображения, а также радиальный профиль фильтра. Идеальность фильтра подчеркивается тем, что все частоты внутри круга радиуса D_0 проходят без изменения, в то время как частоты вне круга подавляются полностью.

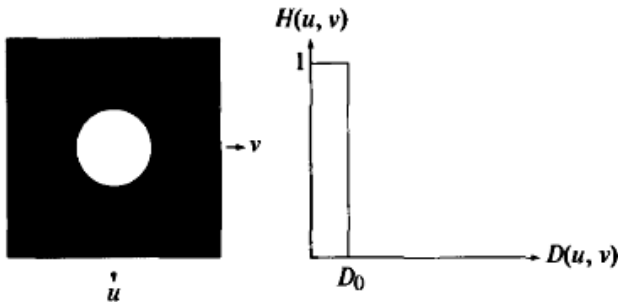


Рис. 3.3. Слева: представление идеального низкочастотного фильтра в виде изображения, справа: радиальный профиль фильтра

Низкочастотный идеальный фильтр обладает свойством симметрии относительно начала координат, поэтому для того чтобы однозначно задать фильтр, достаточно построить один радиальный профиль – функцию расстояния от начала координат. Передаточная функция фильтра в координатах $H(u, v)$ получается вращением профиля на 360° вокруг начал координат.

Одним из способов выбора эталонного набора положений частот среза состоит в том, чтобы определить круги, в которых заключена заданная часть полной энергии изображения P_T . Полная энергия будет определена как сумма компонент энергетического спектра во всех точках (u, v) ; $u = 0, 1, 1 \dots M - 1$; $v = 0, 1, 1 \dots N - 1$.

$$P_T = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} P(u, v),$$

где величина $P(u, v)$ определяется как:

$$P(u, v) = |F(u, v)|^2,$$

где $F(u, v)$ – Фурье-преобразование исходного изображения $f(x, y)$.

Частота $r(\alpha)$ определятся как радиус круга с центром в центре частотного прямоугольника, содержащего α процентов энергии спектра изображения, т.е.:

$$\alpha = 100 \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} \frac{P(u, v)}{P_T}.$$

Выбор неоптимального значения частоты среза фильтра может привести к значительным искажениям обрабатываемого изображения: выбор слишком малого значения частоты среза приведет к устранению большинства деталей изображения, в то время как выбор относительно большого значения частоты среза приведет к тому, что выходное изображение будет незначительно отличаться от исходного, при этом уровень шума также не изменится.

При использовании идеального фильтра низких частот возникают нежелательные эффекты, такие как эффекты размывания и появления ложных контуров, при этом при уменьшении ширины используемого фильтра в частотной области эффекты размытия изображений усиливаются.

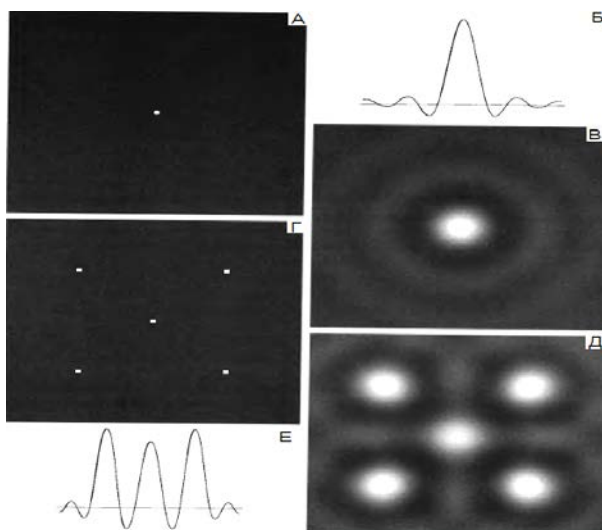


Рис. 3.4. А – изображение идеального фильтра низких частот, Б – профиль яркости в пространственной области, В – изображение идеального низкочастотного фильтра в пространственной области, Г – условное изображение, состоящее из 5 ярких точек в пространственной области, Д – свертка изображений В и Г, Е – профиль яркости отфильтрованного изображения Д

Возникающие нежелательные эффекты при использовании идеального фильтра низких частот получили название «звон» (или эффект Гиббса), выражающийся в появлении ложных контуров вокруг реальных контуров. Структура ложных контуров становится тоньше по мере уменьшения энергии отсекаемой высокочастотной составляющей.

Природа появления ложных контуров может быть объяснена при помощи теоремы о свертке, согласно которой существует пространственный аналог идеального фильтра низких частот. Пространственная функция такого фильтра может быть определена с помощью обратного преобразования Фурье от передаточной функции низкочастотного фильтра и представляет собой последовательность круговых концентрических колец различной яркости, приводящих главным образом к появлению ложных контуров: рис. 3.4 иллюстрирует проявление эффекта Гиббса при использовании идеального фильтра низких частот.

Под понятием профиль яркости подразумевается зависимость изменения яркости пикселей от номера пикселей, расположенных на горизонтальной прямой, как правило, проходящей через центр изображения; отсчет пикселей по прямой начинается с крайнего левого пикселя и заканчивается крайним правым пикселем.

Таким образом, приведенные результаты фильтрации с помощью идеального низкочастотного фильтра показывают, что идеальные фильтры низких частот не имеют большого практического применения.

Фильтры низких частот Баттерворта

Передачная функция низкочастотного фильтра Баттерворта порядка n с частотой среза на расстоянии D_0 от начала координат задается формулой:

$$H(u, v) = \frac{1}{1 + [D(u, v) / D_0]^{2n}}.$$

Радиальные профили передачной функции низкочастотного фильтра Баттерворта в зависимости от порядка фильтра представлены на рис. 3.5.

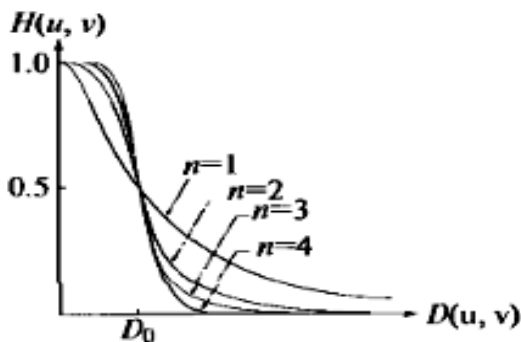


Рис. 3.5. Радиальные профили передаточной функции низкочастотного фильтра Баттерворта в зависимости от порядка фильтра

В отличие от идеального низкочастотного фильтра передаточная функция низкочастотного фильтра Баттерворта не имеет раз-

рыва, который устанавливает точную границу между пропускаемыми и обрезаемыми частотами. К преимуществам низкочастотных фильтров Баттерворта относится намного меньшее проявление нежелательных эффектов размытия и появления ложных контуров по сравнению с идеальными низкочастотными фильтрами. С увеличением порядка низкочастотного фильтра Баттерворта возрастает проявление эффектов размытия. Принято считать, что низкочастотный фильтр Баттерворта второго порядка является оптимальным с точки зрения компромисса между эффективностью низкочастотной фильтрации и приемлемым уровнем проявления ложных контуров и общего размытия изображения.

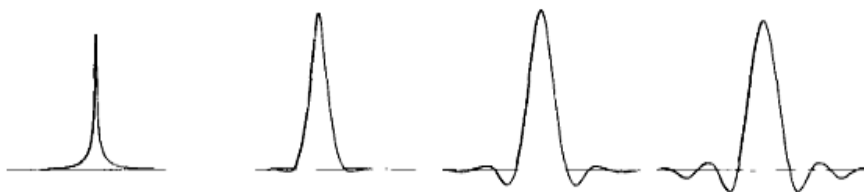


Рис. 3.6. Слева направо: профили яркости низкочастотных фильтров Баттерворта с одинаковой частотой среза порядка 1, 2, 5 и 20 соответственно

На рис. 3.6 приведены профили яркости изображений низкочастотных фильтров Баттерворта с одинаковой частотой среза порядка 1, 2, 5 и 20 соответственно.

Гауссовы фильтры низких частот

В двумерном случае гауссовы фильтры низких частот задаются формулой:

$$H(u, v) = e^{-\frac{D^2(u, v)}{2\sigma^2}}.$$

При величине $D(u, v) = D_0$ (D_0 – частота среза фильтра) значение передаточной функции фильтра составляет 0,667 от своего максимального значения. На рис. 3.7 приведены радиальные профили гауссова фильтра при различных значениях D_0 .

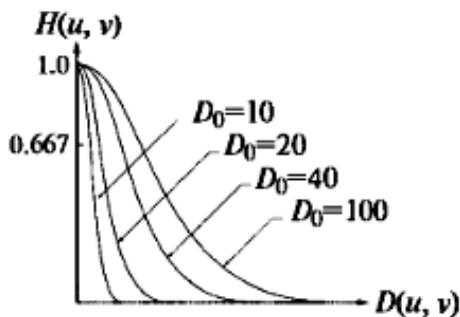


Рис. 3.7. Радиальные профили передаточной функции низкочастотного гауссова фильтра при различных значениях частоты среза D_0

Как известно, обратное преобразование Фурье от гауссовой функции является также гауссовой функцией. Пространственный гауссов фильтр низких частот, полученный с помощью обратного преобразования Фурье от передаточной функции частотного гауссова фильтра, будет положительным и не будет иметь концентрических колец, что приведет к полному отсутствию проявлений эффекта Гиббса на обрабатываемом изображении. Что является основным преимуществом гауссовых низкочастотных фильтров по сравнению с фильтрами Баттерворта. Тем не менее применение низкочастотного гауссова фильтра обеспечивает зачастую меньшее сглаживание при одинаковой частоте среза по сравнению с фильтрами Баттерворта, поэтому в тех случаях, когда требуется жесткий контроль переходной зоны от низких частот к высоким частотам, фильтры Баттерворта представляются более подходящим выбором. В задачах обработки биомедицинских изображений отсутствие ложных контуров и недопустимость проявления артефактов любого рода предпочтительным является использование гауссовых фильтров.

3.5 Частотные фильтры повышения резкости

Контурные и другие скачкообразные изменения яркости на изображении связаны с высокочастотными составляющими Фурье-преобразования изображения. Повышение резкости изображения может быть достигнуто в частотной области при помощи процедуры

высокочастотной фильтрации, которая подавляет низкочастотные составляющие и не затрагивает высокочастотную часть Фурье-преобразования.

Передаточная функция высокочастотных фильтров может быть получена при помощи следующего соотношения:

$$H_{hp}(u, v) = 1 - H_{lp}(u, v),$$

где $H_{lp}(u, v)$ – передаточная функция соответствующего низкочастотного фильтра.

Рассмотрим следующие виды высокочастотных фильтров: идеальные высокочастотные фильтры, высокочастотные фильтры Баттерворта и гауссовы высокочастотные фильтры.

Идеальные фильтры высоких частот

Двумерные идеальные высокочастотные фильтры определяются формулой:

$$H(u, v) = \begin{cases} 0, & \text{при } D(u, v) \leq D_0, \\ 1, & \text{при } D(u, v) > D_0 \end{cases}$$

где D_0 – частота среза, $D(u, v)$ – расстояние от точки с координатами (u, v) до центра частотного прямоугольника (начала координат).

На рис. 3.8 приведен радиальный профиль передаточной функции идеального высокочастотного фильтра, функция идеального высокочастотного фильтра в пространственной области и соответствующий профиль яркости.

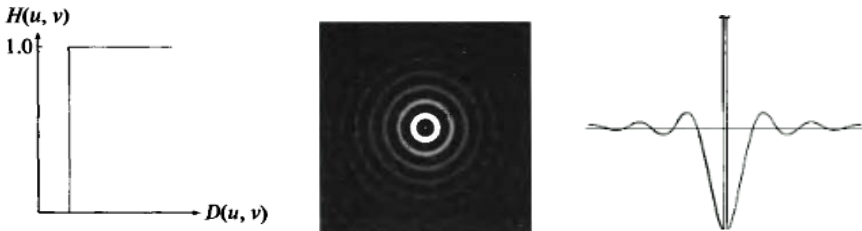


Рис. 3.8. Слева направо: радиальный профиль передаточной функции идеального высокочастотного фильтра; функция идеального высокочастотного фильтра в пространственной области; соответствующий профиль яркости

Идеальным высокочастотным фильтром, так же как идеальным низкочастотным фильтром, присущ эффект Гиббса, который приводит к появлению «звона» на обрабатываемом изображении.

По аналогии с идеальными низкочастотными фильтрами идеальные высокочастотные фильтры также почти не имеют реального практического применения.

Фильтры высоких частот Баттерворта

Передаточная функция высокочастотного фильтра Баттерворта порядка n с частотой среза на расстоянии D_0 от начала координат определяется выражением:

$$H(u, v) = \frac{1}{1 + [D_0 / D(u, v)]^{2n}}.$$

На рис. 3.9 приведены радиальный профиль передаточной функции высокочастотного фильтра Баттерворта второго порядка, функция высокочастотного фильтра Баттерворта в пространственной области, а также профиль яркости пространственной функции фильтра.

Фильтры Баттерворта высоких частот приводят к гораздо меньшим искажениям границ объектов, чем идеальные высокочастотные фильтры Баттерворта. С увеличением порядка высокочастотного фильтра Баттерворта искажения границ объектов заметно увеличиваются.

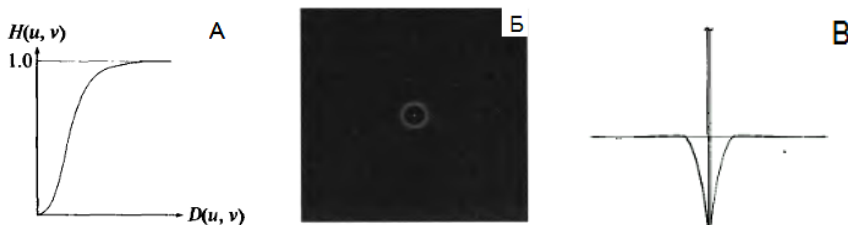


Рис. 3.9. А – радиальный профиль передаточной функции высокочастотного фильтра Баттерворта второго порядка, Б – функция высокочастотного фильтра Баттерворта в пространственной области, В – профиль яркости пространственной функции фильтра

Гауссовы фильтры высоких частот

Передаточная функция гауссова фильтра высоких частот с частотой среза, расположенной на расстоянии D_0 от начала координат, задается выражением:

$$H(u, v) = 1 - e^{-\frac{D^2(u, v)}{2D_0^2}}.$$

На рис. 3.10 приведены радиальный профиль передаточной функции гауссова фильтра высоких частот, функция высокочастотного гауссова фильтра в пространственной области, а также профиль яркости пространственной функции гауссова фильтра.

Высокочастотный гауссов фильтр обеспечивает высокое качество фильтрации даже для маленьких объектов и тонких полос, характеризуется полным отсутствием искажений.

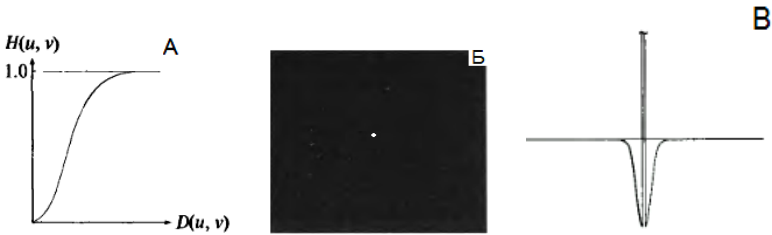


Рис. 3.10. А – радиальный профиль передаточной функции высокочастотного гауссова фильтра, Б – функция высокочастотного гауссова фильтра Баттерворта в пространственной области, В – профиль яркости пространственной функции фильтра

3.6 Нерезкое маскирование. Высокочастотная фильтрация с подъемом частотной характеристики

Изображения, получаемые в результате высокочастотной фильтрации, имеют среднее значение яркости фона, близкое к нулю, так как высокочастотные фильтры, применяемые для обработки данных изображений, уничтожают постоянную составляющую (нулевую компоненту) их Фурье-преобразования. Для устранения данного недостатка можно к результату фильтрации добавить некоторую долю исходного изображения в качестве так называемой «под-

ложки изображения». В ряде случаев представляется необходимым увеличить вклад, приносимый исходным изображением в итоговый результат фильтрации. Такой подход называется методом высокочастотной фильтрации с подъемом частотной характеристики и представляет собой обобщение метода нерезкого маскирования.

Нерезкое маскирование состоит в формировании резкого изображения путем вычитания из оригинала его сглаженной копии. В терминах частотных преобразований это означает, что имеет место высокочастотная фильтрация, которая достигается вычитанием из исходного изображения $f(x, y)$ результата его низкочастотной фильтрации $f_{lp}(x, y)$:

$$f_{hp}(x, y) = f(x, y) - f_{lp}(x, y).$$

Обобщением последнего выражения является следующее выражение, описывающее высокочастотную фильтрацию с подъемом частотной характеристики:

$$f_{hb}(x, y) = A \cdot f(x, y) - f_{lp}(x, y).$$

Фильтрация с подъемом частотной характеристики позволяет изменять вклад, вносимый исходным изображением в конечный результат обработки.

Фильтрация с усилением высоких частот

При решении некоторых задач обработки изображений выгодно усилить его высокочастотную составляющую. Передаточная функция фильтра с усилением высоких частот задается выражением:

$$H_{hfe}(u, v) = a + b \cdot H_{lp}(u, v),$$

где $H_{lp}(u, v)$ – передаточная функция низкочастотного фильтра.

Характерные значения a находятся в диапазоне от 0,25 до 0,5; характерные значения b – в диапазоне от 1,5 до 2,5 (при этом важно отметить, что $b > a$). При значениях $b > 1$ происходит усиление высоких частот, откуда и происходит название метода.

Основная сфера применения фильтрации с усилением высоких частот в задачах обработки биомедицинских изображений является повышение резкости рентгенографических изображений. Рентгеновские лучи не могут быть сфокусированы с помощью линз, как световые лучи, поэтому большинство рентгеновских снимков выглядят расплывчато. Зачастую яркость рентгенографических изо-

бражений сдвинута в темную область, поэтому также необходимо использовать методы обработки изображения для увеличения яркости контрастности изображения.

3.7 Реализация методов частотной фильтрации в среде MATLAB

Для выполнения прямого преобразования Фурье в пакете MATLAB используется функция **fft2**, которая имеет следующий синтаксис:

$$\mathbf{F}=\text{fft2}(\mathbf{f})$$

где **f** – исходное изображение, **F** – преобразование Фурье двумерной функции **f(x, y)**.

Для получения спектра изображения необходимо выполнить следующую команду:

$$\mathbf{S}=\text{abs}(\mathbf{F})$$

Команда **abs** вычисляет абсолютную величину (модуль) комплексной функции **F**.

Для смещения начала координат Фурье-преобразования в центр частотной области используется функция **fftshift** со следующим синтаксисом:

$$\mathbf{Fc}=\text{fftshift}(\mathbf{F})$$

где **Fc** – центрированное Фурье-преобразование.

В силу того, что динамический диапазон изображения Фурье-преобразования очень широкий, необходимо выполнять операции преобразования диапазона яркости для корректного отображения Фурье-преобразования на дисплее. Данную операцию можно выполнить с помощью логарифмического преобразования следующим образом:

$$\mathbf{S2}=\text{log}(1+\text{abs}(\mathbf{Fc}))$$

Обратное преобразование Фурье выполняется с помощью функции **ifft2**, которая имеет следующий синтаксис:

$$\mathbf{f}=\text{ifft2}(\mathbf{F})$$

где **F** – преобразование Фурье, **f** – соответствующее изображение.

После выполнения обратного Фурье-преобразования необходимо выделить вещественную часть результата с помощью команды следующего вида:

$$\mathbf{f}=\text{real}(\text{ifft2}(\mathbf{F}))$$

В среде MATLAB существуют специальные функции для составления матриц и векторов. Простейшим способом задать одномерный массив (вектор) является запись команды следующего вида:

$$[\mathbf{name}]=\mathbf{X1:dX:Xk}$$

где **name** – имя переменной, которое будет присвоено сформированному массиву, **X1** – значение первого элемента массива, **Xk** – значение последнего элемента, **dX** – шаг, с помощью которого формируется каждый следующий элемент (по умолчанию равен 1). Данная запись формирует вектор-строку. Для того чтобы сформировать вектор-столбец, необходимо использовать следующую запись:

$$[\mathbf{name}]=[\mathbf{X1:dX:Xk}]'$$

где символ ' обозначает транспонирование матрицы.

При решении задач цифровой обработки изображений необходимо формировать двумерные массивы координатной сетки. С этой целью в системе MATLAB удобно использовать функцию **meshgrid**, которая формирует выходные массивы, определяющие координаты узлов прямоугольника, задаваемого входными векторами. Формируемый прямоугольник задает область определения функции от двух переменных. Функция **meshgrid** имеет следующий синтаксис:

$$[\mathbf{X}, \mathbf{Y}]=\mathbf{meshgrid}(x, y)$$

Строки матрицы **X** являются копиями вектора x , а столбцы матрицы **Y** – копиями вектора y . Формирование таких массивов упрощает вычисление функций двух переменных, позволяя применять операции над массивами. Пример: пусть $x = (0, 1, 2)$, $y = (0, 1)$. Тогда

$$X = \begin{pmatrix} 0 & 1 & 2 \\ 0 & 1 & 2 \end{pmatrix}, Y = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}. \text{ Формирование таких массивов}$$

упрощает вычисление функций двух переменных, позволяя применять операции над массивами.

Таким образом, для осуществления частотной фильтрации исходного изображения **f(x, y)** необходимо выполнить следующую последовательность операций:

- 1) определить размеры изображения **f(x, y)** с помощью команды **size**; сформировать векторы частотных координат **u** и **v**, размеры которых равны размерам векторов соответствующих пространственных координат **x** и **y**;

2) выполнить прямое преобразование Фурье исходного изображения с помощью команды **fft2** и получить частотное представление $\mathbf{F}(\mathbf{u}, \mathbf{v})$;

3) задать значение частоты среза в пикселях (как правило, определяется единицами процентов от максимального горизонтального или вертикального размера изображения); сформировать передаточную характеристику требуемого фильтра $\mathbf{H}(\mathbf{u}, \mathbf{v})$, представляющую собой двумерную матрицу, размеры которой совпадают с размерами матрицы исходного изображения;

4) получить частотное представление отфильтрованного изображения $\mathbf{G}(\mathbf{u}, \mathbf{v})$ путем выполнения операции поэлементного умножения передаточной характеристики фильтра $\mathbf{H}(\mathbf{u}, \mathbf{v})$ и Фурье-преобразования исходного изображения $\mathbf{F}(\mathbf{u}, \mathbf{v})$;

5) получить пространственное представление отфильтрованного изображения $\mathbf{g}(\mathbf{x}, \mathbf{y})$ путем выполнения процедуры обратного преобразования Фурье от частотного представления $\mathbf{G}(\mathbf{u}, \mathbf{v})$ с помощью команды **ifft2**;

6) выделить вещественную часть результата, полученного на предыдущем этапе с помощью команды **real**.

3.8 Практическое задание № 3

1. Загрузите изображение рентгеновского снимка грудной клетки **xraychest.jpg**. Выполните Фурье-преобразование исходного изображения, получите изображение спектра в частотной области. Для улучшения визуального восприятия воспользуйтесь градационным логарифмическим преобразованием.

2. Примените методику высокочастотной фильтрации с подъемом частотной характеристики исходного изображения рентгеновского снимка грудной клетки **xraychest.jpg**.

2.1. Выполните операцию низкочастотной фильтрации исходного изображения рентгеновского снимка грудной клетки **xraychest.jpg**.

2.2. Подбором параметра A добейтесь наилучшего результата повышения резкости с помощью подъема частотной характеристики.

2.3. Изображение, полученное в результате выполнения пункта 2.2, подвергните процедуре гистограммной эквализации. Для получения справочной информации системы MATLAB по выполнению данной процедуры используйте команду **doc histeq**.

3. Примените методику фильтрации с усилением высоких частот к исходному изображению рентгеновского снимка грудной клетки **xraychest.jpg** с целью повышения резкости изображения.

3.1. Выполните операцию высокочастотной фильтрации исходного изображения рентгеновского снимка грудной клетки **xraychest.jpg**.

3.2. Подбором параметров **a** и **b** добейтесь усиления высоких частот.

3.3. Изображение, полученное в результате выполнения пункта 3.2, подвергните процедуре гистограммной эквализации.

4. Сравните изображения, полученные в результате выполнения второго и третьего пунктов задания. Сделайте выводы о полученных результатах.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Двумерное Фурье-преобразование изображений.
2. Основы частотной фильтрации изображений.
3. Сглаживающие частотные фильтры.
4. Частотные фильтры повышения резкости.
5. Нерезкое маскирование. Высокочастотная фильтрация с подъемом частотной характеристики.
6. Фильтрация с усилением высоких частот.

ГЛАВА 4 ВОССТАНОВЛЕНИЕ МЕДИЦИНСКИХ ИЗОБРАЖЕНИЙ В СРЕДЕ MATLAB

Конечной целью восстановления изображений является повышение качества в некотором заранее определенном смысле. В отличие от процедур улучшения изображений, процесс восстановления изображений имеет в основном объективный характер. При восстановлении изображений делается попытка реконструировать или воссоздать изображение, которое было до этого искажено, используя априорную информацию о явлении, которое вызвало ухудшение изображения. В силу этого методы восстановления изображений основаны на моделировании процессов искажения и применения обратных процедур для воссоздания исходного изображения. Данный подход обычно включает в себя разработку критериев качества, которые дают возможность объективно оценить полученный результат.

4.1 Модель процесса искажения/восстановления изображения

На рис. 4.1 представлена модель процесса искажения/восстановления изображения.

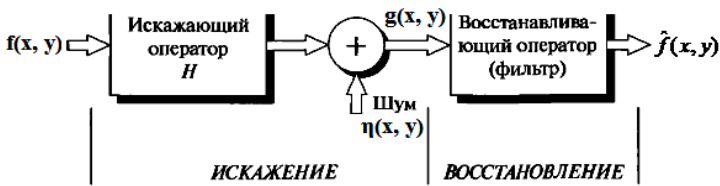


Рис. 4.1. Модель процесса искажения/восстановления изображения

Модель процесса искажения предполагает действие некоторого искажающего оператора H на исходное изображение $f(x, y)$, что после добавления аддитивного шума дает искаженное изображение $g(x, y)$. Задача восстановления состоит в построении некоторого приближения $\hat{f}(x, y)$ исходного изображения по заданному (т.е. искаженному) изображению $g(x, y)$, некоторой информации относительно искажающего оператора H , а также информации относительно аддитивного шума $\eta(x, y)$. Необходимо достичь того, чтобы при-

ближение $\hat{f}(x, y)$ было как можно ближе к исходному изображению $f(x, y)$, что достигается при наличии максимальной информации об операторе H и о функции $\eta(x, y)$. Основу подхода составляет использование операторов или фильтров, восстанавливающих изображение.

Искаженное изображение может быть представлено в пространственной области в виде:

$$g(x, y) = h(x, y) * f(x, y) + \eta(x, y),$$

где $h(x, y)$ – функция, представляющая искажающий оператор в пространственной области.

В частотной области искаженное изображение может быть представлено как:

$$G(u, v) = H(u, v) \cdot F(u, v) + N(u, v).$$

Входящие в данное выражение частотные функции являются Фурье-образами соответствующих функций в пространственной области.

4.2 Модели шума

Основными источниками шума на цифровом изображении являются сам процесс его получения (оцифровки), а также процесс передачи. Работа сенсоров зависит от различных факторов, таких как внешние условия в процессе получения изображения и качество самих сенсоров. Например, в процессе получения изображения с помощью ПЗС – (прибор с зарядовой связью) матриц, основными факторами, влияющими на величину шума, являются уровень освещенности и температура сенсоров. В процессе передачи изображения могут искажаться помехами, возникающими в каналах связи. Например, при передаче изображений по беспроводным сетям оно может быть искажено в результате воздействия различных атмосферных возмущений.

При рассмотрении шумов важными являются параметры, определяющие пространственные характеристики шума, а также наличие или отсутствие корреляции между шумом и изображением. Под частотными характеристиками шума понимаются свойства его Фурье-спектра. Например, шум, спектр которого является постоянной

величиной, называется обычно «белым шумом». Происхождение названия данного вида шума связано с физическими свойствами белого света, который содержит практически все частоты видимого спектра в равных пропорциях. Для упрощения рассматриваемых задач и с учетом незначительной потери в качестве их решений будем в дальнейшем полагать, что шумы изображения не зависят от пространственных координат и не коррелируют с самим изображением.

Наиболее часто модели шумов, присутствующих на изображениях, представляют в пространственной области. Данное представление основано на статистических свойствах значений яркости компонентов шума. Эти значения яркости могут рассматриваться как случайные величины, характеризующиеся функцией плотности распределения вероятностей.

Пусть задана некоторая случайная переменная Z . Функция распределения вероятностей $F(Z)$ случайной величины Z представляет собой функцию, значение которой для каждого z является вероятностью события, заключающегося в том, что случайная величина Z принимает значения меньше переменной z , т.е. функцию вида $F(z) = P(Z < z)$. Данная функция представляет собой неубывающую функцию переменной z , изменяющуюся от $F(-\infty) = 0$ до $F(+\infty) = 1$. Для случайной величины с непрерывной и дифференцируемой функцией распределения $F(z)$ можно найти дифференциальный закон распределения вероятностей, выражаемый как производная от функции $F(z)$, т.е. как $p(z) = F'(z)$. Данная зависимость называется функцией плотности распределения вероятностей.

Далее будут рассмотрены примеры функций плотности распределения вероятностей применительно к наиболее распространенным моделям шумов, встречающихся при обработке цифровых изображений.

Гауссов шум

Математическая простота, характерная для работы с моделями гауссова шума (или нормального шума) как в пространственной, так и в частотной области, обусловила широкое распространение этих моделей на практике.

Функция плотности распределения вероятностей гауссовой случайной величины z задается выражением:

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(z-\mu)^2}{2\sigma^2}},$$

где z – значение яркости, μ – среднее значение случайной величины z , σ – среднеквадратичное отклонение случайной величины z .

На рис. 4.2 А приведен график функции плотности распределения гауссова шума.

Шум Релея

Функция плотности распределения вероятностей шума Релея задается выражением:

$$p(z) = \begin{cases} \frac{2}{b}(z-a)e^{-\frac{(z-a)^2}{b}}, & \text{при } z \geq a \\ 0, & \text{при } z < a \end{cases}.$$

Среднее и дисперсия для этого распределения имеют вид:

$$\mu = a + \sqrt{\pi b/4}, \quad \sigma^2 = \frac{b(4-\pi)}{4}.$$

На рис. 4.2 Б приведен график функции плотности распределения шума Релея.

Шум Эрланга (гамма-шум)

Функция плотности распределения вероятностей шума Эрланга определяется как:

$$p(z) = \begin{cases} \frac{a^b z^{b-1}}{(b-1)!} e^{-az}, & \text{при } z \geq 0 \\ 0, & \text{при } z < 0 \end{cases},$$

где $a > 0$, b – положительное целое число.

Среднее и дисперсия для этого распределения имеют вид:

$$\mu = \frac{b}{a}, \quad \sigma^2 = \frac{b}{a^2}.$$

На рис. 4.2 В приведен график функции плотности распределения шума Эрланга.

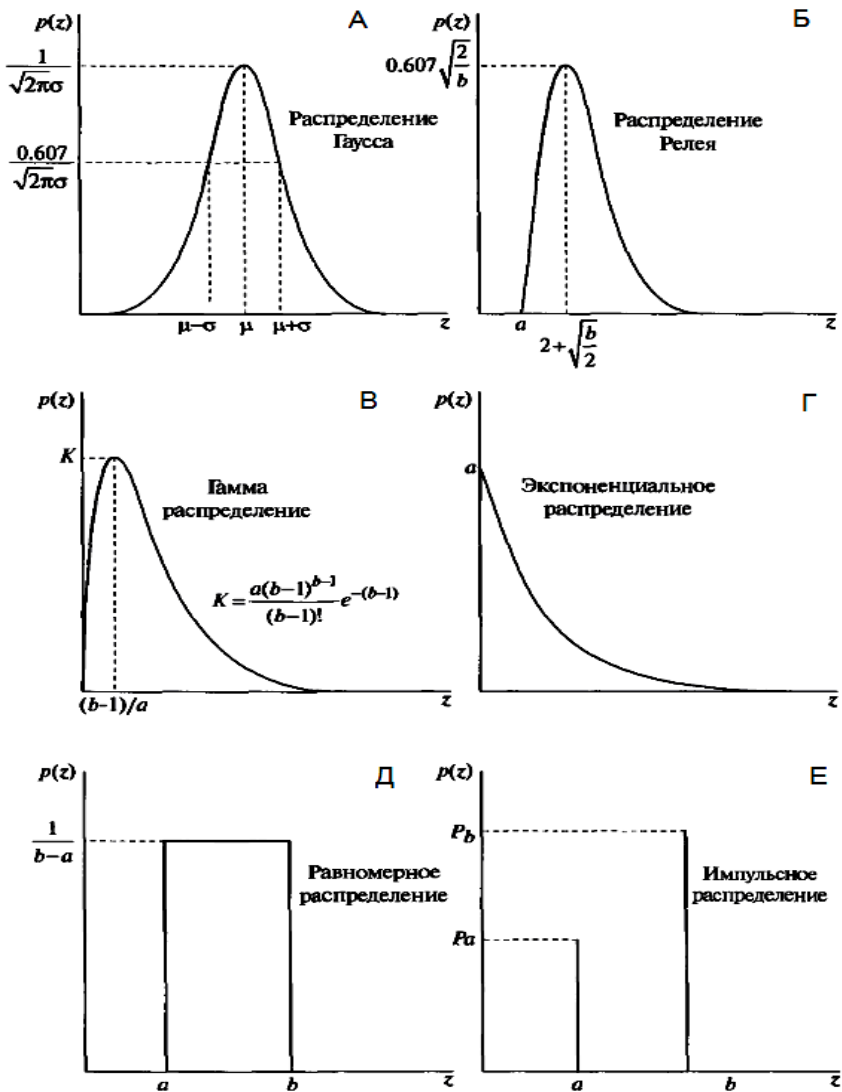


Рис. 4.2. Функции плотности распределения вероятностей основных видов шумов

Экспоненциальный шум

Функция плотности распределения вероятностей экспоненциального шума задается выражением:

$$p(z) = \begin{cases} ae^{-az}, & \text{при } z \geq 0 \\ 0, & \text{при } z < 0 \end{cases},$$

где $a > 0$.

Среднее значение и дисперсия для этого распределения имеют вид:

$$\mu = \frac{1}{a}, \quad \sigma^2 = \frac{1}{a^2}.$$

На рис. 4.2 Г приведен график функции плотности распределения экспоненциального шума. Заметим, что экспоненциальное распределение является частным случаем распределения Эрланга при $b = 1$.

Равномерный шум

Функция плотности распределения вероятностей равномерного шума задается выражением:

$$p(z) = \begin{cases} \frac{1}{b-a}, & \text{при } a \leq z \leq b \\ 0, & \text{в остальных случаях} \end{cases}.$$

Среднее значение и дисперсия равномерного распределения равны:

$$\mu = \frac{a+b}{2}, \quad \sigma^2 = \frac{(b-a)^2}{12}.$$

На рис. 4.2 Д приведен график функции плотности распределения равномерного шума.

Импульсный шум

Функция плотности распределения вероятностей (биполярного) импульсного шума определяется выражением:

$$p(z) = \begin{cases} P_a, & \text{при } z = a \\ P_b, & \text{при } z = b \\ 0, & \text{в остальных случаях} \end{cases}.$$

Если $b > a$, то пиксель с яркостью b выглядит как светлая точка на изображении. Пиксель с яркостью a выглядит, наоборот, как темная точка. Если одно из значений вероятности (P_a или P_b) равно нулю, то импульсный шум называется униполярным. Часто данный вид шума называют шумом типа «соль и перец».

Значения импульсов шума могут быть как положительные, так и отрицательные. При оцифровке изображения обычно происходит масштабирование и ограничение значений яркости. Поскольку величина связанных с импульсным шумом искажений, как правило, велика по сравнению с величиной полезного сигнала, импульсный шум после оцифровки принимает экстремальные значения, что соответствует появлению абсолютно черных и белых точек на изображении. Обычно предполагается, что значения a и b являются интенсивностями шума в том смысле, что они равны максимальному и минимальному значениям, которые в принципе могут присутствовать в оцифрованном изображении. В результате отрицательные импульсы выглядят как черные точки на изображении, положительные импульсы – как белые точки. На рис. 4.2 Е представлен график функции плотности распределения импульсного шума.

4.3 Построение оценок для параметров шумов

Параметры функции плотности распределения вероятностей шума могут быть частично известны исходя из технических характеристик сенсоров, однако зачастую необходимо оценивать эти параметры для конкретной системы, использованной при получении данного изображения. В этом случае одним из простейших способов изучения характеристик системы регистрации изображений, связанных с шумами, является получение набора изображений однородных тестовых объектов. В случае обычной оптической системы таким объектом целесообразно выбрать сплошную равномерно освещенную серую область. Полученные таким образом изображения достаточно полно характеризуют шумы системы.

В тех случаях, когда доступны исключительно изображения, ранее сформированные системой, рассмотрение гистограмм небольших участков изображения примерно постоянной яркости позволяет оценить параметры функции плотности распределения вероятностей шума. Для рассматриваемых участков изображения вычисляют среднее значение и дисперсию яркости на основе применения следующих статистических формул:

$$\mu = \sum_{z_i \in S} z_i p(z_i), \quad \sigma^2 = \sum_{z_i \in S} (z_i - \mu)^2 p(z_i),$$

где z_i – значения яркости элементов части изображения S , $p(z_i)$ – соответствующие нормализованные значения гистограммы.

Вид гистограммы определяет, какая из существующих функций плотности распределения вероятностей является наиболее подходящей. Например, в том случае, если форма гистограммы близка к гауссовой функции плотности распределения вероятностей, то необходимо определить значения среднего и дисперсии. Для распределения других типов рассматриваются выражения для среднего и дисперсии как уравнения для параметров a и b , решение которых позволит найти параметры распределения.

4.4 Подавление шумов методами пространственной фильтрации

В том случае, если искажение изображения обусловлено исключительно наличием шума, то модель искаженного изображения можно представить в виде аддитивной суммы шума и исходного изображения:

$$g(x, y) = f(x, y) + \eta(x, y).$$

Или аналогично в частотной области:

$$G(u, v) = F(u, v) + N(u, v).$$

В общем случае слагаемое, описывающее шум – $\eta(x, y)$ или $N(u, v)$, неизвестно, поэтому его нельзя просто вычесть из функции $g(x, y)$ или $G(u, v)$. В случае периодического шума спектр $G(u, v)$ дает возможность оценить величину $N(u, v)$, в этом случае в целях построения приближения исходного изображения величина $N(u, v)$ может быть вычтена из функции $G(u, v)$.

В тех ситуациях, когда на изображении присутствует только аддитивный шум, пространственная фильтрация является лучшим из возможных методов восстановления.

Фильтр, основанный на вычислении среднего арифметического

Данный тип фильтра называется среднеарифметическим фильтром и является простейшим среди усредняющих фильтров. Пусть S_{xy} обозначает прямоугольную окрестность – множество ко-

ординат точек изображения размерами $m \times n$ с центром в точке (x, y) . Процедура фильтрации предполагает вычисление среднего арифметического значения искаженного изображения $g(x, y)$ по окрестности S_{xy} . Значение восстановленного изображения $\hat{f}(x, y)$ в произвольной точке (x, y) представляет собой среднее арифметическое значений в точках, принадлежавших окрестности S_{xy} :

$$\hat{f}(x, y) = \frac{1}{mn} \sum_{(s,t) \in S_{xy}} g(s, t) \cdot$$

Данная операция может быть реализована в виде свертки с маской, все коэффициенты которой равны $1/m \cdot n$. Усредняющий фильтр сглаживает локальные вариации яркости на изображении, уменьшение шума происходит в результате этого сглаживания.

Фильтры, основанные на порядковых статистиках

Фильтры, основанные на порядковых статистиках, представляют собой пространственные фильтры, вычисление отклика которых требует предварительного упорядочивания или ранжирования значений пикселей, заключенных внутри обрабатываемой фильтром области. Отклик фильтра в любой точке определяется по результатам упорядочивания.

Медианные фильтры

Наиболее известным из фильтров, основанных на порядковых статистиках, является медианный фильтр. Действие этого фильтра состоит в замене значения в точке изображения на медиану значений яркости в окрестности этой точки:

$$\hat{f}(x, y) = \underset{(s,t) \in S_{xy}}{\text{med}} \{g(s, t)\}.$$

Медианные фильтры отлично справляются с задачей подавления некоторых видов случайных шумов и при этом обеспечивают меньшее размытие изображений по сравнению с линейными сглаживающими фильтрами того же размера. Медианные фильтры особенно эффективны при наличии как биполярного, так и униполярного импульсного шума.

Фильтры, основанные на выборе максимального и минимального значения

Медиана выборки представляет собой 50-й перцентиль упорядоченного набора чисел, однако использование других статистических характеристик предоставляет ряд дополнительных возможностей подавления шумов на изображении.

Использование 100-го перцентиля приводит к фильтру, основанному на выборе максимального значения, или фильтру максимума, который задается выражением:

$$\hat{f}(x, y) = \max_{(s,t) \in S_{xy}} \{g(s, t)\}.$$

Такой фильтр полезен при обнаружении наиболее ярких точек на изображении. Применение такого фильтра приводит к уменьшению «черного» импульсного шума, так как в процессе фильтрации из окрестности S_{xy} выбирается максимальное значение.

Использование 0-го перцентиля приводит к фильтру, основанному на выборе минимального значения, или фильтру минимума:

$$\hat{f}(x, y) = \min_{(s,t) \in S_{xy}} \{g(s, t)\}.$$

Данный фильтр полезен при обнаружении наиболее темных точек на изображении. Применение фильтра минимума приводит к уменьшению униполярного «белого» импульсного шума вследствие операции выбора минимума.

Фильтр срединной точки

Применение фильтра срединной точки заключается в вычислении среднего между максимальными и минимальными значениями в соответствующей окрестности:

$$\hat{f}(x, y) = \frac{1}{2} \left[\max_{(s,t) \in S_{xy}} \{g(s, t)\} + \min_{(s,t) \in S_{xy}} \{g(s, t)\} \right].$$

Такой фильтр объединяет в себе методы порядковых статистики и арифметического усреднения. Фильтрация подобного типа наиболее эффективна при наличии случайно распределенных шумов, таких как гауссов или равномерный.

4.5 Инверсная фильтрация

Задача восстановления изображения значительно упрощается, если известна или определена с помощью ранее описанных методов искажающая функция (ядро или оператор H). Простейшим способом восстановления изображения является инверсная фильтрация, которая предполагает получение оценки $\hat{F}(u, v)$ Фурье-преобразования исходного изображения делением Фурье-преобразования искаженного изображения на частотное представление искажающей функции (деление подразумевается как поэлементная операция):

$$\hat{F}(u, v) = \frac{G(u, v)}{H(u, v)}.$$

Введем величину $N(u, v)$, определяемую как Фурье-преобразование функции присутствующего на изображении шума $\eta(x, y)$, и справедливым является следующее выражение:

$$\hat{F}(u, v) = F(u, v) + \frac{N(u, v)}{H(u, v)}.$$

Данное выражение показывает, что, обладая информацией исключительно об искажающей функции, невозможно восстановить неискаженное исходное изображение (с помощью операции обратного Фурье-преобразования функции $F(u, v)$), так как функция $N(u, v)$ представляет собой Фурье-преобразование случайной величины и является неизвестной. Как правило, значения искажающей функции становятся пренебрежимо малыми на больших частотах, в силу этого функция $H(u, v)$ может принимать нулевые или близкие к нулевым значения, поэтому вклад второго слагаемого становится доминирующим, что дополнительно затрудняет процесс восстановления изображения.

Один из способов устранения указанных трудностей состоит в том, чтобы ограничить частоты фильтра значениями вблизи начала координат, так как полагается, что функция $H(u, v)$ обращается в нуль вне некоторой области вблизи начала координат.

Для реализации этого способа достаточно умножить правую часть указанного выше равенства на передаточную функцию некоторого идеального низкочастотного фильтра. Как известно, значение

$H(0, 0)$ равно среднему значению функции $h(x, y)$ и обычно является наибольшим значением $H(u, v)$ в частотной области, поэтому, ограничиваясь рассмотрением частот вблизи начала координат, уменьшается вероятность встретить нулевое значение.

Инверсная фильтрация обладает очевидным недостатком – отсутствие информации о статистических свойствах присутствующих на изображении шумов при восстановлении, что делает данный метод слабо применимым на практике.

4.6 Винеровская фильтрация

Рассмотренный ранее метод инверсной фильтрации не обеспечивает корректной работы по отношению к шуму. Одним из наиболее эффективных методов восстановления изображения, основанных одновременно на учете свойств искажающей функции и статистических свойств шума в процессе восстановления, является винеровская фильтрация или фильтрация методом минимизации среднеквадратического отклонения.

Данный метод основан на рассмотрении изображений и шума как случайных процессов, и задача формулируется следующим образом: необходимо найти такую оценку \hat{f} для неискаженного изображения f , чтобы среднеквадратическое отклонение этих величин друг от друга было минимальным. Среднеквадратическое отклонение e задается формулой:

$$e^2 = E\{(f - \hat{f})^2\},$$

где $E\{\cdot\}$ обозначает математическое ожидание своего аргумента.

Предполагается, что выполнены следующие условия:

- 1) шум и неискаженное изображение не коррелированы между собой;
- 2) либо шум, либо неискаженное изображение имеют нулевое среднее значение;
- 3) оценка линейно зависит от искаженного изображения.

При выполнении этих условий минимум среднеквадратического отклонения достигается для оценивающей исходное изображение функции, заданной в частотной области, следующим образом:

$$\hat{F}(u, v) = \left(\frac{H^*(u, v) S_f(u, v)}{S_f(u, v) |H(u, v)|^2 + S_\eta(u, v)} \right) G(u, v) = \left(\frac{1}{H(u, v)} \cdot \frac{|H(u, v)|^2}{|H(u, v)|^2 + S_\eta(u, v) / S_f(u, v)} \right) G(u, v)$$

где $G(u, v)$ – Фурье-преобразование искаженного изображения; $H(u, v)$ – частотное представление искажающей функции, $H^*(u, v)$ – комплексное сопряжение $H(u, v)$; $|H(u, v)|^2 = H^*(u, v) \cdot H(u, v)$ – энергетический спектр искажающей функции; $S_\eta(u, v) = |N(u, v)|^2$ – энергетический спектр шума; $S_f(u, v) = |F(u, v)|^2$ – энергетический спектр неискаженного изображения.

Приведенный результат был впервые получен Н. Винером, и метод известен как оптимальная фильтрация по Винеру. Фильтр, представленный передаточной функцией внутри скобок, называется винеровским фильтром. Анализируя приведенное выражение, необходимо заметить, что для винеровского фильтра отсутствует проблема обращения в нуль спектра искажающей функции, за исключением того случая, когда функции $H(u, v)$ и $S_\eta(u, v)$ обращаются в некоторых точках в нуль одновременно.

Восстановление изображения в пространственной области достигается применением обратного преобразования Фурье к оценке $\hat{F}(u, v)$. Отметим, что если шум равен нулю, то и его энергетический спектр обращается в нуль, и в этом случае винеровская фильтрация сводится к инверсной фильтрации.

В том случае, если на изображении присутствует белый шум, спектр которого $|N(u, v)|^2$ является постоянной функцией, то выражение для винеровского фильтра упрощается. Однако в большинстве случаев спектр неискаженного изображения редко бывает известен. В тех случаях, когда спектры шума и неискаженного изображения неизвестны и не могут быть оценены, часто используется подход, состоящий в аппроксимации исходного выражения следующим выражением:

$$\hat{F}(u, v) = \left(\frac{1}{H(u, v)} \cdot \frac{|H(u, v)|^2}{|H(u, v)|^2 + K} \right) G(u, v),$$

где K – определенная константа.

Винеровская фильтрация обладает несомненным преимуществом по сравнению с инверсной фильтрацией и позволяет получить результаты восстановления изображения очень близкие по виду к исходному изображению.

4.7 Восстановление изображений в среде MATLAB

В системе MATLAB для моделирования искажения изображения некоторым шумом есть функция **imnoise**, имеющая следующий синтаксис:

g=imnoise(f, 'type', parameters)

f – исходное изображение, **g** – зашумленное изображение, **type** – тип шума, **parameters** – параметры используемого шума. Для получения более подробной информации об использовании данной функции при искажении изображения различными типами шумов используйте справочную систему MATLAB с помощью формирования следующей команды: **doc imnoise**.

Перед заданием параметров шума необходимо учитывать, что функция **imnoise** преобразует исходное изображение к классу **double** в диапазоне значений $[0, 1]$. Например, для того чтобы добавить шум со средним значением 64 и дисперсией 400 к изображению класса **uint8** (8-битное изображение), необходимо преобразовать среднее значение к $64/256$, а дисперсию – к $400/(256)^2$.

Для оценивания параметров шума используется функция **statmoments**, которая вычисляет среднее и статистические моменты до порядка **n** включительно:

[v, unv]=statmoments(p, n)

p – вектор гистограммы изображения, **n** – порядок статистического момента, **v** – выходной вектор нормированных моментов, вычисленный на основе приведения случайных величин к диапазону $[0, 1]$, **unv** – выходной вектор моментов в диапазоне исходных данных.

Для получения маски сглаживающего пространственного фильтра, реализованного на выполнении операции вычисления среднего арифметического, используется функция **fspecial**:

w=fspecial('average', [m, n])

w – маска сглаживающего пространственного фильтра, **m** и **n** – число строк и столбцов окрестности фильтрации соответственно.

Для реализации процедуры медианной фильтрации используется функция **medfilt2**, имеющая следующий синтаксис:

f=medfilt2(g, [m, n])

g – входное зашумленное изображение, **m** и **n** – число строк и столбцов окрестности фильтрации соответственно, **f** – выходное отфильтрованное изображение.

Для реализации фильтров, основанных на выборе максимального и минимального значения, используются следующие функции соответственно:

f=ordfilt2(g, m*n, ones(m, n))

f=ordfilt2(g, 1, ones(m, n))

Для реализации винеровской фильтрации изображений в среде MATLAB используется функция **deconvwnr**, имеющая несколько форм записей:

fr=deconvwnr(g, PSF)

fr – восстановленное изображение, **g** – искаженное изображение, **PSF** – искажающий оператор (искажающая функция).

В данном случае предполагается, что соотношение сигнал/шум равно нулю, в этом случае винеровская фильтрация сводится к инверсной фильтрации.

Следующая форма записи предполагает, что соотношение сигнал/шум известно:

fr=deconvwnr(g, PSF, NSPR)

В случае, если известны автокорреляционные функции шума (**NACORR**) и неискаженного изображения (**FACORR**), используется следующая форма записи:

fr=deconvwnr(g, PSF, NACORR, FACORR)

Согласно теореме Винера-Хинчина для вычисления автокорреляционной функции изображения (или шума) необходимо выполнить операцию обратного преобразования Фурье от энергетического спектра изображения с помощью функции **ifft2**.

Для нахождения энергетического спектра изображения необходимо выполнить прямое преобразование Фурье от исходного изображения с помощью функции **fft2**, вычислить его модуль с помощью команды **abs** и возвести модуль в квадрат.

Для определения соотношения сигнал/шум (**NSPR**) необходимо вычислить отношение значения средней энергии изображения к значению средней энергии шума.

Для моделирования размытия изображения в среде MATLAB можно использовать функцию **fspecial** со следующим синтаксисом:

PSF=fspecial('motion', len, theta)

Данная форма вызова функции **fspecial** возвращает **PSF**, которая аппроксимирует эффекты линейного перемещения камеры на **len** пикселей, угловой параметр **theta** измеряется в градусах и отсчитывается от положительной горизонтальной полуоси против часовой стрелки.

Для построения изображенного, искаженного оператором **PSF**, используется функция **imfilter** следующим образом:

g=imfilter(f, PSF, 'circular')

f – исходное неискаженное изображение, **g** – выходное размытое изображение, **PSF** – искажающий оператор (или маска), параметр **'circular'** используется для подавления граничных эффектов.

4.8 Практическое задание № 4

1. Загрузите исходное изображение черепа человека **scull.jpg**. Сформируйте искаженные изображения черепа человека добавлением модельных аддитивных шумов следующего типа: гауссов шум, гамма-шум и импульсный шум. Постройте гистограммы исходного изображения и сгенерированных искаженных изображений, сравните полученные гистограммы и сделайте выводы.

2. Восстановите изображение черепа человека, искаженного гауссовым шумом, с помощью следующих видов фильтрации: усредняющий фильтр (на основе вычисления среднеарифметического), медианный фильтр, фильтр срединной точки. Сравните полученные результаты, сделайте выводы.

3. Восстановите изображение черепа человека, искаженного импульсным шумом, с помощью следующих видов фильтрации: фильтр максимума, фильтр минимума, медианный фильтр, фильтр срединной точки, усредняющий фильтр (на основе вычисления среднеарифметического).

4. Получите размытое изображение черепа человека, искаженного гауссовым шумом.

5. С помощью методов винеровской фильтрации восстановите изображение, полученное в 4-м пункте. Получите результаты восстановления изображения, используя каждый раз различную априорную информацию: соотношение сигнал/шум, информацию об искажающей функции; информацию о значении автокорреляционной функции шума и неискаженного изображения. Сравните полученные результаты и сделайте выводы.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Модель процесса искажения/восстановления изображения. Модели шумов, присутствующих на изображении.

2. Построение оценок для параметров шума.

3. Подавление шума с помощью методов пространственной фильтрации.

4. Линейные усредняющие фильтры.

5. Фильтры, основанные на порядковых статистиках (ранговые фильтры).

6. Инверсная фильтрация.

7. Винеровская фильтрация.

КОНТРОЛЬНЫЕ ВОПРОСЫ ДЛЯ САМОСТОЯТЕЛЬНОЙ АТТЕСТАЦИИ

В приведенных ниже вопросах выберите один или несколько правильных вариантов ответа из списка предложенных.

1. Модель цифрового изображения может быть представлена:

- а) двумерной функцией от координат x и y ;
- б) одномерной функцией от координаты;
- в) трехмерной функцией от координат x , y и z .

2. Процедурой дискретизацией изображения называется:

- а) представление амплитуды (яркости) изображения дискретными значениями из конечного множества;
- б) нахождение промежуточных значений величины яркости пикселей по имеющемуся дискретному набору известных значений;
- в) представление координат изображения в виде конечного множества отсчетов.

3. Процедурой квантования изображения называется:

- а) представление амплитуды (яркости) изображения дискретными значениями из конечного множества;
- б) нахождение промежуточных значений величины яркости пикселей по имеющемуся дискретному набору известных значений;
- в) представление координат изображения в виде конечного множества отсчетов.

4. Цифровое изображение представляет собой:

- а) одномерный массив значений яркости пикселей изображения;
- б) двумерный массив значений яркости пикселей изображения;
- в) двумерный массив значений координат пикселей изображения.

5. Какое количество памяти запоминающего устройства необходимо для хранения 8-битного изображения размером 256×256 пикселей:

- а) 64 Кбайт;
- б) 128 Кбайт;
- в) 256 Кбайт.

6. Какими критериями определяется увеличение детализации изображения:

- а) только увеличением пространственного разрешения;
- б) только увеличением яркостного разрешения;
- в) увеличением пространственного и яркостного разрешения.

7. По какой формуле определяется евклидово расстояние D между двумя пикселями $p(x, y)$ и $q(s, t)$ цифрового изображения:

- а) $D = \sqrt{(x - s)^2 + (y - t)^2}$;
- б) $D = |x - s| + |y - t|$;
- в) $D = \max(|x - s|, |y - t|)$.

8. При каком значении γ степенное преобразование следующего вида обеспечивает уменьшение яркости изображения $s = c \cdot r^\gamma$, где c – константа, r – значения яркости исходного изображения, s – значение яркости преобразованного изображения:

- а) $\gamma = 0$;
- б) $\gamma > 1$;
- в) $\gamma > 0$.

9. Какое из указанных ниже выражений обеспечивает преобразование исходного изображения с яркостью пикселей в диапазоне $[0; L-1]$ в негативное изображение (r – значение яркости исходного изображения, s – значение яркости преобразованного изображения):

- а) $s = L - 1 - r$;
- б) $s = L \cdot \log(1 + r)$;
- в) $s = r - L - 1$.

10. Гистограмма цифрового изображения представляет собой графическое представление дискретной функции зависимости:

- а) числа пикселей изображения с данным значением яркости от значений дискретных интервалов диапазона яркости;
- б) значения яркости пикселей изображения от количества пикселей изображения с данным значением яркости;
- в) значения яркости пикселя изображения от его порядкового номера.

11. Равномерная гистограмма распределения значений яркости пикселей цифрового изображения характерна для:

- а) темного изображения;
- б) яркого изображения;
- в) высококонтрастного изображения.

12. Процедура эквализации гистограммы изображения обеспечивает:

- а) увеличение яркости исходного изображения;
- б) усиление контраста исходного изображения;
- в) уменьшение яркости исходного изображения.

13. Основной целью усреднения серии изображений одного и того же объекта является:

- а) усиление контраста изображения;
- б) увеличение яркости изображения;
- в) уменьшение зашумленности изображения.

14. Применение усредняющего фильтра к цифровому изображению приводит к следующему:

- а) усилению контрастности изображения;
- б) повышению резкости изображения;
- в) расфокусировке контуров изображения.

15. Какой из приведенных ниже фильтров обеспечивает сглаживание цифрового изображения:

- а) медианный фильтр;
- б) фильтр на основе первой производной;
- в) фильтр на основе второй производной.

16. Каким преимуществом обладает сглаживающий фильтр взвешенного среднего перед однородным усредняющим фильтром:

- а) уменьшение зашумленности изображения;
- б) уменьшение расфокусировки изображения при сглаживании;
- в) усиление контраста изображения.

17. Какой эффект обработки изображения обеспечивает фильтр на основе применения второй производной (оператор Лапласа):

- а) усиление контрастности изображения;
- б) повышение резкости изображения;
- в) уменьшение зашумленности изображения.

18. Какой из приведенных ниже фильтров приводит к наибольшему усилению присутствующих на изображении шумов:

- а) медианный фильтр;
- б) градиентный фильтр или фильтр на основе первой производной;
- в) фильтр на основе второй производной.

19. Суть операции нерезкого маскирования изображения заключается в следующем:

- а) вычитании из исходного изображения его расфокусированной копии;
- б) сложении исходного изображения и его расфокусированной копии;
- в) сложении исходного изображения и негативного изображения, полученного из исходного.

20. Для каких из указанных ниже частотных фильтров не характерен эффект Гиббса – появление ложных контуров после фильтрации изображения:

- а) идеальный фильтр;
- б) гауссов фильтр;
- в) фильтр Баттерворта.

21. Какой эффект обработки изображения обеспечивает фильтрация изображения с помощью фильтров высоких частот:

- а) уменьшение зашумленности изображения;
- б) повышение резкости изображения;
- в) появление размытости изображения.

22. Какой эффект обработки изображения обеспечивает фильтрация изображения с помощью фильтров низких частот:

- а) повышение резкости изображения;
- б) усиление контрастности изображения;
- в) сглаживание изображения и уменьшение зашумленности изображения.

23. Результат применения гомоморфной фильтрации изображений заключается в следующем:

- а) одновременном сжатии динамического диапазона и усилении контраста;
- б) одновременном усилении контраста и уменьшении зашумленности изображения;
- в) только сжатии динамического диапазона изображения.

24. Наиболее эффективным фильтром для подавления импульсных шумов является:

- а) медианный фильтр;
- б) среднеарифметический усредняющий фильтр;
- в) режекторный фильтр.

25. Для устранения периодических шумов на изображении наиболее эффективными являются:

- а) полосовые или режекторные фильтры;
- б) медианные фильтры;
- в) среднегеометрические и среднеарифметические фильтры.

ЗАКЛЮЧЕНИЕ

В данном учебном пособии были рассмотрены основные методы цифровой обработки биомедицинских изображений в среде компьютерных вычислений MATLAB. Учебное пособие содержит справочные сведения о пакете MATLAB, подробный анализ существующих методик пространственной и частотной фильтрации изображений, а также обзор актуальных методов восстановления изображений.

Материалы данного учебного пособия являются введением в теорию и методы компьютерной обработки биомедицинских изображений и дают основы для последующего углубленного изучения проблематики обработки биомедицинских изображений, а также проведения самостоятельных научных исследований в данной области.

Учебное пособие будет полезно широкому кругу читателей: профессионалам в области анализа и обработки биомедицинских изображений, студентам и аспирантам, специализирующимся в области биомедицинской инженерии и прикладной математики, а также всем интересующимся вопросами современных методов компьютерной обработки биомедицинских изображений.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Гонсалес, Р. Цифровая обработка изображений [Текст] / Р. Гонсалес, Р. Вудс. – М.: Техносфера, 2006. – 1072 с.
2. Rangayyan, R.M. Biomedical Image Analysis [Текст] / R.M. Rangayyan. – CRC Press, Boca Raton, 2005. – 1306 p.
3. Физика визуализации изображений в медицине [Текст]: в 2-х томах. Т.1 / под ред. С. Уэбба; пер. с англ. – М.: Мир, 1991. – 408с.
4. Методы компьютерной обработки изображений [Текст] / под ред. В. А. Сойфера. – 2-е изд. – М.: Физматлит, 2003. – 780 с.
5. Гонсалес, Р. Цифровая обработка изображений в среде Matlab [Текст] / Р. Гонсалес, Р. Вудс, С. Эддинс. – М.: Техносфера, 2006. – 616 с.
6. Russ, J.C. The image processing handbook [Текст] / J.C. Russ. – CRC Press, Boca Raton, 1999. – 885 p.
7. Введение в цифровую обработку сигналов и изображений: математические модели изображений [Текст]: учеб. пособие / В. А. Сойфер [и др.]; Федер. агентство по образованию, Самар. гос. аэрокосм. ун-т им. С. П. Королева. – Самара: Изд-во СГАУ, 2006. – 179 с.

Учебное издание

*Федотов Александр Александрович,
Акулов Сергей Анатольевич,
Акулова Анна Сергеевна*

**МЕТОДЫ КОМПЬЮТЕРНОЙ ОБРАБОТКИ
БИОМЕДИЦИНСКИХ ИЗОБРАЖЕНИЙ
В СРЕДЕ MATLAB**

Учебное пособие

Редактор Ю.Н. Литвинова
Доверстка: Е.С. Кочеулова

Подписано в печать 16.03.2015. Формат 60×84 1/16.
Бумага офсетная. Печать офсетная. Печ. л. 5,75.
Тираж 100 экз. Заказ . Арт. - 15/2015.

федеральное государственное автономное
образовательное учреждение высшего образования
«Самарский государственный аэрокосмический
университет имени академика С. П. Королева
(национальный исследовательский университет)» (СГАУ)
443086, Самара, Московское шоссе, 34.

Изд-во СГАУ. 443086, Самара, Московское шоссе, 34.

